Numerische Mathematik (2019) 143:489–527
https://doi.org/10.1007/s00211-019-01060-8

Numerische Mathematik

Convergence analysis of a *periodic-like* waveform relaxation method for initial-value problems via the diagonalization technique

Martin J. Gander[1] · **Shu-Lin Wu**[2]

Received: 21 November 2017 / Revised: 30 November 2018 / Published online: 27 June 2019

Abstract

We present in this paper a time parallel algorithm for $\dot{u} = f(t, u)$ with initial-value $u(0) = u_0$, by using the waveform relaxation (WR) technique, and the diagonalization technique. With a suitable parameter α, the WR technique generates a functional sequence $\{u^k(t)\}$ via the dynamic iterations $\dot{u}^k = f(t, u^k)$, $u^k(0) = \alpha u^k(T) - \alpha u^{k-1}(T) + u_0$, and at convergence we get $u^\infty(t) = u(t)$. Each WR iterate represents a *periodic-like* differential equation, which is very suitable for applying the diagonalization technique yielding direct parallel-in-time computation. The parameter α controls both the roundoff error arising from the diagonalization procedure and the convergence factor of the WR iterations, and we perform a detailed analysis for the influence of the parameter α on the method. We show that the roundoff error is proportional to $\epsilon(2N + 1)\max\{|\alpha|^2, |\alpha|^{-2}\}$ ($N = T/\Delta t$ and ϵ is the machine precision), and the convergence factor can be bounded by $|\alpha|e^{-TL}/(1 - |\alpha|e^{-TL})$, where $L \geq 0$ is the one-sided Lipschitz constant of f. We also perform a convergence analysis at the discrete level and the effect of temporal discretizations is explored. Our analysis includes the heat and wave equations as special cases. Numerical results are given to support our findings.

Mathematics Subject Classification 65R20 · 45L05 · 65L20 · 68Q60

Corresponding author: Shu-Lin Wu.

✉ Shu-Lin Wu
wushulin84@hotmail.com

Martin J. Gander
martin.gander@unige.ch

[1] Section de Mathématiques, University of Geneva, 1211 Geneva, Switzerland

[2] School of Mathematics and Statistics, Northeast Normal University, Changchun 130024, China

1 Introduction

We are interested in the parallel-in-time solution of initial-value differential equations:

$$\dot{u} = f(t,u),\ u(0) = u_0, \tag{1.1}$$

where $t \in (0,T)$ and $f : \mathbb{R}^+ \times \mathbb{R}^m \to \mathbb{R}^m$. For parallel solution of time-dependent partial differential equations (PDEs), the time direction offers a further possibility for parallelization when parallelization in space saturates. Algorithms trying to use the time direction for parallelization therefore attracted a lot of attention during the last few years. Among these, we mention the widely studied parareal algorithm [9,11,28,32,46,47] and the methods based on Laplace inversion [27,33,34,39,40,45]. The parareal algorithm is a temporal two-grid method which iteratively approximates the solution of (1.1) by using two time-integrators. The Laplace inversion technique is based on representing the exact solution via a contour integral and then discretizing such an integral by contour quadrature (e.g., the Trapezoidal rule). This technique is directly parallelizable, but it is limited to linear problems so far. Other efforts towards parallel-in-time computation for differential equations include the PARAEXP algorithm [10] and the space-time multi-grid methods [4,8] and the parallel preconditioner technique [29]. For an overview, see [12].

In this paper, we present a new parallel-in-time algorithm for solving (1.1), which is based on the waveform relaxation (WR) technique [22,26,30,35,36,38] and the diagonalization technique [6,7,31]. The idea can be summarized as follows. First, by noticing that $u(0) = \alpha u(T) - \alpha u(T) + u_0$ holds for all $\alpha \in \mathbb{R}$, we construct the following WR iterations:

$$\begin{cases} \dot{u}^k = f(t,u^k),\ t \in (0,T), \\ u^k(0) = \alpha u^k(T) - \alpha u^{k-1}(T) + u_0, \end{cases} \tag{1.2}$$

where $k \geq 1$ is the iteration index and $u^0(t)$ denotes the initial guess. Then, for each iteration of (1.2), which is a differential equation with *periodic-like* condition, the diagonalization technique is applied to carry out direct parallel-in-time computation. A detailed description of such a diagonalization technique is given in Sect. 2.

In summary, the strategy is to reduce the numerical computation of (1.1) to solve a series of periodic-like differential equations. The key point behind this idea is to obtain a better diagonalization technique than that in [6,7,31], where the diagonalization technique is directly applied to (1.1) and then one has to use different step-sizes, e.g., the geometrically increasing step-sizes $\{\Delta t_n\}_{n=1}^N$ chosen as $\Delta t_n = \nu^{n-1}\Delta t$, where $\nu > 1$ is a free parameter and Δt is some reference step-size. For such a direct application of the diagonalization technique, as shown in [6] it is rather difficult to balance the roundoff error arising from the diagonalization procedure and the discretization error due to the non-uniform step-sizes. In particular, to make the former small, ν should be larger than 1 as much as possible, while to make the later small, ν has to close to 1. Even though the authors in [6] proposed an excellent idea towards balancing these two errors by optimizing the parameter ν, a direct application of the diagonalization technique only works well for the case that N is small (N denotes the number of discrete time points). Generally speaking, for long time simulation the diagonalization

technique itself does not give good numerical results and we have to combine it with the so-called *window* technique.

On the contrary, as we will show in Sect. 2.1 the diagonalization technique is much more suitable for periodic-like differential equations, for at least four reasons. First, for this kind of problems, we can use uniform step-sizes and therefore the diagonalization procedure does not deteriorate the discretization error. Second, it is proved that the aforementioned roundoff error is proportional to $\epsilon(2N+1)\max\{|\alpha|^{-2}, |\alpha|^2\}$, where $N = T/\Delta t$ and ϵ denotes the machine precision. This implies that the roundoff error can be well controlled in practice. Third, the Fast Fourier Transform (FFT) technique is directly applicable, while for initial-value problems this is not the case. We will address this issue in detail in Sect. 4. Last, because uniform step-sizes are permitted, we can extend the diagonalization technique to the widely used Trapezoidal rule, which results in the well-known Crank-Nicolson scheme in PDE numerics. For initial-value problems, the authors in [6,7,31] only considered the Backward-Euler method and as we will comment in Remark 2.1 it seems impossible to make a generalization to the Trapezoidal rule.

For initial-value problems, the WR technique has been extensively studied in the past thirty years; see the numerous papers citing [26]. However, according to our best knowledge, it is the first time to consider a WR method of the form (1.2). Previous studies are based on the idea of *system partitioning*: by choosing a function $H(t,u,v)$ satisfying the consistency condition $H(t,u,u) = f(t,u)$, we solve (1.1) via the following iterations:

$$\begin{cases} \dot{u}^k = H(t,u^k,u^{k-1}),\ t\in(0,T), \\ u^k(0) = u_0. \end{cases} \tag{1.3}$$

Finding a suitable partitioning is, however, not always easy. For this point, we cite from [35]: "*In practice one is interested in knowing what subdivisions yield fast convergence for the iterations.....The splitting into subsystems is assumed to be given. How to split in such a way that the coupling remains 'weak' is an important question*". Great efforts have been devoted to linear problems $f(t,u) = -Au + \tilde{f}(t)$ with $A\in\mathbb{R}^{m\times m}$, for which the WR method is

$$\begin{cases} \dot{u}^k + A_1u^k = A_2u^{k-1} + \tilde{f},\ t\in(0,T), \\ u^k(0) = u_0, \end{cases} \tag{1.4}$$

where $A_1, A_2 \in \mathbb{R}^{m\times m}$ satisfy $A = A_1 - A_2$. This is a natural extension of the stationary iterative methods for systems of algebraic equations. The Jacobi, Gauss-Seidel and SOR WR methods were proved to be convergent, with convergence rates very similar to those of the corresponding stationary version (see [43]). An analogous study for multigrid acceleration of the WR iterations was also extensively studied; see [25] and the work by Vandewalle and his colleagues [16,20,21,43]. Parallel-in-time implementation of the classical WR method (1.4) was explored by Vandewalle *et al.* in [15,42–44]. The idea is based on making a point-wise Jacobi or Gauss-Seidel partition of A, by which the m *scalar* ODEs make up the kernel of computation of (1.4). After temporal discretization by, e.g., multi-step finite difference methods, the evolution of each scalar discrete ODE can be parallelized by *cyclic reduction* [3,23,24], which is a

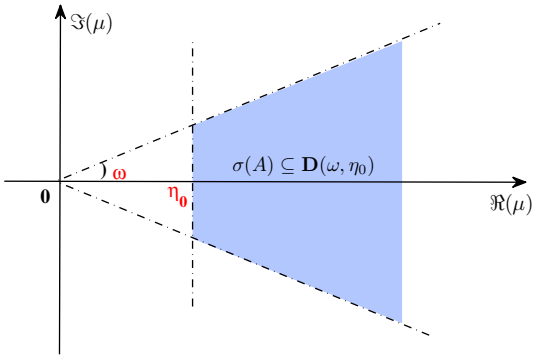

Fig. 1 An illustration of the distribution of the spectrum $\sigma(A)$ in the μ-plane, where $\mu \in \sigma(A)$ denotes an arbitrary eigenvalue of A. The parameters η_0 and ω control the shape of the region $\mathbf{D}$

direct parallel strategy and the parallelism is achieved without increasing the order of the serial complexity.

For the new WR method (1.2), except the potential for direct parallel-in-time implementation, we will show that it also has a robust convergence factor with respect to the discretization parameters. We study the method for linear problems $\dot{u} + Au = \tilde{f}(t)$ both at the continuous level and discrete level, under the assumption that A is diagonalizable and the spectrum $\sigma(A)$ is distributed in a region $\mathbf{D}(\omega, \eta_0)$; see Fig. 1 for illustration:

$$\begin{aligned}\sigma(A) \subseteq \mathbf{D}(\omega, \eta_0) := \{\mu = x + iy : x \geq 0, |y| \leq \tan(\omega)x\} \\ \cap \{\mu = x + iy : x \geq \eta_0, y \in \mathbb{R}\},\end{aligned} \tag{1.5}$$

where $\eta_0 \geq 0$ and $\omega \in [0, \frac{\pi}{2}]$. We will give a precise and explicit relation between the convergence factor and the parameters α, ω and η_0. At the continuous level, we show that the proposed WR method has a constant convergence factor, which can be bounded by $|\alpha|e^{-T\eta_0}/(1 - |\alpha|e^{-T\eta_0})$. At the discrete level we consider two representative temporal discretizations: the Backward-Euler method and the Trapezoidal rule, and we show that these two time-integrators have different effects on the convergence factor. In particular, for the Backward-Euler method, we show that in the asymptotic sense (i.e., when Δt is small) the convergence factor can be bounded by $|\alpha|e^{-T\eta_0}/(1-|\alpha|e^{-T\eta_0})$, while for the Trapezoidal rule this bound becomes $|\alpha|/(1-|\alpha|)$. Our analysis includes the heat and wave equations as special cases.

We also give a convergence analysis of the WR method (1.2) in the nonlinear case at the continuous level, with the assumption that f satisfies the *one-sided* Lipschitz condition:

$$\langle f(t, u_1) - f(t, u_2), u_1 - u_2\rangle \leq -L\|u_1 - u_2\|_2, \forall t \in [0, T], u_{1,2} \in \mathbb{R}^m, \tag{1.6}$$

where $L \geq 0$ is a constant and $\langle\bullet\rangle$ denotes the Euclidean inner product. For (1.6), we show that the convergence factor of the WR method (1.2) can be bounded by $|\alpha|e^{-TL}/(1 - |\alpha|e^{-TL})$.

The rest of this paper is organized as follows: in Sect. 2, we describe the diagonalization technique for each iteration of (1.2), i.e., the periodic-like differential equation.

An analysis of the roundoff error arising from the diagonalization procedure is also given in this section. In Sects. 3 and 4 , we address the convergence properties and give a speedup analysis of the WR method (1.2) in the linear case. The convergence analysis of the WR method (1.2) in the nonlinear case is given in Sect. 5. In Sect. 6, we present several numerical examples to support our theoretical results, and we conclude this paper with comments in Sect. 7.

2 Discretization and diagonalization

To discretize the differential equation in (1.2), we consider the linear θ-method,

$$\begin{cases} \frac{u_n^k - u_{n-1}^k}{\Delta t} = \theta f(t_n, u_n^k) + (1-\theta) f(t_{n-1}, u_{n-1}^k), \\ u_0^k = \alpha u_N^k + R^{k-1} \text{ with } R^{k-1} := -\alpha u_N^{k-1} + u_0, \end{cases} \tag{2.1}$$

where $n = 1, 2, \ldots, N := \frac{T}{\Delta t}$ and $\theta \in [0, 1]$. In (2.1), the choices $\theta = 1$ and $\theta = \frac{1}{2}$ are of particular interest. The former choice corresponds to the Backward-Euler method with global truncation error of order $O(\Delta t)$ over $[0, T]$, and the latter corresponds to the Trapezoidal rule with global truncation error of order $O(\Delta t^2)$ over $[0, T]$.

We next introduce the diagonalization-based implementation of (2.1). Since the diagonalization technique has essential differences for $\alpha \neq 0$ and $\alpha = 0$, we consider these two cases separately for linear problems in Sects. 2.1 and 2.2. The nonlinear case will be addressed in Sect. 2.3. Throughout this section, we denote by $I_x \in \mathbb{R}^{m\times m}$ and $I_t \in \mathbb{R}^{N\times N}$ the identity matrices.

2.1 Diagonalization in the linear case: $\alpha \neq 0$

For the case $f(t, u) = -Au + \tilde{f}(t)$ with $A \in \mathbb{R}^{m\times m}$, we can rewrite (2.1) as

$$(B_1 \otimes I_x + B_2 \otimes A)\, U^k = F, \tag{2.2a}$$

where $U^k = (u_1^k, u_2^k, \ldots, u_N^k)^\top$, $B_1, B_2 \in \mathbb{R}^{N\times N}$ and $F \in \mathbb{R}^{mN}$ are given by

$$\begin{aligned} B_1 &= \frac{1}{\Delta t}\begin{bmatrix} 1 & & & -\alpha \\ -1 & 1 & & \\ & \ddots & \ddots & \\ & & -1 & 1 \end{bmatrix}, \quad B_2 = \begin{bmatrix} \theta & & & (1-\theta)\alpha \\ 1-\theta & \theta & & \\ & \ddots & \ddots & \\ & & 1-\theta & \theta \end{bmatrix}, \\ F &= \left(R^{k-1}\left(\frac{1}{\Delta t} I_x + (1-\theta)A\right) + \bar{f}_1, \bar{f}_2, \ldots, \bar{f}_N \right)^\top \\ &\quad \text{with } \bar{f}_n := \theta \tilde{f}_n + (1-\theta)\tilde{f}_{n-1}. \end{aligned} \tag{2.2b}$$

Clearly, with the matrix $B(\alpha,\tau)$ defined by

$$B(\alpha,\tau) := \begin{bmatrix} 1 & & & \tau\alpha \\ \tau & 1 & & \\ & \ddots & \ddots & \\ & & \tau & 1 \end{bmatrix} \in \mathbb{R}^{N\times N}, \tag{2.3}$$

we have $B_1 = \frac{1}{\Delta t} B(\alpha,-1)$ and $B_2 = \theta B(\alpha, \frac{1-\theta}{\theta})$.

Lemma 2.1 *For the matrix $B(\alpha,\tau)$ defined by (2.3), we have*

$$B(\alpha,\tau) = S(\alpha) D(\alpha,\tau) S^{-1}(\alpha), \;\; S(\alpha) := \Lambda(\alpha) V_N, \tag{2.4a}$$

where $\Lambda(\alpha)$, V and $D(\alpha,\tau)$ are defined by

$$\begin{aligned} &\Lambda(\alpha) = \mathrm{diag}(1, \alpha^{-\frac{1}{N}}, \ldots, \alpha^{-\frac{N-1}{N}}), \\ &V_N = [v_1, v_2, \ldots, v_N], \text{ with } v_n = \left[1, e^{i\frac{2(n-1)\pi}{N}}, \ldots, e^{i\frac{2(n-1)(N-1)\pi}{N}}\right]^\top, \\ &D(\alpha,\tau) = \mathrm{diag}(\lambda_1(\alpha,\tau), \ldots, \lambda_N(\alpha,\tau)), \text{ with } \lambda_n(\alpha,\tau) = 1 + \tau\alpha^{\frac{1}{N}} e^{-i\frac{2(n-1)\pi}{N}}. \end{aligned} \tag{2.4b}$$

Proof Let $\tilde{v}_n := \Lambda(\alpha) v_n = \left[1, \alpha^{-\frac{1}{N}} e^{i\frac{2(n-1)\pi}{N}}, \ldots, \alpha^{-\frac{N-1}{N}} e^{i\frac{2(n-1)(N-1)\pi}{N}}\right]^\top$. Then, we have

$$\begin{aligned} B(\alpha,\tau)\tilde{v}_{n+1} &= \begin{bmatrix} 1 + \tau\alpha^{\frac{1}{N}} e^{i\frac{2n(N-1)\pi}{N}} \\ \tau + \alpha^{-\frac{1}{N}} e^{i\frac{2n\pi}{N}} \\ \vdots \\ \alpha^{-\frac{n}{N}} e^{i\frac{2nj\pi}{N}} + \tau\alpha^{-\frac{n-1}{N}} e^{i\frac{2n(j-1)\pi}{N}} \\ \vdots \\ \alpha^{-\frac{N-1}{N}} e^{i\frac{2n(N-1)\pi}{N}} + \tau\alpha^{-\frac{N-2}{N}} e^{i\frac{2n(N-2)\pi}{N}} \end{bmatrix} = \begin{bmatrix} 1 + \tau\alpha^{\frac{1}{N}} e^{-i\frac{2n\pi}{N}} \\ (1 + \tau\alpha^{\frac{1}{N}} e^{-i\frac{2n\pi}{N}})\alpha^{-\frac{1}{N}} e^{i\frac{2n\pi}{N}} \\ \vdots \\ (1 + \tau\alpha^{\frac{1}{N}} e^{-i\frac{2n\pi}{N}})\alpha^{-\frac{n}{N}} e^{i\frac{2nj\pi}{N}} \\ \vdots \\ (1 + \tau\alpha^{\frac{1}{N}} e^{-i\frac{2n\pi}{N}})\alpha^{-\frac{N-1}{N}} e^{i\frac{2n(N-1)\pi}{N}} \end{bmatrix} \\ &= \lambda_{n+1}(\alpha,\tau)\tilde{v}_{n+1}, \end{aligned}$$

which holds for all $n = 0, 1, \ldots, N-1$. Hence, (2.4a) holds. □

Note that the eigenvector matrix S given in Lemma 2.1 is independent of τ and thus B_1 and B_2 given by (2.2b) are simultaneously diagonalizable:

$$B_1 = \frac{1}{\Delta t} S(\alpha) D(\alpha,-1) S^{-1}(\alpha), \;\; B_2 = \theta S(\alpha) D\left(\alpha, \frac{1-\theta}{\theta}\right) S^{-1}(\alpha). \tag{2.5}$$

Substituting this into (2.2a) gives

$$\left[(S(\alpha)\otimes I_x)\left(\frac{1}{\Delta t} D(\alpha,-1)\otimes I_x + \theta D\left(\alpha, \frac{1-\theta}{\theta}\right)\otimes A\right)(S^{-1}(\alpha)\otimes I_x)\right] U = F.$$

This implies that the computation of U can be divided into the following three steps:

$$\begin{aligned} &(a) \qquad (S(\alpha)\otimes I_x)G = F, \\ &(b) \qquad \left(\lambda_{1,n} I_x + \Delta t \lambda_{2,n} A\right) w_n = \Delta t g_n,\ n = 1, 2, \ldots, N, \\ &(c) \qquad (S^{-1}(\alpha)\otimes I_x)U^k = W, \end{aligned} \tag{2.6}$$

where $G = (g_1, \ldots, g_N)^\top$ and $W = (w_1, \ldots, w_N)^\top$. Now, step (b) is entirely parallel for all N time points. The quantities $\lambda_{1,n}$ and $\lambda_{2,n}$ are defined by

$$\lambda_{1,n} := 1 - \alpha^{\frac{1}{N}} e^{-i\frac{2n\pi}{N}},\ \lambda_{2,n} := \theta + (1-\theta)\alpha^{\frac{1}{N}} e^{-i\frac{2n\pi}{N}}. \tag{2.7}$$

In (2.6), steps (a) and (c) are dual and the computation of these two steps can be carried out by FFT by noticing that $S(\alpha) = \Lambda(\alpha) V_N$ and V_N is a Fourier matrix (more details for this aspect will be given in Sect. 4). The major computational cost is to solve the linear algebraic system in step (b), for which many existing linear solvers are applicable. In particular, as shown in [48] the multi-grid method is a good choice.

Let U^k and $\widehat{U}^k$ be respectively the exact solution and computed solution of (2.6). Then, the roundoff error arising from these two steps may cause inaccuracy between U^k and $\widehat{U}^k$. This effect is characterized by the so-called relative error specified as follows.

Theorem 2.1 *Let U^k be the exact solution of (2.2a) and $\widehat{U}^k$ be the solution obtained by applying the diagonalization technique (2.6) to (2.2a). Assume that Step-(b) of (2.6) is solved in a direct manner (by e.g., the LU factorization) and that the matrix A is diagonalizable as $A = V_A D_A V_A^{-1}$. Then, we have*

$$\frac{|||U^k - \widehat{U}^k|||_2}{|||U^k|||_2} \le \epsilon(2N+1)\max\{|\alpha|^2, |\alpha|^{-2}\} \max_{\mu\in\sigma(A)} \frac{\left|1 + \Delta t\mu\theta + |1 - \Delta t\mu(1-\theta)|\,|\alpha|^{\frac{1}{N}}\right|}{\left|1 + \Delta t\mu\theta - |1 - \Delta t\mu(1-\theta)|\,|\alpha|^{\frac{1}{N}}\right|}, \tag{2.8}$$

where ϵ is the machine precision and the norm $|||\bullet|||_2$ is defined for any $U \in \mathbb{R}^{mN}$ by $|||U|||_2 = \|(I_t \otimes V_A)U\|_2$ with $I_t \in \mathbb{R}^{N\times N}$ being the identity matrix.

Proof Similar to [6], we consider an arbitrary eigenvalue μ of the matrix A and perform the relative error analysis by replacing A by μ in (2.2a) and (2.6). In this case $I_x = 1$ and the linear system (2.2a) is reduced to

$$\widetilde{B}U^k = F,\ \text{with}\ \widetilde{B} = B_1 + \mu B_2.$$

Let $\widetilde{D} := \frac{1}{\Delta t} D(\alpha, -1) + \mu\theta D(\alpha, \frac{1-\theta}{\theta})$ with $D(\bullet)$ being the diagonal matrix given in Lemma 2.1. Then, according to the backward error analysis [13, pp.122–126], the solution obtained by the diagonalization technique (2.6) satisfies the perturbed systems

$$(S + \delta S_1)\widehat{G} = F,\ (\widetilde{D} + \delta\widetilde{D})\widehat{W} = \widehat{G},\ (S^{-1} + \delta S_2)\widehat{U}^k = \widehat{W}, \tag{2.9}$$

where δS_1, δS_2 and $\delta \widetilde{D}$ denote the roundoff error of the matrices S, S^{-1} and $\widetilde{D}$. From [13, pp.122-126] we have

$$\|\delta S_1\|_2 \le \epsilon N \|S\|_2 + O(\epsilon^2),\ \|\delta S_2\|_2 \le \epsilon N \|S^{-1}\|_2 + O(\epsilon^2),\ \|\delta \widetilde{D}\| \le \epsilon \|\widetilde{D}\|_2 + O(\epsilon^2),$$

where the last inequality follows from the fact that $\widetilde{D}$ is a diagonal matrix. Note that solving $\widetilde{B}U^k = F$ by diagonalization is equivalent to solving $\left(\widetilde{B} + \delta\widetilde{B}\right)\widehat{U}^k = F$ with some suitable perturbation $\delta\widetilde{B}$. Moreover, from (2.9) we have

$$(S + \delta S_1)(\widetilde{D} + \delta\widetilde{D})(S^{-1} + \delta S_2)\widehat{U}^k = F.$$

From these two relations, we can estimate $\delta\tilde{B}$ as follows (see [6]):

$$\|\delta\tilde{B}\|_2 \le \epsilon(2N+1)\|S\|_2\|S^{-1}\|_2\|\widetilde{D}\|_2 + O(\epsilon^2).$$

The relative error of $\widehat{U}^k$ then satisfies (see [13, pp.122-126])

$$\begin{aligned}\frac{|||U^k - \widehat{U}^k|||_2}{|||U^k|||_2} &\le \mathrm{Cond}_2(\widetilde{B})\frac{\|\delta\widetilde{B}\|_2}{\|\widetilde{B}\|_2} \le \epsilon(2N+1)\|S\|_2\|S^{-1}\|_2\|\widetilde{D}\|_2\|\widetilde{B}\|_2 \\ &= \epsilon(2N+1)\mathrm{Cond}_2(S)\|\widetilde{B}^{-1}\|_2\|\widetilde{D}\|_2.\end{aligned} \tag{2.10}$$

For the matrix S given in Lemma 2.1, we have

$$\|S\|_2 \le \|\Lambda\|_2\|V\|_2 \le \max\left\{1, |\alpha|^{-\frac{N-1}{N}}\right\}\sqrt{N},\ \|S^{-1}\|_2 \le \max\left\{1, |\alpha|^{\frac{N-1}{N}}\right\}\frac{1}{\sqrt{N}}.$$

Hence,

$$\mathrm{Cond}_2(S) \le \max\left\{|\alpha|, |\alpha|^{-1}\right\}. \tag{2.11a}$$

For the diagonal matrix $\widetilde{D}$, it holds that

$$\begin{aligned}\|\widetilde{D}^{-1}\|_2 &= \frac{1}{\min_{n=1,2,\ldots,N}\left|\frac{1}{\Delta t}(1 - \alpha^{\frac{1}{N}}e^{-i\frac{2(n-1)\pi}{N}}) + \mu\theta + \mu(1-\theta)\alpha^{\frac{1}{N}}e^{-i\frac{2(n-1)\pi}{N}}\right|} \\ &\le \frac{\Delta t}{\left|1 + \Delta t\mu\theta - |1 - \Delta t\mu(1-\theta)|\,|\alpha|^{\frac{1}{N}}\right|}, \\ \|\widetilde{D}\|_2 &= \max_{n=1,2,\ldots,N}\left|\frac{1}{\Delta t}(1 - \alpha^{\frac{1}{N}}e^{-i\frac{2(n-1)\pi}{N}}) + \mu\theta + \mu(1-\theta)\alpha^{\frac{1}{N}}e^{-i\frac{2(n-1)\pi}{N}}\right| \\ &\le \frac{1}{\Delta t}\left(\left|1 + \Delta t\mu\theta + |1 - \Delta t\mu(1-\theta)|\,|\alpha|^{\frac{1}{N}}\right|\right).\end{aligned}$$

Now, by using (2.11a) we have

$$\begin{aligned}\|\widetilde{B}^{-1}\|_2\|\widetilde{D}\|_2 &\le \|S\|_2\|S^{-1}\|_2\|\widetilde{D}^{-1}\|_2\|\widetilde{D}\|_2 \le \max\{|\alpha|,|\alpha|^{-1}\}\mathrm{Cond}_2(\widetilde{D})\\ &\le \max\{|\alpha|,|\alpha|^{-1}\}\frac{\left|1+\Delta t\mu\theta+|1-\Delta t\mu(1-\theta)|\,|\alpha|^{\frac{1}{N}}\right|}{\left|1+\Delta t\mu\theta-|1-\Delta t\mu(1-\theta)|\,|\alpha|^{\frac{1}{N}}\right|}.\end{aligned} \tag{2.11b}$$

Substituting (2.11a) and (2.11b) into (2.10) finishes the proof. □

2.2 Diagonalization in the linear case: $\alpha = 0$

The diagonalization technique presented above is different from the one studied in [6,31]. There, the authors applied this technique, together with the Backward-Euler method, to $\dot{u}(t)+Au(t)=\tilde{f}(t)$ with initial-value condition $u(0)=u_0$. In this case, to make the diagonalization technique applicable, the authors in [6,31] suggested to use a series of different step-sizes $\{\Delta t_n\}_{n=1}^N$ for temporal discretization: $\frac{u_n-u_{n-1}}{\Delta t_n}+Au_n=\tilde{f}_n$, which can be represented as

$$(B_1\otimes I_x + I_t\otimes A)U=\tilde{F} \text{ with } B_1=\begin{bmatrix}\frac{1}{\Delta t_1} & & & \\ -\frac{1}{\Delta t_2} & \frac{1}{\Delta t_2} & & \\ & \ddots & \ddots & \\ & & -\frac{1}{\Delta t_N} & \frac{1}{\Delta t_N}\end{bmatrix}, \tag{2.12}$$

where $U=(u_1,\ldots,u_N)^\top$ and $\tilde{F}=(\tilde{f}(t_1)+\frac{u_0}{\Delta t_1},\tilde{f}(t_2),\ldots,\tilde{f}(t_N))^\top$.

With distinguishing step-sizes, the matrix B_1 can be diagonalized as $B_1=SDS^{-1}$ with $D=\mathrm{diag}(\frac{1}{\Delta t_1},\ldots,\frac{1}{\Delta t_N})$. For general choice of these step-sizes, we have to rely on numerical computation (e.g., the shifted QR factorization) to yield S and S^{-1}; see comments in [31, Sectoin 4]. Interestingly, in some special cases the matrices S and S^{-1} can be written down. For example, if we fix the step-sizes as $\Delta t_n=\Delta t\,\nu^{n-1}$ with $\nu>1$ being a free parameter and Δt being the reference step-size, the authors in [6] proved that S and S^{-1} are lower tri-diagonal Toeplitz matrices with explicit formula for each element. However, in this case the condition number of S increases rapidly as N increases, which implies rapid increase of the roundoff error. In particular, by letting

$$\nu=1+\tau \text{ with } \tau>0,\ \phi(N)=\begin{cases}\frac{N}{2}!(\frac{N}{2}-1)!, & \text{if } N \text{ is even},\\ (\frac{N-1}{2}!)^2, & \text{if } N \text{ is odd},\end{cases} \tag{2.13}$$

it holds that (see [6, Theorem 6])

$$\frac{|||U-\widehat{U}|||_\infty}{|||U|||_\infty}\le\epsilon\frac{N^2(2N+1)(N+\mu_{\max}T)}{\phi(N)}\tau^{-(N-1)}, \tag{2.14}$$

where U and $\widehat{U}$ are respectively the exact solution of (2.12) and the diagonalization-based numerical solution, $\mu_{\max}$ denotes the maximal eigenvalue of A and the norm $|||\bullet|||_\infty$ is defined for any $U \in \mathbb{R}^{mN}$ by $|||U|||_\infty = \|(I_t \otimes V_A)U\|_\infty$ with $I_t \in \mathbb{R}^{N\times N}$ being the identity matrix and V_A being the eigenvector matrix of A. To obtain a small discretization error we have to make the step-sizes as *equal* as possible (see [6, Theorem 1] for explanation), i.e., τ should be as small as possible. But, for $\tau \ll 1$ the estimate (2.14) implies that the relative error increases very fast as N increases. Clearly, this is quite different from the relative error for the diagonalization technique applied to the periodic-like problems; see (2.8).

Remark 2.1 For differential equations with initial condition, it is hard to apply the diagonalization technique with $\alpha = 0$ if the Trapezoidal rule is used. This can be explained as follows: as before, for initial-value problems we need to use variable step-sizes to discretize the temporal derivative, which results in

$$\frac{u_n - u_{n-1}}{\Delta t_n} + \frac{1}{2}(Au_n + Au_{n-1}) = \frac{1}{2}(\tilde{f}_n + \tilde{f}_{n-1}),\ n = 1, 2, \ldots, N.$$

Then, similar to (2.12) we can represent this discrete system as

$$(B_1 \otimes I_x + B_2 \otimes A)\, U = \tilde{F},\ B_1 = \begin{bmatrix} \frac{1}{\Delta t_1} & & & \\ -\frac{1}{\Delta t_2} & \frac{1}{\Delta t_2} & & \\ & \ddots & \ddots & \\ & & -\frac{1}{\Delta t_N} & \frac{1}{\Delta t_N} \end{bmatrix},$$
$$B_2 = \frac{1}{2}\begin{bmatrix} 1 & & & \\ -1 & 1 & & \\ & \ddots & \ddots & \\ & & -1 & 1 \end{bmatrix}, \tag{2.15}$$

where $\tilde{F} = (\tilde{f}(t_1) + \frac{u_0}{2\Delta t_1} - \frac{1}{2}Au_0, \tilde{f}(t_2), \ldots, \tilde{f}(t_N))^\top$. Clearly, the matrix B_2 is not diagonalizable and therefore the diagonalization technique can not be applied to (2.15).

2.3 Diagonalization in the nonlinear case

We now show how to apply the diagonalization technique to nonlinear differential equations with periodic-like condition. Here, we only consider the case $\alpha \neq 0$. The case $\alpha = 0$, i.e., the initial-value problems, are considered in [7]. For the discrete ODE system (2.1), we can represent the discrete solutions by

$$(B_1 \otimes I_x)U^k = F(U^k) := \begin{pmatrix} \frac{1}{\Delta t}R^{k-1} + (1-\theta)f(t_0, \alpha u_N^k + R^{k-1}) + \theta f(t_1, u_1^k) \\ (1-\theta)f(t_1, u_1^k) + \theta f(t_2, u_2^k) \\ \vdots \\ (1-\theta)f(t_{N-1}, u_{N-1}^k) + \theta f(t_N, u_N^k) \end{pmatrix}, \tag{2.16}$$

where $B_1 \in \mathbb{R}^{N\times N}$ is the matrix defined by (2.2b) and $I_x \in \mathbb{R}^{m\times m}$ is the identity matrix.

We now apply a quasi-Newton method to solve the nonlinear system (2.16). This leads with some initial iterate $U^k_{[0]}$ to the iteration

$$U^k_{[l+1]} = U^k_{[l]} - \mathbf{J}^{-1}(U^k_{[l]})\left((B_1 \otimes I_x)U^k_{[l]} - F(U^k_{[l]})\right), \tag{2.17}$$

where $U^k_{[l]} = (u^k_{[l],1}, \ldots, u^k_{[l],N})^\top$ and $\mathbf{J}(U^k_{[l]})$ is an approximation to the Jacobian $B_1 \otimes I_x - \partial_U F(U^k_{[l]})$. We determine $\mathbf{J}(U^k_{[l]})$ as follows (see [7]): we have

$$\partial_U F(U^k_{[l]}) = \begin{pmatrix} \theta\partial_u f_1 & & & \alpha(1-\theta)\partial_u f_0^* \\ (1-\theta)\partial_u f_1 & \theta\partial_u f_2 & & \\ & \ddots & \ddots & \\ & & (1-\theta)\partial_u f_{N-1} & \theta\partial_u f_N \end{pmatrix}, \tag{2.18a}$$

where $\partial_u f_n = \partial_u f(t_n, u^k_{[l],n})$ with $n = 1, 2, \ldots, N$ and $\partial_u f_0^* = \partial_u f(t_0, \alpha u^k_{[l],N} + R^{k-1})$. We then approximate the $N+1$ Jacobian matrices $\{\partial_u f_n\}_{n=1}^N$ and $\partial_u f_0^*$ by a single matrix $\frac{1}{N}\sum_{n=1}^N \partial_u f(t_n, u^k_{[l],n})$. Define

$$A^k_{[l]} := -\frac{1}{N}\sum_{n=1}^{N} \partial_u f(t_n, u^k_{[l],n}). \tag{2.18b}$$

Then, from (2.18a) we have $\partial_U F(U^k_{[l]}) \approx -B_2 \otimes A^k_{[l]}$. This relation gives the choice of $\mathbf{J}(U^k_{[l]})$:

$$\mathbf{J}(U^k_{[l]}) := B_1 \otimes I_x + B_2 \otimes A^k_{[l]}. \tag{2.18c}$$

Now, a routine calculation yields that (2.17) can be represented as

$$\mathbf{J}(U^k_{[l]})U^k_{[l+1]} = F(U^k_{[l]}) - (B_2 \otimes A^k_{[l]})U^k_{[l]}. \tag{2.19}$$

As before, we diagonalize B_1 and B_2 according to Lemma 2.1 and then represent $\mathbf{J}(U^k_{[l]})$ as

$$\mathbf{J}(U^k_{[l]}) = (S(\alpha) \otimes I_x)\left(\frac{1}{\Delta t}D(\alpha, -1) \otimes I_x + \theta D\left(\alpha, \frac{1-\theta}{\theta}\right) \otimes A^k_{[l]}\right)(S^{-1}(\alpha) \otimes I_x).$$

Hence, similar to (2.6) we can solve $U^k_{[l+1]}$ from (2.19) by the three steps

$$\begin{aligned} &(a) \qquad (S(\alpha)\otimes I_x)G = F(U^k_{[l]}) - (B_2\otimes A^k_{[l]})U^k_{[l]},\\ &(b) \qquad \left(\lambda_{1,n}I_x + \Delta t\lambda_{2,n}A^k_{[l]}\right)w_n = \Delta t g_n,\ n=1,2,\ldots,N,\\ &(c) \qquad (S^{-1}(\alpha)\otimes I_x)U^k_{[l+1]} = W, \end{aligned} \tag{2.20}$$

where $G=(g_1,\ldots,g_N)^\top$ and $W=(w_1,\ldots,w_N)^\top$. Now, step (b) is parallel for all the N time points. Here, the quantities $\lambda_{1,n}$ and $\lambda_{2,n}$ are given in (2.7).

3 Convergence analysis for linear problems

In this section, we perform a convergence analysis for (1.2) in the linear case,

$$\begin{cases} \dot{u}^k + Au^k = \tilde{f}(t),\ t\in(0,T),\\ u^k(0) = \alpha u^k(T) - \alpha u^{k-1}(T) + u_0, \end{cases} \tag{3.1}$$

where $A\in\mathbb{R}^{m\times m}$.

3.1 Continuous case

Let $e^k(t) := u^k(t) - u(t)$ be the error function of the kth iteration. Then, we have

$$e^k(t) = e^{-At}e^k(0),\ e^{k-1}(t) = e^{-At}e^{k-1}(0). \tag{3.2}$$

By letting $t=T$ in these two equalities and by substituting them into the periodic-like condition $e^k(0) = \alpha e^k(T) - \alpha e^{k-1}(T)$, we get $e^k(0) = \alpha e^{-AT}e^k(0) - \alpha e^{-AT}e^{k-1}(0)$. Hence,

$$e^k(0) = \frac{-\alpha e^{-AT}}{1-\alpha e^{-AT}}e^{k-1}(0). \tag{3.3}$$

As in Theorem 2.1, we assume that A is diagonalizable as

$$A = V_A D_A V_A^{-1},\ D_A = \mathrm{diag}(\mu_1,\mu_2,\ldots,\mu_m). \tag{3.4}$$

Then, for any vector norm $\|\bullet\|$ it holds that

$$\|V_A e^k(0)\| \le \max_{z\in\sigma(AT)} W(z)\|V_A e^{k-1}(0)\|,\ W(z) := \frac{|\alpha e^{-z}|}{|1-\alpha e^{-z}|}. \tag{3.5}$$

To make a quantitive analysis of $W(z)$, we consider a representative distribution of $\sigma(A)$: $\sigma(A)\subseteq\mathbf{D}(\omega,\eta_0)$, as given by (1.5). By using the maximum principle for

analytic functions and the symmetry of $W(z)$ with respect to the real axis, it is easy to see that

$$\max_{z\in\sigma(AT)} W(z) \le \max_{z\in\partial_+\mathbf{D}(\omega,\eta)} W(z), \tag{3.6}$$

where $\partial_+\mathbf{D}(\omega,\eta) := \{z=\eta+iy : 0\le y\le \tan(\omega)x\} \cup \{z=x+iy : x\ge\eta, y=\tan(\omega)x\}$ with $\eta := T\eta_0$ is the *half* boundary of $\sigma(AT)$ in the first quadrant (see Fig. 1).

Theorem 3.1 *Suppose that* (3.4) *holds and that* $\sigma(A)$ *lies in the region* $\mathbf{D}(\omega,\eta_0)$ *with* $\eta_0 \ge 0$ *and* $\omega\in[0,\frac{\pi}{2}]$. *Then, for* $\alpha\in(-1,1)$ *the error function* $e^k(t)$ *of the WR method* (3.1) *satisfies*

$$\max_{t\in[0,T]} \|V_A e^k(t)\| \le \rho^k(\alpha,\omega,\eta)\|V_A e^0(0)\|, \tag{3.7a}$$

where the convergence factor $\rho(\alpha,\omega,\eta)$ *of the WR method* (3.1) *is*

$$\rho(\alpha,\omega,\eta) = \begin{cases} \frac{|\alpha|e^{-\eta}}{1-\alpha e^{-\eta}}, & \text{if } \omega=0,\\ \max\{H(\eta),\ H(x_\dagger)\}, & \text{if } \omega\in(0,\frac{\pi}{2}), \tan(\omega)\eta<\pi \text{ and } \alpha<0,\\ \frac{|\alpha|e^{-\eta}}{1-|\alpha|e^{-\eta}}, & \text{otherwise,}\end{cases} \tag{3.7b}$$

where

$$\begin{aligned} H(x) &= \frac{|\alpha|e^{-x}}{\sqrt{1+\alpha^2e^{-2x}-2\alpha e^{-x}\cos(x\tan(\omega))}},\\ x_\dagger &= \begin{cases}\eta, & \text{if } H_1(\frac{\pi}{2\tan(\omega)})\le 0,\\ \max\{\eta,x_*\}, & \text{otherwise,}\end{cases}\end{aligned} \tag{3.7c}$$

and x_* *is the unique root of* $H_1(x)$ *for* $x\in[\frac{\pi}{2\tan(\omega)},\frac{\pi}{\tan(\omega)}]$ *with*

$$H_1(x) = |\alpha|e^{-x}[\tan(\omega)\sin(\tan(\omega)x)-\cos(\tan(\omega)x)]-1. \tag{3.7d}$$

Proof From (3.5) and (3.6), we have

$$\|V_A e^k(0)\| \le \max_{z\in\partial_+\mathbf{D}(\omega,\eta)} W(z)\|V_A e^{k-1}(0)\|. \tag{3.8a}$$

Since $\sigma(A)\subseteq\mathbf{D}(\omega,\eta_0)$, i.e., all the eigenvalues of A have nonnegative real parts, from (3.2) we have $\max_{t\in[0,T]}\|e^k(t)\|\le\|e^k(0)\|$ and this together with (3.8a) gives

$$\max_{t\in[0,T]}\|e^k(t)\| \le \left(\max_{z\in\partial_+\mathbf{D}(\omega,\eta)} W(z)\right)^k \|V_A e^0(0)\|. \tag{3.8b}$$

Hence, we only need to prove that $\max_{z\in\partial_+\mathbf{D}(\omega,\eta)} W(z) \le \rho(\alpha,\omega,\eta)$.
For $\omega = 0$, $\partial_+\mathbf{D}(\omega,\eta) := \{z = x : x \ge \eta\}$ and thus by using $\alpha \in (-1,1)$ we have

$$\max_{z\in\partial_+\mathbf{D}(\omega,\eta)} W(z) = \max_{z\ge\eta} \frac{|\alpha|e^{-z}}{1-\alpha e^{-z}} = \frac{|\alpha|e^{-\eta}}{1-\alpha e^{-\eta}}.$$

For $\omega = \frac{\pi}{2}$, $\partial_+\mathbf{D}(\omega,\eta) := \{z = \eta + iy : y \ge 0\}$. Then, we have

$$\max_{z\in\partial_+\mathbf{D}(\omega,\eta)} W(z) = \max_{y\ge 0} \frac{|\alpha|e^{-\eta}}{|1-\alpha e^{-\eta-iy}|} \le \frac{|\alpha|e^{-\eta}}{1-|\alpha|e^{-\eta}},$$

and this gives the third result in (3.7b) for $\omega = \frac{\pi}{2}$. For the case $\omega \in (0, \frac{\pi}{2})$, we split the analysis of finding the maximum of $W(z)$ into two parts:

$$\begin{aligned} &\max_{z\in\partial_+\mathbf{D}(\omega,\eta)} W(z) = \max\{W_{1,\max}, W_{2,\max}\}, \\ &W_{1,\max} := \max_{z=\eta+iy,\, y\in[0,\tan(\omega)\eta]} W(z), \quad W_{2,\max} := \max_{z=x(1+i\tan(\omega)),\, x\ge\eta} W(z). \end{aligned} \tag{3.9}$$

For $W_{1,\max}$, with the function $H(x)$ given by (3.7c), we obtain by a direct computation

$$\begin{aligned} W_{1,\max} &= \max_{y\in[0,\tan(\omega)\eta]} \frac{|\alpha|e^{-\eta}}{\sqrt{1+\alpha^2e^{-2\eta}-2\alpha\eta^{-\eta}\cos(y)}} \\ &= \begin{cases} \frac{|\alpha|e^{-\eta}}{\sqrt{1+\alpha^2e^{-2\eta}-2\alpha e^{-\eta}\cos(\eta\tan(\omega))}} = H(\eta), & \text{if } \tan(\omega)\eta < \pi \text{ and } \alpha < 0, \\ \frac{|\alpha|e^{-\eta}}{1-|\alpha|e^{-\eta}}, & \text{otherwise.} \end{cases} \end{aligned} \tag{3.10}$$

It remains to estimate $W_{2,\max}$, and we get

$$\begin{aligned} W_{2,\max} = \max_{x\ge\eta} \frac{|\alpha|e^{-x}}{\sqrt{1+\alpha^2e^{-2x}-2\alpha e^{-x}\cos(x\tan(\omega))}} \le \frac{|\alpha|e^{-\eta}}{1-|\alpha|e^{-\eta}}, \\ \forall\omega \in \left(0, \frac{\pi}{2}\right), \alpha \in (-1,1). \end{aligned} \tag{3.11a}$$

Since we are interested in the maximum of $W_{1,\max}$ and $W_{2,\max}$ (see (3.9)), from (3.10) we know that we can not use (3.11a) for the case $\tan(\omega)\eta < \pi$ and $\alpha < 0$, since otherwise the effect of ω is entirely neglected. We now give an explicit expression of $W_{2,\max}$ for $\tan(\omega)\eta < \pi$ and $\alpha < 0$. To this end, with the function $H(x)$ given by (3.7c) we claim

$$W_{2,\max} = \max_{x\in[\eta,\,\frac{\pi}{\tan(\omega)}]} H(x). \tag{3.11b}$$

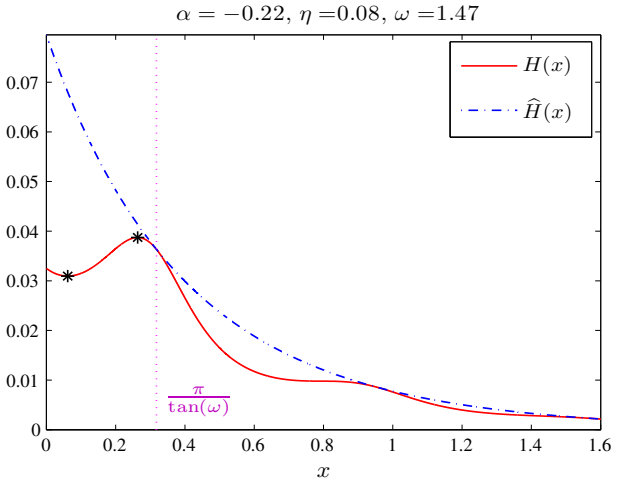

Fig. 2 Illustration of the relation between $H(x)$ and $\widehat{H}(x)$. The function $H(x)$ has at most two local extrema in the relevant interval $[0, \frac{\pi}{\tan(\omega)}]$, as indicated by the stars

Since $W_{2,\max} = \max_{x\geq\eta} H(x)$, it suffices to prove $\max_{x\geq\eta} H(x) = \max_{x\in[\eta,\frac{\pi}{\tan(\omega)}]} H(x)$. To this end, we define another function $\widehat{H}(x) := \frac{|\alpha|e^{-x}}{1-\alpha e^{-x}}$. Clearly, $H(x) \leq \widehat{H}(x)$ for all $x \geq \eta$. Moreover, it holds that $H(\frac{\pi}{\tan(\omega)}) = \widehat{H}(\frac{\pi}{\tan(\omega)})$ and that $\widehat{H}(x)$ is a decreasing function of x. Hence, a direct computation yields that the maximum of $H(x)$ must be attained in the interval $[\eta, \frac{\pi}{\tan(\omega)}]$. An illustration of the relation between $H(x)$ and $\widehat{H}(x)$ is shown in Fig. 2.

For the function $H(x)$ with $\alpha < 0$, a routine calculation yields

$$\text{sign}(H'(x)) = \text{sign}(H_1(x)),\ \ H_1'(x) = |\alpha|(1+\tan^2(\omega))e^{-x}\cos(\tan(\omega)x), \tag{3.11c}$$

where $H_1(x)$ is the function given by (3.7d). To derive the maximum of $H(x)$ for $x \in [\eta, \frac{\pi}{\tan(\omega)}]$, it is sufficient to determine the roots of $H_1(x)$. We first study the roots of $H_1(x)$ in the interval $[0, \frac{\pi}{\tan(\omega)}]$ and then we consider the shorter interval $[\eta, \frac{\pi}{\tan(\omega)}]$. From the expression of $H_1'(x)$ given by (3.11c), it is clear that $H_1(x)$ is an increasing function for $x \in [0, \frac{\pi}{2\tan(\omega)}]$ and it is a decreasing function for $x \in [\frac{\pi}{2\tan(\omega)}, \frac{\pi}{\tan(\omega)}]$. The unique maximizer of $H_1(x)$ is $x = \frac{\pi}{2\tan(\omega)}$. Hence, $H_1(x)$ can have at most 2 roots for $x \in [0, \frac{\pi}{\tan(\omega)}]$. This, together with the fact $H_1(0) < 0$, implies that we only need to consider three situations, as illustrated in Fig. 3. The last situation can not occur, because in this case $H_1(x) > 0$ for $x \in [\frac{\pi}{2\tan(\omega)}, \frac{\pi}{\tan(\omega)}]$ and therefore for a sufficiently small $\delta > 0$ it holds that $H(\frac{\pi}{\tan(\omega)} + \delta) > H(\frac{\pi}{\tan(\omega)}) = \widehat{H}(\frac{\pi}{\tan(\omega)})$, where $\widehat{H}(x) = \frac{|\alpha|e^{-x}}{1-\alpha e^{-x}}$. This is a contradiction, as we can see in Fig. 2.

In summary, if $H_1(\frac{\pi}{2\tan(\omega)}) \leq 0$ the function $H(x)$ is decreasing for $x \in [0, \frac{\pi}{\tan(\omega)}]$; otherwise $H(x)$ has one local minimum and one local maximum in the interval $[0, \frac{\pi}{\tan(\omega)}]$ (see Fig. 2 for illustration). The local maximum point, denoted by

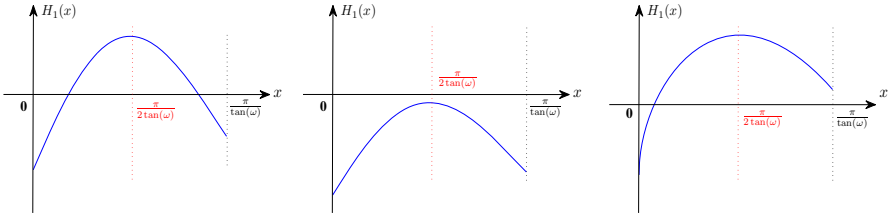

Fig. 3 The three cases for the roots of the function $H_1(x)$ given by (3.7c). The last case can not occur

x_*, is the unique root of $H_1(x)$ in the half interval $[\frac{\pi}{2\tan(\omega)}, \frac{\pi}{\tan(\omega)}]$. Restricting to the interval $[\eta, \frac{\pi}{\tan(\omega)}]$ that we are interested in, we have $\max_{x\in[\eta,\frac{\pi}{\tan(\omega)}]} H(x) = \max\{H(\eta), H(x_\dagger)\}$, where $x_\dagger$ is the quantity given by (3.7c). Substituting this into (3.11b) gives

$$W_{2,\max} = \max\{H(\eta), H(x_\dagger)\}, \text{ if } \eta\tan(\omega) < \pi \text{ and } \alpha < 0. \tag{3.11d}$$

Now, if $\eta\tan(\omega) < \pi$ and $\alpha < 0$, (3.11d) and (3.10) prove the second result in (3.7b); otherwise (3.11a) and (3.10) prove the third result in (3.7b). □

3.2 Discrete case

We now perform a convergence analysis of the iterative method (3.1) at the discrete level. Such an analysis reveals how the convergence rate depends on the time-integrator and the step-size Δt. Let u_n^k be the numerical solution of (3.1) at time point $t = t_n$ and u_n be the converged solution. Then, for linear θ-method it is easy to see that the error $e_n^k := u_n^k - u_n$ satisfies

$$\begin{cases} (I_x + \theta\Delta t A)e_n^k = (I_x - (1-\theta)\Delta t A)e_{n-1}^k, \ n = 1, 2, \ldots, N, \\ e_0^k = \alpha e_N^k - \alpha e_N^{k-1}. \end{cases}$$

From this we have

$$\begin{cases} e_N^k = \mathrm{R}_\theta^N(\Delta t A)e_0^k, \ \mathrm{R}_\theta(\Delta t A) := (I_x + \theta\Delta t A)^{-1}(I_x - (1-\theta)\Delta t A), \\ e_0^k = \alpha e_N^k - \alpha e_N^{k-1}. \end{cases} \tag{3.12}$$

Let $A = V_A D_A V_A^{-1}$ with D_A being the diagonal matrix. Then, similar to the analysis in the continuous case it follows from (3.12) that

$$\|V_A e^k(0)\| \le \max_{z\in\sigma(\Delta t A)} \widetilde{W}(z)\|V_A e^{k-1}(0)\|, \ \widetilde{W}(z) := \frac{|\alpha \mathrm{R}_\theta^N(z)|}{\left|1 - \alpha \mathrm{R}_\theta^N(z)\right|}. \tag{3.13}$$

To get the maximum of $\widetilde{W}(z)$ for $z \in \sigma(\Delta t A)$, by using the maximum principle for analytic functions and the symmetry of $\widetilde{W}(\mu)$ with respect to the real axis we have

$$\max_{z\in\mathbf{D}(\omega,\Delta t\eta_0)} \widetilde{W}(z) = \max_{z\in\partial_+\widetilde{\mathbf{D}}(\omega,\tilde{\eta})} \widetilde{W}(z), \tag{3.14}$$

where $\partial_+\widetilde{\mathbf{D}}(\omega,\tilde{\eta}) := \{z = \tilde{\eta} + iy : 0 \le y \le \tan(\omega)x\} \cup \{z = x + iy : x \ge \tilde{\eta}, y = \tan(\omega)x\}$ and $\tilde{\eta} := \Delta t\eta_0$. In the following, we consider $\theta = 1$ and $\theta = \frac{1}{2}$.

3.2.1 The Backward-Euler method ($\theta = 1$)

Let $z = x + iy \in \partial_+\widetilde{\mathbf{D}}(\omega,\tilde{\eta})$. Then, for $\theta = 1$ we have

$$\mathrm{R}_1(z) = \frac{1}{1+x+iy} = |\mathrm{R}_1(z)|e^{-i\psi(z)},\ \psi(z) = \arcsin\left(\frac{y}{|\mathrm{R}_1(z)|}\right). \tag{3.15}$$

Theorem 3.2 *Suppose that* (3.4) *holds and that* $\sigma(A)$ *lies in the region* $\mathbf{D}(\omega,\eta_0)$ *with* $\eta_0 \ge 0$ *and* $\omega \in [0, \frac{\pi}{2}]$. *Then, for* $\alpha \in (-1, 1)$ *the discrete error* $\{e_n^k\}$ *of the WR method* (3.1) *satisfies*

$$\max_{n=0,1,\ldots,N} \|V_A e_n^k\| \le \widetilde{\rho}_1^k(\alpha,\omega,\tilde{\eta})\|V_A e^0(0)\|, \tag{3.16a}$$

where $\widetilde{\rho}_1(\alpha,\omega,\tilde{\eta})$ *is the convergence factor of the discrete WR method using the Backward-Euler method,*

$$\widetilde{\rho}_1(\alpha,\omega,\tilde{\eta}) = \begin{cases} \frac{|\alpha|\mathrm{R}_1^N(\tilde{\eta})}{1-\alpha\mathrm{R}_1^N(\tilde{\eta})}, & \text{if } \omega = 0, \\ \max\{\widetilde{W}_{1,\max}, \widetilde{W}_{2,\max}\}, & \text{if } \omega \in (0, \frac{\pi}{2}),\ \psi_\dagger < \frac{\pi}{N} \text{ and } \alpha < 0, \\ \frac{|\alpha|\mathrm{R}_1^N(\tilde{\eta})}{1-|\alpha|\mathrm{R}_1^N(\tilde{\eta})}, & \text{otherwise.} \end{cases} \tag{3.16b}$$

In (3.16b), *the quantities* $\psi_\dagger$, $\widetilde{W}_{1,\max}$ *and* $\widetilde{W}_{2,\max}$ *are given by*

$$\psi_\dagger = \arcsin\left(\frac{\tilde{\eta}\tan(\omega)}{\sqrt{(1+\tilde{\eta})^2+\tilde{\eta}^2\tan^2(\omega)}}\right),\ \widetilde{W}_{1,\max} = \frac{|\alpha|\mathrm{R}_1^N(\tilde{\eta})}{\sqrt{1+\alpha^2\mathrm{R}_1^{2N}(\tilde{\eta})-2\alpha\mathrm{R}_1^N(\tilde{\eta})\cos(N\psi_\dagger)}}, \tag{3.16c}$$

$$\widetilde{W}_{2,\max} = \max_{x\in[\tilde{\eta},\,\tilde{x}_\dagger]} \frac{|\alpha||\mathrm{R}_1^N(x(1+i\tan(\omega)))|}{|1-\alpha\mathrm{R}_1^N(x(1+i\tan(\omega)))|} \text{ with } \tilde{x}_\dagger = \frac{1}{\tan(\omega)\sqrt{\frac{1}{\sin^2(\frac{\pi}{N})}-1}-1}. \tag{3.16d}$$

Proof From (3.13) and (3.14), we have

$$\|V_A e_0^k\| \le \max_{z\in\partial_+\mathbf{D}(\omega,\tilde\eta)} \widetilde W(z)\|V_A e_0^{k-1}\|,$$

where $\widetilde W(z)$ is given by (3.13) with $\theta = 1$. Since $\sigma(A) \subseteq \mathbf{D}(\omega, \eta_0)$ (i.e., all the eigenvalues of A have nonnegative real parts) and the Backward-Euler method is A-stable, from (3.12) we have

$$\|e_n^k\| \le \max_{z\in\partial_+\mathbf{D}(\omega,\tilde\eta)} |\mathrm{R}_1^n(z)|\|e_0^k\| \le \|e_0^k\|,\ \forall n = 0, 1, \ldots, N.$$

This gives $\max_{n=0,1,\ldots,N} \|e_n^k\| \le \left(\max_{z\in\partial_+\mathbf{D}(\omega,\tilde\eta)} \widetilde W(z)\right)^k \|V_A e_0^0\|$. Hence, similar to the continuous case we only need to prove that $\max_{z\in\partial_+\mathbf{D}(\omega,\tilde\eta)} \widetilde W(z) \le \widetilde\rho_1(\alpha, \omega, \tilde\eta)$.

If $\omega = 0$, we have $z = x \ge \tilde\eta$ and thus $\mathrm{R}_1(z) = \frac{1}{1+x} \in \left(0, \frac{1}{1+\tilde\eta}\right]$. Hence, from (3.14) we have $\max_{z=x\ge\tilde\eta} \widetilde W(z) = \frac{|\alpha|\mathrm{R}^N(\tilde\eta)}{1-\alpha\mathrm{R}^N(\tilde\eta)}$. If $\omega = \frac{\pi}{2}$, we have $z = \tilde\eta + iy$ with $y \ge 0$ and thus from (3.15)

$$\max_{z=\tilde\eta+iy,y\ge0} \widetilde W(z) = \max_{z=\tilde\eta+iy,y\ge0} \frac{|\alpha||\mathrm{R}_1^N(z)|}{|1-\alpha|\mathrm{R}_1^N(z)|e^{-iN\psi}|} = \frac{|\alpha|\mathrm{R}_1^N(\tilde\eta)}{1-|\alpha|\mathrm{R}_1^N(\tilde\eta)}.$$

This gives the third result in (3.16b) for $\omega = \frac{\pi}{2}$.

It remains to consider $\omega \in (0, \frac{\pi}{2})$. It holds that

$$\max_{z\in\partial_+\widetilde{\mathbf{D}}(\omega,\tilde\eta)} \widetilde W(z) = \max\left\{\max_{y\in(0,\tan(\omega)\tilde\eta]} \widetilde W(\tilde\eta + iy), \max_{x\ge\tilde\eta} \widetilde W(x + ix\tan(\omega))\right\} \quad (3.17)$$

Let $\psi(y) = \arcsin\left(\frac{y}{\sqrt{(1+\tilde\eta)^2+y^2}}\right)$. Then with $\psi_\dagger$ given by (3.16c) we have $\psi(y) \le \psi_\dagger$ for $y \in [0, \tan(\omega)\tilde\eta]$, and hence

$$\begin{aligned}
\max_{y\in(0,\tan(\omega)\tilde\eta]} \widetilde W(\tilde\eta + iy) &= \max_{z=\tilde\eta+iy,y\in(0,\tan(\omega)\tilde\eta]} \frac{|\alpha||\mathrm{R}_1^N(z)|}{\sqrt{1+\alpha^2|\mathrm{R}_1^N(z)|^2 - 2\alpha|\mathrm{R}_1^N(z)|\cos(N\psi(y))}} \\
&\le \max_{z=\tilde\eta+iy,y\in(0,\tan(\omega)\tilde\eta]} \frac{|\alpha|\mathrm{R}_1^N(\tilde\eta)}{\sqrt{1+\alpha^2\mathrm{R}_1^{2N}(\tilde\eta) - 2\alpha\mathrm{R}_1^N(\tilde\eta)\cos(N\psi(y))}} \\
&= \begin{cases} \frac{|\alpha|\mathrm{R}_1^N(\tilde\eta)}{\sqrt{1+\alpha^2\mathrm{R}_1^{2N}(\tilde\eta)-2\alpha\mathrm{R}_1^N(\tilde\eta)\cos(N\psi_\dagger)}}, & \text{if } \psi_\dagger < \frac{\pi}{N} \text{ and } \alpha < 0, \\ \frac{|\alpha|\mathrm{R}_1^N(\tilde\eta)}{1-|\alpha|\mathrm{R}_1^N(\tilde\eta)}, & \text{otherwise.} \end{cases} \quad (3.18a)
\end{aligned}$$

For $z = x(1 + i\tan(\omega))$ with $x \ge \tilde\eta$, by noticing that $|\mathrm{R}_1(z)| \le |\mathrm{R}_1(\tilde\eta(1 + i\tan(\omega)))|$ we have

$$\max_{x\ge\tilde\eta} \widetilde W(x(1 + i\tan(\omega))) \le \frac{|\alpha\mathrm{R}_1^N(\tilde\eta(1 + i\tan(\omega)))|}{1-|\alpha\mathrm{R}_1^N(\tilde\eta(1 + i\tan(\omega)))|} \le \frac{|\alpha|\mathrm{R}_1^N(\tilde\eta)}{1-|\alpha|\mathrm{R}_1^N(\tilde\eta)}, \quad (3.18b)$$

which holds for all $\alpha \in (-1, 1)$ and $\omega \in \left(0, \frac{\pi}{2}\right)$. For the special case $\psi_\dagger < \frac{\pi}{N}$ and $\alpha < 0$, we let $\widetilde{H}(x) = \frac{|\alpha||\mathrm{R}_1^N(x(1+i\tan(\omega)))|}{|1-\alpha\mathrm{R}_1^N(x(1+i\tan(\omega)))|}$ and then similar to (3.11b) we have

$$\max_{x\geq\tilde{\eta}} \widetilde{W}(x(1+i\tan(\omega))) = \max_{x\in[\tilde{\eta},\,\tilde{x}_\dagger]} \widetilde{H}(x), \tag{3.18c}$$

where $\tilde{x}_\dagger$ is determined by $N \arcsin\left(\frac{x\tan(\omega)}{\sqrt{(1+x)^2+(x\tan(\omega))^2}}\right) = \pi$ (solving this nonlinear equation gives the expression of $\tilde{x}_\dagger$ given by (3.16c)). To get (3.18c), we rewrite $\widetilde{H}(x)$ as

$$\widetilde{H}(x) = \frac{|\alpha||\mathrm{R}_1^N(x(1+i\tan(\omega)))|}{\sqrt{1+\alpha^2|\mathrm{R}_1^{2N}(x(1+i\tan(\omega)))| - 2\alpha|\mathrm{R}_1^N(x(1+i\tan(\omega)))|\cos(N\psi(x))}}.$$

where $\psi(x) = \arcsin\left(\frac{\tan(\omega)x}{\sqrt{(1+x)^2+(x\tan(\omega))^2}}\right)$ and the quantity $\tilde{x}_\dagger$ satisfies $N\psi(\tilde{x}_\dagger) = \pi$. From this we have $\widetilde{H}(x) \leq \widehat{\widetilde{H}}(x) := \frac{|\alpha||\mathrm{R}_1^N(x(1+i\tan(\omega)))|}{1-\alpha|\mathrm{R}_1^N(x(1+i\tan(\omega)))|}$. Since $\widehat{\widetilde{H}}(x)$ is a decreasing function of x and $\widehat{\widetilde{H}}(\tilde{x}_\dagger) = \widetilde{H}(\tilde{x}_\dagger)$, we have $\max_{x\geq\tilde{\eta}} \widetilde{H}(x) = \max_{x\in[\tilde{\eta},\tilde{x}_\dagger]} \widetilde{H}(x)$, which gives (3.18c).

Now, if $\psi_\dagger < \frac{\pi}{N}$ and $\alpha < 0$, we get the second result in (3.16b) by substituting (3.18c) and (3.18a) into (3.17); otherwise by substituting (3.18b) and (3.18a) into (3.17) we get the third result in (3.16b). □

3.2.2 The Trapezoidal rule ($\theta = \frac{1}{2}$)

We next consider the Trapezoidal rule, for which the stability function is

$$\mathrm{R}_{\frac{1}{2}}(z) = \frac{1-\frac{x}{2}-i\frac{y}{2}}{1+\frac{x}{2}+i\frac{y}{2}} = |\mathrm{R}_{\frac{1}{2}}(z)|e^{-i\psi},\ \psi = \arccos\left(\frac{1-\frac{x^2+y^2}{4}}{\sqrt{(1-\frac{x^2+y^2}{4})^2+y^2}}\right). \tag{3.19}$$

Theorem 3.3 *Suppose that* (3.4) *holds and that* $\sigma(A)$ *lies in the region* $\mathbf{D}(\omega, \eta_0)$ *with* $\eta_0 \geq 0$ *and* $\omega \in [0, \frac{\pi}{2}]$. *Then, for* $\alpha \in (-1, 1)$ *the error function* $e^k(t)$ *of the WR method* (3.1) *satisfies*

$$\max_{t\in[0,T]} \|V_A e^k(t)\| \leq \tilde{\rho}_{\frac{1}{2}}^k(\alpha, \omega, \tilde{\eta})\|V_A e^0(0)\|, \tag{3.20a}$$

where $\tilde{\eta} := \Delta t \eta_0$. *The quantity* $\widetilde{\rho}_{\frac{1}{2}}(\alpha, \omega, \tilde{\eta})$ *is the convergence factor of the discrete WR method using the Trapezoidal rule,*

$$\widetilde{\rho}_{\frac{1}{2}}(\alpha, \omega, \tilde{\eta}) = \frac{|\alpha|}{1-|\alpha|}, \ \forall \omega \in \left[0, \frac{\pi}{2}\right]. \tag{3.20b}$$

Proof Similar to Theorem 3.2, we only need to prove $\max_{z\in\partial_+\mathbf{D}(\omega,\tilde{\eta})} \widetilde{W}(z) \leq \widetilde{\rho}_{\frac{1}{2}}(\alpha, \omega, \tilde{\eta})$, where $\widetilde{W}(z)$ is given by (3.13) with $\theta = \frac{1}{2}$. Since $|\mathrm{R}_{\frac{1}{2}}(z)| \leq 1$ for $z \in \partial_+\mathbf{D}(\omega, \tilde{\eta})$, we have

$$\max_{z\in\partial_+\mathbf{D}(\omega,\tilde{\eta})} \widetilde{W}(z) \leq \max_{z\in\mathbf{D}(\omega,\eta_0)} \frac{|\alpha|}{1-|\alpha|\left|\mathrm{R}^N_{\frac{1}{2}}(z)\right|} \leq \frac{|\alpha|}{1-|\alpha|},$$

which gives (3.20b). □

Remark 3.1 The convergence factor $\widetilde{\rho}_{\frac{1}{2}}(\alpha, \omega, \tilde{\eta})$ given by (3.20b) is sharp and this can be explained as follows: we have

$$\max_{\mu\in\mathbf{D}(\omega,\eta_0)} \|\widetilde{W}(\mu)\|_\infty \geq \lim_{\mu\in\mathbf{D}(\omega,\eta_0),|\mu|\to\infty} \frac{|\alpha|}{\left|1-\alpha\mathrm{R}^N_{\frac{1}{2}}(\mu)\right|} = \frac{|\alpha|}{1-(-1)^N\alpha}.$$

where the '$(-1)^N$' comes from the fact that $\lim_{|\mu|\to\infty} \mathrm{R}_{\frac{1}{2}}(\mu) = -1$. Clearly, if $\alpha \in (-1, 0)$ (resp. $\alpha \in (0, 1)$) and if N is even (resp. odd), it holds that $\max_{\mu\in\mathbf{D}(\omega,\eta_0)} \|\widetilde{W}(\mu)\|_\infty = \frac{|\alpha|}{1-|\alpha|}$.

3.3 Discussion of the results

In this section, we comment the results given by Theorems 3.1, 3.2 and 3.3 .

3.3.1 Effect of temporal discretization

We first discuss the effect of the temporal discretization on the convergence factor. For the Backward-Euler method, since $N = \frac{T}{\Delta t}$, $\tilde{\eta} = \Delta t \eta_0$ and $\mathrm{R}_1(z) = \frac{1}{1+z}$ we have

$$\lim_{\Delta t\to 0} \mathrm{R}_1^N(\Delta t(1 + i\tan(\omega))) = e^{-T(1+i\tan(\omega))}, \quad \lim_{\Delta t\to 0} N\psi_\dagger = \tan(\omega)T\eta_0 = \tan(\omega)\eta,$$
$$\lim_{\Delta t\to 0} \mathrm{R}_1^N(\tilde{\eta}) = e^{-T\eta_0} = e^{-\eta}, \quad \lim_{\Delta t\to 0} N\tilde{x}_\dagger = \frac{\pi}{\tan(\omega)},$$

where $\eta = T\eta_0$. From the first two results, we therefore get

$$\lim_{\Delta t\to 0}\widetilde{W}_{2,\max} = \lim_{\Delta t\to 0}\max_{x\in[\eta_0,\ N\tilde{x}_\dagger/T]}\frac{|\alpha||\mathrm{R}_1^N(\Delta t x(1+i\tan(\omega)))|}{|1-\alpha\mathrm{R}_1^N(\Delta t x(1+i\tan(\omega)))|}$$
$$= \max_{x\in[\eta_0,\ \frac{\pi}{T\tan(\omega)}]}\frac{|\alpha|e^{-Tx}}{|1-\alpha e^{-Tx(1+i\tan(\omega))}|} = \max_{x\in[\eta,\ \frac{\pi}{\tan(\omega)}]}\frac{|\alpha|e^{-x}}{|1-\alpha e^{-x(1+i\tan(\omega))}|}.$$

Hence, by the proof of Theorem 3.1 we get $\lim_{\Delta t\to 0}\widetilde{W}_{2,\max} = W_{2,\max} = \max\{H(\eta), H(x_\dagger)\}$ (see (3.11d)).

From these calculations and by comparing (3.7b) to (3.16b), it is clear that the convergence factor $\widetilde{\rho}_1(\alpha,\omega,\tilde{\eta})$ approaches to the continuous convergence factor $\rho(\alpha,\omega,\eta)$ as Δt goes to 0. By choosing $T = 2$ and two values of α, we illustrate this point in Fig. 4 on the top row. However, for the Trapezoidal rule such a consistency does not hold between ρ and $\widetilde{\rho}_{\frac{1}{2}}$; see Fig. 4 on the bottom row. This can be also seen by comparing $\widetilde{\rho}_{\frac{1}{2}}$ given by (3.20b) to the quantity ρ given by (3.7b). Such an *inconsistency* comes from the fact that the stability function $\mathrm{R}_{\frac{1}{2}}(z)$ satisfies $\lim_{z\to\infty}\mathrm{R}_{\frac{1}{2}}(z) = -1$ and thus as we commented in Remark 3.1 the maximum of $\widetilde{W}(z)$ defined by (3.13) (with $\theta = \frac{1}{2}$) is $|\alpha|/(1-|\alpha|)$. In other words, the convergence factor $\widetilde{\rho}_{\frac{1}{2}}$ of the WR method using the Trapezoidal rule as the time-integrator depends on $|\alpha|$ only.

Remark 3.2 (Worst case estimate of the convergence factor) From Theorems 3.2 and 3.3, we have the following worst case estimate of the convergence factors of the proposed WR method at the discrete level:

$$\begin{cases}\widetilde{\rho}_1(\alpha,\omega,\tilde{\eta}) \le \frac{|\alpha|e^{-T\eta_0}}{1-|\alpha|e^{-T\eta_0}}, & \text{Backward-Euler(in the asymptotic sense, i.e.,}\Delta\text{t is small),}\\ \widetilde{\rho}_{\frac{1}{2}}(\alpha,\omega,\tilde{\eta}) \le \frac{|\alpha|}{1-|\alpha|}, & \text{Trapezoidal Rule,}\end{cases} \tag{3.21}$$

which holds for all $\omega\in[0,\frac{\pi}{2}]$ and $\eta_0 \ge 0$. The first result in (3.21) is interesting, since it implies that for an ODE system $u'(t)+Au(t) = \tilde{f}(t)$ with $\min_{\mu\in\sigma(A)}\Re(\mu) = \eta_0 > 0$ the WR method converges faster as T increases. We will illustrate this in Sect. 6.1 by using transmission line circuits [14] as the model. For time-dependent PDEs, the worst case estimate (3.21) implies that the WR method proposed in this paper has a robust convergence rate with respect to the discretization parameters Δx and Δt.

3.3.2 Two special cases: heat equations and wave equations

Two special cases are of particular interest, $\omega = 0$ and $\omega = \frac{\pi}{2}$. The former corresponds to the case where A is a (semi-)positive definite matrix, which often arises from discretizing the heat equations. The latter case may arise from discretizing the wave equation $\partial_{tt}u + \Delta u = f$. For these two cases we show in Fig. 5 the convergence factor ρ as a function of $\alpha\in[-\frac{1}{2},\frac{1}{2}]$ for three values of η. Clearly, for the heat equations it is better to use a negative parameter α. For the wave equations, α and $-\alpha$ have the same effect.

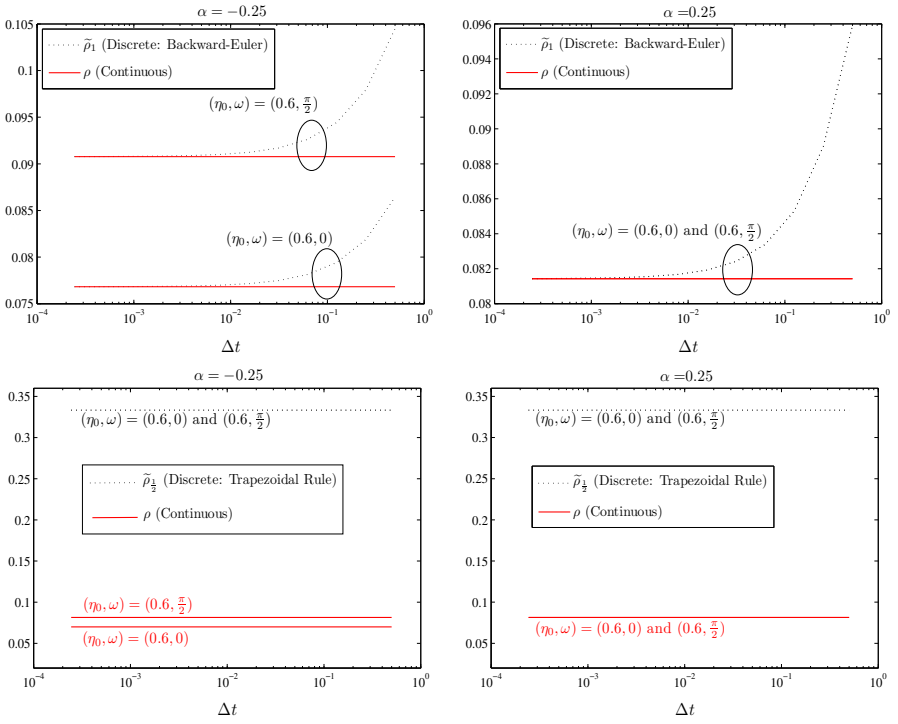

Fig. 4 Convergence factors of the WR method (3.1) at the continuous level and discrete level, when the time step size Δt varies. Top row: ρ (continuous) and $\widetilde{\rho}_1$ (discrete case with Backward-Euler) given by Theorems 3.1 and 3.2 . Bottom row: ρ (continuous) and $\widetilde{\rho}_{\frac{1}{2}}$ (discrete case with Trapezoidal Rule) given by Theorems 3.1 and 3.3 . Left column: $\alpha = -0.25$. Right column: $\alpha = 0.25$

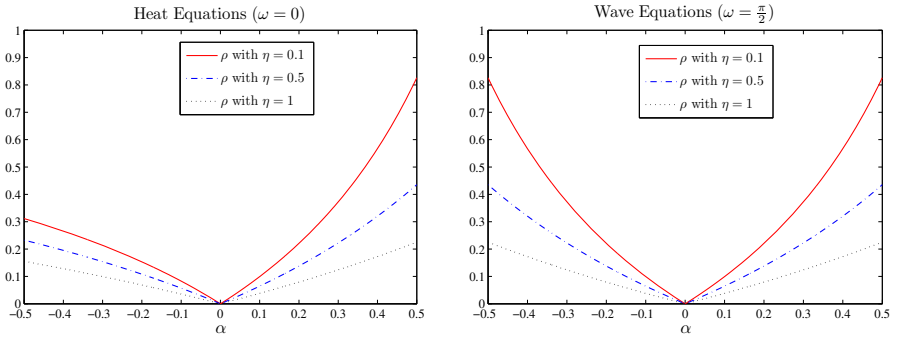

Fig. 5 The convergence factor ρ given by Theorem 3.1 as a function of $\alpha \in [-\frac{1}{2}, \frac{1}{2}]$. Left: the heat equations, i.e., $\omega = 0$; Right: the wave equations, i.e., $\omega = \frac{\pi}{2}$

4 Speedup analysis

We now discuss the speedup of the proposed WR method. The speedup is defined by

$$\text{speedup} = \frac{\mathbf{T}_{\text{serial}}}{\mathbf{T}_{\text{parallel}}}, \tag{4.1}$$

where $\mathbf{T}_{\text{serial}}$ denotes the total cost for completing the computation of the N discrete solutions step by step and $\mathbf{T}_{\text{parallel}}$ denotes the cost of the WR method to reach a prescribed tolerance. We denote by $\mathbf{M}$ the total real floating point operations (flops) for solving one time step of the ODE system and therefore the cost for the serial computation is

$$\mathbf{T}_{\text{serial}} = N\mathbf{M}. \tag{4.2}$$

We next analyze the cost for the WR method. Let ε be the prescribed tolerance and $\tilde{\rho}$ be the convergence factor.[1] Then, if $\|V_A e^0(0)\| = \mathcal{O}(1)$ the number of iterations is

$$K = \frac{\log_2 \varepsilon}{\log_2 \tilde{\rho}}. \tag{4.3a}$$

It is then sufficient to estimate the cost (denoted by $\mathbf{M}_{\text{diag}}$) for implementing the diagonalization procedure (2.6), which consists of two parts. We suppose that N processors are available.

• **Parallel implementation of** (2.6)-(a) **and** (2.6)**-(c)**. We start from (2.6)-(c), because for this step a *forward* FFT is involved (for (2.6)-(a) we need to do inverse FFT). For (2.6)-(c), since $S(\alpha) = \Lambda(\alpha)V_N$ we have $U^k = (S(\alpha) \otimes I_x)W = (\Lambda(\alpha) \otimes I_x)(V_N \otimes I_x)W$ and therefore the computation of U^k can be divided into two steps, $\widetilde{U}^k := (V_N \otimes I_x)W$ and $U^k = (\Lambda(\alpha) \otimes I_x)\widetilde{U}^k$. Following the Cooley-Tukey algorithm [2] the first matrix-vector product can be carried out by using FFT since V_N is a Fourier matrix. The computational cost of such a FFT is

$$M_{\text{serial}-\text{FFT}} := (5N \log_2 N)m.$$

The appearance of m in $M_{\text{serial}-\text{FFT}}$ is because of the fact that the vector W consists of N subvectors $\{w_n\}_{n=1}^N$ with $w_n \in \mathbb{C}^m$ and thus during the FFT every element of V_N acts on vectors (of length m) instead of scalar complex numbers. In the past decades, there was a lot of effort toward reducing the computational cost of FFT on parallel architectures; see, e.g., [1,5,18,19,37,41]. According to these studies, the speedup of the parallel FFT increases linearly when N increases. In other words, for the matrix-vector product $(V_N \otimes I_x)W$ the computational cost of the parallel FFT is of order $\mathcal{O}(m \log_2 N)$.

[1] For the Backward-Euler method $\tilde{\rho} = \tilde{\rho}_1$ and for the Trapezoidal rule $\tilde{\rho} = \tilde{\rho}_{\frac{1}{2}}$, where $\tilde{\rho}_1$ and $\tilde{\rho}_{\frac{1}{2}}$ are given by Theorems 3.2 and 3.3.

More precisely, if both N and P (the number of used processors) are powers of 2 and $P < N$, we can finish the computation of $(V_N \otimes I_x)W$ by the radix-4 BSP (bulk synchronous parallel) FFT algorithm with total flop count (see [18, Section 4]):

$$M_{\text{paralllel}-\text{FFT}} := \frac{17}{4}\frac{Nm}{P}\log_2 N + \frac{3}{4}\frac{Nm}{P}\left[\left(\log_2 \frac{N}{P} \bmod 2\right)\left\lfloor \log_{\frac{N}{P}} N\right\rfloor + \left(\log_2 N \bmod \log_2 \frac{N}{P}\right) \bmod 2\right].$$

For given N, we consider in the following the maximal value of P permitted by the radix-4 BSP FFT algorithm: $P = \frac{N}{2}$,[2] which leads to

$$M_{\text{paralllel}-\text{FFT}} = \frac{17}{2}m\log_2 N + \frac{3}{2}m\left(\lfloor \log_2 N\rfloor + \log_2 N \bmod 2\right) \approx 10m\log_2 N.$$

After obtaining the intermediate vector $\widetilde{U}^k$, the computation of $U^k = (\Lambda(\alpha)\otimes I_x)\widetilde{U}^k$ is naturally parallel since $\Lambda(\alpha)$ is a diagonal matrix. For this part, the flop count for each processor is $4m$. Therefore, the total flop count for parallel implementation of (2.6)-(c) is

$$M_{(2.6)-(c)} = 4m + \frac{17}{2}m\log_2 N + \frac{3}{2}m\left(\lfloor \log_2 N\rfloor + \log_2 N \bmod 2\right).$$

For (2.6)-(a), we have

$$G = (S(\alpha)\otimes I_x)^{-1}F = (V_N^{-1}\Lambda^{-1}(\alpha)\otimes I_x)F = (V_N^{-1}\otimes I_x)\left[(\Lambda^{-1}(\alpha)\otimes I_x)F\right].$$

Hence, we can first compute the matrix-vector product in the brackets with computational cost $4m$ and then carry out an inverse FFT. According to [18], the above 4-radix BSP FFT algorithm is directly applicable to such an inverse procedure. In summary, the total flop count for completing the matrix-vector products in (2.6)-(a) and (2.6)-(c) is

$$M_{(2.6)-(a,c)} := 2M_{(2.6)-(a)} = 8m + 17m\log_2 N + 3m\left(\lfloor \log_2 N\rfloor + \log_2 N \bmod 2\right).$$

• **Parallel implementation of (2.6)-(b).** All the N linear systems in (2.6)-(b) are completely independent and thus this step is naturally parallel. Similar to [31], we assume that the computational cost for solving each of these N linear systems is **M** as well. That is, the computational cost for solving each diagonalized linear system in (2.6)-(b) is comparable to that of forwarding one step of the time-integrator in the standard approach.

[2] This choice may be not optimal in practice.

According to the above analysis, the total computational cost for implementing the diagonalization procedure (2.6) (i.e., the computational cost for each WR iteration) is[3]

$$\mathbf{M}_{\text{diag}} = 8m + 17m\log_2 N + 3m\left(\lfloor \log_2 N \rfloor + \log_2 N \bmod 2\right) + \mathbf{M}.$$

Hence, by using (4.3a) we know that to reach the prescribed tolerance ε the computational cost for the WR method is

$$\begin{aligned}\mathbf{T}_{\text{parallel}} &= K\left(\mathbf{M}_{\text{diag}} + M_{\text{comm}}\right)\\ &= K\left[8m + 17m\log_2 N + 3m\left(\lfloor \log_2 N \rfloor + \log_2 N \bmod 2\right) + \mathbf{M} + M_{\text{comm}}\right],\end{aligned} \tag{4.3b}$$

where M_{comm} denotes communication cost for each WR iteration.

Now, substituting (4.3b) and (4.2) into (4.1) gives

$$\text{speedup} = \frac{N\mathbf{M}}{K\left[8m + 17m\log_2 N + 3m\left(\lfloor \log_2 N \rfloor + \log_2 N \bmod 2\right) + \mathbf{M} + M_{\text{comm}}\right]}. \tag{4.4}$$

If $\mathbf{M} = cm$ with some $c > 1$, $N \ll \mathbf{M}$ and $M_{\text{comm}} \ll \mathbf{M}$, from (4.4) we have

$$\text{speedup} \approx \frac{cN}{K\left[11 + 20\log_2 N + c\right]}. \tag{4.5}$$

From the worst case estimate of the convergence factor given by Remark 3.2, the convergence factor is robust with respect to N and therefore increasing N does not increase K. Hence, from (4.5) we know that the speedup of the WR method increases up to the log factor linearly as N increases. In the case $c \gg 1$, the speedup is of order $\mathcal{O}(N/K)$.

The assumption $\mathbf{M} = cm$ with $c > 1$ holds when the coefficient matrix A is spare and a robust linear solver is used (e.g., V-cycle and W-cycle multigrid methods [48]). The assumption $N \ll \mathbf{M}$ holds naturally when high dimension time-dependent PDE is concerned. Finally, the assumption $M_{\text{comm}} \ll \mathbf{M}$ holds naturally as well, because in each WR iteration we only need to make an update as $u^k(0) = \alpha u^k(T) - \alpha u^{k-1}(T) + u_0$ and just the value of the previous iterate $u^{k-1}(t)$ at the final time point $t = T$ is required. Therefore, the length of data transported between processors is m.

We now show some plots for the speedup according to (4.5) in the *worst* case by letting $c = 1$ and by choosing for $\tilde{\rho}$ in (4.3a) the estimate given by Remark 3.2, i.e.,

$$\tilde{\rho} = \begin{cases} \frac{|\alpha| e^{-T\eta_0}}{1 - |\alpha| e^{-T\eta_0}}, & \text{Backward-Euler},\\ \frac{|\alpha|}{1-|\alpha|}, & \text{Trapezoidal Rule}. \end{cases} \tag{4.6a}$$

[3] According to [18, Section 5], the communication cost for the parallel implementation of FFT is negligible.

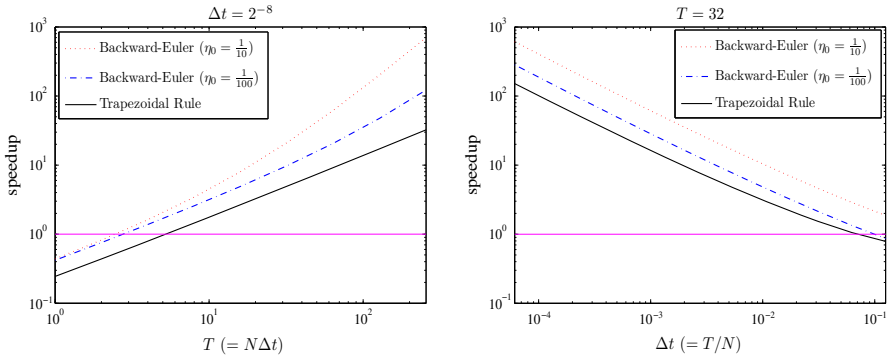

Fig. 6 Speedup of the WR method in the *worst* case by letting $c = 1$ and choosing (4.6a) for $\tilde{\rho}$ in (4.5). The tolerance ε is chosen as (4.6b) and $\alpha = 0.1$ is fixed. For the Trapezoidal rule there is only one line in each subfigure, because the $\tilde{\rho}$ is independent of η_0. Left: $\Delta t = 2^{-8}$ and T varies; Right: $T = 32$ and Δt varies

For the tolerance ε, we set

$$\varepsilon = \begin{cases} \frac{\Delta t}{10}, & \text{Backward-Euler}, \\ \frac{\Delta t^2}{10}, & \text{Trapezoidal Rule}, \end{cases} \tag{4.6b}$$

which is sufficient to match the temporal discretization error in practice. With these configurations and $\alpha = 0.1$ and two values of η_0, the speedup is plotted in Fig. 6. We see that for N small (i.e., T small or Δt large) there is no speedup, while when N becomes large the speedup increases at least linearly. For the Backward-Euler method, if Δt is fixed and $\eta_0 > 0$ the speedup increases *superlinearly* as T increases. This is because the convergence factor $\tilde{\rho}$ decreases (and thus the quantity K defined by (4.3a) decreases) as T increases.

5 Convergence analysis for nonlinear problems

We now analyze the convergence properties of the WR method (1.2) in the nonlinear case. We assume that the function f satisfies the one-sided Lipschitz condition (1.6). To control the length of this paper, we only perform such a convergence analysis at the continuous level. Convergence at the discrete level can be analyzed similarly.

Theorem 5.1 *Let $\{u^k\}_{k\geq 1}$ be the functions generated by the WR method (1.2), where $|\alpha| < 1$ and the nonlinear function f satisfies the one-sided Lipschitz condition (1.6) with some constant $L \geq 0$. Then, the error function $e^k(t) = u^k(t) - u(t)$ satisfies*

$$\max_{t\in[0,T]} \|e^k(t)\|_2 \leq \left(\frac{|\alpha|e^{-LT}}{1-|\alpha|e^{-LT}}\right)^k \|e^0(0)\|_2. \tag{5.1}$$

Proof From (1.1) and (1.2), we have

$$\begin{cases} \dot{e}^k = f(t, u^k) - f(t, u), \ t \in (0, T), \\ e^k(0) = \alpha e^k(T) - \alpha e^{k-1}(T). \end{cases} \tag{5.2}$$

For the Euclidean inner product, it holds for any differentiable function $v(t) \in \mathbb{R}^m$ that

$$\begin{cases} \frac{d\|v(t)\|_2^2}{dt} = 2\langle \dot{v}(t), v(t) \rangle \\ \frac{d\|v(t)\|_2^2}{dt} = 2\|v(t)\|_2 \frac{d\|v(t)\|_2}{dt} \end{cases} \Rightarrow \langle \dot{v}(t), v(t) \rangle = \|v(t)\|_2 \frac{d\|v(t)\|_2}{dt}.$$

Applying this relation to the differential equation in (5.2) gives

$$\langle \dot{e}^k(t), e^k(t) \rangle = \|e^k(t)\|_2 \frac{d\|e^k(t)\|_2}{dt} = \langle f(t, u^k) - f(t, u), u^k - u \rangle, \ t \in (0, T).$$

Then, by using the one-sided Lipschitz condition (1.6) we have

$$\frac{d\|e^k(t)\|_2}{dt} \leq -L\|e^k(t)\|_2, \ t \in (0, T). \tag{5.3}$$

Integrating this differential equation from 0 to T and using the periodic-like condition in (5.2), we get $\|e^k(T)\|_2 \leq e^{-LT}\|e^k(0)\|_2$ and $\|e^k(0)\|_2 \leq |\alpha|\|e^k(T)\|_2 + |\alpha|\|e^{k-1}(T)\|_2$. Hence, by using the first result twice we have

$$\|e^k(0)\|_2 \leq |\alpha|e^{-LT}\|e^k(0)\|_2 + |\alpha|e^{-LT}\|e^{k-1}(0T)\|_2,$$

which together with $L \geq 0$ gives

$$\|e^k(0)\|_2 \leq \frac{|\alpha|e^{-LT}}{1 - |\alpha|e^{-LT}}\|e^{k-1}(0)\|_2. \tag{5.4}$$

From (5.3), we have $\max_{t\in[0,T]} \|e^k(t)\|_2 \leq \|e^k(0)\|_2$ and this together with (5.4) gives (5.1). □

6 Numerical results

In this section, we present numerical results to illustrate our convergence analysis of the proposed WR method (1.2). In the first example, we consider the transmission line circuits studied in [14]. This is a linear ODE system with *wave* property, because the imaginary parts of the eigenvalues of the coefficient matrix are much larger than the real parts. The second example is the PLATE problem, which is a linear PDE in 2-D. In the last example we consider the 1-D Brusselator reaction-diffusion equation, which is a typical nonlinear dynamical system consisting of two coupled PDEs. The PLATE

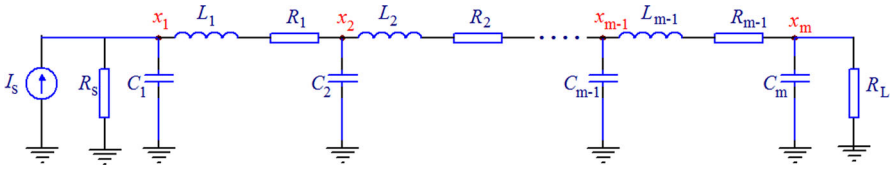

Fig. 7 Illustration of the transmission line circuit studied in [14], where $\{x_j\}_{j=1}^m$ denote the nodal voltages

problem and the Brusselator problem are members of the 'twelve test problems' used by Hairer and Wanner in their monograph [17] and they present severe challenges for numerical computation.

For the WR iterations, all experiments start from a random initial guess and the iteration stops when the error is less than 10^{-12}, i.e.,

$$\max_n \|u_n^k - u_n^{\text{ref}}\|_\infty \leq 10^{-12}, \tag{6.1}$$

where $\{u_n^{\text{ref}}\}$ denotes the reference solution computed by directly applying the temporal discretization to the differential equations.

6.1 The transmission line circuits

Transmission lines (TLs) are fundamental circuit elements for the modeling of many different structures and have many different applications in practice. Here, we consider the ladder-type TL circuits as the model problem that we want to solve (see Fig. 7 for illustration).

The circuit equations corresponding to Fig. 7 are specified as modified nodal analysis equations of tri-diagonal structure and are given by

$$\begin{bmatrix} C_1 & & & & & \\ & L_1 & & & & \\ & & C_2 & & & \\ & & & L_2 & & \\ & & & & \ddots & \\ & & & & & L_{m-1} \\ & & & & & & C_m \end{bmatrix} \begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \\ \dot{x}_4 \\ \vdots \\ \dot{x}_{m-1} \\ \dot{x}_m \end{bmatrix} + \begin{bmatrix} \frac{1}{R_s} & 1 & & & & & \\ -1 & R_1 & 1 & & & & \\ & -1 & 0 & 1 & & & \\ & & -1 & R_2 & 1 & & \\ & & & \ddots & \ddots & \ddots & \\ & & & & -1 & R_{m-1} & 1 \\ & & & & & -1 & \frac{1}{R_L} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ \vdots \\ x_{m-1} \\ x_m \end{bmatrix} = f(t), \tag{6.2}$$

where $f(t)$ is the input current source and $\{x_j(t)\}$ denote the nodal voltages as shown in Fig. 7. More details about the TL circuits can be found in [14] and the references cited therein.

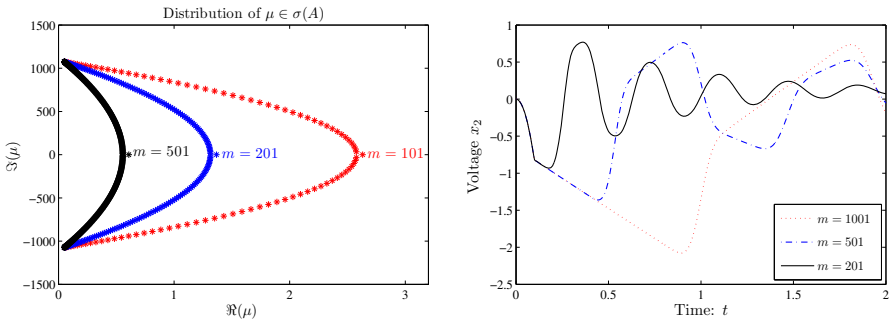

Fig. 8 Left: eigenvalues of the coefficient matrix A of the TL circuit equation (6.2) in the complex plane. Right: for three sizes of the TL circuit the evolution of the voltage at the second nodal point

We use the following circuit elements considered in [14] for our numerical experiments:

$$L_j = \frac{4.95 \times 10^{-3}}{30}\mu\text{H},\ C_j = \frac{0.63}{30}p\text{F},\ R_j = \frac{0.5 \times 10^{-3}}{30}\text{k},\ x_j(0) = 0,\ j = 1, 2, \ldots, m,$$
$$R_s = 0.02\text{k},\ R_L = 0.0005\text{k},\ f(t) = \begin{cases} 20t\ \text{mA}, & t \in [0, 0.1], \\ 2\ \text{mA}, & t \in [0.1, T]. \end{cases} \tag{6.3}$$

Let C be the diagonal matrix and D be the tri-diagonal matrix in (6.2). Then the coefficient matrix of the circuit equation (6.2) is $A = C^{-1}D$. In Fig. 8 on the left we show $\sigma(A)$ in the complex plane for three values of m and we see that as m increases $\sigma(A)$ approaches to the imaginary axis and therefore, mathematically, the circuit equation (6.2) has similar properties as the wave equation for m large (see Fig. 8 on the right). Moreover,

$$\min_{\mu \in \sigma(A)} \Re(\mu) \geq \frac{1}{20}, \forall m \in [10, 10^3]. \tag{6.4}$$

With three values of α, we now show in Fig. 9 the measured convergence rate of the WR method using the Backward-Euler method and the Trapezoidal rule. For each α, we show two errors: $\mathbf{Error_{Diag}}$ (solid line) denoting the diagonalization-based WR iterations and $\mathbf{Error_{Direct}}$ (dash-dot line) denoting the WR iterations implemented by directly inverting the large-scale matrix $(B_1 \otimes I_x + B_2 \otimes A)$ for each iteration.

We get four messages from Fig. 9. First, for both the Backward-Euler method and the Trapezoidal rule, a smaller $|\alpha|$ results in a better convergence rate. This confirms our theoretical analysis for the convergence factor of the WR method very well, because from Theorem 3.2 and Theorem 3.3 we know that a smaller $|\alpha|$ gives a smaller convergence factor. Second, the convergence rates of the WR method with $\alpha = 0.1$ and $\alpha = -0.1$ are almost the same. We also performed numerical experiments for $\alpha = 0.3$ and the convergence rate in this case is almost the same as that of $\alpha = -0.3$. This implies that for the TL circuit (6.2), α and $-\alpha$ have the same effect on the convergence rate of the WR iterations. This point confirms our comments in Sect. 3.3:

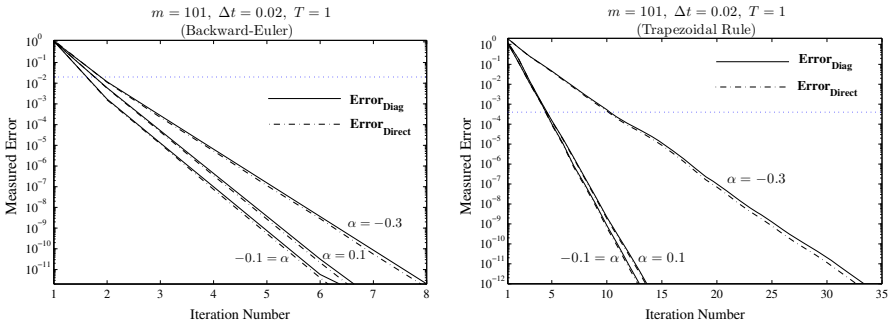

Fig. 9 Measured error of the discrete WR iterations. Left: Backward-Euler; Right: Trapezoidal rule. The horizontal line denotes the discretization error, which indicates how many iterations are needed in practice

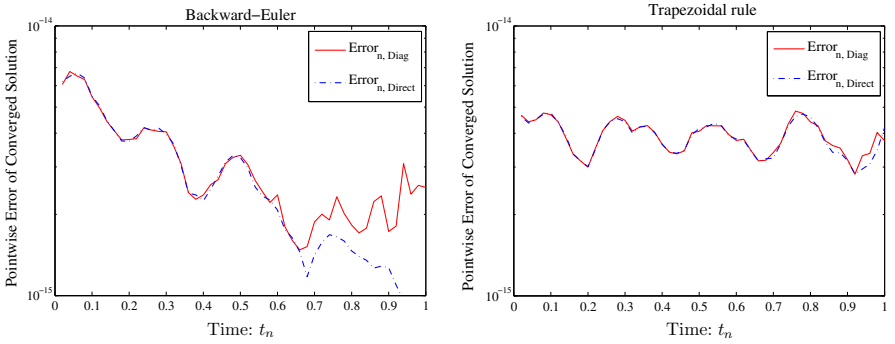

Fig. 10 The pointwise error between the WR iterate (after 7 iterations for the Backward-Euler method and 14 iterations for the Trapezoidal rule) and the reference solution $\{u_n^{\text{ref}}\}$ obtained by directly applying the time-integrators to (6.2). Here, we use $\alpha = 0.1$ and the plotting for other values of α look similar

for wave equations changing the sign of α does not affect the convergence rate of the WR method. Third, by comparing the left subfigure to the right one, we see that the Backward-Euler method results in significantly faster convergence for the WR method than the Trapezoidal rule does. This can be explained by using (6.4) and the first result of (3.21). Last, it is clear that for these six experiments the error $\mathbf{Error_{Diag}}$ is very close to $\mathbf{Error_{Direct}}$ and this implies that the diagonalization procedure does not affect the convergence rate of the WR method. This point implies that the round-off error is negligible and particularly this error is less than the tolerance 10^{-12}. To illustrate this, we show in Fig. 10 the pointwise error between the converged WR iterate and the reference solution. We see that for both the direct implementation and the diagonalization-based implementation the error between the converged WR iterate and the reference solution is close to machine precision.

In Fig. 11 we show the error of the WR iterations measured in practice and the error predicted by the convergence factors $\tilde{\rho}_1$ and $\tilde{\rho}_{\frac{1}{2}}$ given in Theorems 3.2 and 3.3 . From the top row, we see that for the Backward-Euler method the convergence factor $\tilde{\rho}_1$ is not sharp when m is small and it becomes sharp when m increases. For the Trapezoidal

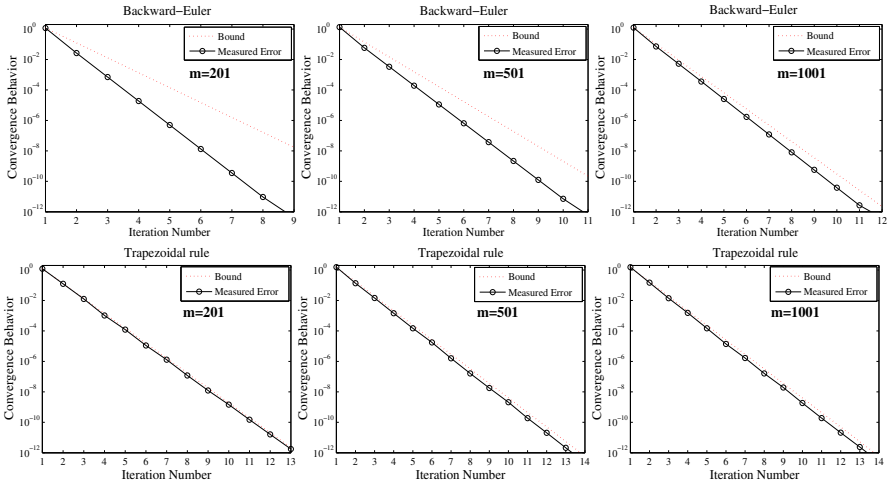

Fig. 11 For the transmission line circuit (6.2)–(6.3) with different values of m, the error measured in practice and the error predicted by the bound of the convergence factor. From left to right: $m = 201$, $m = 501$ and $m = 1001$. Top row: Backward-Euler method; Bottom row: Trapezoidal rule. Here, the parameter α is $\alpha = 0.1$

rule, from the bottom row we see that the convergence factor $\tilde{\rho}_{\frac{1}{2}}$ is sharp for all these three m.

We next show how the convergence behavior of the WR method depends on m, T and Δt. To this end, with $\alpha = 0.1$ we show in Fig. 12 the iteration number needed to satisfy the tolerance (6.1) by varying one of these parameters and keeping the other two fixed. Clearly, the results shown in Fig. 12 reveal that the convergence behavior of the WR method is robust with respect to these three parameters. Here, we only consider the case that the WR method is implemented by the diagonalization technique. For the case of direct implementation, the plot looks similar. From the left and right subfigures, we see that the WR methods using the Backward-Euler method and the Trapezoidal rule both have robust convergence behavior with respect to m and Δt. The middle subfigure is very interesting, since it shows that the WR method using the Trapezoidal rule has a robust convergence behavior with respect to T, while the WR method using Backward-Euler has a faster convergence behavior as T increases. All these numerical results confirm the worst case estimate (3.21) very well; see Remark 3.2.

6.2 The PLATE problem

The PLATE problem is a linear and non-autonomous equation, which describes the movement of a rectangular plate under the load of a car passing across it,

$$\begin{cases} \frac{\partial^2 u}{\partial t^2} + \phi \frac{\partial u}{\partial t} + \delta \Delta^2 u = f(x, y, t), & (x, y, t) \in \Omega \times (0, T), \\ \partial_{\mathrm{n}} u(x, y, t) = 0, \, \Delta u(x, y, t) = 0, & (x, y, t) \in \partial\Omega \times (0, T), \\ u(x, y, 0) = 0, \, \partial_t u(x, y, 0) = 0, & (x, y) \in \Omega, \end{cases} \tag{6.5}$$

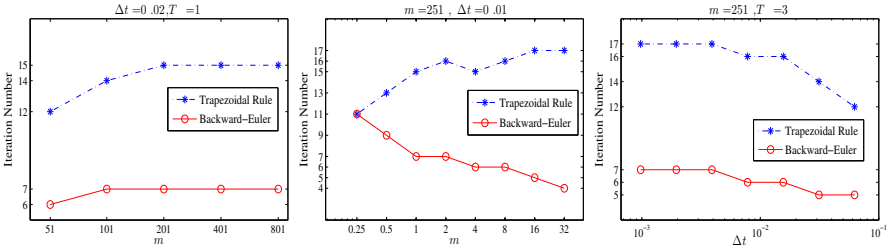

Fig. 12 Dependence of the convergence behavior of the WR method (implemented by the diagonalization technique) on the quantity m (left), T (middle) and Δt (right). Here we choose $\alpha = 0.1$

where $\phi = 10^4$, $\delta = 10^2$, $\Omega = (0, 1)^2$ and $\partial_n u$ denotes the outer normal derivative. We partition the space domain Ω with mesh-size Δx and denote the interior points by $\{(x_j, y_k) : x_j = j\Delta x, y_k = k\Delta x, 1 \le j, k \le m\}$, where $m = \frac{1}{\Delta x} - 1$. The load $f(x, y, t)$ is idealized by the sum of two Gaussian curves which move in the x-direction and reside on 'four wheels',

$$f(x, y, t) = \begin{cases} 200\left(e^{-5(t-x-2)^2} + e^{-5(t-x-5)^2}\right), & \text{if } y = y_2 \text{ or } y_4, \\ 0, & \text{otherwise.} \end{cases} \tag{6.6}$$

We first discretize the operator Δ^2 by the centered finite difference formula with Δx,

$$\Delta^2 \approx Q := (A_1 A_2) \otimes I_x + I_x \otimes (A_1 A_2),$$
$$A_1 = \frac{1}{\Delta x^2}\begin{bmatrix} -2 & 1 & & & \\ 1 & -2 & 1 & & \\ & \ddots & \ddots & \ddots & \\ & & 1 & -2 & 1 \\ & & & 1 & -2 \end{bmatrix}_{m\times m}, \quad A_2 = \frac{1}{\Delta x^2}\begin{bmatrix} -1 & 1 & & & \\ 1 & -2 & 1 & & \\ & \ddots & \ddots & \ddots & \\ & & 1 & -2 & 1 \\ & & & 1 & -1 \end{bmatrix}_{m\times m}, \tag{6.7}$$

where $I_x \in \mathbb{R}^{m\times m}$ is the identity matrix. Then, after lexicographically ordering the spatial grid, we get the second-order differential system:

$$\begin{cases} \ddot{\mathbf{u}}(t) + \phi\dot{\mathbf{u}}(t) + \delta Q\mathbf{u}(t) = \mathbf{f}(t), t \in (0, T), \\ \mathbf{u}(0) = 0, \ \mathbf{u}'(0) = 0, \end{cases} \tag{6.8}$$

where $\mathbf{f}(t)$ is the discrete version of $f(x, y, t)$. Let $\tilde{\mathbf{u}}(t) = \mathbf{u}'(t)$. Then, we can rewrite (6.8) as

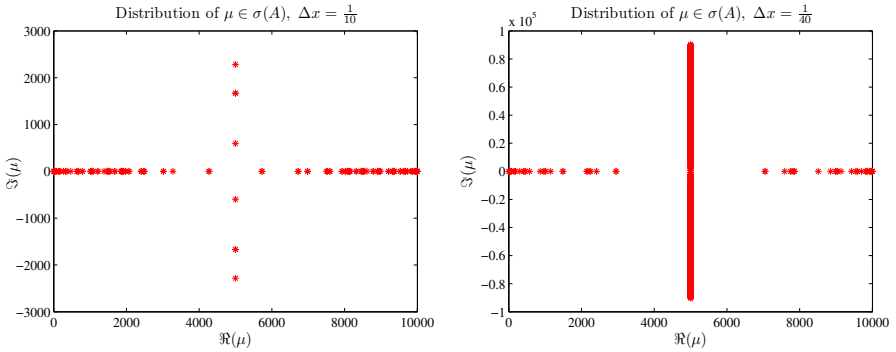

Fig. 13 Distribution of the spectrum of the matrix A defined by (6.9) for two values of the mesh-size Δx: $\Delta x = \frac{1}{10}$ (left) and $\Delta x = \frac{1}{40}$ (right). For the first case $\omega = 0.428$ and for the second case $\omega = 1.515$

$$\dot{\mathbf{U}}(t) + A\mathbf{U}(t) = \mathbf{F}(t),\ \mathbf{U}(t) := \begin{bmatrix} \mathbf{u}(t) \\ \tilde{\mathbf{u}}(t) \end{bmatrix},\ A := \begin{bmatrix} -\mathbf{I}_x \\ \delta Q\ \ \phi \mathbf{I}_x \end{bmatrix},\ \mathbf{F}(t) := \begin{bmatrix} 0 \\ \mathbf{f}(t) \end{bmatrix}, \tag{6.9}$$

where $\mathbf{I}_x = \mathbf{R}^{m^2 \times m^2}$ is an identity matrix.

The spectrum $\sigma(A)$ is distributed in the region $\mathbf{D}(\omega, \eta_0)$ given by (1.5) with $\eta_0 = 0$ and ω depending on Δx. In Fig. 13, we show the distribution of $\sigma(A)$ for two values of Δx: $\Delta x = \frac{1}{10}$ (left) and $\Delta x = \frac{1}{40}$ (right). The quantity ω corresponding to these two Δx is $\omega = 0.428$ and $\omega = 1.515$. Therefore, for the former case the ODE system (6.9) is nearly a SPD problem, while for the latter case it is nearly a wave problem. This property leads to different convergence rates of the discrete WR method as shown in Fig. 14. For $\Delta x = \frac{1}{10}$, from the top row we see that $\alpha = 0.2$ and $\alpha = -0.2$ result in the same convergence rates for the WR method using the Trapezoidal rule, while for the WR method using Backward-Euler the latter results in faster convergence. For $\Delta x = \frac{1}{40}$, from the bottom row we see that $\alpha = 0.2$ and $\alpha = -0.2$ result in the same convergence rates for both the Trapezoidal rule and the Backward-Euler method. These numerical results confirm our discussion in Sect. 3.2.2.

We now show the dependence of the iteration numbers of the WR method on the mesh parameter and the length of the time interval. To this end, we show in Fig. 15 the iteration number needed to reach the tolerance (6.1) when one parameter is fixed and the other one varies. In particular, in the top row we fix $T = 7$ and vary $\Delta x = \Delta t$ from 2^{-3} to 2^{-8}; in the bottom row we fix $\Delta x = \Delta t = \frac{1}{50}$ and vary T from 2 to 128. For the PLATE problem, the angle ω associated with the spectrum $\sigma(A)$ approaches $\frac{\pi}{2}$ as Δx decreases (see Fig. 13) and therefore for both the Backward-Euler method and the Trapezoidal rule changing the sign of the parameter α does not affect the convergence behavior of the WR method when Δx is small (see our discussion in Sect. 3.2.2). This confirms the numerical results given in Fig. 15 on the top row very well. Since $0 \in \sigma(A)$, we have $\eta_0 = \min_{\mu \in \sigma(A)} \Re(\mu) = 0$ and thus from the worst case estimate of the convergence factor given by (3.21) we know that the WR method using the Trapezoidal rule and the Backward-Euler method has a robust convergence

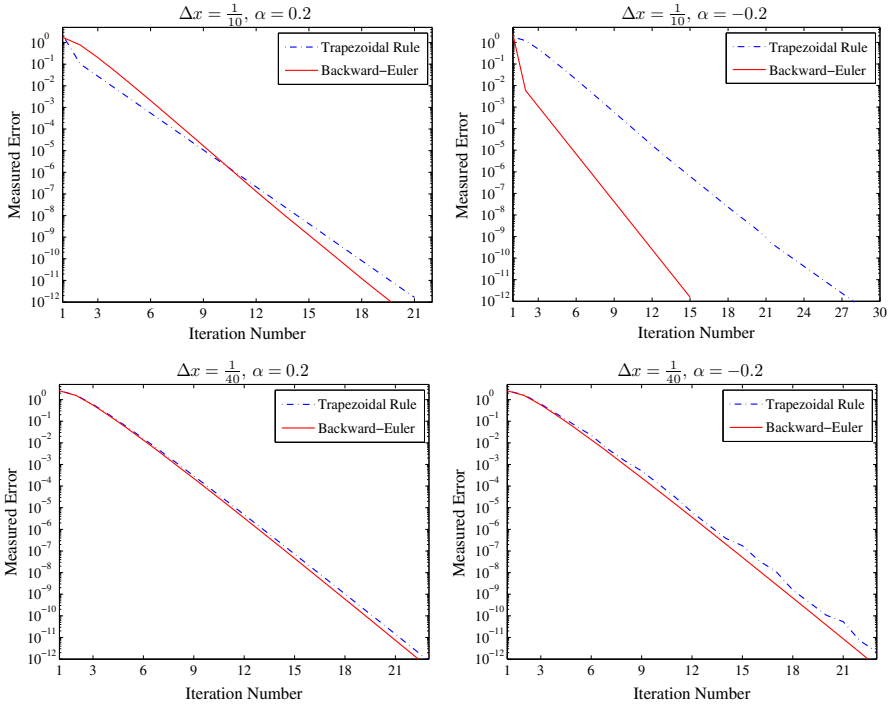

Fig. 14 Comparison of the measured convergence rates of the two discrete WR methods. Top row: $\Delta x = \frac{1}{10}$; bottom row: $\Delta x = \frac{1}{40}$; left column: $\alpha = 0.2$; right column: $\alpha = -0.2$. Here, $\Delta t = \frac{1}{50}$ and $T = 7$

behavior with respect to T. This prediction is confirmed by the numerical results given in Fig. 15 in the bottom row. In particular, for the Backward-Euler method, by comparing the middle subfigure in Fig. 12 to the top-right subfigure in Fig. 15 we see that the parameter η_0 has an important effect on the convergence behavior of the WR method.

6.3 The Brusselator reaction–diffusion equation

At the end of this section, we consider the Brusselator reaction–diffusion equation

$$\begin{cases} \partial_t u_1 = 0.1\partial_{xx} u_1 + u_1^2 u_2 - 4.4u_1 + f(x,t), \\ \partial_t u_2 = 0.1\partial_{xx} u_2 - u_1^2 u_2 - 3.4u_1, \end{cases} \tag{6.10a}$$

where $x \in (0, 1)$ and $t \in (0, T)$. We use the following data

$$\begin{aligned} &u_1(x,0) = 22x(1-x)^{1.5},\ u_2(x,0) = 27x(1-x)^{1.5}, \\ &u_1(0,t) = u_1(1,t) = 0,\ u_2(0,t) = u_2(1,t) = 0, \\ &f(x,t) = \begin{cases} 6, & \text{if } (x-0.3)^2 \le 0.1^2 \text{ and } t \ge 1.1, \\ 1, & \text{otherwise.} \end{cases} \end{aligned} \tag{6.10b}$$

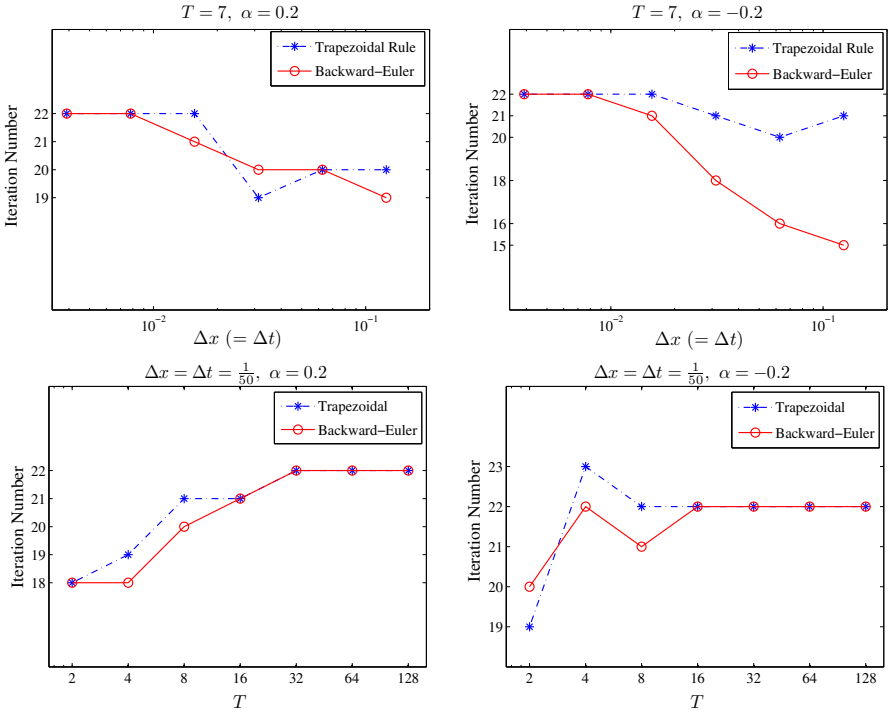

Fig. 15 For the semi-discrete PLATE problem (6.9), iteration number needed to reach the tolerance (6.1) in four different situations. Top row: $T = 7$ is fixed and $\Delta x = \Delta t$ varies from 2^{-3} to 2^{-8}. Bottom row: $\Delta x = \Delta t = \frac{1}{50}$ is fixed and T varies from 2 to 128. Left column: $\alpha = 0.2$; right column: $\alpha = -0.2$

We discretize the Laplacian ∂_{xx} via the centered finite difference formula with mesh-size Δx.

We implemented the WR iterations by using the *nonlinear* diagonalization technique described in Sect. 2.3, where the quasi-Newton method is used as the inner solver for each WR iteration. Similar to Fig. 15, we show in Fig. 16 the iteration number of the WR method needed to satisfy the tolerance (6.1) in four situations. We see that for nonlinear problems the convergence behavior of the WR method proposed in this paper is still satisfactory. In particular, for both the Trapezoidal rule and the Backward-Euler method, the iteration number is robust with respect to the mesh parameters (see Fig. 16 on the top row). Similar to the transmission line circuits considered in Sect. 6.1, for Backward-Euler the WR method converges faster as T increases (see Fig. 16 in the bottom row). A possible explanation is that the Brusselator reaction-diffusion equation is *dissipative* and therefore the spectrum of the Jacobian of the semi-discrete system of (6.10a) is distributed in a region $\mathbf{D}(\omega, \eta_0)$ with positive η_0, i.e., $\eta_0 > 0$. Hence, the worst case estimate (3.21) implies that the WR method has a better convergence behavior when T becomes larger. For the Trapezoidal rule, the convergence behavior of the WR method is insensitive to T.

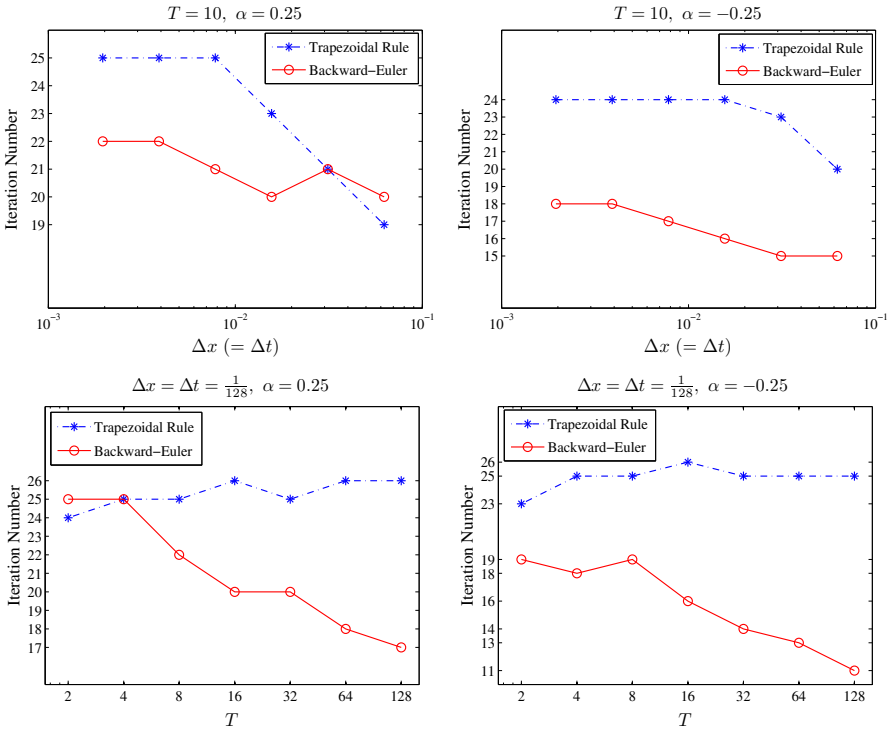

Fig. 16 For the Brusselator equation (6.10a)–(6.10b), iteration number needed to reach the tolerance (6.1) in four different situations. Top row: $T = 10$ is fixed and $\Delta x = \Delta t$ varies from 2^{-4} to 2^{-9}. Bottom row: $\Delta x = \Delta t = \frac{1}{128}$ is fixed and T varies from 2 to 128. Left column: $\alpha = 0.25$; right column: $\alpha = -0.25$

7 Conclusions

We have proposed a WR method for solving initial-value problems in parallel. The main idea lies in the simple observation that the initial condition $u(0) = u_0$ can be recovered through the periodic-like condition $u^k(0) = \alpha u^k(T) - \alpha u^{k-1}(T) + u_0$ upon convergence. Each iteration of the WR method is to solve a differential equation with periodic-like condition, for which the diagonalization technique proposed recently [6,7,31] can be used. Such a technique yields a direct parallel-in-time solver and was originally proposed to solve differential equations with initial conditions. In this paper, we show that it is more suitable to solve differential equations with periodic-like conditions.

The parameter α controls both the roundoff error arising from the diagonalization procedure and the convergence rate of the WR method. We have made a thorough analysis for the new WR method and we show that the roundoff error is proportional to $\epsilon(2N+1)\max\{|\alpha|^{-2}, |\alpha|^2\}$ (with ϵ being the machine precision), and our numerical results indicate that the roundoff error is negligible compared to the discretization error in practice. Our numerical results also indicate that the diagonalization-based WR method has the same convergence rate as the WR method implemented directly

(without diagonalization). In particular, our analysis reveals a rich relationship between the convergence factor: first, using Backward-Euler or the Trapezoidal rule as the time-integrator, the WR method has a robust convergence rate with respect to the discretization parameters Δt and/or Δx. Second, for linear problems $\dot{u}(t)+Au(t) = \tilde{f}$ with $\sigma(A) \subseteq \mathbf{D}(\omega, \eta_0)$ the convergence factor of the WR method can be bounded by $\frac{|\alpha|e^{-T\eta_0}}{1-|\alpha|e^{-T\eta_0}}$, if the Backward-Euler method is used. For the Trapezoidal rule, the convergence factor can be bounded by $\frac{|\alpha|}{1-|\alpha|}$. These two bounds hold for all $\omega \in [0, \frac{\pi}{2}]$ and are worst case estimates. Third, for the Trapezoidal rule the sign of α does not affect the convergence rate, while for Backward-Euler a negative α is better in most cases, i.e., $\omega \in [0, \frac{\pi}{2})$ (for $\omega = \frac{\pi}{2}$ the sign of α does not make any difference).

Acknowledgements The authors are very grateful to the anonymous referees for their careful reading of a preliminary version of the manuscript and their valuable suggestions, which greatly improved the quality of this paper. The second author is supported by the NSF of China (No. 11771313).

Appendix

The main notations and symbols of this paper are listed in the following table.

A	Coefficient matrix	θ	Linear θ-method ($\theta = 1$ or $\theta = \frac{1}{2}$)
T	Length of time interval	Δt	Time step-size
N	Number of time steps	Δx	Space mesh-size
m	Dimension of IVP	Cond$(\cdot)$	Condition number of a matrix
ω	Opening angle of sector	L	Lipschitz constant
μ	Eigenvalue of A	ε	Tolerance for WR iterations
η_0	Minimal real part of $\mu(A)$	ϵ	Machine precision
η	$\eta = \eta_0 T$	ρ	Convergence factor (continuous)
$\tilde{\eta}$	$\tilde{\eta} = \eta_0 \Delta t$	$\tilde{\rho}$	Convergence factor (discrete)
α	Parameter used in WR iteration	I_t	$I_t \in \mathbb{R}^{N\times N}$ is an identity matrix
k	Index of WR iteration	I_x	$I_t \in \mathbb{R}^{m\times m}$ is an identity matrix

References

1. Averbuch, A., Gabber, E., Gordissky, B., Medan, Y.: A parallel FFT on an MIMD machine. Parallel Comput. **15**, 61–74 (1990)
2. Cooley, J.C., Tukey, J.W.: An algorithm for the machine computation of complex Fourier series. Math. Comput. **19**, 291–301 (1965)
3. Chen, S., Kuck, D.: Time and parallel processor bounds for linear recurrence systems. IEEE Trans. Comput. **C–24**(7), 701–717 (1975)
4. Falgout, R.D., Friedhoff, S., Kolev, T.V., MacLachlan, S.P., Schroder, J.B.: Parallel time integration with multigrid. SIAM J. Sci. Comput. **36**, C635–C661 (2014)
5. Gupta, A., Kumar, V.: The scalability of FFT on parallel computers. IEEE Trans. Parallel Distrib. Syst. **4**, 922–932 (1993)
6. Gander, M.J., Halpern, L., Ryan, J., Tran, T.T.B.: A direct solver for time parallelization. In: Domain Decomposition Methods in Science and Engineering XXII, pp. 491–499. Springer, Berlin (2016)

7. Gander, M.J., Halpern, L.: Time parallelization for nonlinear problems based on diagonalization. In: Domain Decomposition Methods in Science and Engineering XXIII, pp. 163–170. Springer, Berlin (2017)
8. Gander, M.J., Neumüller, M.: Analysis of a new space-time parallel multigrid algorithm for parabolic problems. SIAM J. Sci. Comput. **38**, A2173–A2208 (2016)
9. Gander, M.J., Vandewalle, S.: Analysis of the parareal time-parallel time-integration method. SIAM J. Sci. Comput. **29**, 556–578 (2007)
10. Gander, M.J., Güttel, S.: PARAEXP: a parallel integrator for linear initial-value problems. SIAM J. Sci. Comput. **35**, C123–C142 (2013)
11. Gander, M.J., Jiang, Y.L., Song, B., Zhang, H.: Analysis of two parareal algorithms for time-periodic problems. SIAM J. Sci. Comput. **35**, A2393–A2415 (2013)
12. Gander, M.J.: 50 years of time parallel time integration. In: Carraro, T., Geiger, M., Körkel, S., Rannacher, R. (eds.) Multiple Shooting and Time Domain Decomposition, pp. 69–114. Springer, Berlin (2015)
13. Golub, G.H., Van Loan, C.F.: Matrix Computations, 4th edn. Johns Hopkins Studies in the Mathematical Sciences. Johns Hopkins University Press, Baltimore (2013)
14. Gander, M.J., Al-Khaleel, M., Ruehli, A.E.: Optimized waveform relaxation methods for longitudinal partitioning of transmission lines. IEEE Trans. Circuits Syst. I Regul. Pap. **56**, 1732–1743 (2009)
15. Horton, G., Vandewalle, S., Worley, P.: An algorithm with polylog parallel complexity for solving parabolic partial differential equations. SIAM J. Sci. Comput. **16**, 531–541 (1995)
16. Horton, G., Vandewalle, S.: A space-time multigrid method for parabolic partial differential equations. SIAM J. Sci. Comput. **16**, 848–864 (1995)
17. Hairer, E., Wanner, G.: Solving Ordinary Differential Equations II: Stiff and Differential-Algebraic Problems. Springer, Berlin (1996)
18. Inda, M.A., Bisseling, R.H.: A simple and efficient parallel FFT algorithm using the BSP model. Parallel Comput. **27**, 1847–1878 (2001)
19. Johnsson, S.L., Krawitz, R.L.: Cooley-Tukey FFT on the connection machine. Parallel Comput. **18**, 1201–1221 (1992)
20. Janssen, J., Vandewalle, S.: Multigrid waveform relaxation of spatial finite element meshes: the continuous-time case. SIAM J. Numer. Anal. **33**, 456–474 (1996)
21. Janssen, J., Vandewalle, S.: Multigrid waveform relaxation on spatial finite element meshes: the discrete-time case. SIAM J. Sci. Comput. **17**, 133–155 (1996)
22. Jiang, Y.L.: The Waveform Relaxation Methods (in Chinese). Science Press, Beijing (2009)
23. Kogge, E.: Parallel solution of recurrence problems. IBM J. Res. Dev. **18**, 138–148 (1974)
24. Kogge, E., Stone, S.: A parallel algorithm for the efficient solution of a general class of recurrence equations. IEEE Trans. Comput. **C–22**, 786–793 (1973)
25. Lubich, C., Ostermann, A.: Multigrid dynamic iteration for parabolic equations. BIT **27**, 216–234 (1987)
26. Lelarasmee, E., Ruehli, A.E., Sangiovanni-Vincentelli, A.L.: The waveform relaxation method for time-domain analysis of large scale integrated circuits. IEEE Trans. Comput. Aided Des. Integr. Circuits Syst, **1**, 131–145 (1982)
27. López-Fernández, C.: Palencia, and Schädle, A spectral order method for inverting sectorial Laplace transforms. SIAM J. Numer. Anal. **44**, 1332–1350 (2006)
28. Lions, J.L., Maday, Y., Turinici, G.: A "parareal" in time discretization of PDE's. C. R. Acad. Sci. Paris Sér. I Math. **332**, 661–668 (2001)
29. McDonald, E., Wathen, A.: A simple proposal for parallel computation over time of an evolutionary process with implicit time stepping. Lect. Notes Comput. Sci. Eng. **112**, 285–293 (2016)
30. Miekkala, U., Nevanlinna, O.: Convergence of dynamic iteration methods for initial value problems. SIAM J. Sci. Stat. Comput. **8**, 459–482 (1987)
31. Maday, Y., Rønquist, E.M.: Parallelization in time through tensor product space-time solvers. C. R. Math. Acad. Sci. Paris **346**, 113–118 (2008)
32. Minion, M.L., Speck, R., Bolten, M., Emmett, M., Ruprecht, D.: Interweaving PFASST and parallel multigrid. SIAM J. Sci. Comput. **37**, S244–S263 (2015)
33. Mclean, W., Sloan, I.H., Thomée, V.: Time discretization via Laplace transformation of an integro-differential equation of parabolic type. Numer. Math. **102**, 497–522 (2006)

34. Mclean, W., Thomée, V.: Maximum-norm error analysis of a numerical solution via Laplace transformation and quadrature of a fractional-order evolution equation. IMA J. Numer. Anal. **30**, 208–230 (2010)
35. Nevanlinna, O.: Remarks on Picard-Lindelöf iterations, part I. BIT **29**, 328–346 (1989)
36. Nevanlinna, O.: Remarks on Picard-Lindelöf iterations, part II. BIT **29**, 535–562 (1989)
37. Pippig, M.: PFFT: an extension of FFTW to massively parallel architectures. SIAM J. Sci. Comput. **35**, C213–C236 (2013)
38. Ruehli, A.E., Johnson, T.A.: Circuit Analysis Computing by Waveform Relaxation, in Wiley Encyclopedia of Electrical Electronics Engineering. Wiley, New York (1999)
39. Sheen, D., Sloan, I.H., Thomée, V.: A parallel method for time discretization of parabolic problems based on contour integral representation and quadrature. Math. Comput. **69**, 177–195 (1999)
40. Trefethen, L.N., Weideman, J.A.C.: The exponentially convergent trapezoidal rule. SIAM Rev. **56**, 385–458 (2014)
41. Van Loan, C.: Computational Frameworks for the Fast Fourier Transform. SIAM, Philadelphia (1992)
42. Vandewalle, S., Piessens, R.: Numerical experiments with nonlinear multigrid waveform relaxation on a parallel processor. Appl. Numer. Math. **8**, 149–161 (1991)
43. Vandewalle, S.: Parallel Multigrid Waveform Relaxation for Parabolic Problems. B. G. Teubner, Stuttgart (1993)
44. Vandewalle, S., Piessens, R.: Efficient parallel algorithms for solving initial-boundary value and time-periodic parabolic partial differential equations. SIAM J. Sci. Stat. Comput. **13**, 1330–1346 (1992)
45. Wu, S.L.: Laplace inversion for the solution of an abstract heat equation without the forward transform of the source term. J. Numer. Math. **25**, 185–198 (2017)
46. Wu, S.L., Zhou, T.: Convergence analysis for three parareal solvers. SIAM J. Sci. Comput. **37**, A970–A992 (2015)
47. Wu, S.L.: Toward parallel coarse grid correction for the parareal algorithm. SIAM J. Sci. Comput. **40**, A1446–A1472 (2018)
48. Wu, S.L., Zhou, T.: A diagonalization-based multi-grid method for time-periodic fractional diffusion equations. Numer. Linear Algebra Appl. **25**, e2178 (2018). https://doi.org/10.1002/nla.2178