SIAM J. MATRIX ANAL. APPL.
Vol. 41, No. 4, pp. 1912–1943

A FAST BLOCK α-CIRCULANT PRECONDITONER FOR ALL-AT-ONCE SYSTEMS FROM WAVE EQUATIONS*

JUN LIU† AND SHU-LIN WU‡

Abstract. In this paper, we propose a fast block α-circulant preconditioner for solving the nonsymmetric linear system arising from an all-at-once implicit discretization scheme in time for the wave equation. As a generalization of the well-known block circulant preconditioning technique, the proposed block α-circulant preconditioner can also be efficiently inverted in a parallel-in-time manner. The complex eigenvalues of the preconditioned matrix are fully derived in explicit expression and its diagonalizability is also shown. Furthermore, a mesh-independent convergence rate of the preconditioned GMRES method is derived under certain conditions. Building upon the proposed preconditioner, two stationary iterative methods with uniform asymptotic convergence rates were also presented. The extension of our preconditioner within a simplified Newton iteration to the nonlinear wave equation is also discussed. Both linear and nonlinear numerical examples are given to illustrate the promising performance of our proposed block α-circulant preconditioner and also validate our theoretical results on convergence analysis.

Key words. all-at-once scheme, wave equations, α-circulant preconditioner, parallel-in-time (PinT), diagonalization

AMS subject classifications. 65M55, 65M12, 65M15, 65Y05

DOI. 10.1137/19M1309869

1. Introduction.

Wave propagation is ubiquitous in all fields of science and engineering. Development of efficient and accurate numerical methods [8, 10, 30, 33] for simulating the underlying wave equations has been an active topic in the last few decades. In particular, the study of the parallel-in-time (PinT) algorithms[1] for such time-dependent PDE problems has attracted a lot of attention recently. In this paper, we continue to give impetus to this field by proposing new PinT algorithms for the following linear wave equation model:

$$\begin{cases} y_{tt} - \Delta y = f & \text{in } \Omega \times (0,T), \\ y = 0 & \text{on } \partial\Omega \times (0,T), \\ y(\cdot,0) = \psi_0,\ y_t(\cdot,0) = \psi_1 & \text{in } \Omega, \end{cases} \tag{1.1}$$

where $\Omega \subset \mathbb{R}^d$ with $d \geq 1$ is a bounded and open domain with Lipschitz boundary, ψ_0 and ψ_1 are given compatible initial conditions, and f is a given source term. Such hyperbolic type PDEs present fundamental challenges toward the PinT computation of wave propagation problems. Over the past decade, there were considerable efforts to apply the parareal algorithm to wave propagation problems; see, e.g., [6, 9, 11, 18, 23, 47, 55]. However, as pointed out by Ruprecht in [54], these existing parareal-based approaches are *typically with significant overhead, leading to further degradation of*

*Received by the editors December 30, 2019; accepted for publication (in revised form) by L. Giraud September 14, 2020; published electronically December 10, 2020.
https://doi.org/10.1137/19M1309869
Funding: The second author's work was supported by NSF of China (11771313) and NSF of Sichuan (2018JY0469).

†Department of Mathematics and Statistics, Southern Illinois University Edwardsville, Edwardsville, IL 62026 USA (juliu@siue.edu).
‡Corresponding author. School of Mathematics and Statistics, Northeast Normal University, Changchun 130024, China (wushulin84@hotmail.com).

[1]We refer the interested reader to http://parallel-in-time.org for an overview of the PinT research.

efficiency, or limited applicability. Another interesting PinT algorithm for solving (1.1) was recently proposed in [22], where variable nonuniform step-sizes for time-discretization are used and the space-time discretization is formulated into an *all-at-once* system. Such an all-at-once system has a special structure, namely the time-discretization matrix can be diagonalized with closed formulas due to the variable step-sizes, which permits us to solve it in a direct PinT way via the so-called *diagonalization technique* proposed by Maday and Rønquist [41] in 2008. This algorithm, however, has the difficulty of balancing the discretization error and the roundoff error when the number of time points is large. We will focus on developing fast iterative methods for the discretized sparse all-at-once system.

Inspired by the recent works [24, 42], in this paper we propose a different efficient diagonalization-based PinT algorithm for solving (1.1). The basic idea is as follows. We first discretize (1.1) all-at-once in time by the implicit leap-frog finite difference scheme established in [35] (but other time schemes can also be adopted), which was shown to be unconditionally stable without imposing the restrictive Courant–Friedrichs–Lewy (CFL) condition on spatial and temporal mesh step sizes. Precisely, given two positive integers N_x and N_t, let $h = \frac{1}{N_x+1}$ and $\tau = \frac{T}{N_t}$ be the space and time mesh sizes, and then we partition the time interval $[0, T]$ uniformly by time points $\{t_n\}_{n=0}^{N_t}$ with $t_n = n\tau$. Let Δ_h be the second-order accurate discrete matrix approximation of the Laplacian operator Δ in (1.1) obtained by using the central finite difference method with the given boundary condition. The implicit leap-frog finite difference scheme [35] for discretizing (1.1) reads

(1.2)
$$\frac{Y_{n+1} - 2Y_n + Y_{n-1}}{\tau^2} - \Delta_h \frac{Y_{n+1} + Y_{n-1}}{2} = F_n + O(\tau^2 + h^2), \quad n = 1, 2, \ldots, N_t - 1,$$

where Y_n is a column vector containing the approximate values of $y(\cdot, t_n)$ over all the grid nodes of the spatial mesh (similar notation also applies to F_n, Ψ_0, and Ψ_1 corresponding to f, ψ_0, and ψ_1, respectively). Here the averaged Laplacian term maintains the same second order of accuracy, which is crucial to obtain an unconditionally stable scheme and the desirable block Toeplitz matrix structure along the time direction. The initial condition $y_t(\cdot, 0) = \psi_1$ is approximated by using the Taylor expansions of $y(\cdot, \tau)$:

$$y(\cdot, \tau) = y(\cdot, 0) + \tau y_t(\cdot, 0) + \frac{\tau^2}{2} y_{tt}(\cdot, 0) + O(\tau^3),$$

which, employing the implicit approximation $y_{tt}(\cdot, 0) \approx \Delta y(\cdot, \tau) + f(\cdot, 0)$ from the governing equation, gives

$$Y_1 - \frac{\tau^2 \Delta_h}{2} Y_1 = \Psi_0 + \tau \Psi_1 + \frac{\tau^2}{2} F_0 + O(\tau^2 + h^2). \tag{1.3}$$

We will focus on discussing the two-dimensional (2D) case with $\Omega \subset \mathbb{R}^2$. Let $I_t \in \mathbb{R}^{N_t \times N_t}$ and $I_x \in \mathbb{R}^{N_x \times N_x}$ be the identity matrices and define $L = I_x - \frac{\tau^2}{2}\Delta_h \in \mathbb{R}^{N_x \times N_x}$. With Kronecker product notation, we can represent the above implicit finite difference scheme (1.2)–(1.3) by an all-at-once non-symmetric linear system:

(1.4a)
$$\mathcal{K} y_h := \frac{1}{\tau^2} \begin{bmatrix} L & & & & \\ -2I_x & L & & & \\ L & -2I_x & L & & \\ & \ddots & \ddots & \ddots & \\ & & L & -2I_x & L \end{bmatrix} y_h = \frac{1}{\tau^2} \left(B_1 \otimes L - B_2 \otimes 2I_x \right) y_h = b_h,$$

where

$$B_1 = \begin{bmatrix} 1 & & & & \\ 0 & 1 & & & \\ 1 & 0 & 1 & & \\ & \ddots & \ddots & \ddots & \\ & & 1 & 0 & 1 \end{bmatrix}, \quad B_2 = \begin{bmatrix} 0 & & & & \\ 1 & 0 & & & \\ & 1 & 0 & & \\ & & \ddots & \ddots & \\ & & & 1 & 0 \end{bmatrix},$$

(1.4b)

$$y_h = \begin{bmatrix} Y_1 \\ Y_2 \\ \vdots \\ Y_{N_t-1} \\ Y_{N_t} \end{bmatrix}, \quad b_h = \begin{bmatrix} F_0/2 + \Psi_1/\tau + \Psi_0/\tau^2 \\ F_1 - L\Psi_0/\tau^2 \\ F_2 \\ \vdots \\ F_{N_t-1} \end{bmatrix}.$$

For a general irregular spatial domain, the standard finite element discretization [15, 29] (e.g, P_1 element) can be used in space, where the identity matrix I_x and discrete negative Laplacian matrix $(-\Delta_h)$ need to be replaced by the mass matrix and stiffness matrix, respectively. Our following proposed preconditioner and presented analysis are still applicable upon some appropriate modifications (as demonstrated in Example 3 in section 5.3).

We will solve (1.4a) iteratively by the preconditioned GMRES [58] and stationary iterative methods. The key idea in [24, 42] is to construct an efficient and effective preconditioner, where the two time-discretization Toeplitz matrices B_1 and B_2 are replaced by two Strang-type circulant matrices [5]

$$C_1 = \begin{bmatrix} 1 & & & 1 & 0 \\ 0 & 1 & & & 1 \\ 1 & 0 & 1 & & \\ & \ddots & \ddots & \ddots & \\ & & 1 & 0 & 1 \end{bmatrix}, \quad C_2 = \begin{bmatrix} 0 & & & & 1 \\ 1 & 0 & & & \\ & 1 & 0 & & \\ & & \ddots & \ddots & \\ & & & 1 & 0 \end{bmatrix}, \tag{1.5}$$

respectively, and then define the block-circulant matrix

$$\mathcal{P} = \frac{1}{\tau^2}\left(C_1 \otimes L - C_2 \otimes 2I_x\right)$$

as the preconditioner. The matrices C_1 and C_2 can be diagonalized simultaneously, and thus for any vector r we can compute $\mathcal{P}^{-1}r$ parallel in time via the FFT-based diagonalization technique (cf. section 2.1 for details). Unfortunately, as indicated by Figure 1.1, the preconditioner constructed along this idea does not achieve a satisfactory convergence rate for the hyperbolic wave equation (1.1). Nevertheless, for parabolic equations, e.g., the heat equations studied in [24, 42], the similar preconditioner obtained in this way indeed shows a rather satisfactory convergence rate. Notice that the all-at-once systems derived from hyperbolic wave equations have very different algebraic properties from the parabolic equations.

Interestingly, as indicated by the results in Figure 1.1, the slow convergence of this preconditioner $\mathcal{P}$ can be dramatically improved by some slight generalization for the two matrices C_1 and C_2. The key idea is to modify them into α-circulant (sometimes also known as ω-circulant [1] in literature) matrices $C_1^{(\alpha)}$ and $C_2^{(\alpha)}$:

$$C_1^{(\alpha)} = \begin{bmatrix} 1 & & & \alpha & 0 \\ 0 & 1 & & & \alpha \\ 1 & 0 & 1 & & \\ & \ddots & \ddots & \ddots & \\ & & 1 & 0 & 1 \end{bmatrix}, \quad C_2^{(\alpha)} = \begin{bmatrix} 0 & & & & \alpha \\ 1 & 0 & & & \\ & 1 & 0 & & \\ & & \ddots & \ddots & \\ & & & 1 & 0 \end{bmatrix}, \tag{1.6}$$

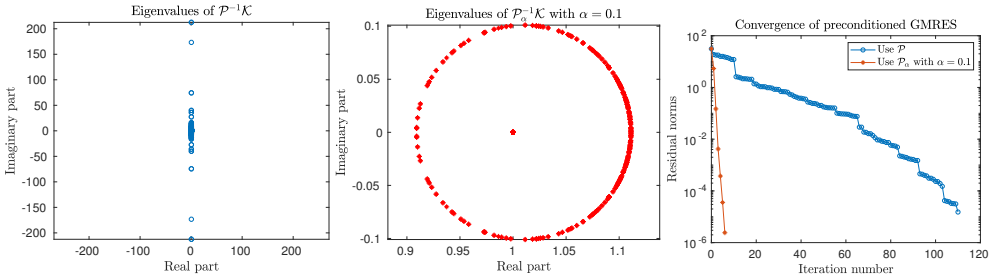

FIG. 1.1. *Left: the eigenvalues distribution of $\mathcal{P}^{-1}\mathcal{K}$; Middle: the eigenvalues distribution of $\mathcal{P}_\alpha^{-1}\mathcal{K}$ with $\alpha = 0.1$; Right: the residual norm at each iteration for the GMRES method using preconditioner $\mathcal{P}$ and $\mathcal{P}_\alpha^{-1}$. Here, we used the 2D data given in Example 2 (see section 5.2 below) with $T = 2$, $N_x = 16$, and $N_t = 16$.*

where $\alpha \in (0, 1]$ is a free parameter. This leads to a generalized block α-circulant preconditioner

$$\mathcal{P}_\alpha = \frac{1}{\tau^2}\left(C_1^{(\alpha)} \otimes L - C_2^{(\alpha)} \otimes 2I_x\right). \tag{1.7}$$

As will be shown in section 2.1, $\mathcal{P}_\alpha^{-1} r$ can also be computed parallel in time, because $C_1^{(\alpha)}$ and $C_2^{(\alpha)}$ are simultaneously diagonalizable. We mention that similar block α-circulant matrices were studied in [38, 39] as an approximate inverse and in [1, 17, 36] as preconditioners, where the involved dense Toeplitz matrices have different algebraic properties due to the underlying fractional differential operators. Although such an idea of a (block) α-circulant preconditioner is not new, most of the existing works deal with only parabolic equations, which is very different from the wave equation that is usually more difficult to treat.

The goal of this paper is to make a thorough analysis of such an interesting preconditioning technique for the wave equation (1.1). In particular, we will perform a precise spectral analysis of the preconditioned matrix $\mathcal{P}_\alpha^{-1}\mathcal{K}$ and prove that all the eigenvalues of this matrix distribute within an annulus with inner radius $\frac{\alpha}{1+\alpha}$ and outer radius $\frac{\alpha}{1-\alpha}$. A mesh-independent convergence rate is derived for the preconditioned GMRES under certain condition on α. The required number of preconditioned GMRES iterations for achieving a given stopping tolerance can be approximately estimated from the corresponding stationary iterative method.

It is worthwhile to mention another related work [65], where the author proposed a new parareal algorithm using a diagonalization-based coarse grid correction, which is obtained by applying the so-called $\mathcal{G}$-propagator to the ODE system with a parameterized head-tail coupling condition (instead of the initial-value condition). Such a head-tail coupling condition also leads to a similar α-circulant matrix representing the time scheme. Interestingly, in the extreme case of using one-step time scheme (e.g., backward Euler or trapezoidal rule) with the same fine and coarse time steps and simultaneously the two propagators $\mathcal{F}$ and $\mathcal{G}$ are also identical, the corresponding parareal algorithm reduces to our proposed stationary iteration. However, the convergence analysis in [65] excludes such a very special configuration and hence does not directly apply to our proposed stationary iteration. Moreover, the implicit leapfrog time scheme used in this paper is a two-step method, which is not directly suitable for the parareal algorithm. Therefore, the convergence analysis here is fundamentally different from that in [65].

The remainder of this paper is organized as follows. In the next section, we show details about the diagonalization of the proposed preconditioner $\mathcal{P}_\alpha$ and its parallel implementation in time. We also analyze the eigenvalues and diagonalizability of the preconditioned matrix $\mathcal{P}_\alpha^{-1}\mathcal{K}$. In section 3, a mesh-independent convergence rate of preconditioned GMRES is derived under a certain condition, and two stationary iterative methods based on the same preconditioner are also analyzed. In section 4, a simplified Newton method based on the similar preconditioner with a simple averaging technique for Kronecker product approximation is presented for solving a semilinear wave equation. In section 5, several numerical examples are given to illustrate the performance of our proposed preconditioner and validate our convergence analysis. Finally, some conclusions are summarized in section 6.

2. The preconditioner $\mathcal{P}_\alpha$ and its analysis. In this section, we show that our proposed preconditioner $\mathcal{P}_\alpha$ can be efficiently inverted. We will also analyze the eigenvalues and diagonalizability of the preconditioned system, which is analogous to the results on parabolic equations in [42]. The difference between the spectra of preconditioned matrices $\mathcal{P}_\alpha^{-1}\mathcal{K}$ and $\mathcal{P}^{-1}\mathcal{K}$ is shown to explain the discrepancy in their convergence rates. Notice that $\mathcal{P}$ is a special case of $\mathcal{P}_\alpha$ with $\alpha = 1$.

2.1. PinT diagonalization of the preconditioner. Let

$$\mathbb{F} = \frac{1}{\sqrt{N_t}} \left[\omega^{(l_1-1)(l_2-1)}\right]_{l_1,l_2=1}^{N_t}$$

(with $\mathrm{i} = \sqrt{-1}$ and $\omega = e^{\frac{2\pi \mathrm{i}}{N_t}}$) be the discrete Fourier matrix, and define a diagonal matrix (for any given parameter $\alpha \in (0, 1]$)

$$\Gamma_\alpha = \begin{bmatrix} 1 & & & \\ & \alpha^{\frac{1}{N_t}} & & \\ & & \ddots & \\ & & & \alpha^{\frac{N_t-1}{N_t}} \end{bmatrix}. \tag{2.1}$$

The above α-circulant matrices $C_1^{(\alpha)}$ and $C_2^{(\alpha)}$ can be simultaneously diagonalized [3] as

$$C_j^{(\alpha)} = V D_j V^{-1}, \quad j = 1, 2,$$

where $V = \Gamma_\alpha^{-1}\mathbb{F}^*$ and $D_{1,2} = \mathrm{diag}\big(\sqrt{N_t}\mathbb{F}\Gamma_\alpha C_{1,2}^{(\alpha)}(:, 1)\big)$ with $C_{1,2}^{(\alpha)}(:, 1)$ being the first column of $C_{1,2}^{(\alpha)}$.

It obviously holds that

$$\mathcal{P}_\alpha = \frac{1}{\tau^2}(V \otimes I_x)\,(D_1 \otimes L - 2D_2 \otimes I_x)\,(V^{-1} \otimes I_x).$$

Therefore, for an input vector r, the inversion computation of $z := \mathcal{P}_\alpha^{-1} r$ can be implemented via three steps

$$\begin{aligned} &\text{step (a)} \quad S_1 = (V^{-1} \otimes I_x)r, \\ &\text{step (b)} \quad S_{2,n} = \tau^2(\lambda_{1,n}L - 2\lambda_{2,n}I_x)^{-1}S_{1,n}, \quad n = 1, 2, \ldots, N_t, \\ &\text{step (c)} \quad z = (V \otimes I_x)S_2, \end{aligned} \tag{2.2}$$

where $D_j = \mathrm{diag}(\lambda_{j,1}, \ldots, \lambda_{j,N_t})$ and $S_{j,n} = S_j((n-1)N_x : nN_x)$ denotes the nth block of S_j for $j = 1, 2$.

Noticing $V^{-1} = \mathbb{F}\Gamma_\alpha$ in (2.2), steps (a) and (c) can be computed efficiently via FFT in time direction with $O(N_x N_t \log N_t)$ operations. In step (b), if $\lambda_{1,n} = 0$, then it is trivial to solve the linear system. If $\lambda_{1,n} \neq 0$ and we let $\eta_n := (1 - \frac{2\lambda_{2,n}}{\lambda_{1,n}})$, then we essentially need to solve the complex-shifted Laplacian systems

$$\left(-\frac{1}{2}\Delta_h + \frac{\eta_n}{\tau^2} I_x\right) S_{2,n} = \frac{1}{\lambda_{1,n}} S_{1,n}.$$

Efficient computation of such a Helmholtz-like complex symmetric system has been extensively studied in the literature, and we refer to [7, 40, 48, 66] for the multigrid method and to [20, 27] for the preconditioned GMRES method. If these inner systems are solved by preconditioned GMRES, then the outer system can be solved by the flexible GMRES [56, 60], but its convergence is not guaranteed by our analysis for the full GMRES.

2.2. Eigenvalue analysis and diagonalizability of the preconditioned matrix. Let e_i be the ith column of the identity matrix I_t and define $M_i = e_i \otimes I_x$. We can represent $\mathcal{P}_\alpha$ as a low-rank perturbation of $\mathcal{K}$, i.e.,

$$\mathcal{P}_\alpha = \mathcal{K} + \frac{\alpha}{\tau^2}\begin{bmatrix} 0 & \cdots & 0 & L & -2I_x \\ & 0 & \cdots & 0 & L \\ & & 0 & \cdots & 0 \\ & & & \ddots & \vdots \\ & & & & 0 \end{bmatrix} =: \mathcal{K} + \alpha\mathcal{R},$$

where $\alpha\mathcal{R}$ is rank-$2N_x$ matrix having a simple decomposition

$$\alpha\mathcal{R} = \frac{1}{\tau^2}\begin{bmatrix} M_1 & M_2 \end{bmatrix}\begin{bmatrix} \alpha L & -2\alpha I_x \\ 0 & \alpha L \end{bmatrix}\begin{bmatrix} M_{N_t-1} & M_{N_t} \end{bmatrix}^\top =: \frac{1}{\tau^2} U C V^\top.$$

Now, by using the Sherman–Morrison–Woodbury formula [25] we get

$$\begin{aligned}\mathcal{P}_\alpha^{-1}\mathcal{K} &= (\mathcal{K} + \alpha\mathcal{R})^{-1}\mathcal{K} = \left(\mathcal{K} + \frac{1}{\tau^2} U C V^{\mathsf{T}}\right)^{-1}\mathcal{K} \\ &= \mathcal{I} - \mathcal{K}^{-1}U(\underbrace{\tau^2 C^{-1} + V^{\mathsf{T}}\mathcal{K}^{-1}U}_{=:Z})^{-1}V^{\mathsf{T}} =: \mathcal{I} - \mathcal{K}^{-1}UZ^{-1}V^{\mathsf{T}},\end{aligned}$$

where $\mathcal{I} = I_t \otimes I_x$ and

$$C^{-1} = \alpha^{-1}\begin{bmatrix} L & -2I_x \\ 0 & L \end{bmatrix}^{-1} = \alpha^{-1}\begin{bmatrix} L^{-1} & 2L^{-2} \\ 0 & L^{-1} \end{bmatrix}.$$

Since $\mathcal{K}^{-1}$ is also block lower triangular and block Toeplitz matrix, we can write

$$\mathcal{K}^{-1} = \begin{bmatrix} K_0^{-1} & & & & \\ K_1^{-1} & K_0^{-1} & & & \\ K_2^{-1} & K_1^{-1} & K_0^{-1} & & \\ \vdots & \ddots & \ddots & \ddots & \\ K_{N_t-1}^{-1} & \cdots & K_2^{-1} & K_1^{-1} & K_0^{-1} \end{bmatrix}.$$

A routine calculation yields

$$Z = \tau^2 C^{-1} + V^{\mathsf{T}}\mathcal{K}^{-1}U = \tau^2\alpha^{-1}\begin{bmatrix} L^{-1} & 2L^{-2} \\ 0 & L^{-1} \end{bmatrix} + \begin{bmatrix} K_{N_t-2}^{-1} & K_{N_t-3}^{-1} \\ K_{N_t-1}^{-1} & K_{N_t-2}^{-1} \end{bmatrix}. \tag{2.3}$$

By writing Z^{-1} into 2-by-2 blocks form

$$Z^{-1} = \begin{bmatrix} Z_{1,1}^{-1} & Z_{1,2}^{-1} \\ Z_{2,1}^{-1} & Z_{2,2}^{-1} \end{bmatrix},$$

we get

$$UZ^{-1}V^{\mathsf{T}} = \begin{bmatrix} 0 & \cdots & 0 & Z_{1,1}^{-1} & Z_{1,2}^{-1} \\ 0 & \cdots & 0 & Z_{2,1}^{-1} & Z_{2,2}^{-1} \\ 0 & \cdots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}.$$

Hence it holds that (here N_t is assumed to be even for simplicity)

$$\begin{aligned} \mathcal{P}_\alpha^{-1}\mathcal{K} &= \mathcal{I} - \mathcal{K}^{-1}UZ^{-1}V^{\mathsf{T}} \\ &= \mathcal{I} - \begin{bmatrix} K_0^{-1} & & & & \\ K_1^{-1} & K_0^{-1} & & & \\ K_2^{-1} & K_1^{-1} & K_0^{-1} & & \\ \vdots & \ddots & \ddots & \ddots & \\ K_{N_t-1}^{-1} & \cdots & K_2^{-1} & K_1^{-1} & K_0^{-1} \end{bmatrix} \begin{bmatrix} 0 & \cdots & 0 & Z_{1,1}^{-1} & Z_{1,2}^{-1} \\ 0 & \cdots & 0 & Z_{2,1}^{-1} & Z_{2,2}^{-1} \\ 0 & \cdots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \\ &= \mathcal{I} - \begin{bmatrix} \mathbf{0} & \dots & \mathbf{0} & \begin{bmatrix} K_0^{-1} & 0 \\ K_1^{-1} & K_0^{-1} \end{bmatrix} \begin{bmatrix} Z_{1,1}^{-1} & Z_{1,2}^{-1} \\ Z_{2,1}^{-1} & Z_{2,2}^{-1} \end{bmatrix} \\ \vdots & \vdots & \vdots & \vdots \\ \mathbf{0} & \dots & \mathbf{0} & \begin{bmatrix} K_{N_t-4}^{-1} & K_{N_t-5}^{-1} \\ K_{N_t-3}^{-1} & K_{N_t-4}^{-1} \end{bmatrix} \begin{bmatrix} Z_{1,1}^{-1} & Z_{1,2}^{-1} \\ Z_{2,1}^{-1} & Z_{2,2}^{-1} \end{bmatrix} \\ \mathbf{0} & \dots & \mathbf{0} & \begin{bmatrix} K_{N_t-2}^{-1} & K_{N_t-3}^{-1} \\ K_{N_t-1}^{-1} & K_{N_t-2}^{-1} \end{bmatrix} \begin{bmatrix} Z_{1,1}^{-1} & Z_{1,2}^{-1} \\ Z_{2,1}^{-1} & Z_{2,2}^{-1} \end{bmatrix} \end{bmatrix}, \end{aligned}$$

where $\mathbf{0} \in \mathbb{R}^{2N_x \times 2N_x}$ is a zero matrix. If N_t is even, the top-right corner block is of the rectangular form $K_0^{-1} \begin{bmatrix} Z_{1,1}^{-1} & Z_{1,2}^{-1} \end{bmatrix}$ and the following analysis remains valid upon minimal modification if necessary.

By the above expression of Z in (2.3), we get

$$\begin{aligned} &\begin{bmatrix} K_{N_t-2}^{-1} & K_{N_t-3}^{-1} \\ K_{N_t-1}^{-1} & K_{N_t-2}^{-1} \end{bmatrix} \begin{bmatrix} Z_{1,1}^{-1} & Z_{1,2}^{-1} \\ Z_{2,1}^{-1} & Z_{2,2}^{-1} \end{bmatrix} \\ &= I_{2x} - \underbrace{\tau^2\alpha^{-1} \begin{bmatrix} L^{-1} & 2L^{-2} \\ 0 & L^{-1} \end{bmatrix} \begin{bmatrix} Z_{1,1}^{-1} & Z_{1,2}^{-1} \\ Z_{2,1}^{-1} & Z_{2,2}^{-1} \end{bmatrix}}_{=:Y} =: I_{2x} - Y, \end{aligned} \tag{2.4}$$

where $I_{2x} \in \mathbb{R}^{2N_x \times 2N_x}$ is an identity matrix. Therefore, $(N_t - 2)N_x$ eigenvalues of $\mathcal{P}_\alpha^{-1}\mathcal{K}$ are exactly one and the remaining $2N_x$ eigenvalues are determined by its last block, that is Y.

Next, we will analyze the eigenvalues of Y. For this purpose, we write

$$\begin{aligned} \mathcal{K} = \frac{1}{\tau^2}\left(B_1 \otimes L - B_2 \otimes 2I_x\right) &= \frac{1}{\tau^2}(I_t \otimes L)\underbrace{\left(B_1 \otimes I_x - B_2 \otimes 2L^{-1}\right)}_{=:\mathcal{J}} \\ &=: \frac{1}{\tau^2}(I_t \otimes L)\mathcal{J}, \end{aligned} \tag{2.5}$$

where

$$\mathcal{J} := \begin{bmatrix} I_x & & & & \\ -2L^{-1} & I_x & & & \\ I_x & -2L^{-1} & I_x & & \\ & \ddots & \ddots & \ddots & \\ & & I_x & -2L^{-1} & I_x \end{bmatrix}.$$

Clearly, the matrix $\mathcal{J}$ is invertible and we explicitly write the inverse of $\mathcal{J}^{-1}$ as

$$\mathcal{J}^{-1} = \begin{bmatrix} J_0^{-1} & & & \\ J_1^{-1} & J_0^{-1} & & \\ \vdots & \ddots & \ddots & \\ J_{N_t-1}^{-1} & \cdots & J_1^{-1} & J_0^{-1} \end{bmatrix}, \tag{2.6}$$

where J_n^{-1} denotes the nth main diagonal block of $\mathcal{J}^{-1}$. Using the block triangular Toeplitz structure, a simple inductive argument can be used to show the following recurrence relationship:

$$J_{n+1}^{-1} = 2L^{-1}J_n^{-1} - J_{n-1}^{-1}, \quad n = 1, 2, \ldots, N_t - 1, \tag{2.7}$$

starting with $J_0^{-1} = I_x$ and $J_1^{-1} = 2L^{-1}$. Using this recursion formula, we can generate all the diagonal blocks of J^{-1}. For instance, we can easily get $J_2^{-1} = 4L^{-2} - I_x$ and $J_3^{-1} = 8L^{-3} - 4L^{-1}$. Applying the well-known characteristic root technique [34, p. 172] for solving the recursion, we can derive a formal explicit expression

$$J_n^{-1} = \Theta_1 \left[L^{-1} - (L^{-2} - I_x)^{\frac{1}{2}}\right]^n + \Theta_2 \left[L^{-1} + (L^{-2} - I_x)^{\frac{1}{2}}\right]^n, \tag{2.8}$$

where

$$\Theta_1 = \frac{1}{2}\left[I_x - (I_x - L^2)^{-\frac{1}{2}}\right], \quad \Theta_2 = \frac{1}{2}\left[I_x + (I_x - L^2)^{-\frac{1}{2}}\right].$$

We emphasize that in the actual implementation there is no need to sequentially compute J_n^{-1}, which is defined only for analysis purpose. There holds $J_n^{-1}L^{-1} = L^{-1}J_n^{-1}$ since J_n^{-1} is a matrix polynomial of L^{-1}.

Since $(-\Delta_h)$ is symmetric positive definite, $L = I_x - \frac{\tau^2}{2}\Delta_h$ is also symmetric positive definite with $\lambda(L) > 1$ and the principal square roots $(L^{-2} - I_x)^{\frac{1}{2}} = \mathrm{i}(I_x - L^{-2})^{\frac{1}{2}}$ and $(I_x - L^2)^{-\frac{1}{2}} = \mathrm{i}(L^2 - I_x)^{-\frac{1}{2}}$ are well defined by introducing imaginary unit $\mathrm{i} = \sqrt{-1}$. By the definition of $\mathcal{J}$ in (2.5), we have $\mathcal{K}^{-1} = \tau^2\mathcal{J}^{-1}(I_t \otimes L^{-1})$, which

implies that the nth main block diagonals of $\mathcal{K}^{-1}$ are $K_n^{-1} = \tau^2 J_n^{-1} L^{-1}$. Therefore,

(2.9)

$$
\begin{aligned}
Y &= \tau^2\alpha^{-1}\begin{bmatrix} L^{-1} & 2L^{-2} \\ 0 & L^{-1} \end{bmatrix}\left(\tau^2\alpha^{-1}\begin{bmatrix} L^{-1} & 2L^{-2} \\ 0 & L^{-1} \end{bmatrix} + \begin{bmatrix} K_{N_t-2}^{-1} & K_{N_t-3}^{-1} \\ K_{N_t-1}^{-1} & K_{N_t-2}^{-1} \end{bmatrix}\right)^{-1} \\
&= \left(I_{2x} + \frac{\alpha}{\tau^2}\begin{bmatrix} K_{N_t-2}^{-1} & K_{N_t-3}^{-1} \\ K_{N_t-1}^{-1} & K_{N_t-2}^{-1} \end{bmatrix}\begin{bmatrix} L & -2I_x \\ 0 & L \end{bmatrix}\right)^{-1} \\
&= \left(I_{2x} + \alpha\begin{bmatrix} J_{N_t-2}^{-1} & J_{N_t-3}^{-1} \\ J_{N_t-1}^{-1} & J_{N_t-2}^{-1} \end{bmatrix}\begin{bmatrix} I_x & -2L^{-1} \\ 0 & I_x \end{bmatrix}\right)^{-1} \\
&= \left(I_{2x} + \alpha\begin{bmatrix} J_{N_t-2}^{-1} & J_{N_t-2}^{-1}(-2L^{-1}) + J_{N_t-3}^{-1} \\ J_{N_t-1}^{-1} & J_{N_t-1}^{-1}(-2L^{-1}) + J_{N_t-2}^{-1} \end{bmatrix}\right)^{-1} \\
&= \left(I_{2x} + \alpha\underbrace{\begin{bmatrix} J_{N_t-2}^{-1} & -J_{N_t-1}^{-1} \\ J_{N_t-1}^{-1} & -J_{N_t}^{-1} \end{bmatrix}}_{=:X}\right)^{-1} =: (I_{2x} + \alpha X)^{-1},
\end{aligned}
$$

where $J_{N_t}^{-1} := 2L^{-1}J_{N_t-1}^{-1} - J_{N_t-2}^{-1}$ is defined from the same recursion formula (2.7).

To further estimate the eigenvalues of X, we need to discuss the eigenvalues of $J_{N_t-2}^{-1}$, $J_{N_t-1}^{-1}$, and $J_{N_t}^{-1}$. Let $L = Q\Lambda Q^{\mathsf{T}}$ be the spectrum decomposition of the real symmetric matrix $L(= I_x - \frac{\tau^2}{2}\Delta_h)$ with an orthogonal matrix Q and a real diagonal matrix $\Lambda = \operatorname{diag}(\lambda_1, \ldots, \lambda_{N_x})$ including all the sorted (increasing) eigenvalues with $1 < \lambda_1 < \cdots < \lambda_{N_x}$. Define $\theta_j := \arctan\left(\sqrt{\lambda_j^2 - 1}\right) \in (0, \pi/2)$ and hence $\cos\theta_j = \lambda_j^{-1} \in (0, 1)$. Using the similarity transformation with Q, from the expression (2.8) of J_n^{-1} we obtain

$$
\begin{aligned}
\Sigma_n := Q^T J_n^{-1} Q = &\frac{1}{2}\left[I_x - (I_x - \Lambda^2)^{-\frac{1}{2}}\right]\left[\Lambda^{-1} - (\Lambda^{-2} - I_x)^{\frac{1}{2}}\right]^n \\
&+ \frac{1}{2}\left[I_x + (I_x - \Lambda^2)^{-\frac{1}{2}}\right]\left[\Lambda^{-1} + (\Lambda^{-2} - I_x)^{\frac{1}{2}}\right]^n,
\end{aligned}
$$

which shows that all the N_x real eigenvalues $\mu_{n,j}$ for $j = 1, \ldots, N_x$ of J_n^{-1} are given by

$$
\begin{aligned}
\mu_{n,j} =& \frac{1}{2}\left[1 - (1 - \lambda_j^2)^{-1/2}\right]\left[\lambda_j^{-1} - (\lambda_j^{-2} - 1)^{1/2}\right]^n \\
&+ \frac{1}{2}\left[1 + (1 - \lambda_j^2)^{-1/2}\right]\left[\lambda_j^{-1} + (\lambda_j^{-2} - 1)^{1/2}\right]^n \\
=& \frac{1}{2}\left[1 + \mathrm{i}(\lambda_j^2 - 1)^{-1/2}\right]\left[\lambda_j^{-1} - \mathrm{i}(1 - \lambda_j^{-2})^{1/2}\right]^n \\
&+ \frac{1}{2}\left[1 - \mathrm{i}(\lambda_j^2 - 1)^{-1/2}\right]\left[\lambda_j^{-1} + \mathrm{i}(1 - \lambda_j^{-2})^{1/2}\right]^n \\
=& \frac{1}{2}\left[1 + \mathrm{i}(\lambda_j^2 - 1)^{-1/2}\right][\cos\theta_j - \mathrm{i}\sin\theta_j]^n + \frac{1}{2}\left[1 - \mathrm{i}(\lambda_j^2 - 1)^{-1/2}\right][\cos\theta_j + \mathrm{i}\sin\theta_j]^n \\
=& \frac{1}{2}\left[1 + \mathrm{i}(\lambda_j^2 - 1)^{-1/2}\right]e^{-\mathrm{i}n\theta_j} + \frac{1}{2}\left[1 - \mathrm{i}(\lambda_j^2 - 1)^{-1/2}\right]e^{\mathrm{i}n\theta_j} \\
=& \frac{1}{2}(e^{-\mathrm{i}n\theta_j} + e^{\mathrm{i}n\theta_j}) + (\lambda_j^2 - 1)^{-1/2}\frac{\mathrm{i}}{2}(e^{-\mathrm{i}n\theta_j} - e^{\mathrm{i}n\theta_j}) \\
=& \cos(n\theta_j) + \cot(\theta_j)\,\sin(n\theta_j) = \frac{\sin((n+1)\theta_j)}{\sin\theta_j},
\end{aligned}
$$

where we have used the Euler's formula $e^{\pm \mathrm{i}\theta} = \cos\theta \pm \mathrm{i}\sin\theta$. It is worthwhile to mention that the use of the trigonometric substitution here is crucial to simplify our analysis and reveal some interesting connections.

Now, the eigenvalues of X in (2.9) can be computed from the block matrix with diagonal blocks

$$\begin{bmatrix} Q^{\mathsf{T}} & 0 \\ 0 & Q^{\mathsf{T}} \end{bmatrix} X \begin{bmatrix} Q & 0 \\ 0 & Q \end{bmatrix} = \begin{bmatrix} Q^{\mathsf{T}} & 0 \\ 0 & Q^{\mathsf{T}} \end{bmatrix} \begin{bmatrix} J_{N_t-2}^{-1} & -J_{N_t-1}^{-1} \\ J_{N_t-1}^{-1} & -J_{N_t}^{-1} \end{bmatrix} \begin{bmatrix} Q & 0 \\ 0 & Q \end{bmatrix} = \begin{bmatrix} \Sigma_{N_t-2} & -\Sigma_{N_t-1} \\ \Sigma_{N_t-1} & -\Sigma_{N_t} \end{bmatrix}.$$

Upon a simple permutation in both rows and columns, we obtain a simpler block diagonal structure

$$P^{\mathsf{T}} \begin{bmatrix} \Sigma_{N_t-2} & -\Sigma_{N_t-1} \\ \Sigma_{N_t-1} & -\Sigma_{N_t} \end{bmatrix} P = \mathrm{blkdiag}\left(\Phi_1, \Phi_2, \ldots, \Phi_j, \ldots, \Phi_{N_x}\right).$$

where each block is given by a 2-by-2 nonsymmetric matrix

$$\Phi_j := \begin{bmatrix} \mu_{N_t-2,j} & -\mu_{N_t-1,j} \\ \mu_{N_t-1,j} & -\mu_{N_t,j} \end{bmatrix}.$$

Obviously, the two conjugate eigenvalues of each Φ_j are explicitly given by

$$\nu_j^{\pm} = \frac{1}{2}\left(\mu_{N_t-2,j} - \mu_{N_t,j} \pm \sqrt{(\mu_{N_t-2,j} + \mu_{N_t,j})^2 - 4\mu_{N_t-1,j}^2}\right).$$

From the recursion formula (2.7) we have

$$\mu_{n,j} = 2\lambda_j^{-1}\mu_{n-1,j} - \mu_{n-2,j}, \tag{2.10}$$

which leads to

$$\nu_j^{\pm} = (\mu_{N_t-2,j} - \lambda_j^{-1}\mu_{N_t-1,j}) \pm \mathrm{i}|\mu_{N_t-1,j}|\sqrt{1-\lambda_j^{-2}}.$$

With the trigonometric expressions $\mu_{n,j} = \frac{\sin((n+1)\theta_j)}{\sin\theta_j}$, $\sqrt{1-\lambda_j^{-2}} = \sin\theta_j > 0$, and $\lambda_j^{-1} = \cos\theta_j$, we get

$$\begin{aligned} \nu_j^{\pm} &= \left(\frac{\sin((N_t-1)\theta_j)}{\sin\theta_j} - \cos\theta_j \frac{\sin(N_t\theta_j)}{\sin\theta_j}\right) \pm \mathrm{i}\frac{|\sin(N_t\theta_j)|}{\sin\theta_j}\sin\theta_j \\ &= \frac{-\cos(N_t\theta_j)\sin\theta_j}{\sin\theta_j} \pm \mathrm{i}|\sin(N_t\theta_j)| = -e^{\pm \mathrm{i}N_t\theta_j}, \end{aligned} \tag{2.11}$$

where we used the identity $\sin(a-b) = \sin a\cos b - \cos a\sin b$. The $2N_x$ *distinct* eigenvalues of X are $\{\nu_j^{\pm}\}_{j=1}^{N_x}$ and hence the $2N_x$ *distinct* eigenvalues of $Y = (I_{2x} + \alpha X)^{-1}$ in (2.9) are given by $\sigma(Y) = \{\frac{1}{1+\alpha\nu_j^{\pm}}\}_{j=1}^{N_x}$.

In summary, the above eigenvalue analysis implies

$$\sigma(\mathcal{P}_\alpha^{-1}\mathcal{K}) = \underbrace{\{1,1,\ldots,1\}}_{(N_t-2)N_x} \cup \sigma(Y) = \underbrace{\{1,1,\ldots,1\}}_{(N_t-2)N_x} \cup \left\{\frac{1}{1+\alpha\nu_j^{\pm}}\right\}_{j=1}^{N_x},$$

where $\nu_j^{\pm}$ are given by (2.11). For $\alpha \in (0,1)$ (the case $\alpha = 1$ will be treated separately below), we have

$$\frac{\alpha}{1+\alpha} \le \left|\frac{1}{1+\alpha\nu_j^{\pm}} - 1\right| = \frac{\alpha}{|1+\alpha\nu_j^{\pm}|} \le \frac{\alpha}{1-\alpha|\nu_j^{\pm}|} = \frac{\alpha}{1-\alpha},$$

where we have used $|\nu_j^{\pm}| = 1$ (which can be deduced from (2.11)) and the triangle inequality

$$0 < 1 - \alpha = 1 - \alpha|\nu_j^{\pm}| \le |1 + \alpha\nu_j^{\pm}| \le 1 + \alpha|\nu_j^{\pm}| = 1 + \alpha.$$

This implies that the spectrum of $\mathcal{P}_\alpha^{-1}\mathcal{K}$, denoted by $\sigma(\mathcal{P}_\alpha^{-1}\mathcal{K})$, can be bounded by an annulus $\mathbb{A}_\alpha$ centered at $(1, 0)$ with outer radius $\frac{\alpha}{1-\alpha}$ and inner radius $\frac{\alpha}{1+\alpha}$, that is

$$\sigma(\mathcal{P}_\alpha^{-1}\mathcal{K}) \subset \mathbb{A}_\alpha := \left\{ z \in \mathbb{C} : \frac{\alpha}{1+\alpha} \le |z - 1| \le \frac{\alpha}{1-\alpha} \right\}.$$

Clearly, both the outer and inner radii are independent of mesh step sizes. Moreover, both the outer radius $\frac{\alpha}{1-\alpha}$ and the inner radius $\frac{\alpha}{1+\alpha}$ are monotone increasing functions of $\alpha \in (0, 1)$. Hence, a smaller α would lead to a smaller and thinner annulus, indicating all nonunit eigenvalues of $\mathcal{P}_\alpha^{-1}\mathcal{K}$ become more clustered around 1. In Figure 2.1, we plot the eigenvalues of $\mathcal{P}_\alpha^{-1}\mathcal{K}$ together with the estimated annulus for different α and N_t. Clearly, the nonunit eigenvalues are tightly and uniformly bounded within the annulus, whose radii and bandwidth (i.e., the difference between the outer and inner radii) are independent of mesh step sizes and decrease proportionally as α gets smaller.

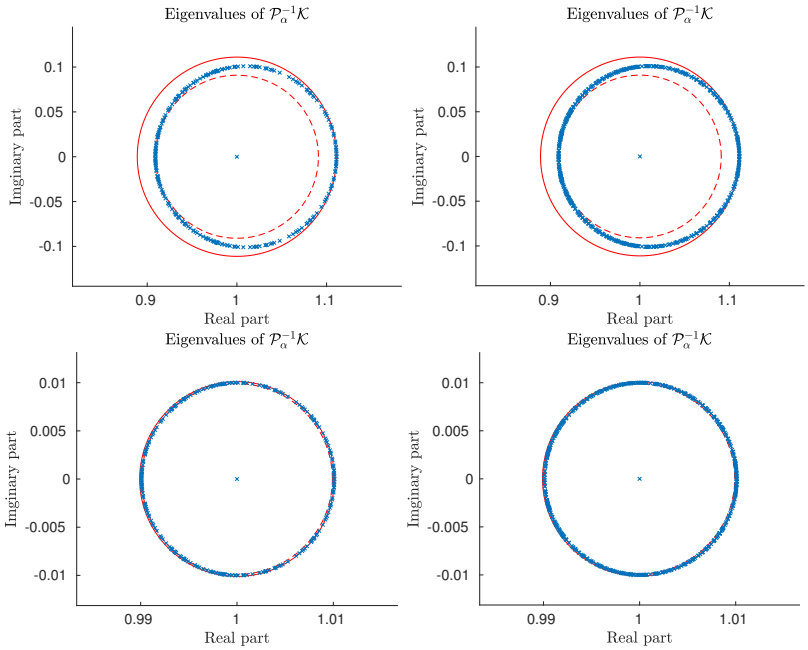

FIG. 2.1. *The eigenvalues (in blue cross) distribution of* $\mathcal{P}_\alpha^{-1}\mathcal{K}$ *with* $\alpha = 0.1$ *(top row) and* $\alpha = 0.01$ *(bottom row) and the estimated annulus* $\mathbb{A}_\alpha$ *(in red) (left column:* $N_t = 128$ *and right column:* $N_t = 256$*). Notice that the plotted annulus in the bottom row has a much smaller radius and bandwidth, which is zoomed-in for better visualization. (Figure in color online.)*

The limiting case with $\alpha = 1$. For the special case with $\alpha = 1$ (i.e., the preconditioner $\mathcal{P}$), the estimated annulus becomes a vertical line segment centered at

$(1/2, 0)$ and a point $(1, 0)$, since we have

$$\begin{aligned}\omega_j^{\pm} &= \frac{1}{1+\nu_j^{\pm}} = \frac{1}{1-\cos(N_t\theta_j) \pm \mathrm{i}|\sin(N_t\theta_j)|} = \frac{1-\cos(N_t\theta_j) \pm \mathrm{i}|\sin(N_t\theta_j)|}{(1-\cos(N_t\theta_j))^2+\sin^2(N_t\theta_j)} \\ &= \frac{1-\cos(N_t\theta_j) \pm \mathrm{i}|\sin(N_t\theta_j)|}{2-2\cos(N_t\theta_j)} = \frac{1}{2}\left(1 \pm \mathrm{i}\frac{\sin(N_t\theta_j)}{1-\cos(N_t\theta_j)}\right) \\ &= \frac{1}{2}\left(1 \pm \mathrm{i}\cot\left(\frac{N_t\theta_j}{2}\right)\right),\end{aligned}$$

which explains $\Re(\omega_j^{\pm}) = 1/2$. This also shows the imaginary part $\Im(\omega_j^{\pm}) = \frac{1}{2}\cot(N_t\theta_j/2)$ with $\theta_j \in (0, \pi/2)$ is not uniformly bounded (whenever $N_t\theta_j \approx 2k\pi$), and hence the preconditioner $\mathcal{P}_\alpha$ with $\alpha = 1$ *may* give a mesh-dependent convergence rate. Figure 2.2 shows the distribution of the eigenvalues of $\mathcal{P}_\alpha^{-1}\mathcal{K}$ with $\alpha = 1$, where we observed the range of the imaginary part $\Im(\omega_j^{\pm})$ is dramatically enlarged as the mesh is refined.

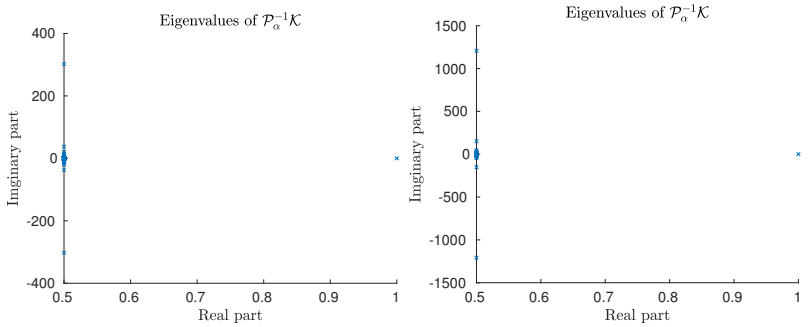

FIG. 2.2. *The eigenvalues distribution of $\mathcal{P}_\alpha^{-1}\mathcal{K}$ with $\alpha = 1$ (left: $N_t = 128$ and right: $N_t = 256$).*

To summarize, we get the following result regarding the spectrum of the preconditioned matrix $\mathcal{P}_\alpha^{-1}\mathcal{K}$.

THEOREM 2.1. *The eigenvalues of the matrix $\mathcal{P}_\alpha^{-1}\mathcal{K}$ are explicitly given by*

$$\sigma(\mathcal{P}_\alpha^{-1}\mathcal{K}) = \underbrace{\{1, 1, \ldots, 1\}}_{(N_t-2)N_x} \cup \left\{\frac{1}{1-\alpha e^{\pm \mathrm{i}N_t\theta_j}}\right\}_{j=1}^{N_x},$$

where $\theta_j := \arctan\left(\sqrt{\lambda_j^2 - 1}\right) \in (0, \frac{\pi}{2})$ with λ_j being an eigenvalue of $L = I_x - \frac{\tau^2}{2}\Delta_h$. Moreover, we have

1. *If $\alpha \in (0, 1)$, then*

$$\sigma(\mathcal{P}_\alpha^{-1}\mathcal{K}) \subset \mathbb{A}_\alpha := \left\{z \in \mathbb{C} : \frac{\alpha}{1+\alpha} \le |z-1| \le \frac{\alpha}{1-\alpha}\right\}.$$

2. *If $\alpha = 1$, then*

$$\sigma(\mathcal{P}_\alpha^{-1}\mathcal{K}) = \{1\} \cup \left\{\frac{1}{2} \pm \frac{1}{2}\mathrm{i}\cot\left(\frac{N_t\theta_j}{2}\right)\right\}_{j=1}^{N_x}.$$

Let y_h^k denote the approximate solution at the kth iteration of GMRES applied to the preconditioned system $\mathcal{P}_\alpha^{-1}\mathcal{K}y_h = \mathcal{P}_\alpha^{-1}b_h$ with a given initial guess y_h^0. Then

the residual vector $r^k := \mathcal{P}_\alpha^{-1}(b_h - \mathcal{K}y_h^k)$ satisfies the minimal residual property [57]

$$\|r^k\|_2 = \min_{p\in\mathbb{P}_k, p(0)=1} \|p(\mathcal{P}_\alpha^{-1}\mathcal{K})r^0\|_2 \le \left(\min_{p\in\mathbb{P}_k, p(0)=1} \|p(\mathcal{P}_\alpha^{-1}\mathcal{K})\|_2\right)\|r^0\|_2, \tag{2.12}$$

where $\mathbb{P}_k$ is the set of polynomials of degree k or less. For a given initial r^0, the last "worst-case" inequality can be highly overestimated in some cases [16, 63]. It is usually difficult to directly estimate the norm $\|p(\mathcal{P}_\alpha^{-1}\mathcal{K})\|_2$ from only the spectrum information of the nonnormal matrix $\mathcal{P}_\alpha^{-1}\mathcal{K}$. However, if $\mathcal{P}_\alpha^{-1}\mathcal{K}$ is diagonalizable as $\mathcal{P}_\alpha^{-1}\mathcal{K} = \mathcal{S}\Upsilon\mathcal{S}^{-1}$ with Υ being a diagonal matrix containing all the eigenvalues in $\sigma(\mathcal{P}_\alpha^{-1}\mathcal{K})$, it is natural to obtain a convenient upper bound

$$\|p(\mathcal{P}_\alpha^{-1}\mathcal{K})\|_2 = \|p(\mathcal{S}\Upsilon\mathcal{S}^{-1})\|_2 = \|\mathcal{S}p(\Upsilon)\mathcal{S}^{-1}\|_2 \le \|\mathcal{S}\|_2\|p(\Upsilon)\|_2\|\mathcal{S}^{-1}\|_2 = \mathrm{cond}(\mathcal{S})\left(\max_{\lambda\in\sigma(\mathcal{P}_\alpha^{-1}\mathcal{K})}|p(\lambda)|\right),$$

where $\mathrm{cond}(\mathcal{S}) := \|\mathcal{S}\|_2\|\mathcal{S}^{-1}\|_2$ denotes the 2-norm condition number of eigenvector matrix $\mathcal{S}$ and $\sigma(\mathcal{P}_\alpha^{-1}\mathcal{K})$ is explicitly given in Theorem 2.1. In particular, this bound indicates that a highly clustered spectrum away from zero will likely give a fast convergence rate if $\mathrm{cond}(\mathcal{S})$ is not too large. Following the arguments in [42, Thm. 3], we can prove that $\mathcal{P}_\alpha^{-1}\mathcal{K}$ is diagonalizable, and hence the above convergence bound is indeed applicable. Nevertheless, it is still difficult to estimate $\mathrm{cond}(\mathcal{S})$, which motivates us to use an alternative approach to obtain a different convergence rate bound, as explained in the next section.

THEOREM 2.2. *The preconditioned matrix $\mathcal{P}_\alpha^{-1}\mathcal{K}$ is diagonalizable for any $\alpha \in (0, 1]$.*

Proof. Since all the eigenvalues of Y in (2.4) are distinct, it is diagonalizable and we let $Y = S\Theta S^{-1}$ be its diagonalization. Here $(\Theta - I_{2x})^{-1}$ exists since Y has no unit eigenvalue, i.e., $1 \notin \sigma(Y)$. We recall that

$$\begin{aligned}
\mathcal{P}_\alpha^{-1}\mathcal{K} - \mathcal{I} &= \begin{bmatrix}
\mathbf{0} & \dots & \mathbf{0} & -\begin{bmatrix} K_0^{-1} & 0 \\ K_1^{-1} & K_0^{-1} \end{bmatrix}\begin{bmatrix} Z_{1,1}^{-1} & Z_{1,2}^{-1} \\ Z_{2,1}^{-1} & Z_{2,2}^{-1} \end{bmatrix} \\
\vdots & \vdots & \vdots & \vdots \\
\mathbf{0} & \dots & \mathbf{0} & -\begin{bmatrix} K_{N_t-4}^{-1} & K_{N_t-5}^{-1} \\ K_{N_t-3}^{-1} & K_{N_t-4}^{-1} \end{bmatrix}\begin{bmatrix} Z_{1,1}^{-1} & Z_{1,2}^{-1} \\ Z_{2,1}^{-1} & Z_{2,2}^{-1} \end{bmatrix} \\
\mathbf{0} & \dots & \mathbf{0} & -\begin{bmatrix} K_{N_t-2}^{-1} & K_{N_t-3}^{-1} \\ K_{N_t-1}^{-1} & K_{N_t-2}^{-1} \end{bmatrix}\begin{bmatrix} Z_{1,1}^{-1} & Z_{1,2}^{-1} \\ Z_{2,1}^{-1} & Z_{2,2}^{-1} \end{bmatrix}
\end{bmatrix} \\
&=: \begin{bmatrix}
\mathbf{0} & \dots & \mathbf{0} & W_1 \\
\mathbf{0} & \ddots & \mathbf{0} & W_3 \\
\vdots & \vdots & \vdots & \vdots \\
\mathbf{0} & \dots & \mathbf{0} & W_{N_t-3} \\
\mathbf{0} & \dots & \mathbf{0} & W_{N_t-1}
\end{bmatrix},
\end{aligned}$$

where by (2.4) it holds that

$$W_{N_t-1} = Y - I_{2x} = S(\Theta - I_{2x})S^{-1}.$$

Let $X_n := W_n S(\Theta - I_{2x})^{-1}$. A simple calculation yields the following diagonalization:

$$\mathcal{P}_\alpha^{-1}\mathcal{K} - \mathcal{I} = \underbrace{\begin{bmatrix} I_{2x} & \mathbf{0} & \dots & \mathbf{0} & X_1 \\ \mathbf{0} & I_{2x} & \ddots & \mathbf{0} & X_3 \\ \vdots & \ddots & \vdots & \vdots & \vdots \\ \mathbf{0} & \mathbf{0} & \dots & I_{2x} & X_{N_t-3} \\ \mathbf{0} & \mathbf{0} & \dots & \mathbf{0} & S \end{bmatrix}}_{=:\mathcal{S}} \underbrace{\begin{bmatrix} \mathbf{0} & \mathbf{0} & \dots & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \ddots & \mathbf{0} & \mathbf{0} \\ \vdots & \ddots & \vdots & \vdots & \vdots \\ \mathbf{0} & \mathbf{0} & \dots & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \dots & \mathbf{0} & (\Theta - I_{2x}) \end{bmatrix}}_{=:\widehat{\Theta}}$$

$$\underbrace{\begin{bmatrix} I_{2x} & \mathbf{0} & \dots & \mathbf{0} & X_1 \\ \mathbf{0} & I_{2x} & \ddots & \mathbf{0} & X_3 \\ \vdots & \ddots & \vdots & \vdots & \vdots \\ \mathbf{0} & \mathbf{0} & \dots & I_{2x} & X_{N_t-3} \\ \mathbf{0} & \mathbf{0} & \dots & \mathbf{0} & S \end{bmatrix}^{-1}}_{=:\mathcal{S}^{-1}},$$

which leads to the desired diagonalization

$$\mathcal{P}_\alpha^{-1}\mathcal{K} = \mathcal{S}(\mathcal{I} + \widehat{\Theta})\mathcal{S}^{-1} =: \mathcal{S}\Upsilon\mathcal{S}^{-1}, \tag{2.13}$$

where the diagonal entries of Υ consist of the spectrum set $\sigma(\mathcal{P}_\alpha^{-1}\mathcal{K})$. ∎

3. Convergence rate of preconditioned GMRES and two stationary iterative methods. In view of (2.12), the convergence rate of GMRES is not conclusively determined by the eigenvalues of the preconditioned system alone (see, e.g., [28, 43, 64]), where the condition number of the corresponding eigenvector matrix (i.e., cond($\mathcal{S}$) in (2.13)) and the right-hand-side also play important roles. But it is usually very difficult to completely characterize all these influences. Another way of determining the convergence behavior of GMRES is to find the degree of its minimal polynomial, which is also difficult to achieve in our case. There are at least two possible alternative Krylov subspace methods whose convergence rates depend completely on the spectrum information that is relatively easier to treat. First, one may use the preconditioned conjugate gradient method applied to the normal equations (CGNE) [5, 46], which essentially applies the preconditioned conjugate gradients (PCG) method to solve the symmetric positive definite system $\mathcal{K}\mathcal{K}^{\mathsf{T}} z_h = b_h$ with the *squared* preconditioner $\mathcal{P}_\alpha\mathcal{P}_\alpha^{\mathsf{T}}$ and then get $y_h = \mathcal{K}^{\mathsf{T}} z_h$. In this case, the convergence rate of the preconditioned CGNE is completely determined by the distribution of the eigenvalues of $(\mathcal{P}_\alpha\mathcal{P}_\alpha^{\mathsf{T}})^{-1}\mathcal{K}\mathcal{K}^{\mathsf{T}}$ or $(\mathcal{P}_\alpha^{-1}\mathcal{K})(\mathcal{P}_\alpha^{-1}\mathcal{K})^{\mathsf{T}}$ or, equivalently, the singular values of $\mathcal{P}_\alpha^{-1}\mathcal{K}$. In particular, if the singular values of $\mathcal{P}_\alpha^{-1}\mathcal{K}$ are clustered near 1 without outliers, then the preconditioned CGNE converges superlinearly [5, 46]. Even with a finite number of outliers, the preconditioned CGNE converges linearly and the number of iterations is independent of the system size. Therefore, we may switch to estimate the distribution of the singular values of $\mathcal{P}_\alpha^{-1}\mathcal{K}$. Unfortunately, based on our numerical tests, this pathway again seems to be challenging to obtain any useful estimates, since the singular values of $\mathcal{P}_\alpha^{-1}\mathcal{K}$ do not admit explicit expressions and they are also not uniformly bounded with respect to the mesh sizes. Second, one may use preconditioned MINRES upon symmetrizing the involved Toeplitz matrices [52, 53], where the construction and the analysis of a positive definite block α-circulant preconditioner as in our problem seems to be extremely complicated.

In the following subsections, we will present a mesh-independent convergence rate of the preconditioned GMRES and also analyze two stationary iteration methods

based on our proposed preconditioner. Both convergence analyses largely rely on the eigenvalue estimate results in Theorem 2.1.

3.1. A restrictive convergence rate of preconditioned GMRES. Let $\mathbb{H}(A) := (A + A^{\mathsf{T}})/2$ be the symmetric part of A. According to the convergence bound of preconditioned GMRES [12, 61] for nonsymmetric but positive definite systems, we have the following relative residual norm estimate:

$$\frac{\|r^k\|_2}{\|r^0\|_2} \le \left(1 - \frac{\lambda_{\min}^2(\mathbb{H}(\mathcal{P}_\alpha^{-1}\mathcal{K}))}{\|\mathcal{P}_\alpha^{-1}\mathcal{K}\|_2^2}\right)^{k/2}, \tag{3.1}$$

where $r^k = \mathcal{P}_\alpha^{-1}(b_h - \mathcal{K}y_h^k)$ with y_h^k being the approximate solution at k—the GMRES iteration and r^0 being the initial residual. We highlight that such a (possibly nonsharp) convergence rate requires the symmetric part of the preconditioned matrix (i.e., $\mathbb{H}(\mathcal{P}_\alpha^{-1}\mathcal{K})$) to be positive definite, which was proved in Theorem 2.1.

From $\mathcal{P}_\alpha = \mathcal{K} + \alpha\mathcal{R}$, we have

$$\|\mathcal{P}_\alpha^{-1}\mathcal{K}\|_2 = \|\mathcal{I} - \alpha\mathcal{P}_\alpha^{-1}\mathcal{R}\|_2 \le 1 + \alpha\|\mathcal{P}_\alpha^{-1}\mathcal{R}\|_2 \le 1 + \alpha\|\mathcal{P}_\alpha^{-1}\mathcal{K}\|_2\|\mathcal{K}^{-1}\mathcal{R}\|_2,$$

where the quality $\|\mathcal{P}_\alpha^{-1}\mathcal{R}\|_2$ is much more difficult to estimate directly. Assuming that $\alpha \in (0, 1]$ is sufficiently small such that $\alpha\|\mathcal{K}^{-1}\mathcal{R}\|_2 < 1$, it holds that

$$\|\mathcal{P}_\alpha^{-1}\mathcal{K}\|_2 \le \frac{1}{1 - \alpha\|\mathcal{K}^{-1}\mathcal{R}\|_2}. \tag{3.2}$$

With the same assumption $\alpha\|\mathcal{K}^{-1}\mathcal{R}\|_2 < 1$, by the Neumann series expansion [44, p. 618] we have

(3.3)

$$\begin{aligned}
\|\mathcal{P}_\alpha^{-1}\mathcal{K} - \mathcal{I}\|_2 &= \|(\mathcal{K} + \alpha\mathcal{R})^{-1}\mathcal{K} - \mathcal{I}\|_2 = \|(\mathcal{I} + \alpha\mathcal{K}^{-1}\mathcal{R})^{-1} - I\|_2 \\
&= \|\textstyle\sum_{l=0}^{\infty}(-\alpha\mathcal{K}^{-1}\mathcal{R})^l - I\|_2 \\
&= \|\textstyle\sum_{l=1}^{\infty}(-\alpha\mathcal{K}^{-1}\mathcal{R})^l\|_2 \le \sum_{l=1}^{\infty}(\alpha\|\mathcal{K}^{-1}\mathcal{R}\|_2)^l = \frac{\alpha\|\mathcal{K}^{-1}\mathcal{R}\|_2}{1 - \alpha\|\mathcal{K}^{-1}\mathcal{R}\|_2}.
\end{aligned}$$

If assuming $\alpha\|\mathcal{K}^{-1}\mathcal{R}\|_2 < 1/2$, with the inequalities $|\lambda_{\min}(\mathbb{H}(A))| \le \|\mathbb{H}(A)\|_2 \le \|A\|_2$ [32, p. 160], we get

$$\begin{aligned}
\lambda_{\min}(\mathbb{H}(\mathcal{P}_\alpha^{-1}\mathcal{K})) &= \lambda_{\min}(\mathcal{I} + \mathbb{H}(\mathcal{P}_\alpha^{-1}\mathcal{K} - \mathcal{I})) = 1 + \lambda_{\min}(\mathbb{H}(\mathcal{P}_\alpha^{-1}\mathcal{K} - \mathcal{I})) \\
&\ge 1 - \|\mathbb{H}(\mathcal{P}_\alpha^{-1}\mathcal{K} - \mathcal{I})\|_2 \\
&\ge 1 - \|\mathcal{P}_\alpha^{-1}\mathcal{K} - \mathcal{I}\|_2 \ge 1 - \frac{\alpha\|\mathcal{K}^{-1}\mathcal{R}\|_2}{1 - \alpha\|\mathcal{K}^{-1}\mathcal{R}\|_2} > 0.
\end{aligned} \tag{3.4}$$

Hence, combining (3.2) and (3.4) we obtained the following GMRES convergence rate estimate:

$$\frac{\|r^k\|_2}{\|r^0\|_2} \le \left(1 - \left[\frac{1 - \frac{\alpha\|\mathcal{K}^{-1}\mathcal{R}\|_2}{1-\alpha\|\mathcal{K}^{-1}\mathcal{R}\|_2}}{\frac{1}{1-\alpha\|\mathcal{K}^{-1}\mathcal{R}\|_2}}\right]^2\right)^{k/2} = \left(1 - \left(1 - 2\alpha\|\mathcal{K}^{-1}\mathcal{R}\|_2\right)^2\right)^{k/2}. \tag{3.5}$$

Next, we estimate an upper bound for $\|\mathcal{K}^{-1}\mathcal{R}\|_2$ from its definition (recall $K_n^{-1} = \tau^2 J_n^{-1} L^{-1}$):

$$\mathcal{K}^{-1}\mathcal{R} = \frac{1}{\tau^2}\begin{bmatrix} K_0^{-1} & & & & \\ K_1^{-1} & K_0^{-1} & & & \\ K_2^{-1} & K_1^{-1} & K_0^{-1} & & \\ \vdots & \ddots & \ddots & \ddots & \\ K_{N_t-1}^{-1} & \cdots & K_2^{-1} & K_1^{-1} & K_0^{-1} \end{bmatrix}\begin{bmatrix} 0 & \cdots & 0 & L & -2I_x \\ & 0 & \cdots & 0 & L \\ & & 0 & \cdots & 0 \\ & & & \ddots & \vdots \\ & & & & 0 \end{bmatrix}$$

$$= \begin{bmatrix} \mathbf{0} & \dots & \mathbf{0} & \begin{bmatrix} J_0^{-1} & 0 \\ J_1^{-1} & J_0^{-1} \end{bmatrix} \begin{bmatrix} I_x & -2L^{-1} \\ 0 & I_x \end{bmatrix} \\ \vdots & \vdots & \vdots & \vdots \\ \mathbf{0} & \dots & \mathbf{0} & \begin{bmatrix} J_{N_t-4}^{-1} & J_{N_t-5}^{-1} \\ J_{N_t-3}^{-1} & J_{N_t-4}^{-1} \end{bmatrix} \begin{bmatrix} I_x & -2L^{-1} \\ 0 & I_x \end{bmatrix} \\ \mathbf{0} & \dots & \mathbf{0} & \begin{bmatrix} J_{N_t-2}^{-1} & J_{N_t-3}^{-1} \\ J_{N_t-1}^{-1} & J_{N_t-2}^{-1} \end{bmatrix} \begin{bmatrix} I_x & -2L^{-1} \\ 0 & I_x \end{bmatrix} \end{bmatrix}$$

$$= \begin{bmatrix} \mathbf{0} & \dots & \mathbf{0} & \begin{bmatrix} J_0^{-1} & -J_1^{-1} \\ J_1^{-1} & -J_2^{-1} \end{bmatrix} \\ \vdots & \vdots & \vdots & \vdots \\ \mathbf{0} & \dots & \mathbf{0} & \begin{bmatrix} J_{N_t-4}^{-1} & -J_{N_t-3}^{-1} \\ J_{N_t-3}^{-1} & -J_{N_t-2}^{-1} \end{bmatrix} \\ \mathbf{0} & \dots & \mathbf{0} & \begin{bmatrix} J_{N_t-2}^{-1} & -J_{N_t-1}^{-1} \\ J_{N_t-1}^{-1} & -J_{N_t}^{-1} \end{bmatrix} \end{bmatrix} =: \begin{bmatrix} \mathbf{0} & \dots & \mathbf{0} & G_1 \\ \mathbf{0} & \dots & \mathbf{0} & G_3 \\ \vdots & \vdots & \vdots & \vdots \\ \mathbf{0} & \dots & \mathbf{0} & G_{N_t-3} \\ \mathbf{0} & \dots & \mathbf{0} & G_{N_t-1} \end{bmatrix}.$$

Since $Q^{\mathsf{T}} J_n^{-1} Q = \Sigma_n$, we have

$$\begin{bmatrix} Q^{\mathsf{T}} & 0 \\ 0 & Q^{\mathsf{T}} \end{bmatrix} G_n \begin{bmatrix} Q & 0 \\ 0 & Q \end{bmatrix} = \begin{bmatrix} Q^{\mathsf{T}} & 0 \\ 0 & Q^{\mathsf{T}} \end{bmatrix} \begin{bmatrix} J_{n-1}^{-1} & -J_n^{-1} \\ J_n^{-1} & -J_{n+1}^{-1} \end{bmatrix} \begin{bmatrix} Q & 0 \\ 0 & Q \end{bmatrix} = \begin{bmatrix} \Sigma_{n-1} & -\Sigma_n \\ \Sigma_n & -\Sigma_{n+1} \end{bmatrix},$$

and hence

$$\begin{bmatrix} Q^{\mathsf{T}} & 0 \\ 0 & Q^{\mathsf{T}} \end{bmatrix} G_n^{\mathsf{T}} G_n \begin{bmatrix} Q & 0 \\ 0 & Q \end{bmatrix} = \begin{bmatrix} \Sigma_{n-1}^2 + \Sigma_n^2 & -\Sigma_n(\Sigma_{n-1} + \Sigma_{n+1}) \\ -\Sigma_n(\Sigma_{n-1} + \Sigma_{n+1}) & \Sigma_n^2 + \Sigma_{n+1}^2 \end{bmatrix}.$$

The matrix form of (2.10) gives $\Sigma_{n+1} = 2\Lambda^{-1}\Sigma_n - \Sigma_{n-1}$ and hence $(\Sigma_{n-1} + \Sigma_{n+1}) = 2\Lambda^{-1}\Sigma_n$. It follows from $\Lambda = \operatorname{diag}(\lambda_1, \dots, \lambda_{N_x})$ with $\lambda_j^{-1} \in (0,1)$ that $\|\Lambda^{-1}\|_\infty \le 1$. This further implies that (with $\theta_j \in (0, \frac{\pi}{2})$)

$$\begin{aligned}
\|\mathcal{K}^{-1}\mathcal{R}\|_2^2 &= \lambda_{\max}\left((\mathcal{K}^{-1}\mathcal{R})^{\mathsf{T}}\mathcal{K}^{-1}\mathcal{R}\right) = \lambda_{\max}\left(\sum_{n=1,3,\dots}^{N_t-1} G_n^{\mathsf{T}} G_n\right) \\
&= \lambda_{\max}\left(\sum_{n=1,3,\dots}^{N_t-1} \begin{bmatrix} \Sigma_{n-1}^2 + \Sigma_n^2 & -2\Lambda^{-1}\Sigma_n^2 \\ -2\Lambda^{-1}\Sigma_n^2 & \Sigma_n^2 + \Sigma_{n+1}^2 \end{bmatrix}\right) \\
&\le \left\| \begin{bmatrix} \sum_{n=0}^{N_t-1} \Sigma_n^2 & -2\Lambda^{-1}(\sum_{n=1,3,\dots}^{N_t-1} \Sigma_n^2) \\ -2\Lambda^{-1}(\sum_{n=1,3,\dots}^{N_t-1} \Sigma_n^2) & \sum_{n=1}^{N_t} \Sigma_n^2 \end{bmatrix} \right\|_\infty \\
&\le (1 + 2\|\Lambda^{-1}\|_\infty) \left\| \sum_{n=0}^{N_t} \Sigma_n^2 \right\|_\infty \\
&\le 3 \max_{1 \le j \le N_x} \left(\sum_{n=0}^{N_t} \frac{\sin^2((n+1)\theta_j)}{\sin^2 \theta_j} \right) \\
&\le 3 \max_{\theta \in [0, \pi/2]} \underbrace{\left(\sum_{n=1}^{N_t+1} \frac{\sin^2(n\theta)}{\sin^2 \theta} \right)}_{=:\phi_{N_t+1}(\theta)},
\end{aligned}$$

where we defined the following function:

$$\phi_{N_t}(\theta) := \sum_{n=1}^{N_t} \frac{\sin^2(n\theta)}{\sin^2 \theta}. \tag{3.6}$$

In Appendix A, we proved that, for a fixed $N_t \geq 1$, the function $\phi_{N_t}(\theta)$ attains its maximum at $\theta = 0$ with

$$\max_{\theta\in[0,\pi/2]} \phi_{N_t}(\theta) = \phi_{N_t}(0) := \lim_{\theta\to 0^+} \phi_{N_t}(\theta) = \frac{1}{6}N_t(N_t+1)(2N_t+1), \tag{3.7}$$

which leads to (recall that $\tau = T/N_t$)

$$\begin{aligned}\|\mathcal{K}^{-1}\mathcal{R}\|_2 \leq \sqrt{3\max_{\theta\in[0,\pi/2]} \phi_{N_t+1}(\theta)} = \sqrt{\frac{3}{6}(N_t+1)(N_t+2)(2N_t+3)} \leq (N_t+2)^{3/2} \\ = (T/\tau+2)^{3/2}. \end{aligned} \tag{3.8}$$

In view of (3.5), we arrive at the following linear convergence rate of the preconditioned GMRES.

THEOREM 3.1. *For any given positive constant $\delta \in (0, 1/2)$, if we choose a small $\alpha = \delta(T/\tau+2)^{-3/2}$ such that $\alpha\|\mathcal{K}^{-1}\mathcal{R}\|_2 \leq \delta < 1/2$, then the left-preconditioned GMRES for solving $\mathcal{K}y_h = b_h$ with the preconditioner $\mathcal{P}_\alpha$ achieves the following mesh-independent linear convergence rate:*

$$\begin{aligned}\frac{\|r^k\|_2}{\|r^0\|_2} &\leq \left(1-\left(1-2\alpha(T/\tau+2)^{3/2}\right)^2\right)^{k/2} \\ &\leq \left(1-(1-2\delta)^2\right)^{k/2} = \left(2\sqrt{\delta(1-\delta)}\right)^k, \end{aligned} \tag{3.9}$$

where r^k denotes the residual vector at the kth iteration of the preconditioned GMRES.

This theorem indicates that the number of the preconditioned GMRES iterations will be mesh-independent if α is sufficiently small. It also shows that a smaller α leads to a faster convergence rate, as indeed observed in numerical experiments. Among our tested examples, we observe that a fixed $\alpha = 0.1$ or 0.01 already provides very fast mesh-independent convergence rate. However, the theoretical condition $\alpha = O(\tau^{3/2})$ is rather restrictive, because it implies that α goes to 0 rapidly as we refine the time step-size τ. In practical computation, the parameter α cannot be arbitrarily small, because a too small α results in unacceptable roundoff errors in step-(a) and step-(c) of the diagonalization procedure (2.2), which may seriously pollute the obtained numerical solution; see [22, 65] for more details about the roundoff error arising in the diagonalization technique. In essence, such a restrictive condition originates from the significant (though low-rank) difference between the matrices $\mathcal{P}_\alpha$ and $\mathcal{K}$ when the time step-size τ becomes small. To see this, we recall $\mathcal{P}_\alpha = \mathcal{K}+\alpha\mathcal{R}$, and the difference norm $\|\mathcal{P}_\alpha - \mathcal{K}\|_2 = \|\alpha\mathcal{R}\|_2$ can be bounded from below [26] by

$$\|\mathcal{P}_\alpha - \mathcal{K}\|_2 = \alpha\|\mathcal{R}\|_2 = \frac{\alpha}{\tau^2}\left\|\begin{bmatrix} L & -2I_x \\ 0 & L\end{bmatrix}\right\|_2 \geq \frac{\alpha}{\tau^2}\| - 2I_x\|_2 = \frac{2\alpha}{\tau^2}. \tag{3.10}$$

Hence, it seems unavoidable to pick a small α such that $\|\mathcal{P}_\alpha^{-1}\mathcal{R}\|_2$ is uniformly bounded by a small constant. During the revision of this paper, we noticed a related recent work [37], where the authors derived a comparable convergence rate under weaker condition $\alpha = O(\tau^{1/2})$ through directly estimating $\|\mathcal{P}_\alpha^{-1}\mathcal{K}\|_2$ based on a first-order backward Euler scheme in time for the parabolic equations. Our derived stronger condition $\alpha = O(\tau^{3/2})$ is also reasonable considering the wave equation has a second-order derivative in time.

3.2. Two convergent stationary iterative methods. To provide another simple perspective on explaining the observed fast convergence rate of preconditioned GMRES, we construct a simple stationary iterative method and then utilize its asymptotic convergence rate as an indicator for estimating the convergence rate of any minimum residual-based Krylov subspace methods [19, Theorem 2]. Since GMRES generates the optimal polynomial in $\mathcal{P}_\alpha^{-1}\mathcal{K}$ (in terms of minimizing residual norms) whereas the corresponding stationary iteration produces a particular polynomial $(\mathcal{I}-\mathcal{P}_\alpha^{-1}\mathcal{K})^k$, then GMRES must converge in a number of iterations that is less than or equal to the iteration number required for convergence of the stationary iteration. More specifically, we suggest using the matrix splitting $\mathcal{K}=\mathcal{P}_\alpha-(\mathcal{P}_\alpha-\mathcal{K})$ to get

$$\mathcal{P}_\alpha y_h=(\mathcal{P}_\alpha-\mathcal{K})y_h+b_h=\mathcal{P}_\alpha y_h+(b_h-\mathcal{K}y_h),$$

which leads to the following simple stationary iterative method:

$$y_h^{k+1}=(\mathcal{I}-\mathcal{P}_\alpha^{-1}\mathcal{K})y_h^k+\mathcal{P}_\alpha^{-1}b_h=y_h^k+\mathcal{P}_\alpha^{-1}(b_h-\mathcal{K}y_h^k),\quad k=0,1,2,\dots \tag{3.11}$$

with an initial guess y_h^0. The asymptotic convergence rate of (3.11) is determined by the spectral radius of its iteration matrix $\mathcal{A}:=(\mathcal{I}-\mathcal{P}_\alpha^{-1}\mathcal{K})$. From Theorem 2.1, we have

$$\rho(\mathcal{A})=\max_{\lambda\in\sigma(\mathcal{P}_\alpha^{-1}\mathcal{K})}|1-\lambda|\le\frac{\alpha}{1-\alpha}<1,$$

whenever $\alpha\in(0,1/2)$. This also shows that a smaller α gives a smaller spectral radius, indicating faster asymptotic convergence rate, though not necessarily faster iteration-by-iteration convergence in computation.

Let $e^k=y_h^k-y_h$ denote the error vector, and let $r^k=\mathcal{P}_\alpha^{-1}(b_h-\mathcal{K}y_h^k)$ be the preconditioned residual vector at the kth iteration. Then we have $e^k=\mathcal{A}e^{k-1}=\mathcal{A}^k e^0$ (or, equivalently, $r^k=\mathcal{A}r^{k-1}=\mathcal{A}^k r^0$). Since $\rho(\mathcal{A})\le\frac{\alpha}{1-\alpha}<1$ implies $\lim_{k\to\infty}\mathcal{A}^k=0$, we get

$$\lim_{k\to\infty}\|e^k\|=\lim_{k\to\infty}\|\mathcal{A}^k e^0\|=0,\quad \lim_{k\to\infty}\|r^k\|=\lim_{k\to\infty}\|\mathcal{A}^k r^0\|=0.$$

In particular, the asymptotic convergence rate of the residual vector sequence $\{r^k\}_{k=0}^\infty$ is given by $\rho(\mathcal{A})\le\frac{\alpha}{1-\alpha}<1$ when $\alpha\in(0,1/2)$. Roughly speaking, for a given accuracy tolerance $\texttt{tol}$, it takes about

$$q(\texttt{tol};\alpha):=\left\lceil\frac{\ln(\texttt{tol})}{\ln\alpha-\ln(1-\alpha)}\right\rceil \tag{3.12}$$

iterations to attain the stopping condition $\|r^k\|/\|r^0\|\le\texttt{tol}$. Here $\lceil s\rceil$ denotes the smallest integer greater than or equal to s. This estimated iteration number $q(\texttt{tol};\alpha)$ may be too optimistic in practice, since it does not take into account the nonnormality of the iteration matrix $\mathcal{A}$, which can lead to significant error growth before eventually contracting at the asymptotic convergence rate determined by its spectral radius.

Finally, we remark that the restriction $\alpha<\frac{1}{2}$ can be relaxed via a *damped* version of the above stationary iterative method:

$$y_h^{k+1}=(\mathcal{I}-\beta\mathcal{P}_\alpha^{-1}\mathcal{K})y_h^k+\beta\mathcal{P}_\alpha^{-1}b_h=y_h^k+\beta\mathcal{P}_\alpha^{-1}(b_h-\mathcal{K}y_h^k) \tag{3.13}$$

with $\beta=1-\alpha$. For the iteration matrix $\mathcal{B}:=\mathcal{I}-\beta\mathcal{P}_\alpha^{-1}\mathcal{K}$, from Theorem 2.1 we have

$$\begin{aligned}\rho(\mathcal{B})=\max_{1\le j\le N_x}\left\{|\beta-1|,\left|\tfrac{\beta}{1+\alpha\nu_j^{\pm}}-1\right|\right\}&=\max_{1\le j\le N_x}\left\{\alpha,\tfrac{|1+\nu_j^{\pm}|}{|\alpha^{-1}+\nu_j^{\pm}|}\right\}\\&\le\max\left\{\alpha,\tfrac{2\alpha}{1+\alpha}\right\}=\tfrac{2\alpha}{1+\alpha}<1,\end{aligned}$$

where $\nu_j^{\pm} = -e^{\pm \mathrm{i} N_t \theta_j}$, $\alpha < \frac{2\alpha}{1+\alpha}$ since $\alpha^2 < \alpha$, and the inequalities are derived as

$$\frac{|1+\nu_j^{\pm}|}{|\alpha^{-1}+\nu_j^{\pm}|} \le \max_{\theta\in[0,2\pi]} \frac{|1+\cos\theta+i\sin\theta|}{|\alpha^{-1}+\cos\theta+i\sin\theta|} = \frac{2}{\alpha^{-1}+1} = \frac{2\alpha}{1+\alpha} < \frac{1+\alpha}{1+\alpha} = 1.$$

To summarize, we obtained the following convergence results of two stationary iterative methods.

THEOREM 3.2. *For the all-at-once system $\mathcal{K} y_h = b_h$, we have two stationary iterative methods:*

(1) *If $\alpha \in (0, 1/2)$, the fixed-point iteration*

$$y_h^{k+1} = \mathcal{A} y_h^k + \mathcal{P}_\alpha^{-1} b_h = y_h^k + \mathcal{P}_\alpha^{-1}(b_h - \mathcal{K} y_h^k) \tag{3.14}$$

is convergent and the iteration matrix $\mathcal{A} := \mathcal{I} - \mathcal{P}_\alpha^{-1}\mathcal{K}$ satisfies $\rho(\mathcal{A}) \le \frac{\alpha}{1-\alpha} < 1$.

(2) *If $\alpha \in (0, 1)$, the damped fixed-point iteration (with $\beta = 1-\alpha$)*

$$y_h^{k+1} = \mathcal{B} y_h^k + \beta \mathcal{P}_\alpha^{-1} b_h = y_h^k + \beta \mathcal{P}_\alpha^{-1}(b_h - \mathcal{K} y_h^k) \tag{3.15}$$

is convergent and the iteration matrix $\mathcal{B} := (\mathcal{I} - \beta \mathcal{P}_\alpha^{-1}\mathcal{K})$ satisfies $\rho(\mathcal{B}) \le \frac{2\alpha}{1+\alpha} < 1$.

This theorem indicates that a smaller α gives smaller $\rho(\mathcal{A})$ and $\rho(\mathcal{B})$ and hence a lesser number of iterations are needed to reach a given convergence tolerance. However, due to the nonnormality of the iteration matrices $\mathcal{A}$ and $\mathcal{B}$, the spectral radius conditions $\rho(\mathcal{A}) < 1$ and $\rho(\mathcal{B}) < 1$ cannot exclude the possibility of significant initial growth in $\|e^k\|$ before the asymptotic convergent rate eventually kicks in. This implies that we may observe a nonmonotone decay of the residual norms at the first few iterations. Figure 3.1 illustrates the convergence behavior of $\|\mathcal{A}^k\|_2$ (with $\alpha = 0.49$) and $\|\mathcal{B}^k\|_2$ (with $\alpha = 0.99$), where $\|\mathcal{B}^k\|_2$ indeed shows moderate initial growth while $\|\mathcal{A}^k\|_2$ seems to only show monotone convergence. Fortunately, for our proposed preconditioner $\mathcal{P}_\alpha$ with a relatively small α, it is unlikely to observe such an initial growth. In particular, for all the numerical experiments performed in section 5 we do not observe such phenomenon. For any $\delta \in (0, 1/2)$, under the same condition $\alpha\|\mathcal{K}^{-1}\mathcal{R}\|_2 \le \delta < 1/2$, such monotone convergence with respect to the $\|\cdot\|_2$ norm can be obviously inferred from the above derived bound (3.3), that is,

$$\|\mathcal{A}\|_2 = \|\mathcal{I} - \mathcal{P}_\alpha^{-1}\mathcal{K}\|_2 \le \frac{\alpha\|\mathcal{K}^{-1}\mathcal{R}\|_2}{1-\alpha\|\mathcal{K}^{-1}\mathcal{R}\|_2} \le \frac{\delta}{1-\delta} < 1.$$

In practice, we would anticipate the preconditioned GMRES to converge in approximately $q(\texttt{tol}; \alpha)$ iterations. For example, with $\alpha = 0.1$ and $\texttt{tol} = 10^{-6}$ we have $q(\texttt{tol}; \alpha) = 7$, and thus the preconditioned GMRES will likely converge in about seven iterations, as we indeed observed numerically. We remark that it is, in general, too pessimistic or impractical to estimate the convergence rate of preconditioned GMRES through the corresponding stationary iterative method. This approach, however, is particularly useful in our considered problem, since the spectrum of the preconditioned matrix is confined within a uniformly bounded annulus (whose radii and bandwidth depend on α only), and the nonnormality of the preconditioned matrix also does not seem to significantly deteriorate the practical step-by-step convergence rates in the initial stage.

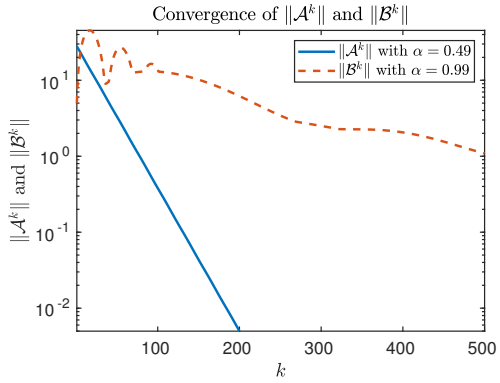

FIG. 3.1. *The convergence behavior of $\|\mathcal{A}^k\|_2$ (with $\alpha = 0.49$) and $\|\mathcal{B}^k\|_2$ (with $\alpha = 0.99$). Here we used a one-dimensional (1D) domain $\Omega = (0,1)$ with $T = 2, N_x = 16, N_t = 16$. Notice that the initial values $\|\mathcal{A}\|_2$ and $\|\mathcal{B}\|_2$ and the convergence rates depend on α.*

4. Application to nonlinear wave equations. We now extend the previous diagonalization-based iterative method to nonlinear cases. To this end, we consider a semilinear wave equation:

$$\begin{cases} y_{tt} - \Delta y + \psi(y) = f & \text{in } \Omega \times (0,T), \\ y = 0 & \text{on } \partial\Omega \times (0,T), \\ y(\cdot,0) = \psi_0,\ y_t(\cdot,0) = \psi_1 & \text{in } \Omega. \end{cases} \tag{4.1}$$

Similar to (1.4a), the corresponding all-at-once finite difference discretization scheme for (4.1) is

$$\mathcal{K} y_h + \mathcal{F}(y_h) = b_h, \tag{4.2}$$

where $\mathcal{K}$ and y_h are the same as in (1.4b). The nonlinear function $\mathcal{F}$ and the right-hand-side vector b_h are

$$\mathcal{F}(y_h) = \begin{bmatrix} \psi(Y_1)/2 \\ \psi(Y_1) \\ \vdots \\ \psi(Y_{N_t-1}) \end{bmatrix}, \quad b_h = \begin{bmatrix} (F_0 - \psi(Y_0))/2 + \Psi_1/\tau + \Psi_0/\tau^2 \\ F_1 - L\Psi_0/\tau^2 \\ F_2 \\ \vdots \\ F_{N_t-1} \end{bmatrix}.$$

Applying Newton's iteration to (4.2) gives

$$y_h^{k+1} = y_h^k + (\mathcal{K} + \nabla\mathcal{F}^k)^{-1}(b_h - \mathcal{K} y_h^k - \mathcal{F}(y_h^k)), \tag{4.3}$$

where the Jacobian matrix is

$$\nabla\mathcal{F}^k := \nabla\mathcal{F}(y_h^k) = \begin{bmatrix} \nabla\psi(Y_1^k)/2 & & & \\ & \nabla\psi(Y_1^k) & & \\ & & \ddots & \\ & & & \nabla\psi(Y_{N_t-1}^k) \end{bmatrix}.$$

To implement each iteration of (4.3) by the diagonalization technique, we first replace each diagonal block $\nabla\psi(Y_j^k)$ in $\nabla\mathcal{F}^k$ by its average:

$$\nabla\psi(Y_j^k) \approx \overline{\nabla F}^k = \frac{1}{N_t}\left(\sum_{n=1}^{N_t} \nabla\psi(Y_n^k)\right), \quad j = 1, 2, \ldots, N_t - 1.$$

Hence, we can approximate the Kronecker product approximation as $\nabla\mathcal{F}^k \approx I_t \otimes \overline{\nabla F}^k$.

The construction of the average Jacobian matrix $\overline{\nabla F}^k$ is motivated by the work in [21], where a nonlinear parabolic PDE discretized by the backward Euler method was studied. In this way, the whole Jacobian matrix $(\mathcal{K} + \nabla\mathcal{F}^k)$ in (4.3) can be preconditioned by the block α-circulant type matrix

$$\mathcal{P}_\alpha^k := \mathcal{P}_\alpha + I_t \otimes \overline{\nabla F}^k = \frac{1}{\tau^2}\left(C_1^{(\alpha)} \otimes L - C_2^{(\alpha)} \otimes 2I_x\right) + I_t \otimes \overline{\nabla F}^k$$

with $\mathcal{P}_\alpha$ given by (1.7). Therefore, similar to the iteration (3.11), we obtained a simplified Newton iteration

$$y_h^{k+1} = y_h^k + (\mathcal{P}_\alpha^k)^{-1}(b_h - \mathcal{K}y_h^k - \mathcal{F}(y_h^k)). \tag{4.4}$$

It is also possible to use $\mathcal{P}_\alpha^k$ as preconditioner of GMRES for solving the standard Newton iteration (4.3), but the above simplified Newton iteration often shows better computational efficiency, since it avoids inner iterations within a GMRES solver. Similar to (2.2), by letting $r^k := b_h - \mathcal{K}y_h^k - \mathcal{F}(y_h^k)$ we can compute $z^k := (\mathcal{P}_\alpha^k)^{-1}r^k$ via the similar 3-step diagonalization procedure:

$$\begin{aligned}
&\text{Step-(a)} \quad S_1 = (V^{-1} \otimes I_x)r^k,\\
&\text{Step-(b)} \quad S_{2,n} = \tau^2\left(\lambda_{1,n}L - 2\lambda_{2,n}I_x + \tau^2\overline{\nabla F}^k\right)^{-1} S_{1,n}, \quad n = 1, 2, \ldots, N_t,\\
&\text{Step-(c)} \quad z^k = (V \otimes I_x)S_2.
\end{aligned} \tag{4.5}$$

We notice that only the diagonal average matrix $\overline{\nabla F}^k$ changes over the iterations. At this moment, the convergence analysis of the above simplified Newton iteration is widely open and left as our future work.

5. Numerical examples. In this section, we provide several numerical examples to demonstrate the convergence rates and computational efficiency of our proposed block α-circulant preconditioner. All simulations are implemented using MATLAB on a Dell Precision 7520 Laptop with Intel Core i7-7700HQ CPU @ 2.80GHz and 32GB RAM. The CPU time (in seconds) is estimated by using the timing functions `tic`/`toc`, based on the sequential-in-time implementation of the preconditioned iterative algorithms. Here we mainly focus on investigating the convergence properties of the algorithms and the numerical results based on PinT computation in a real parallel circumstance will be reported in our forthcoming work. Our analysis is based on the left-preconditioned GMRES, but the right-preconditioned GMRES is preferred in practice since its termination criterion is based on the reliable true residuals, rather than the potentially significantly larger and smaller preconditioned residuals [57]. We employ the right-preconditioned GMRES solver (without restarts) in the IFISS package [13, 14, 59], and choose a zero initial guess and a stopping tolerance `tol` based on the reduction in relative residual norms.

We mention that when memory is limited the restarted GMRES or other memory-efficient Krylov subspace methods (e.g., Bi-CGSTAB) can also be used, but their convergence behaviors may be different, which is not further pursued in this paper. In rectangular domains with regular grids, the complex-shifted systems in (2.2) are solved by MATLAB's sparse direct solver (Thomas algorithm) and fast direct solver [4, 31, 57] (based on discrete sine transform) for 1D and 2D cases, respectively. For the more general domains with irregular grids, fast iterative solvers (e.g., the algebraic multigrid method) could be used for approximately solving the complex-shifted systems in (2.2). The prototype MATLAB codes for implementing

our algorithms are included in the freely available ParaDiag package at the website https://github.com/wushulin/ParaDIAG.

According to the error estimate in [35], we will measure the discrete $L^\infty((0,T);L^2(\Omega))$ norm errors of the numerical approximation as e_y, and then estimate the experimental order of accuracy by calculating the logarithmic ratio of the approximation errors between two successive refined meshes, i.e.,

$$\text{Order} = \log_2\left(\frac{e_y(h,\tau)}{e_y(2h,2\tau)}\right),$$

which should be close to two for a second-order accuracy if solution is sufficiently smooth.

5.1. Example 1 [24]. In our first example, we choose $\Omega = (0,1)$ and

$$\psi_0(x) = \chi_{[3/8,5/8]}(x)\cos^2(4\pi(x-1/2)), \quad \psi_1(x) = 0, \ f = 0,$$

such that the exact solution is

$$y(x,t) = \sum_{n=1}^{\infty} \frac{64(\cos(5n\pi/8) - \cos(3n\pi/8))}{\pi(n^3 - 64n)} \sin(n\pi x)\cos(n\pi t).$$

Here χ_E is an indicator function on the set E. In this example, due to the nonsmoothness of the initial condition, the solution is not smooth across some characteristic lines. The second-order approximation accuracy by the finite difference scheme is clearly deteriorated, which is reasonable since its discretization errors rely on the suitable regularity of solution via Taylor series expansions. From Table 5.1, we see that the preconditioner with $\alpha = 1$ performs poorly, while the preconditioner with $\alpha = 0.1$ delivers excellent mesh-independent convergence rate. The observed slower convergence rate of preconditioned GMRES with $\alpha = 1$ is partially due to the influence of nonsmooth right-hand side, as also observed in [24] and thoroughly investigated in [63]. In particular, with $\alpha = 0.1$ the required iteration numbers are indeed slightly less than the estimated upper bound $q(\texttt{tol} = 10^{-6}; \alpha = 0.1) = 7$. The CPU times show an expected quasi-linear time complexity of the proposed algorithm.

TABLE 5.1
Error and convergence results for Example 1 with our preconditioner ($T = 1$, $\texttt{tol} = 10^{-6}$).

	$\alpha = 1$				$\alpha = 0.1$			
(N_x, N_t)	Error	Order	Iter	CPU	Error	Order	Iter	CPU
(256,256)	1.11E-02	1.87	89	1.78	1.11E-02	1.87	5	0.15
(512,512)	3.03E-03	1.87	116	11.23	3.04E-03	1.87	4	0.57
(1024,1024)	8.32E-04	1.87	155	74.47	8.34E-04	1.86	4	2.16
(2048,2048)	4.03E-04	1.05	203	626.32	4.03E-04	1.05	3	8.09

5.2. Example 2. In our second example, we choose $\Omega = (0,1)^2$ and

$$\psi_0(x_1,x_2) = 0, \quad \psi_1(x_1,x_2) = x_1(x_1-1)x_2(x_2-1),$$
$$f = -\frac{1}{(1+t)^2}x_1(x_1-1)x_2(x_2-1) - 2\ln(t+1)(x_1(x_1-1) + x_2(x_2-1)),$$

such that the exact solution is $y(x_1,x_2,t) = x_1(x_1-1)x_2(x_2-1)\ln(t+1)$. As shown in Table 5.2, the second-order approximation accuracy was clearly observed. The preconditioner with $\alpha = 0.1$ shows a very robust mesh-independent convergence rate, which confirms our eigenvalue and convergence analysis very well. In particular, it

costs six iterations to arrive at the prescribed tolerance, which matches the predicted $q(\texttt{tol} = 10^{-6}; \alpha = 0.1) = 7$ iterations for the stationary iterative method. However, the iteration numbers for the preconditioner with $\alpha = 1$ grows dramatically as we refine the mesh. This is expected since the linear systems from 2D problem become very ill-conditioned quickly as the mesh is refined. The observed six iterations with $\alpha = 0.1$ is very satisfactory for solving such a large-scale nonsymmetric sparse linear system with about 16.8 million unknowns.

TABLE 5.2
Error and convergence results for Example 2 *with our preconditioner* ($T = 2$, $\texttt{tol} = 10^{-6}$).

	$\alpha = 1$				$\alpha = 0.1$			
(N_x, N_x, N_t)	Error	Order	Iter	CPU	Error	Order	Iter	CPU
(32,32,32)	2.92E-04	1.9	74	2.78	2.92E-04	1.9	6	0.06
(64,64,64)			>300		7.42E-05	2.0	6	0.29
(128,128,128)					1.86E-05	2.0	6	1.80
(256,256,256)					4.66E-06	2.0	6	18.31

The convergence results of the preconditioned GMRES and the stationary iterative method (3.14) with different α are compared in Tables 5.3 and 5.4, respectively, where the required iteration numbers match well with the estimated bound $q(\texttt{tol}; \alpha)$ provided in the last row of Table 5.4. As shown in Table 5.3, the required GMRES iteration number decreases if a smaller α is used. But a very small α should be avoided, since in this case the roundoff error may eventually affect the accuracy of the obtained numerical solution. Interestingly, for this example even with $\alpha = 10^{-8}$ the preconditioned GMRES still works very well. Notice that the approximation errors are the same for different α since they solve the same system. The observed fast convergence rate for a very small α is not surprising, since $\mathcal{P}_\alpha$ indeed converges to $\mathcal{K}$ as α goes to 0.

TABLE 5.3
Convergence results for Example 2 *with our preconditioner and different* α ($T = 2$, $\texttt{tol} = 10^{-6}$).

	$\alpha = 0.1$		$\alpha = 0.01$		$\alpha = 10^{-4}$		$\alpha = 10^{-6}$		$\alpha = 10^{-8}$	
(N_x, N_x, N_t)	Error	Iter	Error	Iter	Error	Iter	Error	Iter	Error	Iter
(32,32,32)	2.9E-04	6	2.9E-04	3	2.9E-04	2	2.9E-04	2	2.9E-04	1
(64,64,64)	7.4E-05	6	7.4E-05	3	7.4E-05	2	7.4E-05	2	7.4E-05	1
(128,128,128)	1.9E-05	6	1.9E-05	3	1.9E-05	2	1.9E-05	2	1.9E-05	1
(256,256,256)	4.7E-06	6	4.7E-06	4	4.7E-06	2	4.7E-06	2	4.7E-06	2

TABLE 5.4
Convergence results for Example 2 *with the stationary iterative method* (3.14) *and different* α ($T = 2$, $\texttt{tol} = 10^{-6}$).

	$\alpha = 0.1$		$\alpha = 0.01$		$\alpha = 10^{-4}$		$\alpha = 10^{-6}$		$\alpha = 10^{-8}$	
(N_x, N_x, N_t)	Error	Iter	Error	Iter	Error	Iter	Error	Iter	Error	Iter
(32,32,32)	2.9E-04	7	2.9E-04	4	2.9E-04	2	2.9E-04	2	2.9E-04	1
(64,64,64)	7.4E-05	7	7.4E-05	4	7.4E-05	2	7.4E-05	2	7.4E-05	1
(128,128,128)	1.9E-05	8	1.9E-05	4	1.9E-05	2	1.9E-05	2	1.9E-05	1
(256,256,256)	4.7E-06	8	4.7E-06	4	4.7E-06	2	4.7E-06	2	4.7E-06	2
$q(\texttt{tol}; \alpha)$	7		4		2		2		1	

We now demonstrate the speedup potential of our PinT preconditioner over the serial time-stepping method for Example 2. We choose $T = 4$, $(N_x, N_x) = (256, 256)$,

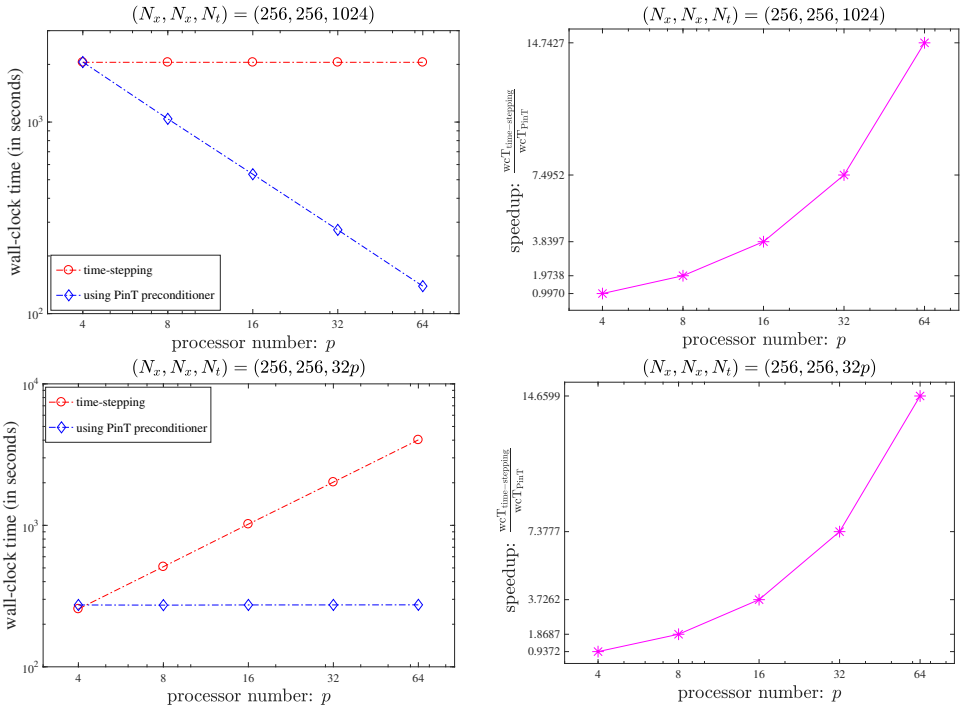

FIG. 5.1. *Comparison of the measured wall-clock-time for the serial time-stepping method using a single processor and the PinT stationary iterative method* (3.14) *using p processors. Top row: fixed* $N_t = 1024$. *Bottom row: increasing* N_t *as* $N_t = 32p$. *Left column: the measured wall-clock-time (wcT) in seconds. Right column: the speedup measured as* $\frac{\text{wcT}_{\text{time-stepping}}}{\text{wcT}_{\text{PinT}}}$.

and a parameter $\alpha = 10^{-2}$ (with this parameter the proposed iterative algorithm using the PinT preconditioner needs five iterations to arrive at the tolerance $\texttt{tol} = 10^{-7}$). We implemented our PinT algorithm with Fortran and tested its parallel efficiency using up to 64 processors on the Tianhe-1 supercomputer [67]. The strong and weak scaling timing results are shown in Figure 5.1, where the serial time-stepping method is also compared. In particular, in Figure 5.1 on the top-left panel, we show the wall-clock time for the time-stepping method on a single processor and the strong scaling results (with fixed $N_t = 1024$) of the PinT stationary iterative method (3.14) with respect to different p processors. As p increases, it is clear that the wall-clock time of the PinT stationary iterative method reduces accordingly and it is obviously smaller than that of the serial time-stepping method when p is large. On the top-right panel, we show the speedup measured as $\texttt{speedup} = \frac{\text{wcT}_{\text{time-stepping}}}{\text{wcT}_{\text{PinT}}}$. In Figure 5.1 on the bottom row, we show the weak scaling results when $\tau = \frac{1}{256}$ is fixed and $N_t = 32p$ grows proportionally with respect to p, the number of used processors. In this case, we notice from the bottom-left panel that the wall-clock time for the time-stepping method increases linearly as N_t grows, while the time for the iterative algorithm using the PinT preconditioner (almost) keeps constant. Both the strong and weak scaling results indicate that the proposed stationary iterative algorithm has a satisfactory parallel efficiency and it indeed has a promising speedup potential over the serial time-stepping method.

5.3. Example 3. In the third example, we choose $\Omega = \{(x_1, x_2)|x_1^2 + x_2^2 < 1\}$ to be the unit disk with

$$\psi_0(x_1, x_2) = 0, \quad \psi_1(x_1, x_2) = (1 - (x_1^2 + x_2^2)^2),$$
$$f = -\frac{2t}{(1+t^2)^2}(1 - (x_1^2 + x_2^2)^2) + 16(x_1^2 + x_2^2)\arctan(t),$$

such that the exact solution is $y(x_1, x_2, t) = (1 - (x_1^2 + x_2^2)^2)\arctan(t)$. For space discretization, we use P_1 finite element implemented in T-IFISS [2] based on DistMesh [51]. Figure 5.2 illustrates the generated uniform triangular meshes of the unit disk with 228 nodes and the comparison of typical exact and computed solutions at the final time $T = 2$.

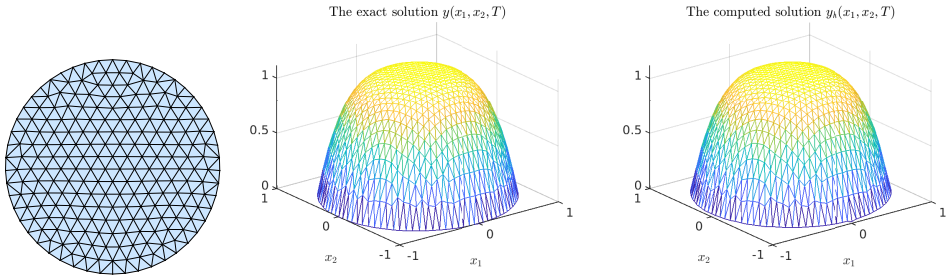

FIG. 5.2. *A uniform triangular mesh (left) of the unit disk domain with* 228 *nodes and the comparison of the exact solution (middle) and computed solution (right) with* $N_t = 32$ *at the final time* $T = 2$ *in Example* 3.

As reported in Table 5.5, the second-order approximation accuracy was clearly observed, where DoF_x, the total number of nodes (i.e., the degree of freedom in space) in the triangular mesh, roughly increases by four times as the mesh is refined. The preconditioner with $\alpha = 0.1$ shows very robust mesh-independent convergence rate, while the iteration numbers for the preconditioner with $\alpha = 1$ grows dramatically as the mesh is refined. In this case, the complex-shifted systems in (2.2) are solved by MATLAB's backslash sparse direct solver (with much higher CPU times), since the fast direct solvers used for the finite difference discretization are not directly applicable. Alternatively, one may use the algebraic multigrid solver AGMG [45, 48, 49, 50], which gives a similar numbers of outer GMRES iterations. The development of more efficient structure-exploiting solvers for the complex-shifted systems in (2.2) will be pursued in our future work.

TABLE 5.5
Error and convergence results for Example 3 *with our preconditioner* ($T = 2$, $\texttt{tol} = 10^{-6}$).

	$\alpha = 1$				$\alpha = 0.1$			
(DoF_x, N_t)	Error	Order	Iter	CPU	Error	Order	Iter	CPU
(925,32)	9.52E-03	2.5	161	34.70	9.52E-03	2.5	6	1.37
(3715,64)			>300		1.95E-03	2.3	6	12.58
(14840,128)					4.39E-04	2.2	6	128.04
(59422,256)					1.02E-04	2.1	6	1631.02

5.4. Example 4. In our nonlinear example [62], we choose $\Omega = (0, 1)$, $\psi(y) = y^5$, and

$$\psi_0(x) = 3(e^x - 1)(e^x - e)\cos 1, \ \psi_1(x) = -3(e^x - 1)(e^x - e)\sin 1, \ f = y_{tt} - y_{xx} + \psi(y).$$

The exact solution is $y(x,t) = 3(e^x-1)(e^x-e)\cos(e^t)$. According to section 4, we use the simplified Newton iteration (4.4) to solve the all-at-once nonlinear system, and for each iteration we solve the approximate Jacobian system with the diagonalization technique; cf. section 4. In Table 5.6, we see that the convergence rate of the proposed algorithm is mesh-independent and the algorithm converges faster when a smaller α is used. Here a tighter tolerance $\texttt{tol} = 10^{-8}$ is needed to obtain a clear second-order accuracy. However, the simplified Newton iteration indeed diverges with $\alpha = 1$ and the presentation of such a divergence result is omitted. We mention that such a simplified Newton iteration is only locally convergent, which may become divergent when T gets larger or the nonlinearities become not suitable for using the averaged Kronecker product approximation along the diagonals. It, however, remains widely open to understand what conditions can assure the global and/or local convergence of such a simplified Newton iteration or the Newton method.

TABLE 5.6
Convergence results for Example 4 with the proposed simplified Newton iteration ($T = 2$, $\texttt{tol} = 10^{-8}$).

	$\alpha = 0.1$				$\alpha = 0.01$			
(N_x, N_t)	Error	Order	Iter	CPU	Error	Order	Iter	CPU
(128,128)	1.94E-03	1.98	41	0.39	1.94E-03	1.98	24	0.24
(256,256)	4.86E-04	1.99	43	1.25	4.86E-04	1.99	24	0.71
(512,512)	1.22E-04	2.00	51	5.36	1.22E-04	2.00	26	3.33
(1024,1024)	3.05E-05	2.00	48	20.08	3.03E-05	2.01	26	11.28
(2048,2048)	7.68E-06	1.99	48	72.23	7.39E-06	2.03	26	41.02

5.5. Example 5. In our last example, we consider the wave equation with time-varying coefficient:

$$\begin{cases} y_{tt} - c(t)\Delta y = f & \text{in } \Omega \times (0,T), \\ y = 0 & \text{on } \partial\Omega \times (0,T), \\ y(\cdot,0) = \psi_0,\ y_t(\cdot,0) = \psi_1 & \text{in } \Omega, \end{cases} \tag{5.1}$$

where $\Omega = (0,1)^2$ and $c(t) > 0$ is a given coefficient function. With the same notation for the implicit leapfrog scheme, we arrive at the following all-at-once system:

$$\widehat{\mathcal{K}} y_h := \frac{1}{\tau^2} \begin{bmatrix} L_0 & & & & & \\ -2I_x & L_1 & & & & \\ L_2 & -2I_x & L_2 & & & \\ & \ddots & \ddots & \ddots & & \\ & & L_{N_t-1} & -2I_x & L_{N_t-1} \end{bmatrix} y_h = \frac{1}{\tau^2}\left(\widehat{B}_1 \otimes I_x - \frac{\tau^2}{2}\widehat{B}_2 \otimes \Delta_h\right) y_h = \widehat{b}_h,$$

where $L_n := I_x - \frac{\tau^2}{2} c_n \Delta_h \in \mathbb{R}^{N_x \times N_x}$, $c_n := c(t_n)$, and

$$\widehat{B}_1 = \begin{bmatrix} 1 & & & & \\ -2 & 1 & & & \\ 1 & -2 & 1 & & \\ & \ddots & \ddots & \ddots & \\ & & 1 & -2 & 1 \end{bmatrix}, \quad \widehat{B}_2 = \begin{bmatrix} c_0 & & & & \\ 0 & c_1 & & & \\ c_2 & 0 & c_2 & & \\ & \ddots & \ddots & \ddots & \\ & & c_{N_t-1} & 0 & c_{N_t-1} \end{bmatrix},$$

$$\widehat{b}_h = \begin{bmatrix} F_0/2 + \Psi_1/\tau + \Psi_0/\tau^2 \\ F_1 - L_1\Psi_0/\tau^2 \\ F_2 \\ \vdots \\ F_{N_t-1} \end{bmatrix}.$$

Following the similar idea for treating the nonlinear case in section 4, we can construct the following block α-circulant preconditioner:

$$\widehat{\mathcal{P}}_\alpha = \frac{1}{\tau^2}\left(\widehat{C}_1^{(\alpha)} \otimes I_x - \frac{\tau^2}{2}\widehat{C}_2^{(\alpha)} \otimes \Delta_h\right), \tag{5.2}$$

with

$$\widehat{C}_1^{(\alpha)} = \begin{bmatrix} 1 & & & \alpha & -2\alpha \\ -2 & 1 & & & \alpha \\ 1 & -2 & 1 & & \\ & \ddots & \ddots & \ddots & \\ & & 1 & -2 & 1 \end{bmatrix}, \quad \widehat{C}_2^{(\alpha)} = \bar{c}\begin{bmatrix} 1 & & & \alpha & 0 \\ 0 & 1 & & & \alpha \\ 1 & 0 & 1 & & \\ & \ddots & \ddots & \ddots & \\ & & 1 & 0 & 1 \end{bmatrix}, \tag{5.3}$$

where c_n in $\widehat{B}_2$ is replaced by the average $\bar{c} = \frac{1}{N_t}\sum_{n=0}^{N_t-1} c_n$ to get the desired α-circulant structure. Clearly, the preconditioning step $z = \widehat{\mathcal{P}}_\alpha^{-1} r$ can be computed parallel in time based on the diagonalization technique.

TABLE 5.7
Error and convergence results for Example 5 with $c(t) = 1 + \cos(\pi t)/2$ ($T = 2$, $tol = 10^{-6}$).

$\alpha = 0.1$	Preconditioned GMRES				Stationary iterations (3.14)			
(N_x, N_x, N_t)	Error	Order	Iter	CPU	Error	Order	Iter	CPU
(32,32,32)	4.80E-04	1.9	17	0.26	4.80E-04	1.9	28	0.25
(64,64,64)	1.23E-04	2.0	21	2.41	1.23E-04	2.0	141	6.04
(128,128,128)	3.09E-05	2.0	39	34.98			>300	

TABLE 5.8
Error and convergence results for Example 5 with $c(t) = 11 - 10.9\chi_{\{1\}}(t)$ ($T = 2$, $tol = 10^{-6}$).

$\alpha = 0.1$	Preconditioned GMRES				Stationary iterations (3.14)			
(N_x, N_x, N_t)	Error	Order	Iter	CPU	Error	Order	Iter	CPU
(32,32,32)	9.95E-04	1.8	19	0.33	9.95E-04	1.8	210	1.95
(64,64,64)	2.53E-04	2.0	18	2.19	2.53E-04	2.0	184	9.07
(128,128,128)	6.14E-05	2.0	14	12.84	6.14E-05	2.0	157	49.80

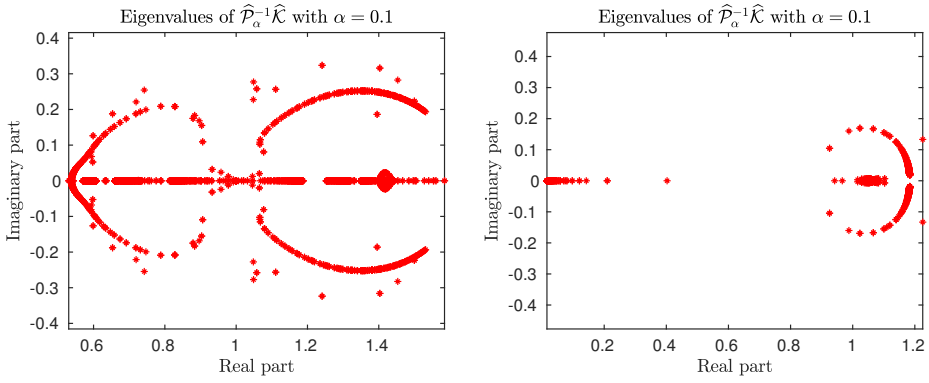

FIG. 5.3. *The spectrum distribution of $\widehat{\mathcal{P}}_\alpha^{-1}\widehat{\mathcal{K}}$ with $c(t) = 1 + \cos(\pi t)/2$ (left) and $c(t) = 11 - 10.9\chi_{\{1\}}(t)$ (right), respectively, in Example 5. Here, $T = 2, N_t = 16, \alpha = 0.1$.*

We will test the preconditioner $\widehat{\mathcal{P}}_\alpha$ and the corresponding stationary iterations (3.14) using the same solution in Example 2 (the source term f is suitably changed).

We consider two choices of $c(t)$: a continuous function $c(t) = 1 + \cos(\pi t)/2$ and a discontinuous function $c(t) = 11 - 10.9\chi_{\{1\}}(t)$. In Tables 5.7 and 5.8, we compare the convergence results of the preconditioned GMRES and the corresponding stationary iterations, where the former shows faster and more robust (though not always mesh-independent) convergence rate than the latter. This clear advantage as the mesh is refined is expected since the convergence rate of GMRES is not dominated by the spectral radius of the iteration matrix $\mathcal{A}$, which is greatly affected by the given coefficient function $c(t)$, as illustrated in Figure 5.3. The spectrum distribution of $\widehat{\mathcal{P}}_\alpha^{-1}\widehat{\mathcal{K}}$ matches the observed iteration numbers. This example reveals that the preconditioned GMRES can indeed significantly outperform the stationary iteration method in more general cases. In particular, the stationary iteration method (3.11) with $\alpha = 0.1$ become divergent if taking $c(t) = 1 + 0.99\cos(\pi t)$.

6. Conclusions. With the advent of massively parallel processors, parallelizable numerical algorithms for solving evolutionary PDEs are in great demand. Built upon an all-at-once (i.e., one-shot) implicit time-integrator for discretizing linear wave equations, a parallel-in-time block α-circulant preconditioner is proposed and the spectrum of the *diagonalizable* preconditioned system is precisely characterized. All the nonunit eigenvalues of the preconditioned system are located within a uniformly bounded annulus whose bandwidth is determined only by the parameter α. Our eigenvalue analysis also explains why the preconditioner with $\alpha = 1$ may perform poorly in some cases. Making use of the obtained eigenvalue bounds, a mesh-independent convergence rate of the preconditioned GMRES is proved under a certain condition regarding the choice of α. Numerical results with both linear and nonlinear models confirmed our theoretical findings and the satisfactory efficiency of our proposed preconditioner.

The promising improvement of an α-circulant preconditioner over the standard circulant preconditioner was inadequately discussed in the preconditioning literature, which deserves more exploration, especially for those widely known iterative solvers (e.g., in solving nonsymmetric block Toeplitz systems) based on block circulant preconditioners. Our preliminary numerical results also demonstrate the great potential of such α-circulant preconditioners when applied to time-varying and nonlinear wave equations, but their rigorous convergence analysis deserves further investigation.

Appendix A. In this appendix, we will prove (3.7) regarding the key function $\phi_{N_t}(\theta) = \sum_{n=1}^{N_t} \frac{\sin^2(n\theta)}{\sin^2\theta}$ given in (3.6).

LEMMA A.1. *Let $\theta \in [0, \pi/2]$. For any positive integer $n \geq 1$, there holds*

$$|\sin(n\theta)| \leq n|\sin\theta|. \tag{A.1}$$

In particular, by defining (here $\theta = 0$ should be defined as its right limit)

$$g_n(\theta) := \frac{|\sin(n\theta)|}{|\sin\theta|},$$

it holds that

$$g_n(\theta) \leq n = g_n(0) := \lim_{\theta\to 0^+} \frac{|\sin(n\theta)|}{|\sin\theta|}. \tag{A.2}$$

Proof. We will proceed the proof via a mathematical induction. For $n = 1$, the inequality (A.1) obviously holds. We assume that (A.1) holds for $n = k$, i.e., $|\sin(k\theta)| \leq k|\sin\theta|$. Then, for $n = k + 1$ we have $\leq k|\sin\theta| + |\sin\theta| = (k + 1)|\sin\theta|$, which proves the inequality (A.1) for all $n \geq 1$ by the mathematical induction.

The second inequality (A.2) follows directly from (A.1) when $\theta > 0$. The fact $g_n(0) = n$ holds according to the following limit:

$$\lim_{\theta\to 0^+} \frac{|\sin(n\theta)|}{|\sin\theta|} = n \lim_{\theta\to 0^+} \frac{|\sin(n\theta)|}{|n\theta|} \frac{|\theta|}{|\sin\theta|} = n,$$

where for the second limit we used the well-known result $\lim_{\theta\to 0} \frac{\sin\theta}{\theta} = 1$. ∎

The above lemma, Lemma A.1, essentially says that $g_n(\theta)$ always attains its global maximum at $\theta = 0$ with $\max_{\theta\in[0,\pi/2]} g_n(\theta) = g_n(0) = n$, which leads to

$$\begin{aligned}\max_{\theta\in[0,\pi/2]} \phi_{N_t}(\theta) &= \max_{\theta\in[0,\pi/2]} \sum_{n=1}^{N_t} g_n^2(\theta) \le \sum_{n=1}^{N_t} \left(\max_{\theta\in[0,\pi/2]} g_n^2(\theta) \right) \\ &= \sum_{n=1}^{N_t} n^2 = \frac{1}{6} N_t(N_t+1)(2N_t+1) = \phi_{N_t}(0),\end{aligned}$$

where the equality holds if and only if $\theta = 0$. This proves the desired conclusion in (3.7).

Acknowledgments. The authors sincerely appreciate the associate editor, Prof. Andy Wathen, and two anonymous referees for their valuable comments and constructive revision suggestions that have greatly improved the original manuscript. The authors also want to thank Dr. Xiaoqiang Yue from Xiangtan University for helping do the parallel experiments on Tianhe-1 supercomputer.

REFERENCES

[1] D. Bertaccini and M. K. Ng, *Block omega-circulant preconditioners for the systems of differential equations*, Calcolo, 40 (2003), pp. 71–90, https://doi.org/10.1007/s100920300004.

[2] A. Bespalov, L. Rocchi, and D. Silvester, *T-IFISS: a toolbox for adaptive FEM computation*, Comput. Math. Appl., to appear.

[3] D. A. Bini, G. Latouche, and B. Meini, *Numerical Methods for Structured Markov Chains*, Oxford University Press, New York, 2005, https://doi.org/10.1093/acprof:oso/9780198527688.001.0001.

[4] B. L. Buzbee, G. H. Golub, and C. W. Nielson, *On direct methods for solving Poisson's equations*, SIAM J. Numer. Anal., 7 (1970), pp. 627–656, https://doi.org/10.1137/0707049.

[5] R. Chan and X. Jin, *An Introduction to Iterative Toeplitz Solvers*, SIAM, Philadelphia, 2007, https://doi.org/10.1137/1.9780898718850.

[6] F. Chen, J. S. Hesthaven, , and X. Zhu, *On the use of reduced basis methods to accelerate and stabilize the parareal method*, in Reduced Order Methods for Modeling and Computational Reduction, MS&A. Model. Simul. Appl. 9, Springer, Cham, 2014, pp. 187–214.

[7] P.-H. Cocquet and M. J. Gander, *How large a shift is needed in the shifted Helmholtz preconditioner for its effective inversion by multigrid?*, SIAM J. Sci. Comput., 39 (2017), pp. A438–A478, https://doi.org/10.1137/15m102085x.

[8] G. Cohen, *Higher-Order Numerical Methods for Transient Wave Equations*, Scientific Computation, Springer, Berlin, Heidelberg, 2001, https://books.google.com/books?id=SSmjr45ULnMC.

[9] X. Dai and Y. Maday, *Stable parareal in time method for first- and second-order hyperbolic systems*, SIAM J. Sci. Comput., 35 (2013), pp. A52–A78, https://doi.org/10.1137/110861002.

[10] D. Durran, *Numerical Methods for Wave Equations in Geophysical Fluid Dynamics*, Texts in Applied Mathematics, Springer New York, 2013, https://books.google.com/books?id=QhIhswEACAAJ.

[11] A. Eghbal, A. G. Gerber, and E. Aubanel, *Acceleration of unsteady hydrodynamic simulations using the parareal algorithm*, J. Comput. Sci., 19 (2017), pp. 57–76.

[12] S. C. Eisenstat, H. C. Elman, and M. H. Schultz, *Variational iterative methods for nonsymmetric systems of linear equations*, SIAM J. Numer. Anal., 20 (1983), pp. 345–357, https://doi.org/10.1137/0720023.

[13] H. Elman, A. Ramage, and D. Silvester, *Algorithm 866: IFISS, a Matlab toolbox for modelling incompressible flow*, ACM Trans. Math. Softw., 33 (2007), pp. 2–14.
[14] H. Elman, A. Ramage, and D. Silvester, *IFISS: A computational laboratory for investigating incompressible flow problems*, SIAM Rev., 56 (2014), pp. 261–273, https://doi.org/10.1137/120891393.
[15] H. C. Elman, D. J. Silvester, and A. J. Wathen, *Finite Elements and Fast Iterative Solvers: With Applications in Incompressible Fluid Dynamics*, Oxford University Press, Oxford, 2014.
[16] M. Embree, *How Descriptive are GMRES Convergence Bounds?*, University of Oxford Computing Lab, Oxford, UK, NA Report 99/08, 1999. https://ora.ox.ac.uk/objects/uuid:8ca2d383-4d7d-4e21-805c-98e16537d3d3.
[17] Z.-W. Fang, H.-W. Sun, and H.-Q. Wei, *An approximate inverse preconditioner for spatial fractional diffusion equations with piecewise continuous coefficients*, Int. J. Comput. Math. 97 (2020), pp. 523–545.
[18] C. Farhat, J. Cortial, C. Dastillung, and H. Bavestrello, *Time-parallel implicit integrators for the near-real-time prediction of linear structural dynamic responses*, Int. J. Numer. Methods Eng., 67 (2006), pp. 697–724.
[19] M. J. Gander, *On the origins of linear and non-linear preconditioning*, in Domain Decomposition Methods in Science and Engineering XXIII, Lect. Notes Comput. Sci. Eng. 116, Springer, Cham, 2017, pp. 153–161.
[20] M. J. Gander, I. Graham, and E. Spence, *Applying GMRES to the Helmholtz equation with shifted Laplacian preconditioning: what is the largest shift for which wavenumber-independent convergence is guaranteed?*, Numer. Math., 131 (2015), pp. 567–614.
[21] M. J. Gander and L. Halpern, *Time parallelization for nonlinear problems based on diagonalization*, in Domain Decomposition Methods in Science and Engineering XXIII, Lect. Notes Comput. Sci. Eng. 116, Springer, Cham, 2017, pp. 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), pp. A220–A245, https://doi.org/10.1137/17M1148347.
[23] M. J. Gander and M. Petcu, *Analysis of a modified parareal algorithm for second-order ordinary differential equations*, AIP Conference Proceedings, 936 (2007), pp. 233–236.
[24] A. Goddard and A. Wathen, *A note on parallel preconditioning for all-at-once evolutionary pdes*, Electron. Trans. Numer. Anal., 51 (2019), pp. 135–150.
[25] G. Golub and C. Van Loan, *Matrix Computations*, Johns Hopkins University Press, Baltimore, MD, 2012, https://books.google.com/books?id=5U-l8U3P-VUC.
[26] W. Govaerts and J. Pryce, *A singular value inequality for block matrices*, Linear Algebra Appl., 125 (1989), pp. 141–148.
[27] I. Graham, E. Spence, and E. Vainikko, *Domain decomposition preconditioning for high-frequency Helmholtz problems with absorption*, Math. Comp., 86 (2017), pp. 2089–2127.
[28] A. Greenbaum, V. Pták, and Z. e. k. Strakoš, *Any nonincreasing convergence curve is possible for GMRES*, SIAM J. Matrix Anal. Appl., 17 (1996), pp. 465–469, https://doi.org/10.1137/S0895479894275030.
[29] W. Hackbusch, *Elliptic Differential Equations: Theory and Numerical Treatment*, Springer Series in Computational Mathematics, Springer, Berlin, Heidelberg, 2017, https://books.google.com/books?id=gmsmDwAAQBAJ.
[30] J. S. Hesthaven, *Numerical Methods for Conservation Laws: From Analysis to Algorithms*, SIAM, Philadelphia, 2018, https://doi.org/10.1137/1.9781611975109.
[31] R. W. Hockney, *A fast direct solution of Poisson's equation using Fourier analysis*, J. ACM, 12 (1965), pp. 95–113, https://doi.org/10.1145/321250.321259.
[32] R. A. Horn, R. A. Horn, and C. R. Johnson, *Topics in Matrix Analysis*, Cambridge University Press, Cambridge, 1994.
[33] R. J. LeVeque, *Numerical Methods for Conservation Laws*, 2nd ed., Birkhäuser Verlag, Basel, 1992.
[34] O. Levin, *Discrete Mathematics. An Open Introduction*, 3rd ed., 2019, http://discrete.openmathbooks.org/dmoi3.html.
[35] B. Li, J. Liu, and M. Xiao, *A fast and stable preconditioned iterative method for optimal control problem of wave equations*, SIAM J. Sci. Comput., 37 (2015), pp. A2508–A2534, https://doi.org/10.1137/15M1020526.
[36] X. Lin, M. K. Ng, and H. Sun, *A separable preconditioner for time-space fractional Caputo-Riesz diffusion equations*, Numer. Math. Theory Methods Appl., 11 (2018), pp. 827–853.

[37] X.-L. Lin and M. Ng, *An All-at-Once Preconditioner for Evolutionary Partial Differential Equations*, preprint https://arxiv.org/abs/2002.01108, 2020.
[38] X. Lu, H.-K. Pang, and H.-W. Sun, *Fast approximate inversion of a block triangular Toeplitz matrix with applications to fractional sub-diffusion equations*, Numer. Linear Algebra Appl., 22 (2015), pp. 866–882.
[39] X. Lu, H.-K. Pang, H.-W. Sun, and S.-W. Vong, *Approximate inversion method for time-fractional subdiffusion equations*, Numer. Linear Algebra Appl., 25 (2018), e2132.
[40] S. P. MacLachlan and C. W. Oosterlee, *Algebraic multigrid solvers for complex-valued matrices*, SIAM J. Sci. Comput., 30 (2008), pp. 1548–1571, https://doi.org/10.1137/070687232.
[41] Y. Maday and E. M. Rønquist, *Parallelization in time through tensor-product space–time solvers*, C. R. Math. Acad. Sci. Paris, 346 (2008), pp. 113–118, https://doi.org/10.1016/j.crma.2007.09.012.
[42] E. McDonald, J. Pestana, and A. Wathen, *Preconditioning and iterative solution of all-at-once systems for evolutionary partial differential equations*, SIAM J. Sci. Comput., 40 (2018), pp. A1012–A1033, https://doi.org/10.1137/16m1062016.
[43] G. Meurant and J. D. Tebbens, *The role eigenvalues play in forming GMRES residual norms with non-normal matrices*, Numer. Algorithms, 68 (2015), pp. 143–165.
[44] C. D. Meyer, *Matrix Analysis and Applied Linear Algebra*, SIAM, Philadelphia, 2000.
[45] A. Napov and Y. Notay, *An algebraic multigrid method with guaranteed convergence rate*, SIAM J. Sci. Comput., 34 (2012), pp. A1079–A1109, https://doi.org/10.1137/100818509.
[46] M. Ng, *Iterative Methods for Toeplitz Systems*, Iterative Methods for Toeplitz Systems, Oxford University Press, New York, 2004, https://books.google.com/books?id=o4I9TTWRE50C.
[47] H. Nguyen and R. Tsai, *A stable parareal-like method for the second order wave equation*, J. Comput. Phys., 405 (2020), pp. 109–156.
[48] Y. Notay, *An aggregation-based algebraic multigrid method*, Electron. Trans. Numer. Anal., 37 (2010), pp. 123–146.
[49] Y. Notay, *Aggregation-based algebraic multigrid for convection-diffusion equations*, SIAM J. Sci. Comput., 34 (2012), pp. A2288–A2316, https://doi.org/10.1137/110835347.
[50] Y. Notay, *AGMG Software and Documentation*, http://agmg.eu, 2020.
[51] P.-O. Persson and G. Strang, *A simple mesh generator in MATLAB*, SIAM Rev., 46 (2004), pp. 329–345, https://doi.org/10.1137/S0036144503429121.
[52] J. Pestana, *Preconditioners for symmetrized Toeplitz and multilevel Toeplitz matrices*, SIAM J. Matrix Anal. Appl., 40 (2019), pp. 870–887, https://doi.org/10.1137/18M1205406.
[53] J. Pestana and A. J. Wathen, *A preconditioned MINRES method for nonsymmetric Toeplitz matrices*, SIAM J. Matrix Anal. Appl., 36 (2015), pp. 273–288, https://doi.org/10.1137/140974213.
[54] D. Ruprecht, *Wave propagation characteristics of parareal*, Comput. Visual Sci., 59 (2018), pp. 1–17.
[55] D. Ruprecht and R. Krause, *Explicit parallel-in-time integration of a linear acoustic-advection system*, Comput. Fluids, 59 (2012), pp. 72–83.
[56] Y. Saad, *A flexible inner-outer preconditioned GMRES algorithm*, SIAM J. Sci. Comput., 14 (1993), pp. 461–469, https://doi.org/10.1137/0914028.
[57] Y. Saad, *Iterative Methods for Sparse Linear Systems: Second Edition*, SIAM, Philadelphia, 2003, https://doi.org/10.1137/1.9780898718003.
[58] Y. Saad and M. H. Schultz, *GMRES: a generalized minimal residual algorithm for solving nonsymmetric matrix systems*, SIAM J. Sci. Stat. Comput., 7 (1986), pp. 856–869, https://doi.org/10.1137/0907058.
[59] D. Silvester, H. Elman, and A. Ramage, *Incompressible Flow and Iterative Solver Software (IFISS) version* 3.6, February 2019, http://www.manchester.ac.uk/ifiss/.
[60] V. Simoncini and D. B. Szyld, *Flexible inner-outer Krylov subspace methods*, SIAM J. Numer. Anal., 40 (2002), pp. 2219–2239, https://doi.org/10.1137/S0036142902401074.
[61] V. Simoncini and D. B. Szyld, *Recent computational developments in Krylov subspace methods for linear systems*, Numer. Linear Algebra Appl., 14 (2007), pp. 1–59.
[62] M. Struwe, *Semi-linear wave equations*, Bull. Amer. Math. Soc., 26 (1992), pp. 53–85.
[63] D. Titley-Peloquin, J. Pestana, and A. J. Wathen, *GMRES convergence bounds that depend on the right-hand-side vector*, IMA J. Numer. Anal., 34 (2013), pp. 462–479.
[64] A. J. Wathen, *Preconditioning*, Acta Numer., 24 (2015), pp. 329–376.
[65] S.-L. Wu, *Toward parallel coarse grid correction for the parareal algorithm*, SIAM J. Sci. Comput., 40 (2018), pp. A1446–A1472, https://doi.org/10.1137/17m1141102.

[66] S.-L. Wu, H. Zhang, and T. Zhou, *Solving time-periodic fractional diffusion equations via diagonalization technique and multigrid*, Numer. Linear Algebra Appl., 25 (2018), e2178, https://doi.org/10.1002/nla.2178.

[67] X. Yang, X. Liao, W. Xu, J. Song, Q. Hu, J. Su, L. Xiao, K. Lu, Q. Dou, J. Jiang, and C. Yang, *Th-1: China's first petaflop supercomputer*, Front. Comput. Sci. China, 4 (2010), pp. 445–455.