Journal of Computational Mathematics
Vol.xx, No.x, 2025, 1–33.

http://www.global-sci.org/jcm
doi:10.4208/jcm.2412-m2024-0049

ANALYSIS OF TWO NEW PARAREAL ALGORITHMS BASED ON DIAGONALIZATION FOR TIME-PERIODIC PROBLEMS*

Bo Song[1)] and Jing-Yi Wang
School of Mathematics and Statistics, Northwestern Polytechnical University, Xi'an 710129, China
Emails: bosong@nwpu.edu.cn, jingyiwang@mail.nwpu.edu.cn
Yao-Lin Jiang
School of Mathematics and Statistics, Xi'an Jiaotong University, Xi'an 710049, China
Email: yljiang@mail.xjtu.edu.cn

Abstract

Numerical simulation of time-periodic problems is a special area of research, since the time periodicity modifies the problem structure, and then it is desirable to use parallel methods to solve such problems. The classical parareal algorithm for time-periodic problems, which is parallel in time, solving an initial value coarse problem, called the periodic parareal algorithm with initial value coarse problem (PP-IC), usually converges very slowly, and even diverges for wave propagation problems. In this paper, we first present a new PP-IC algorithm based on a diagonalization technique proposed recently. In this new algorithm, we approximate the coarse propagator G in the classical PP-IC algorithm with a head-tail coupled condition such that G can be parallelized using diagonalization in time. We analyze the convergence factors of the diagonalization-based PP-IC algorithm for both the linear and nonlinear cases. Then, we further design and analyze a new parallel-in-time algorithm for time-periodic problems by combining the Krylov subspace method with the diagonalization-based PP-IC algorithm to accelerate the convergence. Finally, we also determine an appropriate choice of the parameter α in the head-tail coupling condition, and illustrate our theoretical results with several numerical experiments, both for model problems and the realistic application of Maxwell's equations.

Mathematics subject classification: 65L20, 65L70, 65Y05.
Key words: Parareal algorithm, Diagonalization technique, Krylov subspace, Time-periodic problem, Convergence analysis.

1. Introduction

In this paper, we consider some numerical aspects on parallel-in-time algorithms of the following time-periodic model problem:

$$\begin{aligned} &\mathbf{u}'(t) = \mathbf{f}\big(\mathbf{u}(t), t\big), \quad t \in [0, T], \\ &\mathbf{u}(0) = \mathbf{u}(T), \end{aligned} \tag{1.1}$$

where $\mathbf{f} : \mathbb{R}^d \times [0, T] \to \mathbb{R}^d$ satisfies $\mathbf{f}(\cdot, 0) = \mathbf{f}(\cdot, T)$, and $\mathbf{u} : [0, T] \to \mathbb{R}^d$. Time-periodic problems arise from various practical engineering problems, such as eddy current simulations [1], fluid-structure interaction and cardiac flow problems [34], periodically forced reactors [53, 54]. Different from initial-value problems, since the time periodicity modifies the problem structure

[*] Received March 5, 2024 / Revised version received August 8, 2024 / Accepted December 23, 2024 / Published online February 24, 2025 /

[1)] Corresponding author

and solution methods significantly, the numerical simulation of time-periodic problems is quite special and difficult, see for example [4,45,52].

When the scale of time-periodic problems increases, it is desirable to use parallel methods to solve such problems. Classical parallel methods proposed to solve time-periodic problems usually use the spatial parallelism, such as multigrid methods [32] and waveform relaxation methods [35,55], which have proved to be quite effective. While since the parallelization in space often saturates, parallel-in-time methods have been widely concerned and developed rapidly over the last twenty years. The parareal algorithm is a parallel-in-time method that was first proposed by Lions, Maday, and Turinici [39], where the initial value problems are solved on time subintervals in parallel on fine grids, and through iterations the initial values on each subinterval are corrected to converge to the correct values of the overall solution. A precise convergence analysis has been systematically studied for linear ordinary and partial differential equations [30], and for nonlinear problems [19]. The parareal algorithm has also been applied to many different kinds of problems and application areas, such as dynamical problems [6,12], quantum control problems [42,43], time-periodic problems [25], optimal control problems governed by PDEs [27], fractional PDEs [58], highly oscillatory systems [57], and so on. Besides, other new parallel-in-time algorithms, e.g. PFASST [11,44], MGRIT [10,13], and several direct time parallel methods, e.g. diagonalization technique [21,23,28,41], PARAEXP [18], have also been presented. More recently, several new space-time parallel methods have also been investigated, which are in parallel in both space and time directions, combining the parareal algorithm with waveform relaxation methods [38,40] or Schwarz waveform relaxation methods [5,24]. For a complete overview of historical development of parallel-in-time methods, see [17].

The classical parareal algorithm can produce a speed-up for first order ODEs [16], but when applied to some second order ODEs [2,14,15] or most hyperbolic problems [51], such as wave equation and Hamilton system [20], the method usually becomes unstable, resulting in less than ideal convergence performance or even non convergence. Therefore in [29], a Krylov subspace enhanced parareal algorithm based on the modified PITA framework in [15] was presented and analyzed, as a quite effective remedy for a beating phenomenon when the parareal algorithm is applied to second order ODEs. The parareal algorithm with projection is also considered in [9] and proved to be stable for hyperbolic systems. A direct time parallel method using diagonalization has been presented for the wave equation [22], which is based on the idea first introduced by Maday and Rønquist [41], and further the parareal algorithm using diagonalization as the coarse propagator is also effective for both dissipative and wave propagation problems [31].

Parallel-in-time methods have also been developed for time-periodic problems in recent years. In [25], two parareal algorithms for time-periodic problems were proposed and analyzed: The first one is called the periodic parareal algorithm with periodic coarse problem (PP-PC), and the second one is called the periodic parareal algorithm with initial value coarse problem (PP-IC). Furthermore, by using waveform relaxation as the fine propagator on each subinterval in the parareal framework for time-periodic problems, two new space-time parallel algorithms have been developed based on the PP-PC [48] and PP-IC [49] algorithms. The parareal algorithm for time-periodic problems with discontinuous inputs has also been developed [26]. The ideas based on PP-PC and PP-IC have also been widely used to solve eddy current simulations [37] and cardiac flow problems [34]. Meanwhile, solving time-periodic wave problems can usually use solving the corresponding time-harmonic problems by some fast and parameter-robust preconditioned solvers, see [36,46]. As a guiding example, solving the Helmholtz equation is also a difficult research field, which is concerned and developed gradually.

In this paper, we mainly consider the parareal algorithm that is a parallel-in-time method to fast solve various types of time-periodic problems in the time direction. We could find that the classical PP-PC and PP-IC algorithms usually converge slowly, and using PP-IC even diverges for wave propagation problems. Therefore, we have already proposed the Krylov subspace enhanced PP-PC algorithm in [50]. However, a costly time-periodic coarse problem for each iteration in the PP-PC framework needs to be solved. So in order to reduce the computational cost, we further construct the Krylov subspace enhanced PP-IC algorithm, then see that it has quite bad convergence behavior and also diverges for wave propagation problems and Burgers equation. Consequently, in this work we design two new PP-IC algorithms to improve the convergence behavior of the classical PP-IC algorithm. Firstly, for the coarse propagator in the classical PP-IC we introduce a diagonalization technique proposed by [31]. Secondly, since the diagonalization-based PP-IC algorithm still converges slowly, we further consider combining it with the Krylov subspace method to accelerate convergence speed.

The rest of our paper is organized as follows. In Section 2, we construct a new parareal algorithm for time-periodic problems by taking a diagonalization technique into the PP-IC framework and present the convergence analysis both for the linear and the nonlinear cases. In Section 3, we apply the Krylov subspace enhanced method to the algorithm proposed in Section 2 and prove the convergence result of the new algorithm for the general nonlinear system (1.1). In Section 5, we illustrate our theoretical analysis with several numerical experiments, and our conclusions are presented finally in Section 6.

2. Diagonalization-based PP-IC Algorithm

In this section, we first construct a diagonalization-based parareal algorithm for the time-periodic model problem (1.1). As in the classical parareal algorithm [30], we decompose the time interval $[0, T]$ into N equal subintervals $[T_n, T_{n+1}], n = 0, 1, \dots, N-1$ with $0 = T_0 < T_1 < \dots < T_N = T$, and the time step $\Delta T = T/N, N \geq 1$. For each $[T_n, T_{n+1}]$, we continue to divide into M equal subintervals, with the time step $\Delta t = \Delta T/M$. Within the parareal algorithm, two propagators are applied: the fine propagator $F(T_n, T_{n-1}, \mathbf{x})$ and the coarse propagator $G(T_n, T_{n-1}, \mathbf{x})$. In particular, the coarse propagator $G(T_n, T_{n-1}, \mathbf{x})$ provides a rough approximation of the solution $\mathbf{u}(T_n)$, where $\mathbf{u}$ is the solution of (1.1) with $\mathbf{u}(T_{n-1}) = \mathbf{x}$ as an initial condition, while the fine propagator $F(T_n, T_{n-1}, \mathbf{x})$ provides a more accurate approximation of the same solution $\mathbf{u}(T_n)$. In [25], a parallel-in-time algorithm for time-periodic problems presented is PP-IC: periodic parareal algorithm with initial value coarse problem. The PP-IC algorithm can be written for $k = 0, 1, \dots$ as follows:

$$\begin{aligned}
&\mathbf{U}_0^{k+1} = \mathbf{U}_N^k, \\
&\mathbf{U}_{n+1}^{k+1} = F\big(T_{n+1}, T_n, \mathbf{U}_n^k\big) + G\big(T_{n+1}, T_n, \mathbf{U}_n^{k+1}\big) - G\big(T_{n+1}, T_n, \mathbf{U}_n^k\big), \\
&n = 0, \dots, N-1,
\end{aligned} \tag{2.1}$$

where $\mathbf{U}_n^0, n = 0, 1, \dots, N$ is an initial guess of the PP-IC iteration.

We now consider the construction of a diagonalization-based parareal algorithm for the system (1.1). In [31], an idea to parallelize the coarse propagator G in the PP-IC algorithm (2.1) using diagonalization in time was proposed: we solve with G on each subinterval $[T_n, T_{n+1}]$ a head-tail coupled problem

$$\mathbf{u}'(t) = \mathbf{f}\big(\mathbf{u}(t), t\big), \quad \mathbf{u}(T_n) = \alpha \mathbf{u}(T_{n+1}) + (1-\alpha)\mathbf{U}_n^k, \quad t \in [T_n, T_{n+1}], \tag{2.2}$$

where $\alpha \in (0,1)$ is a parameter. We will next see that the head-tail coupled condition results in circulant matrices of the time discretization such that we can apply the diagonalization technique to solve (2.2) efficiently. Then we obtain the diagonalization-based parareal algorithm for time-periodic problems as follows:

$$\begin{aligned}
&\mathbf{U}_0^{k+1} = \mathbf{U}_N^k,\\
&\mathbf{U}_{n+1}^{k+1} = F\big(T_{n+1}, T_n, \mathbf{U}_n^k\big) + F^*\big(\alpha, T_{n+1}, T_n, \mathbf{U}_n^{k+1}\big) - F^*\big(\alpha, T_{n+1}, T_n, \mathbf{U}_n^k\big),\\
&n = 0, \dots, N-1,
\end{aligned} \tag{2.3}$$

where $F^*(\alpha, T_{n+1}, T_n, \mathbf{U}_n^k)$ denotes the solution of (2.2) at $t = T_{n+1}$. We call the algorithm (2.3) the diagonalization-based PP-IC algorithm (Diag PP-IC). For example, using the linear θ-method, $F^*(\alpha, T_{n+1}, T_n, \mathbf{U}_n^k)$ is specified as

$$\begin{aligned}
&\mathbf{z}_0 = \alpha \mathbf{z}_M + (1-\alpha)\mathbf{U}_n^k,\\
&\frac{\mathbf{z}_{j+1} - \mathbf{z}_j}{\Delta t} = \theta \mathbf{f}(\mathbf{z}_{j+1}, t_{n,j+1}) + (1-\theta)\mathbf{f}(\mathbf{z}_j, t_{n,j}), \quad j = 0, 1, \dots, M-1,\\
&F^*\big(\alpha, T_{n+1}, T_n, \mathbf{U}_n^k\big) := \mathbf{z}_M,
\end{aligned} \tag{2.4}$$

where $t_{n,j} = T_n + j\Delta t$. Here we use the same step-size for both the F and F^* propagators to reduce the computational cost.

We next explain the details for computing $F^*(\alpha, T_{n+1}, T_n, \mathbf{U}_n^k)$ by using the linear θ-method (2.4) and then present the convergence analysis both for the linear and nonlinear cases.

2.1. The linear case

We first consider the following linear ODEs:

$$\begin{aligned}
&\mathbf{u}'(t) + A\mathbf{u}(t) = \mathbf{f}(t), \quad t \in [0, T],\\
&\mathbf{u}(0) = \mathbf{u}(T),
\end{aligned} \tag{2.5}$$

where $A \in \mathbb{R}^{d\times d}$, $\mathbf{f} : [0,T] \to \mathbb{R}^d$ satisfies $\mathbf{f}(0) = \mathbf{f}(T)$, and $\mathbf{u} : [0,T] \to \mathbb{R}^d$. The computation of $F^*(\alpha, T_{n+1}, T_n, \mathbf{U}_n^k)$ using the linear θ-method (2.4) can be represented as

$$\left(\frac{1}{\Delta t}\underbrace{\begin{bmatrix} 1 & & & -\alpha \\ -1 & 1 & & \\ & \ddots & \ddots & \\ & & -1 & 1 \end{bmatrix}}_{=:B_1} \otimes I_x + \underbrace{\begin{bmatrix} \theta & & & \alpha(1-\theta) \\ 1-\theta & \theta & & \\ & \ddots & \ddots & \\ & & 1-\theta & \theta \end{bmatrix}}_{=:B_2} \otimes A\right) \underbrace{\begin{pmatrix} \mathbf{z}_1 \\ \mathbf{z}_2 \\ \vdots \\ \mathbf{z}_M \end{pmatrix}}_{=:\mathbf{Z}} = \mathbf{W}_n^k, \tag{2.6}$$

where

$$\begin{aligned}
&\mathbf{W}_n^k = \left(\mathbf{g}_{n,1}^T + (1-\alpha)\left(\frac{I_x}{\Delta t} - (1-\theta)A\right)\left(\mathbf{U}_n^k\right)^T, \mathbf{g}_{n,2}^T, \dots, \mathbf{g}_{n,M}^T\right)^T,\\
&\mathbf{g}_{n,j+1} = \theta \mathbf{f}(t_{n,j+1}) + (1-\theta)\mathbf{f}(t_{n,j}), \quad j = 0, 1, \dots, M-1,
\end{aligned}$$

i.e. (2.6) can be expressed as

$$(B_1 \otimes I_x + \Delta t B_2 \otimes A)\mathbf{Z} = \Delta t \mathbf{W}_n^k, \quad F^*\big(\alpha, T_{n+1}, T_n, \mathbf{U}_n^k\big) =: \mathbf{z}_M.$$

Note that both B_1 and B_2 are circulant matrices, and this means that they can be simultaneously diagonalized as follows.

Lemma 2.1 ([31, Lemma 2.1]). *The matrices B_1 and $B_2 \in \mathbb{R}^{M\times M}$ can be diagonalized as*

$$B_1 = V_\alpha D_1 V_\alpha^{-1}, \quad B_2 = V_\alpha D_2 V_\alpha^{-1}, \tag{2.7}$$

where $V_\alpha = \Lambda_\alpha F$, and

$$\begin{aligned}
&\Lambda_\alpha = \mathrm{diag}\big(1, \alpha^{-\frac{1}{M}}, \dots, \alpha^{-\frac{M-1}{M}}\big), && \\
&F = (\omega_1, \omega_2, \dots, \omega_M), && \omega_j = \big(1, e^{i\frac{2(j-1)\pi}{M}}, \dots, e^{i\frac{2(j-1)(M-1)\pi}{M}}\big)^T, \\
&D_1 = \mathrm{diag}(\lambda_1, \lambda_2, \dots, \lambda_M), && \lambda_j = 1 - \alpha^{\frac{1}{M}} e^{-i\frac{2(j-1)\pi}{M}}, \\
&D_2 = \mathrm{diag}(\tilde\lambda_1, \tilde\lambda_2, \dots, \tilde\lambda_M), && \tilde\lambda_j = \theta + (1-\theta)\alpha^{\frac{1}{M}} e^{-i\frac{2(j-1)\pi}{M}}.
\end{aligned}$$

Substituting (2.7) into (2.6) gives the following three steps for the computation of $F^*(\alpha, T_{n+1}, T_n, \mathbf{U}_n^k)$:

$$\begin{aligned}
&\text{(a)} \quad \big((\Lambda_\alpha F)\otimes I_x\big)\mathbf{P} = \Delta t \mathbf{W}_n^k, \\
&\text{(b)} \quad \big(\lambda_j I_x + \tilde\lambda_j(\Delta t A)\big)\mathbf{q}_j = \mathbf{p}_j, \quad j = 1, 2, \dots, M, \\
&\text{(c)} \quad \big((\Lambda_\alpha F)^{-1}\otimes I_x\big)\mathbf{Z} = \mathbf{Q},
\end{aligned} \tag{2.8}$$

where

$$\mathbf{P} = \big(\mathbf{p}_1^T, \mathbf{p}_2^T, \dots, \mathbf{p}_M^T\big)^T, \quad \mathbf{Q} = \big(\mathbf{q}_1^T, \mathbf{q}_2^T, \dots, \mathbf{q}_M^T\big)^T \in \mathbb{C}^{dM}.$$

In (2.8), step (b) is highly parallelizable for all M time points, while step (a) and (c) can be done by the fast fourier transform (FFT) since F is a Fourier matrix.

We next analyze the convergence factor of the Diag PP-IC algorithm (2.3) for the linear ODEs (2.5). The following lemmas are useful to estimate the spectral radius of the iteration matrix $M_1^{-1}N_1$.

Lemma 2.2 ([31, Lemma 3.1]). *Let $\sigma \in [0, 1]$. Then, for the matrix*

$$M_1 = \begin{pmatrix} 1 & & & \\ -\sigma & 1 & & \\ & \ddots & \ddots & \\ & & -\sigma & 1 \end{pmatrix} \in \mathbb{C}^{N\times N},$$

the inverse M_1^{-1} is a non-negative matrix, and $\|M_1^{-1}\|_\infty = (1-\sigma^N)/(1-\sigma)$.

Remark 2.1. In Lemma 2.2, when $\sigma = 1$, then $(1-\sigma^N)/(1-\sigma)$ should be understood from the limiting sense, i.e. $\lim_{\sigma\to 1}((1-\sigma^N)/(1-\sigma)) = N$.

Lemma 2.3 ([25, Remark 3.2]). *For the matrices*

$$M_1 = \begin{pmatrix} 1 & & & \\ -\sigma & 1 & & \\ & \ddots & \ddots & \\ & & -\sigma & 1 \end{pmatrix}, \quad N_1 = \begin{pmatrix} 0 & & & 1 \\ \kappa & 0 & & \\ & \ddots & \ddots & \\ & & \kappa & 0 \end{pmatrix},$$

if $|\sigma| + |\kappa| < 1$, then the spectral radius $\rho(M_1^{-1}N_1)$ of the matrix $M_1^{-1}N_1$ is bounded by $\rho(M_1^{-1}N_1) < x_l < 1, l \geq 1$, where $x_l = (|\sigma| x_{l-1} + |\kappa|)^{N/(N+1)}, x_0 = 1$.

Theorem 2.1. *For the linear ODEs* (2.5)*, assume that the coefficient matrix A is diagonalizable with $A = S\mathrm{diag}(\mu_1, \mu_2, \ldots, \mu_d)S^{-1}$, and $\sigma(A) = \{\mu_1, \mu_2, \ldots, \mu_d\} \subseteq \mathbb{R}^+$. Let the F-propagator be an A-stable implicit Runge-Kutta (IRK) method with stability function $\mathcal{R}(z)$, and $M \geq 2$ be an even integer. Let $\{\mathbf{u}(T_n)\}_{n=0}^N$ be the solutions generated by F at the coarse time points $\{T_n\}_{n=0}^N$, i.e.*

$$\mathbf{u}(T_n) = F\big(T_n, T_{n-1}, \mathbf{u}(T_{n-1})\big), \quad n = 1, \ldots, N,$$

$\mathbf{u}(T_0) = \mathbf{u}(T_N)$, and $\{\mathbf{U}_n^k\}_{n=0}^N$ be the k-th iterative solutions generated by the Diag PP-IC algorithm (2.3)*. Then, the iteration error $\mathbf{e}_n^k := \mathbf{u}(T_n) - \mathbf{U}_n^k$, i.e. $\mathbf{e}^k := (\mathbf{e}_0^k, \mathbf{e}_1^k, \ldots, \mathbf{e}_N^k)^T$ satisfies $S\mathbf{e}^{k+1} \leq M_1^{-1}N_1 S\mathbf{e}^k$ for any $k \geq 1$, and the asymptotic convergence factor of the Diag PP-IC algorithm* (2.3) *is bounded by $\rho(M_1^{-1}N_1) < x_l < 1$ for any $l \geq 1$, where*

$$x_l = (|\sigma| x_{l-1} + |\kappa|)^{\frac{N}{N+1}}, \quad x_0 = 1,$$

and

$$\begin{aligned}
&\sigma = |\mathcal{R}_g(\alpha, M, z)|,\\
&\kappa = \big|[\mathcal{R}(z/M)]^M - \mathcal{R}_g(\alpha, M, z)\big|,\\
&\mathcal{R}_g(\alpha, M, z) = \frac{(1-\alpha)[\mathcal{R}(z/M)]^M}{1 - \alpha[\mathcal{R}(z/M)]^M}.
\end{aligned}$$

Proof. We estimate the error of the Diag PP-IC algorithm (2.3) at iteration step $k+1$,

$$\begin{aligned}
\mathbf{u}(T_n) - \mathbf{U}_n^{k+1} = &\big[F^*\big(\alpha, T_n, T_{n-1}, \mathbf{u}(T_{n-1})\big) - F^*\big(\alpha, T_n, T_{n-1}, \mathbf{U}_{n-1}^{k+1}\big)\big]\\
&+ \big[F\big(T_n, T_{n-1}, \mathbf{u}(T_{n-1})\big) - F\big(T_n, T_{n-1}, \mathbf{U}_{n-1}^k\big)\big]\\
&- \big[F^*\big(\alpha, T_n, T_{n-1}, \mathbf{u}(T_{n-1})\big) - F^*\big(\alpha, T_n, T_{n-1}, \mathbf{U}_{n-1}^k\big)\big], \quad n = 1, \ldots, N. \quad (2.9)
\end{aligned}$$

Assume that performing one-step of the IRK method can be expressed as

$$\mathbf{u}(t_{n,j+1}) = \mathcal{R}(\Delta t A)\mathbf{u}(t_{n,j}) + \tilde{\mathbf{g}}_{n,j} = \mathcal{R}(\Delta T A/M)\mathbf{u}(t_{n,j}) + \tilde{\mathbf{g}}_{n,j},$$

where $t_{n,j} = T_n + j\Delta t$. Therefore,

$$F\big(T_n, T_{n-1}, \mathbf{u}(T_{n-1})\big) - F\big(T_n, T_{n-1}, \mathbf{U}_{n-1}^k\big) = [\mathcal{R}(\Delta T A/M)]^M \mathbf{e}_{n-1}^k. \quad (2.10)$$

Moreover, $F^*(\alpha, T_{n+1}, T_n, \mathbf{U}_n^k)$ can be computed as

$$\begin{aligned}
&\mathbf{z}_0 = \alpha \mathbf{z}_M + (1-\alpha)\mathbf{U}_n^k,\\
&\mathbf{z}_{j+1} = \mathcal{R}(\Delta T A/M)\mathbf{z}_j + \tilde{\mathbf{g}}_j, \quad j = 0, 1, \ldots, M-1,\\
&\mathbf{z}_M = F^*\big(\alpha, T_{n+1}, T_n, \mathbf{U}_n^k\big).
\end{aligned}$$

Then, we have

$$\begin{aligned}
\mathbf{z}_M &= \mathcal{R}(\Delta T A/M)\mathbf{z}_{M-1} + \tilde{\mathbf{g}}_{M-1}\\
&= \mathcal{R}(\Delta T A/M)[\mathcal{R}(\Delta T A/M)\mathbf{z}_{M-2} + \tilde{\mathbf{g}}_{M-2}] + \tilde{\mathbf{g}}_{M-1}\\
&= \cdots = [\mathcal{R}(\Delta T A/M)]^M \mathbf{z}_0 + \sum_{i=1}^M [\mathcal{R}(\Delta T A/M)]^{i-1}\tilde{\mathbf{g}}_{M-i}\\
&= \alpha[\mathcal{R}(\Delta T A/M)]^M \mathbf{z}_M + (1-\alpha)[\mathcal{R}(\Delta T A/M)]^M \mathbf{U}_n^k + \sum_{i=1}^M [\mathcal{R}(\Delta T A/M)]^{i-1}\tilde{\mathbf{g}}_{M-i}.
\end{aligned}$$

Hence,

$$\begin{aligned}&F^*\big(\alpha, T_{n+1}, T_n, \mathbf{U}_n^k\big)\\&= (1-\alpha)\big\{I_x - \alpha[\mathcal{R}(\Delta TA/M)]^M\big\}^{-1}[\mathcal{R}(\Delta TA/M)]^M\mathbf{U}_n^k\\&\quad + \big\{I_x - \alpha[R(\Delta TA/M)]^M\big\}^{-1}\sum_{i=1}^{M}[\mathcal{R}(\Delta TA/M)]^{i-1}\tilde{\mathbf{g}}_{M-i}.\end{aligned}$$

Then, we get

$$\begin{aligned}&F^*\big(\alpha, T_n, T_{n-1}, \mathbf{u}(T_{n-1})\big) - F^*\big(\alpha, T_n, T_{n-1}, \mathbf{U}_{n-1}^k\big)\\&= (1-\alpha)\big\{I_x - \alpha[\mathcal{R}(\Delta TA/M)]^M\big\}^{-1}[\mathcal{R}(\Delta TA/M)]^M\mathbf{e}_{n-1}^k\\&= \mathcal{R}_g(\alpha, M, \Delta TA)\mathbf{e}_{n-1}^k,\end{aligned}\tag{2.11}$$

where

$$\mathcal{R}_g(\alpha, M, \Delta TA) = (1-\alpha)\big\{I_x - \alpha[\mathcal{R}(\Delta TA/M)]^M\big\}^{-1}[\mathcal{R}(\Delta TA/M)]^M.$$

Substituting (2.10) and (2.11) into (2.9) gives

$$\begin{aligned}\mathbf{e}_n^{k+1} &= \mathcal{R}_g(\alpha, M, \Delta TA)\mathbf{e}_{n-1}^{k+1} + [\mathcal{R}(\Delta TA/M)]^M\mathbf{e}_{n-1}^k\\&\quad - \mathcal{R}_g(\alpha, M, \Delta TA)\mathbf{e}_{n-1}^k, \quad n = 1, \dots, N,\end{aligned}\tag{2.12}$$

and $\mathbf{e}_0^{k+1} = \mathbf{e}_N^k$. Since $A = S\boldsymbol{\mu}S^{-1}$ with $\boldsymbol{\mu} = \text{diag}(\mu_1, \mu_2, \dots, \mu_d)$, then from (2.12), we obtain

$$\begin{aligned}S\mathbf{e}_n^{k+1} &= S\mathcal{R}_g(\alpha, M, \Delta TA)S^{-1}S\mathbf{e}_{n-1}^{k+1} + S[\mathcal{R}(\Delta TA/M)]^M S^{-1}S\mathbf{e}_{n-1}^k\\&\quad - S\mathcal{R}_g(\alpha, M, \Delta TA)S^{-1}S\mathbf{e}_{n-1}^k, \quad n = 1, \dots, N,\end{aligned}$$

and $S\mathbf{e}_0^{k+1} = S\mathbf{e}_N^k$. Let $z = \Delta T\mu$, where $\mu \in \sigma(A)$ is an arbitrary eigenvalue of A and $\zeta_n^k(\mu)$ is the corresponding component of $S\mathbf{e}_n^k$. Then we have

$$\begin{aligned}\zeta_n^{k+1}(\mu) &= \mathcal{R}_g(\alpha, M, z)\zeta_{n-1}^{k+1}(\mu) + [\mathcal{R}(z/M)]^M\zeta_{n-1}^k(\mu)\\&\quad - \mathcal{R}_g(\alpha, M, z)\zeta_{n-1}^k(\mu), \quad n = 1, \dots, N,\end{aligned}$$

and $\zeta_0^{k+1}(\mu) = \zeta_N^k(\mu)$. Further,

$$\begin{aligned}\big|\zeta_n^{k+1}(\mu)\big| &\le \big|\mathcal{R}_g(\alpha, M, z)\big|\,\big|\zeta_{n-1}^{k+1}(\mu)\big|\\&\quad + \big|[\mathcal{R}(z/M)]^M - \mathcal{R}_g(\alpha, M, z)\big|\,\big|\zeta_{n-1}^k(\mu)\big|, \quad n = 1, \dots, N,\end{aligned}\tag{2.13}$$

and $\zeta_0^{k+1}(\mu) = \zeta_N^k(\mu)$. From (2.13), we have

$$\underbrace{\begin{pmatrix}1 & & & \\ -\sigma & 1 & & \\ & \ddots & \ddots & \\ & & -\sigma & 1\end{pmatrix}}_{=:M_1}\underbrace{\begin{pmatrix}\zeta_0^{k+1}(\mu)\\ \zeta_1^{k+1}(\mu)\\ \vdots\\ \zeta_N^{k+1}(\mu)\end{pmatrix}}_{=:\boldsymbol{\zeta}^{k+1}(\mu)} \le \underbrace{\begin{pmatrix}0 & & & 1\\ \kappa & 0 & & \\ & \ddots & \ddots & \\ & & \kappa & 0\end{pmatrix}}_{=:N_1}\underbrace{\begin{pmatrix}\zeta_0^k(\mu)\\ \zeta_1^k(\mu)\\ \vdots\\ \zeta_N^k(\mu)\end{pmatrix}}_{=:\boldsymbol{\zeta}^k(\mu)},\tag{2.14}$$

where

$$\sigma = |\mathcal{R}_g(\alpha, M, z)|, \quad \kappa = \big|[\mathcal{R}(z/M)]^M - \mathcal{R}_g(\alpha, M, z)\big|.$$

So (2.14) can be written as

$$M_1\boldsymbol{\zeta}^{k+1}(\mu) \leq N_1\boldsymbol{\zeta}^k(\mu). \tag{2.15}$$

Since the F-propagator is an A-stable IRK method, then

$$|[\mathcal{R}(z/M)]^M| < 1, \quad \forall z \in \sigma(\Delta TA).$$

Therefore, for $\alpha \in (0,1)$, we have

$$\sigma = |\mathcal{R}_g(\alpha, M, z)| = \frac{(1-\alpha)|[\mathcal{R}(z/M)]^M|}{|1-\alpha[\mathcal{R}(z/M)]^M|} \leq \frac{(1-\alpha)|[\mathcal{R}(z/M)]^M|}{1-|\alpha[\mathcal{R}(z/M)]^M|} < \frac{1-\alpha}{1-|\alpha|} = 1.$$

Hence, by using Lemma 2.2, we can see that M_1^{-1} is a non-negative matrix, and from (2.15) it holds that

$$\boldsymbol{\zeta}^{k+1}(\mu) \leq M_1^{-1}N_1\boldsymbol{\zeta}^k(\mu), \tag{2.16}$$

i.e.

$$S\mathbf{e}^{k+1} \leq M_1^{-1}N_1S\mathbf{e}^k. \tag{2.17}$$

Furthermore, we get

$$\begin{aligned} |\sigma| + |\kappa| &= |\mathcal{R}_g(\alpha, M, z)| + \left|[\mathcal{R}(z/M)]^M - \mathcal{R}_g(\alpha, M, z)\right| \\ &= \frac{(1-\alpha)|[\mathcal{R}(z/M)]^M|}{|1-\alpha[\mathcal{R}(z/M)]^M|} + \left|[\mathcal{R}(z/M)]^M - \frac{(1-\alpha)[\mathcal{R}(z/M)]^M}{1-\alpha[\mathcal{R}(z/M)]^M}\right| \\ &= \frac{(1-\alpha)|[\mathcal{R}(z/M)]^M| + \alpha|[\mathcal{R}(z/M)]^M|\,|1-[\mathcal{R}(z/M)]^M|}{|1-\alpha[\mathcal{R}(z/M)]^M|}. \end{aligned}$$

Since $\sigma(A) \subseteq \mathbb{R}^+$, we have $z \in \sigma(\Delta TA) \subseteq \mathbb{R}^+$. Moreover, since F is an A-stable IRK method with stability function $\mathcal{R}(z)$, and $M \geq 2$ is an even integer, we get $[\mathcal{R}(z/M)]^M \in (0,1)$. Therefore,

$$|\sigma| + |\kappa| = [\mathcal{R}(z/M)]^M < 1. \tag{2.18}$$

From (2.18), using Lemma 2.3 completes the proof. □

Remark 2.2. In Theorem 2.1, for the linear ODEs (2.5), we mainly consider a representative case, i.e. the heat equation, where A is a symmetric positive definite matrix such that $\sigma(A) \subseteq \mathbb{R}^+$.

2.2. The nonlinear case

Now, we consider the general nonlinear problem (1.1). We first explain the details of the diagonalization technique for computing nonlinear $F^*(\alpha, T_{n+1}, T_n, \mathbf{U}_n^k)$. By using again the linear θ-method (2.4), we have

$$\left(\frac{1}{\Delta t}B_1 \otimes I_x\right)\mathbf{Z} = \underbrace{\begin{bmatrix} \theta\mathbf{f}(\mathbf{z}_1, t_{n,1}) + (1-\theta)\mathbf{f}(\alpha\mathbf{z}_M + (1-\alpha)\mathbf{U}_n^k, t_{n,0}) \\ \theta\mathbf{f}(\mathbf{z}_2, t_{n,2}) + (1-\theta)\mathbf{f}(\mathbf{z}_1, t_{n,1}) \\ \vdots \\ \theta\mathbf{f}(\mathbf{z}_M, t_{n,M}) + (1-\theta)\mathbf{f}(\mathbf{z}_{M-1}, t_{n,M-1}) \end{bmatrix}}_{=:\mathbf{F}} + \mathbf{W}_n^k, \tag{2.19}$$

where

$$\mathbf{W}_n^k = \left(\frac{(1-\alpha)\big(\mathbf{U}_n^k\big)^T}{\Delta t}, \mathbf{0}, \dots, \mathbf{0} \right)^T, \quad \mathbf{Z} = \big(\mathbf{z}_1^T, \mathbf{z}_2^T, \dots, \mathbf{z}_M^T\big)^T,$$

and the matrix B_1 is given by (2.6). Applying Newton's method to (2.19) gives

$$\mathbf{Z}^{(l+1)} = \mathbf{Z}^{(l)} - \left(\frac{1}{\Delta t} B_1 \otimes I_x - \mathbf{J}(\mathbf{Z}^{(l)}) \right)^{-1} \left(\left(\frac{1}{\Delta t} B_1 \otimes I_x \right) \mathbf{Z}^{(l)} - \mathbf{F}(\mathbf{Z}^{(l)}) - \mathbf{W}_n^k \right),$$

i.e.

$$\left(\frac{1}{\Delta t} B_1 \otimes I_x - \mathbf{J}(\mathbf{Z}^{(l)}) \right) \mathbf{Z}^{(l+1)} = \mathbf{F}(\mathbf{Z}^{(l)}) + \mathbf{W}_n^k - \mathbf{J}(\mathbf{Z}^{(l)})\mathbf{Z}^{(l)}, \tag{2.20}$$

where

$$\mathbf{Z}^{(l)} = \left(\big(\mathbf{z}_1^{(l)}\big)^T, \big(\mathbf{z}_2^{(l)}\big)^T, \dots, \big(\mathbf{z}_M^{(l)}\big)^T \right)^T,$$

and the Jacobian matrix $\mathbf{J}(\mathbf{Z}^{(l)})$ of $\mathbf{F}$ is

$$\mathbf{J}(\mathbf{Z}^{(l)}) = \begin{bmatrix} \theta \nabla \mathbf{f}\big(\mathbf{z}_1^{(l)}, t_{n,1}\big) & & & (1-\theta)\alpha \nabla \mathbf{f}\big(\alpha \mathbf{z}_M^{(l)} + (1-\alpha)\mathbf{U}_n^k, t_{n,0}\big) \\ (1-\theta)\nabla \mathbf{f}\big(\mathbf{z}_1^{(l)}, t_{n,1}\big) & \theta \nabla \mathbf{f}\big(\mathbf{z}_2^{(l)}, t_{n,2}\big) & & \\ & \ddots & \ddots & \\ & & (1-\theta)\nabla \mathbf{f}\big(\mathbf{z}_{M-1}^{(l)}, t_{n,M-1}\big) & \theta \nabla \mathbf{f}\big(\mathbf{z}_M^{(l)}, t_{n,M}\big) \end{bmatrix}.$$

To solve the linear system (2.20) by the diagonalization technique, following the idea in [21], we replace the Jacobian matrix $\mathbf{J}(\mathbf{Z}^{(l)})$ by $I_M \otimes \tilde{\mathbf{J}}(\mathbf{Z}^{(l)})$, where

$$\tilde{\mathbf{J}}(\mathbf{Z}^{(l)}) := \frac{1}{M} \left(\sum_{j=1}^{M-1} \theta \nabla \mathbf{f}\big(\mathbf{z}_j^{(l)}, t_{n,j}\big) + (1-\theta)\alpha \nabla \mathbf{f}\big(\alpha \mathbf{z}_M^{(l)} + (1-\alpha)\mathbf{U}_n^k, t_{n,0}\big) \right). \tag{2.21}$$

Substituting (2.21) into (2.20) leads to the simplified Newton iteration

$$\left(\frac{1}{\Delta t} B_1 \otimes I_x - I_M \otimes \tilde{\mathbf{J}}(\mathbf{Z}^{(l)}) \right) \mathbf{Z}^{(l+1)} = \mathbf{F}(\mathbf{Z}^{(l)}) + \mathbf{W}_n^k - \big(I_M \otimes \tilde{\mathbf{J}}(\mathbf{Z}^{(l)})\big)\mathbf{Z}^{(l)}. \tag{2.22}$$

Therefore, with the spectral decomposition of B_1 in Lemma 2.1, similar to (2.8), we can solve $\mathbf{Z}^{(l+1)}$ by the diagonalization technique as follows:

$$\begin{aligned} &\text{(a)} \quad \big((\Lambda_\alpha F) \otimes I_x\big)\mathbf{P} = \Delta t \mathbf{F}(\mathbf{Z}^{(l)}) + \Delta t \mathbf{W}_n^k - \Delta t\big(I_M \otimes \tilde{\mathbf{J}}(\mathbf{Z}^{(l)})\big)\mathbf{Z}^{(l)}, \\ &\text{(b)} \quad \big(\lambda_j I_x + \Delta t \tilde{\mathbf{J}}(\mathbf{Z}^{(l)})\big)\mathbf{q}_j = \mathbf{p}_j, \quad j = 1, 2, \dots, M, \\ &\text{(c)} \quad \big((\Lambda_\alpha F)^{-1} \otimes I_x\big)\mathbf{Z}^{(l+1)} = \mathbf{Q}, \end{aligned} \tag{2.23}$$

where

$$\mathbf{P} = \big(\mathbf{p}_1^T, \mathbf{p}_2^T, \dots, \mathbf{p}_M^T\big)^T, \quad \mathbf{Q} = \big(\mathbf{q}_1^T, \mathbf{q}_2^T, \dots, \mathbf{q}_M^T\big)^T \in \mathbb{C}^{dM}.$$

We next present the convergence analysis of the Diag PP-IC algorithm (2.3) for the nonlinear model problem (1.1). We first make the following assumptions.

Assumption 2.1. For the nonlinear function $\mathbf{f}(\cdot, t)$, there exists a nonnegative constant L_1 such that

$$\langle \mathbf{f}(\mathbf{u}, t) - \mathbf{f}(\mathbf{v}, t), \mathbf{u} - \mathbf{v} \rangle \leq -L_1 \|\mathbf{u} - \mathbf{v}\|_2^2, \quad \forall\, t \in [0, T], \quad \forall\, \mathbf{u}, \mathbf{v} \in \mathbb{R}^d,$$

where $\langle \cdot\, , \cdot \rangle$ denotes the Euclidean inner product.

Assumption 2.2. For the nonlinear function $\mathbf{f}(\cdot,t)$, there exists a nonnegative constant L_2 such that

$$\begin{aligned}&\big\langle \mathbf{f}(\mathbf{u}_1,t)-\mathbf{f}(\mathbf{u}_2,t)-\big(\mathbf{f}(\mathbf{v}_1,t)-\mathbf{f}(\mathbf{v}_2,t)\big),\mathbf{u}_1-\mathbf{u}_2-(\mathbf{v}_1-\mathbf{v}_2)\big\rangle\\ &\le -L_2\|\mathbf{u}_1-\mathbf{u}_2-(\mathbf{v}_1-\mathbf{v}_2)\|_2^2,\quad \forall\, t\in[0,T],\quad \forall\,\mathbf{u}_1,\mathbf{u}_2,\mathbf{v}_1,\mathbf{v}_2\in\mathbb{R}^d,\end{aligned}$$

where $\langle\cdot\,,\cdot\rangle$ denotes the Euclidean inner product.

Assumption 2.3. For the nonlinear function $\mathbf{f}(\cdot,t)$, there exists a nonnegative constant L_3 such that

$$\|\mathbf{f}(\mathbf{u},t)-\mathbf{f}(\mathbf{v},t)\|_2\le L_3\|\mathbf{u}-\mathbf{v}\|_2,\quad \forall\, t\in[0,T],\quad \forall\,\mathbf{u},\mathbf{v}\in\mathbb{R}^d.$$

We further assume that the F-propagator is an exact solver, and have the following lemmas.

Lemma 2.4 ([25, Lemma 5.1]). *Under Assumption* 2.1*, the exact propagator F satisfies*

$$\|F(T_{n+1},T_n,\mathbf{x})-F(T_{n+1},T_n,\mathbf{y})\|_2\le e^{-L_1\Delta T}\|\mathbf{x}-\mathbf{y}\|_2.$$

Lemma 2.5 ([31, Lemma 4.1]). *Under Assumption* 2.1*, the F^*-propagator verifies that*

$$\|F^*(\alpha,T_{n+1},T_n,\mathbf{x})-F^*(\alpha,T_{n+1},T_n,\mathbf{y})\|_2\le\frac{(1-\alpha)e^{-L_1\Delta T}}{1-\alpha e^{-L_1\Delta T}}\|\mathbf{x}-\mathbf{y}\|_2.$$

Lemma 2.6 ([31, Lemma 4.2]). *Under Assumptions* 2.1-2.3*, it holds that*

$$\begin{aligned}&\big\|F(T_{n+1},T_n,\mathbf{x})-F^*(\alpha,T_{n+1},T_n,\mathbf{x})-\big(F(T_{n+1},T_n,\mathbf{y})-F^*(\alpha,T_{n+1},T_n,\mathbf{y})\big)\big\|_2\\ &\le\alpha\frac{L_3}{L_2-(1-\alpha)L_1}\big(e^{-L_1\Delta T}-e^{-\frac{L_2}{1-\alpha}\Delta T}\big)\|\mathbf{x}-\mathbf{y}\|_2.\end{aligned}$$

Therefore, we can present two different estimates on the convergence factor of the Diag PP-IC algorithm (2.3) under different assumptions for the nonlinear case (1.1).

Theorem 2.2. *Under Assumptions* 2.1-2.3*, and the condition $|\sigma|+|\kappa|<1$, where*

$$\sigma=\frac{(1-\alpha)e^{-L_1\Delta T}}{1-\alpha e^{-L_1\Delta T}},\quad \kappa=\alpha\frac{L_3}{L_2-(1-\alpha)L_1}\big(e^{-L_1\Delta T}-e^{-\frac{L_2}{1-\alpha}\Delta T}\big),$$

then the errors of the Diag PP-IC algorithm (2.3) *are dominated by a linear iteration $\tilde{S}$, i.e. $\mathbf{e}^{k+1}\le\tilde{S}\mathbf{e}^k$ holds componentwise for*

$$\mathbf{e}^k:=\big(\|\mathbf{u}(T_0)-\mathbf{U}_0^k\|_2,\|\mathbf{u}(T_1)-\mathbf{U}_1^k\|_2,\dots,\|\mathbf{u}(T_N)-\mathbf{U}_N^k\|_2\big)^T,$$

and the spectral radius of $\tilde{S}$ is bounded by $\rho(\tilde{S})<x_l<1, l\ge1$, where

$$x_l=(|\sigma|x_{l-1}+|\kappa|)^{\frac{N}{N+1}},\quad x_0=1.$$

Proof. Let

$$\mathbf{u}(T_n)=F\big(T_n,T_{n-1},\mathbf{u}(T_{n-1})\big)$$

be the exact solution of the equation. We estimate the error of the Diag PP-IC algorithm (2.3) at iteration step $k+1$,

$$\mathbf{u}(T_n)-\mathbf{U}_n^{k+1}=F\big(T_n,T_{n-1},\mathbf{u}(T_{n-1})\big)-F\big(T_n,T_{n-1},\mathbf{U}_{n-1}^k\big)$$

$$\begin{aligned}
&- F^*\big(\alpha, T_n, T_{n-1}, \mathbf{U}_{n-1}^{k+1}\big) + F^*\big(\alpha, T_n, T_{n-1}, \mathbf{U}_{n-1}^{k}\big)\\
&= F\big(T_n, T_{n-1}, \mathbf{u}(T_{n-1})\big) - F^*\big(\alpha, T_n, T_{n-1}, \mathbf{u}(T_{n-1})\big)\\
&\quad - \big(F\big(T_n, T_{n-1}, \mathbf{U}_{n-1}^{k}\big) - F^*\big(\alpha, T_n, T_{n-1}, \mathbf{U}_{n-1}^{k}\big)\big)\\
&\quad + F^*\big(\alpha, T_n, T_{n-1}, \mathbf{u}(T_{n-1})\big) - F^*\big(\alpha, T_n, T_{n-1}, \mathbf{U}_{n-1}^{k+1}\big).
\end{aligned}$$

Under Assumptions 2.1-2.3, and using Lemmas 2.5, 2.6, we have

$$\begin{aligned}
e_n^{k+1} &:= \big\|\mathbf{u}(T_n) - \mathbf{U}_n^{k+1}\big\|_2\\
&\le \big\|F\big(T_n, T_{n-1}, \mathbf{u}(T_{n-1})\big) - F^*\big(\alpha, T_n, T_{n-1}, \mathbf{u}(T_{n-1})\big)\\
&\qquad - \big(F\big(T_n, T_{n-1}, \mathbf{U}_{n-1}^{k}\big) - F^*\big(\alpha, T_n, T_{n-1}, \mathbf{U}_{n-1}^{k}\big)\big)\big\|_2\\
&\quad + \big\|F^*\big(\alpha, T_n, T_{n-1}, \mathbf{u}(T_{n-1})\big) - F^*\big(\alpha, T_n, T_{n-1}, \mathbf{U}_{n-1}^{k+1}\big)\big\|_2\\
&\le \alpha\frac{L_3}{L_2-(1-\alpha)L_1}\big(e^{-L_1\Delta T} - e^{-\frac{L_2}{1-\alpha}\Delta T}\big)\big\|\mathbf{u}(T_{n-1}) - \mathbf{U}_{n-1}^{k}\big\|_2\\
&\quad + \frac{(1-\alpha)e^{-L_1\Delta T}}{1-\alpha e^{-L_1\Delta T}}\big\|\mathbf{u}(T_{n-1}) - \mathbf{U}_{n-1}^{k+1}\big\|_2.
\end{aligned}$$

Then we can get the following error bound:

$$\begin{aligned}
e_n^{k+1} &\le \alpha\frac{L_3}{L_2-(1-\alpha)L_1}\big(e^{-L_1\Delta T} - e^{-\frac{L_2}{1-\alpha}\Delta T}\big)e_{n-1}^{k} + \frac{(1-\alpha)e^{-L_1\Delta T}}{1-\alpha e^{-L_1\Delta T}}e_{n-1}^{k+1}\\
&:= \kappa e_{n-1}^{k} + \sigma e_{n-1}^{k+1}, \quad n = 1, 2, \dots, N,
\end{aligned} \tag{2.24}$$

and $e_0^{k+1} = e_N^k$. In matrix form, (2.24) can be written as

$$\underbrace{\begin{pmatrix} 1 & & & \\ -\sigma & 1 & & \\ & \ddots & \ddots & \\ & & -\sigma & 1 \end{pmatrix}}_{=:M_1}\underbrace{\begin{pmatrix} e_0^{k+1}\\ e_1^{k+1}\\ \vdots\\ e_N^{k+1}\end{pmatrix}}_{=:\mathbf{e}^{k+1}} \le \underbrace{\begin{pmatrix} 0 & & & 1\\ \kappa & 0 & & \\ & \ddots & \ddots & \\ & & \kappa & 0 \end{pmatrix}}_{=:N_1}\underbrace{\begin{pmatrix} e_0^{k}\\ e_1^{k}\\ \vdots\\ e_N^{k}\end{pmatrix}}_{=:\mathbf{e}^{k}}, \tag{2.25}$$

i.e.

$$M_1\mathbf{e}^{k+1} \le N_1\mathbf{e}^k. \tag{2.26}$$

Using the property of M-matrices [56, Theorem 3.18, Definition 3.22], we have

$$\mathbf{e}^{k+1} \le M_1^{-1}N_1\mathbf{e}^k, \tag{2.27}$$

and the proof can be completed by using Lemma 2.3. □

Theorem 2.3. *Under Assumption* 2.1*, and the condition* $|\sigma| + |\kappa| < 1$*, where*

$$\sigma = \frac{(1-\alpha)e^{-L_1\Delta T}}{1-\alpha e^{-L_1\Delta T}}, \quad \kappa = \left(1 + \frac{1-\alpha}{1-\alpha e^{-L_1\Delta T}}\right)e^{-L_1\Delta T},$$

then the errors of the Diag PP-IC algorithm (2.3) *are dominated by a linear iteration* $\tilde{S}$*, i.e.* $\mathbf{e}^{k+1} \le \tilde{S}\mathbf{e}^k$ *holds componentwise for*

$$\mathbf{e}^k := \big(\big\|\mathbf{u}(T_0) - \mathbf{U}_0^k\big\|_2, \big\|\mathbf{u}(T_1) - \mathbf{U}_1^k\big\|_2, \dots, \big\|\mathbf{u}(T_N) - \mathbf{U}_N^k\big\|_2\big)^T,$$

and the spectral radius of $\tilde{S}$ *is bounded by* $\rho(\tilde{S}) < x_l < 1$*,* $l \ge 1$*, where*

$$x_l = (|\sigma|x_{l-1} + |\kappa|)^{\frac{N}{N+1}}, \quad x_0 = 1.$$

Proof. Similar to the proof in Theorem 2.2, we obtain

$$\begin{aligned}\mathbf{u}(T_n) - \mathbf{U}_n^{k+1} &= F\big(T_n, T_{n-1}, \mathbf{u}(T_{n-1})\big) - F\big(T_n, T_{n-1}, \mathbf{U}_{n-1}^k\big)\\ &\quad - F^*\big(\alpha, T_n, T_{n-1}, \mathbf{u}(T_{n-1})\big) + F^*\big(\alpha, T_n, T_{n-1}, \mathbf{U}_{n-1}^k\big)\\ &\quad + F^*\big(\alpha, T_n, T_{n-1}, \mathbf{u}(T_{n-1})\big) - F^*\big(\alpha, T_n, T_{n-1}, \mathbf{U}_{n-1}^{k+1}\big).\end{aligned}$$

Under Assumption 2.1, and using Lemmas 2.4, 2.5, we obtain

$$\begin{aligned}e_n^{k+1} &:= \big\|\mathbf{u}(T_n) - \mathbf{U}_n^{k+1}\big\|_2\\ &\le \big\|F\big(T_n, T_{n-1}, \mathbf{u}(T_{n-1})\big) - F\big(T_n, T_{n-1}, \mathbf{U}_{n-1}^k\big)\big\|_2\\ &\quad + \big\|F^*\big(\alpha, T_n, T_{n-1}, \mathbf{u}(T_{n-1})\big) - F^*\big(\alpha, T_n, T_{n-1}, \mathbf{U}_{n-1}^k\big)\big\|_2\\ &\quad + \big\|F^*\big(\alpha, T_n, T_{n-1}, \mathbf{u}(T_{n-1})\big) - F^*\big(\alpha, T_n, T_{n-1}, \mathbf{U}_{n-1}^{k+1}\big)\big\|_2\\ &\le e^{-L_1\Delta T}\big\|\mathbf{u}(T_{n-1}) - \mathbf{U}_{n-1}^k\big\|_2 + \frac{(1-\alpha)e^{-L_1\Delta T}}{1-\alpha e^{-L_1\Delta T}}\big\|\mathbf{u}(T_{n-1}) - \mathbf{U}_{n-1}^{k+1}\big\|_2\\ &\quad + \frac{(1-\alpha)e^{-L_1\Delta T}}{1-\alpha e^{-L_1\Delta T}}\big\|\mathbf{u}(T_{n-1}) - \mathbf{U}_{n-1}^k\big\|_2.\end{aligned}$$

Then we have

$$\begin{aligned}e_n^{k+1} &\le \left(1 + \frac{1-\alpha}{1-\alpha e^{-L_1\Delta T}}\right) e^{-L_1\Delta T} e_{n-1}^k + \frac{(1-\alpha)e^{-L_1\Delta T}}{1-\alpha e^{-L_1\Delta T}} e_{n-1}^{k+1}\\ &=: \kappa e_{n-1}^k + \sigma e_{n-1}^{k+1}, \quad n = 1, 2, \ldots, N,\end{aligned} \tag{2.28}$$

and $e_0^{k+1} = e_N^k$. Noting that the relation (2.28) is of the same form as the relation (2.24), so the remainder of the proof is similar to the proof of Theorem 2.2. □

3. Krylov Subspace Enhanced Diagonalization-based PP-IC Algorithm

In order to improve the convergence performance of the Diag PP-IC algorithm (2.3), in this section, we further construct the Krylov subspace enhanced diagonalization-based parareal algorithm for the time-periodic problem (1.1) by combining the Krylov subspace method from [29] with the Diag PP-IC algorithm (2.3).

We first define the space

$$\mathcal{S}^k = \operatorname{span}\big\{\mathbf{U}_n^l;\ 0 \le l \le k,\ 0 \le n \le N-1\big\}, \tag{3.1}$$

and replace in the PP-IC algorithm (2.1) the coarse propagation $G(T_{n+1}, T_n, \mathbf{Y})$ by $K^*(\alpha, T_{n+1}, T_n, \mathbf{Y})$, the new propagator K^* is defined by

$$K^*(\alpha, T_{n+1}, T_n, \mathbf{Y}) = F\big(T_{n+1}, T_n, P^k\mathbf{Y}\big) + F^*\big(\alpha, T_{n+1}, T_n, (I - P^k)\mathbf{Y}\big), \tag{3.2}$$

where P^k is the orthogonal projection onto the space $\mathcal{S}^k$ with

$$P^k = S^k(S^k)^T, \tag{3.3}$$

and S^k is the matrix composed of an orthogonal basis for the space $\mathcal{S}^k$ as columns. Introducing this into the PP-IC algorithm (2.1), we obtain the following Krylov subspace enhanced diagonalization-based parareal algorithm for time-periodic problems:

$$\begin{aligned}&\mathbf{U}_0^{k+1} = \mathbf{U}_N^k,\\ &\mathbf{U}_{n+1}^{k+1} = F\big(T_{n+1}, T_n, \mathbf{U}_n^k\big) + K^*\big(\alpha, T_{n+1}, T_n, \mathbf{U}_n^{k+1}\big) - K^*\big(\alpha, T_{n+1}, T_n, \mathbf{U}_n^k\big),\\ &n = 0, \ldots, N-1.\end{aligned} \tag{3.4}$$

We call the algorithm (3.4) the Krylov subspace enhanced diagonalization-based PP-IC algorithm (Krylov Diag PP-IC).

To clearly show the parallelism of algorithm (3.4), we next consider the linear ODEs (2.5). Since $\mathbf{U}_n^k \in \mathcal{S}^k$, and $P^k = S^k(S^k)^T$ is the orthogonal projection onto the space $\mathcal{S}^k$, so we can get $P^k\mathbf{U}_n^k = \mathbf{U}_n^k$. Then, we substitute (3.2) into (3.4) and simplify to obtain the following Krylov Diag PP-IC algorithm:

$$\begin{aligned}\mathbf{U}_0^{k+1} &= \mathbf{U}_N^k,\\ \mathbf{U}_{n+1}^{k+1} &= F\big(T_{n+1}, T_n, P^k\mathbf{U}_n^{k+1}\big) + F^*\big(\alpha, T_{n+1}, T_n, (I-P^k)\mathbf{U}_n^{k+1}\big)\\ &\quad - F^*(\alpha, T_{n+1}, T_n, \mathbf{0}), \quad n = 0, \ldots, N-1.\end{aligned} \tag{3.5}$$

In the linear case, the fine propagator F holds

$$\begin{aligned}&F(T_{n+1}, T_n, \alpha\mathbf{X} + \beta\mathbf{Y})\\ &= F^M(\alpha\mathbf{X} + \beta\mathbf{Y}) + F(T_{n+1}, T_n, \mathbf{0})\\ &= \alpha F^M\mathbf{X} + \beta F^M\mathbf{Y} + F(T_{n+1}, T_n, \mathbf{0})\\ &= \alpha\big(F(T_{n+1}, T_n, \mathbf{X}) - F(T_{n+1}, T_n, \mathbf{0})\big)\\ &\quad + \beta\big(F(T_{n+1}, T_n, \mathbf{Y}) - F(T_{n+1}, T_n, \mathbf{0})\big) + F(T_{n+1}, T_n, \mathbf{0}),\end{aligned} \tag{3.6}$$

where $\mathbf{X}, \mathbf{Y} \in \mathbb{R}^d, \alpha, \beta \in \mathbb{R}$. In the above expression, F is a matrix defined by the fine propagator. According to (3.6), the quantity $F(T_{n+1}, T_n, P^k\mathbf{U}_n^{k+1})$ can be computed as

$$\begin{aligned}&F\big(T_{n+1}, T_n, P^k\mathbf{U}_n^{k+1}\big)\\ &= F\big(T_{n+1}, T_n, S^k(S^k)^T\mathbf{U}_n^{k+1}\big)\\ &= \big[F\big(T_{n+1}, T_n, S^k\big) - F(T_{n+1}, T_n, \mathbf{0})\big](S^k)^T\mathbf{U}_n^{k+1}\\ &\quad + F(T_{n+1}, T_n, \mathbf{0}),\end{aligned} \tag{3.7}$$

where $F(T_{n+1}, T_n, S^k)$ is the matrix having as columns the evolution of the column vectors of the matrix S^k by the fine propagator F, and $F(T_{n+1}, T_n, \mathbf{0})$ is the matrix having as columns the vector $F(T_{n+1}, T_n, \mathbf{0})$ and the same dimension as the matrix S^k. Here the matrix $F(T_{n+1}, T_n, S^k)$ is known by computing in parallel the evolution of $\mathbf{U}_n^k$ using the fine propagator F, i.e.

$$F\big(T_{n+1}, T_n, S^k\big) = [F(T_{n+1}, T_n, U) - F(T_{n+1}, T_n, \mathbf{0})] \cdot R^{-1} + F(T_{n+1}, T_n, \mathbf{0}), \tag{3.8}$$

where U is the matrix of all the generators of $\mathcal{S}^k$, and we compute the QR factorization of U ($U = QR$). Therefore, we describe the update steps of the Krylov Diag PP-IC algorithm (3.5) as follows:

Step 0: The algorithm starts with an initial approximations $\mathbf{U}_n^0, n = 0, \ldots, N$, which can be found for example solving a coupled system on the coarse grid by the coarse propagator G, i.e.

$$\mathbf{U}_{n+1}^0 = G\big(T_{n+1}, T_n, \mathbf{U}_n^0\big), \quad n = 0, \ldots, N-1, \quad \mathbf{U}_0^0 = \mathbf{U}_N^0.$$

The initial subspace $\mathcal{S}^{-1}$ is empty. In addition, we compute the quantities $F(T_{n+1}, T_n, \mathbf{0})$ and $F^*(\alpha, T_{n+1}, T_n, \mathbf{0})$, which are used to perform Step $k+1$.

Step $k+1$:

- We enhance the subspace by

$$\mathcal{S}^k = \operatorname{span}\big(\mathcal{S}^{k-1} \cup \big\{\mathbf{U}_n^k, n = 0, \ldots, N-1\big\}\big), \tag{3.9}$$

  where all initial conditions $\mathbf{U}_n^k$ are known. Then we construct an orthogonal basis for the space $\mathcal{S}^k$, whose columns we store in the matrix S^k, and the associated orthogonal projector P^k using (3.3).
- Compute in parallel $F(T_{n+1}, T_n, \mathbf{U}_n^k)$.
- We sequentially update the approximate solution $\mathbf{U}_n^{k+1}$ by (3.5), where $F^*(\alpha, T_{n+1}, T_n, (I-P^k)\mathbf{U}_n^{k+1})$ can be computed in parallel using (2.8) or (2.23).

Before we reach the convergence result of the Krylov Diag PP-IC algorithm (3.4), we need to derive two technical lemmas.

Lemma 3.1. *Under Assumption* 2.1*, the propagator* K^* *satisfies*

$$\|K^*(\alpha, T_{n+1}, T_n, \mathbf{x}) - K^*(\alpha, T_{n+1}, T_n, \mathbf{y})\|_2 \le \left(1 + \frac{2-2\alpha}{1-\alpha e^{-L_1\Delta T}}\right) e^{-L_1\Delta T} \|\mathbf{x}-\mathbf{y}\|_2.$$

Proof. Under Assumption 2.1, and using Lemmas 2.4, 2.5, we get

$$\begin{aligned}
&\|K^*(\alpha, T_{n+1}, T_n, \mathbf{x}) - K^*(\alpha, T_{n+1}, T_n, \mathbf{y})\|_2 \\
&= \big\|F\big(T_{n+1}, T_n, P^k\mathbf{x}\big) + F^*\big(\alpha, T_{n+1}, T_n, (I-P^k)\mathbf{x}\big) \\
&\quad - F\big(T_{n+1}, T_n, P^k\mathbf{y}\big) - F^*\big(\alpha, T_{n+1}, T_n, (I-P^k)\mathbf{y}\big)\big\|_2 \\
&\le \big\|F\big(T_{n+1}, T_n, P^k\mathbf{x}\big) - F\big(T_{n+1}, T_n, P^k\mathbf{y}\big)\big\|_2 \\
&\quad + \big\|F^*\big(\alpha, T_{n+1}, T_n, (I-P^k)\mathbf{x}\big) - F^*\big(\alpha, T_{n+1}, T_n, (I-P^k)\mathbf{y}\big)\big\|_2 \\
&\le e^{-L_1\Delta T}\|P^k(\mathbf{x}-\mathbf{y})\|_2 + \frac{(1-\alpha)e^{-L_1\Delta T}}{1-\alpha e^{-L_1\Delta T}}\|(I-P^k)(\mathbf{x}-\mathbf{y})\|_2 \\
&\le \left(1 + \frac{2-2\alpha}{1-\alpha e^{-L_1\Delta T}}\right) e^{-L_1\Delta T}\|\mathbf{x}-\mathbf{y}\|_2.
\end{aligned}$$

The proof is complete. □

Lemma 3.2. *Under Assumptions* 2.1-2.3*, the relationship between the propagator* K^* *and the exact propagator* F *satisfies*

$$\begin{aligned}
&\big\|F(T_{n+1}, T_n, \mathbf{x}) - K^*(\alpha, T_{n+1}, T_n, \mathbf{x}) - \big(F(T_{n+1}, T_n, \mathbf{y}) - K^*(\alpha, T_{n+1}, T_n, \mathbf{y})\big)\big\|_2 \\
&\le \left[4e^{-L_1\Delta T} + 2\alpha\frac{L_3}{L_2-(1-\alpha)L_1}\big(e^{-L_1\Delta T} - e^{-\frac{L_2}{1-\alpha}\Delta T}\big)\right]\|\mathbf{x}-\mathbf{y}\|_2.
\end{aligned}$$

Proof. Under Assumptions 2.1-2.3, and using Lemmas 2.4, 2.6, we get

$$\begin{aligned}
&\big\|F(T_{n+1}, T_n, \mathbf{x}) - K^*(\alpha, T_{n+1}, T_n, \mathbf{x}) - \big(F(T_{n+1}, T_n, \mathbf{y}) - K^*(\alpha, T_{n+1}, T_n, \mathbf{y})\big)\big\|_2 \\
&= \big\|F(T_{n+1}, T_n, \mathbf{x}) - F\big(T_{n+1}, T_n, P^k\mathbf{x}\big) - F^*\big(\alpha, T_{n+1}, T_n, (I-P^k)\mathbf{x}\big) \\
&\quad - \big(F(T_{n+1}, T_n, \mathbf{y}) - F\big(T_{n+1}, T_n, P^k\mathbf{y}\big) - F^*\big(\alpha, T_{n+1}, T_n, (I-P^k)\mathbf{y}\big)\big)\big\|_2 \\
&= \big\|F(T_{n+1}, T_n, \mathbf{x}) - F\big(T_{n+1}, T_n, P^k\mathbf{x}\big) - F\big(\alpha, T_{n+1}, T_n, (I-P^k)\mathbf{x}\big)
\end{aligned}$$

$$\begin{aligned}
&\quad + F\big(\alpha, T_{n+1}, T_n, (I-P^k)\mathbf{x}\big) - F^*\big(\alpha, T_{n+1}, T_n, (I-P^k)\mathbf{x}\big)\\
&\quad - \big(F\big(\alpha, T_{n+1}, T_n, (I-P^k)\mathbf{y}\big) - F^*\big(\alpha, T_{n+1}, T_n, (I-P^k)\mathbf{y}\big)\big)\\
&\quad - F(T_{n+1}, T_n, \mathbf{y}) + F\big(T_{n+1}, T_n, P^k\mathbf{y}\big) + F\big(\alpha, T_{n+1}, T_n, (I-P^k)\mathbf{y}\big)\big\|_2\\
&\le \big\|F\big(\alpha, T_{n+1}, T_n, (I-P^k)\mathbf{x}\big) - F^*\big(\alpha, T_{n+1}, T_n, (I-P^k)\mathbf{x}\big)\\
&\quad - \big(F\big(\alpha, T_{n+1}, T_n, (I-P^k)\mathbf{y}\big) - F^*\big(\alpha, T_{n+1}, T_n, (I-P^k)\mathbf{y}\big)\big)\big\|_2\\
&\quad + \|F(T_{n+1}, T_n, \mathbf{x}) - F(T_{n+1}, T_n, \mathbf{y})\|_2 + \big\|F\big(T_{n+1}, T_n, P^k\mathbf{x}\big) - F\big(T_{n+1}, T_n, P^k\mathbf{y}\big)\big\|_2\\
&\quad + \big\|F\big(\alpha, T_{n+1}, T_n, (I-P^k)\mathbf{x}\big) - F\big(\alpha, T_{n+1}, T_n, (I-P^k)\mathbf{y}\big)\big\|_2\\
&\le \alpha\frac{L_3}{L_2-(1-\alpha)L_1}\big(e^{-L_1\Delta T} - e^{-\frac{L_2}{1-\alpha}\Delta T}\big)\|(I-P^k)(\mathbf{x}-\mathbf{y})\|_2 + e^{-L_1\Delta T}\|\mathbf{x}-\mathbf{y}\|_2\\
&\quad + e^{-L_1\Delta T}\|P^k(\mathbf{x}-\mathbf{y})\|_2 + e^{-L_1\Delta T}\|(I-P^k)(\mathbf{x}-\mathbf{y})\|_2\\
&\le \left[4e^{-L_1\Delta T} + 2\alpha\frac{L_3}{L_2-(1-\alpha)L_1}\big(e^{-L_1\Delta T} - e^{-\frac{L_2}{1-\alpha}\Delta T}\big)\right]\|\mathbf{x}-\mathbf{y}\|_2.
\end{aligned}$$

The proof is complete. □

Next, we present two different estimates on the convergence factor of the Krylov Diag PP-IC algorithm (3.4) under different assumptions. Our analysis shows that there are different convergence regimes, and our estimates are adapted to this.

Theorem 3.1. *Under Assumptions* 2.1-2.3*, and the condition* $|\sigma| + |\kappa| < 1$*, where*

$$\begin{aligned}
\sigma &= \left(1 + \frac{2-2\alpha}{1-\alpha e^{-L_1\Delta T}}\right)e^{-L_1\Delta T},\\
\kappa &= 4e^{-L_1\Delta T} + 2\alpha\frac{L_3}{L_2-(1-\alpha)L_1}\big(e^{-L_1\Delta T} - e^{-\frac{L_2}{1-\alpha}\Delta T}\big),
\end{aligned}$$

then the errors of the Krylov Diag PP-IC algorithm (3.4) *are dominated by a linear iteration* $\tilde{S}$*, i.e.* $\mathbf{e}^{k+1} \le \tilde{S}\mathbf{e}^k$ *holds componentwise for*

$$\mathbf{e}^k := \big(\big\|\mathbf{u}(T_0) - \mathbf{U}_0^k\big\|_2, \big\|\mathbf{u}(T_1) - \mathbf{U}_1^k\big\|_2, \ldots, \big\|\mathbf{u}(T_N) - \mathbf{U}_N^k\big\|_2\big)^T,$$

and the spectral radius of $\tilde{S}$ *is bounded by* $\rho(\tilde{S}) < x_l < 1$*,* $l \ge 1$*, where*

$$x_l = (|\sigma|x_{l-1} + |\kappa|)^{\frac{N}{N+1}}, \quad x_0 = 1.$$

Proof. Similar to the proof in Theorem 2.2, we estimate the error of the Krylov Diag PP-IC algorithm (3.4) at iteration step $k+1$,

$$\begin{aligned}
\mathbf{u}(T_n) - \mathbf{U}_n^{k+1} &= F\big(T_n, T_{n-1}, \mathbf{u}(T_{n-1})\big) - F\big(T_n, T_{n-1}, \mathbf{U}_{n-1}^k\big)\\
&\quad - K^*\big(\alpha, T_n, T_{n-1}, \mathbf{U}_{n-1}^{k+1}\big) + K^*\big(\alpha, T_n, T_{n-1}, \mathbf{U}_{n-1}^k\big)\\
&= F\big(T_n, T_{n-1}, \mathbf{u}(T_{n-1})\big) - K^*\big(\alpha, T_n, T_{n-1}, \mathbf{u}(T_{n-1})\big)\\
&\quad - \big(F\big(T_n, T_{n-1}, \mathbf{U}_{n-1}^k\big) - K^*\big(\alpha, T_n, T_{n-1}, \mathbf{U}_{n-1}^k\big)\big)\\
&\quad + K^*\big(\alpha, T_n, T_{n-1}, \mathbf{u}(T_{n-1})\big) - K^*\big(\alpha, T_n, T_{n-1}, \mathbf{U}_{n-1}^{k+1}\big).
\end{aligned}$$

Under Assumptions 2.1-2.3, and using Lemmas 3.1, 3.2, we obtain

$$e_n^{k+1} := \big\|\mathbf{u}(T_n) - \mathbf{U}_n^{k+1}\big\|_2$$

$$
\begin{aligned}
&\leq \big\|F\big(T_n,T_{n-1},\mathbf{u}(T_{n-1})\big)-K^*\big(\alpha,T_n,T_{n-1},\mathbf{u}(T_{n-1})\big)\\
&\quad -\big(F\big(T_n,T_{n-1},\mathbf{U}^k_{n-1}\big)-K^*\big(\alpha,T_n,T_{n-1},\mathbf{U}^k_{n-1}\big)\big)\big\|_2\\
&\quad +\big\|K^*\big(\alpha,T_n,T_{n-1},\mathbf{u}(T_{n-1})\big)-K^*\big(\alpha,T_n,T_{n-1},\mathbf{U}^{k+1}_{n-1}\big)\big\|_2\\
&\leq \left[4e^{-L_1\Delta T}+2\alpha\frac{L_3}{L_2-(1-\alpha)L_1}\big(e^{-L_1\Delta T}-e^{-\frac{L_2}{1-\alpha}\Delta T}\big)\right]\big\|\mathbf{u}(T_{n-1})-\mathbf{U}^k_{n-1}\big\|_2\\
&\quad +\left(1+\frac{2-2\alpha}{1-\alpha e^{-L_1\Delta T}}\right)e^{-L_1\Delta T}\big\|\mathbf{u}(T_{n-1})-\mathbf{U}^{k+1}_{n-1}\big\|_2.
\end{aligned}
$$

Then we have

$$
\begin{aligned}
e^{k+1}_n &\leq \left[4e^{-L_1\Delta T}+2\alpha\frac{L_3}{L_2-(1-\alpha)L_1}\big(e^{-L_1\Delta T}-e^{-\frac{L_2}{1-\alpha}\Delta T}\big)\right]e^k_{n-1}\\
&\quad +\left(1+\frac{2-2\alpha}{1-\alpha e^{-L_1\Delta T}}\right)e^{-L_1\Delta T}e^{k+1}_{n-1}\\
&=: \kappa e^k_{n-1}+\sigma e^{k+1}_{n-1},\quad n=1,2,\ldots,N, \qquad (3.10)
\end{aligned}
$$

and $e^{k+1}_0=e^k_N$. We see again that (3.10) is of the same form as (2.24), so the remainder of the proof is similar to the proof of Theorem 2.2. □

Theorem 3.2. *Under Assumption* 2.1, *and the condition* $|\sigma|+|\kappa|<1$, *where*

$$
\sigma=\left(1+\frac{2-2\alpha}{1-\alpha e^{-L_1\Delta T}}\right)e^{-L_1\Delta T},\quad \kappa=\left(2+\frac{2-2\alpha}{1-\alpha e^{-L_1\Delta T}}\right)e^{-L_1\Delta T},
$$

then the errors of the Krylov Diag PP-IC algorithm (3.4) *are dominated by a linear iteration* $\tilde{S}$, *i.e.* $\mathbf{e}^{k+1}\leq\tilde{S}\mathbf{e}^k$ *holds componentwise for*

$$
\mathbf{e}^k:=\big(\|\mathbf{u}(T_0)-\mathbf{U}^k_0\|_2,\|\mathbf{u}(T_1)-\mathbf{U}^k_1\|_2,\ldots,\|\mathbf{u}(T_N)-\mathbf{U}^k_N\|_2\big)^T,
$$

and the spectral radius of $\tilde{S}$ *is bounded by* $\rho(\tilde{S})<x_l<1$, $l\geq 1$, *where*

$$
x_l=(|\sigma|x_{l-1}+|\kappa|)^{\frac{N}{N+1}},\quad x_0=1.
$$

Proof. Similar to the proof in Theorem 2.2, we have

$$
\begin{aligned}
\mathbf{u}(T_n)-\mathbf{U}^{k+1}_n &= F\big(T_n,T_{n-1},\mathbf{u}(T_{n-1})\big)-F\big(T_n,T_{n-1},\mathbf{U}^k_{n-1}\big)\\
&\quad -K^*\big(\alpha,T_n,T_{n-1},\mathbf{U}^{k+1}_{n-1}\big)+K^*\big(\alpha,T_n,T_{n-1},\mathbf{u}(T_{n-1})\big)\\
&\quad +K^*\big(\alpha,T_n,T_{n-1},\mathbf{U}^k_{n-1}\big)-K^*\big(\alpha,T_n,T_{n-1},\mathbf{u}(T_{n-1})\big).
\end{aligned}
$$

Under Assumption 2.1, and using Lemmas 2.4, 3.1, we obtain

$$
\begin{aligned}
e^{k+1}_n &:= \big\|\mathbf{u}(T_n)-\mathbf{U}^{k+1}_n\big\|_2\\
&\leq \big\|F\big(T_n,T_{n-1},\mathbf{u}(T_{n-1})\big)-F\big(T_n,T_{n-1},\mathbf{U}^k_{n-1}\big)\big\|_2\\
&\quad +\big\|K^*\big(\alpha,T_n,T_{n-1},\mathbf{u}(T_{n-1})\big)-K^*\big(\alpha,T_n,T_{n-1},\mathbf{U}^{k+1}_{n-1}\big)\big\|_2\\
&\quad +\big\|K^*\big(\alpha,T_n,T_{n-1},\mathbf{u}(T_{n-1})\big)-K^*\big(\alpha,T_n,T_{n-1},\mathbf{U}^k_{n-1}\big)\big\|_2\\
&\leq e^{-L_1\Delta T}\big\|\mathbf{u}(T_{n-1})-\mathbf{U}^k_{n-1}\big\|_2\\
&\quad +\left(1+\frac{2-2\alpha}{1-\alpha e^{-L_1\Delta T}}\right)e^{-L_1\Delta T}\big\|\mathbf{u}(T_{n-1})-\mathbf{U}^{k+1}_{n-1}\big\|_2\\
&\quad +\left(1+\frac{2-2\alpha}{1-\alpha e^{-L_1\Delta T}}\right)e^{-L_1\Delta T}\big\|\mathbf{u}(T_{n-1})-\mathbf{U}^k_{n-1}\big\|_2.
\end{aligned}
$$

Then we have

$$\begin{aligned} e_n^{k+1} &\le \left(2+\frac{2-2\alpha}{1-\alpha e^{-L_1\Delta T}}\right)e^{-L_1\Delta T}e_{n-1}^k+\left(1+\frac{2-2\alpha}{1-\alpha e^{-L_1\Delta T}}\right)e^{-L_1\Delta T}e_{n-1}^{k+1} \\ &=: \kappa e_{n-1}^k+\sigma e_{n-1}^{k+1}, \quad n=1,2,\ldots,N, \end{aligned} \tag{3.11}$$

and $e_0^{k+1}=e_N^k$. Noting again that (3.11) is of the same form as (2.24), so the remainder of the proof is similar to the proof of Theorem 2.2. □

4. Parallel Speedup

Similar to the analysis of the parallel speedup in [40,47,48], we first introduce the following notations:

- T is the length of the time interval.
- N is the number of time subintervals.
- ΔT is the length of each time subinterval and the time increment for the coarse propagator with $\Delta T=T/N$.
- Δt is the time increment for the fine propagator with $\Delta t=\Delta T/M$.
- k_1 is the iteration number of the classical PP-IC algorithm.
- k_2 is the iteration number of the Diag PP-IC algorithm.
- k_3 is the iteration number of the Krylov Diag PP-IC algorithm.

Denote by τ the time cost for solving an initial value problem at one time step of the coarse propagator and the fine propagator, by τ^* the time cost solving a head-tail coupled problem on each time subinterval for the F^* propagator, and by $\tau_{\mathrm{QR}}(k)$ the time cost of computing the subspace update in iteration k. Furthermore, we also denote by τ_0 the time cost solving a time-periodic problem on the coarse grid for the initial guess, and by $\tau_{\Delta t}$ the time cost of a direct method for the original time-periodic problem with the time increment Δt.

According to the analysis in [48], the total time computation cost of the classical PP-IC algorithm is

$$C_{\mathrm{PP-IC}}=\tau_0+\tau+k_1(M\tau+N\tau)=\tau_0+k_1M\tau+(k_1N+1)\tau,$$

then the parallel speedup is

$$S_{\mathrm{PP-IC}}=\frac{\tau_{\Delta t}}{\tau_0+k_1M\tau+(k_1N+1)\tau}.$$

Analogously, the total time computation cost of the Diag PP-IC algorithm is

$$C_{\mathrm{Diag}}=\tau_0+\tau^*+k_2(M\tau+N\tau^*)=\tau_0+k_2M\tau+(k_2N+1)\tau^*,$$

and the parallel speedup is

$$S_{\mathrm{Diag}}=\frac{\tau_{\Delta t}}{\tau_0+k_2M\tau+(k_2N+1)\tau^*}.$$

Since the dimension of $\mathcal{S}^k$ increases monotonically with the number of iterations performed, i.e. $\tau_{\mathrm{QR}}(k) \leq \tau_{\mathrm{QR}}(k_3)$. Therefore, the time cost of the Krylov Diag PP-IC algorithm can be estimated by

$$\begin{aligned} C_{\mathrm{Krylov}} &= \tau_0 + \tau^* + k_3(M\tau + N\tau^*) + k_3\tau_{\mathrm{QR}}(k_3) \\ &= \tau_0 + k_3 M\tau + (k_3 N + 1)\tau^* + k_3\tau_{\mathrm{QR}}(k_3), \end{aligned}$$

then the speedup obtainable in one parallel step is

$$S_{\mathrm{Krylov}} = \frac{\tau_{\Delta t}}{\tau_0 + k_3 M\tau + (k_3 N + 1)\tau^* + k_3\tau_{\mathrm{QR}}(k_3)}.$$

Here, we consider the linear time-periodic ODEs (2.5). We assume that the time complexity for solving a d-dimensional linear system is $\mathcal{O}(d^{\tilde{p}})$ with $\tilde{p} \geq 1$. In particular, if $\tilde{p} = 1$, so an efficient implementation should yield $\tau = \mathcal{O}(d)$. The FFT can be implemented in parallel at the cost $\mathcal{O}(d \log_2 M)$, see [31], then we have $\tau^* = \mathcal{O}(d \log_2 M)$. Therefore, by ignoring the cost of computing the initial iteration value, the cost of the classical PP-IC algorithm is

$$C_{\mathrm{PP-IC}} = k_1 M d + (k_1 N + 1)d,$$

the cost of the Diag PP-IC algorithm is

$$C_{\mathrm{Diag}} = k_2 M d + (k_2 N + 1)d \log_2 M,$$

and from [47] neglecting the cost of the subspace update for the Krylov Diag PP-IC algorithm gives the cost

$$C_{\mathrm{Krylov}} = k_3 M d + (k_3 N + 1)d \log_2 M.$$

We next show more details about the parallel speedup and the time computation cost in the numerical experiments.

5. Numerical Experiments

In this section, we present several numerical experiments to illustrate convergence analysis of our new proposed algorithms. Here all the iteration errors are measured in the infinity norm between the iterative approximate solution produced by the algorithms at each iteration and the fine solution, and all the numerical convergence factors are denoted by $(e^k/e^0)^{1/k}$, where e^k is the iteration error at the k-th iteration.

5.1. The heat equation

We first consider the following time-periodic heat equation:

$$\begin{aligned} &\frac{\partial u(x,t)}{\partial t} = \frac{\partial^2 u(x,t)}{\partial x^2} + f(x,t), \quad (x,t) \in \Omega \times [0,T], \\ &u(x,0) = u(x,T) \end{aligned} \tag{5.1}$$

with homogeneous boundary conditions, where $f : \Omega \times [0,T] \to \mathbb{R}$ and satisfies $f(x,0) = f(x,T)$ for any $x \in \Omega$.

We solve problem (5.1) on the time interval $[0,1]$ and the spatial domain $\Omega = [0,1]$ with the nonlinear function

$$f(x,t) = 2t(1-t)(1-2t)x(x-1) - 2t^2(1-t)^2$$

such that the exact solution is

$$u_{\text{exact}} = x(x-1)t^2(1-t)^2.$$

The problem (5.1) is discretized in space by using a second-order centered finite difference scheme with a mesh size $\Delta x = 1/128$. Then we get the linear system of ODEs

$$\frac{d\mathbf{u}(t)}{dt} = A\mathbf{u}(t) + \mathbf{f}(t), \quad A := \frac{1}{\Delta x^2}\begin{bmatrix} -2 & 1 & & & \\ 1 & -2 & 1 & & \\ & \ddots & \ddots & \ddots & \\ & & 1 & -2 & 1 \\ & & & 1 & -2 \end{bmatrix}, \quad t \in [0,1].$$

Here we use for both the F and F^* propagators backward Euler with the same time step-size $\Delta t = \Delta T/M$ on each subinterval $[T_n, T_{n+1}]$.

We first show in Fig. 5.1 the measured iteration number of two new algorithms when the parameter $\alpha \in (0,1)$ varies, where we stop the iterations when the iteration errors decrease to 10^{-8}. We see that the convergence rate of the Diag PP-IC algorithm is quite sensitive to the change in α, while for the Krylov Diag PP-IC algorithm the iteration number remains constant when α varies. Thus we will choose the relatively good parameter $\alpha = 0.3$ from Fig. 5.1 in order to converge faster in the following. Furthermore, we also show in Fig. 5.2 the convergence of two algorithms for four values of α, as observed in Fig. 5.1.

We next show the convergence results in Fig. 5.3. From the figure on the left, we can see that both the Diag PP-IC and Krylov Diag PP-IC algorithms converge faster than the PP-IC algorithm. Meanwhile, the performance of the Krylov Diag PP-IC algorithm is better than that of the Diag PP-IC algorithm. Then we compare the numerical convergence factors of two new algorithms with the theoretical bounds from Theorems 2.1, 3.1, 3.2 for the linear case on the right in Fig. 5.3, where “Bound of Diag PP-IC” is from Theorem 2.1, “Bound 1 of Krylov Diag PP-IC” is from Theorem 3.1, and “Bound 2 of Krylov Diag PP-IC” is from Theorem 3.2. Here we choose the constants $-L_1, -L_2 \in \sigma(A)$ are arbitrary eigenvalues of A, $L_3 = \|A\|_2$ in Theorems 3.1, 3.2, which satisfy Assumptions 2.1-2.3. Therefore, we have given explicitly the estimates of the constants $L_1 = 3.61, L_2 = 3.98, L_3 = 6.57$.

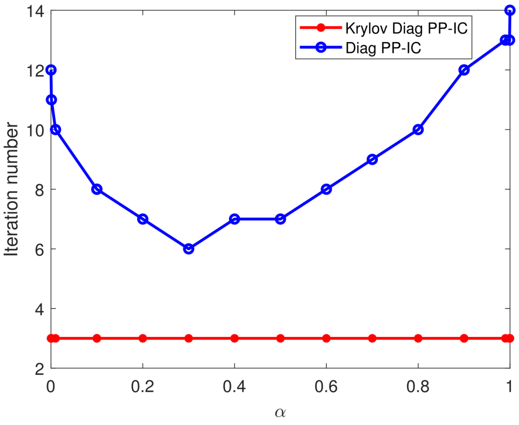

Fig. 5.1. Dependence of the convergence rate of two new algorithms on α for the heat problem (5.1) with $\Delta T = 1/8$, $\Delta t = 1/128$.

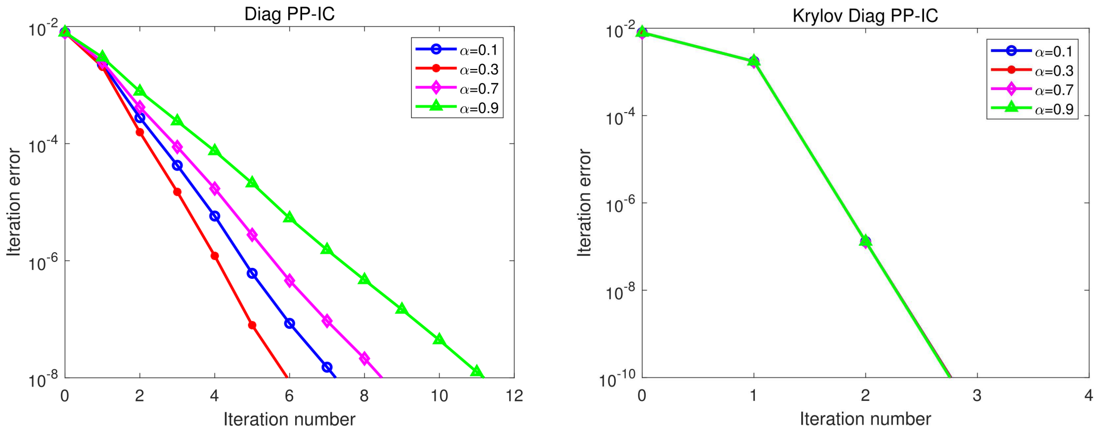

Fig. 5.2. Convergence of the Diag PP-IC (left) and Krylov Diag PP-IC (right) algorithms with various values of α for the heat problem (5.1) with $\Delta T = 1/8, \Delta t = 1/128$.

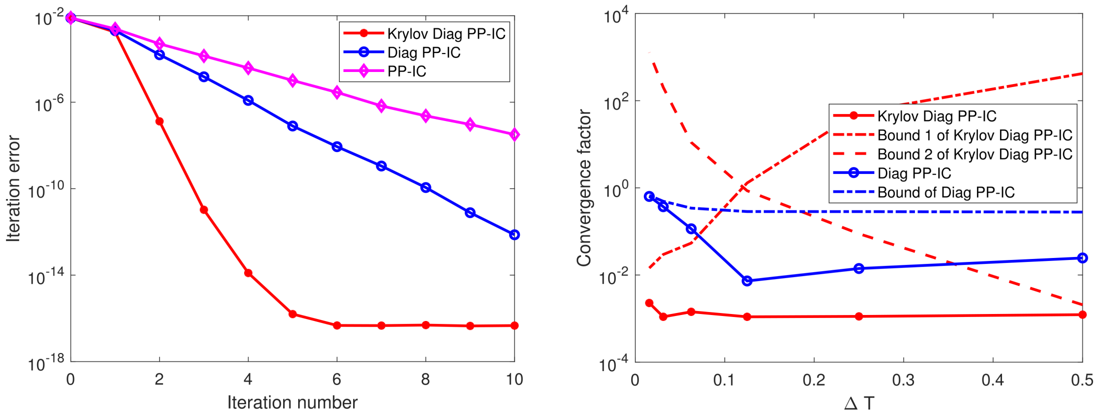

Fig. 5.3. Convergence of two new algorithms versus the PP-IC algorithm for the heat problem (5.1) with $\Delta T = 1/8, \Delta t = 1/128$ (left), and dependence of the convergence factor on ΔT with $\Delta t = 1/128$ (right).

Furthermore, we show the convergence behavior of two new algorithms and their theoretical upper bounds on errors in Fig. 5.4. We see that "Bound of Diag PP-IC" from Theorem 2.1 of the Diag PP-IC algorithm is very effective in different regimes on ΔT. For the Krylov Diag PP-IC algorithm, when $\Delta T = 1/16$ is small, $\sigma = 0.3232, \kappa = 0.4134$ that satisfy the condition $|\sigma|+|\kappa| < 1$ in Theorem 3.1, so "Bound 1 of Krylov Diag PP-IC" from Theorem 3.1 is effective, in contrast when $\Delta T = 1/4$ is large, $\sigma = 0.3459$, $\kappa = 0.3866$ that satisfy the condition $|\sigma| + |\kappa| < 1$ in Theorem 3.2, so "Bound 2 of Krylov Diag PP-IC" from Theorem 3.2 is effective.

According to the analysis of the parallel speedup in Section 4, the comparisons of the estimated speedup of the classical PP-IC, Diag PP-IC and Krylov Diag PP-IC algorithms for the heat problem (5.1) are shown in Table 5.1. From Table 5.1, the computational speed of two new algorithms has a significant improvement compared to the classical one. From the analysis of the time complexity for the heat problem (5.1) in Section 4 and noting that $N = 8, M = 16$ and $d = 127$ in this case, the cost of the classical PP-IC algorithm is $k_1 \times 16 \times 127 + (k_1 \times 8 + 1) \times 127$, the cost of the Diag PP-IC algorithm is $k_2 \times 16 \times 127 + (k_2 \times 8 + 1) \times 127 \times \log_2 16$, and that of the Krylov Diag PP-IC algorithm is $k_3 \times 16 \times 127 + (k_3 \times 8 + 1) \times 127 \times \log_2 16$. In Fig. 5.5, we can observe that the same computational cost is needed for the classical PP-IC and Diag PP-IC algorithms, and the Krylov Diag PP-IC algorithm needs much less computational cost than the Diag PP-IC algorithm to achieve the same accuracy.

Table 5.1: The estimated speedup of the classical PP-IC, Diag PP-IC, Krylov Diag PP-IC algorithms based on the sequential execution time t_s, the number of processors N, the iteration number k, the measured time of the F, G, F^* propagators t_F, t_G, t_{F^*} and the parallel execution time t_p.

Algorithm	t_s	N	k	Error	t_F	t_G or t_{F^*}	t_p	Speedup
PP-IC	$2.8623s$	8	26	10^{-15}	$0.3146s$	$0.2926s$	$0.6368s$	4.4948
Diag PP-IC	$2.8623s$	8	14	10^{-15}	$0.1694s$	$0.2486s$	$0.4476s$	6.3948
Krylov Diag PP-IC	$2.8623s$	8	5	10^{-15}	$0.0605s$	$0.0902s$	$0.1828s$	15.6581

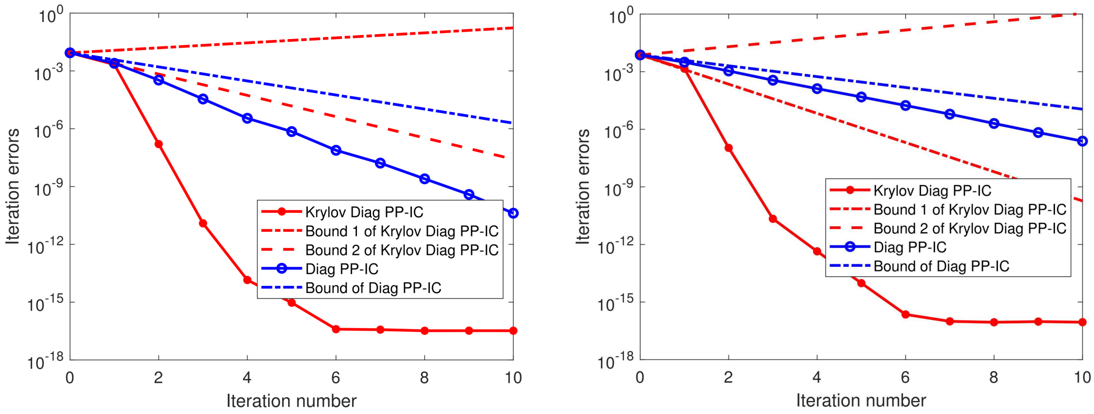

Fig. 5.4. Convergence behavior of two new algorithms for the heat problem (5.1): comparison of the numerical results and the theoretical bounds for $\Delta T = 1/4$ (left), $\Delta T = 1/16$ (right) with $\Delta t = 1/128$.

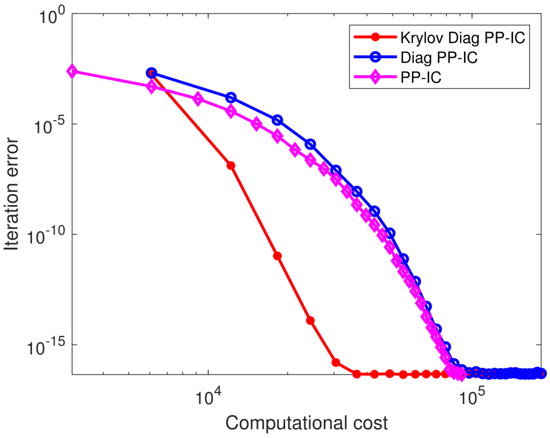

Fig. 5.5. Computational cost of two new algorithms versus the PP-IC algorithm for the heat problem (5.1) with $\Delta T = 1/8, \Delta t = 1/128$.

5.2. The wave equation

We next consider the time-periodic wave equation

$$\begin{aligned} &\frac{\partial^2 u(x,t)}{\partial t^2} = \frac{\partial^2 u(x,t)}{\partial x^2} + \beta u^2(x,t) + f(x,t), \quad (x,t) \in \Omega \times [0,T], \\ &u(x,0) = u(x,T), \\ &u_t(x,0) = u_t(x,T) \end{aligned} \tag{5.2}$$

with homogeneous boundary conditions, where $f : \Omega \times [0,T] \to \mathbb{R}$ and satisfies $f(x,0) = f(x,T)$ for any $x \in \Omega$.

We solve problem (5.2) on the time interval $[0,1]$ and the spatial domain $\Omega = [0,1]$ with the exact solution $u_{\text{exact}} = \sin(4\pi x)\sin(2\pi t)$. The problem (5.2) is discretized in space by using a second-order centered finite difference scheme with a mesh size $\Delta x = 1/64$. Then we get the following ODEs:

$$\frac{d}{dt}\begin{bmatrix} \mathbf{u}(t) \\ \mathbf{v}(t) \end{bmatrix} = \begin{bmatrix} O & I \\ A & O \end{bmatrix}\begin{bmatrix} \mathbf{u}(t) \\ \mathbf{v}(t) \end{bmatrix} + \begin{bmatrix} \mathbf{0} \\ \beta \mathbf{u}^2(t) + \mathbf{f}(t) \end{bmatrix}, \tag{5.3}$$

where

$$A := \frac{1}{\Delta x^2}\begin{bmatrix} -2 & 1 & & & \\ 1 & -2 & 1 & & \\ & \ddots & \ddots & \ddots & \\ & & 1 & -2 & 1 \\ & & & 1 & -2 \end{bmatrix}, \quad t \in [0,1].$$

Similarly, we use for both the F and F^* propagators backward Euler with the same time step-size $\Delta t = \Delta T/M$ on each subinterval $[T_n, T_{n+1}]$.

5.2.1. The linear case: $\beta = 0$

For the linear case $\beta = 0$, the nonlinear function is $f(x,t) = 12\pi^2 \sin(4\pi x)\sin(2\pi t)$. We show in Fig. 5.6 the iteration number of two new algorithms when the parameter $\alpha \in (0,1)$ varies,

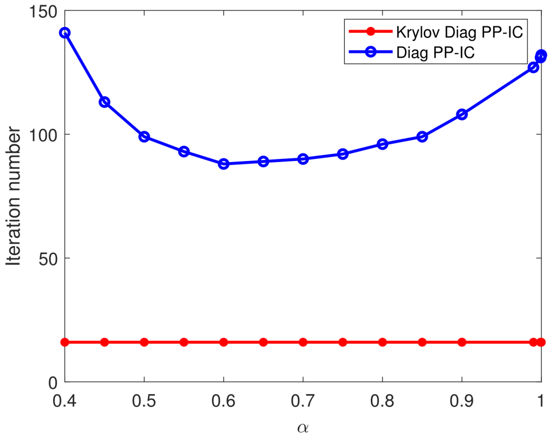

Fig. 5.6. Dependence of the convergence rate of two new algorithms on α for the linear wave problem (5.2) with $\Delta T = 1/8, \Delta t = 1/128$.

where we stop the iterations when the iteration errors decrease to 10^{-4}. We see again that the convergence rate of the Diag PP-IC algorithm is quite sensitive to the change in α, while for the Krylov Diag PP-IC algorithm the iteration number remains constant when α varies. Therefore, we will choose the relatively good parameter $\alpha = 0.6$ in the following. Furthermore, we also show in Fig. 5.7 the convergence rate of two algorithms for four values of α, as observed in Fig. 5.6.

We further show the convergence results in Fig. 5.8. We also observe that the convergence behavior of both the Diag PP-IC and Krylov Diag PP-IC algorithms is better than that of the PP-IC algorithm. Meanwhile, the Krylov Diag PP-IC algorithm converges faster than the Diag PP-IC algorithm.

Similarly, from the theoretical analysis in Section 4, the comparisons of the estimated speedup and the computational cost of the Diag PP-IC and Krylov Diag PP-IC algorithms for the linear wave problem (5.2) are shown in Table 5.2 and in Fig. 5.9 respectively, where

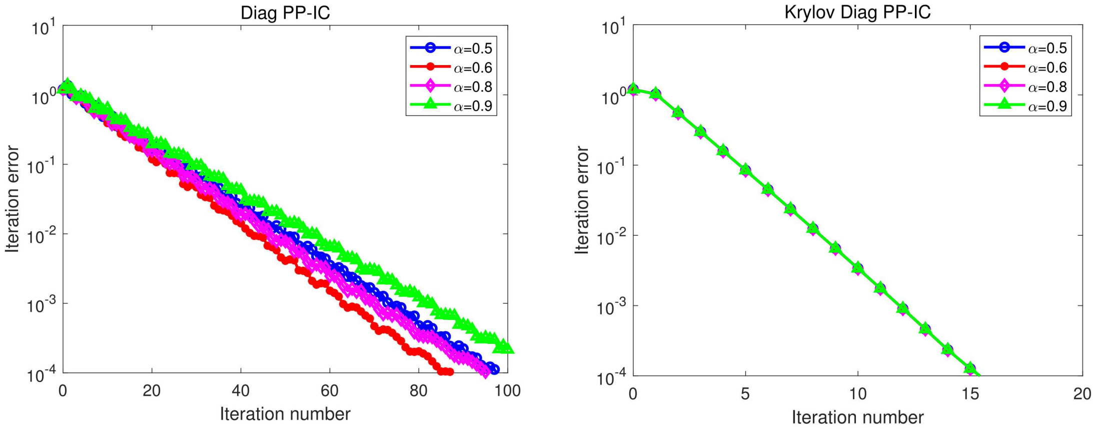

Fig. 5.7. Convergence of the Diag PP-IC (left) and Krylov Diag PP-IC (right) algorithms with various values of α for the linear wave problem (5.2) with $\Delta T = 1/8, \Delta t = 1/128$.

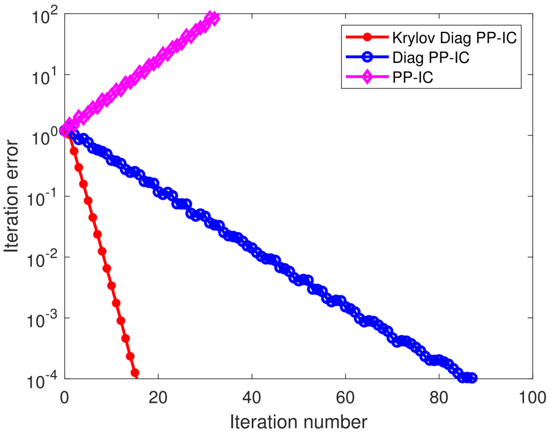

Fig. 5.8. Convergence of two new algorithms versus the PP-IC algorithm for the linear wave problem (5.2) with $\Delta T = 1/8, \Delta t = 1/128$ (left), and dependence of the convergence factor on ΔT with $\Delta t = 1/128$ (right).

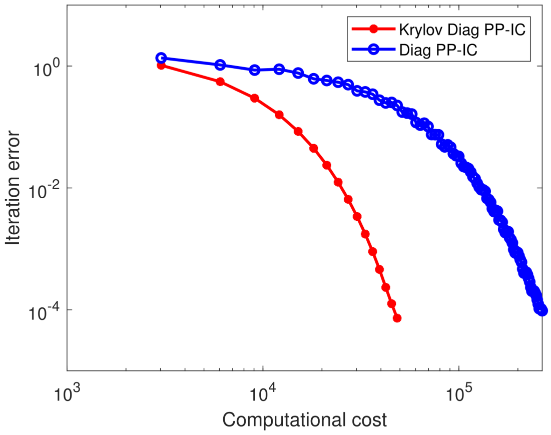

Fig. 5.9. Computational cost of two new algorithms for the linear wave problem (5.2) with $\Delta T = 1/8$, $\Delta t = 1/128$.

Table 5.2: The estimated speedup of the Diag PP-IC and Krylov Diag PP-IC algorithms based on the sequential execution time t_s, the number of processors N, the iteration number k, the measured time of the F, F^* propagators t_F, t_{F^*} and the parallel execution time t_p.

Algorithm	t_s	N	k	Error	t_F	t_{F^*}	t_p	Speedup
Diag PP-IC	$3.1429s$	8	88	10^{-4}	$1.0648s$	$1.5510s$	$2.6726s$	1.1759
Krylov Diag PP-IC	$3.1429s$	8	16	10^{-4}	$0.1936s$	$0.2838s$	$0.5422s$	5.7965

the classical PP-IC algorithm is not displayed since it does not converge. From Table 5.2, the computational speed of two new algorithms has a significant improvement. In Fig. 5.9, we can see that the Krylov Diag PP-IC algorithm takes much less computational cost than the Diag PP-IC algorithm to achieve a desired accuracy.

5.2.2. The nonlinear case: $\beta = 10$

For the nonlinear case $\beta = 10$, the nonlinear function is

$$f(x,t) = 12\pi^2 \sin(4\pi x)\sin(2\pi t) - 10\sin^2(4\pi x)\sin^2(2\pi t).$$

We show in Fig. 5.10 the iteration number of two new algorithms when the parameter $\alpha \in (0,1)$ varies, where we stop the iterations when the iteration errors decrease to 10^{-4}. We see again that the convergence rate of the Diag PP-IC algorithm is quite sensitive to the change in α, while for the Krylov Diag PP-IC algorithm the iteration number remains constant when α varies. Therefore, we will choose the relatively good parameter $\alpha = 0.55$ in the following. Furthermore, we also show in Fig. 5.11 the convergence rate of two algorithms for four values of α, as observed in Fig. 5.10.

We further show the convergence results in Fig. 5.12. From the figure on the left, we also observe that the convergence behavior of both the Diag PP-IC and Krylov Diag PP-IC algorithms is better than that of the PP-IC algorithm. Meanwhile, the Krylov Diag PP-IC algorithm converges faster than the Diag PP-IC algorithm. Then, we compare the numerical convergence factors of two new algorithms with the theoretical bounds from Theorems 2.2, 2.3,

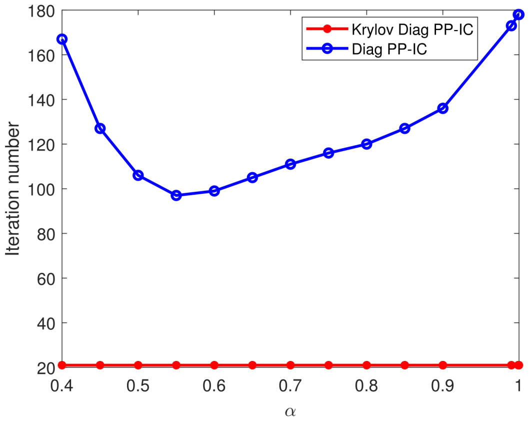

Fig. 5.10. Dependence of the convergence rate of two new algorithms on α for the nonlinear wave problem (5.2) with $\Delta T = 1/8, \Delta t = 1/128$.

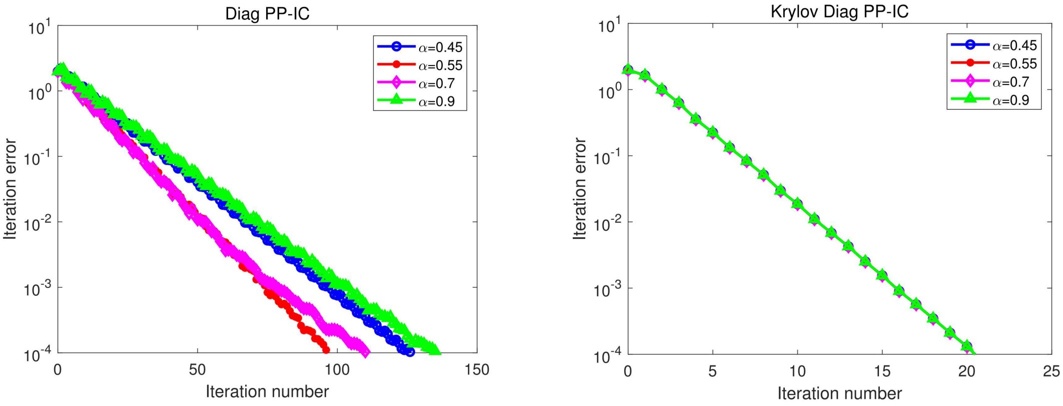

Fig. 5.11. Convergence of the Diag PP-IC (left) and Krylov Diag PP-IC (right) algorithms with various values of α for the nonlinear wave problem (5.2) with $\Delta T = 1/8, \Delta t = 1/128$.

3.1, 3.2 for the nonlinear case on the right in Fig. 5.12, where "Bound 1 of Diag PP-IC" is from Theorem 2.2, "Bound 2 of Diag PP-IC" is from Theorem 2.3, "Bound 1 of Krylov Diag PP-IC" is from Theorem 3.1, "Bound 2 of Krylov Diag PP-IC" is from Theorem 3.2. Here we choose the constants $-L_1, -L_2, L_3$ are from the Jacobian matrix of the right-side nonlinear function in (5.3) obtained by the space discretization in Theorems 2.2, 2.3, 3.1, 3.2, which satisfy Assumptions 2.1-2.3. Therefore, we have given explicitly the estimates of the constants $L_1 = 4.82, L_2 = 1.32, L_3 = 6.95$.

In addition, we also give the convergence behavior of two new algorithms and their theoretical upper bounds on errors in Fig. 5.13. We see that for the Diag PP-IC algorithm when $\Delta T = 1/16$ is small, $\sigma = 0.5626$, $\kappa = 0.4177$ that satisfy the condition $|\sigma| + |\kappa| < 1$ in Theorem 2.2, so "Bound 1 of Diag PP-IC" from Theorem 2.2 is effective, while when $\Delta T = 1/4$ is large, $\sigma = 0.5169, \kappa = 0.3793$ that satisfy the condition $|\sigma| + |\kappa| < 1$ in Theorem 2.3, so "Bound 2 of Diag PP-IC" from Theorem 2.3 is effective. Furthermore, for the Krylov Diag PP-IC algorithm, when $\Delta T = 1/16$ is small, $\sigma = 0.3144, \kappa = 0.5061$ that satisfy the condition $|\sigma| + |\kappa| < 1$ in Theorem 3.1, so "Bound 1 of Krylov Diag PP-IC" from Theorem 3.1 is effective, in contrast

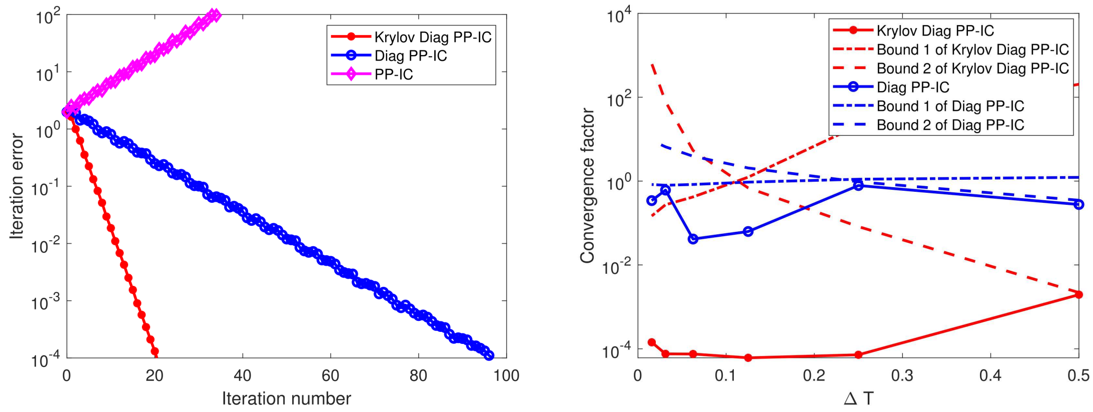

Fig. 5.12. Convergence of two new algorithms versus the PP-IC algorithm for the nonlinear wave problem (5.2) with $\Delta T = 1/8, \Delta t = 1/128$ (left), and dependence of the convergence factor on ΔT with $\Delta t = 1/128$ (right).

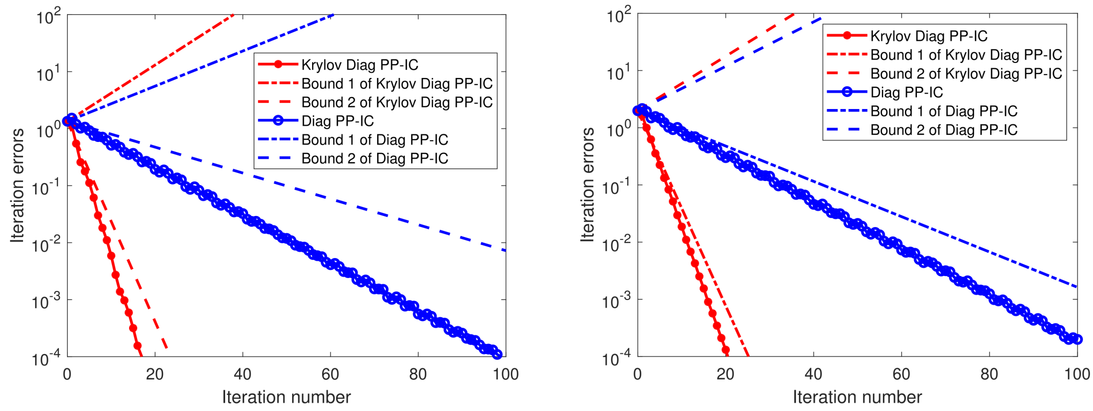

Fig. 5.13. Convergence behavior of two new algorithms for the nonlinear wave problem (5.2): comparison of the numerical results and the theoretical bounds for $\Delta T = 1/4$ (left), $\Delta T = 1/16$ (right) with $\Delta t = 1/128$.

when $\Delta T = 1/4$ is large, $\sigma = 0.1624, \kappa = 0.4636$ that satisfy the condition $|\sigma| + |\kappa| < 1$ in Theorem 3.2, so "Bound 2 of Krylov Diag PP-IC" from Theorem 3.2 is effective.

5.3. The viscous Burgers equation

We finally show numerical results for a nonlinear partial differential equation, the viscous Burgers equation

$$\begin{aligned} & u_t + uu_x = \nu u_{xx} + f(x,t), \quad (x,t) \in \Omega \times [0,T], \\ & u(x,0) = u(x,T) \end{aligned} \tag{5.4}$$

with homogeneous boundary conditions, where $\nu \in \mathbb{R}$ denotes the diffusivity (or viscosity) coefficient, $f : \Omega \times [0,T] \to \mathbb{R}$ and satisfies $f(x,0) = f(x,T)$ for any $x \in \Omega$.

We solve problem (5.4) on the time interval $[0,1]$ and the spatial domain $\Omega = [0,1]$ with the nonlinear function

$$f(x,t) = 2\pi \sin(2\pi x)\cos(2\pi t) + 2\pi \sin(2\pi x)\cos(2\pi x)\sin^2(2\pi t) + 4\pi^2 \nu \sin(2\pi x)\sin(2\pi t)$$

such that the exact solution is $u_{\text{exact}} = \sin(2\pi x)\sin(2\pi t)$. We choose the viscosity parameter $\nu = 1/50$, such that the problem is dominated by the nonlinear convective term. In addition, we use a centered finite difference discretization with spatial step $\Delta x = 1/128$, and a backward Euler discretization in time. We use the Newton's method for the F and F^* propagators to solve the nonlinear problems arising from the implicit time discretization.

We first show in Fig. 5.14 the iteration number of two new algorithms when the parameter $\alpha \in (0,1)$ varies, where we stop the iterations when the iteration errors decrease to 10^{-4}. We observe again that the convergence rate of the Diag PP-IC algorithm is still sensitive to the change in α, while for the Krylov Diag PP-IC algorithm the iteration number remains constant when α varies. Thus we will choose the relatively good parameter $\alpha = 0.7$ from Fig. 5.14 in order to converge faster in the following. Furthermore, we also show in Fig. 5.15 the iteration errors of two algorithms for four values of α, as observed in Fig. 5.14.

We next show the convergence results in Fig. 5.16. From the figure on the left, we also observe that the convergence behavior of both the Diag PP-IC algorithm and the Krylov Diag PP-IC algorithm is better than that of the PP-IC algorithm. At the same time, the Krylov Diag PP-IC algorithm converges faster than the Diag PP-IC algorithm. Then, we compare the numerical convergence factors of two new algorithms with the theoretical bounds from Theorems 2.2, 2.3, 3.1, 3.2 for the nonlinear case on the right in Fig. 5.16, where “Bound 1 of Diag PP-IC” is from Theorem 2.2, “Bound 2 of Diag PP-IC” is from Theorem 2.3, “Bound 1 of Krylov Diag PP-IC” is from Theorem 3.1, “Bound 2 of Krylov Diag PP-IC” is from Theorem 3.2. Here we choose the constants $-L_1, -L_2, L_3$ are from the Jacobian matrix of the nonlinear function obtained by the space discretization in Theorems 2.2, 2.3, 3.1, 3.2, which satisfy Assumptions 2.1-2.3. Therefore, we have given explicitly the estimates of the constants $L_1 = 7.24, L_2 = 1.68, L_3 = 6.45$.

In Fig. 5.17, we still show the convergence behavior of two new algorithms and their theoretical upper bounds on errors. We see that for the Diag PP-IC algorithm when $\Delta T = 1/16$ is small, $\sigma = 0.3439, \kappa = 0.6017$ that satisfy the condition $|\sigma| + |\kappa| < 1$ in Theorem 2.2, so “Bound 1 of Diag PP-IC” from Theorem 2.2 is effective, while when $\Delta T = 1/4$ is large, $\sigma = 0.3827, \kappa = 0.5541$ that satisfy the condition $|\sigma| + |\kappa| < 1$ in Theorem 2.3, so “Bound 2 of Diag PP-IC” from

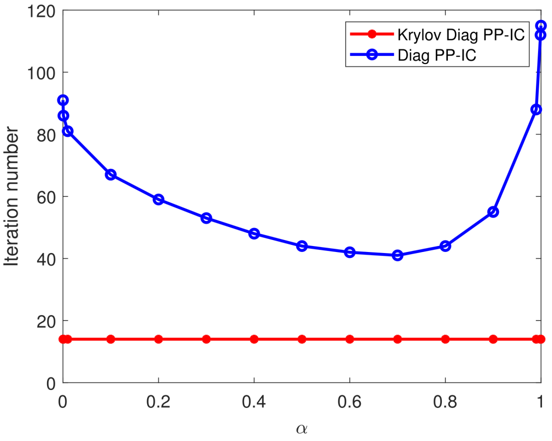

Fig. 5.14. Dependence of the convergence rate of two new algorithms on α for the viscous Burgers problem (5.4) with $\Delta T = 1/8, \Delta t = 1/128$.

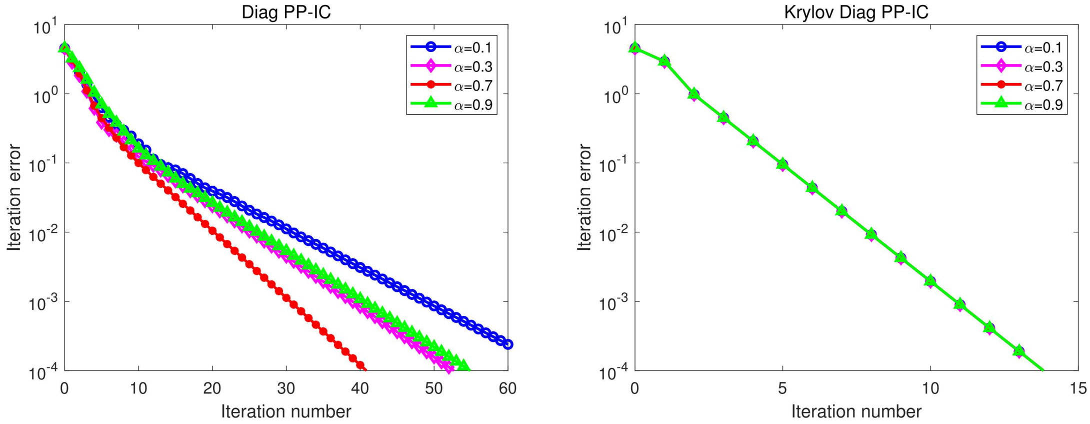

Fig. 5.15. Convergence of the Diag PP-IC (left) and Krylov Diag PP-IC (right) algorithms with various values of α for the viscous Burgers problem (5.4) with $\Delta T = 1/8, \Delta t = 1/128$.

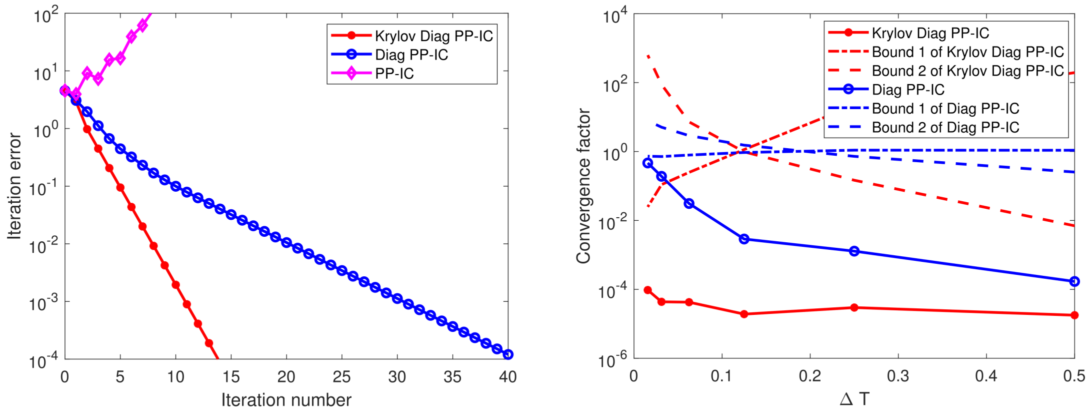

Fig. 5.16. Convergence of two new algorithms versus the PP-IC algorithm for the viscous Burgers problem (5.4) with $\Delta T = 1/8, \Delta t = 1/128$ (left), and dependence of the convergence factor on ΔT with $\Delta t = 1/128$ (right).

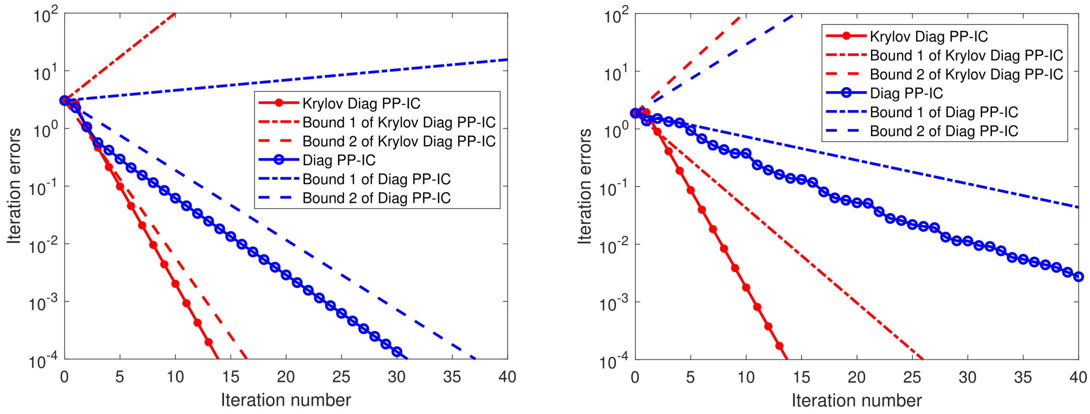

Fig. 5.17. Convergence behavior of two new algorithms for the viscous Burgers problem (5.4): comparison of the numerical results and the theoretical bounds for $\Delta T = 1/4$ (left), $\Delta T = 1/16$ (right) with $\Delta t = 1/128$.

Theorem 2.3 is effective. Similarly, for the Krylov Diag PP-IC algorithm when $\Delta T = 1/16$ is small, $\sigma = 0.2143, \kappa = 0.5731$ that satisfy the condition $|\sigma| + |\kappa| < 1$ in Theorem 2.2, so "Bound 1 of Krylov Diag PP-IC" from Theorem 3.1 is effective, while when $\Delta T = 1/4$ is large, $\sigma = 0.3011, \kappa = 0.4163$ that satisfy the condition $|\sigma| + |\kappa| < 1$ in Theorem 2.3, so "Bound 2 of Krylov Diag PP-IC" from Theorem 3.2 is effective. Finally, we also investigate the influence of the viscosity coefficient ν on the convergence of the Diag PP-IC and Krylov Diag PP-IC algorithms in Fig. 5.18.

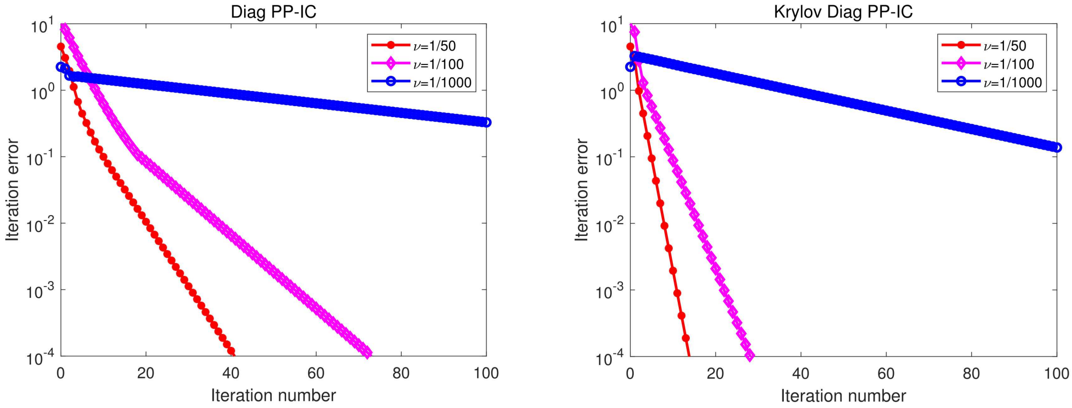

Fig. 5.18. Convergence of the Diag PP-IC (left) and Krylov Diag PP-IC (right) algorithms with various values of ν for the viscous Burgers problem (5.4) with $\Delta T = 1/8, \Delta t = 1/128$.

5.4. The Maxwell's equations

Electrodynamic potential formulations with regularization, based on Maxwell's equations described by electromagnetic phenomena, are frequently used in low frequency [7, 8] and high frequency applications [33]. Here we test the performance of our new proposed algorithms for the simulation of a metal bar [3], so-called an electromagnetic model described by full Maxwell with Coulomb gauge. One contact is excited by a sinusoidal voltage $v(t) = \sin(2\pi t)\mathrm{V}$, and the other contact is grounded.

The spatial discretization of the structure using the finite integration technique with 325 mesh cells and 845 degrees of freedom leads to the following time-periodic problem represented by the following system of differential-algebraic equations (DAEs)

$$\begin{aligned} &M\frac{d}{dt}\mathbf{u}(t) + K\mathbf{u}(t) = \mathbf{f}(t), \quad t \in [0, T], \\ &\mathbf{u}(0) = \mathbf{u}(T), \end{aligned} \tag{5.5}$$

where $M \in \mathbb{R}^{d\times d}$ is a singular constant matrix, $K \in \mathbb{R}^{d\times d}$ is a constant matrix, $\mathbf{u} : [0, T] \to \mathbb{R}^d$ contains the time-dependent degrees of freedom, and the right-hand side $\mathbf{f} : [0, T] \to \mathbb{R}^d$ satisfies $\mathbf{f}(0) = \mathbf{f}(T)$, $T = 1$s. The implicit Euler scheme is applied to both the F and F^* propagators with the time step $\Delta t = 10^{-4}$s, and the coarse propagator G with the time step $\Delta T = 10^{-1}$s. The numerical solution of system (5.5) is given on the left in Fig. 5.19. The convergence results of the algorithms are shown on the right in Fig. 5.19. We see that both the Diag PP-IC algorithm and the Krylov Diag PP-IC algorithm converge much more quickly than the PP-IC algorithm. Meanwhile, the convergence behavior of the Krylov Diag PP-IC algorithm is better than that of the Diag PP-IC algorithm.

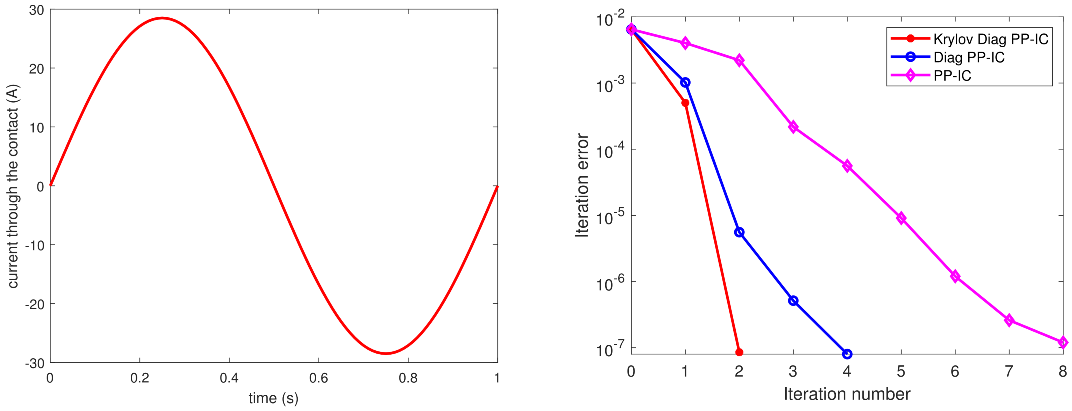

Fig. 5.19. Current through the metal bar (left), and convergence behavior of the two new algorithms versus the PP-IC algorithm for the system (5.5) (right).

6. Conclusion

We designed and analyzed two new parareal algorithms based on diagonalization for time-periodic problems: the first one is called Diag PP-IC algorithm, and the second one is called Krylov Diag PP-IC algorithm. Comparing with the classical PP-IC algorithm which accelerates sequential computations by parallelization on the fine grid, these new proposed algorithms increase parallelization by diagonalization on the coarse grid to have the effect of acceleration. We analyzed theoretical bounds of the convergence factors of two new proposed algorithms. Finally, in numerical experiments, we derived a superior choice of the parameter α in the head-tail coupled condition to lead to faster convergence numerically for model problems. Moreover, we observed that the performance of two new algorithms is better than that of the PP-IC algorithm, and the Krylov Diag PP-IC algorithm converges faster than the Diag PP-IC algorithm, while numerical results also illustrate well our theoretical results. In the future work, we will further combine waveform relaxation methods [38, 40] or Schwarz waveform relaxation methods [5, 24] to obtain more efficient space-time parallel algorithms for time-periodic problems.

Acknowledgements. The authors are very grateful to the editor and the referees for their careful reading of a preliminary version of the manuscript and their valuable suggestions, which greatly improved the quality of this paper.

This work was supported by the Natural Science Foundation of China (NSFC) (Grant No. 12271426), by the Key Research and Development Projects of Shaanxi Province (Grant No. 2023-YBSF-399), by the Natural Science Basic Research Plan in Shaanxi Province of China (Grant Nos. 2023-JC-YB-030, 2022JQ-067), and by the China Postdoctoral Science Foundation (Grant Nos. 2022T150535, 2020M673465).

References

[1] F. Bachinger, U. Langer, and J. Schöberl, Efficient solvers for nonlinear time-periodic eddy current problems, *Comput. Visualization Sci.*, **9** (2006), 197–207.

[2] G. Bal, On the convergence and the stability of the parareal algorithm to solve partial differential equations, in: *Domain Decomposition Methods in Science and Engineering*, Vol. 40, Springer, (2005), 425–432.

[3] S. Baumanns, M. Clemens, and S. Schöps, Structural aspects of regularized full Maxwell electrodynamic potential formulations using FIT, in: *2013 International Symposium on Electromagnetic Theory*, IEEE, (2013), 1007–1010.

[4] C. Bernardi, Numerical approximation of a periodic linear parabolic problem, *SIAM J. Numer. Anal.*, **19** (1982), 1196–1207.

[5] D.Q. Bui, C. Japhet, Y. Maday, and P. Omnes, Coupling parareal with optimized Schwarz waveform relaxation for parabolic problems, *SIAM J. Numer. Anal.*, **60** (2022), 913–939.

[6] B. Carrel, M.J. Gander, and B. Vandereycken, Low-rank parareal: A low-rank parallel-in-time integrator, *BIT Numer. Math.*, **63** (2023), 13.

[7] M. Clemens, S. Schöps, H. De Gersem, and A. Bartel, Decomposition and regularization of nonlinear anisotropic curl-curl DAEs, *COMPEL*, **30** (2011), 1701–1714.

[8] M. Clemens and T. Weiland, Regularization of eddy-current formulations using discrete grad-div operators, *IEEE Trans. Magn.*, **38** (2002), 569–572.

[9] X.Y. Dai and Y. Maday, Stable parareal in time method for first- and second-order hyperbolic systems, *SIAM J. Sci. Comput.*, **35** (2013), A52–A78.

[10] V.A. Dobrev, T. Kolev, N.A. Petersson, and J.B. Schroder, Two-level convergence theory for multigrid reduction in time (MGRIT), *SIAM J. Sci. Comput.*, **39** (2017), S501–S527.

[11] M. Emmett and M.L. Minion, Toward an efficient parallel in time method for partial differential equations, *Commun. Appl. Math. Comput. Sci.*, **7** (2012), 105–132.

[12] S. Engblom, Parallel in time simulation of multiscale stochastic chemical kinetics, *Multiscale Model. Simul.*, **8** (2009), 46–68.

[13] R.D. Falgout, S. Friedhoff, T.V. Kolev, S.P. MacLachlan, and J.B. Schroder, Parallel time integration with multigrid, *SIAM J. Sci. Comput.*, **36** (2014), C635–C661.

[14] C. Farhat and M. Chandesris, Time decomposed parallel time integrators: Theory and feasibility studies for fluid, structure, and fluid structure applications, *Int. J. Numer. Methods Eng.*, **58** (2010), 1397–1434.

[15] C. Farhat, J. Cortial, D. Climène, and H. Bavestrello, Time parallel implicit integrators for the near real time prediction of linear structural dynamic responses, *Int. J. Numer. Methods Eng.*, **67** (2010), 697–724.

[16] M.J. Gander, Analysis of the parareal algorithm applied to hyperbolic problems using characteristics, in: *Boletin de la Sociedad Espanola de Matemática Aplicada*, (2008), 5–19.

[17] M.J. Gander, 50 years of time parallel time integration, in: *Multiple Shooting and Time Domain Decomposition Methods*, Vol. 9, Springer, (2015), 69–113.

[18] M.J. Gander and S. Güttel, PARAEXP: A parallel integrator for linear initial-value problems, *SIAM J. Sci. Comput.*, **35** (2013), C123–C142.

[19] M.J. Gander and E. Hairer, Nonlinear convergence analysis for the parareal algorithm, in: *Domain Decomposition Methods in Science and Engineering XVII. Lecture Notes in Computational Science and Engineering*, Vol. 60, Springer, (2008), 45–56.

[20] M.J. Gander and E. Hairer, Analysis for parareal algorithms applied to Hamiltonian differential equations, *J. Comput. Appl. Math.*, **259** (2014), 2–13.

[21] M.J. Gander and L. Halpern, Time parallelization for nonlinear problems based on diagonalization, in: *Domain Decomposition Methods in Science and Engineering XXIII. Lecture Notes in Computational Science and Engineering*, Vol. 116, Springer, (2017), 163–170.

[22] M.J. Gander, L. Halpern, J. Rannou, and J. Ryan, A direct time parallel solver by diagonalization for the wave equation, *SIAM J. Sci. Comput.*, **41** (2019), A220–A245.

[23] M.J. Gander, L. Halpern, J. Ryan, and T.T.B. Tran, A direct solver for time parallelization, in: *Domain Decomposition Methods in Science and Engineering XXII. Lecture Notes in Computational Science and Engineering*, Vol. 104, Springer, (2016), 491–499.

[24] M.J. Gander, Y.L. Jiang, and B. Song, A superlinear convergence estimate for the parareal Schwarz waveform relaxation algorithm, *SIAM J. Sci. Comput.*, **41** (2019), A1148–A1169.

[25] M.J. Gander, Y.L. Jiang, B. Song, and H. Zhang, Analysis of two parareal algorithms for time-periodic problems, *SIAM J. Sci. Comput.*, **35** (2013), A2393–A2415.

[26] M.J. Gander, I. Kulchytska-Ruchka, and S. Schöps, A new parareal algorithm for time-periodic problems with discontinuous inputs, in: *Domain Decomposition Methods in Science and Engineering XXV. Lecture Notes in Computational Science and Engineering*, Vol. 138, Springer, (2020), 243–250.

[27] M.J. Gander, F. Kwok, and J. Salomon, PARAOPT: A parareal algorithm for optimality systems, *SIAM J. Sci. Comput.*, **42** (2020), A2773–A2802.

[28] M.J. Gander and D. Palitta, A new ParaDiag time-parallel time integration method, *SIAM J. Sci. Comput.*, **46**:2 (2023), A697–A718.

[29] M.J. Gander and M.E. Petcu, Analysis of a Krylov subspace enhanced parareal algorithm for linear problems, *ESAIM: Proc.*, **25** (2008), 114–129.

[30] M.J. Gander and S. Vandewalle, Analysis of the parareal time parallel time integration method, *SIAM J. Sci. Comput.*, **29** (2007), 556–578.

[31] M.J. Gander and S.L. Wu, A diagonalization-based parareal algorithm for dissipative and wave propagation problems, *SIAM J. Numer. Anal.*, **58** (2020), 2981–3009.

[32] W. Hackbusch, Fast numerical solution of time-periodic parabolic problems by a multigrid method, *SIAM J. Sci. Statist. Comput.*, **2** (1981), 198–206.

[33] P. Hahne and T. Weiland, 3D eddy current computation in the frequency domain regarding the displacement current, *IEEE Trans. Magn.*, **28** (1992), 1801–1804.

[34] A. Hessenthaler, R.D. Falgout, J.B. Schroder, A. de Vecchi, D. Nordsletten, and O. Rohrle, Time-periodic steady-state solution of fluid-structure interaction and cardiac flow problems through multigrid-reduction-in-time, *Comput. Methods Appl. Mech. Engrg.*, **389** (2022), 114368.

[35] Y.L. Jiang, Periodic waveform relaxation solutions of nonlinear dynamic equations, *Appl. Math. Comput.*, **135** (2003), 219–226.

[36] M. Kolmbauer and U. Langer, A robust preconditioned MinRes solver for distributed time-periodic eddy current optimal control problems, *SIAM J. Sci. Comput.*, **34** (2012), B785–B809.

[37] I. Kulchytska-Ruchka and S. Schöps, Efficient parallel-in-time solution of time-periodic problems using a multiharmonic coarse grid correction, *SIAM J. Sci. Comput.*, **43** (2021), C61–C88.

[38] J. Li and Y.L. Jiang, Analysis of a new accelerated waveform relaxation method based on the time-parallel algorithm, *J. Sci. Comput.*, **96** (2023), 68.

[39] J.L. Lions, Y. Maday, and G. Turinici, A "parareal" in time discretization of PDE's, *C. R. Acad. Sci. Paris Sér. I Math.*, **332** (2001), 661–668.

[40] J. Liu and Y.L. Jiang, A parareal algorithm based on waveform relaxation, *Math. Comput. Simulation*, **82** (2012), 2167–2181.

[41] Y. Maday and E.M. Rønquist, Parallelization in time through tensor-product space-time solvers, *C. R. Math.*, **346** (2008), 113–118.

[42] Y. Maday, J. Salomon, and G. Turinici, Monotonic parareal control for quantum systems, *SIAM J. Numer. Anal.*, **45** (2007), 2468–2482.

[43] Y. Maday and G. Turinici, A parareal in time procedure for the control of partial differential equations, *C. R. Math.*, **335** (2002), 387–392.

[44] M.L. Minion, A hybrid parareal spectral deferred corrections method, *Commun. Appl. Math. Comput. Sci.*, **5** (2010), 265–301.

[45] C.V. Pao, Numerical methods for time-periodic solutions of nonlinear parabolic boundary value problems, *SIAM J. Numer. Anal.*, **39** (2001), 647–667.

[46] A. Potschka, M.S. Mommer, J.P. Schlöder, and H.G. Bock, Newton-Picard-based preconditioning for linear-quadratic optimization problems with time-periodic parabolic PDE constraints, *SIAM J. Sci. Comput.*, **34** (2012), A1214–A1239.

[47] D. Ruprecht and R. Krause, Explicit parallel-in-time integration of a linear acoustic-advection system, *Comput. & Fluids*, **59** (2012), 72–83.

[48] B. Song and Y.L. Jiang, Analysis of a new parareal algorithm based on waveform relaxation method for time-periodic problems, *Numer. Algorithms*, **67** (2014), 599–622.

[49] B. Song and Y.L. Jiang, A new parareal waveform relaxation algorithm for time-periodic problems, *Int. J. Comput. Math.*, **92** (2015), 377–393.

[50] B. Song, J.Y. Wang, and Y.L. Jiang, Analysis of a new Krylov subspace enhanced parareal algorithm for time-periodic problems, *Numer. Algorithms*, **97** (2024), 289–310.

[51] G.A. Staff and E.M. Rønquist, Stability of the parareal algorithm, in: *Domain Decomposition Methods in Science and Engineering. Lecture Notes in Computational Science and Engineering*, Vol 40, Springer, (2005), 449–456.

[52] M. Steuerwalt, The existence, computation, and number of solutions of periodic parabolic problems, *SIAM J. Numer. Anal.*, **16** (1979), 402–420.

[53] B.A. van de Rotten, S.M.V. Lunel, and A. Bliek, Efficient simulation of periodically forced reactors with radial gradients, *Chem. Eng. Sci.*, **61** (2006), 6981–6994.

[54] T.L. van Noorden, S.M.V. Lunel, and A. Bliek, The efficient computation of periodic states of cyclically operated chemical processes, *IMA J. Appl. Math.*, **68** (2003), 149–166.

[55] S. Vandewalle and R. Piessens, On dynamic iteration methods for solving time-periodic differential equations, *SIAM J. Numer. Anal.*, **30** (1993), 286–303.

[56] R.S. Varga, *Matrix Iterative Analysis*, Springer-Verlag, 2000.

[57] B. Wang and Y.L. Jiang, Improved uniform error bounds on parareal exponential algorithm for highly oscillatory systems, *BIT Numer. Math.*, **64** (2024).

[58] S.L. Wu and T. Zhou, Fast parareal iterations for fractional diffusion equations, *J. Comput. Phys.*, **329** (2017), 210–226.