J Sci Comput (2016) 67:644–668
DOI 10.1007/s10915-015-0100-x

Convergence Analysis of the Parareal-Euler Algorithm for Systems of ODEs with Complex Eigenvalues

Shu-Lin Wu[1]

Received: 23 September 2014 / Revised: 5 May 2015 / Accepted: 7 September 2015 /
Published online: 11 September 2015

Abstract Parareal is an iterative algorithm and is characterized by two propagators $\mathscr{G}$ and $\mathscr{F}$, which are respectively associated with large step size ΔT and small step size Δt, where $\Delta T = J\Delta t$ and $J \geq 2$ is an integer. The choice $\mathscr{G} = \mathscr{F}$ =Backward-Euler denotes the simplest implicit parareal solver, which we call Parareal-Euler, and has been studied widely in recent years. For linear problem $\mathbf{U}'(t) + \mathbf{A}\mathbf{U}(t) = \mathbf{g}(t)$ with $\mathbf{A}$ being a symmetric positive definite matrix, this algorithm converges very fast and the convergence rate is insensitive to the change of J and Δt. However, for the case that the spectrum of $\mathbf{A}$ contains complex values, no provable results are available in the literature so far. Previous studies based on numerical plotting show that we can not expect convergence for the Parareal-Euler algorithm on the whole right-hand side of the complex plane. Here, we consider a representative situation: $\sigma(\mathbf{A}) \subseteq \mathbf{D}(\theta) := \{(x, iy) \in \mathbf{C} : x \geq 0, |y| \leq \tan(\theta)x\}$ with $\theta \in (0, \frac{\pi}{2})$, i.e., the spectrum $\sigma(A)$ is contained in a sectorial region. Spectrum distribution of this type arises naturally for semi-discretizing a wide rang of time-dependent PDEs, e.g., the Fokker-Planck equations. We derive condition, which is independent of J and depends on θ only, to ensure convergence of the Parareal-Euler algorithm. Numerical results for initial value and time-periodic problems are provided to support our theoretical conclusions.

Keywords Parareal algorithm · Backward-Euler · Convergence analysis · Complex eigenvalues

Mathematics Subject Classification 65R20 · 45L05 · 65L20

✉ Shu-Lin Wu
wushulin_ylp@163.com

[1] School of Science, Sichuan University of Science and Engineering, Zigong, Sichuan 643000, People's Republic of China

1 Introduction

In 2001, Lions, Maday and Turinici proposed the so-called *parareal* algorithm to solve ordinary differential equations (ODEs) in the parallel-in-time pattern [20]. The algorithm is iterative and is defined by two numerical propagators, namely the coarse propagator $\mathscr{G}$ and fine propagator $\mathscr{F}$, which are respectively associated with large step size ΔT and small step size Δt, where $\Delta T = J\Delta t$ and $J \geq 2$ is an integer. These two grids are combined in a predictor-corrector manner, in which a novel correction scheme is used to create updates for the entire time-interval. The correction scheme is sequentially performed on the coarse grids, while the computation on the fine grids in each coarse subinterval can start in parallel (more details about this algorithm are given in the next section). Upon convergence, the approximation on the coarse grids will have achieved the accuracy of using the $\mathscr{F}$-propagator step by step with the small step size Δt. Applications and analysis of the parareal algorithm are revisited in the following.

For applications, parareal is successfully used in many fields. Farhat et al. proposed the so-called PITA algorithm, which is a modification of the parareal algorithm and equals to parareal in the liner case, for solving the structural (fluid)-dynamics problems [3,8,9]. Dai et al. proposed the *projection-based-parareal* algorithm for solving the Hamiltonian systems [5,14] and the wave equations [4], which consists of projecting the values obtained by the correction scheme to some specified constant manifold, e.g., the energy manifold, in each iteration. Maday et al. [21,22] and Sarkis et al. [6,23] embedded the parareal algorithm into the numerical discretization of the parabolic PDE-constrained control and optimization, where the parareal algorithm is assigned to treat the most heavy part in the whole computation, e.g., generating good proconditioner. Parareal as well as its other variants [7,16,24,32], e.g., the so-called SDC-Pipelined-Parareal proposed by Minion et al. [7,24], are also successfully used for Volterra integral equations [18], turbulent plasma simulations [25–27], thermo-viscoplastic problems [2], highly oscillatory PDEs [17] and singularly perturbed ODEs [19], etc. (There is an increasing interest in parareal, and it is very possible that some important references are not mentioned here.)

Compared to numerous applications in many fields, convergence properties of the parareal algorithm are less found in the literature. Previous work, such as [1,2,11–13,19,32], studied the convergence properties under the simplified assumption $\mathscr{F}$='exact-numerical-propagator', i.e., $\mathscr{F} = e^{-\mathbf{A}\Delta T}$ for the following linear problem

$$\mathbf{U}'(t) + \mathbf{A}\mathbf{U}(t) = \mathbf{g}(t). \tag{1.1}$$

For the case that the coefficient matrix $\mathbf{A}$ is symmetric and positive definite (SPD), this simplification asserts that: the convergence factor ρ satisfies $\rho \approx 0.3$, independent of the mesh ratio J, Δt and the distribution of the eigenvalues of the coefficient matrix $\mathbf{A}$. (The definition of the convergence factor ρ is given in Theorem 2.1 in the next section.) Note that, this property is important for practical computation: if (1.1) arises from semi-discretizing a time-dependent PDE with mesh size Δx, such an '*independence*' implies that, refining the discretization parameters Δt and Δx to improve the accuracy of the converged solution will not deteriorate the convergence rate of the parareal algorithm. However, this simplified assumption ignores the influence of the mesh ratio J and the formula of a concrete numerical time-integrator, and therefore the conclusion '$\rho \approx 0.3$' does not always coincide with the results observed in practical computation very well. For example, when we choose for $\mathscr{G}$ the Backward-Euler method and for $\mathscr{F}$ the Trapezoidal rule, which leads to the simplest implicit parareal algorithm with order 2 of the converged solution, the convergence rate can be arbitrarily slow and therefore is far away from the prediction by '$\rho \approx 0.3$'. The first work

which treats the real situation is given in [23], there the authors analyzed the convergence properties of the Parareal-Euler algorithm consisting of using the Backward-Euler method as both the coarse and fine propagators, and they proved that the property '$\rho \approx 0.3$' is preserved by this algorithm. Recently, we prove that such a beautiful property also holds for some other familiar choices of the $\mathscr{F}$-propagator [31], such as $\mathscr{F}$=TR/BDF2 (i.e., the ODEs solver `ode23tb` in Matlab), $\mathscr{F}$=2nd-SDIRK and $\mathscr{F}$=3rd-SDIRK. (Here and hereafter, the $\mathscr{G}$-propagator is fixed to the Backward-Euler method.)

All the aforementioned analysis is performed for SPD problems, and so far no provable properties for the parareal algorithm are known when the spectrum of the coefficient matrix $\mathbf{A}$ in (1.1) contains complex numbers. For this aspect, Bal [1], Gander and Vandewalle [13], and Celledoni and Kvamsdal [2] provided some heuristic experience, who investigated the convergence properties of the parareal algorithm in a *case-by-case* manner: for a given mesh ratio J, plotting the contour of the contraction factor of the algorithm on the right-hand side of the complex plane and then observing the behavior of the contraction factor. A clarification is necessary here: 'contraction factor' denotes the error damping factor corresponding to a single eigenvalue of $\mathbf{A}$, while 'convergence factor' denotes the maximal value of the 'contraction factor' over all eigenvalues (the concrete definitions of 'convergence factor' and 'contraction factor' will be given in Sect. 2). Their investigation shows that: we can not expect $\rho < 1$ on the whole right-hand side of the complex plane and the contraction factor presents high complexity as a function of $z := \Delta T\lambda$, where λ denotes the eigenvalue of the matrix $\mathbf{A}$. In the face of this, however, we should not stop our steps to exploring the convergence properties of the parareal algorithm more carefully and pertinently. Precisely, for a wide range of problems, the distribution of $\sigma(\mathbf{A})$ [i.e., the spectrum of the coefficient matrix $\mathbf{A}$ in (1.1)] presents some special characteristics and this should be taken into account for the convergence analysis. A representative distribution of $\sigma(\mathbf{A})$ is $\sigma(\mathbf{A}) \subset \mathbf{D}(\theta) := \{(x, iy) \in \mathbf{C} : |y| \leq \tan(\theta)x, x \geq 0\}$, where the domain $\mathbf{D}(\theta)$ is shown in Fig. 1. Spectrum distribution of this type arises frequently from semi-discretizing time-dependent PDEs, e.g., the Fokker-Planck equations semi-discretized by the mixed Hermite spectral-finite difference method [10,30], which will be described in detail in Sect. 4.

In this paper, for the model problem (1.1) with spectrum $\sigma(\mathbf{A})$ satisfying the just mentioned sectorial distribution, we analyze the convergence properties for the Parareal-Euler algorithm. We derive sharp estimate for the convergence factor, which is independent of the mesh ratio J and depends on the angle θ only. Our analysis is essentially different from the previous 'case-by-case' study based on contour plotting. The remainder of this paper is organized as follows. In Sect. 2, we review the parareal framework and the result given by Gander

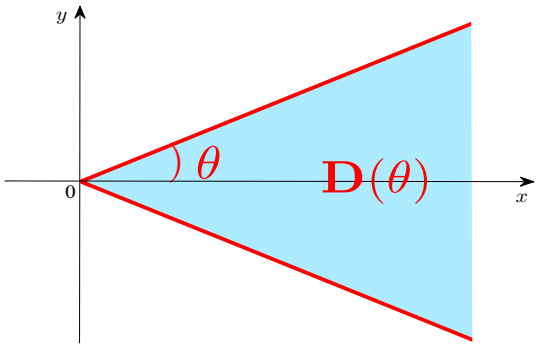

Fig. 1 A representative distribution of $\sigma(\mathbf{A})$ on the complex plane

and Vandewalle [13], which is the starting point of our analysis. Our main results, the sharp estimate for the convergence factor and the related proofs, are presented in Sect. 3. In Sect. 4, we first discuss how to embed the the work in this paper into the work by Tang et al. [10,30] about the mixed Hermite spectral-finite difference method with scaling factor for the Fokker-Planck equations. Then, we provide numerical results to verify our theoretical conclusions for the Parareal-Euler algorithm. We conclude this paper in Sect. 5.

2 The Parareal Framework

For ODEs system:

$$\begin{cases} u'(t) = f(t, u(t)), & t \in [0, T], \\ u(0) = u_0, & t = 0, \end{cases} \tag{2.1}$$

where the function $f: \mathbb{R} \times \mathbb{R}^m \to \mathbb{R}^m$ is Lipschitz continuous, the parareal algorithm studied in this paper can be described as follows. First, the whole time-interval $[0, T]$ is divided into $\mathbb{N}$ large time-intervals $[T_n, T_{n+1}]$, $n = 0, 1, \ldots, \mathbb{N}-1$. (We suppose that all the large time-intervals are of uniform size, i.e., $T_{n+1} - T_n = \Delta T = \frac{T}{\mathbb{N}}$). Second, we divide each large time-interval $[T_n, T_{n+1}]$ into $J(\geq 2)$ small time-intervals $\left[T_{n+\frac{j\Delta T}{J}}, T_{n+\frac{(j+1)\Delta T}{J}}\right]$, $j = 0, 1, \ldots, J-1$. We designate by the symbol $\ominus$ the time sequential parts of the algorithm, and by the symbol $\oplus$ the time-parallel parts. Then, the parareal algorithm studied in this paper is given as Algorithm 2.1.

Algorithm 2.1 Parareal algorithm

$\ominus$ **Initialization**: Choose $\{u_n^0\}_{n=1}^{\mathbb{N}}$ randomly or by the sequential computation $u_{n+1}^0 = \mathscr{G}(T_n, u_n^0, \Delta T)$ with $u_0^0 = u_0$ and $n = 0, 1, \ldots, \mathbb{N}-1$;

For $k = 0, 1, \ldots$

$\oplus$ **Step-I** On each large time-interval $[T_n, T_{n+1}]$, perform the following computation with small step size Δt:

$$u_{n+\frac{j+1}{J}} = \mathscr{F}\left(T_{n+\frac{j}{J}}, u_{n+\frac{j}{J}}, \Delta t\right) \text{ with } u_n = u_n^k, \ j = 0, \ldots, J-1, \tag{2.2}$$

$\ominus$ **Step-II** Perform sequential corrections $u_{n+1}^{k+1} = \mathscr{G}\left(T_n, u_n^{k+1}, \Delta T\right) + u_{n+1} - \mathscr{G}\left(T_n, u_n^k, \Delta T\right)$, where $u_0^{k+1} = u_0$ and $n = 0, 1, \ldots, \mathbb{N}-1$;

$\ominus$ **Step-III** If $\left\{u_n^{k+1}\right\}$ satisfy stopping criteria, terminate the iteration; otherwise turn to **Step-I**.

Let $\mathscr{F}^J\left(T_n, u_n^k, \Delta t\right)$ be the value calculated by (2.2). Then, Algorithm 2.1 can be rewritten as

$$u_{n+1}^{k+1} = \mathscr{G}\left(T_n, u_n^{k+1}, \Delta T\right) + \mathscr{F}^J\left(T_n, u_n^k, \Delta t\right) - \mathscr{G}\left(T_n, u_n^k, \Delta T\right), \ n = 0, 1, \ldots, \mathbb{N}-1. \tag{2.3}$$

Since the Backward-Euler method is stable for all possible choices of ΔT and in the parareal framework the coarse propagator is forced to take large step size, using this method as the $\mathscr{G}$-propagator is therefore reasonable. Throughout this paper, we use the Backward-Euler

method as the coarse propagator. Moreover, we assume that the matrix $\mathbf{A}$ can be diagonalized. Then, following the analysis given by Gander and Vandewalle [13], we can get the following result, which is the starting point of our analysis in the next section.

Theorem 2.1 (General result deduced from [13]) *Let both the $\mathscr{G}$- and $\mathscr{F}$-propagators be chosen as the Backward-Euler method. Let $\mathbf{A} \in \mathbf{C}^{m\times m}$ in (1.1) be a diagonalizable matrix with spectrum $\sigma(\mathbf{A}) = \{\lambda_1, \cdots, \lambda_m\}$ and $\sigma(\mathbf{A}) \subset \mathbf{C}^+ \cup \{(x, iy) \in \mathbf{C} : x = 0, y \in \mathbf{R}\}$. Then, the errors $\left\{\mathbf{e}_n^k\right\}$ of the Parareal Algorithm 2.1 satisfy*

$$\sup_n \|V\mathbf{e}_n^k\|_\infty \le \rho^k \sup_n \|V\mathbf{e}_n^0\|_\infty, \tag{2.4}$$

where $V \in \mathbb{R}^{m\times m}$ consists of the eigenvectors of $\mathbf{A}$, $k \ge 1$ is the iteration index, and the quantity ρ, which we call the convergence factor of the parareal algorithm, is defined by

$$\rho := \max_{z=\Delta T\lambda, \lambda\in\sigma(\mathbf{A})} \mathscr{K}(z, J) \text{ with } \mathscr{K}(z, J) := \frac{\left|\left(\frac{1}{1+\frac{z}{J}}\right)^J - \frac{1}{1+z}\right|}{1 - \left|\frac{1}{1+z}\right|}. \tag{2.5}$$

We now show some plots to illustrate the behavior of the quantity $\mathscr{K}$, which we call the *contraction factor* of the parareal algorithm. Let $z = \frac{y}{\tan(\theta)} + iy$ (with $y \ge 0$) in (2.5). Then, we show $\mathscr{K}$ as a function of y. For three choices of θ, we show $\mathscr{K}$ for the Parareal-Euler algorithm in Fig. 2. We see that the mesh ratio J has a visible effect on the behavior of $\mathscr{K}$ and that we can not expect $\mathscr{K} < 1$ for all $\theta \in (0, \pi/2)$, since for θ close to $\pi/2$ there exists a rang of y such that $\mathscr{K} > 1$, as shown clearly in Fig. 2 on the bottom.

3 Convergence Analysis

In this section, we perform a convergence analysis for the Parareal-Euler algorithm, when the spectrum $\sigma(\mathbf{A})$ satisfies a distribution as shown in Fig. 1. For any integer $J \ge 2$, it is easy to know that the contraction factor $\mathscr{K}$ is analytic on the right-hand side of the complex plane. Moreover, the limit of $\mathscr{K}$ for $z = re^{i\tau}$ with $\tau \in (-\pi/2, \pi/2)$ is zero as $r \to +\infty$, i.e., we have one limit in all directions. By using the maximal principle of analytic functions and the symmetry of the contraction factor $\mathscr{K}$ with respect to the real axis, we have

$$\max_{z=x+iy\in\mathbf{D}(\theta))} \mathscr{K}(z, J) = \max_{z=y/\kappa+iy, y\ge 0} \mathscr{K}(z, J), \text{ with } \kappa = \tan(\theta). \tag{3.1}$$

Define

$$\begin{aligned} \gamma(s, y) &= \left(\left(1 + s\frac{y}{\kappa}\right)^2 + s^2y^2\right)^{\frac{1}{2s}}, \ \omega(s, y) = \arccos\left(\frac{1 + s\frac{y}{\kappa}}{\gamma^s(s, y)}\right), \\ \mathscr{N}(s, y) &= \left(\frac{\gamma(1, y)}{\gamma(s, y)}\right)^2 - 2\cos\left(\frac{\omega(s, y)}{s} - \omega(1, y)\right)\left(\frac{\gamma(1, y)}{\gamma(s, y)}\right) + 1. \end{aligned} \tag{3.2}$$

Then, by letting $s = \frac{1}{J}$ in $\mathscr{N}$, some algebra shows

$$\mathscr{K}(z, J)\big|_{z=\frac{y}{\kappa}+iy} = \frac{\sqrt{\mathscr{N}\left(\frac{1}{J}, y\right)}}{\gamma(1, y) - 1}. \tag{3.3}$$

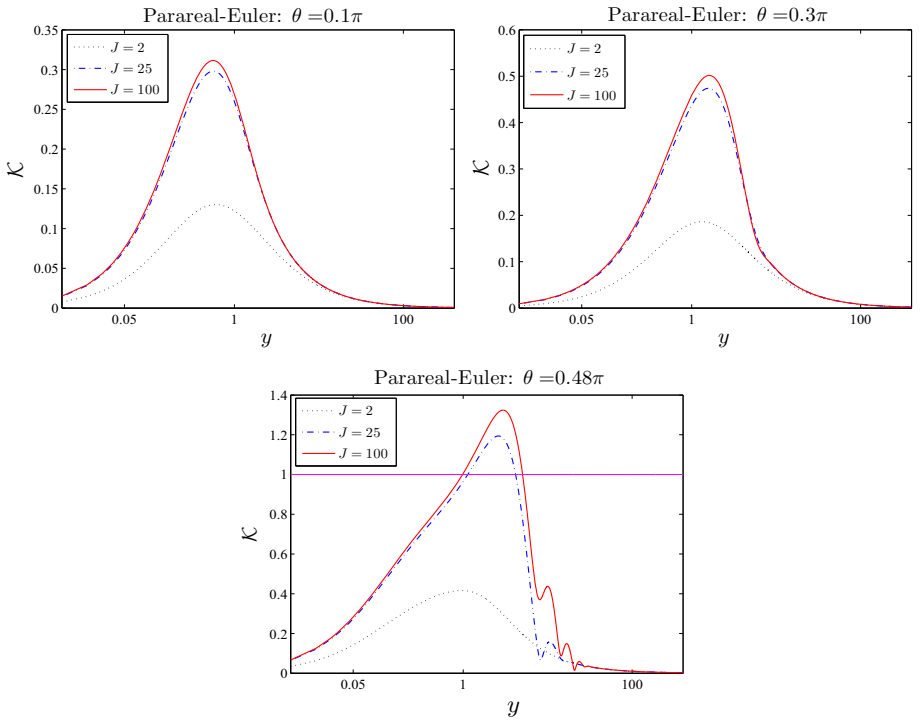

Fig. 2 For $z = \frac{y}{\tan(\theta)} + iy$, the contraction factor $\mathscr{K}$ of the Parareal-Euler algorithm as a function of y under three choices of the mesh ratio J. *Top left* $\theta = 0.1\pi$. *Top right* $\theta = 0.3\pi$. *Bottom* $\theta = 0.48\pi$

Clearly, for any $s > 0$, it holds that $\mathscr{N}(s, y) \le \left(\frac{\gamma(1,y)}{\gamma(s,y)}\right)^2 + 2\left(\frac{\gamma(1,y)}{\gamma(s,y)}\right) + 1 = \left(\frac{\gamma(1,y)}{\gamma(s,y)} + 1\right)^2$, which gives

$$\mathscr{K}(z, J)\big|_{z=\frac{y}{\kappa}+iy} \le \widetilde{\mathscr{K}}\left(\frac{1}{J}, y\right), \text{ with } \widetilde{\mathscr{K}}(s, y) = \frac{\gamma(1, y)/\gamma(s, y) + 1}{\gamma(1, y) - 1}. \tag{3.4}$$

The following lemma, which reveals some special properties of the functions $\gamma(s, y)$, $\omega(s, y)$ and $\widetilde{\mathscr{K}}(s, y)$, is useful for deriving a sharp and J-independent upper bound of the contraction factor of the Parareal-Euler algorithm.

Lemma 3.1 *Let $\gamma(s, y)$ and $\omega(s, y)$ be the functions defined by (3.2), and $\widetilde{\mathscr{K}}(s, y)$ be the function defined by (3.4). Then, for $\kappa > 0$ it holds that*

1. $\partial_s\left(\frac{\omega(s,y)}{s}\right) < 0$, $\forall s > 0$ *and* $\forall y > 0$;
2. $\partial_y \widetilde{\mathscr{K}}(s, y) < 0$, $\forall s > 0$ *and* $\forall y > 0$;
3. $\min_{s\in\left(0,\frac{1}{3}\right]} \gamma(s, y) = \min\left\{e^{y/k}, \gamma(\frac{1}{3}, y)\right\}$ *and* $\max_{s\in\left(0,\frac{1}{3}\right]} \gamma(s, y) = \gamma(s_*, y)$, $\forall y > 0$.

The quantity s_* in the third claim depends on y and is defined by

$$s_* = \begin{cases} 0, & \text{if } 1 \ge \kappa, \\ \frac{1}{3}, & \text{if } 1 < \kappa, \gamma_1(\frac{1}{3}, y) \ge 0, \\ s_0, & \text{if } 1 < \kappa, \gamma_1(\frac{1}{3}, y) < 0, \end{cases} \tag{3.5}$$

where $\gamma_1(s,y) = 2sy\frac{sy(1+\kappa^{-2})+\kappa^{-1}}{(1+sy/\kappa)^2+(sy)^2} - \log[(1+sy/\kappa)^2+(sy)^2]$ *and* s_0 *is the unique positive root of* $\gamma_1(s,y)=0$.

Proof Let $\tilde{s} = \frac{1}{s}$. Then, it holds that $\frac{\omega(s,y)}{s} = \tilde{\omega}(\tilde{s},y) := \tilde{s}\arccos\left(\left(1+\frac{y}{\kappa\tilde{s}}\right)\left(\left(1+\frac{y}{\kappa\tilde{s}}\right)^2 + \frac{y^2}{\tilde{s}^2}\right)^{-\frac{1}{2}}\right)$. We have

$$\partial_{\tilde{s}}\tilde{\omega}(\tilde{s},y) = \tilde{\omega}_1(\tilde{s},y) := \arccos\left(\frac{\tilde{s}+y/\kappa}{\sqrt{(\tilde{s}+y/\kappa)^2+y^2}}\right) - \frac{\tilde{s}y}{(\tilde{s}+y/\kappa)^2+y^2},$$
$$\partial_y\tilde{\omega}_1(\tilde{s},y) := \frac{2\tilde{s}y\left(y(1+\kappa^{-2})+s/\kappa\right)}{[(\tilde{s}+y/\kappa)^2+y^2]^2} > 0.$$

This, together with $\tilde{\omega}_1(\tilde{s},0) = \arccos(1) = 0$, implies that $\tilde{\omega}_1(\tilde{s},y) > 0$ for all $y>0$ and $\tilde{s}>0$. Hence, $\partial_{\tilde{s}}\tilde{\omega}(\tilde{s},y) > 0$ and this implies $\partial_s\left(\frac{\omega(s,y)}{s}\right) < 0$, since $\tilde{s} = \frac{1}{s}$. To prove the second claim, we rewrite $\widetilde{\mathscr{K}}(s,y)$ as

$$\begin{aligned}\widetilde{\mathscr{K}}(s,y) = \frac{Y(y)}{Y(y)-1}\times\frac{1}{[(1+sy/\kappa)^2+s^2y^2]^{\frac{1}{2s}}} \\ + \frac{1}{Y(y)-1}, \quad \text{with } Y(y)=\sqrt{(1+y/\kappa)^2+y^2}.\end{aligned} \tag{3.6}$$

Clearly, all the three terms $\frac{Y(y)}{Y(y)-1}$, $\frac{1}{Y(y)-1}$ and $[(1+sy/\kappa)^2+s^2y^2]^{-\frac{1}{2s}}$ are non-negative and it is easy to verify that they are decreasing functions of y. Hence, we get $\partial_y\widetilde{\mathscr{K}}(s,y) < 0$ for $s>0$ and $y>0$.

It remains to prove the third claim. Let $\tilde{\gamma}(s) = \frac{\log[(1+sy/\kappa)^2+s^2y^2]}{s}$. Then, it holds that $\gamma(s,y) = e^{\frac{\tilde{\gamma}(s)}{2}}$. With $\gamma_1(s,y) = 2sy\frac{sy(1+\kappa^{-2})+\kappa^{-1}}{(1+sy/\kappa)^2+(sy)^2} - \log[(1+sy/\kappa)^2+(sy)^2]$, we have $\tilde{\gamma}'(s) = \frac{1}{s^2}\gamma_1(s,y)$. Routine calculation yields

$$\partial_s\gamma_1(s,y) = -2s(\kappa^2+1)y^3\left(s-\frac{\kappa(\kappa-1)}{y(\kappa^2+1)}\right)\times\frac{\kappa(\kappa+1)+s(\kappa^2+1)y}{[(\kappa+sy)^2+\kappa^2s^2y^2]^2}. \tag{3.7}$$

Based on (3.7), we consider three cases in the following.

1. $\kappa\in(0,1]$. In this case, we have $\partial_s\gamma_1(s,y)\le 0$ for $s\in\left(0,\frac{1}{3}\right]$. This, together with $\gamma_1(0,y)=0$, gives $\gamma_1(s,y)\le 0$ for $s\in\left(0,\frac{1}{3}\right]$. Therefore, $\min_{s\in\left(0,\frac{1}{3}\right]}\gamma(s,y) = \gamma(1/3,y)$ and $\max_{s\in\left(0,\frac{1}{3}\right]}\gamma(s,y) = \lim_{s\to 0}\gamma(s,y) = e^{y/\kappa}$.
2. $\kappa>1$ and $\gamma_1(1/3,y)\ge 0$. For $\kappa>1$, it is easy to see that $\partial_s\gamma_1(s,y)=0$ has a unique positive root $s=\frac{\kappa(\kappa-1)}{y(\kappa^2+1)}$ and that $\partial_s\gamma_1(s,y)>0$ for all $s\in\left(0,\frac{\kappa(\kappa-1)}{y(\kappa^2+1)}\right)$ and $\partial_s\gamma_1(s,y)<0$ for $s>\frac{\kappa(\kappa-1)}{y(\kappa^2+1)}$. Hence, if $\gamma_1(1/3,y)\ge 0$, it must hold $\gamma_1(s,y)\ge 0$ for $s\in(0,1/3]$ and $y>0$, i.e., $\gamma(s,y)$ is an increasing function of s. Therefore, $\min_{s\in\left(0,\frac{1}{3}\right]}\gamma(s,y) = \lim_{s\to 0}\gamma(0,y) = e^{y/\kappa}$ and $\max_{s\in\left(0,\frac{1}{3}\right]}\gamma(s,y) = \gamma(1/3,y)$.
3. $\kappa>1$ and $\gamma_1(1/3,y)<0$. In this case, γ_1 is respectively increasing and decreasing function for $s\in\left(0,\frac{\kappa(\kappa-1)}{y(\kappa^2+1)}\right)$ and $s>\frac{\kappa(\kappa-1)}{y(\kappa^2+1)}$. Since $\gamma_1(0,y)=0$ and $\gamma_1(1/3,y)<0$, we know that $\gamma_1(s,y)$ has a unique positive root $s_0\in\left(\frac{\kappa(\kappa-1)}{y(\kappa^2+1)},\frac{1}{3}\right]$, which corresponds

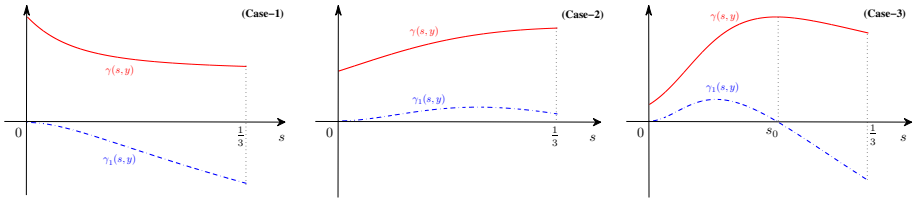

Fig. 3 Three relations between $\gamma_1(s, y)$ and $\gamma(s, y)$, which corresponds to the three cases in the proof of the third claim

to the global maximizer of $\gamma(s, y)$, i.e., $\max_{s\in\left(0,\frac{1}{3}\right]}\gamma(s, y) = \gamma(s_0, y)$. The function $\gamma(s, y)$ is respectively increasing and decreasing for $s \in (0, s_0)$ and $s \in (s_0, 1/3]$; hence,

$$\min_{s\in\left(0,\frac{1}{3}\right]}\gamma(s, y) = \min\left\{\lim_{s\to 0}\gamma(s, y), \gamma(1/3, y)\right\} = \min\{e^{y/\kappa}, \gamma(1/3, y)\}.$$

An illustration of the above three cases is shown in Fig. 3. □

Using the three results given by Lemma 3.1, we can derive a J-independent estimate of the contraction factor of the Parareal-Euler algorithm.

Theorem 3.1 *Let* $\mathbf{D}(\theta) = \{(x, iy) \in \mathbf{C} : |y| \le \kappa x, x \ge 0\}$ *with* $\kappa = \tan(\theta)$ *and* $\theta \in (0, \frac{\pi}{2})$, $\phi(y) = y - \omega(1, y)$ *and* $y_* \in [\pi, 4.5]$ *be the unique root of* $\phi(y) = \pi$. *Then, we have*

$$\mathscr{K}(z, J) := \frac{\left|\left(\frac{1}{1+\frac{z}{J}}\right)^J - \frac{1}{1+z}\right|}{1 - \left|\frac{1}{1+z}\right|} \le \widetilde{\rho}(\theta), \ \forall z \in \mathbf{D}(\theta) \text{ and } \forall J \ge 3, \tag{3.8}$$

where $\widetilde{\rho}(\theta)$ *is defined by* $\widetilde{\rho}(\theta) = \max_{y\in[0,y_*]}\sqrt{\widetilde{\mathscr{N}}(y)}\big/[\gamma(1, y) - 1]$ *and*

$$\begin{aligned}
&\widetilde{\mathscr{N}}(y) = \max\left\{\gamma_{\min}^2(y) - 2\cos(\phi(y))\gamma_{\min}(y),\ \gamma_{\max}^2(y) - 2\cos(\phi(y))\gamma_{\max}(y)\right\} + 1,\\
&\gamma_{\min}(y) = \frac{\gamma(1, y)}{\gamma(s_*, y)} (\text{the quantity } s_* \text{ is given by (3.5)}),\\
&\gamma_{\max}(y) = \frac{\gamma(1, y)}{\min\{\gamma(0, y), \gamma(1/3, y)\}}.
\end{aligned} \tag{3.9}$$

Before we start the proof, we first point out that the quantity y_* depends on θ and that y_* is an increasing function of θ (this can be verified by routine calculations). Therefore, for $\theta \in (0, \pi/2)$ the minimal and maximal values of y_* are respectively π and 4.4934, i.e., $y_* \in [\pi, 4.5]$.

Proof For $y \in [0, y_*]$ and $s \in (0, 1)$, using the first result in Lemma 3.1, we have

$$0 \le \frac{\omega(s, y)}{s} - \omega(1, y) \le \lim_{s\to 0}\frac{\omega(s, y)}{s} - \omega(1, y) = y - \omega(1, y) \le \pi.$$

(The limit is obtained by using the L'Hôpital's rule.) Then, for the function $\mathscr{N}(s, y)$ defined by (3.2), we have

$$\begin{aligned}\mathscr{N}(s, y) &\le \left(\frac{\gamma(1, y)}{\gamma(s, y)}\right)^2 - 2\cos(y - \omega(1, y))\left(\frac{\gamma(1, y)}{\gamma(s, y)}\right) + 1 \\ &= \left(\frac{\gamma(1, y)}{\gamma(s, y)}\right)^2 - 2\cos(\phi(y))\left(\frac{\gamma(1, y)}{\gamma(s, y)}\right) + 1. \end{aligned} \tag{3.10}$$

Moreover, by using the third result in Lemma 3.1, we have $\gamma_{\min}(y) \le \frac{\gamma(1,y)}{\gamma(s,y)} \le \gamma_{\max}(y)$, where $\gamma_{\min}(y)$ and $\gamma_{\max}(y)$ are defined by (3.9). Hence, by regarding $\left(\frac{\gamma(1,y)}{\gamma(s,y)}\right)$ as a variable in (3.10), it holds that

$$\begin{aligned}\mathscr{N}(s, y) \le \max&\left\{\gamma_{\min}^2(y) - 2\cos(\phi(y))\gamma_{\min},\ \gamma_{\max}^2(y) - 2\cos(\phi(y))\gamma_{\max}\right\} \\ &+1, \forall s \in \left(0, \frac{1}{3}\right],\end{aligned}$$

i.e., $\mathscr{N}(s, y) \le \widetilde{\mathscr{N}}(y)$, where $\widetilde{\mathscr{N}}(y)$ is defined by (3.9). Letting $s = \frac{1}{J}$ and then using (3.3), we have

$$\mathscr{K}(z, J)\big|_{z=\frac{y}{\kappa}+iy} \le \frac{\sqrt{\widetilde{\mathscr{N}}(y)}}{\gamma(1, y) - 1},\ \forall y \in [0, y_*], \forall J \ge 3. \tag{3.11}$$

Now, by noticing (3.1) we see that, to arrive at (3.8), it suffices to prove

$$\mathscr{K}(z, J)\big|_{z=\frac{y}{\kappa}+iy} \le \frac{\sqrt{\widetilde{\mathscr{N}}(y_*)}}{\gamma(1, y_*) - 1},\ \forall y > y_*, \forall J \ge 3, \tag{3.12}$$

since this inequality implies that the function $\mathscr{K}(z, J)\big|_{z=\frac{y}{\kappa}+iy}$ (with $J \ge 3$) always lies below the line $\frac{\sqrt{\widetilde{\mathscr{N}}(y_*)}}{\gamma(1,y_*)-1}$ for $y > y_*$. An illustration of (3.11) and (3.12) is shown in Fig. 4.

For $y \ge y_*$, by using (3.4) and the second result in Lemma 3.1, we have

$$\mathscr{K}(z, J)\big|_{z=\frac{y}{\kappa}+iy} \le \widetilde{\mathscr{K}}\left(\frac{1}{J}, y\right) \le \widetilde{\mathscr{K}}\left(\frac{1}{J}, y^*\right). \tag{3.13}$$

Since $\phi(y_*) = \pi$, from the definition of the function $\widetilde{\mathscr{N}}(y)$ given by (3.9), we have

$$\begin{aligned}\widetilde{\mathscr{N}}(y_*) = (1 + \gamma_{\max}(y_*))^2 &= \left(1 + \frac{\gamma(1, y_*)}{\min_{s\in(0,1/3]}\gamma(s, y_*)}\right)^2 \\ &\ge \left(\frac{\gamma(1, y_*)}{\gamma(s, y_*)} + 1\right)^2,\ \forall s \in \left(0, \frac{1}{3}\right].\end{aligned} \tag{3.14}$$

where the first '=' follows from the fact that $\gamma_{\max}(y) > \gamma_{\min}(y)$ for $y > 0$. (For any $y > 0$, the third result of Lemma 3.1 implies that $\min\{\gamma(0, y), \gamma(1/3, y)\}$ and $\gamma(s_*, y)$ are respectively minimum and maximum of $\gamma(s, y)$ for $s \in (0, 1/3]$, and therefore $\gamma_{\max}(y) > \gamma_{\min}(y)$ follows.) Hence, by letting $s = \frac{1}{J}$ in (3.14) we see that

$$\frac{\sqrt{\widetilde{\mathscr{N}}(y_*)}}{\gamma(1, y_*) - 1} \ge \frac{\frac{\gamma(1,y_*)}{\gamma(1/J,y_*)} + 1}{\gamma(1, y_*) - 1} = \widetilde{\mathscr{K}}\left(\frac{1}{J}, y_*\right),\ \forall J \ge 3, \tag{3.15}$$

where the '=' follows from the definition of $\widetilde{\mathscr{K}}$ given by (3.4). Clearly, (3.13) and (3.15) give (3.12), which, together with (3.1) and (3.11), implies (3.8). □

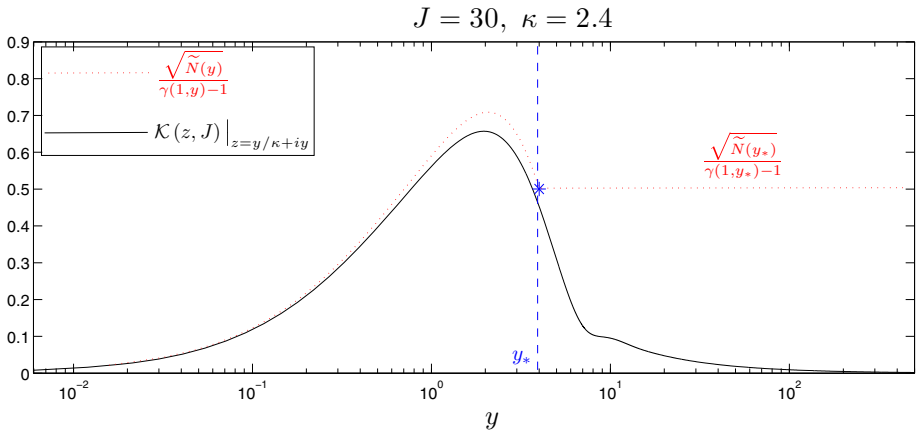

Fig. 4 Illustration of (3.11) (*left side* of the *dashed vertical line*) and (3.12) (*right side* of the *dashed vertical line*)

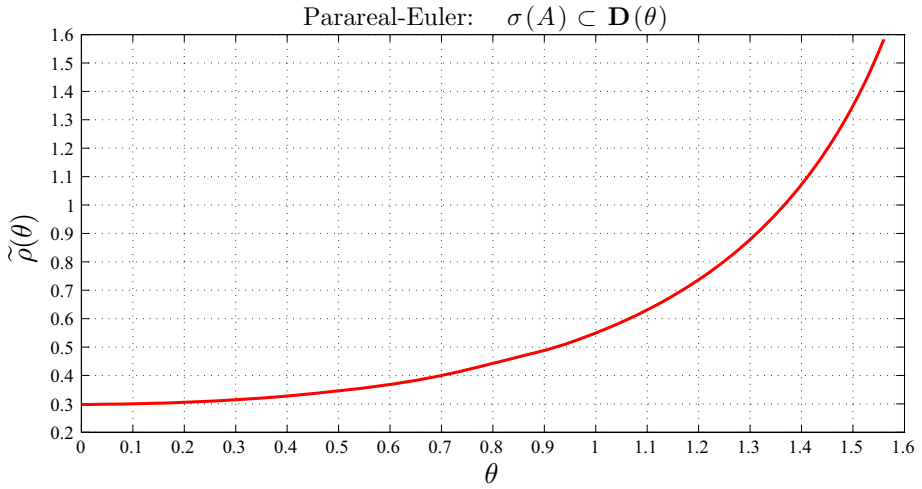

Fig. 5 The profile of $\widetilde{\rho}(\theta)$ as a function of θ

Table 1 Parareal-Euler: some typical values of $\widetilde{\rho}$ and the corresponding θ

$\sigma(\mathbf{A}) \subset \mathbf{D}(\theta)$						
$\widetilde{\rho} \leq$	$\frac{1}{3}$	0.4	0.5	0.6	0.8	1
$\theta \leq$	0.435 (24°55′)	0.7 (40°6′)	0.923 (52°53′)	1.06 (60°44′)	1.25 (71°37′)	1.365 (78°12′)

The bound of the contraction factor $\mathcal{K}(z, J)$, i.e., $\widetilde{\rho}$, only depends on θ. For given θ, it is easy to get this quantity through numerical optimization, e.g., by using the chebfun commands '`min`' and '`max`' in Matlab. In Fig. 5, we show the profile of $\widetilde{\rho}$ as a function of θ. As expected, $\widetilde{\rho}$ is an increasing function of θ. Some typical values of $\widetilde{\rho}$ and the corresponding θ are listed in Table 1.

Remark 3.1 (what can we get from Fig. 5 and Table 1?) The results shown in Fig. 5 and Table 1 provide us an important information: *although a larger amplitude angle θ has a worse effect on the convergence rate of the Parareal-Euler algorithm, this deterioration effect is negligible for $\theta \in [0, 0.7]$, because $\widetilde{\rho}$ grows slowly for $\theta \in [0, 0.7]$ and the increment of $\widetilde{\rho}$ is at most 10 %.* This conclusion has practical implication for computation. Suppose the initial error $\{\mathbf{e}_n^0\}$ satisfies $\max_n \|V\mathbf{e}_n^0\|_\infty = \mathscr{O}(1)$ and the machine precision is $2^{-53} \approx 1.1 \times 10^{-16}$. For SPD problem (i.e., $\theta = 0$), it has been observed in many places, e.g., [23,31], that the Parareal-Euler algorithm converges rapidly with a convergence factor around 0.3, and thus we need approximately 30 iterations to reach the machine precision. Therefore, the just mentioned '*10 % increment*' implies that: compared to the iteration number required for solving the SPD problem, at most 3 additional iterations are needed when we are now solving a non-SPD problem with $\theta \leq 0.7$. If we do not need to reach the machine precision (e.g., 10^{-8} or 10^{-12} in most cases), the additional iteration number could be 0 or 1. We will show numerical evidences in the next section for this.

It would be interesting to check whether θ is indeed the only parameter controlling the convergence rate of the algorithm. For this respect, we compare the results for three matrices $\mathbf{A}$:

(a) in the first one, all eigenvalues share the same positive real part x_0, and their imaginary part satisfies $|y| \leq x_0 \tan(\theta_0)$;
(b) in the second one, all eigenvalues share the same positive real part $x_1 > x_0$, and their imaginary part satisfies $|y| \leq x_1 \tan(\theta_0)$;
(c) in the third one, all eigenvalues share the same positive real part x_0 (the same as in the first matrix), and their imaginary part satisfies $|y| \leq x_0 \tan(\theta_1)$ with $\theta_1 > \theta_0$.

We consider the following ODEs system with zero initial condition $\mathbf{U}(0) = 0$:

$$\mathbf{U}'(t) + \mathbf{A}\mathbf{U}(t) = \begin{bmatrix} -\sin(18t) \\ -\sin(17t) \\ \vdots \\ 0 \\ \vdots \\ \sin(17t) \\ \sin(18t) \end{bmatrix}, \text{ with } \mathbf{A} = \begin{bmatrix} d_{18}^- & & & & & & \\ \frac{1}{4} & d_{17}^- & & & & & \\ & \ddots & \ddots & & & & \\ & & \frac{1}{4} & 0 & & & \\ & & & \ddots & \ddots & & \\ & & & & \frac{1}{4} & d_{17}^+ & \\ & & & & & \frac{1}{4} & d_{18}^+ \end{bmatrix}, \tag{3.16}$$

where $d_j^\pm = x\left(1 \pm \frac{j\tan(\theta)}{18}\right)$. We choose two values for θ: $\theta_0 = \frac{1}{3}$ and $\theta_1 = 1.2$, and two values for x: $x_0 = 0.2$ and $x_1 = 0.4$. The three coefficient matrices corresponding to the above criterion (a), (b) and (c) are denoted by $\mathbf{A}_a$, $\mathbf{A}_b$ and $\mathbf{A}_c$. The spectrums of these three matrices are shown in Fig. 5 on the left. On the right, we show the measured convergence rates of the Parareal-Euler algorithm applied to (3.16) with $\mathbf{A} = \mathbf{A}_a$, $\mathbf{A} = \mathbf{A}_b$ and $\mathbf{A} = \mathbf{A}_c$. We see that the first matrix $\mathbf{A}_a$ and the second one $\mathbf{A}_b$ give almost the same results in terms of convergence, while $\mathbf{A}_a$ and $\mathbf{A}_c$ give different results. These numerical plotting shown in Fig. 6 on the right imply that the convergence rate of the Parareal-Euler algorithm is mainly dominated by the angle θ.

We next provide evidence to show that the estimate $\widetilde{\rho}$ of the convergence factor is sharp. To this end, we continue to use the ODEs system (3.16) but now with $d_j^\pm = \frac{j(1\pm\tan(\theta)i)}{15}$. By choosing $\theta = 1.36$, 1.38, 1.39 and 1.40, we get four spectrums $\sigma(\mathbf{A})$ as shown in Fig. 7 on the left. (These values of θ are chosen such that $\widetilde{\rho}(\theta)$ is close to 1, i.e., the region where

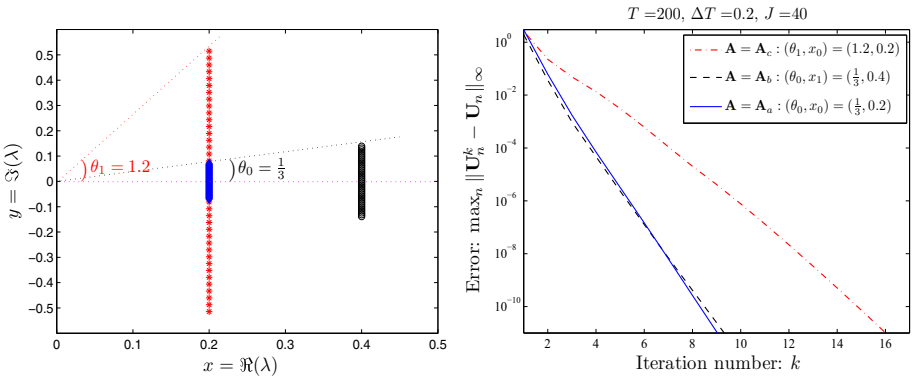

Fig. 6 *Left* the spectrum of the matrix $\mathbf{A}$ in (3.16) with $d_j^{\pm} = x\left(1 \pm \frac{j\tan(\theta)}{18}\right)$ and $(x, \theta) = (0.2, \frac{1}{3})$, $(0.2, 1.2)$ and $(0.4, \frac{1}{3})$. *Right* comparison of the convergence rates of the Parareal-Euler algorithm for the three matrices

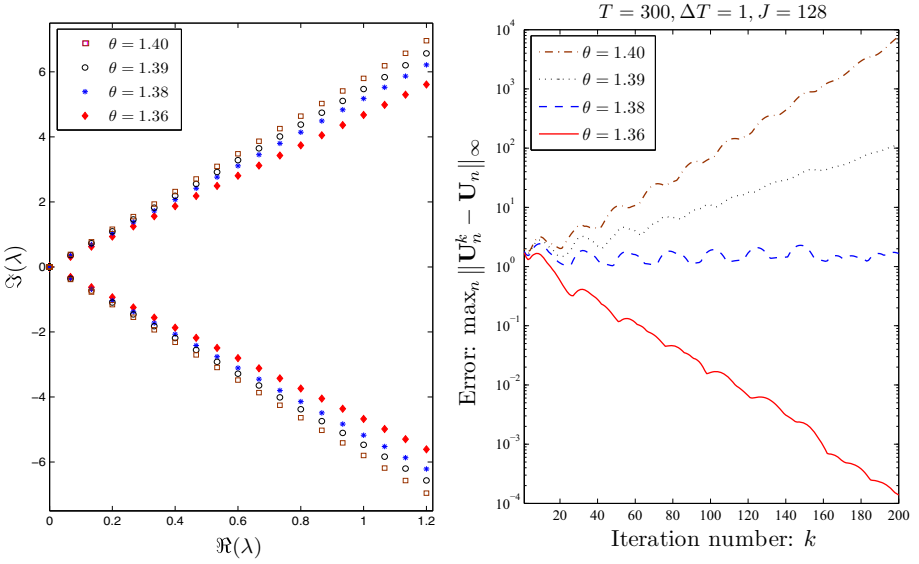

Fig. 7 *Left* the spectrum of the matrix $\mathbf{A}$ in (3.16) with $d_j^{\pm} = \frac{j(1\pm\tan(\theta)i)}{15}$ and four choices of θ. *Right* for each θ, the measured error of the Parareal-Euler algorithm for the first 200 iterations

the algorithm switches from being a convergent to being a divergent algorithm.) Then, by choosing $J = 128$, $\Delta T = 1$ and $t \in [0, 300]$, we show in Fig. 7 on the right the error between the iterate and the target solution for the first 200 iterations. (The initial iterate is chosen randomly.) We see that, for $\theta = 1.36$ which leads to $\widetilde{\rho} = 0.9868 < 1$, although the error $\max_n \|\mathbf{U}_n^k - \mathbf{U}_n\|_\infty$ oscillates with respect to the iteration index k, the global diminishing tendency is apparent. For $\theta = 1.38$, which leads to an upper bound $\widetilde{\rho} = 1.0279$ of the convergence factor, the error presents stagnating behavior, neither diminishing nor increasing. For the other two choices: $\theta = 1.39$ and $\theta = 1.40$, which respectively lead to $\widetilde{\rho} = 1.0496$ and $\widetilde{\rho} = 1.0721$, we can see clearly that the error increases as the algorithm proceeds. All of these indicate that the quantity $\widetilde{\rho}$ defined by Theorem 3.1 is a sharp estimate for the convergence

factor of the Parareal-Euler algorithm and can be used to precisely predict the convergence behavior.

Remark 3.2 (How to choose an ideal mesh ratio J?) We now analyze how to choose J to maximize the gain of speedup by using the parareal algorithm. We assume that the length of the interval, i.e., T, and the fine step size Δt are fixed quantities. Let the computational time for moving forward one step of the Backward-Euler method be $\mathbb{T}_{\text{unit}}$. Then, the total computational time of the sequential computation using the Backward-Euler method is $\mathbf{T}_{\text{seq}} = J \times \mathbb{N} \times \mathbb{T}_{\text{unit}} = \mathbf{N}\mathbb{T}_{\text{unit}}$, where $\mathbb{N} = \frac{T}{\Delta T}$ is the number of the paralleled coarse time-intervals and $\mathbf{N} = J\mathbb{N} = T/\Delta t$ is the number of the total fine time grids. The computational time for a single iteration of the parareal algorithm is $\mathbb{N} \times \mathbb{T}_{\text{unit}} + J\mathbb{T}_{\text{unit}} + \mathbb{T}_{\text{unit}} = (\mathbb{N} + 1 + J)\mathbb{T}_{\text{unit}}$ and the number of iterations required to achieve the prescribed tolerance ε is $k \approx \frac{\log \varepsilon}{\log\left(\max_{z\in\sigma(\Delta T\mathbf{A})} \mathscr{K}(z,J)\right)}$, where $\sigma(\Delta T\mathbf{A})$ denotes the spectrum of the matrix $\Delta T\mathbf{A}$ and $\mathscr{K}$ is the contraction factor of the Parareal-Euler algorithm defined by (2.5). Hence, assuming the initial iterate is chosen randomly, the total computational time of the parareal algorithm to achieve the prescribed tolerance is

$$\mathbf{T}_{\text{para}} := \frac{(\mathbb{N} + 1 + J)\mathbb{T}_{\text{unit}} \log \varepsilon}{\log\left(\max_{z\in\sigma(\Delta T\mathbf{A})} \mathscr{K}(z, J)\right)}.$$

Then, the gain of speedup by using the parareal algorithm can be expressed as

$$\begin{aligned}\mathbf{S}(J) := \frac{\mathbf{T}_{\text{seq}}}{\mathbf{T}_{\text{para}}} &= \frac{\mathbf{N}}{\frac{(\mathbb{N}+1+J)\log\varepsilon}{\log\left(\max_{z\in\sigma(\Delta T\mathbf{A})} \mathscr{K}(z,J)\right)}} \\ &= \frac{\mathbf{N}}{\mathbf{N}/J + 1 + J} \times \frac{\log\left(\max_{z\in\sigma(\Delta T\mathbf{A})} \mathscr{K}(z, J)\right)}{\log\varepsilon}. \end{aligned} \tag{3.17}$$

For $\sigma(\mathbf{A}) \subset \mathbf{D}(\theta)$, we have $\sigma(\Delta T\mathbf{A}) \subset \mathbf{D}(\theta)$. Therefore, by using Theorem 3.1 we have

$$\mathbf{S}(J) \approx \widetilde{\mathbf{S}}(J) := \frac{\mathbf{N}}{\mathbf{N}/J + 1 + J} \times \frac{\log \widetilde{\rho}(\theta)}{\log \varepsilon} \Rightarrow J_{\text{opt}} = \sqrt{\mathbf{N}}. \tag{3.18}$$

In Fig. 8, we show $\widetilde{\mathbf{S}}$ as a function of the mesh ratio J. As expected, a larger θ corresponds to a smaller speedup factor, since $\widetilde{\rho}(\theta)$ is an increasing function of θ as shown in Fig. 5. We see that a maximal speedup of about 4.2 can be reached for an angle of 0.2, and when the angle θ is 1.2, the maximal speedup is one, both reached for a mesh ratio of about 200. This mean that no speedup is possible with this algorithm, no matter how many processors are used, when the angle θ is 1.2 or larger.

Remark 3.3 (application to time-periodic problems) For problem (1.1) with time-periodic condition $\mathbf{U}(0) = \mathbf{U}(T)$, Gander et al. [12] proposed the so-called PP-PC (periodic parareal with periodic coarse problem) and PP-IC (periodic parareal with initial value coarse problem) algorithms. Here, we are interested in PP-PC, because in general it converges faster than PP-IC. Similar to (2.3), the PP-PC algorithm can be written compactly as

$$\begin{cases} u_0^{k+1} = \mathscr{G}\left(T_{\mathbb{N}-1}, u_{\mathbb{N}-1}^{k+1}, \Delta T\right) + \mathscr{F}^J\left(T_{\mathbb{N}-1}, u_{\mathbb{N}-1}^{k}, \Delta t\right) - \mathscr{G}\left(T_{\mathbb{N}-1}, u_{\mathbb{N}-1}^{k}, \Delta T\right), \\ u_{n+1}^{k+1} = \mathscr{G}\left(T_n, u_n^{k+1}, \Delta T\right) + \mathscr{F}^J\left(T_n, u_n^{k}, \Delta t\right) - \mathscr{G}\left(T_n, u_n^{k}, \Delta T\right), \end{cases} \tag{3.19}$$

where $n = 0, 1, \ldots, \mathbb{N} - 2$. In (3.19), the fine propagator is applied to nonperiodic problems and the time periodicity on the coarse time grids will be recovered upon convergence. Then,

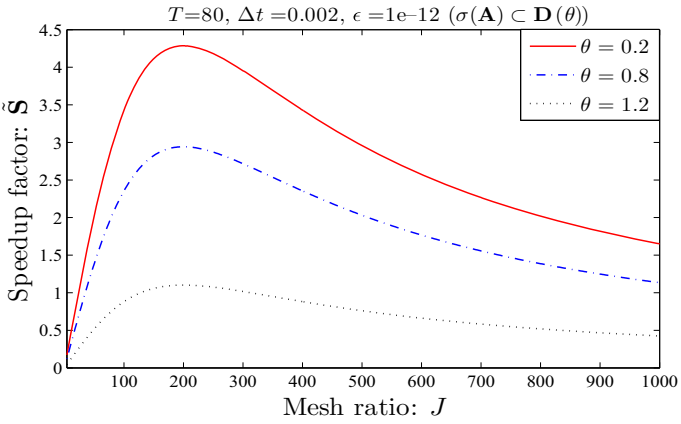

Fig. 8 The speedup factor $\widetilde{\mathbf{S}}$ (defined by (3.18)) as a function of J

if $\mathscr{G} = \mathscr{F}$=Backward-Euler, following the analysis in [12], the errors $\{\mathbf{e}_n^k\}$ and $\{\mathbf{e}_n^{k+1}\}$ also satisfy (2.4)-(2.5). Hence, all the statements for the initial-value problems are also applicable to the PP-PC algorithm (3.19).

Remark 3.4 (Other choices of the $\mathscr{F}$-propagator) At the end of this section, we show that the results that we analyzed for the Parareal-Euler algorithm also hold for some other choices of the $\mathscr{F}$-propagator. To validate this, we consider four time-integrators for the $\mathscr{F}$-propagator, the 2nd-order SDIRK (singly diagonally implicit Runge-Kutta) method, the TR/BDF2 method (i.e. the `ode23tb` solver in Matlab, a 2nd-order method), the 3rd-order Radau IIA method and the 4th-order Lobatto IIIC method. The $\mathscr{G}$-propagator is fixed to the Backward-Euler method. The resulting four parareal algorithms are denoted by Parareal-SDIRK2, Parareal-TR/BDF2, Parareal-RIIA3 and Parareal-LIIIC4, with contraction factors $\mathscr{K}_{\text{SDIRK2}}$, $\mathscr{K}_{\text{TR/BDF2}}$, $\mathscr{K}_{\text{RIIA3}}$ and $\mathscr{K}_{\text{LIIIC4}}$. The definitions of these four contraction factors are similar to that of the Parareal-Euler method given by (2.5), i.e.,

$$\mathscr{K}_{\text{SDIRK2,TR/BDF2,RIIA3,LIIIC4}}(z, J) = \frac{\left|\mathscr{R}^J_{\text{SDIRK2,TR/BDF2,RIIA3,LIIIC4}}\left(\frac{z}{J}\right) - \frac{1}{1+z}\right|}{1 - \left|\frac{1}{1+z}\right|},$$

where $\mathscr{R}_{\text{SDIRK2,TR/BDF2,RIIA3,LIIIC4}}$ denote the stability functions of the four time-integrators. For any integer $J \geq 2$, by using the maximal principle of analytic functions and the symmetry of the contraction factors with respect to the real axis, we have

$$\begin{aligned}&\max_{z\in\mathbf{D}(\theta)} \mathscr{K}_{\text{SDIRK2,TR/BDF2,RIIA3,LIIIC4}}(z, J)\\ &\quad= \max_{z=y(1/\tan(\theta)+i),\, y\geq 0} \mathscr{K}_{\text{SDIRK2,TR/BDF2,RIIA3,LIIIC4}}(z, J),\end{aligned}$$

where $\mathbf{D}(\theta)$ denotes the sectorial region as shown in Fig. 1. Then, we define

$$\begin{aligned}&\widetilde{\rho}^*_{\text{SDIRK2,TR/BDF2,RIIA3,LIIIC4}}(\theta)\\ &:= \max_{2\leq J\leq 10^6} \max_{z=y(1/\tan(\theta)+i),\, y\geq 0} \mathscr{K}_{\text{SDIRK2,TR/BDF2,RIIA3,LIIIC4}}(z, J). \qquad (3.20)\end{aligned}$$

(For each $J \in [2, 10^6]$ the maximum $\max_{z=y(1/\tan(\theta)+i),\, y\geq 0} \mathscr{K}_{\text{SDIRK2,TR/BDF2,RIIA3,LIIIC4}}(z, J)$ is obtained by using the chebfun command '`max`' in Matlab.) The profile of

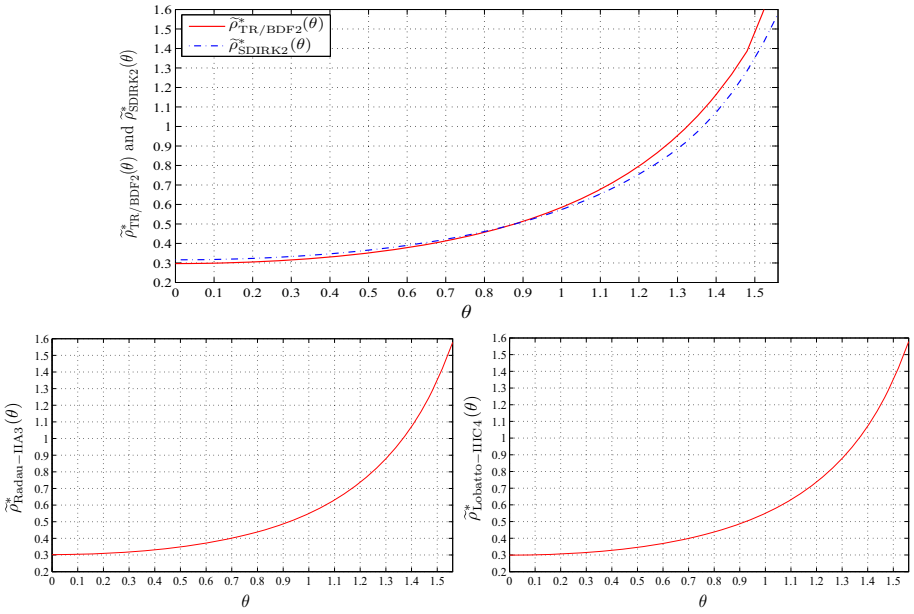

Fig. 9 The profile of $\widetilde{\rho}^*_{\mathrm{SDIRK2,TR/BDF2,RIIA3,LIIIC4}}(\theta)$ defined by (3.20). *Top*: $\mathscr{F}$ =TR/BDF2 and $\mathscr{F}$ = 2nd-order SDIRK; *Bottom left* $\mathscr{F}$ = 3rd-order Radau IIA; *Bottom right* $\mathscr{F}$=4th-order Lobatto IIIC. The $\mathscr{G}$-propagator is fixed to the Backward-Euler method

$\widetilde{\rho}^*_{\mathrm{SDIRK2,TR/BDF2,RIIA3,LIIIC4}}(\theta)$ is shown in Fig. 9, where we can see very similar results as shown in Fig. 5 for the Parareal-Euler algorithm. At the moment, sharp estimates depending on θ only are not available for the contractions factors $\mathscr{K}_{\mathrm{SDIRK2,TR/BDF2,RIIA3,LIIIC4}}$, because the analysis for the Parareal-Euler algorithm can not be simply generalized to these four parareal algorithms.

4 Numerical Results

For the case that the ODEs system (1.1) arises from semi-discretizing a time-dependent PDE defined on specified spatial domain, the spectrum $\sigma(\mathbf{A})$ is determined by the spatial discretization. On the one hand, the spatial discretization should reach the anticipated order of accuracy. On the other hand, if we attempt to solve the resulting ODEs system in the parallel-in-time pattern by the Parareal-Euler algorithm, the spatial discretization should

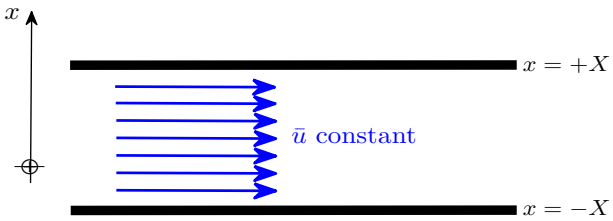

Fig. 10 Illustration of a simple pipe flow

lead to a coefficient matrix with amplitude angle as small as possible. As we will show in the following, it is possible to realize these two goals concurrently.

As an illustration, we consider here the one-dimensional kinetic equation describing the dispersion of particles in inhomogeneous turbulence. Precisely, for an axisymmetric pipe in which the mean carrier flow $\bar{u}$ is uniform and axial (see Fig. 10) and in which the distribution of particles exhibits no spatial or velocity gradients in the axial direction, the probability density function $W = W(x, v, t)$ modeling the probability of occurrence of particles with a given velocity is governed by the so-called Fokker-Planck equation:

$$\begin{aligned}&\frac{\partial W(x,v,t)}{\partial t} + v\frac{\partial W(x,v,t)}{\partial x} - \beta\frac{\partial (vW(x,v,t))}{\partial v}\\ &\quad + F(x)\frac{\partial W(x,v,t)}{\partial v} - \beta\mu\frac{\partial^2 W(x,v,t)}{\partial v^2} = 0,\end{aligned} \tag{4.1a}$$

where $v \in \mathbf{R}$, $x \in [-X, X]$ and $\beta, \mu > 0$ are constants. To make a well-posed problem, we consider the case that there are absorbing walls at the bottom- and top-hand sides of the spatial domain. In this case, we supplement (4.1a) with the following initial and boundary conditions:

$$\begin{cases} W(-X, v, t) = b_l(v, t), & v \geq 0, \quad t \geq 0,\\ W(X, v, t) = b_r(v, t), & v \leq 0, \quad t \geq 0,\\ W(x, v, 0) = W_0(x, v), & x \in [-X, X], \quad v \in \mathbf{R}, \end{cases} \tag{4.1b}$$

where $b_{l,r}(v, t)$ and $W_0(x, v)$ are known functions. (For more details about the absorbing boundary conditions for the Fokker-Planck equation, the interested reader can refer to [10, Section 1].)

A powerful numerical treatment for (4.1a)–(4.1b) is to use the hybrid Hermite spectral-finite difference method established by Tang et al. [10,30], i.e., using the Hermite spectral approximation in the velocity direction and then using finite-difference in the x-direction. One of the main advantages of this strategy is that it has correct natural boundary conditions in the velocity space. For completeness, we briefly introduce the formula given in [10,30]. Let $H_n(v)$ and $\widetilde{H}_n(v)$ be the n-th Hermite polynomial and the generalized Hermite function:

$$\begin{aligned} H_n(v) &= (-1)^n e^{v^2}\frac{d^n}{dv^n}\left(e^{-v^2}\right),\\ \widetilde{H}_n(v) &= d_n H_n(\alpha v) e^{-\alpha^2 v^2} \ (\text{with } d_n = (2^n n!)^{-1/2} \text{ and } \alpha > 0), \end{aligned} \tag{4.2}$$

where $n \geq 0$, α is a free parameter and is called 'scaling factor'. As we will see a little later, this scaling factor α plays an important role in several aspects. It is well-known that $\{\widetilde{H}_n(v)\}$ forms the $L^2_\omega(\mathbf{R})$-orthogonal system with weighted function $\omega(v) = e^{\alpha^2 v^2}$, namely

$$\int_{\mathbf{R}} \widetilde{H}_m(v)\widetilde{H}_n(v)\omega(v)dv = \frac{\sqrt{\pi}}{\alpha}\delta_{m,n}, \tag{4.3}$$

where $\delta_{m,n}$ is the Kronecker delta. Hence, for $W \in L^2_\omega(\mathbf{R})$ we can expand it through the Hermite functions as $W(x, v, t) = \sum_{n=0}^{+\infty} \widehat{W}_n(x, t)\widetilde{H}_n(v)$, where $\left\{\widehat{W}_n(x, t)\right\}$ is our target to be computed. The truncated expansion reads

$$W_N(x, v, t) = \sum_{n=0}^{N} \widehat{W}_n(x, t)\widetilde{H}_n(v). \tag{4.4}$$

By replacing $W(x, v, t)$ by $W_N(x, v, t)$ in (4.1a) and (4.1b), and by using the recurrence relations and the spectral collocation idea concerning the generalized Hermite function (see

[10,29,30]), we arrive at the following finite system of hyperbolic equations:

$$\begin{cases} \partial_t \mathbf{u}(x,t) + \frac{1}{\alpha}\left[\Lambda_- \partial_x \mathbf{u}(x,t) + \Lambda_+ \partial_x \mathbf{u}(x,t)\right] = \mathbf{B}(x)\mathbf{u}(x,t), & x \in [-X, X], t \geq 0, \\ \mathbf{D}_2 \mathbf{u}(-X,t) = \mathbf{V}_l(t), & t \geq 0, \\ \mathbf{D}_1 \mathbf{u}(X,t) = \mathbf{V}_r(t), & t \geq 0, \\ \mathbf{u}(x,0) = \mathbf{u}_0(x), & x \in [-X, X], \end{cases} \tag{4.5}$$

where $\mathbf{u}(x,t) = \mathbf{P}^\top \mathbf{W}(x,t)$, $\mathbf{u}_0(t) = \mathbf{P}^\top \mathbf{W}_0(t)$, $\mathbf{B}(x) = \mathbf{P}^\top \mathbf{S}(x)\mathbf{P}$ with $\mathbf{W}(x,t) = \left[\widehat{W}_0(x,t), \ldots, \widehat{W}_N(x,t)\right]^\top$ and

$$\begin{aligned} \mathbf{p}_k &= [p_{0,k}, \ldots, p_{N,k}]^\top \text{ (with } p_{n,k} = c_k d_n H_n(\lambda_k)), \\ \mathbf{P} &= [\mathbf{p}_0, \mathbf{p}_1, \ldots, \mathbf{p}_N], \ \Lambda_- = \operatorname{diag}\left(\lambda_0, \cdots, \lambda_{N_1}, 0, \cdots, 0\right), \\ \Lambda_+ &= \operatorname{diag}\left(0, \cdots, 0, \lambda_{N_1+1}, \cdots, \lambda_N\right), \mathbf{S}(x) = \begin{pmatrix} s_0 & & & & \\ \gamma_1(x) & s_1 & & & \\ \delta_2 & \gamma_2(x) & s_2 & & \\ & \ddots & \ddots & \ddots & \\ & & \delta_N & \gamma_N(x) & s_N \end{pmatrix}, \\ \sigma_n &= \sqrt{\frac{n}{2}}, \ s_n = -\beta n, \ \ \gamma_n(x) = \alpha\sqrt{2n}F(x), \ \ \delta_n = \beta(2\alpha^2\mu - 1)\sqrt{n(n-1)}, \\ \mathbf{V}_l(t) &= \left[0, \cdots, 0, c_{N_1+1} b_l(\lambda_{N_1+1}/\alpha, t) e^{\lambda_{N_1+1}^2}, \cdots, c_N b_l(\lambda_N/\alpha, t) e^{\lambda_N^2}\right]^\top, \\ \mathbf{V}_r(t) &= \left[c_0 b_l(\lambda_0/\alpha, t) e^{\lambda_0^2}, \cdots, c_{N_1} b_l(\lambda_{N_1}/\alpha, t) e^{\lambda_{N_1}^2}, 0, \cdots, 0\right]^\top, \\ \mathbf{W}_0(x) &= \left[\widehat{W}_{0,0}(x), \ldots, \widehat{W}_{0,N}(x)\right]^\top, \ \mathbf{D}_1 = \operatorname{diag}\left(\underbrace{1, \ldots, 1}_{N_1+1}, 0, \ldots, 0\right), \\ \mathbf{D}_2 &= \operatorname{diag}\left(0, \ldots, 0, \underbrace{1, \ldots, 1}_{N_1+1}\right). \end{aligned} \tag{4.6}$$

For simplicity, we assume that N is an odd integer. In (4.6), $N_1 = \frac{N-1}{2}$ and the constants $\{c_k\}_{k=0}^N$ are defined by

$$c_k = \left(\sum_{n=0}^{N} d_n^2 H_n^2(\lambda_k)\right)^{-\frac{1}{2}},$$

where $\{d_n\}_{n=0}^N$ are given in (4.2) and the quantities $\{\lambda_k\}_{k=0}^N$ are the zeros of the Hermite polynomial $H_{N+1}(v)$ and it is well-known that they are distinct real numbers and satisfy

$$\lambda_0 < \lambda_1 < \cdots < \lambda_{N-1} < \lambda_N, \ \lambda_n = -\lambda_{N-n} \ (n = 0, 1, \ldots, N). \tag{4.7}$$

From (4.7), it is easy to see $\lambda_n < 0$ for $n \in \{0, 1, \ldots, N_1\}$ and $\lambda_n > 0$ for $n \in \{N_1 + 1, N_1 + 2, \ldots, N\}$. This implies that the diagonal matrices Λ_- and Λ_+ defined in (4.6) are respectively negative and positive. Therefore, the following up-winding discretization in the

x-direction is reasonable for the coupled hyperbolic system (4.5):

$$\begin{cases} \partial_t \mathbf{U}_j + \frac{1}{\alpha}\left[\Lambda_- \frac{\mathbf{U}_{j+1}-\mathbf{U}_j}{\Delta x} + \Lambda_+ \frac{\mathbf{U}_j-\mathbf{U}_{j-1}}{\Delta x}\right] = \mathbf{B}_j \mathbf{U}_j, & j \in \{-M, \cdots, 0, \cdots, M\}, t \geq 0, \\ \mathbf{D}_2 \mathbf{U}_{-M} = \mathbf{V}_l(t), & t \geq 0, \\ \mathbf{D}_1 \mathbf{U}_M = \mathbf{V}_r(t), & t \geq 0, \\ \mathbf{U}_j(0) = \mathbf{u}_0(x_j), & j \in \{-M, \cdots, 0, \cdots, M\}, \end{cases} \tag{4.8}$$

where $\Delta x = \frac{X}{M}$, $\mathbf{U}_j(t) \approx \mathbf{u}(x_j, t)$, $\mathbf{B}_j = \mathbf{B}(x_j)$, $x_j = j\Delta x$. Let $\mathbf{g}(t) = \frac{1}{\alpha \Delta x}\left[\Lambda_+ \mathbf{V}_l^\top, 0, \cdots, 0, -\Lambda_- \mathbf{V}_r^\top\right]^\top$ and $\mathbf{U}(t) = \left[\mathbf{U}_{-M+1}^\top(t), \cdots, \mathbf{U}_{M-1}^\top(t)\right]^\top$. Then, by writing (4.8) compactly, we arrive at

$$\begin{cases} \mathbf{U}'(t) + \mathbf{A}\mathbf{U}(t) = \mathbf{g}(t), \\ \mathbf{A} = \frac{1}{\alpha \Delta x} \begin{bmatrix} \tilde{\mathbf{B}}_{-M+1} & \Lambda_- & & & \\ -\Lambda_+ & \tilde{\mathbf{B}}_{-M+2} & \Lambda_- & & \\ & \ddots & \ddots & \ddots & \\ & & -\Lambda_+ & \tilde{\mathbf{B}}_{M-2} & \Lambda_- \\ & & & -\Lambda_+ & \tilde{\mathbf{B}}_{M-1} \ \Lambda_- \end{bmatrix} \text{ (with } \tilde{\mathbf{B}}_j = \Lambda_+ - \Lambda_- - \alpha \Delta x \mathbf{B}_j). \end{cases} \tag{4.9}$$

This is the semi-discrete version of the Fokker-Planck equation and for more details of its derivation the interested reader can refer to [10, Sections 2 and 3]. After computing $\mathbf{U}(t)$, using (4.6) we get $\mathbf{W}(x_j, t) = \mathbf{P}\mathbf{u}(x_j, t)$, i.e., we get the coefficients $\{\widehat{W}_n(x_j, t)\}_{n=0}^N$ in the Hermite spectral approximation (4.4).

The accuracy of $W_N(x_j, v, t)$ is determined by two parts: the discretization in the x-direction by the up-winding scheme and the discretization in v-direction by the Hermite spectral approximation. The accuracy of the discretization in the x-direction is well-understood. For the accuracy of the spectral approximation in the v-direction, it is known long that the scaling factor α appearing in the Hermite function $\widetilde{H}_n(v)$ plays an important role [15,28,29]. For $\alpha = 1$, i.e., using the classical Hermite functions as the basis functions, it is reported by Gottlieb and Orszag [15] that the resolution quality of the Hermite spectral approximation is very poor and a huge number of Hermite functions are needed to reach moderate accuracy. (This implies that N will be a large integer and thus the size of the coefficient matrix $\mathbf{A}$ in (4.9) will be huge.) By using a carefully selected scaling factor, only reasonable numbers of Hermite functions are needed. How to choose a clever scaling factor α to make the accuracy of the spectral approximation optimal is still an open problem for general problem setting. The choice suggested by Tang et al. [10,29,30] depends on some apriori knowledge about the analytic solution. Particularly, for problem (4.1a) with $X = 1$ and $F(x) = 0$, if we choose initial and boundary functions in (4.1b) as

$$\begin{cases} W_0(x, v) = [1 + \sin(\tau(v - \beta x))]\, e^{-\frac{v^2}{2\mu}}, \\ b_l(v, t) = [1 + \sin(\tau(v + \beta))]\, e^{-\frac{v^2}{2\mu} - \beta\mu\tau^2 t}, \\ b_r(v, t) = [1 + \sin(\tau(v - \beta))]\, e^{-\frac{v^2}{2\mu} - \beta\mu\tau^2 t}, \end{cases} \quad (\tau \neq 0 \text{ is a constant}) \tag{4.10}$$

it can be verified that the exact solution is

$$W(x, v, t) = \bar{h}(x, v, t) e^{-\frac{v^2}{2\mu}} \quad \left(\text{with } \bar{h}(x, v, t) = [1 + \sin(\tau(v - \beta x))]\, e^{-\beta\mu\tau^2 t}\right). \tag{4.11}$$

Then, the analysis given by Tang et al. [10,29,30] shows that the best scaling factor is

$$\alpha_{\text{opt}} \approx \sqrt{\frac{1}{2\mu}}. \tag{4.12}$$

Hereafter, we let $X = 1$ and $F(x) = 0$. Then, as we will show in what follows, it is very interesting that the best choice of the scaling factor α, which makes the amplitude angle of the matrix $\mathbf{A}$ in (4.9) minimal, is also around $\sqrt{1/(2\mu)}$! To fix the idea, we choose $\beta = 0.7$ and two values for μ: $\mu = 5$ and $\mu = 2$. Then, from (4.12) we know that the best scaling factors minimizing the global error of the Hermite spectral approximation are around $\sqrt{0.1}(\approx 0.316)$ and 0.5, respectively. Now, for given N and Δx, i.e., for specified matrix $\mathbf{A}$, we use the command '`eig`' in Matlab to get $\sigma(\mathbf{A})$. Then, we can calculate the amplitude angle of $\mathbf{A}$ by $\theta = \arctan\left(\max_{\lambda\in\sigma(\mathbf{A})} |\Im(\lambda)/\Re(\lambda)|\right)$. In Fig. 11 on the left, we show θ as a function of α for three choices of N, the number of the Hermite functions used in the spectral approximation. On the right column, we show $\sigma(\mathbf{A})$ under the numerically optimized scaling factor implied by the left sub-figure (each right sub-figure contains two panels, $\mu = 5$ on the top and $\mu = 2$ on the bottom). Similar result, when N is fixed ($N = 11$) and Δx varies, is shown in Fig. 12.

From Figs. 11 and 12, we see that: the best scaling factor minimizing the amplitude angle of the matrix $\mathbf{A}$ is also around $\sqrt{1/(2\mu)}$ and such a best factor is insensitive to the change of N and Δx. (We have tested many other values of the problem parameters μ and β and this conclusion always holds.) *This information is inspiring, since it implies that: for given N and Δx, the optimal scaling factor which minimizes the global error of the Hermite spectral approximation can concurrently realize another goal, minimizing convergence factor of the Parareal-Euler algorithm*. If, in some situation, these two goals conflict with each other, we suggest sacrificing the latter, because, on the one hand the accuracy of the Hermite spectral approximation heavily depends on the scaling factor [10,28–30], while on the other hand, as we have stated in Remark 3.1, the influence of the amplitude angle θ of the coefficient matrix $\mathbf{A}$ on the convergence factor of the Parareal-Euler algorithm is negligible for $\theta \in [0, 0.7]$. Of course, this suggestion could be misleading when $\theta > 0.7$; a precise analysis for this case is extremely complicated and beyond the scope of this paper.

All the above discussions only concern the influence of spatial discretization on the amplitude angle of the derived matrix $\mathbf{A}$. For initial value problem, these are instructive for the Parareal-Euler algorithm, since the theoretical results of the convergence properties predict numerical experiments very well (see Fig. 7 and Remark 3.1). In the remainder of this section, we present numerical evidences to show that these predictions are also applicable to the time-periodic problems. We consider the semi-discrete Fokker-Planck equations (4.9) with condition $\mathbf{U}(0) = \mathbf{U}(T)$. Choosing $\beta = 0.7$, $\mu = 5$ for the problem parameters and $\Delta x = \frac{1}{20}$, $N = 11$, $\Delta T = 0.1$, $J = 25$ for the discretization parameters, then for 13 values of the scaling factor α we show in Fig. 13 the amplitude angle θ of the matrix $\mathbf{A}$ in (4.9). We see that, for $\alpha \leq 0.5$ the angle θ is less than 0.7 and thus the influence of θ on the convergence rate of the Parareal-Euler algorithm is negligible (see Remark 3.1).

We now apply the Parareal-Euler algorithm to the time-periodic Fokker-Planck equation with the same boundary conditions given by (4.10). The initial iterate is chosen randomly and the iteration process stops when the error between the iterate and the target solution satisfies $\max_n \left\|\mathbf{U}_n^k - \mathbf{U}_n\right\|_\infty \leq 10^{-12}$. Let $\mathbb{N} = \frac{T}{\Delta T}$, $\mathbf{G} = -(I + \Delta T\mathbf{A})^{-1}$ and $\mathscr{J}_n^k = \mathscr{F}^J\left(T_n, u_n^k, \Delta t\right) - \mathscr{G}\left(T_n, u_n^k, \Delta T\right) - \Delta T\mathbf{G}\mathbf{g}(T_n)$. Then, for (4.9) with time-periodic condition, the Parareal-Euler algorithm can be written compactly as

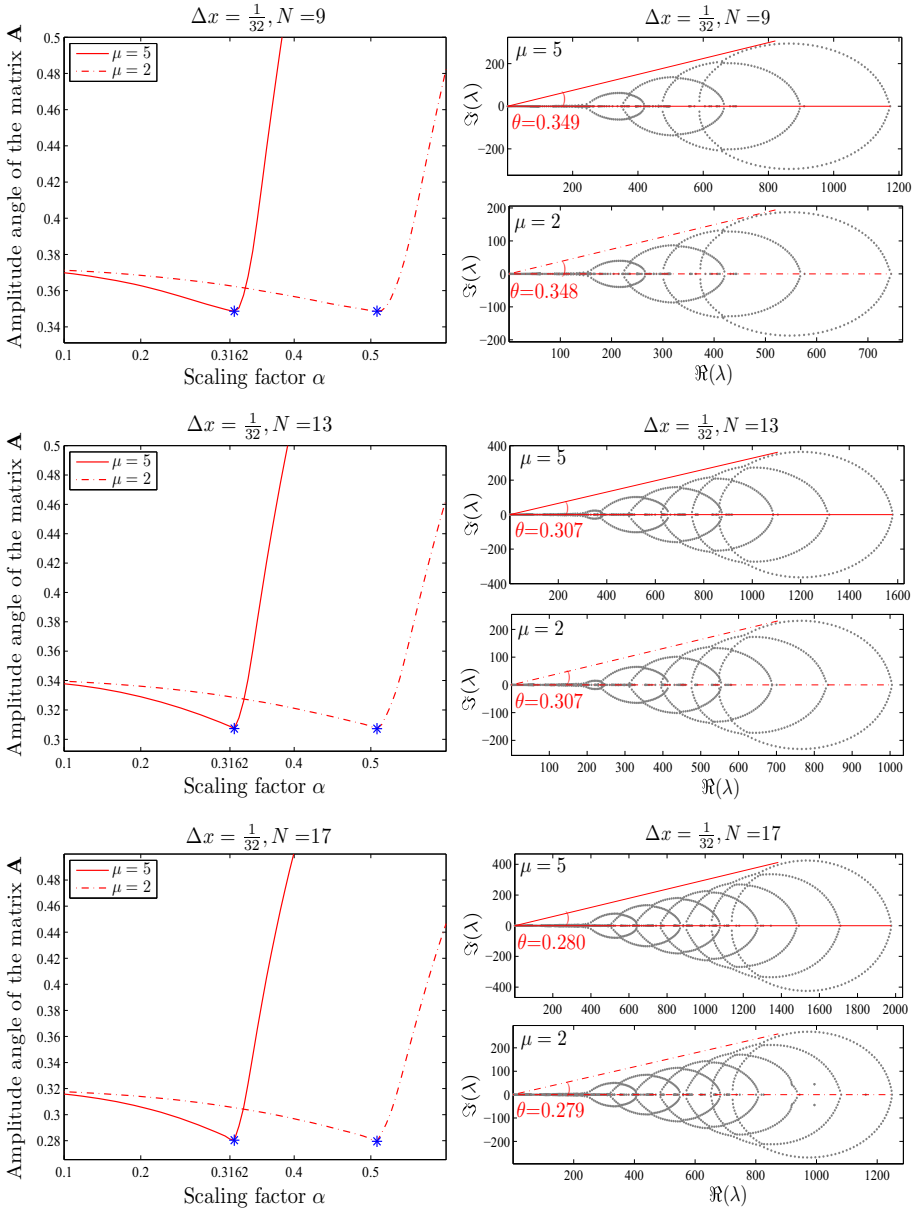

Fig. 11 *Left* the amplitude angle of the matrix **A** as a function of the scaling factor α; *Right* the distribution of the spectrum of **A** under the numerically optimized α. From *top to bottom* $N = 9, 13, 17$

$$\underbrace{\begin{bmatrix} I & & \mathbf{G} \\ \mathbf{G} & I & \\ & \ddots & \ddots \\ & & \mathbf{G} \quad I \end{bmatrix}}_{\mathbf{M}} \underbrace{\begin{bmatrix} \mathbf{U}_0^{k+1} \\ \mathbf{U}_1^{k+1} \\ \vdots \\ \mathbf{U}_{\mathbb{N}-1}^{k+1} \end{bmatrix}}_{\mathbf{X}} = \underbrace{\begin{bmatrix} \mathscr{J}_{\mathbb{N}}^{k} \\ \mathscr{J}_{1}^{k} \\ \vdots \\ \mathscr{J}_{\mathbb{N}-1}^{k} \end{bmatrix}}_{\mathbf{b}}. \tag{4.13}$$

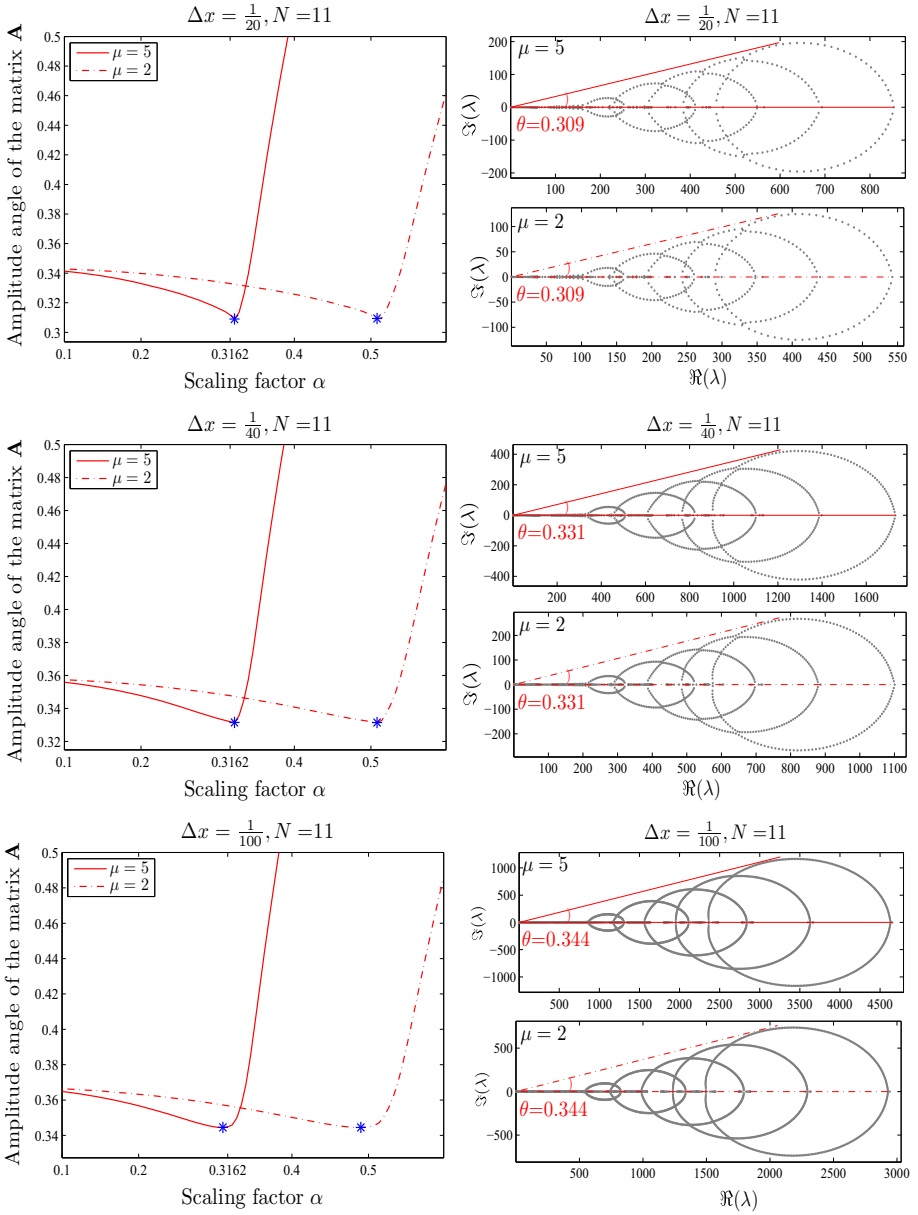

Fig. 12 *Left* the amplitude angle of the matrix $\mathbf{A}$ as a function of the scaling factor α; *Right* the distribution of the spectrum of $\mathbf{A}$ under the numerically optimized α. From *top to bottom* $\Delta x = \frac{1}{20}, \frac{1}{40}, \frac{1}{100}$

To further specify the problem setting, we choose $x \in [-1, 1]$, $N = 11$ and $\Delta x = 0.05$, which leads to $\mathbf{A} \in \mathbf{R}^{468\times 468}$. Then, we choose $T = 5$ and $\Delta T = 0.1$ to test the Parareal-Euler algorithm (i.e., the PP-PC algorithm defined by (3.19) with $\mathscr{G} = \mathscr{F}$=Backward-Euler). This choice leads to $\mathbb{N} = \frac{T}{\Delta T} = 50$ and therefore $\mathbf{M} \in \mathbf{R}^{23400\times 23400}$, where $\mathbf{M}$ is the coefficient matrix defined by (4.13). For each parareal iteration, the large scale linear system

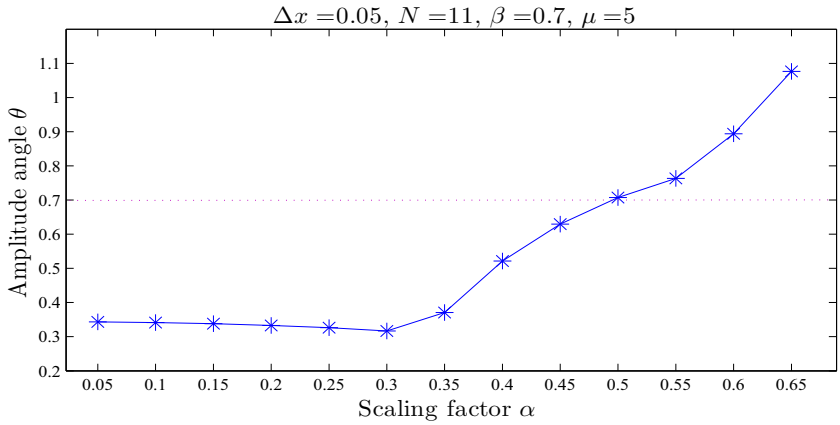

Fig. 13 The amplitude angles θ of the matrix $\mathbf{A}$ in (4.9) for 13 scaling factors α

(4.13): $\mathbf{MX} = \mathbf{b}$, is solved iteratively as

$$\mathbf{M}_1\mathbf{X}_{m+1} = -\mathbf{M}_2\mathbf{X}_m + \mathbf{b},\ m = 0, 1, \ldots, \tag{4.14}$$

where $\mathbf{M}_1 := \begin{bmatrix} I & & & \\ \mathbf{G} & I & & \\ & \ddots & \ddots & \\ & & \mathbf{G} & I \end{bmatrix}$ and $\mathbf{M}_2 := \mathbf{M} - \mathbf{M}_1$. In (4.14), the initial guess $\mathbf{X}_0$ is chosen as $\mathbf{X}_0 = \begin{bmatrix} \mathbf{U}_0^k \\ \mathbf{U}_1^k \\ \vdots \\ \mathbf{U}_{\mathbb{N}-1}^k \end{bmatrix}$ and the iteration proceeds until $\|\mathbf{X}_{m+1} - \mathbf{X}_m\|_\infty \leq 10^{-12}$. In the top panel of Fig. 14, we show the iteration number for the 13 values of α tested in Fig. 13. In the bottom panel, we show iteration number for 6 values of the mesh ratio J, where we considered two problem parameters $\beta = 10^{-3}$ and $\beta = 0.7$. We see that, all the conclusions for the initial-value problems also hold for the time-periodic problems.

5 Conclusions

We have analyzed the convergence properties of the Parareal-Euler algorithm, which is defined by choosing the Backward-Euler method as both the coarse and fine propagators, for linear ODEs system $\mathbf{U}'(t) + \mathbf{AU}(t) = \mathbf{g}$, when the spectrum $\sigma(\mathbf{A})$ is contained in a sectorial region with opening angle $\theta \in (0, \frac{\pi}{2})$ to the positive real axis. We derived upper bound for the convergence factor of the algorithm, namely $\widetilde{\rho}$, which is independent of the mesh ratio J and depends on the angle θ only. Numerical results show that this bound is sharp and can be used to precisely predict the convergence behavior of the Parareal-Euler algorithm. The profile of $\widetilde{\rho}$ (as a function of θ) shown in Fig. 5 implies that: the Parareal-Euler algorithm possesses an ideal convergence factor for $\theta \leq 0.7$ and $\widetilde{\rho}$ grows slowly for $\theta \in [0, 0.7]$. A direct implication of this observation is that, for $\theta \in [0, 0.7]$ the influence of θ on the convergence rate is almost invisible in a concrete computation. This is perfectly validated

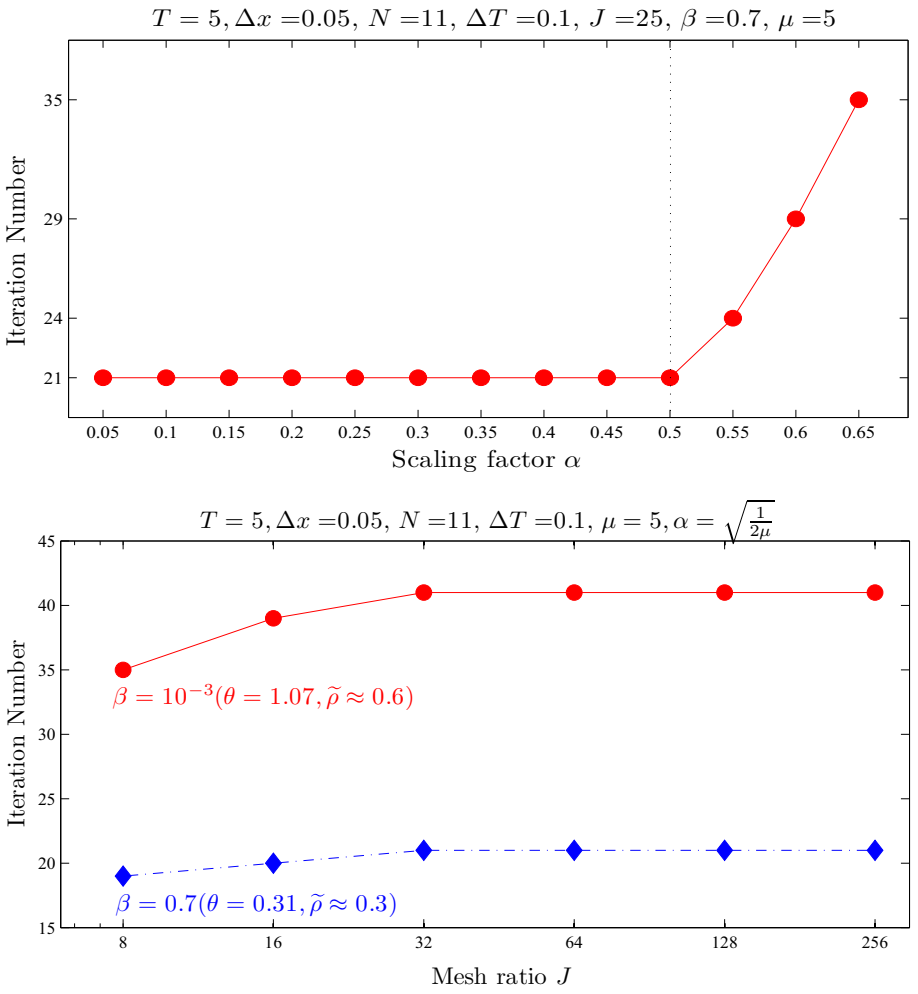

Fig. 14 Iteration number of the Parareal-Euler algorithm applied to the Fokker-Planck Eq. (4.9) with time-periodic condition $\mathbf{U}(0) = \mathbf{U}(T)$. *Top* iteration number for 13 values of the scaling factor α. *Bottom* for two values of the problem parameter β, iteration number for 6 values of the mesh ratio J

by numerical experiments for the Fokker-Planck equation semi-discretized by the hybrid Hermite spectral-finite difference method [10,30].

For ODEs systems arising from semi-discretizing time-dependent PDEs, to make the parareal algorithm converges rapidly, it is clear that the spatial discretization should result in an angle θ as small as possible. Spatial discretization should also make the truncation error as small as possible. We have illustrated that it is possible to realize these two goals concurrently. Precisely, for the just mentioned Fokker-Planck equation semi-discretized by the mixed Hermite spectral-finite difference method, it was shown that the best choice of the scaling factor α, which optimizes the resolution quality of the Hermite spectral approximation, is also the best choice for the Parareal-Euler algorithm, because it makes the amplitude angle θ of the coefficient matrix $\mathbf{A}$ minimal. Does this also hold in more general situation [e.g., other

form of the analytic solution $W(x, v, t)$ instead of the one given by (4.11)]? Does this also hold for other problems instead of the Fokker-Planck equation? These issues are challenging and are worth further investigation in the future. But one thing is clear: if the two goals, i.e., minimizing the amplitude angle θ and minimizing the truncation error, conflict with each other, the aforementioned property '*the influence of* θ *on the convergence rate of the Parareal-Euler algorithm is almost invisible for* $\theta \in [0, 0.7]$' can be used as a freedom.

Acknowledgments The authors are very grateful to the anonymous referees for the careful reading of a preliminary version of the manuscript and their valuable suggestions and comments, which greatly improved the quality of this paper. This work was supported by the NSF of China (11301362, 11371157, 91130003), the NSF of Technology & Education of Sichuan Province (2014JQ0035,15ZA0220), the project of key laboratory of bridge non-destruction detecting and computing (2013QZY01) and the NSF of SUSE (2015LX01).

References

1. Bal, G.: On the convergence and the stability of the parareal algorithm to solve partial differential equations. Lect. Notes Comput. Sci. Eng. **40**, 426–432 (2003)
2. Celledoni, E., Kvamsdal, T.: Parallelization in time for thermo-viscoplastic problems in extrusion of aluminium. Int. J. Numer. Methods Eng. **79**(5), 576–598 (2009)
3. Cortial, J., Farhat, C.: A time-parallel implicit method for accelerating the solution of nonlinear structural dynamics problems. Int. J. Numer. Methods Eng. **77**(4), 451–470 (2009)
4. Dai, X.Y., Maday, Y.: Stable parareal in time method for first- and second-order hyperbolic systems. SIAM J. Sci. Comput. **35**(1), A52–A78 (2013)
5. Dai, X.Y., Le Bris, C., Legoll, F., Maday, Y.: Symmetric parareal algorithms for Hamiltonian systems. M2AN Math. Model Numer. Anal. **47**(3), 717–742 (2013)
6. Du, X.H., Sarkis, M., Schaerer, C.F., Szyld, D.B.: Inexact and truncated parareal-in-time Krylov subspace methods for parabolic optimal control problems. Electron. Trans. Numer. Anal. **40**, 36–57 (2013)
7. Emmett, M., Minion, M.: Toward an efficient parallel in time method for partial differential equations. Commun. Appl. Math. Comput. Sci. **7**, 105–132 (2012)
8. Farhat, C., Cortial, J., Dastillung, C., Bavestrello, H.: Time-parallel implicit integrators for the near-real-time prediction of linear structural dynamic responses. Int. J. Numer. Methods Eng. **67**(5), 697–724 (2006)
9. Farhat, C., Chandesris, M.: Time-decomposed parallel time-integrators: theory and feasibility studies for fluid, structure, and fluid-structure applications. Int. J. Numer. Methods Eng. **58**(9), 1397–1434 (2003)
10. Fok, J.C.M., Guo, B.Y., Tang, T.: Combined Hermite spectral-finite difference method for the Fokker-Planck equation. Math. Comput. **71**(240), 1497–1528 (2002)
11. Gander, M.J., Vandewalle, S.: On the superlinear and linear convergence of the parareal algorithm. Lect. Notes Comput. Sci. Eng. **55**, 291–298 (2005)
12. Gander, M.J., Jiang, Y.L., Song, B., Zhang, H.: Analysis of two parareal algorithms for time-periodic problems. SIAM J. Sci. Comput. **35**(5), A2393–A2415 (2013)
13. Gander, M.J., Vandewalle, S.: Analysis of the parareal time-parallel time-integration method. SIAM J. Sci. Comput. **29**(2), 556–578 (2007)
14. Gander, M.J., Hairer, E.: Analysis for parareal algorithms applied to Hamiltonian differential equations. J. Comput. Appl. Math. **259**, 2–13 (2014)
15. Gottlieb, D., Orszag, S.: Numerical analysis of spectral methods: theory and applications. In: CBMS-NSF Regional Conference Series in Applied Mathematics, 26. SIAM , Philadelphia (1977)
16. He, L., He, M.: Parareal in time simulation of morphological transformation in cubic alloys with spatially dependent composition. Commun. Comput. Phys. **11**(5), 1697–1717 (2012)
17. Haut, T., Wingate, B.: An asymptotic parallel-in-time method for highly oscillatory PDEs. SIAM J. Sci. Comput. **36**(2), A693–A713 (2014)
18. Li, X.J., Tang, T., Xu, C.J.: Parallel in time algorithm with spectral-subdomain enhancement for Volterra integral equations. SIAM J. Numer. Anal. **51**(3), 1735–1756 (2013)
19. Legoll, F., Lelièvre, T., Samaey, G.: A micro-macro parareal algorithm: application to singularly perturbed ordinary differential equations. SIAM J. Sci. Comput. **35**(4), A1951–A1986 (2013)
20. Lions, J.L., Maday, Y., Turinici, G.: A "parareal" in time discretization of PDE's. C. R. Acad. Sci. Paris Sér. I Math. **332**(7), 661–668 (2001)

21. Maday, Y., Riahi, M.K., Salomon, J.: Parareal in time intermediate targets methods for optimal control problems. In: Control and Optimization with PDE Constraints, pp. 79–92. Springer, Basel (2013)
22. Maday, Y., Salomon, J., Turinici, G.: Monotonic parareal control for quantum systems. SIAM J. Numer. Anal. **45**(6), 2468–2482 (2007)
23. Mathew, T.R., Sarkis, M., Schaerer, C.E.: Analysis of block parareal preconditioners for parabolic optimal controal problems. SIAM J. Sci. Comput. **32**(3), 1180–1200 (2010)
24. Minion, M.L.: A hybrid parareal spectral deferred corrections method. Appl. Math. Comput. Sci. **5**(2), 265–301 (2010)
25. Reynolds-Barredo, J.M., Newman, D.E., Sanchez, R.: An analytic model for the convergence of turbulent simulations time-parallelized via the parareal algorithm. J. Comput. Phys. **255**(255), 293–315 (2013)
26. Reynolds-Barredo, J.M., Newman, D.E., Sanchez, R., Samaddar, D., Berry, L.A., Elwasif, W.R.: Mechanisms for the convergence of time-parallelized, parareal turbulent plasma simulations. J. Comput. Phys. **231**(23), 7851–7867 (2012)
27. Samaddar, D., Newman, D.E., Sánchez, R.: Parallelization in time of numerical simulations of fully-developed plasma turbulence using the parareal algorithm. J. Comput. Phys. **229**(18), 6558–6573 (2010)
28. Shen, J., Wang, L.L.: Review article: some recent advances on spectral methods for unbounded domains. Commun. Comput. Phys. **5**(2–4), 195–241 (2009)
29. Tang, T.: The Hermite spectral method for Gaussian-type functions. SIAM J. Sci. Comput. **14**, 594–606 (1993)
30. Tang, T., Mckee, S., Reeks, M.W.: A spectral method for the numerical solutions of a kinetic equation describing the dispersion of small particles in a turbulent flow. J. Comput. Phys. **103**, 222–230 (1991)
31. Wu, S.L.: Convergence analysis of some second-order parareal algorithms. IMA J. Numer. Anal. (2014). doi:10.1093/imanum/dru031
32. Wu, S.L., Shi, B.C., Huang, C.M.: Parareal-Richardson algorithm for solving nonlinear ODEs and PDEs. Commun. Comput. Phys. **6**(4), 883–902 (2009)