Implicit ODE Solvers with Good Local Error Control for the Transient Analysis of Markov Models

Víctor Suñé* Juan Antonio Carrasco†

September 14, 2016

Abstract

Obtaining the transient probability distribution vector of a continuous-time Markov chain (CTMC) using an implicit ordinary differential equation (ODE) solver tends to be advantageous in terms of run-time computational cost when the product of the maximum output rate of the CTMC and the largest time of interest is large. In this paper, we show that when applied to the transient analysis of CTMCs, many implicit ODE solvers are such that the linear systems involved in their steps can be solved by using iterative methods with strict control of the 1-norm of the error. This allows the development of implementations of those ODE solvers for the transient analysis of CTMCs that can be more efficient and more accurate than more standard implementations.

Keywords
Implicit ODE solvers; continuous-time Markov chains; transient analysis.

1 Introduction

Consider a finite continuous-time Markov chain (CTMC) $X = \{X(t); t \geq 0\}$ with infinitesimal generator. The state space of X will be denoted by $\Omega = \{1, \ldots, m\}$, its initial probability distribution (column) vector by $\mathbf{p}^0$, and the transpose of its infinitesimal generator by $\mathbf{Q} = (q_{i,j})_{1 \leq i,j \leq m}$, i.e., $q_{i,j}$, $i \neq j$, will denote the transition rate from state j to state i and $|q_{i,i}| = -q_{i,i} = \sum_{\substack{j=1 \\ j \neq i}}^{m} q_{j,i}$ will denote the output rate from state i. In this paper, we will be concerned with the computation of the transient probability distribution vector of X, $\mathbf{p}(t) = (P[X(t) = i])_{1 \leq i \leq m}$, $t \geq 0$, when $\mathbf{Q}$ is large and sparse.

When $\mathbf{Q}$ is large, the only practical methods to compute $\mathbf{p}(t)$ are uniformization (also called randomization) [1, 2], uniformization-based methods (see, e.g., [3], [4], [5]), the Krylov-based method described in [6], and formulating $\mathbf{p}(t)$ as the solution of an initial value problem (IVP) for the Kolmogorov system of ordinary differential equations (ODEs) and solving the IVP using an ODE solver. When $qt \gg 1$, where $q = \max_{1 \leq i \leq m} |q_{i,i}|$, this latter alternative can be very attractive from the point of view of run-time computational cost. The IVP to be solved is

$$\begin{cases} \dfrac{\mathrm{d}\mathbf{p}(t)}{\mathrm{d}t} = \mathbf{Q}\,\mathbf{p}(t), \quad t \geq 0\,, \\ \mathbf{p}(0) = \mathbf{p}^0\,. \end{cases} \tag{1}$$

That IVP is stable because, using the Gershgorin circle theorem [7], which states that the eigenvalues of $\mathbf{Q}^{\mathrm{T}}$, which are the same as those of $\mathbf{Q}$, lie in the union of the m discs in the complex plane with centers $q_{i,i} \leq 0$ and radii $\sum_{\substack{j=1 \\ j \neq i}}^{m} |q_{j,i}| = |q_{i,i}|$, it turns out that the eigenvalues of $\mathbf{Q}$ different

*Departament d'Enginyeria Electrònica, Universitat Politècnica de Catalunya. C. Colom, 1, 08222 Terrassa, Catalonia (Spain). Email address: victor.sunye@upc.edu.

†Departament d'Enginyeria Electrònica, Universitat Politècnica de Catalunya. Av. Diagonal, 647, planta 9, 08028 Barcelona, Catalonia (Spain). Email address: juan.a.carrasco@upc.edu.

from 0 have negative real part. As in [8], we will say that the IVP (1) is stiff if $qt \gg 1$. That such a stiffness criterion is reasonable is argued in [8] as follows. For each distinct eigenvalue λ_j of $\mathbf{Q}$, component i of $\mathbf{p}(t)$, $1 \leq i \leq m$, will include a factor of the form $\sum_{k=0}^{m_j-1} c_{i,j,k} t^k \mathrm{e}^{\lambda_j t}$, where m_j denotes the multiplicity of λ_j and the $c_{i,j,k}$ are appropriate constants. Consequently, when $\max_j |\lambda_j| t \gg 1$, we can expect $\mathbf{p}(t)$ to have components with large relative variation in the time interval $[0, t)$, making the IVP (1) stiff. But, again by the Gershgorin circle theorem, $\max_j |\lambda_j|$ is bounded from above by $\|\mathbf{Q}\|_1 = 2q$. Therefore, if $qt \gg 1$, it is reasonable to expect the IVP (1) to be stiff. For an ODE solver applied to the solution of the IVP (1) to be effective when the IVP is stiff, the ODE solver should be implicit and, preferably, A-stable [9]. In this paper, we will consider the use of implicit ODE solvers to solve the IVP (1).

Several papers have considered the use of implicit ODE solvers for solving the IVP (1) [8, 10, 11, 12, 13, 14, 15, 16]. In [8], the performance of TR-BDF2 [17], an L-stable [18] linear two-step implicit ODE solver, was analyzed. The conclusions were that the run-time computational cost of TR-BDF2 was lower than the run-time computational cost of both uniformization and the explicit ODE solver RKF45 [19], but that, being only a second-order method, it required a small step size to achieve high accuracies. In [10], TR-BDF2 and a third-order L-stable implicit Runge-Kutta ODE solver (IRK3) were compared between them and with several uniformization variants. The conclusions were that both TR-BDF2 and IRK3 were preferable to the uniformization variants and that when the required accuracy was high, the run-time computational cost of IRK3 was lower than that of TR-BDF2. The performances of TR-BDF2 and IRK3 were analyzed in [11] for the case in which X is nearly completely decomposable [20], with the conclusion that those implicit ODE solvers should be implemented exploiting the near complete decomposability of X. In [12], the performance of IRK3 was analyzed when X is acyclic so that $\mathbf{Q}$ can be put into lower triangular form and the linear system involved in each step of the method can be solved very efficiently using direct methods. The conclusions were that IRK3 could be highly accurate and that its run-time computational cost was lower than the run-time computational cost of a uniformization variant and lower than the run-time computational cost of an improved version of an specific method for the transient analysis of acyclic CTMCs [21]. In [13], it was proposed to combine RKF45 and either TR-BDF2 or IRK3 in such a way that RKF45 is used up to some intermediate time and the other method is used from that point on. The result was a significant reduction in run-time computational cost compared with any of the ODE solvers used in isolation. In [14], the performance of several ODE solvers was analyzed (implicit Euler [22], trapezoidal rule [22], 2-stage Radau IIA [18, 23], 2-stage Gauss [24], a singly diagonally implicit Runge-Kutta ODE solver of order two with two stages, and another one of order three with two stages) with the conclusion that the trapezoidal rule implemented using extrapolation, which gives a fourth-order implicit method, was a good alternative. In [15], it was proposed to combine uniformization and TR-BDF2 in such a way that uniformization is used up to some intermediate time and TR-BDF2 is used from that point on with significant reductions in run-time computational cost compared with TRB-BDF2 and uniformization. Finally, in [16] some of the ODE solvers in the VODPK package [25] were compared with uniformization and with the Krylov-based method described in [6], with the conclusion that for $\|\mathbf{Q}\|t > 500$, the Krylov-based method and a variant of the implicit ODE solver based on backward differentiation formulae (BDFs) with variable coefficients available in the package had a run-time computational cost lower than that of uniformization.

Applied to the IVP (1), an ODE solver produces approximations $\mathbf{p}_n$, $n = 0, 1, \ldots$, to $\mathbf{p}(t_n)$ at a set of time points t_n starting from $t_0 = 0$ and $\mathbf{p}_0 = \mathbf{p}^0$. Step $n \geq 1$ spans the solution from $t = t_{n-1}$ to $t = t_n$ and has step size $h_n = t_n - t_{n-1}$. If the ODE solver is implicit, computing the approximation $\mathbf{p}_n$, $n = 1, 2, \ldots$, requires solving one or more linear systems of equations with matrices related to $\mathbf{Q}$. In the case of the implicit ODE solvers considered in [8, 10, 11, 12, 13, 14, 15, 16], the linear systems were solved by using iterative methods including Gauss-Seidel (GS) [26], a mixture of Jacobi [26] and GS, successive overrelaxation (SOR) [26], and a variant of restarted GMRES [27]. The convergence of some of those methods was only analyzed in [14], where it was noted that the Jacobi method was guaranteed to converge for some of the implicit ODE solvers considered. At the best of the authors' knowledge, the impact of the errors with which the linear systems are solved has never been explicitly analyzed. It is clear,

however, that if those errors are large, they can affect adversely not only the accuracy but also the run-time computational cost of the ODE solvers. Roughly speaking, the reason is that those errors may introduce significant spurious components in the computed solution associated with eigenvalues of $\mathbf{Q}$ of large absolute value, which are the ones that will limit most the step size and, then, to reduce to an acceptable level the impact of those spurious components in the error of the computed solution, the ODE solver may be forced to take steps smaller than the ones it would take if those errors were absent. The errors introduced in the solution of the linear systems will be referred to as step approximation errors. To clarify, the local error at step n of the ODE solver, $\mathbf{e}_n = \mathbf{p}_n - \mathrm{e}^{\mathbf{Q}h_n}\mathbf{p}_{n-1}$, has three components. The first component, called here the inherent local error, reflects the inherent error of the approximation formula underlying the ODE solver and is $\mathbf{i}_n = \mathbf{p}_n^+ - \mathrm{e}^{\mathbf{Q}h_n}\mathbf{p}_{n-1}$, where $\mathbf{p}_n^+$ is $\mathbf{p}_n$ computed exactly. The second component, which is the one that we call step approximation error, reflects the error introduced by solving the linear systems using iterative and, hence, inexact methods. Formally, that error is $\mathbf{s}_n = \widehat{\mathbf{p}}_n - \mathbf{p}_n^+$, where $\widehat{\mathbf{p}}_n$ is $\mathbf{p}_n$ computed by solving the linear systems using iterative methods, ignoring the impact of round-off errors. The third component of the local error collects the impact of round-off errors on $\mathbf{p}_n$.

The 1-norm is a convenient norm for measuring and controlling local error components for the IVP (1). The reason is that $\left(\mathrm{e}^{\mathbf{Q}\tau}\right)^{\mathrm{T}} = \mathrm{e}^{\mathbf{Q}^{\mathrm{T}}\tau}$, $\tau \geq 0$, being a stochastic matrix, we have $\|\mathrm{e}^{\mathbf{Q}\tau}\|_1 = 1$, $\tau \geq 0$, implying that, measured in the 1-norm, local errors will not be amplified as they propagate through the solution to define the global error. Therefore, according to the discussion in the previous paragraph, it is convenient that $\|\mathbf{s}_n\|_1$, $n = 0, 1, \ldots$, be small enough. But, $\|\mathbf{s}_n\|_1$ will be determined by the stopping criterion of the iterative method with which the involved linear systems are solved and unless there exists strict control of $\|\mathbf{s}_n\|_1$, any reasonable implementation of the ODE solver will have to use heuristic stopping criteria trying to guarantee that $\|\mathbf{s}_n\|_1$ are indeed small enough. If such criteria are optimistic, in the sense that the actual $\|\mathbf{s}_n\|_1$ are larger than the intended ones, the ODE solver may be forced to take smaller steps than necessary, impacting adversely the run-time computational cost of the ODE solver. On the contrary, if the stopping criteria are pessimistic, the number of required iterations in the solution of the linear systems can be larger than necessary, again impacting adversely the run-time computational cost of the ODE solver. In conclusion, having strict control of the step approximation errors can be convenient from the point of view of both the quality of control of the error and the run-time computational cost of the ODE solver.

In this paper, we show that when applied to the IVP (1), many implicit ODE solvers are such that, using a known upper bound for the ∞-norm of the inverse of a strictly diagonally dominant matrix, the properties of the matrix $\mathbf{Q}$ allow to adapt some classes of iterative methods to solve the linear systems with strict control of the 1-norm of the step approximation error. Using some of the adapted iterative methods, we will describe an implementation with strict control of the 1-norm of the step approximation error of an implicit ODE solver based on BDFs with variable coefficients and will show, using numerical experiments, that that implementation can be more efficient and more accurate than a reasonable standard implementation.

The rest of the paper is organized as follows. In Section 2, we show how two classes of iterative methods to solve linear systems can be modified to provide strict error control when applied to linear systems with strictly diagonally dominant matrices. In Section 3, we identify three classes of implicit ODE solvers that when applied to the IVP (1), the properties of the matrix $\mathbf{Q}$ allow to solve the linear systems involved in each step by using iterative methods with the modifications discussed in Section 2 with strict control of the 1-norm of the step approximation error. In Section 4, we review an implicit ODE solver based on BDFs with variable coefficients when applied to the IVP (1) and describe two implementations of that ODE solver: An implementation that provides strict control of the 1-norm of the step approximation error and a reasonable standard implementation that does not provide such a control. In Section 5, we compare the performance of the implementation of VBDF with strict control of the 1-norm of the step approximation error with that of the standard implementation. Finally, Section 6 presents the conclusions. The online supplement collects the mathematical proofs of some theoretical results used in the paper.

2 Strictly Diagonally Dominant Matrices and Iterative Methods

Assume we want to solve the linear system

$$\mathbf{V}\mathbf{x} = \mathbf{u}\,, \tag{2}$$

where $\mathbf{V} = (v_{i,j})_{1 \leq i,j \leq n}$ is a real or complex nonsingular matrix of dimension n. This section shows how iterative methods that produce the residual after each iteration or after a predefined number of iterations and splitting-based iterative methods can be modified to provide strict control of the error of the solution of (2), ignoring the impact of round-off errors.

Iterative methods producing the residual after each iteration or after a predefined number of iterations include well-known Krylov subspace methods such as restarted GMRES, CGS [28], the implementation of QMR described in [29], and Bi-CGSTAB [30]. Splitting-based iterative methods include GS, Jacobi, and SOR. In all those iterative methods, iterates $\mathbf{x}^{(l)}$, $l = 0, 1, 2, \ldots$ are obtained which should converge to the solution of the linear system.

We will say that $\mathbf{V}$ is strictly row diagonally dominant (SRDD) if $|v_{i,i}| > \sum_{\substack{j=1 \\ j \neq i}}^{n} |v_{i,j}|$, $1 \leq i \leq n$. This collects the usual notion of a strictly diagonally dominant matrix. Similarly, $\mathbf{V}$ will be said to be strictly column diagonally dominant (SCDD) if $|v_{i,i}| > \sum_{\substack{j=1 \\ j \neq i}}^{n} |v_{j,i}|$, $1 \leq i \leq n$. Note that, using the Gershgorin circle theorem [7] and the fact that a matrix and its transpose have the same eigenvalues, both SRDD and SCDD matrices are nonsingular, since the union of the corresponding Gershgorin circles does not include the origin.

The following proposition collects several results that can be easily obtained from a well-known result on the ∞-norm of the inverse of an SRDD matrix [31, 32].

Proposition 1. *Consider the linear system* (2) *and let* $\mathbf{y}$ *be an arbitrary vector of dimension* n*. Then:*

1. *If* $\mathbf{V}$ *is SRDD,*
$$\|\mathbf{V}^{-1}\|_\infty \leq \frac{1}{\min_{1 \leq i \leq n} \{|v_{i,i}| - \sum_{\substack{j=1 \\ j \neq i}}^{n} |v_{i,j}|\}}\,.$$
2. *If* $\mathbf{V}$ *is SCDD,*
$$\|\mathbf{V}^{-1}\|_1 \leq \frac{1}{\min_{1 \leq i \leq n} \{|v_{i,i}| - \sum_{\substack{j=1 \\ j \neq i}}^{n} |v_{j,i}|\}}\,.$$
3. *If* $\mathbf{V}$ *is SRDD,*
$$\|\mathbf{x} - \mathbf{y}\|_\infty \leq \frac{\|\mathbf{u} - \mathbf{V}\mathbf{y}\|_\infty}{\min_{1 \leq i \leq n} \{|v_{i,i}| - \sum_{\substack{j=1 \\ j \neq i}}^{n} |v_{i,j}|\}}\,.$$
4. *If* $\mathbf{V}$ *is SCDD,*
$$\|\mathbf{x} - \mathbf{y}\|_1 \leq \frac{\|\mathbf{u} - \mathbf{V}\mathbf{y}\|_1}{\min_{1 \leq i \leq n} \{|v_{i,i}| - \sum_{\substack{j=1 \\ j \neq i}}^{n} |v_{j,i}|\}}\,.$$

Proof. See the online supplement. □

Consider the application to the solution of (2) of iterative methods that produce the residual, $\mathbf{u} - \mathbf{V}\mathbf{x}^{(l)}$, after each iteration or after a predefined number of iterations. The following theorem, which is an almost direct consequence of Proposition 1, specifies when to stop the iterations to provide strict control of the ∞-norm or the 1-norm of the error.

Theorem 1. *Let* $\delta > 0$ *and consider the application to the solution of* (2) *of an iterative method that produces the residual after each iteration or after a predefined number of iterations. If the*

matrix $\mathbf{V}$ *is SRDD, to guarantee, if the method is successful,* $\|\mathbf{x}-\mathbf{x}^{(l)}\|_\infty \leq \delta$, $l \geq 0$, *it suffices to stop the method when the residual satisfies*

$$\|\mathbf{u}-\mathbf{V}\mathbf{x}^{(l)}\|_\infty \leq \delta \min_{1\leq i\leq n}\left\{|v_{i,i}| - \sum_{\substack{j=1\\ j\neq i}}^{n}|v_{i,j}|\right\},$$

and, if the matrix $\mathbf{V}$ *is SCDD, to guarantee, if the method is successful,* $\|\mathbf{x}-\mathbf{x}^{(l)}\|_1 \leq \delta$, $l \geq 0$, *it suffices to stop the method when the residual satisfies*

$$\|\mathbf{u}-\mathbf{V}\mathbf{x}^{(l)}\|_1 \leq \delta \min_{1\leq i\leq n}\left\{|v_{i,i}| - \sum_{\substack{j=1\\ j\neq i}}^{n}|v_{j,i}|\right\}.$$

Proof. See the online supplement. □

Often, restarted GMRES, CGS, QMR, and Bi-CGSTAB use preconditioning to speed up their convergence. Preconditioning means replacing the linear system (2) by the equivalent one

$$\mathbf{M}_1^{-1}\mathbf{V}\mathbf{M}_2^{-1}\mathbf{y} = \mathbf{M}_1^{-1}\mathbf{u}, \quad \mathbf{y} = \mathbf{M}_2\mathbf{x},$$

where $\mathbf{M}_1$ and $\mathbf{M}_2$ are the left- and right-preconditioner. In practice, it is not necessary to use both preconditioners. Theorem 1 also holds for the particular case of right-preconditioned methods, i.e., when $\mathbf{M}_1$ is an identity matrix, which still produce the true residual.

Splitting-based iterative methods are based on an splitting of $\mathbf{V}$, $\mathbf{V} = \mathbf{M} - \mathbf{N}$, and are defined by the recurrence

$$\mathbf{x}^{(l+1)} = \mathbf{M}^{-1}\mathbf{N}\mathbf{x}^{(l)} + \mathbf{M}^{-1}\mathbf{u}. \tag{3}$$

With the decomposition $\mathbf{V} = \mathbf{D} + \mathbf{L} + \mathbf{U}$, where $\mathbf{D}$ is the diagonal of $\mathbf{V}$, $\mathbf{L}$ is the strict lower part of $\mathbf{V}$, and $\mathbf{U}$ is the strict upper part of $\mathbf{V}$, Jacobi is obtained for $\mathbf{M} = \mathbf{D}$ and $\mathbf{N} = -(\mathbf{L}+\mathbf{U})$, GS for $\mathbf{M} = \mathbf{D} + \mathbf{L}$ and $\mathbf{N} = -\mathbf{U}$, and SOR with relaxation parameter ω for $\mathbf{M} = \mathbf{D}/\omega + \mathbf{L}$ and $\mathbf{N} = ((1-\omega)/\omega)\mathbf{D} - \mathbf{U}$. Strict diagonal dominance by rows of the matrix $\mathbf{V}$ ensures convergence of Jacobi and convergence of SOR for $0 < \omega < 2/(1+\rho_{\mathrm{J}})$, where ρ_{J} denotes the spectral radius of the Jacobi iteration matrix $-\mathbf{D}^{-1}(\mathbf{L}+\mathbf{U})$ [33]. From [34], strict diagonal dominance by columns of the matrix $\mathbf{V}$ also ensures convergence of Jacobi. Finally, strict diagonal dominance by rows or by columns of the matrix $\mathbf{V}$ ensures convergence of GS [34].

Consider the application to the solution of (2) of splitting-based iterative methods. The following theorem specifies when to stop the iterations to provide strict control of the ∞-norm or the 1-norm of the error.

Theorem 2. *Let* $\delta > 0$ *and consider the application to the solution of* (2) *of an splitting-based method defined by* (3). *If the matrix* $\mathbf{V}$ *is SRDD, to guarantee, if the method is successful,* $\|\mathbf{x}-\mathbf{x}^{(l)}\|_\infty \leq \delta$, $l \geq 1$, *it suffices to stop the method when*

$$\|\mathbf{x}^{(l)}-\mathbf{x}^{(l-1)}\|_\infty \leq \delta\,\frac{\min_{1\leq i\leq n}\{|v_{i,i}| - \sum_{\substack{j=1\\ j\neq i}}^{n}|v_{i,j}|\}}{\|\mathbf{N}\|_\infty},$$

and, if the matrix $\mathbf{V}$ *is SCDD, to guarantee, if the method is successful,* $\|\mathbf{x}-\mathbf{x}^{(l)}\|_1 \leq \delta$, $l \geq 1$, *it suffices to stop the method when*

$$\|\mathbf{x}^{(l)}-\mathbf{x}^{(l-1)}\|_1 \leq \delta\,\frac{\min_{1\leq i\leq n}\{|v_{i,i}| - \sum_{\substack{j=1\\ j\neq i}}^{n}|v_{j,i}|\}}{\|\mathbf{N}\|_1}.$$

Proof. See the online supplement. □

3 Strict Control of the Step Approximation Error

This section will identify three classes of implicit ODE solvers such that when applied to the IVP (1), ignoring the impact of round-off errors, the properties of the matrix $\mathbf{Q}$ allow to solve the linear systems involved in each step by using iterative methods with strict control of the 1-norm of the step approximation error.

Before identifying the classes, we need to describe briefly implicit Runge-Kutta (IRK) ODE solvers and linearly implicit Runge-Kutta (LIRK) ODE solvers (also called Rosenbrock ODE solvers). A step of an s-stage IRK ODE solver applied to the IVP (1) is defined by

$$\mathbf{p}_n = \mathbf{p}_{n-1} + h_n \sum_{j=1}^{s} b_j \mathbf{Q}\, \mathbf{g}_j, \quad n = 1, 2, \dots ,$$

where the stage vectors $\mathbf{g}_i$, $k = 1, \dots, s$, are the solution to the $m \times s$-dimensional linear system

$$\mathbf{g}_i = \mathbf{p}_{n-1} + \mathbf{p}_{n-1} + h_n \sum_{j=1}^{s} a_{i,j} \mathbf{Q}\, \mathbf{g}_j, \quad i = 1, \dots, s\,.$$

The real quantities $a_{i,j}$, b_j, $1 \le i, j \le s$, are coefficients of the IRK ODE solver and at least one coefficient $a_{i,j}$, $j \ge i$, is nonzero. Let $\mathbf{A} = (a_{i,j})_{1 \le i,j \le s}$ and $\mathbf{b} = (b_i)_{1 \le i \le s}$. Also, let $\otimes$ denote the Kronecker product, let $\mathbf{1}_k$ denote an all-ones vector of dimension k, and let $\mathbf{I}_n$ denote an identity matrix of dimension n. In practice [22], it is more convenient to introduce the vectors

$$\mathbf{z}_i = \mathbf{g}_i - \mathbf{p}_{n-1}, \quad 1 \le i \le s\,,$$

and $\mathbf{z} = ({\mathbf{z}_1}^{\mathrm{T}}, \dots, {\mathbf{z}_s}^{\mathrm{T}})^{\mathrm{T}}$ and formulate the IRK ODE solver as

$$(\mathbf{I}_s \otimes \mathbf{I}_m - h_n \mathbf{A} \otimes \mathbf{Q})\mathbf{z} = h_n(\mathbf{A}\,\mathbf{1}_s) \otimes (\mathbf{Q}\,\mathbf{p}_{n-1}), \tag{4}$$

$$\mathbf{p}_n = \mathbf{p}_{n-1} + h_n(\mathbf{b}^{\mathrm{T}} \otimes \mathbf{I}_m)\big((\mathbf{I}_s \otimes \mathbf{Q})\mathbf{z} + \mathbf{1}_s \otimes (\mathbf{Q}\mathbf{p}_{n-1})\big)\,. \tag{5}$$

If the matrix $\mathbf{A}$ is invertible, we have, from (4),

$$(\mathbf{A}^{-1} \otimes \mathbf{I}_m)\mathbf{z} = h_n\big((\mathbf{I}_s \otimes \mathbf{Q})\mathbf{z} + \mathbf{1}_s \otimes (\mathbf{Q}\mathbf{p}_{n-1})\big)$$

and, therefore, denoting $\mathbf{d}^{\mathrm{T}} = (d_1, \dots, d_s) = \mathbf{b}^{\mathrm{T}}\mathbf{A}^{-1}$, (5) is equivalent to the more convenient expression

$$\mathbf{p}_n = \mathbf{p}_{n-1} + (\mathbf{d}^{\mathrm{T}} \otimes \mathbf{I}_m)\mathbf{z}\,. \tag{6}$$

For the IVP (1), a step of an s-stage LIRK ODE solver is defined by

$$\mathbf{p}_n = \mathbf{p}_{n-1} + \sum_{j=1}^{s} \beta_j \boldsymbol{\eta}_j, \quad n = 1, 2, \dots ,$$

where the vectors $\boldsymbol{\eta}_i$, $i = 1, \dots, s$ are the solution to the $m \times s$-dimensional linear system

$$\boldsymbol{\eta}_i = h_n \mathbf{Q}\mathbf{p}_{n-1} + h_n \mathbf{Q}\Big(\sum_{j=1}^{i-1} (\alpha_{i,j} + \gamma_{i,j})\boldsymbol{\eta}_j + \gamma_{i,i}\boldsymbol{\eta}_i\Big), \quad i = 1, \dots, s\,.$$

The real quantities $\alpha_{i,j}$, β_i, $\gamma_{i,k}$, $1 \le i \le s$, $1 \le j \le i-1$, $1 \le k \le i$, are the coefficients of the LIRK ODE solver. Let $\boldsymbol{\Theta} = (\theta_{i,j})_{1 \le i,j \le s} = (\gamma_{i,j} + \alpha_{i,j})_{1 \le i,j \le s}$. If $\boldsymbol{\Theta}$ is invertible, then, denoting $\boldsymbol{\Theta}^{-1} = (\theta^{(-1)}_{i,j})_{1 \le i,j \le s}$ and defining $\boldsymbol{\omega} = ({\boldsymbol{\omega}_1}^{\mathrm{T}}, \dots, {\boldsymbol{\omega}_s}^{\mathrm{T}})^{\mathrm{T}} = (\boldsymbol{\Theta} \otimes \mathbf{I}_m)({\boldsymbol{\eta}_1}^{\mathrm{T}}, \dots, {\boldsymbol{\eta}_s}^{\mathrm{T}})^{\mathrm{T}}$, the method is more conveniently formulated as

$$(\boldsymbol{\Theta}^{-1} \otimes \mathbf{I}_m)\boldsymbol{\omega} = \mathbf{I}_s \otimes (h_n \mathbf{Q}\mathbf{p}_{n-1}) + (\mathbf{I}_s \otimes h_n\mathbf{Q})\boldsymbol{\omega}, \tag{7}$$

$$\mathbf{p}_n = \mathbf{p}_{n-1} + \Big(\big((\beta_1, \dots, \beta_s)\boldsymbol{\Theta}^{-1}\big) \otimes \mathbf{I}_m\Big)\boldsymbol{\omega} = \mathbf{p}_{n-1} + \sum_{k=1}^{s} \Big(\sum_{j=k}^{s} \beta_j \theta^{(-1)}_{j,k}\Big)\boldsymbol{\omega}_k\,. \tag{8}$$

We claim that when applied to the IVP (1), the following three classes of implicit ODE solvers are such that the linear systems involved in each step can be solved by using iterative methods with strict control of the 1-norm of the step approximation error:

1. Implicit ODE solvers for which $\mathbf{p}_n$ is the solution to a linear system of the form

$$(a(n)\mathbf{I}_m - \mathbf{Q})\mathbf{p}_n = \mathbf{u}_n, \quad n = 1, 2, \dots , \tag{9}$$

   where $a(n) > 0$ for all n.

2. IRK ODE solvers such that either $a_{i,j} = 0$ for all $j > i$ (such ODE solvers are called diagonally implicit Runge-Kutta —DIRK— ODE solvers), $a_{1,1} \geq 0$, and $a_{i,i} > 0$ for all $i \geq 2$, or else the matrix $\mathbf{A}$ is invertible and the eigenvalues of $\mathbf{A}^{-1}$ are all distinct and have positive real part.

3. LIRK ODE solvers with $\gamma_{i,i} > 0$ for all i.

To justify the claim, we will need the following two results. The first one states that under certain conditions, a matrix of the form $(\chi + \mathrm{j}\,\omega)\mathbf{I}_m - \xi\mathbf{Q}$, where χ, ω, ξ are real quantities and $\mathrm{j} = \sqrt{-1}$, is SCDD. The second result will be needed to justify the claim in the case of DIRK and LIRK ODE solvers.

Proposition 2. *Let χ, ω, ξ be real quantities satisfying $\chi\,\xi > 0$, or $\chi = 0$, $\omega \neq 0$. Then, the possibly complex matrix $\mathbf{V} = (v_{i,j})_{1\leq i,j\leq m} = (\chi + \mathrm{j}\,\omega)\mathbf{I}_m - \xi\mathbf{Q}$ is SCDD and*

$$\min_{1\leq i\leq m} \Big(|v_{i,i}| - \sum_{\substack{j=1 \\ j\neq i}}^{m} |v_{j,i}| \Big) = |\chi + \mathrm{j}\,\omega + \xi q| - |\xi| q\,.$$

Proof. See the online supplement. □

Proposition 3. *Let n be a positive integer, let $\chi_{k,j}$, $\xi_{k,j}$, $1 \leq k, j \leq n$, be real quantities such that $\chi_{k,k}\xi_{k,k} > 0$, and consider the sets of vectors $\{\mathbf{x}_k, 1 \leq k \leq n \ : \ (\chi_{k,k}\mathbf{I}_m - \xi_{k,k}\mathbf{Q})\mathbf{x}_k = \mathbf{u}_k + \sum_{j=1}^{k-1}(\xi_{k,j}\mathbf{Q} + \chi_{k,j}\mathbf{I}_m)\mathbf{x}_j\}$, $\{\widetilde{\mathbf{x}}_k^*, 1 \leq k \leq n\}$, and $\{\mathbf{x}_k^*, 1 \leq k \leq n \ : \ (\chi_{k,k}\mathbf{I}_m - \xi_{k,k}\mathbf{Q})\mathbf{x}_k^* = \mathbf{u}_k + \sum_{j=1}^{k-1}(\xi_{k,j}\mathbf{Q} + \chi_{k,j}\mathbf{I}_m)\widetilde{\mathbf{x}}_j^*\}$, where $\mathbf{u}_k$, $1 \leq k \leq n$, are real vectors. Then,*

$$\|\mathbf{x}_k - \widetilde{\mathbf{x}}_k^*\|_1 \leq \sum_{j=1}^{k-1} \Big(2|\xi_{k,j}| \min\Big\{ \frac{q}{|\chi_{k,k}|}, \frac{1}{|\xi_{k,k}|} \Big\} + \frac{|\chi_{k,j}|}{|\chi_{k,k}|} \Big) \|\mathbf{x}_j - \widetilde{\mathbf{x}}_j^*\|_1 + \|\mathbf{x}_k^* - \widetilde{\mathbf{x}}_k^*\|_1, \quad 1 \leq k \leq n\,.$$

Proof. See the online supplement. □

We will now justify that the linear systems involved in each of the three classes of ODE solvers previously mentioned can be solved by using iterative methods with strict control of the 1-norm of the step approximation error.

3.1 Class 1

This class includes the implicit ODE solvers for which $\mathbf{p}_n$ is the solution to (9) with $a(n) > 0$, $n = 1, 2, \dots$. By Proposition 2 with $\chi = a(n)$, $\omega = 0$, and $\xi = 1$, it follows that the matrix $\mathbf{V} = (v_{i,j})_{1\leq i,j\leq m} = a(n)\mathbf{I}_m - \mathbf{Q}$ is SCDD and that $\min_{1\leq i\leq m}\{|v_{i,i}| - \sum_{\substack{j=1 \\ j\neq i}}^{m} |v_{j,i}|\} = |a(n) + q| - q = a(n)$. Therefore, given $\varepsilon > 0$, if the linear system (9) is solved using iterative methods that produce the residual after each iteration or after a predefined number of iterations, with stopping criterion

$$\|\mathbf{u}_n - (a(n)\mathbf{I}_m - \mathbf{Q})\mathbf{p}_n^{(l)}\|_1 \leq \varepsilon\, a(n), \quad l \geq 0\,, \tag{10}$$

or is solved using splitting-based iterative methods defined by (3) with $\mathbf{x} = \mathbf{p}_n$, $\mathbf{M}-\mathbf{N} = a(n)\mathbf{I}_m - \mathbf{Q}$, and $\mathbf{u} = \mathbf{u}_n$, with stopping criterion

$$\|\mathbf{p}_n^{(l)} - \mathbf{p}_n^{(l-1)}\|_1 \leq \varepsilon \frac{a(n)}{\|\mathbf{N}\|_1}, \quad l \geq 1\,, \tag{11}$$

by Theorems 1 and 2 with $\delta = \varepsilon$ and $\min_{1\leq i\leq m}\{|v_{i,i}| - \sum_{\substack{j=1\\ j\neq i}}^{m} |v_{j,i}|\} = a(n)$ we will have $\|\mathbf{s}_n\|_1 \leq \varepsilon$, i.e., strict control of the 1-norm of the step approximation error.[1]

3.2 Class 2

This class includes DIRK ODE solvers with $a_{1,1} \geq 0$, and $a_{i,i} > 0$ for all $i \geq 2$ and IRK ODE solvers such that the matrix $\mathbf{A}$ is invertible and the eigenvalues of $\mathbf{A}^{-1}$ are all distinct and have positive real part.

Consider first an s-stage DIRK ODE solver with $a_{i,i} > 0$, $1 \leq i \leq s$. In this case, the $m \times s$-dimensional linear system (4) can be split into a set of s coupled linear systems of dimension m each,

$$\Big(\frac{1}{h_n a_{k,k}}\mathbf{I}_m - \mathbf{Q}\Big)\mathbf{z}_k = \frac{1}{a_{k,k}}\Big(\sum_{j=1}^{k} a_{k,j}\Big)\mathbf{Q}\mathbf{p}_{n-1} + \sum_{j=1}^{k-1}\frac{a_{k,j}}{a_{k,k}}\mathbf{Q}\mathbf{z}_j, \quad k = 1,\ldots,s\,.$$

These linear systems can be solved for increasing values of k starting at $k = 1$. This implies that the linear systems actually dealt with will be

$$\Big(\frac{1}{h_n a_{k,k}}\mathbf{I}_m - \mathbf{Q}\Big)\mathbf{z}_k^* = \frac{1}{a_{k,k}}\Big(\sum_{j=1}^{k} a_{k,j}\Big)\mathbf{Q}\mathbf{p}_{n-1} + \sum_{j=1}^{k-1}\frac{a_{k,j}}{a_{k,k}}\mathbf{Q}\widetilde{\mathbf{z}}_j^*, \quad k = 1,\ldots,s\,, \tag{12}$$

where $\widetilde{\mathbf{z}}_k^*$, $1 \leq k \leq s$, denotes the computed approximation for $\mathbf{z}_k^*$. Let $\mathbf{V} = (v_{i,j})_{1\leq i,j\leq m} = (1/(h_n a_{k,k}))\mathbf{I}_m - \mathbf{Q}$. By Proposition 2 with $\chi = 1/(h_n a_{k,k})$, $\omega = 0$, and $\xi = 1$, it follows that the matrix $\mathbf{V}$ is SCDD and that $\min_{1\leq i\leq m}\{|v_{i,i}| - \sum_{\substack{j=1\\ j\neq i}}^{m} |v_{j,i}|\} = |1/(h_n a_{k,k}) + q| - q = 1/(h_n a_{k,k})$. Therefore, if the kth linear system (12) is solved using iterative methods that produce the residual after each iteration or after a predefined number of iterations, with stopping criterion

$$\left\| \frac{1}{a_{k,k}}\Big(\sum_{j=1}^{k} a_{k,j}\Big)\mathbf{Q}\mathbf{p}_{n-1} + \sum_{j=1}^{k-1}\frac{a_{k,j}}{a_{k,k}}\mathbf{Q}\widetilde{\mathbf{z}}_j^* - \Big(\frac{1}{h_n a_{k,k}}\mathbf{I}_m - \mathbf{Q}\Big)\mathbf{z}_k^{*(l)} \right\|_1 \leq \delta \frac{1}{h_n a_{k,k}}, \quad l \geq 0\,, \tag{13}$$

for some $\delta > 0$, or is solved using splitting-based iterative methods defined by (3) with $\mathbf{x} = \mathbf{z}_k^*$, $\mathbf{M} - \mathbf{N} = (1/(h_n a_{k,k}))\mathbf{I}_m - \mathbf{Q}$, and $\mathbf{u} = (1/a_{k,k})(\sum_{j=1}^{k} a_{k,j})\mathbf{Q}\mathbf{p}_{n-1} + \sum_{j=1}^{k-1}(a_{k,j}/a_{k,k})\mathbf{Q}\widetilde{\mathbf{z}}_j^*$, with stopping criterion

$$\|\mathbf{z}_k^{*(l)} - \mathbf{z}_k^{*(l-1)}\|_1 \leq \delta \frac{1}{h_n a_{k,k}\|\mathbf{N}\|_1}, \quad l \geq 1\,, \tag{14}$$

for some $\delta > 0$, by Theorems 1 and 2 with $\min_{1\leq i\leq m}\{|v_{i,i}| - \sum_{\substack{j=1\\ j\neq i}}^{m} |v_{j,i}|\} = 1/(h_n a_{k,k})$, we will have $\|\mathbf{z}_k^* - \widetilde{\mathbf{z}}_k^*\|_1 \leq \delta$. Then, by Proposition 3 with $\mathbf{x}_k = \mathbf{z}_k$, $\chi_{k,k} = 1/(h_n a_{k,k})$, $\xi_{k,k} = 1$, $\mathbf{u}_k = (1/(a_{k,k}))(\sum_{j=1}^{k} a_{k,j})\mathbf{Q}\mathbf{p}_{n-1}$, $\xi_{k,j} = a_{k,j}/a_{k,k}$, $1 \leq j \leq k-1$, $\chi_{k,j} = 0$, $1 \leq j \leq k-1$, $\widetilde{\mathbf{x}}_k^* = \widetilde{\mathbf{z}}_k^*$, $\mathbf{x}_k^* = \mathbf{z}_k^*$, and $n = s$, it follows that

$$\|\mathbf{z}_k - \widetilde{\mathbf{z}}_k^*\|_1 \leq \sum_{j=1}^{k-1} 2\frac{|a_{k,j}|}{a_{k,k}} \min\Big\{q h_n a_{k,k}, 1\Big\}\nu_j(\delta) + \delta = \nu_k(\delta), \quad k = 1,\ldots,s\,.$$

Now, since $\mathbf{A}$ is invertible because, as assumed, $a_{i,i} > 0$, $1 \leq i \leq s$, we can compute $\mathbf{p}_n$ using (6) with $\mathbf{z}_k$ replaced by the approximation $\widetilde{\mathbf{z}}_k^*$, incurring a step approximation error $\mathbf{s}_n$ that will satisfy

[1] Note that, since we are ignoring the impact of round-off errors, we have, in terms of the notation introduced in Section 1, $\widehat{\mathbf{p}}_n = \mathbf{p}_n^{(l)}$, $\mathbf{p}_n^+ = \mathbf{p}_n$ and, therefore, $\mathbf{s}_n = \widehat{\mathbf{p}}_n - \mathbf{p}_n^+ = \mathbf{p}_n^{(l)} - \mathbf{p}_n$.

$\|\mathbf{s}_n\|_1 \leq \sum_{k=1}^{s}|d_k|\|\mathbf{z}_k - \widetilde{\mathbf{z}}_k^*\|_1 \leq \sum_{k=1}^{s}|d_k|\nu_k(\delta)$. Therefore, given some $\varepsilon > 0$, taking for δ in (13), (14) the value of x that satisfies $\sum_{k=1}^{s}|d_k|\nu_k(x) = \varepsilon$ will result in $\|\mathbf{s}_n\|_1 \leq \varepsilon$, i.e., strict control of the 1-norm of the step approximation error.

Consider next an s-stage DIRK ODE solver with $a_{1,1} = 0$ and $a_{i,i} > 0$, $2 \leq i \leq s$. In this case, the $m \times s$-dimensional linear system (4) can be split into $\mathbf{z}_1 = (0, \ldots, 0)^{\mathrm{T}}$ and the set of $s-1$ coupled linear systems

$$\Big(\frac{1}{h_n a_{k,k}}\mathbf{I}_m - \mathbf{Q}\Big)\mathbf{z}_k = \frac{1}{a_{k,k}}\Big(\sum_{j=1}^{k} a_{k,j}\Big)\mathbf{Q}\mathbf{p}_{n-1} + \sum_{j=2}^{k-1}\frac{a_{k,j}}{a_{k,k}}\mathbf{Q}\mathbf{z}_j, \quad k = 2, \ldots, s\,.$$

Those linear systems can be solved for increasing values of k starting at $k = 2$. This implies that the linear systems actually dealt with will be

$$\Big(\frac{1}{h_n a_{k+1,k+1}}\mathbf{I}_m - \mathbf{Q}\Big)\mathbf{z}_{k+1}^* = \frac{1}{a_{k+1,k+1}}\Big(\sum_{j=1}^{k+1} a_{k+1,j}\Big)\mathbf{Q}\mathbf{p}_{n-1} + \sum_{j=1}^{k-1}\frac{a_{k+1,j+1}}{a_{k+1,k+1}}\mathbf{Q}\widetilde{\mathbf{z}}_{j+1}^*, \quad k = 1, \ldots, s-1\,, \tag{15}$$

where $\widetilde{\mathbf{z}}_k^*$, $2 \leq k \leq s$, denotes the computed approximation for $\mathbf{z}_k^*$. We have already argued that the matrix $\mathbf{V} = (v_{i,j})_{1\leq i,j\leq m} = (1/(h_n a_{k+1,k+1}))\mathbf{I}_m - \mathbf{Q}$ is SCDD and that $\min_{1\leq i\leq m}\{|v_{i,i}| - \sum_{\substack{j=1\\ j\neq i}}^{m}|v_{j,i}|\} = 1/(h_n a_{k+1,k+1})$. Consequently, if the kth linear system (15) is solved using iterative methods that produce the residual after each iteration or after a predefined number of iterations, with stopping criterion

$$\begin{aligned}&\Big\|\frac{1}{a_{k+1,k+1}}\Big(\sum_{j=1}^{k+1} a_{k+1,j}\Big)\mathbf{Q}\mathbf{p}_{n-1} + \sum_{j=1}^{k-1}\frac{a_{k+1,j+1}}{a_{k+1,k+1}}\mathbf{Q}\widetilde{\mathbf{z}}_{j+1}^* - \Big(\frac{1}{h_n a_{k+1,k+1}}\mathbf{I}_m - \mathbf{Q}\Big)\mathbf{z}_{k+1}^{*(l)}\Big\|_1\\ &\leq \delta\frac{1}{h_n a_{k+1,k+1}}, \quad l \geq 0\,,\end{aligned} \tag{16}$$

for some $\delta > 0$, or is solved using splitting-based iterative methods defined by (3) with $\mathbf{x} = \mathbf{z}_{k+1}^*$, $\mathbf{M} - \mathbf{N} = (1/(h_n a_{k+1,k+1}))\mathbf{I}_m - \mathbf{Q}$, and $\mathbf{u} = (1/a_{k+1,k+1})(\sum_{j=1}^{k+1} a_{k+1,j})\mathbf{Q}\mathbf{p}_{n-1} + \sum_{j=1}^{k-1}(a_{k+1,j+1}/a_{k+1,k+1})\mathbf{Q}\widetilde{\mathbf{z}}_{j+1}^*$, with stopping criterion

$$\|\mathbf{z}_{k+1}^{*(l)} - \mathbf{z}_{k+1}^{*(l-1)}\|_1 \leq \delta\frac{1}{h_n a_{k+1,k+1}\|\mathbf{N}\|_1}, \quad l \geq 1\,, \tag{17}$$

for some $\delta > 0$, by Theorems 1 and 2 with $\min_{1\leq i\leq m}\{|v_{i,i}| - \sum_{\substack{j=1\\ j\neq i}}^{m}|v_{j,i}|\} = 1/(h_n a_{k+1,k+1})$, we will have $\|\mathbf{z}_{k+1}^* - \widetilde{\mathbf{z}}_{k+1}^*\|_1 \leq \delta$. Therefore, by Proposition 3 with $\mathbf{x}_k = \mathbf{z}_{k+1}$, $\chi_{k,k} = 1/(h_n a_{k+1,k+1})$, $\xi_{k,k} = 1$, $\mathbf{u}_k = (1/(a_{k+1,k+1}))(\sum_{j=1}^{k+1} a_{k+1,j})\mathbf{Q}\mathbf{p}_{n-1}$, $\xi_{k,j} = a_{k+1,j+1}/a_{k+1,k+1}$, $1 \leq j \leq k-1$, $\chi_{k,j} = 0$, $1 \leq j \leq k-1$, $\widetilde{\mathbf{x}}_k^* = \widetilde{\mathbf{z}}_{k+1}^*$, $\mathbf{x}_k^* = \mathbf{z}_{k+1}^*$, and $n = s-1$, it follows that

$$\begin{aligned}\|\mathbf{z}_{k+1} - \widetilde{\mathbf{z}}_{k+1}^*\|_1 &\leq \sum_{j=1}^{k-1} 2\frac{|a_{k+1,j+1}|}{a_{k+1,k+1}}\min\Big\{q h_n a_{k+1,k+1}, 1\Big\}\nu_{j+1}(\delta) + \delta\\ &= \nu_{k+1}(\delta), \quad k = 1, \ldots, s-1\,.\end{aligned}$$

Now, denoting $\mathbf{A}' = (a_{i,j})_{2\leq i,j\leq s}$, $\mathbf{z}' = (\mathbf{z}_2{}^{\mathrm{T}}, \ldots, \mathbf{z}_s{}^{\mathrm{T}})^{\mathrm{T}}$ and recalling that $\mathbf{z}_1 = (0, \ldots, 0)^{\mathrm{T}}$, we have, by (4),

$$(\mathbf{I}_{s-1} \otimes \mathbf{I}_m - h_n\mathbf{A}' \otimes \mathbf{Q})\mathbf{z}' = h_n\big((\mathbf{A}'\,\mathbf{1}_{s-1}) \otimes (\mathbf{Q}\,\mathbf{p}_{n-1}) + (a_{2,1}, \ldots, a_{s,1})^{\mathrm{T}} \otimes (\mathbf{Q}\,\mathbf{p}_{n-1})\big)\,,$$

which, since the matrix $\mathbf{A}'$ is invertible because, as assumed, $a_{i,i} > 0$, $2 \leq i \leq s$, implies

$$((\mathbf{A}')^{-1} \otimes \mathbf{I}_m)\mathbf{z}' = h_n\big((\mathbf{I}_{s-1} \otimes \mathbf{Q})\mathbf{z}' + \mathbf{1}_{s-1} \otimes (\mathbf{Q}\mathbf{p}_{n-1}) + \big((\mathbf{A}')^{-1}(a_{2,1}, \ldots, a_{s,1})^{\mathrm{T}}\big) \otimes (\mathbf{Q}\mathbf{p}_{n-1})\big)\,.$$

Combining this expression with (5) gives, recalling again that $\mathbf{z}_1 = (0,\dots,0)^{\mathrm{T}}$ and denoting $(d'_2,\dots,d'_s) = (b_2,\dots,b_s)(\mathbf{A}')^{-1}$,

$$\begin{aligned}\mathbf{p}_n &= \mathbf{p}_{n-1} + h_n b_1 \mathbf{Q}\mathbf{p}_{n-1} + h_n((b_2,\dots,b_s)\otimes\mathbf{I}_m)\big((\mathbf{I}_{s-1}\otimes\mathbf{Q})\mathbf{z}' + \mathbf{1}_{s-1}\otimes(\mathbf{Q}\mathbf{p}_{n-1})\big)\\ &= \mathbf{p}_{n-1} + h_n b_1 \mathbf{Q}\mathbf{p}_{n-1} + ((b_2,\dots,b_s)\otimes\mathbf{I}_m)\Big(\big((\mathbf{A}')^{-1}\otimes\mathbf{I}_m\big)\mathbf{z}'\\ &\qquad\qquad - h_n((\mathbf{A}')^{-1}(a_{2,1},\dots,a_{s,1})^{\mathrm{T}})\otimes(\mathbf{Q}\mathbf{p}_{n-1})\Big)\\ &= \mathbf{p}_{n-1} + h_n\big(b_1 - (d'_2,\dots,d'_s)(a_{2,1},\dots,a_{s,1})^{\mathrm{T}}\big)\mathbf{Q}\mathbf{p}_{n-1} + ((d'_2,\dots,d'_s)\otimes\mathbf{I}_m)\mathbf{z}'\,.\end{aligned}$$

Then, $\mathbf{p}_n$ can be computed using the above expression with $\mathbf{z}_k$ replaced by the approximation $\widetilde{\mathbf{z}}^*_k$, incurring a step approximation error $\mathbf{s}_n$ that will satisfy $\|\mathbf{s}_n\|_1 \le \sum_{k=2}^s |d'_k| \|\mathbf{z}_k - \widetilde{\mathbf{z}}^*_k\|_1 \le \sum_{k=2}^s |d'_k|\nu_k(\delta)$. Therefore, given some $\varepsilon > 0$, taking for δ in (16), (17) the value of x that satisfies $\sum_{k=2}^s |d'_k|\nu_k(x) = \varepsilon$ will yield $\|\mathbf{s}_n\|_1 \le \varepsilon$, i.e., strict control of the 1-norm of the step approximation error.

Finally, consider an s-stage IRK ODE solver whose matrix $\mathbf{A}$ is invertible and the eigenvalues of $\mathbf{A}^{-1}$ are all distinct and have positive real part. Following standard practice [22], in that case it is possible to transform the $m\times s$-dimensional linear system (4) into several, uncoupled linear systems of dimension m each as follows. First, we left multiply (4) by $(h_n\mathbf{A})^{-1}\otimes\mathbf{I}_m$ and obtain

$$((h_n\mathbf{A})^{-1}\otimes\mathbf{I}_m - \mathbf{I}_s\otimes\mathbf{Q})\mathbf{z} = \mathbf{1}_s\otimes(\mathbf{Q}\mathbf{p}_{n-1})\,. \tag{18}$$

Next, using a change of basis matrix $\mathbf{T} = (t_{i,j})_{1\le i,j\le s}$, we apply a similarity transformation $\mathbf{T}^{-1}\mathbf{A}^{-1}\mathbf{T} = \mathbf{\Lambda}$, obtaining

$$(h_n^{-1}\mathbf{\Lambda}\otimes\mathbf{I}_m - \mathbf{I}_s\otimes\mathbf{Q})\mathbf{v} = (\mathbf{T}^{-1}\mathbf{1}_s)\otimes(\mathbf{Q}\mathbf{p}_{n-1})\,, \tag{19}$$

where

$$\mathbf{v} = ({\mathbf{v}_1}^{\mathrm{T}},\dots,{\mathbf{v}_s}^{\mathrm{T}})^{\mathrm{T}} = (\mathbf{T}^{-1}\otimes\mathbf{I}_m)\mathbf{z}\,. \tag{20}$$

Assume that $\mathbf{A}^{-1}$ has r real eigenvalues $\rho_1,\dots,\rho_r$ and, consequently, $(s-r)/2$ complex conjugate eigenvalue pairs $\sigma_{r+1}\pm\mathrm{j}\phi_{r+2},\dots,\sigma_{s-1}\pm\mathrm{j}\phi_s$, let $\mathbf{r}_k$, $1\le k\le r$, denote the real eigenvector associated with ρ_k, and let $\mathbf{r}_k\pm\mathrm{j}\mathbf{c}_{k+1}$, $r+1\le k\le s-1$, denote the complex conjugate eigenvector pair associated with $\sigma_k\pm\mathrm{j}\phi_{k+1}$. Then, by setting to $\mathbf{r}_k$ columns $k = 1,\dots,r,r+1,r+3,\dots,s-1$ of $\mathbf{T}$ and to $-\mathbf{c}_k$ columns $k = r+2,r+4,\dots,s$ of $\mathbf{T}$, the matrix $\mathbf{\Lambda}$ becomes block diagonal with diagonal blocks the 1×1 matrices (ρ_k) and the 2×2 matrices $\mathbf{\Lambda}_k = \left(\begin{smallmatrix}\sigma_k & -\phi_{k+1}\\ \phi_{k+1} & \sigma_k\end{smallmatrix}\right)$. Hence, denoting $\mathbf{T}^{-1} = (t^{(-1)}_{i,j})_{1\le i,j\le s}$, it is easily seen that the linear system (19) can be split into r real and $(s-r)/2$ complex linear systems of dimension m each,

$$\Big(\frac{\rho_k}{h_n}\mathbf{I}_m - \mathbf{Q}\Big)\mathbf{v}_k = \Big(\sum_{j=1}^s t^{(-1)}_{k,j}\Big)\mathbf{Q}\mathbf{p}_{n-1},\quad 1\le k\le r\,, \tag{21}$$

$$\begin{aligned}\Big(\frac{\sigma_k+\mathrm{j}\phi_{k+1}}{h_n}\mathbf{I}_m - \mathbf{Q}\Big)(\mathbf{v}_k+\mathrm{j}\mathbf{v}_{k+1}) &= \Big(\sum_{j=1}^s t^{(-1)}_{k,j} + \mathrm{j}\sum_{j=1}^s t^{(-1)}_{k+1,j}\Big)\mathbf{Q}\mathbf{p}_{n-1},\\ &\qquad k = r+1,r+3,\dots,s-1\,.\end{aligned} \tag{22}$$

Once the $\mathbf{v}_k$, $1\le k\le s$, are known, from (6) and, by (20), $\mathbf{z} = (\mathbf{T}\otimes\mathbf{I}_m)\mathbf{v}$, the vector $\mathbf{p}_n$ can be computed using

$$\mathbf{p}_n = \mathbf{p}_{n-1} + ((\mathbf{d}^{\mathrm{T}}\mathbf{T})\otimes\mathbf{I}_m)\mathbf{v} = \mathbf{p}_{n-1} + \sum_{k=1}^s\Big(\sum_{j=1}^s d_j t_{j,k}\Big)\mathbf{v}_k\,. \tag{23}$$

We discuss next the computation of the vectors $\mathbf{v}_k$, $1\le k\le r$, by solving the linear systems (21) and the computation of the vectors $\mathbf{v}_k$, $r+1\le k\le s$, by solving the linear systems (22). Let $\varepsilon > 0$ and consider first the computation of the vectors $\mathbf{v}_k$, $1\le k\le r$. By Proposition 2 with

$\chi = \rho_k/h_n$, $\omega = 0$, and $\xi = 1$, it follows that the matrix $\mathbf{V} = (v_{i,j})_{1\le i,j\le m} = (\rho_k/h_n)\mathbf{I}_m - \mathbf{Q}$ is SCDD and that $\min_{1\le i\le m}\{|v_{i,i}| - \sum_{\substack{j=1\\ j\ne i}}^{m}|v_{j,i}|\} = |\rho_k/h_n + q| - q = \rho_k/h_n$. Therefore, if the kth linear system (21) is solved using iterative methods that produce the residual after each iteration or after a predefined number of iterations, with stopping criterion

$$\Big\| \Big(\sum_{j=1}^{s} t_{k,j}^{(-1)} \Big) \mathbf{Q}\mathbf{p}_{n-1} - \Big(\frac{\rho_k}{h_n}\mathbf{I}_m - \mathbf{Q} \Big) \mathbf{v}_k^{(l)} \Big\|_1 \le \frac{\varepsilon}{\sum_{l=1}^{s}|\sum_{j=1}^{s} d_j t_{j,l}|} \frac{\rho_k}{h_n}, \quad l \ge 0\,, \tag{24}$$

or is solved using splitting-based iterative methods defined by (3) with $\mathbf{x} = \mathbf{v}_k$, $\mathbf{M} - \mathbf{N} = (\rho_k/h_n)\mathbf{I}_m - \mathbf{Q}$, and $\mathbf{u} = (\sum_{j=1}^{s} t_{k,j}^{(-1)})\mathbf{Q}\mathbf{p}_{n-1}$, with stopping criterion

$$\|\mathbf{v}_k^{(l)} - \mathbf{v}_k^{(l-1)}\|_1 \le \frac{\varepsilon}{\sum_{l=1}^{s}|\sum_{j=1}^{s} d_j t_{j,l}|} \frac{\rho_k}{h_n \|\mathbf{N}\|_1}, \quad l \ge 1\,, \tag{25}$$

by Theorems 1 and 2 with $\min_{1\le i\le m}\{|v_{i,i}| - \sum_{\substack{j=1\\ j\ne i}}^{m}|v_{j,i}|\} = \rho_k/h_n$ and $\delta = \varepsilon/\sum_{l=1}^{s}|\sum_{j=1}^{s} d_j t_{j,l}|$ we will have

$$\|\mathbf{v}_k - \widetilde{\mathbf{v}}_k\|_1 \le \frac{\varepsilon}{\sum_{l=1}^{s}|\sum_{j=1}^{s} d_j t_{j,l}|}, \quad 1 \le k \le r\,, \tag{26}$$

where $\widetilde{\mathbf{v}}_k$ denotes the computed approximation for $\mathbf{v}_k$. Consider now the computation of the vectors $\mathbf{v}_k$, $r+1 \le k \le s$, by solving the linear systems (22). By Proposition 2 with $\chi = \sigma_k/h_n$, $\omega = \phi_{k+1}/h_n$, and $\xi = 1$, it follows that the matrix $\mathbf{V} = (v_{i,j})_{1\le i,j\le m} = ((\sigma_k + \mathrm{j}\phi_{k+1})/h_n)\mathbf{I}_m - \mathbf{Q}$ is SCDD and that $\min_{1\le i\le m}\{|v_{i,i}| - \sum_{\substack{j=1\\ j\ne i}}^{m}|v_{j,i}|\} = \big|(\sigma_k + \mathrm{j}\phi_{k+1})/h_n + q\big| - q$. Therefore, if the kth linear system (22) is solved using iterative methods that produce the residual after each iteration or after a predefined number of iterations, with stopping criterion

$$\begin{aligned}\Big\| \Big(\sum_{j=1}^{s} t_{k,j}^{(-1)} + \mathrm{j}\sum_{j=1}^{s} t_{k+1,j}^{(-1)} \Big) \mathbf{Q}\mathbf{p}_{n-1} - \Big(\frac{\sigma_k + \mathrm{j}\phi_{k+1}}{h_n}\mathbf{I}_m - \mathbf{Q} \Big) (\mathbf{v}_k^{(l)} + \mathrm{j}\mathbf{v}_{k+1}^{(l)}) \Big\|_1 \\ \le \frac{\varepsilon}{\sum_{l=1}^{s}|\sum_{j=1}^{s} d_j t_{j,l}|} \Big(\Big| \frac{\sigma_k + \mathrm{j}\phi_{k+1}}{h_n} + q \Big| - q \Big), \quad l \ge 0\,,\end{aligned} \tag{27}$$

or is solved using splitting-based iterative methods defined by (3) with $\mathbf{x} = \mathbf{v}_k + \mathrm{j}\mathbf{v}_{k+1}$, $\mathbf{M} - \mathbf{N} = ((\sigma_k + \mathrm{j}\phi_{k+1})/h_n)\mathbf{I}_m - \mathbf{Q}$, and $\mathbf{u} = (\sum_{j=1}^{s} t_{k,j}^{(-1)} + \mathrm{j}\sum_{j=1}^{s} t_{k+1,j}^{(-1)})\mathbf{Q}\mathbf{p}_{n-1}$, with stopping criterion

$$\begin{aligned}\big\| \mathbf{v}_k^{(l)} + \mathrm{j}\mathbf{v}_{k+1}^{(l)} - (\mathbf{v}_k^{(l-1)} + \mathrm{j}\mathbf{v}_{k+1}^{(l-1)}) \big\|_1 \\ \le \frac{\varepsilon}{\sum_{l=1}^{s}|\sum_{j=1}^{s} d_j t_{j,l}|} \Big(\Big| \frac{\sigma_k + \mathrm{j}\phi_{k+1}}{h_n} + q \Big| - q \Big) \frac{1}{\|\mathbf{N}\|_1}, \quad l \ge 1\,,\end{aligned} \tag{28}$$

by Theorems 1 and 2 with $\min_{1\le i\le m}\{|v_{i,i}| - \sum_{\substack{j=1\\ j\ne i}}^{m}|v_{j,i}|\} = |(\sigma_k + \mathrm{j}\phi_{k+1})/h_n + q| - q$ and $\delta = \varepsilon/\sum_{l=1}^{s}|\sum_{j=1}^{s} d_j t_{j,l}|$ we will have

$$\|\mathbf{v}_k + \mathrm{j}\mathbf{v}_{k+1} - (\widetilde{\mathbf{v}}_k + \mathrm{j}\widetilde{\mathbf{v}}_{k+1})\|_1 \le \frac{\varepsilon}{\sum_{l=1}^{s}|\sum_{j=1}^{s} d_j t_{j,l}|}\,,$$

implying, since $\max\{\|\mathbf{v}_k - \widetilde{\mathbf{v}}_k\|_1, \|\mathbf{v}_{k+1} - \widetilde{\mathbf{v}}_{k+1}\|_1\} \le \|\mathbf{v}_k + \mathrm{j}\mathbf{v}_{k+1} - (\widetilde{\mathbf{v}}_k + \mathrm{j}\widetilde{\mathbf{v}}_{k+1})\|_1$,

$$\|\mathbf{v}_k - \widetilde{\mathbf{v}}_k\|_1 \le \frac{\varepsilon}{\sum_{l=1}^{s}|\sum_{j=1}^{s} d_j t_{j,l}|}, \quad k = r+1, r+3, \ldots, s\,. \tag{29}$$

Then, by (26), (29), the 1-norm of the step approximation error $\mathbf{s}_n$ that will result from the computation of $\mathbf{p}_n$ using (23) with $\mathbf{v}_k$ replaced by $\widetilde{\mathbf{v}}_k$ will satisfy,

$$\|\mathbf{s}_n\|_1 \le \sum_{k=1}^{s} \Big| \sum_{j=1}^{s} d_j t_{j,k} \Big| \|\mathbf{v}_k - \widetilde{\mathbf{v}}_k\|_1 \le \sum_{k=1}^{s} \Big| \sum_{j=1}^{s} d_j t_{j,k} \Big| \frac{\varepsilon}{\sum_{l=1}^{s}|\sum_{j=1}^{s} d_j t_{j,l}|} = \varepsilon\,,$$

yielding strict control of the 1-norm of the step approximation error.

3.3 Class 3

This class includes s-stage LIRK ODE solvers with $\gamma_{i,i} > 0$ for all i. In that case, the matrix $\mathbf{\Theta}$ is invertible and $\theta_{i,i}^{(-1)} = \gamma_{i,i}^{-1}$. Therefore, the linear system (7) can be split into a set of s coupled linear systems of dimension m each,

$$\Big(\frac{1}{h_n\gamma_{k,k}}\mathbf{I}_m - \mathbf{Q}\Big)\boldsymbol{\omega}_k = \mathbf{Q}\mathbf{p}_{n-1} - \sum_{j=1}^{k-1}\frac{\theta_{k,j}^{(-1)}}{h_n}\boldsymbol{\omega}_j, \quad k = 1, \ldots, s\,.$$

These linear systems can be solved for increasing values of k starting at $k = 1$. This implies that the linear systems actually dealt with will be

$$\Big(\frac{1}{h_n\gamma_{k,k}}\mathbf{I}_m - \mathbf{Q}\Big)\boldsymbol{\omega}_k^* = \mathbf{Q}\mathbf{p}_{n-1} - \sum_{j=1}^{k-1}\frac{\theta_{k,j}^{(-1)}}{h_n}\widetilde{\boldsymbol{\omega}}_j^*, \quad k = 1, \ldots, s\,. \tag{30}$$

where $\widetilde{\boldsymbol{\omega}}_k^*$, $1 \leq k \leq s$, denotes the computed approximation for $\boldsymbol{\omega}_k^*$. By Proposition 2 with $\chi = 1/(h_n\gamma_{k,k})$, $\omega = 0$, and $\xi = 1$, it follows that the matrix $\mathbf{V} = (v_{i,j})_{1\leq i,j\leq m} = (1/(h_n\gamma_{k,k}))\mathbf{I}_m - \mathbf{Q}$ is SCDD and that $\min_{1\leq i\leq m}\{|v_{i,i}| - \sum_{\substack{j=1\\ j\neq i}}^{m}|v_{j,i}|\} = |1/(h_n\gamma_{k,k}) + q| - q = 1/(h_n\gamma_{k,k})$. Hence, if the kth linear system (30) is solved using iterative methods that produce the residual after each iteration or after a predefined number of iterations, with stopping criterion

$$\Big\|\,\mathbf{Q}\mathbf{p}_{n-1} - \sum_{j=1}^{k-1}\frac{\theta_{k,j}^{(-1)}}{h_n}\widetilde{\boldsymbol{\omega}}_j^* - \Big(\frac{1}{h_n\gamma_{k,k}}\mathbf{I}_m - \mathbf{Q}\Big)\boldsymbol{\omega}_k^{*(l)}\,\Big\|_1 \leq \delta\frac{1}{h_n\gamma_{k,k}}, \quad l \geq 0\,, \tag{31}$$

for some $\delta > 0$, or is solved using splitting-based iterative methods defined by (3) with $\mathbf{x} = \boldsymbol{\omega}_k^*$, $\mathbf{M} - \mathbf{N} = (1/(h_n\gamma_{k,k}))\mathbf{I}_m - \mathbf{Q}$, and $\mathbf{u} = \mathbf{Q}\mathbf{p}_{n-1} - \sum_{j=1}^{k-1}(\theta_{k,j}^{(-1)}/h_n)\widetilde{\boldsymbol{\omega}}_j^*$, with stopping criterion

$$\|\boldsymbol{\omega}_k^{*(l)} - \boldsymbol{\omega}_k^{*(l-1)}\|_1 \leq \delta\frac{1}{h_n\gamma_{k,k}\|\mathbf{N}\|_1}, \quad l \geq 1\,, \tag{32}$$

for some $\delta > 0$, by Theorems 1 and 2 with $\min_{1\leq i\leq m}\{|v_{i,i}| - \sum_{\substack{j=1\\ j\neq i}}^{m}|v_{j,i}|\} = 1/(h_n\gamma_{k,k})$ we will have $\|\boldsymbol{\omega}_k^* - \widetilde{\boldsymbol{\omega}}_k^*\|_1 \leq \delta$. Then, by Proposition 3 with $\mathbf{x}_k = \boldsymbol{\omega}_k$, $\chi_{k,k} = 1/(h_n\gamma_{k,k})$, $\xi_{k,k} = 1$, $\mathbf{u}_k = \mathbf{Q}\mathbf{p}_{n-1}$, $\xi_{k,j} = 0$, $1 \leq j \leq k-1$, $\chi_{k,j} = -\theta_{k,j}^{(-1)}/h_n$, $1 \leq j \leq k-1$, $\widetilde{\mathbf{x}}_k^* = \widetilde{\boldsymbol{\omega}}_k^*$, $\mathbf{x}_k^* = \boldsymbol{\omega}_k^*$, and $n = s$, it follows that

$$\|\boldsymbol{\omega}_k - \widetilde{\boldsymbol{\omega}}_k^*\|_1 \leq \sum_{j=1}^{k-1}\gamma_{k,k}|\theta_{k,j}^{(-1)}|\nu_j(\delta) + \delta = \nu_k(\delta), \quad k = 1, \ldots, s\,.$$

We can then compute $\mathbf{p}_n$ using (8) with $\boldsymbol{\omega}_k$ replaced by the approximation $\widetilde{\boldsymbol{\omega}}_k^*$, incurring a step approximation error $\mathbf{s}_n$ that will satisfy $\|\mathbf{s}_n\|_1 \leq \sum_{k=1}^{s}|\sum_{j=k}^{s}\beta_j\theta_{j,k}^{(-1)}|\|\boldsymbol{\omega}_k - \widetilde{\boldsymbol{\omega}}_k^*\|_1 \leq \sum_{k=1}^{s}|\sum_{j=k}^{s}\beta_j\theta_{j,k}^{(-1)}|\nu_k(\delta)$. Therefore, given some $\varepsilon > 0$, taking for δ in (31), (32) the value of x that satisfies $\sum_{k=1}^{s}|\sum_{j=k}^{s}\beta_j\theta_{j,k}^{(-1)}|\nu_k(x) = \varepsilon$, we will have $\|\mathbf{s}_n\|_1 \leq \varepsilon$, i.e., strict control of the 1-norm of the step approximation error. This concludes the justification of the claim.

3.4 ODE Solvers Falling in Classes 1, 2 and 3

We end this section by noting that implicit ODE solvers falling in classes 1, 2 and 3 include, among others, those considered in [8, 10, 11, 12, 13, 14, 15, 16], i.e., TR-BDF2, which can be regarded as a 3-stage DIRK ODE solver with $a_{1,1} = 0$, $a_{2,2} = a_{3,3} = 1 - \sqrt{2}/2$ [35], the implicit Euler method, which can be regarded as a 1-stage DIRK ODE solver with $a_{1,1} = 1$, the trapezoidal rule, which can also be regarded as a 1-stage DIRK ODE solver with $a_{1,1} = 1/2$, the Radau IIA and Gauss families of IRK ODE solvers,[2] whose matrix $\mathbf{A}$ is invertible and the eigenvalues of $\mathbf{A}^{-1}$ are all

[2]This includes the implicit ODE solver IRK3 described in [10] because for the IVP (1), a step of IRK3 is mathematically equivalent [23] to a step of the 2-stage Radau IIA implicit ODE solver.

distinct and have positive real part, the two 2-stage DIRK ODE solvers considered in [14], which have $a_{1,1} = a_{2,2} = 1 - \sqrt{2}/2 > 0$ and $a_{1,1} = a_{2,2} = 1/2 + \sqrt{3}/6 > 0$, respectively, and the variable-step, variable-order implicit ODE solver based on BDFs with variable coefficients (VBDF) [36], in which each step is the solution to a linear system of the form (9) with $a(n) = \sum_{i=1}^{ord} h_{n-i}^{-1} > 0$, where *ord* denotes the order of accuracy. Interestingly, there are two other variable-step, variable-order implicit ODE solvers based on BDFs whose steps are the solution to a linear system of the form (9) with $a(n) > 0$, $n = 1, 2, \ldots$, and hence the steps can be computed by using iterative methods with the modifications discussed in Section 2 with strict control of the 1-norm of the step approximation error. Those ODE solvers are the variable-step, variable-order ODE solver based on fixed-coefficient BDFs that is described in [37], for which $a(n)$ is equal to h_n^{-1} times a positive coefficient that depends on the current order of accuracy, and the variable-step, variable-order ODE solver based on fixed leading-coefficient BDFs that is described in [38], for which $a(n) = \sum_{i=1}^{ord} 1/i > 0$.

4 Implementations of VBDF

This section reviews a widely used implicit ODE solver belonging to the classes identified in Section 3, namely VBDF, and, using the results of that section, develops an implementation of that ODE solver when applied to the IVP (1) that provides strict control of the 1-norm of the step approximation error. For comparison purposes, it also develops a reasonable standard implementation of the method that does not provide such a strict error control

As previously mentioned, VBDF is a variable-step, variable-order implicit ODE solver based on BDFs with variable coefficients [36]. Applied to the IVP (1), each step of the method is the solution to a linear system of the form (9) with $a(n) > 0$, $n \geq 0$. More precisely, the linear system is

$$\Big(\Big(\sum_{i=1}^{ord} \frac{1}{h_{n-i}}\Big)\mathbf{I}_m - \mathbf{Q}\Big)\mathbf{p}_n = \Big(\sum_{i=1}^{ord} \frac{1}{h_{n-i}}\Big)\mathbf{p}_{n,0} - \frac{1}{h_n}\mathbf{p}_{n,0}^{(1)}\,, \tag{33}$$

where $\mathbf{p}_{n,0}$ denotes a predictor for $\mathbf{p}_n$, *ord* denotes the current order of accuracy of the method, which for stability reasons is restricted to satisfy $1 \leq ord \leq 5$, and $\mathbf{p}_{n,0}^{(k)}$ denotes a predictor for the approximation, $\mathbf{p}_n^{(k)}$, produced by the method for the kth derivative of $\mathbf{p}(t)$ at $t = t_n$. These predictors are computed from $\mathbf{p}_{n-1}$ and $\mathbf{p}_{n-1}^{(i)}$, $1 \leq i \leq ord$ [36].

The implementation of VBDF that provides strict control of the 1-norm of the step approximation error is essentially the one described in [36] except for the way in which the linear system (33) is solved and the mechanism used to adjust the order and the step size. The linear system is solved using GS and right-preconditioned Bi-CGSTAB. The GS is modified as described in Section 3 for splitting-based iterative methods in the case of implicit ODE solvers for which $\mathbf{p}_n$ is the solution to a linear system of the form (9) with $a(n) = \sum_{i=1}^{ord} 1/h_{n-i} > 0$, $n \geq 0$, and Bi-CGSTAB is modified as described in that section for iterative methods that produce the residual after each iteration or a predefined number of them, in the case of implicit ODE solvers for which $\mathbf{p}_n$ is the solution to a linear system of the form (9) with $a(n) = \sum_{i=1}^{ord} 1/h_{n-i} > 0$, $n \geq 0$. Taking into account (9), (33), (11), (10), and recalling that the 1-norm of the strict upper part of the left-hand side matrix in (33) is $\max_{1 \leq i \leq m} \sum_{j=1}^{i-1} q_{j,i}$, those modifications amount to use the stopping criterion

$$\|\mathbf{p}_n^{(l)} - \mathbf{p}_n^{(l-1)}\|_1 \leq \varepsilon \Big(\sum_{i=1}^{ord} \frac{1}{h_{n-i}}\Big) \frac{1}{\max_{1 \leq i \leq m} \sum_{j=1}^{i-1} q_{j,i}}, \quad l \geq 1\,,$$

for GS and the stopping criterion

$$\Big\| \Big(\sum_{i=1}^{ord} \frac{1}{h_{n-i}}\Big)\mathbf{p}_{n,0} - \frac{1}{h_n}\mathbf{p}_{n,0}^{(1)} - \Big(\Big(\sum_{i=1}^{ord} \frac{1}{h_{n-i}}\Big)\mathbf{I}_m - \mathbf{Q}\Big)\mathbf{p}_n^{(l)} \Big\|_1 \leq \varepsilon \Big(\sum_{i=1}^{ord} \frac{1}{h_{n-i}}\Big), \quad l \geq 0\,.$$

for Bi-CGSTAB. In all cases, we set $\varepsilon = tol/10$, where *tol* denotes a user-given local error tolerance.

The GS is used for the first and following steps as long as the method is able to fulfill the corresponding stopping criterion in no more than $4 \times ord$ iterations. The starting values for the iterations are $\mathbf{p}_n^{\{0\}} = \mathbf{p}_{n,0}$. If in some step GS reaches the iteration limit, GS is not used anymore and the step is repeated using right-preconditioned Bi-CGSTAB. The starting values for the iterations and the limit on the number of iterations are the same as for GS. If Bi-CGSTAB breaks down or reaches the iteration limit for any of the involved linear systems, the step size is halved and the step repeated.

The right-preconditioner we use for Bi-CGSTAB is a variant of the incomplete LU factorization with threshold ILUT(p,τ) [39] of the left-hand side matrix of the linear system being solved. In the variant, only the $\lceil p \times nl(i) \rceil$ largest off-diagonal entries of the ith row of the L part of the factorization are kept and only the $\lceil p \times nu(i) \rceil$ largest off-diagonal entries of the ith row of the U part of the factorization are kept, where $nl(i)$ and $nu(i)$ are the number of nonzero off-diagonal entries in the L and U parts of the ith row of the matrix, respectively, and $\lceil x \rceil$ denotes the smallest integer $\geq x$. Besides, the tolerance for the dropping rule for the ith row of the factorization is obtained as the threshold τ times the 1-norm of the ith row of the matrix divided by $nl(i)+nu(i)+1$. The effectiveness of the ILUT(p,τ) factorization tends to increase with p and to decrease with τ [39]. Consequently, we adjust dynamically these parameters so that p is increased and τ decreased when Bi-CGSTAB encounters difficulties and, conversely, p is decreased and τ increased when Bi-CGSTAB performs well. To be precise, we set $p = 1$ and $\tau = 1/(2 \times ord)$ the first time Bi-CGSTAB is used. Then, when a step has to be repeated because Bi-CGSTAB broke down or reached the iteration limit, we (possibly) enlarge p by setting it to $\min\{p\sqrt{ord}, ord\}$ and (possibly) reduce τ by setting it to $\max\{\tau\sqrt{1/20}, 1/(40 \times ord)\}$, and when Bi-CGSTAB succeeds, we (possibly) reduce p by setting it to $\max\{p \times (1/ord)^{1/(ord \times 50)}, 1\}$ and (possibly) enlarge τ by setting it to $\min\{\tau \times 20^{1/(ord \times 50)}, 1/(2 \times ord)\}$. In this way, after at most two repetitions of a step because of poor convergence of Bi-CGSTAB, the parameter p reaches its maximal value ord for the current order of accuracy and the threshold τ reaches its minimal value $1/(40 \times ord)$ for the current order of accuracy, and when Bi-CGSTAB performs well, p and τ are, respectively, reduced and enlarged gradually until they reach the values 1 and $1/(2 \times ord)$. The ILUT(p,τ) factorizations are computed only the first time Bi-CGSTAB is used, when a step has to be repeated, and when the current step size h_n and the step size the last time the factorizations were obtained, $h_{n'}$, are such that $|\log(h_n/h_{n'})| > \log 1.5$. Other than these, the last computed factorizations are used. The rationale behind this strategy is the empirical observation that after a moderate change in the step size, very often the factorizations can be reused with little impact on their effectiveness.

The mechanism used for the automatic selection of step size and order is a combination of the one described in [36] with ideas taken from [40]. Let $\widetilde{\mathbf{e}}_n(ord)$ denote the estimate defined in [36] for the local error at step n, where ord is the order with which the step has been performed. The mechanism is as follows. We start with $ord = 1$, $h_1 = \min\{10^{-6}, t\}$, and $\mathbf{p}_0^{(1)} = \mathbf{Q}\mathbf{p}^0$. After completing a step, say step n, at order ord, it is accepted if $\|\widetilde{\mathbf{e}}_n(ord)\|_1 \leq tol$.[3] In that case, the computation terminates if $t_n + h_n = t$. Otherwise, the guess for the size of the next step at the same order, $h_{n+1,ord}$, is computed as $h_{n+1,ord} = h_n \min\{factmax, \chi_{ord}\}$, where the factor $factmax$ is explained later and $\chi_{ord} = (tol/(6\|\widetilde{\mathbf{e}}_n(ord)\|_1)^{1/(ord+1)}$; the guess for the size of the next step at order $ord - 1$, $h_{n+1,ord-1}$, is computed as $h_{n+1,ord-1} = h_n \min\{factmax, \chi_{ord-1}\}$, where $\chi_{ord-1} = (tol/(6\|\widetilde{\mathbf{e}}_n(ord-1))\|_1)^{1/ord}$ if $ord > 1$ and the last $ord + 1$ or more steps have been performed at order ord, and $\chi_{ord-1} = 0$ otherwise; and the guess for the size of the next step at order $ord + 1$, $h_{n+1,ord+1}$, is computed as $h_{n+1,ord+1} = h_n \min\{factmax, \chi_{ord+1}\}$, where $\chi_{ord+1} = (tol/(10\|\widetilde{\mathbf{e}}_n(ord+1))\|_1)^{1/(ord+2)}$ if $ord < 5$ and the last $ord+1$ or more steps have been performed at order ord, and otherwise $\chi_{ord+1} = 0$. Then, we reduce the order by one and set the size h_{n+1} of the next step to $\min\{h_{n+1,ord-1}, t-t_n\}$ if $h_{n+1,ord-1} \geq \max\{h_{n+1,ord}, h_{n+1,ord+1}\}$, leave the order unchanged and set h_{n+1} to $\min\{h_{n+1,ord}, t-t_n\}$ if $h_{n+1,ord} \geq \max\{h_{n+1,ord-1}, h_{n+1,ord+1}\}$, and otherwise increase the order by one and set h_{n+1} to $\min\{h_{n+1,ord+1}, t-t_n\}$. If $\|\widetilde{\mathbf{e}}_n(ord)\|_1 > tol$, the step fails. In that case, we set h_n to $\min\{h_n \min\{factmax, \chi_{ord}\}, t-t_n\}$ if this is the first time the local error test fails for the step, to $\min\{h_n \min\{0.2, \chi_{ord}\}, t-t_n\}$ if this is the second

[3] In case $\|\widetilde{\mathbf{e}}_n(ord)\|_1$ is numerically zero, if $n \geq 2$, we set $\|\widetilde{\mathbf{e}}_n(ord)\|_1$ to $\|\widetilde{\mathbf{e}}_{n-1}(ord_{n-1})\|_1$, where ord_{n-1} is the order with which step $n-1$ has been performed, and otherwise set $\|\widetilde{\mathbf{e}}_n(ord)\|_1$ to a tiny positive value.

time the local error test fails for the step, and to min $\{h_n \min\{0.1, \chi_{ord}\}, t - t_n\}$ after the third and following failures of the local error test for the step. In this latter case, the order is reduced by one if it is > 1 and otherwise the method is restarted by computing $\mathbf{p}_n^{(1)}$ as $\mathbf{p}_n^{(1)} = \mathbf{Q}\mathbf{p}_{n-1}$. Once the size and order of the failed step have been adjusted, the step is repeated. The factor *factmax* is mainly intended as a safeguard against poor convergence of Bi-CGSTAB and is set as follows. After solving the first step, we set *factmax* to 10 000 if the step has been repeated at most once and otherwise set *factmax* to 1, and after solving any step different from the first one, we set *factmax* to $\min\{10^{1/(50\times ord)} factmax, 10\}$ if the step has not been repeated or it has been repeated once because GS reached the iteration limit and otherwise set *factmax* to 1. In this way, if Bi-CGSTAB encounters difficulties the step size is not allowed to increase and when Bi-CGSTAB performs well, the limit on the rate of growth of the step size is increased gradually until 10. The rationale behind this strategy is the empirical observation that when $\mathbf{p}(t_n)$ is far away from the steady-state probability distribution of X, the larger the step size, which reduces the degree of diagonal dominance of the left-hand side matrices of the linear system (33), the larger is the number of iterations required by Bi-CGSTAB. Finally, to avoid the step size to become unacceptably small, the computation is aborted if for 10 times in succession, $h_n \leq 25 \times \text{EPS} \times t$, where EPS denotes the machine epsilon.

The reasonable standard implementation of VBDF is identical to the implementation just described except for changes in the stopping criteria for GS and Bi-CGSTAB. The new stopping criteria are

$$\|\mathbf{p}_n^{(l)} - \mathbf{p}_n^{(l-1)}\|_1 \leq \frac{tol}{10} \times \frac{1}{100}, \quad l \geq 1\,,$$

for GS and

$$\Big\| \Big(\sum_{i=1}^{ord} \frac{1}{h_{n-i}} \Big) \mathbf{p}_{n,0} - \frac{1}{h_n} \mathbf{p}_{n,0}^{(1)} - \Big(\Big(\sum_{i=1}^{ord} \frac{1}{h_{n-i}} \Big) \mathbf{I}_m - \mathbf{Q} \Big) \mathbf{p}_n^{(l)} \Big\|_1 \leq \frac{tol}{10} \times \frac{1}{1\,000}, \quad l \geq 0\,,$$

for Bi-CGSTAB. With these changes, it is apparent that the standard implementation does not provide strict control of the 1-norm of the step approximation error.

5 Numerical Experiments

This section compares, using two examples, the performance of the implementation of VBDF with strict control of the 1-norm of the step approximation error with the performance of the standard implementation. The examples have been chosen so as to cover two representative scenarios: One in which the spectrum of $\mathbf{Q}$ includes real and complex eigenvalues and another in which all the eigenvalues of $\mathbf{Q}$ are real.

All computations were performed on a workstation equipped with a four-core Intel i7-2630QM 2.00 GHz processor with 4 GB of RAM memory, using only one core. The implementations were coded using the C programming language and were compiled using the standard GNU compiler collection C-compiler [41], which supports complex arithmetic, with the `O2` optimization option. All floating-point computations were carried out using the IEEE 754 [42] double format.

The first example is the CTMC model of a system made up of five identical, independent components. The state of each component is modeled by the CTMC with the state diagram shown in Figure 1, left. For this example, $\mathbf{Q}$ is a matrix of dimension 32 768 with 299 008 nonnull entries and $q = 7.505 \times 10^1\,\text{h}^{-1}$. The eigenvalues of the infinitesimal generator of each component are, up to the sixth digit, $0\,\text{h}^{-1}$, $(-6.91983 \pm \text{j}\,9.53004) \times 10^{-7}\,\text{h}^{-1}$, $(-1.81301 \pm \text{j}\,0.589268) \times 10^{-6}\,\text{h}^{-1}$, $-6.65780 \times 10^{-6}\,\text{h}^{-1}$, $-1.01000\,\text{h}^{-1}$, and $-1.50200 \times 10^1\,\text{h}^{-1}$. Consequently, the two smallest (in absolute value) eigenvalues of $\mathbf{Q}$ are $0\,\text{h}^{-1}$ and $(-6.91983 \pm \text{j}\,9.53004) \times 10^{-7}\,\text{h}^{-1}$ and the largest one is $5 \times (-1.50200 \times 10^1)\,\text{h}^{-1}$. The initial probability distribution vector is one for the state in which each component is in state 3 and zero for the remaining states. To illustrate how $\mathbf{p}(t)$ varies with t, in Figure 1, right we plot the evolution of the entry of $\mathbf{p}(t)$ that corresponds to each component being in state i, denoted $p^{[i]}(t)$, for $1 \leq i \leq 8$. For this example, we target the

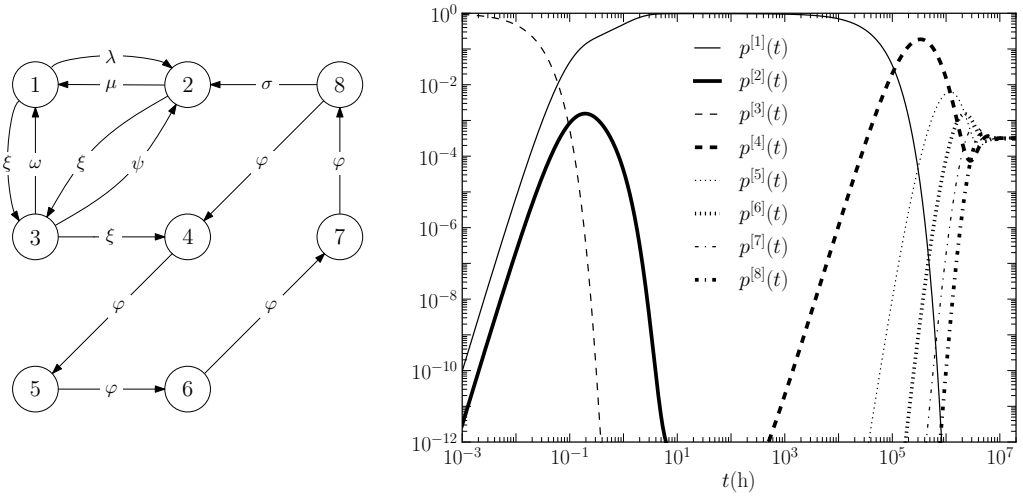

Figure 1: Left: State diagram of the CTMC model of one component of the first example ($\lambda = 2 \times 10^{-6}\,\mathrm{h}^{-1}$, $\mu = 1\,\mathrm{h}^{-1}$, $\xi = 10^{-2}\,\mathrm{h}^{-1}$, $\omega = 10\,\mathrm{h}^{-1}$, $\psi = 5\,\mathrm{h}^{-1}$, $\varphi = 10^{-6}\,\mathrm{h}^{-1}$, $\sigma = 10^{-8}\,\mathrm{h}^{-1}$); right: Probabilities $p^{[i]}(t)$, $1 \leq i \leq 8$, as a function of t in double logarithmic scale for the first example.

computation of $\mathbf{p}(t)$ for $t \in \mathcal{T} = \{10^{-3}, 10^{-2}, \ldots, 10^{8}\}$ h. Therefore, for the largest value of t we consider, we have $qt = 7.505 \times 10^{9}$.

The second example is the CTMC model of a system consisting of six identical, independent components whose infinitesimal generator has real eigenvalues only. The state of each component is modeled by the CTMC with the state diagram shown in Figure 2, left. For this example, $\mathbf{Q}$ is a matrix of dimension 15 625 with 390 625 nonnull entries and $q = 6.0012\,\mathrm{h}^{-1}$. The eigenvalues of the infinitesimal generator of each component are, up to the sixth digit, $0\,\mathrm{h}^{-1}$, $-1.04000 \times 10^{-4}\,\mathrm{h}^{-1}$ $-2.03000 \times 10^{-4}\,\mathrm{h}^{-1}$, $-5.00210 \times 10^{-1}\,\mathrm{h}^{-1}$, and $-1.00021\,\mathrm{h}^{-1}$. Consequently, the two smallest (in absolute value) eigenvalues of $\mathbf{Q}$ are $0\,\mathrm{h}^{-1}$ and $-1.04000 \times 10^{-4}\,\mathrm{h}^{-1}$ and the largest one is $6 \times (-1.00021)\,\mathrm{h}^{-1}$. The initial probability distribution vector is one for the state in which each component is in state 5 and zero for the remaining states. To illustrate how $\mathbf{p}(t)$ varies with t, in Figure 2, right we plot the evolution of $p^{[i]}(t)$, for $1 \leq i \leq 5$. For this example, we target the computation of $\mathbf{p}(t)$ for $t \in \mathcal{T}$. Therefore, for the largest value of t we consider, we have $qt = 6.0012 \times 10^{8}$, so this example is less stiff than the first one.

To compare the implementation with strict control of the 1-norm of the step approximation error with the standard implementation, for each example we obtained $\mathbf{p}(t)$, $t \in \mathcal{T}$, by computing with high accuracy using the Maple (TM) software [43] the probability vector, $\mathbf{p}_{\mathrm{comp}}(t)$, of the CTMC of one component and using the well-known fact that for a CTMC model of a system made up of c identical, independent components, we have $\mathbf{p}(t) = \bigotimes_{i=1}^{c} \mathbf{p}_{\mathrm{comp}}(t)$. Next, we computed the approximate solution $\mathbf{p}_n$, $t_n = t \in \mathcal{T}$, with $tol = 10^{-4}, 10^{-5}, \ldots, 10^{-12}$, using the implementations and measured, for each value of tol, the accuracy, defined as $\max_{t \in \mathcal{T}, t_n = t} \|\mathbf{p}(t) - \mathbf{p}_n\|_1$, and the corresponding cumulative computing time to obtain $\mathbf{p}_n$, $t_n = t \in \mathcal{T}$. The results are given in Figures 3 and 4. As we can see, the implementation of VBDF with strict control of the 1-norm of the step approximation error is more efficient than the standard implementation, in the sense of being able to achieve the same accuracy in less computing time. That reduction in computing time is more noticeable for not too tight local error tolerances. We also observe that the implementation of VBDF with strict control of the 1-norm of the step approximation error provides a better control of the global error, in the sense of yielding an accuracy $\max_{t \in \mathcal{T}, t_n = t} \|\mathbf{p}(t) - \mathbf{p}_n\|_1$ that is closer to tol. To better appreciate this fact, in Figures 5 and 6 we give, for each example, the accuracy as a function of tol. As it can be seen, the accuracy yielded by the implementation of VBDF with strict control of the 1-norm of the step approximation error is never worse than that of the standard

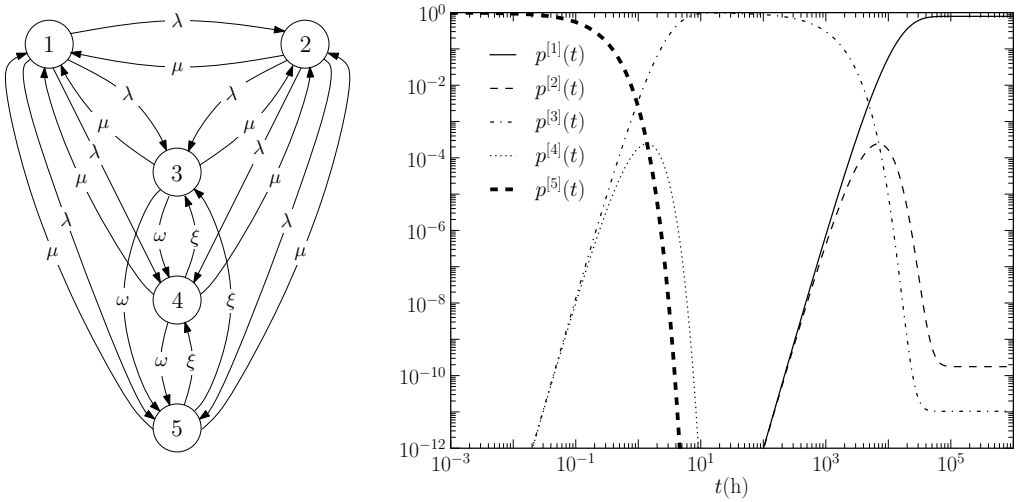

Figure 2: Left: State diagram of the CTMC model of one component of the second example ($\lambda = 1 \times 10^{-6}\,\mathrm{h}^{-1}$, $\mu = 1 \times 10^{-4}\,\mathrm{h}^{-1}$, $\omega = 5 \times 10^{-6}\,\mathrm{h}^{-1}$, $\xi = 5 \times 10^{-1}\,\mathrm{h}^{-1}$); right: Probabilities $p^{[i]}(t)$, $1 \leq i \leq 5$, as a function of t in double logarithmic scale for the second example.

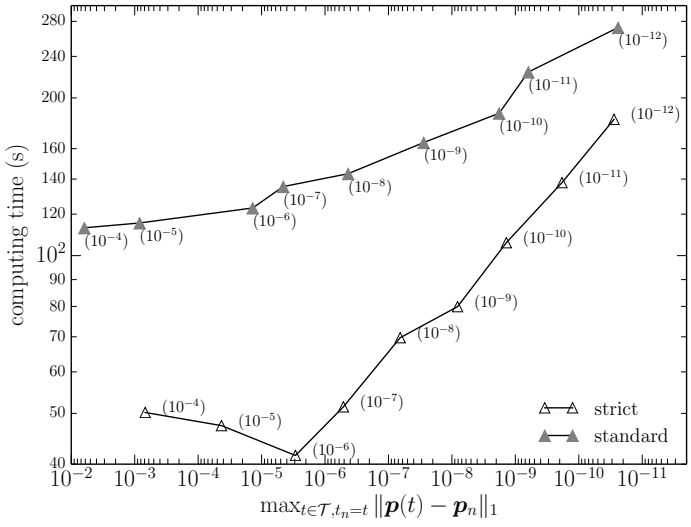

Figure 3: First example: Cumulative computing time to obtain $\mathbf{p}_n$, $t_n = t \in \mathcal{T}$, as a function of the accuracy $\max_{t\in\mathcal{T},t_n=t} \|\mathbf{p}(t) - \mathbf{p}_n\|_1$ for the implementation of VBDF with strict control of the 1-norm of the step approximation error (strict) and the standard implementation (standard). (Next to each point it is given, between parenthesis, the corresponding value of the local error tolerance *tol*.)

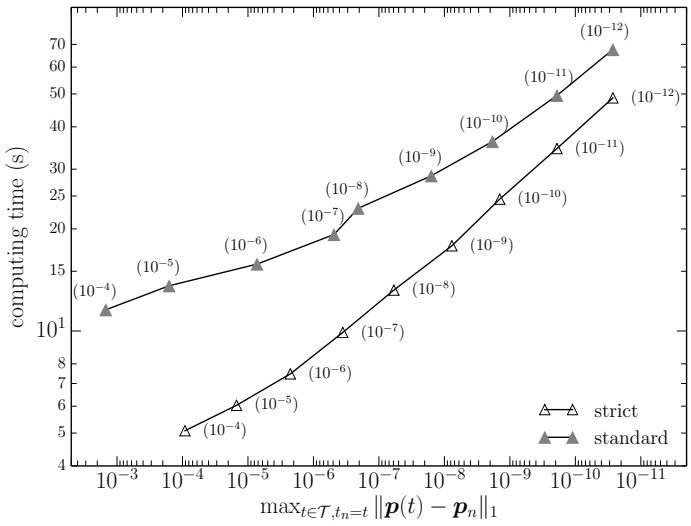

Figure 4: Second example: Cumulative computing time to obtain $\mathbf{p}_n$, $t_n = t \in \mathcal{T}$, as a function of the accuracy $\max_{t\in\mathcal{T},t_n=t} \|\mathbf{p}(t) - \mathbf{p}_n\|_1$ for the implementation of VBDF with strict control of the 1-norm of the step approximation error (strict) and the standard implementation (standard). (Next to each point it is given, between parenthesis, the corresponding value of the local error tolerance *tol*.)

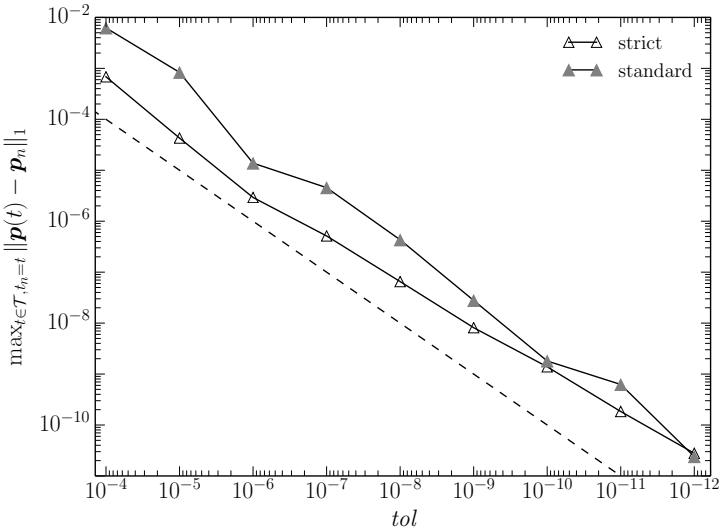

Figure 5: First example: Accuracy $\max_{t\in\mathcal{T},t_n=t} \|\mathbf{p}(t) - \mathbf{p}_n\|_1$ as a function of the local error tolerance *tol* for the implementation of VBDF with strict control of the 1-norm of the step approximation error (strict) and the standard implementation (standard). (The dashed line corresponds to $\max_{t\in\mathcal{T},t_n=t} \|\mathbf{p}(t) - \mathbf{p}_n\|_1 = tol$.)

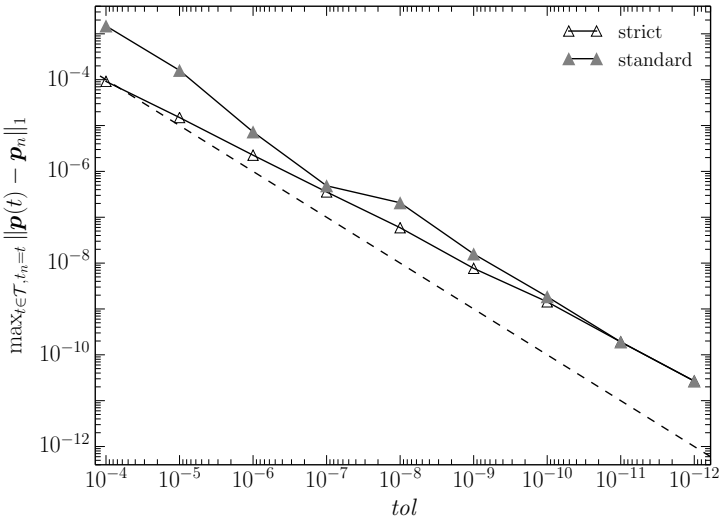

Figure 6: Second example: Accuracy $\max_{t\in\mathcal{T},t_n=t} \|\mathbf{p}(t) - \mathbf{p}_n\|_1$ as a function of the local error tolerance *tol* for the implementation of VBDF with strict control of the 1-norm of the step approximation error (strict) and the standard implementation (standard). (The dashed line corresponds to $\max_{t\in\mathcal{T},t_n=t} \|\mathbf{p}(t) - \mathbf{p}_n\|_1 = tol$.)

implementation and can be noticeably better for not too tight local error tolerances.

6 Conclusions

This paper has identified three classes of implicit ODE solvers such that when applied to the transient analysis of CTMCs, each step can be computed using iterative methods to solve the involved linear systems with strict control of the 1-norm of the step approximation error. Based on these results, an implementation of VBDF with strict control of the 1-norm of the step approximation error has been developed. Using two examples covering two representative scenarios (one in which the spectrum of $\mathbf{Q}$ includes real and complex eigenvalues and another in which all the eigenvalues of $\mathbf{Q}$ are real), it has been shown that the implementation with strict control of the 1-norm of the step approximation error can be more efficient and more accurate than a reasonable standard implementation.

References

[1] Winfried K. Grassmann. Transient solutions in Markovian queueing systems. *Computers & Operations Research*, 4(1):47–53, 1977.

[2] Donald Gross and Douglas R Miller. The randomization technique as a modeling tool and solution procedure for transient Markov processes. *Operations Research*, 32(2):343–361, 1984.

[3] Aad P. A. van Moorsel and William H. Sanders. Adaptive uniformization. *Stochastic Models*, 10(3):619–647, 1994.

[4] Aad P. A. Van Moorsel and William H. Sanders. Transient solution of Markov models by combining adaptive and standard uniformization. *Reliability, IEEE Transactions on*, 46(3):430–440, 1997.

[5] Roger B. Sidje, Kevin Burrage, and Shev MacNamara. Inexact uniformization method for computing transient distributions of Markov chains. *SIAM Journal on Scientific Computing*, 29(6):2562–2580, 2007.

[6] Roger B. Sidje. Expokit: a software package for computing matrix exponentials. *ACM Transactions on Mathematical Software*, 24(1):130–156, 1998.

[7] Semyon Aranovich Gershgorin. Uber die abgrenzung der eigenwerte einer matrix. *Bulletin of the Russian Academy of Sciences-Mathematical series*, (6):749–754, 1931.

[8] Andrew Reibman and Kishor Trivedi. Numerical transient analysis of Markov models. *Computers & Operations Research*, 15(1):19–36, 1988.

[9] Germund G. Dahlquist. A special stability problem for linear multistep methods. *BIT Numerical Mathematics*, 3(1):27–43, 1963.

[10] Manish Malhotra, Jogesh K. Muppala, and Kishor S. Trivedi. Stiffness-tolerant methods for transient analysis of stiff Markov chains. *Microelectronics Reliability*, 34(11):1825–1841, 1994.

[11] Jürgen Dunkel and Harald Stahl. On the transient analysis of stiff Markov chains. In *Dependable Computing for Critical Applications 3*, pages 137–160. Springer, 1993.

[12] Cristoph Lindemann, Manish Malhotra, and Kishor S. Trivedi. Numerical methods for reliability evaluation of Markov closed fault-tolerant systems. *IEEE Transactions on Reliability*, 44(4):694–704, 1995.

[13] Manish Malhotra. A computationally efficient technique for transient analysis of repairable Markovian systems. *Performance Evaluation*, 24:311–331, 1995.

[14] B. Tombuyses and J. Devoogth. Solving Markovian systems of O.D.E. for availability and reliability calculations. *Reliability Engineering and System Safety*, 48:47–55, 1995.

[15] Manish Malhotra. An efficient stiffness-insensitive method for transient analysis of Markov availability models. *IEEE Transactions on Reliability*, 45(3):426–428, 1996.

[16] Roger B. Sidje and William J. Stewart. A numerical study of large sparse matrix exponentials arising in Markov chains. *Computational Statistics & Data Analysis*, 29(3):345–368, 1999.

[17] Randolph E. Bank, William M. Coughran, Wolfgang Fichtner, Eric H. Grosse, Donald J. Rose, and R. Kent Smith. Transient simulation of silicon devices and circuits. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 4(4):436–451, 1985.

[18] Byron L. Ehle. On Padé approximations to the exponential function and A-stable methods for the numerical solution of initial value problems. Technical Report CSRR 2010, Univ. of Waterloo, Dept. AACS, 1969.

[19] E. Fehlberg. Klassische Runge-Kutta-formeln vierter und niedrigerer ordnung mit schrittweiten-kontrolle und ihre anwendung auf wrmeleitungsprobleme. *Computing*, 6(1-2):61–71, 1970.

[20] P. J. Curtois. *Decomposability : queueing and computer system applications.* Academic Press, 1977.

[21] Raymond A Maire, Andrew L Reibman, and Kishor S Trivedi. Transient analysis of acyclic Markov chains. *Performance Evaluation*, 7(3):175–194, 1987.

[22] E. Hairer and G. Wanner. *Solving Ordinary Differential Equations II: Stiff and Differential-Algebraic Problems, 2nd revised edition.* Springer, 1996.

[23] Owe Axelsson. A class of A-stable methods. *BIT Numerical Mathematics*, 9(3):185–199, 1969.

[24] John C. Butcher. Implicit Runge-Kutta processes. *Mathematics of Computation*, 18(85):50–64, 1964.

[25] George D. Byrne. Pragmatic experiments with Krylov methods in the stiff ODE setting. In J. R. Cash and I. Gladwell, editors, *Computational Ordinary Differential Equations*, pages 323–356. Oxford University Press, 1992.

[26] Richard S. Varga. *Matrix iterative analysis*, volume 27 of *Springer Series in Computational Mathematics*. Springer, 2009.

[27] Youcef Saad and Martin H. Schultz. GMRES: A generalized minimal residual algorithm for solving nonsymmetric linear systems. *SIAM Journal on scientific and statistical computing*, 7(3):856–869, 1986.

[28] Peter Sonneveld. CGS, a fast Lanczos-type solver for nonsymmetric linear systems. *SIAM Journal on Scientific and Statistical Computing*, 10(1):36–52, 1989.

[29] R. Barrett, M. Berry, T. F. Chan, J. Demmel, J. Donato, J. Dongarra, V. Eijkhout, R. Pozo, C. Romine, and H. Van der Vorst. *Templates for the Solution of Linear Systems: Building Blocks for Iterative Methods, 2nd Edition*. SIAM, Philadelphia, PA, 1994.

[30] H. A. van der Vorst. Bi-CGSTAB: A fast and smoothly converging variant of Bi-CG for the solution of nonsymmetric linear systems. *SIAM Journal on Scientific and Statistical Computing*, 13(2):631–644, 1992.

[31] J. H. Ahlberg and E. N. Nilson. Convergence properties of the spline fit. *Journal of the Society for Industrial and Applied Mathematics*, 11(1):95–104, 1963.

[32] J. M. Varah. A lower bound for the smallest singular value of a matrix. *Linear Algebra and its Applications*, 11:3–5, 1975.

[33] Richard S Varga. On recurring theorems on diagonal dominance. *Linear Algebra and its Applications*, 13(1):1–9, 1976.

[34] Roberto Bagnara. A unified proof for the convergence of Jacobi and Gauss-Seidel methods. *SIAM review*, 37(1):93–97, 1995.

[35] M. E. Hosea and L. F. Shampine. Analysis and implementation of TR-BDF2. *Applied Numerical Mathematics*, 20(1):21–37, 1996.

[36] George D. Byrne and Alan C. Hindmarsh. A polyalgorithm for the numerical solution of ordinary differential equations. *ACM Transactions on Mathematical Software*, 1(1):71–96, 1975.

[37] C. William Gear. *Numerical initial value problems in ordinary differential equations*. Prentice Hall, 1971.

[38] Kenneth R Jackson and Ron Sacks-Davis. An alternative implementation of variable step-size multistep formulas for stiff ODEs. *ACM Transactions on Mathematical Software*, 6(3):295–318, 1980.

[39] Yousef Saad. ILUT: A dual threshold incomplete LU factorization. *Numerical linear algebra with applications*, 1(4):387–402, 1994.

[40] A. C. Hindmarsh, P. N. Brown, K. E. Grant, S. L. Lee, R. Serban, D. E. Shumaker, and C. S. Woodward. SUNDIALS: Suite of nonlinear and differential/algebraic equation solvers. *ACM Transactions on Mathematical Software*, 31(3):363–396, 2004.

[41] Richard M. Stallman and the GCC Developer Community. *Using the GNU Compiler Collection. For GCC version 4.4.7*. Free Software Foundation, 2012.

[42] IEEE Computer Society. *IEEE Std 754-1985 Standard for Binary Floating-Point Arithmetic*, 1985.

[43] Maple 15. Maplesoft, a division of Waterloo Maple Inc., Waterloo, Ontario., 2011.

Online Supplement of "Implicit ODE Solvers with Good Local Error Control for the Transient Analysis of Markov Models"

Víctor Suñé Juan Antonio Carrasco

September 14, 2016

1 Proof of Proposition 1

Case 1 follows from [1, 2]. If the matrix $\mathbf{V}$ is SCDD, the matrix $\mathbf{V}^{\mathrm{T}}$ is SRDD and, then, from case 1,

$$\|\mathbf{V}^{-1}\|_1 = \|(\mathbf{V}^{-1})^{\mathrm{T}}\|_\infty = \|(\mathbf{V}^{\mathrm{T}})^{-1}\|_\infty \leq \frac{1}{\min_{1\leq i\leq n}\{|v_{i,i}| - \sum_{\substack{j=1\\ j\neq i}}^{n}|v_{j,i}|\}}\,,$$

which is case 2. For case 3, using case 1,

$$\begin{aligned}\|\mathbf{x}-\mathbf{y}\|_\infty &= \|\mathbf{V}^{-1}\mathbf{u} - \mathbf{V}^{-1}\mathbf{V}\mathbf{y}\|_\infty\\ &= \|\mathbf{V}^{-1}(\mathbf{u}-\mathbf{V}\mathbf{y})\|_\infty\\ &\leq \|\mathbf{V}^{-1}\|_\infty\|\mathbf{u}-\mathbf{V}\mathbf{y}\|_\infty\\ &\leq \frac{\|\mathbf{u}-\mathbf{V}\mathbf{y}\|_\infty}{\min_{1\leq i\leq n}\{|v_{i,i}| - \sum_{\substack{j=1\\ j\neq i}}^{n}|v_{i,j}|\}}\,.\end{aligned}$$

Finally, for case 4, using case 2,

$$\begin{aligned}\|\mathbf{x}-\mathbf{y}\|_1 &= \|\mathbf{V}^{-1}\mathbf{u} - \mathbf{V}^{-1}\mathbf{V}\mathbf{y}\|_1\\ &= \|\mathbf{V}^{-1}(\mathbf{u}-\mathbf{V}\mathbf{y})\|_1\\ &\leq \|\mathbf{V}^{-1}\|_1\|\mathbf{u}-\mathbf{V}\mathbf{y}\|_1\\ &\leq \frac{\|\mathbf{u}-\mathbf{V}\mathbf{y}\|_1}{\min_{1\leq i\leq n}\{|v_{i,i}| - \sum_{\substack{j=1\\ j\neq i}}^{n}|v_{j,i}|\}}\,. \quad \square\end{aligned}$$

2 Proof of Theorem 1

If the matrix $\mathbf{V}$ is SRDD, by case 3 of Proposition 1 with $\mathbf{y} = \mathbf{x}^{(l)}$,

$$\|\mathbf{x}-\mathbf{x}^{(l)}\|_\infty \leq \frac{\|\mathbf{u}-\mathbf{V}\mathbf{x}^{(l)}\|_\infty}{\min_{1\leq i\leq n}\{|v_{i,i}| - \sum_{\substack{j=1\\ j\neq i}}^{n}|v_{i,j}|\}} \leq \frac{\delta\min_{1\leq i\leq n}\{|v_{i,i}| - \sum_{\substack{j=1\\ j\neq i}}^{n}|v_{i,j}|\}}{\min_{1\leq i\leq n}\{|v_{i,i}| - \sum_{\substack{j=1\\ j\neq i}}^{n}|v_{i,j}\}|} = \delta\,,$$

and, if the matrix $\mathbf{V}$ is SCDD, by case 4 of Proposition 1 with $\mathbf{y} = \mathbf{x}^{(l)}$,

$$\|\mathbf{x}-\mathbf{x}^{(l)}\|_1 \leq \frac{\|\mathbf{u}-\mathbf{V}\mathbf{x}^{(l)}\|_1}{\min_{1\leq i\leq n}\{|v_{i,i}| - \sum_{\substack{j=1\\ j\neq i}}^{n}|v_{j,i}|\}} \leq \frac{\delta\min_{1\leq i\leq n}\{|v_{i,i}| - \sum_{\substack{j=1\\ j\neq i}}^{n}|v_{j,i}|\}}{\min_{1\leq i\leq n}\{|v_{i,i}| - \sum_{\substack{j=1\\ j\neq i}}^{n}|v_{j,i}|\}} = \delta\,. \quad \square$$

3 Proof of Theorem 2

Assume $l \geq 1$. Using $\mathbf{V} = \mathbf{M} - \mathbf{N}$ and (3) $\mathbf{x}^{(l)} = \mathbf{M}^{-1}\mathbf{N}\mathbf{x}^{(l-1)} + \mathbf{M}^{-1}\mathbf{u}$, we obtain

$$\begin{aligned}
\mathbf{u} - \mathbf{V}\mathbf{x}^{(l)} &= \mathbf{u} - (\mathbf{M} - \mathbf{N})\mathbf{x}^{(l)} \\
&= \mathbf{u} - \mathbf{M}\mathbf{x}^{(l)} + \mathbf{N}\mathbf{x}^{(l)} \\
&= \mathbf{u} - \mathbf{M}(\mathbf{M}^{-1}\mathbf{N}\mathbf{x}^{(l-1)} + \mathbf{M}^{-1}\mathbf{u}) + \mathbf{N}\mathbf{x}^{(l)} \\
&= \mathbf{u} - \mathbf{N}\mathbf{x}^{(l-1)} - \mathbf{u} + \mathbf{N}\mathbf{x}^{(l)} \\
&= \mathbf{N}(\mathbf{x}^{(l)} - \mathbf{x}^{(l-1)})\,. \qquad (34)
\end{aligned}$$

Then, if the matrix $\mathbf{V}$ is SRDD, when the method is stopped we have, using (34) and case 3 of Proposition 1 with $\mathbf{y} = \mathbf{x}^{(l)}$,

$$\begin{aligned}
\|\mathbf{x} - \mathbf{x}^{(l)}\|_\infty &\leq \frac{\|\mathbf{u} - \mathbf{V}\mathbf{x}^{(l)}\|_\infty}{\min_{1\leq i\leq n}\{|v_{i,i}| - \sum_{\substack{j=1\\ j\neq i}}^{n} |v_{i,j}|\}} \\
&= \frac{\|\mathbf{N}(\mathbf{x}^{(l)} - \mathbf{x}^{(l-1)})\|_\infty}{\min_{1\leq i\leq n}\{|v_{i,i}| - \sum_{\substack{j=1\\ j\neq i}}^{n} |v_{i,j}|\}} \\
&\leq \frac{\|\mathbf{N}\|_\infty \|\mathbf{x}^{(l)} - \mathbf{x}^{(l-1)}\|_\infty}{\min_{1\leq i\leq n}\{|v_{i,i}| - \sum_{\substack{j=1\\ j\neq i}}^{n} |v_{i,j}|\}} \\
&\leq \frac{\|\mathbf{N}\|_\infty\, \delta\, \min_{1\leq i\leq n}\{|v_{i,i}| - \sum_{\substack{j=1\\ j\neq i}}^{n} |v_{i,j}|\}/\|\mathbf{N}\|_\infty}{\min_{1\leq i\leq n}\{|v_{i,i}| - \sum_{\substack{j=1\\ j\neq i}}^{n} |v_{i,j}|\}} \\
&= \delta\,,
\end{aligned}$$

and, if the matrix $\mathbf{V}$ is SCDD, when the method is stopped we have, using (34) and case 4 of Proposition 1 with $\mathbf{y} = \mathbf{x}^{(l)}$,

$$\begin{aligned}
\|\mathbf{x} - \mathbf{x}^{(l)}\|_1 &\leq \frac{\|\mathbf{u} - \mathbf{V}\mathbf{x}^{(l)}\|_1}{\min_{1\leq i\leq n}\{|v_{i,i}| - \sum_{\substack{j=1\\ j\neq i}}^{n} |v_{j,i}|\}} \\
&= \frac{\|\mathbf{N}(\mathbf{x}^{(l)} - \mathbf{x}^{(l-1)})\|_1}{\min_{1\leq i\leq n}\{|v_{i,i}| - \sum_{\substack{j=1\\ j\neq i}}^{n} |v_{j,i}|\}} \\
&\leq \frac{\|\mathbf{N}\|_1 \|\mathbf{x}^{(l)} - \mathbf{x}^{(l-1)}\|_1}{\min_{1\leq i\leq n}\{|v_{i,i}| - \sum_{\substack{j=1\\ j\neq i}}^{n} |v_{j,i}|\}} \\
&\leq \frac{\|\mathbf{N}\|_1\, \delta\, \min_{1\leq i\leq n}\{|v_{i,i}| - \sum_{\substack{j=1\\ j\neq i}}^{n} |v_{j,i}|\}/\|\mathbf{N}\|_1}{\min_{1\leq i\leq n}\{|v_{i,i}| - \sum_{\substack{j=1\\ j\neq i}}^{n} |v_{j,i}|\}} \\
&= \delta\,. \quad \square
\end{aligned}$$

4 Proof of Proposition 2

Clearly, $v_{i,i} = \chi + \mathrm{j}\omega - \xi q_{i,i} = \chi + \mathrm{j}\omega + \xi|q_{i,i}|$ and $v_{i,j} = -\xi q_{i,j}$, $i \neq j$. Then, using the fact that $|q_{i,i}| = \sum_{\substack{j=1\\ j\neq i}}^{m} q_{j,i}$ and that, as assumed, $\chi\,\xi > 0$, or $\chi = 0$, $\omega \neq 0$,

$$|v_{i,i}| = \big|\chi + \mathrm{j}\omega + \xi|q_{i,i}|\big| > |\xi||q_{i,i}| = |\xi| \sum_{\substack{j=1\\ j\neq i}}^{m} q_{j,i} = \sum_{\substack{j=1\\ j\neq i}}^{m} |v_{j,i}|\,, \qquad (35)$$

implying that the matrix is SCDD. The equality asserted by the proposition can be justified as follows. Since (35) $|v_{i,i}| - \sum_{\substack{j=1\\ j\neq i}}^{m} |v_{j,i}| = |\chi + \mathrm{j}\omega + \xi|q_{i,i}|| - |\xi||q_{i,i}|$,

$$\min_{1\leq i\leq m} \left(|v_{i,i}| - \sum_{\substack{j=1\\ j\neq i}}^{m} |v_{j,i}| \right) = \min_{1\leq i\leq m} \left(|\chi + \mathrm{j}\omega + \xi|q_{i,i}|| - |\xi||q_{i,i}| \right).$$

Therefore, if $\xi > 0$,

$$\min_{1\leq i\leq m} \left(|v_{i,i}| - \sum_{\substack{j=1\\ j\neq i}}^{m} |v_{j,i}| \right) = \min_{1\leq i\leq m} (|\chi + \mathrm{j}\omega + \xi|q_{i,i}|| - \xi|q_{i,i}|) = |\chi + \mathrm{j}\omega + \xi q| - \xi q\,, \tag{36}$$

because, for x real, the function $|\chi + \mathrm{j}\omega + x| - x$ is (non-strictly) decreasing on x and, for $\xi > 0$, $\max_{1\leq i\leq m} \xi|q_{i,i}| = \xi q$. Similarly, if $\xi \leq 0$,

$$\min_{1\leq i\leq m} \left(|v_{i,i}| - \sum_{\substack{j=1\\ j\neq i}}^{m} |v_{j,i}| \right) = \min_{1\leq i\leq m} (|\chi + \mathrm{j}\omega + \xi|q_{i,i}|| + \xi|q_{i,i}|) = |\chi + \mathrm{j}\omega + \xi q| + \xi q \tag{37}$$

because, for x real, the function $|\chi + \mathrm{j}\omega + x| + x$ is (non-strictly) increasing on x and, for $\xi \leq 0$, $\min_{1\leq i\leq m} \xi|q_{i,i}| = \xi q$. Finally, combining (36), (37),

$$\min_{1\leq i\leq m} \left(|v_{i,i}| - \sum_{\substack{j=1\\ j\neq i}}^{m} |v_{j,i}| \right) = |\chi + \mathrm{j}\omega + \xi q| - |\xi| q\,. \quad \square$$

5 Proof of Proposition 3

Assume $1 \leq k \leq n$. We start by noting that the vectors $\mathbf{x}_k$ and $\mathbf{x}_k^*$ are well-defined because, from Proposition 2 with $\chi = \chi_{k,k}$, $\omega = 0$, and $\xi = \xi_{k,k}$, it follows that the matrix $\mathbf{V}_k = \chi_{k,k}\mathbf{I}_m - \xi_{k,k}\mathbf{Q}$ is SCDD and, consequently, nonsingular. Besides, also from that proposition, $\min_{1\leq i\leq m}\{|v_{i,i}| - \sum_{\substack{j=1\\ j\neq i}}^{m} |v_{j,i}|\} = |\chi_{k,k} + \xi_{k,k}q| - |\xi_{k,k}|q = |\chi_{k,k}|$. Therefore, by case 2 of Proposition 1,

$$\|\mathbf{V}_k^{-1}\|_1 \leq \frac{1}{|\chi_{k,k}|}\,. \tag{38}$$

We can now prove the inequality asserted by the proposition. We have

$$\mathbf{V}_k\mathbf{x}_k = \mathbf{u}_k + \sum_{j=1}^{k-1} (\xi_{k,j}\mathbf{Q} + \chi_{k,j}\mathbf{I}_m)\mathbf{x}_j,$$
$$\mathbf{V}_k\mathbf{x}_k^* = \mathbf{u}_k + \sum_{j=1}^{k-1} (\xi_{k,j}\mathbf{Q} + \chi_{k,j}\mathbf{I}_m)\widetilde{\mathbf{x}}_j^*\,.$$

Therefore,

$$\begin{aligned}\mathbf{x}_k - \mathbf{x}_k^* &= \mathbf{V}_k^{-1}\left(\mathbf{u}_k + \sum_{j=1}^{k-1} (\xi_{k,j}\mathbf{Q} + \chi_{k,j}\mathbf{I}_m)\mathbf{x}_j \right) - \mathbf{V}_k^{-1}\left(\mathbf{u}_k + \sum_{j=1}^{k-1} (\xi_{k,j}\mathbf{Q} + \chi_{k,j}\mathbf{I}_m)\widetilde{\mathbf{x}}_j^* \right) \\ &= \mathbf{V}_k^{-1}\mathbf{Q}\sum_{j=1}^{k-1} \xi_{k,j}(\mathbf{x}_j - \widetilde{\mathbf{x}}_j^*) + \mathbf{V}_k^{-1}\sum_{j=1}^{k-1} \chi_{k,j}(\mathbf{x}_j - \widetilde{\mathbf{x}}_j^*)\,,\end{aligned}$$

which implies

$$\|\mathbf{x}_k - \widetilde{\mathbf{x}}_k^*\|_1 = \|\mathbf{x}_k - \mathbf{x}_k^* + \mathbf{x}_k^* - \widetilde{\mathbf{x}}_k^*\|_1$$

$$
\begin{aligned}
&\leq \|\mathbf{x}_k - \mathbf{x}_k^*\|_1 + \|\mathbf{x}_k^* - \widetilde{\mathbf{x}}_k^*\|_1 \\
&\leq \|\mathbf{V}_k^{-1}\mathbf{Q}\|_1 \sum_{j=1}^{k-1} |\xi_{k,j}| \|\mathbf{x}_j - \widetilde{\mathbf{x}}_j^*\|_1 + \|\mathbf{V}_k^{-1}\|_1 \sum_{j=1}^{k-1} |\chi_{k,j}| \|\mathbf{x}_j - \widetilde{\mathbf{x}}_j^*\|_1 \\
&\quad + \|\mathbf{x}_k^* - \widetilde{\mathbf{x}}_k^*\|_1 \,. \qquad (39)
\end{aligned}
$$

Then, using $\|\mathbf{Q}\|_1 = 2q$, the fact that

$$
\mathbf{V}_k^{-1}\mathbf{Q} = \mathbf{V}_k^{-1}(\chi_{k,k}\mathbf{I}_m - \mathbf{V}_k)\frac{1}{\xi_{k,k}} = \frac{\chi_{k,k}}{\xi_{k,k}}\mathbf{V}_k^{-1} - \frac{1}{\xi_{k,k}}\mathbf{I}_m \,,
$$

and (38),

$$
\|\mathbf{V}_k^{-1}\mathbf{Q}\|_1 \leq \min\left\{\|\mathbf{V}_k^{-1}\|_1 \|\mathbf{Q}\|_1, \left|\frac{\chi_{k,k}}{\xi_{k,k}}\right| \|\mathbf{V}_k^{-1}\|_1 + \frac{1}{|\xi_{k,k}|}\right\} \leq 2\min\left\{\frac{q}{|\chi_{k,k}|}, \frac{1}{|\xi_{k,k}|}\right\}. \qquad (40)
$$

Finally, combining (39), (40), (38),

$$
\begin{aligned}
\|\mathbf{x}_k - \widetilde{\mathbf{x}}_k^*\|_1 &\leq 2\min\left\{\frac{q}{|\chi_{k,k}|}, \frac{1}{|\xi_{k,k}|}\right\} \sum_{j=1}^{k-1} |\xi_{k,j}| \|\mathbf{x}_j - \widetilde{\mathbf{x}}_j^*\|_1 + \frac{1}{|\chi_{k,k}|} \sum_{j=1}^{k-1} |\chi_{k,j}| \|\mathbf{x}_j - \widetilde{\mathbf{x}}_j^*\|_1 \\
&\quad + \|\mathbf{x}_k^* - \widetilde{\mathbf{x}}_k^*\|_1 \\
&= \sum_{j=1}^{k-1} \left(2|\xi_{k,j}| \min\left\{\frac{q}{|\chi_{k,k}|}, \frac{1}{|\xi_{k,k}|}\right\} + \frac{|\chi_{k,j}|}{|\chi_{k,k}|}\right) \|\mathbf{x}_j - \widetilde{\mathbf{x}}_j^*\|_1 + \|\mathbf{x}_k^* - \widetilde{\mathbf{x}}_k^*\|_1 \,. \quad \square
\end{aligned}
$$

References

[1] J. H. Ahlberg and E. N. Nilson. Convergence properties of the spline fit. *Journal of the Society for Industrial and Applied Mathematics*, 11(1):95–104, 1963.

[2] J. M. Varah. A lower bound for the smallest singular value of a matrix. *Linear Algebra and its Applications*, 11:3–5, 1975.