SIAM J. Sci. Comput.
Vol. 46, No. 3, pp. A2073–A2100

EFFICIENT AND PARALLEL SOLUTION OF HIGH-ORDER CONTINUOUS TIME GALERKIN FOR DISSIPATIVE AND WAVE PROPAGATION PROBLEMS*

ZHIMING CHEN† AND YONG LIU‡

Abstract. We propose efficient and parallel algorithms for the implementation of the high-order continuous time Galerkin method for dissipative and wave propagation problems. By using Legendre polynomials as shape functions, we obtain a special structure of the stiffness matrix that allows us to extend the diagonal Padé approximation to solve ordinary differential equations with source terms. The unconditional stability, *hp* error estimates, and *hp* superconvergence at the nodes of the continuous time Galerkin method are proved. Numerical examples confirm our theoretical results.

Key words. implicit time discretization, Padé approximation, parallel implementation

MSC code. 65M60

DOI. 10.1137/23M1572787

1. Introduction.

In this paper, we study the following system of ordinary differential equations (ODEs)

$$\mathbf{Y}'(t) = \mathbb{D}\mathbf{Y}(t) + \mathbf{R}(t) \text{ in } (0,T), \qquad \mathbf{Y}(0) = \mathbf{Y}_0, \tag{1.1}$$

which is obtained from the method-of-lines approach for linear partial differential equations (PDEs) after space discretization. Here $T > 0$ is the length of the time interval, $\mathbf{Y}, \mathbf{R} \in \mathbb{R}^M$, and $\mathbb{D}$ is an $M \times M$ real constant matrix, where M is the number of degrees of freedom of the spatial discretization. Without loss of generality, we assume

$$\mathbb{D} + \mathbb{D}^T \le 0, \tag{1.2}$$

that is, $\mathbb{D} + \mathbb{D}^T$ is a seminegative definite matrix. This condition is satisfied by a large class of linear PDEs including dissipative problems such as parabolic equations and wave propagation problems such as the wave equation and Maxwell equations.

Let $0 = t_0 < t_1 < \cdots < t_N = T$ be a partition of $(0,T)$. If the source $\mathbf{R} = \mathbf{0}$ in (1.1), the exact solution in each time interval (t_n, t_{n+1}) is $\mathbf{Y}(t) = e^{\mathbb{D}(t-t_n)}\mathbf{Y}(t_n)$ for which Padé approximation to the exponential function can be used to construct and analyze numerical schemes to solve (1.1). In [12], by using the partial fraction formula for

*Submitted to the journal's Numerical Algorithms for Scientific Computing section May 15, 2023; accepted for publication (in revised form) March 12, 2024; published electronically June 19, 2024.
https://doi.org/10.1137/23M1572787

Funding: The work of the first author was supported in part by China National Key Technologies R&D Program grant 2019Y-FA0709600 and by National Natural Science Foundation of China grants 11831016 and 12288201. The work of the second author was partially supported by National Natural Science Foundation of China grants 12201621 and 12288201, and by Youth Innovation Promotion Association CAS.

†LSEC, Institute of Computational Mathematics, Academy of Mathematics and Systems Science and School of Mathematical Science, University of Chinese Academy of Sciences, Chinese Academy of Sciences, Beijing 100190, People's Republic of China (zmchen@lsec.cc.ac.cn).

‡LSEC, Institute of Computational Mathematics, Academy of Mathematics and Systems Science, Chinese Academy of Sciences, Beijing 100190, People's Republic of China (yongliu@lsec.cc.ac.cn).

the Padé approximation, the $[r/r]$, $r \geq 1$, Padé approximation leads to the following method:

$$\mathbf{Y}(t_{n+1}) \approx \frac{P_r(\tau_n \mathbb{D})}{P_r(-\tau_n \mathbb{D})}\mathbf{Y}(t_n) = \left[(-1)^r \mathbb{I} + \sum_{j=1}^{r} -\frac{P_r(-\zeta_j)}{P_r'(\zeta_j)}(\zeta_j \mathbb{I} + \tau_n \mathbb{D})^{-1}\right]\mathbf{Y}(t_n), \tag{1.3}$$

where $\tau_n = t_{n+1} - t_n$, $\mathbb{I} \in \mathbb{R}^{M\times M}$ is the identity matrix, $P_r(z)$ is the numerator of the $[r/r]$ Padé approximation to the exponential function e^z, and $\{\zeta_1, \ldots, \zeta_r\}$ are zeros of $P_r(z)$, which are known to be simple and lie in the left half-plane. Formula (1.3) indicates that one can compute the approximation of the solution $\mathbf{Y}(t_{n+1})$ in each time step by solving k complex matrix problems and $r - 2k$ real matrix problems of the form $\zeta \mathbb{I} + \tau_n \mathbb{D}$ in parallel, where k, $0 \leq k \leq r/2$, is the number of complex zeros of $P_r(z)$ (see Remark 3.1 below). The purpose of this paper is to construct algorithms sharing this very desirable property for solving (1.1) with general nonzero sources $\mathbf{R}(t)$.

There exists a large literature on implicit single-step time-stepping methods for solving (1.1) (see, e.g., [15] and the references therein). The following continuous time Galerkin method proposed in [16] is probably the simplest:

$$\mathbf{Y}_r' = \mathbb{D}\mathcal{P}_{r-1}\mathbf{Y}_r + \mathcal{P}_{r-1}\mathbf{R} \quad \text{in } (t_n, t_{n+1}), \quad 0 \leq n \leq N-1, \tag{1.4}$$

where $\mathbf{Y}_r$ is a piecewise polynomial of degree $r \geq 1$ in each interval (t_n, t_{n+1}), continuous at the nodes $t = t_n$, and $\mathcal{P}_{r-1}$ is the local L^2 projection to the space of polynomials of degree at most $(r-1)$ in each interval. It is shown in [16] that (1.4) is equivalent to the r-stage Gauss collocation method at the nodes when $\mathbf{R} = \mathbf{0}$ and has the highest classical order $2r$ among all r-stage Runge–Kutta methods [15, Table 5.12]. The continuous time Galerkin method, together with finite element discretization in space, is used in [2] for the heat equation and in [11], [14] for the wave equation. We refer the reader to [1] for a unified framework and the comparison of the most popular implicit single-step time-stepping methods, including also the discontinuous time Galerkin method and various Runge–Kutta methods.

The difficulty in using the high-order continuous time Galerkin method or any fully implicit time Runge–Kutta methods is that a straightforward implementation requires one to solve a system of linear equations of the size $rM \times rM$, which is not feasible in most time for PDE problems. In a recent work [22], efficient iterative algorithms are developed based on optimal preconditioning of the stage matrix for finding the stage vectors of the implicit Runge–Kutta methods for solving (1.1). For an r-stage implicit Runge–Kutta method, the stage matrix is an $r \times r$ block matrix with each block being an $M \times M$ matrix. One can find further references in [22] for developing efficient algorithms implementing the high-order implicit time discretization methods in the literature. We also refer the reader to [21], [18] for the implementation of the discontinuous time Galerkin method based on the block diagnalization of the stiffness matrix.

In this paper we propose an efficient realization of the method (1.4) that uses Legendre polynomials as shape functions to obtain a new stiffness matrix that is different from the stage matrix in [22] applying to the Gauss collocation method. By exploiting the special structure of the stiffness matrix, we construct an algorithm that computes the solution $\mathbf{Y}_r(t_n)$ at each node by solving k complex matrix problems and $r - 2k$ real matrix problems in parallel, where k, $0 \leq k \leq r/2$, is the number of complex zeros of the $[r/r]$ Padé numerator $P_r(z)$. Moreover, an algorithm is proposed to compute the other coefficients of the solution $\mathbf{Y}_r$ in each time interval (t_n, t_{n+1}) that solves in parallel $k(r+1)$ complex matrix problems and $(r-2k)(r+1)$ real matrix problems. We remark that our algorithm is different from the other parallel-in-time algorithms based on domain decomposition or space-time multigrid techniques in the literature (see, e.g., [13]).

As a by-product of our analysis, we obtain the following formula (Theorem 3.3) to compute the nodal values $\mathbf{Y}_r(t_{n+1})$, $0 \le n \le N-1$, of the solution of (1.4):

$$\mathbf{Y}_r(t_{n+1}) = \frac{P_r(\tau_n\mathbb{D})}{P_r(-\tau_n\mathbb{D})}\mathbf{Y}_r(t_n) + \sum_{k=1}^{r}(-1)^{k+1}\frac{\phi_{k1}(\tau_n\mathbb{D})}{P_r(-\tau_n\mathbb{D})}\mathbf{b}_{k-1} + \tau_n\mathbf{R}_0, \tag{1.5}$$

where for $k = 1,\ldots,r$, $\phi_{k1}(\lambda)$ is a polynomial of degree r satisfying some recurrence relations, and $\mathbf{b}_k$, $\mathbf{R}_0$ are vectors depending on the source $\mathbf{R}$. Formula (1.5) can be viewed as a generalization of the $[r/r]$ Padé approximation (1.3) for solving the ODE system without sources.

The layout of the paper is as follows. In section 2 we introduce the continuous time Galerkin method for (1.1) and prove the strong stability and derive an *hp* error estimate. In section 3 we propose our parallel algorithms to implement the continuous time Galerkin method. In section 4 we prove the optimal stability and error estimates in terms of r when $\mathbb{D}$ is a symmetric or skew-symmetric matrix. In section 5 we consider the application of the algorithms in this paper to solve the linear convection-diffusion equation by using the local discontinuous Galerkin method and the wave equation with discontinuous coefficients by using the unfitted finite element spatial discretization.

2. Implicit time discretization.

In this section, we introduce the continuous time Galerkin method for solving (1.1). Let $0 = t_0 < t_1 < \cdots < t_N = T$ be a partition of the time interval $(0, T)$ with time steps $\tau_n = t_{n+1} - t_n$, $0 \le n \le N-1$. We set $I_n = (t_n, t_{n+1})$ and $\tau = \max_{0\le n\le N-1}\{\tau_n\}$. For any integer $m \ge 1$, we define the finite element space

$$\mathbf{V}_\tau^m := \{\mathbf{v} \in [C(0,T)]^M : \mathbf{v}|_{I_n} \in [P^m]^M, 0 \le n \le N-1\},$$

where P^m is the set of polynomials whose degree is at most m. Define the local projection $\mathcal{P}_m$, $m \ge 0$, such that in each time interval I_n, $\mathcal{P}_m : [L^2(I_n)]^M \to [P^m]^M$ satisfies

$$\int_{I_n}(\mathcal{P}_m\mathbf{v}, \mathbf{w})\,dt = \int_{I_n}(\mathbf{v}, \mathbf{w})\,dt \quad \forall \mathbf{w} \in [P^m]^M,$$

where we denote by $(\cdot,\cdot)$ the inner product of $\mathbb{R}^M$. It is well known (see, e.g., Schötzau and Schwab [20]) that for any $s \ge 0$, $m \ge 0$,

$$\|\mathbf{v} - \mathcal{P}_m\mathbf{v}\|_{L^2(I_n)} \le C\frac{\tau_n^{\min(m+1,s)}}{(m+1)^s}\|\mathbf{v}\|_{H^s(I_n)} \quad \forall \mathbf{v} \in [H^s(I_n)]^M, \tag{2.1}$$

where the constant C is independent of m, τ_n but may depend on s. In this paper, for any integer $d \ge 1$ and Banach space X, we denote by $\|\cdot\|_X$ both the norm of X and that of $[X]^d$.

For any integer $r \ge 1$, the continuous time Galerkin method for solving (1.1) is to find the function $\mathbf{Y}_r \in \mathbf{V}_\tau^r$ such that $\mathbf{Y}_r(0) = \mathbf{Y}_0$ and

$$\mathbf{Y}_r' = \mathbb{D}\mathcal{P}_{r-1}\mathbf{Y}_r + \mathcal{P}_{r-1}\mathbf{R} \ \text{ in } I_n, \quad 0 \le n \le N-1. \tag{2.2}$$

The following stability lemma extends an idea in Griesmaier and Monk [14] where the continuous time Galerkin discretization in time and hybridizable discontinuous Galerkin method in space for the wave equation are considered.

LEMMA 2.1. *Problem* (2.2) *has a unique solution* $\mathbf{Y}_r \in \mathbf{V}_\tau^r$ *that satisfies*

$$\max_{1\le n\le N} \|\mathbf{Y}_r(t_n)\|_{\mathbb{R}^M} \le \|\mathbf{Y}_0\|_{\mathbb{R}^M} + CT^{1/2}\|\mathbf{R}\|_{L^2(0,T)}, \tag{2.3}$$

$$\max_{0\le t\le T} \|\mathbf{Y}_r\|_{\mathbb{R}^M} \le Cr^2(\|\mathbf{Y}_0\|_{\mathbb{R}^M} + T^{1/2}\|\mathbf{R}\|_{L^2(0,T)}), \tag{2.4}$$

where the constant C *is independent of* $r, \tau, \mathbb{D}$, *and* $\mathbf{R}$.

Proof. At each time step, (2.2) is equivalent to a linear system of equations whose existence and uniqueness of the solution follow from the stability estimate (2.4). To prove the stability estimates (2.3)–(2.4), we denote by $\{L_j\}_{j=0}^\infty$ the Legendre polynomials on $(-1,1)$ and define $\widetilde{L}_j = L_j \circ \psi^{-1}$, where $\psi : (-1,1) \to (t_n, t_{n+1})$ is the mapping $\psi(\xi) = \frac{t_n+t_{n+1}}{2} + \frac{t_{n+1}-t_n}{2}\xi$ for $\xi \in (-1,1)$. Then $\{\widetilde{L}_j\}_{j=0}^\infty$ are orthogonal in $L^2(I_n)$, $\widetilde{L}_r(t_n) = (-1)^r, \widetilde{L}_r(t_{n+1}) = 1$, and

$$\int_{I_n} |\widetilde{L}_r|^2 dt = \frac{\tau_n}{2r+1}, \quad \int_{I_n} |\widetilde{L}_r'|^2 dt = \frac{2r(r+1)}{\tau_n}. \tag{2.5}$$

For $n = 0, \ldots, N-1$, let $\mathbf{Y}_r^n = \mathbf{Y}_r(t_n)$ and $\hat{\mathbf{Y}}_r \in [P^r]^M$ satisfy

$$\hat{\mathbf{Y}}_r' = \mathbb{D}\mathcal{P}_{r-1}\hat{\mathbf{Y}}_r \text{ in } I_n, \ \ \hat{\mathbf{Y}}_r(t_n) = \mathbf{Y}_r^n. \tag{2.6}$$

By multiplying (2.6) by $\hat{\mathbf{Y}}_r$ and integrating over I_n, we easily obtain by (1.2) that

$$\frac{1}{2}\|\hat{\mathbf{Y}}_r(t_{n+1})\|_{\mathbb{R}^M}^2 - \frac{1}{2}\|\mathbf{Y}_r^n\|_{\mathbb{R}^M}^2 = \int_{I_n} (\mathbb{D}\mathcal{P}_{r-1}\hat{\mathbf{Y}}_r, \mathcal{P}_{r-1}\hat{\mathbf{Y}}_r) dt \le 0.$$

This implies

$$\|\hat{\mathbf{Y}}_r(t_{n+1})\|_{\mathbb{R}^M} \le \|\mathbf{Y}_r^n\|_{\mathbb{R}^M}. \tag{2.7}$$

Since $\hat{\mathbf{Y}}_r \in [P^r]^M$ in I_n, we have the following decomposition introduced in [14]:

$$\hat{\mathbf{Y}}_r = (-1)^r \mathbf{Y}_r^n \widetilde{L}_r + (t-t_n)\widetilde{\mathbf{Y}}_r, \quad \widetilde{\mathbf{Y}}_r \in [P^{r-1}]^M. \tag{2.8}$$

Notice that $\mathcal{P}_{r-1}\hat{\mathbf{Y}}_r = \mathcal{P}_{r-1}[(t-t_n)\widetilde{\mathbf{Y}}_r]$; substituting this decomposition into (2.6), we have

$$(-1)^r \mathbf{Y}_r^n \widetilde{L}_r' + \widetilde{\mathbf{Y}}_r + (t-t_n)\widetilde{\mathbf{Y}}_r' = \mathbb{D}\mathcal{P}_{r-1}[(t-t_n)\widetilde{\mathbf{Y}}_r] \text{ in } I_n.$$

Multiplying the equation by $\widetilde{\mathbf{Y}}_r \in [P^{r-1}]^M$ and integrating over I_n, we have by (2.5) that

$$\frac{1}{2}\|\widetilde{\mathbf{Y}}_r\|_{L^2(I_n)}^2 + \frac{1}{2}\tau_n\|\widetilde{\mathbf{Y}}_r(t_{n+1})\|_{\mathbb{R}^M}^2 \le C\tau_n^{-1/2} r\|\mathbf{Y}_r^n\|_{\mathbb{R}^M}\|\widetilde{\mathbf{Y}}_r\|_{L^2(I_n)},$$

where we have used the fact that by (1.2)

$$\int_{I_n} (\mathbb{D}\mathcal{P}_{r-1}[(t-t_n)\widetilde{\mathbf{Y}}_r], \widetilde{\mathbf{Y}}_r) dt = \int_{I_n} (t-t_n)(\mathbb{D}\widetilde{\mathbf{Y}}_r, \widetilde{\mathbf{Y}}_r) dt \le 0. \tag{2.9}$$

This yields $\|\widetilde{\mathbf{Y}}_r\|_{L^2(I_n)} \le C\tau_n^{-1/2} r\|\mathbf{Y}_r^n\|_{\mathbb{R}^M}$ and thus by using (2.5)

$$\|\hat{\mathbf{Y}}_r\|_{L^2(I_n)} \le C\tau_n^{1/2} r\|\mathbf{Y}_r^n\|_{\mathbb{R}^M}. \tag{2.10}$$

On the other hand, it follows from (2.2) and (2.6) that

$$(\mathbf{Y}_r - \hat{\mathbf{Y}}_r)' = \mathbb{D}\mathcal{P}_{r-1}(\mathbf{Y}_r - \hat{\mathbf{Y}}_r) + \mathcal{P}_{r-1}\mathbf{R} \ \text{ in } I_n, \quad (\mathbf{Y}_r - \hat{\mathbf{Y}}_r)(t_n) = 0. \tag{2.11}$$

Then $\mathbf{Y}_r - \hat{\mathbf{Y}}_r = (t - t_n)\mathbf{W}_r$ for some $\mathbf{W}_r \in [P^{r-1}]^M$. By substituting this relation into (2.11) we have

$$\mathbf{W}_r + (t - t_n)\mathbf{W}_r' = \mathbb{D}\mathcal{P}_{r-1}[(t - t_n)\mathbf{W}_r] + \mathcal{P}_{r-1}\mathbf{R} \ \text{ in } I_n.$$

By multiplying the equation by $\mathbf{W}_r$ and integrating over I_n, we obtain by a bound similar to that in (2.9) that

$$\frac{1}{2}\|\mathbf{W}_r\|^2_{L^2(I_n)} + \frac{1}{2}\tau_n\|\mathbf{W}_r(t_{n+1})\|^2_{\mathbb{R}^M} \le \|\mathbf{R}\|_{L^2(I_n)}\|\mathbf{W}_r\|_{L^2(I_n)}.$$

This yields $\|\mathbf{W}_r\|_{L^2(I_n)} \le 2\|\mathbf{R}\|_{L^2(I_n)}$ and thus

$$\|\mathbf{Y}_r - \hat{\mathbf{Y}}_r\|_{L^2(I_n)} \le 2\tau_n\|\mathbf{R}\|_{L^2(I_n)}. \tag{2.12}$$

Now by multiplying (2.11) by $\mathbf{Y}_r - \hat{\mathbf{Y}}_r$ and integrating over I_n we obtain by (1.2) and (2.12) that

$$\frac{1}{2}\|(\mathbf{Y}_r - \hat{\mathbf{Y}}_r)(t_{n+1})\|^2_{\mathbb{R}^M} \le \int_{I_n} (\mathcal{P}_{r-1}\mathbf{R}, \mathbf{Y}_r - \hat{\mathbf{Y}}_r)dt \le 2\tau_n\|\mathbf{R}\|^2_{L^2(I_n)},$$

which implies by the triangle inequality and (2.7) that

$$\|\mathbf{Y}_r(t_{n+1})\|_{\mathbb{R}^M} \le \|\mathbf{Y}_r^n\|_{\mathbb{R}^M} + 2\tau_n^{1/2}\|\mathbf{R}\|_{L^2(I_n)}.$$

This yields (2.3). Next by using the triangle inequality, (2.10), and (2.12), we have

$$\|\mathbf{Y}_r\|_{L^2(I_n)} \le C\tau_n^{1/2} r\|\mathbf{Y}_r^n\|_{\mathbb{R}^M} + 2\tau_n\|\mathbf{R}\|_{L^2(I_n)}, \tag{2.13}$$

which implies by the *hp* inverse estimate that

$$\max_{t_n \le t \le t_{n+1}} \|\mathbf{Y}_r\|_{\mathbb{R}^M} \le C\tau_n^{-1/2} r\|\mathbf{Y}_r\|_{L^2(I_n)} \le Cr^2\|\mathbf{Y}_r^n\|_{\mathbb{R}^M} + C\tau_n^{1/2} r\|\mathbf{R}\|_{L^2(I_n)}.$$

This shows (2.4) and completes the proof of the lemma. ∎

To derive an *hp* a priori error estimate for the continuous time Galerkin method (2.2), we first recall an interpolation operator in the literature (see, e.g., [20, Theorem 3.17]).

LEMMA 2.2. *There exists an interpolation operator* $\Pi_r : [H^1(0,T)]^M \to \mathbf{V}_\tau^r$ *such that for any* $\mathbf{v} \in [W^{1+s,\infty}(0,T)]^M$, $s \ge 1$, *and* $n = 0, 1, \dots, N-1$,

$$(\Pi_r\mathbf{v})(t_n) = \mathbf{v}(t_n), \ (\Pi_r\mathbf{v})(t_{n+1}) = \mathbf{v}(t_{n+1}), \ (\Pi_r\mathbf{v})' = \mathcal{P}_{r-1}\mathbf{v}' \ \text{in } I_n, \tag{2.14}$$

$$\|\mathbf{v} - \Pi_r\mathbf{v}\|_{L^2(I_n)} \le C\frac{\tau^{\min(r+1,s)}}{r^s}\|\mathbf{v}\|_{H^s(I_n)}, \tag{2.15}$$

where the constant C *is independent of* τ, r *but may depend on* s.

With the interpolation operator Π_r, one can prove the following theorem on the *hp* error estimates by the standard argument. Here we omit the details.

THEOREM 2.3. *Let $s \geq 1$. Assuming that $\mathbf{R} \in [H^s(0,T)]^M$, $\mathbf{Y} \in [W^{1+s,\infty}(0,T)]^M$ and $\mathbf{Y}_r \in \mathbf{V}_\tau^r$ is the solution of the problem* (2.2), *we have*

$$\max_{1\leq n\leq N} \|(\mathbf{Y}-\mathbf{Y}_r)(t_n)\|_{\mathbb{R}^M} \leq CT^{1/2}\frac{\tau^{\min(r+1,s)}}{r^s}\|\mathbb{D}\mathbf{Y}\|_{H^s(0,T)},$$

$$\max_{0\leq t\leq T} \|\mathbf{Y}-\mathbf{Y}_r\|_{\mathbb{R}^M} \leq C(1+T^{1/2})\frac{\tau^{\min(r+1,s)}}{r^{s-2}}(T^{1/2}\|\mathbf{Y}\|_{W^{s+1,\infty}(0,T)}+\|\mathbf{R}\|_{H^s(0,T)}),$$

where the constant C is independent of $\tau, r, \mathbb{D}$ but may depend on s.

We remark that the first estimate in Theorem 2.3 is optimal in τ and r and the second estimate in the theorem is optimal in τ but suboptimal in r, which is due to the stability estimate (2.4) in Lemma 2.1. In section 4 we will show that the stability in the L^2 norm can be improved to remove the dependence on r in (2.13) when $\mathbb{D}$ is symmetric or skew-symmetric by using the explicit formulas of $\mathbf{Y}_r(t)$ in section 3.

The classical order of Runge–Kutta methods is the convergence order at the nodes $t = t_n$. For the continuous time Galerkin method, it is proved to be $2r$ when $r \geq 2$ in Hulme [16] for nonlinear ODEs and in Aziz and Monk [2] for parabolic equations. The following theorem on the *hp* superconvergence of the continuous time Galerkin method at the nodes can be proved by using the idea of quasi-projection in [2, section 4]. Here we omit the details.

THEOREM 2.4. *Let $s \geq 1$. Assuming that $\mathbb{D}^r\mathbf{Y} \in [H^s(0,T)]^M$ and $\mathbf{Y}_r \in \mathbf{V}_\tau^r$ is the solution of problem* (2.2), *we have*

$$\max_{1\leq n\leq N} \|(\mathbf{Y}-\mathbf{Y}_r)(t_n)\|_{\mathbb{R}^M} \leq CT^{1/2}\frac{\tau^{\min(2r,s+r-1)}}{r^s}\|\mathbb{D}^r\mathbf{Y}\|_{H^s(0,T)},$$

where the constant C is independent of τ, r but may depend on s.

To conclude this section, we recall some facts about Padé approximation to the exponential function, which can be found in Saff and Varga [19] and the references therein. For any integers $m, n \geq 0$, the $[m/n]$ Padé approximation to e^z is defined as the polynomials $P_m(z) \in P^m$, $Q_n(z) \in P^n$, $Q_n(0) = 1$, for which $e^z - \frac{P_m(z)}{Q_n(z)} = O(|z|^{m+n+1})$ as $|z| \to 0$. It is known that

$$P_m(z) = \sum_{j=0}^{m} \frac{(m+n-j)!m!z^j}{(m+n)!j!(m-j)!}, \quad Q_n(z) = \sum_{j=0}^{n} \frac{(m+n-j)!n!(-z)^j}{(m+n)!j!(n-j)!}. \tag{2.16}$$

Obviously, $Q_n(z) = P_n(-z)$. When $m = n$, $P_m(z), Q_m(z)$ are called the diagonal Padé numerator and denominator of type $[m/m]$ for e^z. The following lemma follows easily from (2.16).

LEMMA 2.5. *The diagonal Padé numerator of type $[m/m]$ for e^z satisfies $P_0(z) = 1, P_1(z) = 1 + \frac{1}{2}z$, $P_2(z) = 1 + \frac{1}{2}z + \frac{1}{12}z^2$, and*

$$P_m(z) = P_{m-1}(z) + \frac{z^2}{4(2m-1)(2m-3)}P_{m-2}(z), \qquad m \geq 2.$$

The following lemma is proved in Hairer and Wanner [15, Theorem 4.12] and [19, Theorem 2.4]. It is essential in proving the A-stability of numerical methods for ODEs based on the Padé approximation of the exponential function.

LEMMA 2.6. *All zeros of the diagonal Padé numerator of type $[m/m]$, $m \geq 1$, for e^z are simple and lie in the half-plane $\{z \in \mathbb{C} : Re(z) \leq -2\}$.*

For $m \geq 1$, denote by $\zeta_1, \ldots, \zeta_m \in \mathbb{C}$ the zeros of $P_m(z)$, the diagonal Padé numerator of type $[m/m]$ for e^z. Then by (2.16)

$$P_m(z) = \frac{m!}{(2m)!}(z-\zeta_1)\cdots(z-\zeta_m), \quad Q_m(z) = (-1)^m \frac{m!}{(2m)!}(z+\zeta_1)\cdots(z+\zeta_m).$$

One can compute the zeros of $P_m(z)$ by using the "roots" function in MATLAB [17]. Recall that any polynomial $F \in P^{m-1}$ can be expanded as the Lagrange interpolation function at m distinct zeros of $P_m(-z)$. This yields the following partial fraction formula (see, e.g., Szegö [23, Theorem 3.3.5]):

$$\frac{F(z)}{P_m(-z)} = \sum_{j=1}^{m} -\frac{F(-\zeta_j)}{P'_m(\zeta_j)} \frac{1}{z+\zeta_j}. \tag{2.17}$$

Since $P_m(z) - (-1)^m P_m(-z) \in P^{m-1}$, we obtain the partial fraction formula for the $[m/m]$ Padé approximation of e^z (see Gallopoulos and Saad [12]):

$$R_{m,m}(z) = \frac{P_m(z)}{P_m(-z)} = (-1)^m + \sum_{j=1}^{m} -\frac{P_m(-\zeta_j)}{P'_m(\zeta_j)} \frac{1}{z+\zeta_j}. \tag{2.18}$$

Recall that if $p(z) = \sum_{i=0}^{m} a_i z^i$ is a polynomial of degree m, then $p(\mathbb{X}) := \sum_{i=0}^{m} a_i \mathbb{X}^i$ for any matrix $\mathbb{X} \in \mathbb{R}^{d \times d}$, $d \geq 1$. Obviously, if $p(z) = p_1(z) + p_2(z)$ or $q(z) = p_1(z) \cdot p_2(z)$, where p_1, p_2 are polynomials, then $p(\mathbb{X}) = p_1(\mathbb{X}) + p_2(\mathbb{X})$, $q(\mathbb{X}) = p_1(\mathbb{X}) \cdot p_2(\mathbb{X})$. It follows now from (2.17)–(2.18) that for any $F \in P^{m-1}$ and any matrix $\mathbb{X} \in \mathbb{R}^{d \times d}$ such that $P_m(-\mathbb{X})$ is invertible,

$$\frac{F(\mathbb{X})}{P_m(-\mathbb{X})} = \sum_{j=1}^{m} -\frac{F(-\zeta_j)}{P'_m(\zeta_j)} (\zeta_j \mathbb{I} + \mathbb{X})^{-1}, \tag{2.19}$$

$$\frac{P_m(\mathbb{X})}{P_m(-\mathbb{X})} = (-1)^m \mathbb{I} + \sum_{j=1}^{m} -\frac{P_m(-\zeta_j)}{P'_m(\zeta_j)} (\zeta_j \mathbb{I} + \mathbb{X})^{-1}. \tag{2.20}$$

The identity (2.20) is the basis of the method (1.3) in the introduction.

3. Parallel implementation.

In this section, we propose parallel algorithms to implement problem (2.2) based on finding analytic formulas of the determinant and all factors of the stiffness matrix of the continuous time Galerkin method at each time step. To form the stiffness matrix, we use the Legendre polynomials $\{\widetilde{L}_j\}_{j=0}^{r}$ in I_n as the basis functions. We assume

$$\mathbf{Y}_r(t) = \sum_{j=0}^{r} \mathbf{a}_j \widetilde{L}_j(t), \quad \mathcal{P}_{r-1}\mathbf{R} = \sum_{j=0}^{r-1} \mathbf{R}_j \widetilde{L}_j(t),$$

where $\mathbf{a}_j, \mathbf{R}_j \in \mathbb{R}^M$. Since $\mathcal{P}_{r-1}\widetilde{L}_r = 0$ in I_n, from (2.2), we have

$$\sum_{j=0}^{r} \mathbf{a}_j \widetilde{L}'_j(t) = \mathbb{D} \sum_{j=0}^{r-1} \mathbf{a}_j \widetilde{L}_j(t) + \sum_{j=0}^{r-1} \mathbf{R}_j \widetilde{L}_j(t).$$

For any $k \ge 1$, multiplying the equation by $(t-t_n)(t_{n+1}-t)\widetilde{L}_k'(t)$ and integrating over I_n, we obtain

$$\begin{aligned}\mathbf{a}_k \frac{\tau_n}{2}\frac{k(k+1)}{k+\frac{1}{2}} &= \mathbb{D}\sum_{j=0}^{r-1}\int_{t_n}^{t_{n+1}} \mathbf{a}_j \widetilde{L}_j(t)(t-t_n)(t_{n+1}-t)\widetilde{L}_k'(t)\,dt \\ &\quad + \sum_{j=0}^{r-1}\int_{t_n}^{t_{n+1}} \mathbf{R}_j \widetilde{L}_j(t)(t-t_n)(t_{n+1}-t)\widetilde{L}_k'(t)\,dt,\end{aligned} \tag{3.1}$$

where we have used the fact that

$$\int_{t_n}^{t_{n+1}} \widetilde{L}_j'\widetilde{L}_k'(t-t_n)(t_{n+1}-t)\,dt = \frac{\tau_n}{2}\int_{-1}^{1} L_j'L_k'(1-t^2)dt = \frac{\tau_n}{2}\frac{k(k+1)}{k+\frac{1}{2}}\delta_{j,k}.$$

Here $\delta_{j,k}$ is the Kronecker delta function. By the recursion relation $(2k+1)\widetilde{L}_k(t) = \frac{\tau_n}{2}(\widetilde{L}_{k+1}'(t) - \widetilde{L}_{k-1}'(t))$,

$$\begin{aligned}&\int_{t_n}^{t_{n+1}} \widetilde{L}_j(t)(t-t_n)(t_{n+1}-t)\widetilde{L}_k'(t)\,dt \\ &\quad = \frac{1}{2j+1}\int_{t_n}^{t_{n+1}} \frac{\tau_n}{2}(\widetilde{L}_{j+1}'(t) - \widetilde{L}_{j-1}'(t))\widetilde{L}_k'(t)(t-t_n)(t_{n+1}-t)\,dt \\ &\quad = \frac{\tau_n^2}{4}\frac{k(k+1)}{k+\frac{1}{2}}\left(\frac{1}{2k-1}\delta_{j+1,k} - \frac{1}{2k+3}\delta_{j-1,k}\right).\end{aligned}$$

Substituting the identity into (3.1), we have

$$\mathbf{a}_k = \frac{\tau_n}{2}\mathbb{D}\left(\frac{\mathbf{a}_{k-1}}{2k-1} - \frac{\mathbf{a}_{k+1}}{2k+3}\right) + \frac{\tau_n}{2}\left(\frac{\mathbf{R}_{k-1}}{2k-1} - \frac{\mathbf{R}_{k+1}}{2k+3}\right), \quad 1 \le k \le r-2, \tag{3.2}$$

$$\mathbf{a}_k = \frac{\tau_n}{2}\mathbb{D}\frac{\mathbf{a}_{k-1}}{2k-1} + \frac{\tau_n}{2}\frac{\mathbf{R}_{k-1}}{2k-1}, \quad k = r-1, r. \tag{3.3}$$

By the condition $\mathbf{Y}_r(t_n) = \mathbf{Y}_r^n$, we also have

$$\sum_{j=0}^{r}(-1)^j\mathbf{a}_j = \mathbf{Y}_r^n. \tag{3.4}$$

Equations (3.2)–(3.4) can be written as a system of linear equations

$$\mathbb{A}\mathbf{X} = \mathbf{B}, \tag{3.5}$$

where $\mathbf{X} = (\mathbf{a}_0^T, \mathbf{a}_1^T, \dots, \mathbf{a}_r^T)^T$, $\mathbf{B} = (\mathbf{b}_0^T, \mathbf{b}_1^T, \dots, \mathbf{b}_r^T)^T$ with

$$\mathbf{b}_{k-1} = \begin{cases} -\frac{\tau_n}{2}\left(\frac{\mathbf{R}_{k-1}}{2k-1} - \frac{\mathbf{R}_{k+1}}{2k+3}\right) & \text{if } 1 \le k \le r-2, \\ -\frac{\tau_n}{2}\frac{\mathbf{R}_{k-1}}{2k-1} & \text{if } k = r-1, r, \\ \mathbf{Y}_r^n & \text{if } k = r+1, \end{cases}$$

and

$$\mathbb{A} = \begin{pmatrix} \frac{\tau_n}{2}\mathbb{D} & -\mathbb{I} & -\frac{\tau_n}{2}\mathbb{D}\frac{1}{5} & \cdots & \cdots & \cdots & 0 \\ 0 & \frac{\tau_n}{2}\mathbb{D}\frac{1}{3} & -\mathbb{I} & -\frac{\tau_n}{2}\mathbb{D}\frac{1}{7} & \cdots & \cdots & 0 \\ \vdots & \vdots & \ddots & \ddots & \ddots & \vdots & \vdots \\ 0 & \cdots & \cdots & \frac{\tau_n}{2}\mathbb{D}\frac{1}{2r-5} & -\mathbb{I} & -\frac{\tau_n}{2}\mathbb{D}\frac{1}{2r-1} & 0 \\ 0 & \cdots & \cdots & \cdots & \frac{\tau_n}{2}\mathbb{D}\frac{1}{2r-3} & -\mathbb{I} & 0 \\ 0 & \cdots & \cdots & \cdots & \cdots & \frac{\tau_n}{2}\mathbb{D}\frac{1}{2r-1} & -\mathbb{I} \\ \mathbb{I} & -\mathbb{I} & \mathbb{I} & -\mathbb{I} & \cdots & \cdots & (-1)^{r+2}\mathbb{I} \end{pmatrix}.$$

Here $\mathbb{I}\in\mathbb{R}^{M\times M}$ is the identity matrix. By Lemma 2.1, (3.5) has a unique solution. Since $\mathbb{A}\in\mathbb{R}^{M(r+1)\times M(r+1)}$, it is expensive to solve (3.5) directly when $M\gg 1$. Here we propose efficient and parallel algorithms to solve (3.5).

Notice that $\mathbb{A}$ is a $(r+1)\times(r+1)$ block matrix. For $\lambda\in\mathbb{R}$, we define $\mathbb{E}_{r+1}(\lambda)\in\mathbb{R}^{(r+1)\times(r+1)}$ by

$$\mathbb{E}_{r+1}(\lambda):=\begin{pmatrix} a_1 & -1 & b_1 & \cdots & \cdots & \cdots & 0\\ 0 & a_2 & -1 & b_2 & \cdots & \cdots & 0\\ \vdots & \vdots & \ddots & \ddots & \ddots & \vdots & \vdots\\ 0 & \cdots & \cdots & a_{r-2} & -1 & b_{r-2} & 0\\ 0 & \cdots & \cdots & \cdots & a_{r-1} & -1 & 0\\ 0 & \cdots & \cdots & \cdots & \cdots & a_r & -1\\ c_1 & c_2 & \cdots & \cdots & \cdots & c_r & c_{r+1}\end{pmatrix} \tag{3.6}$$

with

$$a_k=\frac{\lambda}{2}\frac{1}{2k-1},\quad b_k=-\frac{\lambda}{2}\frac{1}{2k+3},\quad c_k=(-1)^{k+1},\quad k=1,\dots,r+1. \tag{3.7}$$

Then $\mathbb{A}=\mathbb{E}_{r+1}(\tau_n\mathbb{D})$ by replacing each element $e_{ij}(\lambda)$ of $\mathbb{E}_{r+1}(\lambda)$ by the $M\times M$ matrix $e_{ij}(\tau_n\mathbb{D})\mathbb{I}$, $i,j=1,\dots,r+1$.

We are going to solve (3.5) by extending the Cramer rule for the block matrices. We first introduce some notation. For any matrix $\mathbb{X}=(X_{ij})_{i,j=1}^d$, $d\ge 1$, we denote by $\mathbb{X}_{i,j}$ the matrix obtained by removing the ith row and jth column, $i,j=1,\dots,d$. We also denote by $\mathbb{X}_{(i_1,\cdots,i_k),(j_1,\cdots,j_l)}$ the matrix obtained by removing the $i_1,\dots,i_k$-th rows and the $j_1,\dots,j_l$-th columns, where $1\le i_1<\cdots<i_k\le d, 1\le j_1<\cdots<j_l\le d$. The following property about the adjugate matrix is well known:

$$(\det\mathbb{X})\delta_{i,j}=\sum_{k=1}^{d}(-1)^{i+k}(\det\mathbb{X}_{k,i})X_{kj}. \tag{3.8}$$

Denote $\mathbb{H}=\mathbb{E}_{r+1}(\lambda)_{(r-1,r,r+1),(r-1,r,r+1)}\in\mathbb{R}^{(r-2)\times(r-2)}$. Then the matrix $\mathbb{E}_{r+1}(\lambda)$ can be partitioned as

$$\mathbb{E}_{r+1}(\lambda)=\left(\begin{array}{ccc|ccc} & \mathbb{H} & & b_{r-3} & 0 & 0\\ & & & -1 & b_{r-2} & 0\\ \hline & & & a_{r-1} & -1 & 0\\ & & & 0 & a_r & -1\\ c_1 & \cdots & c_{r-2} & c_{r-1} & c_r & c_{r+1}\end{array}\right). \tag{3.9}$$

Similarly, we have

$$\mathbb{E}_{r}(\lambda)=\left(\begin{array}{ccc|cc} & \mathbb{H} & & b_{r-3} & 0\\ & & & -1 & 0\\ \hline & & & a_{r-1} & -1\\ c_1 & \cdots & c_{r-2} & c_{r-1} & c_r\end{array}\right),\quad \mathbb{E}_{r-1}(\lambda)=\left(\begin{array}{ccc|c} & \mathbb{H} & & 0\\ & & & -1\\ \hline c_1 & \cdots & c_{r-2} & c_{r-1}\end{array}\right). \tag{3.10}$$

The following simple identities will play an important role in our analysis:

$$\mathbb{E}_{r+1}(\lambda)_{r,(r,r+1)} = \mathbb{E}_r(\lambda)_{*,r}, \tag{3.11}$$

where for any $\mathbb{X} \in \mathbb{R}^{d\times d}$, we denote by $\mathbb{X}_{*,j} \in \mathbb{R}^{d\times(d-1)}$ the matrix by removing the jth column of $\mathbb{X}$. Similarly, we denote by $\mathbb{X}_{i,*} \in \mathbb{R}^{(d-1)\times d}$ the matrix by removing the ith row of $\mathbb{X}$.

The following elementary lemma is useful in our analysis.

LEMMA 3.1. *For any $r \geq 3$, we have*

$$\begin{aligned} \det[\mathbb{E}_{r+1}(\lambda)_{(r-2,r,r+1),(j,r,r+1)}] &= -a_{r-1}\det[\mathbb{E}_{r-1}(\lambda)_{r-1,j}] \\ &= a_{r-1}c_{r-1}\det[\mathbb{E}_{r-1}(\lambda)_{r-2,j}], \quad 1 \leq j \leq r-2, \end{aligned} \tag{3.12}$$

$$\begin{aligned} \det[\mathbb{E}_{r+1}(\lambda)_{(r-1,r,r+1),(j,r,r+1)}] &= -\det[\mathbb{E}_r(\lambda)_{r,j}] \\ &= c_r\det[\mathbb{E}_r(\lambda)_{r-1,j}], \quad 1 \leq j \leq r-1. \end{aligned} \tag{3.13}$$

Proof. We only prove (3.12). The identity (3.13) can be proved similarly. By (3.11),

$$\begin{aligned} \det[\mathbb{E}_{r+1}(\lambda)_{(r-2,r,r+1),(j,r,r+1)}] &= \det[\mathbb{E}_r(\lambda)_{(r-2,r),(j,r)}] \\ &= \det\left(\begin{array}{c|c} \mathbb{H}_{r-2,j} & \begin{array}{c} \\ b_{r-3} \end{array} \\ \hline & a_{r-1} \end{array}\right) \\ &= a_{r-1}\det[\mathbb{H}_{r-2,j}]. \end{aligned}$$

On the other hand, it is easy to see from (3.10) that

$$\det[\mathbb{E}_{r-1}(\lambda)_{r-1,j}] = -\det\,\mathbb{H}_{r-2,j}, \ \det[\mathbb{E}_{r-1}(\lambda)_{r-2,j}] = c_{r-1}\det[\mathbb{H}_{r-2,j}].$$

This completes the proof. $\square$

To proceed, we note that

$$\mathbb{E}_2(\lambda) = \begin{pmatrix} a_1 & -1 \\ 1 & -1 \end{pmatrix}, \qquad \mathbb{E}_3(\lambda) = \begin{pmatrix} a_1 & -1 & 0 \\ 0 & a_2 & -1 \\ 1 & -1 & 1 \end{pmatrix}. \tag{3.14}$$

LEMMA 3.2. *Let $\varphi_r(\lambda) = \det\,\mathbb{E}_{r+1}(\lambda)$ be the determinant of $\mathbb{E}_{r+1}(\lambda)$. Then $\varphi_1(\lambda) = 1 - a_1$, $\varphi_2(\lambda) = 1 - a_1 + a_1a_2$, and*

$$\varphi_r(\lambda) = \varphi_{r-1}(\lambda) + a_ra_{r-1}\varphi_{r-2}(\lambda), \quad r \geq 3. \tag{3.15}$$

Moreover, $\varphi_r(\lambda) = P_r(-\lambda)$, where $P_r(\lambda)$ is the numerator of the $[r/r]$ Padé approximation of e^λ.

Proof. The determinants of $\mathbb{E}_{r+1}(\lambda)$ for $r = 1,2$ follow easily from (3.14). Since by (3.7), $a_r = -b_{r-2}$, we obtain by adding the rth row to the $(r-2)$th row of $\mathbb{E}_{r+1}(\lambda)$ in (3.9) and then expanding the determinant by the rth row that

$$
\begin{aligned}
\varphi_r(\lambda) &= \det \left(\begin{array}{cccc|ccc} & \mathbb{H} & & & b_{r-3} & 0 & 0 \\ & & & & -1 & 0 & -1 \\ \hline & & & & a_{r-1} & -1 & 0 \\ & & & & 0 & a_r & -1 \\ c_1 & \cdots & c_{r-2} & & c_{r-1} & c_r & c_{r+1} \end{array} \right) \\
&= \det \mathbb{E}_r(\lambda) + a_r \det \left(\begin{array}{ccc|cc} & \mathbb{H} & & b_{r-3} & 0 \\ & & & -1 & -1 \\ \hline & & & a_{r-1} & 0 \\ c_1 & \cdots & c_{r-2} & c_{r-1} & c_r \end{array} \right) \\
&= \det \mathbb{E}_r(\lambda) + a_r a_{r-1} \det \mathbb{E}_{r-1}(\lambda),
\end{aligned}
$$

where in the last equality we have expanded the determinant by the $(r-1)$th row. Since $a_1 = \lambda/2$, $a_2 = \lambda/6$, we use Lemma 2.5 to conclude $\varphi_r(\lambda) = P_r(-\lambda)$, where $P_r(\lambda)$ is the numerator of the $[r/r]$ Padé approximation of e^λ. ∎

THEOREM 3.1. *The matrix $\varphi_r(\tau_n \mathbb{D})$ is invertible. The solution of* (3.5) $\mathbf{X} = (\mathbf{a}_0^T, \ldots, \mathbf{a}_r^T)^T$ *satisfies that for $i = 1, \ldots, r+1$,*

$$\mathbf{a}_{i-1} = (-1)^r \delta_{i,r+1} \mathbf{b}_r + \sum_{j=1}^{r} (\zeta_j \mathbb{I} + \tau_n \mathbb{D})^{-1} \left(\sum_{k=1}^{r+1} (-1)^{i+k+1} \frac{\phi_{ki}(-\zeta_j)}{P_r'(\zeta_j)} \mathbf{b}_{k-1} \right),$$

where $\phi_{ki}(\lambda) = \det[\mathbb{E}_{r+1}(\lambda)_{k,i}]$, $k, i = 1, \ldots, r+1$, are the minors of $\mathbb{E}_{r+1}(\lambda)$.

Proof. By Lemma 3.2, $\varphi_r(\lambda) = P_r(-\lambda) = (-1)^r \frac{r!}{(2r)!}(\lambda + \zeta_1) \cdots (\lambda + \zeta_r)$, where $\zeta_1, \ldots, \zeta_r \in \mathbb{C}$ are zeros of the diagonal Padé numerator of type $[r/r]$ for e^λ. By Lemma 2.6, $\mathrm{Re}(\zeta_k) \le -2$, $k = 1, \ldots, r$. On the other hand, (1.2) implies that the eigenvalues of $\mathbb{D}$ lie in the left half-plane. Thus the eigenvalues of $\zeta_k \mathbb{I} + \tau_n \mathbb{D}$, $1 \le k \le r$, lie in the half-plane $\{z \in \mathbb{C} : \mathrm{Re}(z) \le -2\}$. This shows $\varphi_r(\tau_n \mathbb{D})$ is invertible.

Now by (3.8) we have

$$[\det\, \mathbb{E}_{r+1}(\lambda)]\delta_{i,j} = \sum_{k=1}^{r+1} (-1)^{i+k} [\det\, \mathbb{E}_{r+1}(\lambda)_{k,i}] e_{kj}(\lambda),$$

where $e_{kj}(\lambda)$ is the (k, j) element of $\mathbb{E}_{r+1}(\lambda)$. By replacing λ by $\tau_n \mathbb{D}$ in the above equality, we have

$$\varphi_r(\tau_n \mathbb{D})\delta_{i,j} = \sum_{k=1}^{r+1} (-1)^{i+k} \phi_{ki}(\tau_n \mathbb{D}) e_{kj}(\tau_n \mathbb{D}). \tag{3.16}$$

From (3.5) we have

$$\sum_{j=1}^{r+1} e_{ij}(\tau_n \mathbb{D}) \mathbf{a}_{j-1} = \mathbf{b}_{i-1}, \quad i = 1, \ldots, r+1.$$

Thus multiplying (3.16) by $\mathbf{a}_{j-1}$ and summing over j from 1 to $r+1$, we obtain

$$
\begin{aligned}
\varphi_r(\tau_n \mathbb{D}) \mathbf{a}_{i-1} &= \sum_{k=1}^{r+1} (-1)^{i+k} \phi_{ki}(\tau_n \mathbb{D}) \cdot \sum_{j=1}^{r+1} e_{kj}(\tau_n \mathbb{D}) \mathbf{a}_{j-1} \\
&= \sum_{k=1}^{r+1} (-1)^{i+k} \phi_{ki}(\tau_n \mathbb{D}) \mathbf{b}_{k-1}. \qquad (3.17)
\end{aligned}
$$

Note that for $(k,i)\neq(r+1,r+1)$, $\phi_{ki}(\lambda)\in P^m$, $m\le r-1$, by (2.19) we have

$$\frac{\phi_{ki}(\tau_n\mathbb{D})}{\varphi_r(\tau_n\mathbb{D})}=\sum_{j=1}^{r}-\frac{\phi_{ki}(-\zeta_j)}{P_r'(\zeta_j)}(\zeta_j\mathbb{I}+\tau_n\mathbb{D})^{-1}.$$

For $(k,i)=(r+1,r+1)$, we have from (3.6) that $\det[E_{r+1}(\lambda)_{r+1,r+1}]=\prod_{j=1}^r a_j=\frac{r!}{(2r)!}\lambda^r$. Thus $\phi_{r+1,r+1}(\lambda)-(-1)^r\varphi_r(\lambda)\in P^{r-1}$, and by using (2.19) again we obtain

$$\frac{\phi_{r+1,r+1}(\tau_n\mathbb{D})}{\varphi_r(\tau_n\mathbb{D})}=(-1)^r\mathbb{I}+\sum_{j=1}^{r}-\frac{\phi_{r+1,r+1}(-\zeta_j)}{P_r'(\zeta_j)}(\zeta_j\mathbb{I}+\tau_n\mathbb{D})^{-1}.$$

This completes the proof of the lemma. ∎

We remark that (3.17) can also be proved by using an abstract result in Brown [4, Theorem 2.19 and Corollary 2.21], which studies linear algebra when matrix elements are defined over a space of commuting matrices.

From this theorem we know that the discrete problem (2.2) can be solved by solving $r(r+1)$ linear systems of equations of order $M\times M$ in parallel once all minors of $\mathbb{E}_{r+1}(\lambda)$ at $\lambda=-\zeta_j, j=1,\dots,r$, are known. In the following, we will find recursive formulas to computing these minors.

Let $\mathbb{G}_r(\lambda)=\mathbb{E}_{r+1}(\lambda)_{r,r+1}$, $r\ge1$. The determinants of $\mathbb{G}_r(\lambda)$ for $r=1,2$ can be calculated by (3.14). We have the following lemma for det $\mathbb{G}_r(\lambda)$ for $r\ge3$.

Lemma 3.3. *For $r\ge3$, we have*

$$\det\,\mathbb{G}_r(\lambda)=\det\,\mathbb{G}_{r-1}(\lambda)-a_{r-1}b_{r-2}\det\,\mathbb{G}_{r-2}(\lambda)+c_r\prod_{k=1}^{r-1}a_k.$$

Proof. By the definition and the partition in (3.9), we know that

$$\mathbb{G}_r(\lambda)=\left(\begin{array}{ccc|cc} & \mathbb{H} & & b_{r-3} & 0\\ & & & -1 & b_{r-2}\\ \hline & & & a_{r-1} & -1\\ c_1 & \cdots & c_{r-2} & c_{r-1} & c_r\end{array}\right).$$

By expanding the determinant by the $(r-1)$th row and using (3.11), we obtain

$$\begin{aligned}\det\mathbb{G}_r(\lambda)&=\det[\mathbb{E}_{r+1}(\lambda)_{(r-1,r),(r,r+1)}]+a_{r-1}\det\left(\begin{array}{ccc|c} & \mathbb{H} & & \\ & & & b_{r-2}\\ \hline c_1 & \cdots & c_{r-2} & c_r\end{array}\right)\\ &=\det[\mathbb{E}_r(\lambda)_{r-1,r}]+a_{r-1}(c_r\det\mathbb{H}-b_{r-2}\det[\mathbb{E}_{r+1}(\lambda)_{(r-2,r-1,r),(r-1,r,r+1)}])\\ &=\det\mathbb{G}_{r-1}(\lambda)+c_r\prod_{k=1}^{r-1}a_k-a_{r-1}b_{r-2}\det[\mathbb{E}_r(\lambda)_{(r-2,r-1),(r-1,r)}]\\ &=\det\mathbb{G}_{r-1}(\lambda)+c_r\prod_{k=1}^{r-1}a_k-a_{r-1}b_{r-2}\det[\mathbb{E}_{r-1}(\lambda)_{r-2,r-1}]\\ &=\det\mathbb{G}_{r-1}(\lambda)-a_{r-1}b_{r-2}\det\mathbb{G}_{r-2}(\lambda)+c_r\prod_{k=1}^{r-1}a_k,\end{aligned}$$

where we have used the fact that $\det \mathbb{H} = \Pi_{k=1}^{r-2} a_k$ and expanded the determinant by the last column in the second equality. This completes the proof. $\square$

The minors of $\mathbb{G}_r(\lambda)$ for $r = 1, 2$ can be computed directly by (3.14). The following lemma gives the recursive formulas for some of the minors of $\mathbb{G}_r(\lambda)$, which will be used to compute the minors of $\mathbb{E}_{r+1}(\lambda)$.

LEMMA 3.4. *For $r \geq 3$, we have*

$$\begin{aligned}
\det[\mathbb{G}_r(\lambda)_{i,r}] &= (-1)^{r-i-1} \frac{a_1 \cdots a_{r-1}}{a_1 \cdots a_i} \det \mathbb{G}_i(\lambda), \quad 1 \leq i \leq r-1, \\
\det[\mathbb{G}_r(\lambda)_{r-1,j}] &= \det[\mathbb{E}_r(\lambda)_{r-1,j}] - b_{r-2} \det[\mathbb{G}_{r-1}(\lambda)_{r-2,j}], \quad 1 \leq j \leq r-1.
\end{aligned}$$

Proof. For $i = 1, \ldots, r-2$, by definition, we have

$$\begin{aligned}
\det[\mathbb{G}_r(\lambda)_{i,r}] &= \det \begin{pmatrix}
a_1 & -1 & b_1 & & & & & & \\
 & \ddots & \ddots & \ddots & & & & & \\
 & & a_{i-1} & -1 & b_{i-1} & & & & \\
 & & & 0 & a_{i+1} & -1 & b_{i+1} & & \\
 & & & & \ddots & \ddots & \ddots & \ddots & \\
 & & & & & 0 & a_{r-3} & -1 & b_{r-3} \\
 & & & & & & 0 & a_{r-2} & -1 \\
 & & & & & & & 0 & a_{r-1} \\
c_1 & \cdots & \cdots & \cdots & \cdots & \cdots & c_{r-3} & c_{r-2} & c_{r-1}
\end{pmatrix} \\
&= (-a_{r-1}) \cdots (-a_{i+1}) \det \begin{pmatrix}
a_1 & -1 & b_1 & & \\
 & \ddots & \ddots & \ddots & \\
 & & a_{i-2} & -1 & b_{i-2} \\
 & & & a_{i-1} & -1 \\
c_1 & \cdots & \cdots & c_{i-1} & c_i
\end{pmatrix} \\
&= (-1)^{r-i-1} \left(\prod_{k=i+1}^{r-1} a_k \right) \det \mathbb{G}_i(\lambda).
\end{aligned}$$

Finally, for $\mathbb{G}(\lambda)_{i,r}$, $i = r-1$, we have by the definition and using the first identity in (3.11)

$$\mathbb{G}_r(\lambda)_{r-1,r} = \mathbb{E}_{r+1}(\lambda)_{(r-1,r),(r,r+1)} = \mathbb{E}_r(\lambda)_{r-1,r} = \mathbb{G}_{r-1}(\lambda).$$

This shows the first equality of the lemma. To show the second equality, for any $1 \leq j \leq r-1$, we have by using the partition (3.9) that

$$\mathbb{G}_r(\lambda)_{r-1,j} = \left(\begin{array}{ccccccc|cc}
 & & & \mathbb{H}_{*,j} & & & & b_{r-3} & 0 \\
 & & & & & & & -1 & b_{r-2} \\
\hline
c_1 & \cdots & c_{j-1} & c_{j+1} & \cdots & c_{r-2} & & c_{r-1} & c_r
\end{array} \right).$$

Thus by expanding the determinant by the last column, we have by using (3.11), (3.13), and the definition of $\mathbb{G}_{r-1}(\lambda)$ that

$$
\begin{aligned}
&\det[\mathbb{G}_r(\lambda)_{r-1,j}]\\
&\quad = c_r \det[\mathbb{E}_{r+1}(\lambda)_{(r-1,r,r+1),(j,r,r+1)}] - b_{r-2}\det[\mathbb{E}_{r+1}(\lambda)_{(r-2,r-1,r),(j,r,r+1)}]\\
&\quad = \det[\mathbb{E}_r(\lambda)_{r-1,j}] - b_{r-2}\det[\mathbb{E}_r(\lambda)_{(r-2,r-1),(j,r)}]\\
&\quad = \det[\mathbb{E}_r(\lambda)_{r-1,j}] - b_{r-2}\det[\mathbb{G}_{r-1}(\lambda)_{r-2,j}].
\end{aligned}
$$

This completes the proof. ∎

The minors of $\mathbb{E}_{r+1}(\lambda)$ for $r = 1,2$ can be easily computed from (3.14). The following theorem gives the recursive formulas for computing all minors of $\mathbb{E}_{r+1}(\lambda)$ for $r \geq 3$.

THEOREM 3.2. *Letting $r \geq 3$, we have the following:*
1° *For $i = 1,\dots,r-2$,*

$$
\begin{aligned}
&\det[\mathbb{E}_{r+1}(\lambda)_{i,j}] = \det[\mathbb{E}_r(\lambda)_{i,j}] + a_r a_{r-1}\det[\mathbb{E}_{r-1}(\lambda)_{i,j}], \quad 1 \leq j \leq r-2,\\
&\det[\mathbb{E}_{r+1}(\lambda)_{i,j}] = (-1)^{j-i-1}\frac{a_1\cdots a_{j-1}}{a_1\cdots a_i}\det\,\mathbb{G}_i(\lambda), \quad j = r-1, r, r+1.
\end{aligned}
$$

2° *For $i = r-1$, we have*

$$
\begin{aligned}
&\det[\mathbb{E}_{r+1}(\lambda)_{r-1,j}] = -c_{r+1}a_r\det[\mathbb{E}_r(\lambda)_{r,j}] + \det[\mathbb{G}_r(\lambda)_{r-1,j}], \quad 1 \leq j \leq r-1,\\
&\det[\mathbb{E}_{r+1}(\lambda)_{r-1,j}] = (-1)^{j-i-1}\frac{a_1\cdots a_{j-1}}{a_1\cdots a_i}\det\,\mathbb{G}_i(\lambda), \quad j = r, r+1.
\end{aligned}
$$

3° *For $i = r$, we have*

$$
\begin{aligned}
&\det[\mathbb{E}_{r+1}(\lambda)_{r,j}] = \det[\mathbb{E}_r(\lambda)_{r-1,j}] - a_{r-1}b_{r-2}\det[\mathbb{E}_{r-1}(\lambda)_{r-2,j}], \quad 1 \leq j \leq r-2,\\
&\det[\mathbb{E}_{r+1}(\lambda)_{r,j}] = c_{r+1}(-1)^{i-j}\prod_{k=1}^{j-1} a_k, \quad j = r-1, r,\\
&\det[\mathbb{E}_{r+1}(\lambda)_{r,r+1}] = \det\,\mathbb{G}_{r-1}(\lambda) - a_{r-1}b_{r-2}\det\,\mathbb{G}_{r-2}(\lambda) + c_r\prod_{k=1}^{r-1} a_k.
\end{aligned}
$$

4° *For $i = r+1$, we have*

$$
\begin{aligned}
&\det[\mathbb{E}_{r+1}(\lambda)_{r+1,j}] = -\det[\mathbb{E}_r(\lambda)_{r,j}] - a_{r-1}b_{r-2}\det[\mathbb{E}_{r-1}(\lambda)_{r-1,j}], \quad 1 \leq j \leq r-2,\\
&\det[\mathbb{E}_{r+1}(\lambda)_{r+1,j}] = (-1)^{i-j}\prod_{k=1}^{j-1} a_k, \quad j = r-1, r, r+1.
\end{aligned}
$$

Proof. The proof is divided into 4 steps.

STEP 1. The first equality in 1° can be proved by the same argument as that in Lemma 3.2. Here we omit the details. We only prove the second equality in 1° when $j = r+1$. The other cases can be proved similarly. By the partition in (3.9), we know that for $1 \leq i \leq r-2$,

$$
\mathbb{E}_{r+1}(\lambda)_{i,r+1} = \left(\begin{array}{ccc|cc}
 & \mathbb{H}_{i,*} & & \mathbb{F}_{i,*} & \\
\hline
 & & & a_{r-1} & -1\\
 & & & 0 & a_r\\
c_1 & \cdots & c_{r-2} & c_{r-1} & c_r
\end{array}\right), \qquad
\mathbb{F} = \begin{pmatrix}
0 & 0 & 0\\
\vdots & \vdots & \vdots\\
0 & 0 & 0\\
b_{r-3} & 0 & 0\\
-1 & b_{r-2} & 0
\end{pmatrix},
$$

where $\mathbb{F}\in\mathbb{R}^{(r-2)\times 3}$. By expanding the determinant first by the rth and then by the $(r-1)$th rows, we know by (3.11) that

$$\begin{aligned}\det[\mathbb{E}_{r+1}(\lambda)_{i,r+1}] &= a_r a_{r-1}\det[\mathbb{E}_{r+1}(\lambda)_{(i,r-1,r),(r-1,r,r+1)}]\\ &= a_r a_{r-1}\det[\mathbb{E}_r(\lambda)_{(i,r-1),(r-1,r)}]\\ &= a_r a_{r-1}\det[\mathbb{G}_{r-1}(\lambda)_{i,r-1}].\end{aligned}$$

This shows the second equality in 1° by the first identity in Lemma 3.4.

STEP 2. We only prove the first equality in 2° when $1\le j\le r-2$. The other cases can be proved similarly. By the partition in (3.9), we have, for $1\le j\le r-2$,

$$\mathbb{E}_{r+1}(\lambda)_{r-1,j}=\left(\begin{array}{ccc|ccc} & \mathbb{H}_{*,j} & & b_{r-3} & 0 & 0\\ & & & -1 & b_{r-2} & 0\\ \hline & & & 0 & a_r & -1\\ c_1 & \cdots & c_{r-2} & c_{r-1} & c_r & c_{r+1}\end{array}\right).$$

Expanding the determinant by the last column, we obtain

$$\begin{aligned}\det[\mathbb{E}_{r+1}(\lambda)_{r-1,j}] &= c_{r+1}a_r\det[\mathbb{E}_{r+1}(\lambda)_{(r-1,r,r+1),(j,r,r+1)}]+\det[\mathbb{E}_{r+1}(\lambda)_{(r-1,r),(j,r+1)}]\\ &= -c_{r+1}a_r\det[\mathbb{E}_r(\lambda)_{r,j}]+\det[\mathbb{G}_r(\lambda)_{r-1,j}].\end{aligned}$$

This shows the first equality in 2° for $1\le j\le r-2$.

STEP 3. The second equality in 3° can be easily proved. The last equality is shown in Lemma 3.3 since $\mathbb{E}_{r+1}(\lambda)_{r,r+1}=\mathbb{G}_r(\lambda)$ by definition. To show the first equality in 3°, we again use the partition in (3.9) to obtain, for $1\le j\le r-2$,

$$\mathbb{E}_{r+1}(\lambda)_{r,j}=\left(\begin{array}{cccccc|ccc} & & \mathbb{H}_{*,j} & & & & b_{r-3} & 0 & 0\\ & & & & & & -1 & b_{r-2} & 0\\ \hline & & & & & & a_{r-1} & -1 & 0\\ c_1 & \cdots & c_{j-1} & c_{j+1} & \cdots & c_{r-2} & c_{r-1} & c_r & c_{r+1}\end{array}\right).$$

By expanding the determinant successively by the last columns, we have

$$\begin{aligned}&\det[\mathbb{E}_{r+1}(\lambda)_{r,j}]\\ &\quad=-c_{r+1}\det[\mathbb{E}_{r+1}(\lambda)_{(r-1,r,r+1),(j,r,r+1)}]-c_{r+1}b_{r-2}\det[\mathbb{E}_{r+1}(\lambda)_{(r-2,r,r+1),(j,r,r+1)}].\end{aligned}$$

This shows the first equality in 3° by Lemma 3.1.

STEP 4. The first equality in 4° can be proved by the same argument as that in Step 3. The second equality can be easily proved. Here we omit the details. □

This theorem indicates that all minors of $\mathbb{E}_{r+1}(\lambda)$ can be computed once one knows all minors of $\mathbb{E}_m(\lambda)$, $1\le m\le r$, and the minors $\det[\mathbb{G}_r(\lambda)_{r-1,j}]$, $1\le j\le r-1$, which can be computed by Lemma 3.4 recursively based on the information of the minors $E_m(\lambda)$, $1\le m\le r$.

The following lemma indicates that the nodal values of the solution to (2.2) depends only on the coefficient $\mathbf{a}_0$.

LEMMA 3.5. *Let $\mathbf{Y}_r(t)$, $r\ge 1$, be the solution of problem* (2.2). *Then*

$$\mathbf{Y}_r(t_{n+1})=\mathbf{Y}_r(t_n)+\tau_n\mathbb{D}\mathbf{a}_0+\tau_n\mathbf{R}_0,\quad n=0,\dots,N-1.$$

Proof. We integrate (2.2) over I_n and use the orthogonality of Legendre polynomials to obtain

$$\mathbf{Y}_r(t_{n+1}) = \mathbf{Y}_r(t_n) + \int_{I_n} \mathbb{D}\left(\sum_{j=0}^{r-1} \mathbf{a}_j \widetilde{L}_j(t)\right) dt + \int_{I_n} \sum_{j=0}^{r-1} \mathbf{R}_j \widetilde{L}_j(t)\, dt,$$
$$= \mathbf{Y}_r(t_n) + \tau_n \mathbb{D}\mathbf{a}_0 + \tau_n \mathbf{R}_0.$$

This completes the proof. ∎

By Theorem 3.1, we then have

$$\mathbf{Y}_r(t_{n+1}) = \mathbf{Y}_r(t_n) + \sum_{j=1}^{r} (\zeta_j \mathbb{I} + \tau_n \mathbb{D})^{-1} \left(\sum_{k=1}^{r+1} (-1)^k \frac{\phi_{k1}(-\zeta_j)}{P_r'(\zeta_j)} (\tau_n \mathbb{D})\, \mathbf{b}_{k-1}\right) + \tau_n \mathbf{R}_0.$$

This leads to the following parallel algorithm to compute the nodal values of the solution $\mathbf{Y}_r$ to the problem (2.2).

ALGORITHM 3.1. *Given $\mathbf{Y}_r(t_0) = \mathbf{Y}_0$. For $n = 0, \ldots, N-1$, do the following:*
1° *Compute $\mathbf{v}_j \in \mathbb{R}^M$, $j = 1, \ldots, r$, in parallel, where*

$$\mathbf{v}_j = \sum_{k=1}^{r+1} (-1)^k \frac{\phi_{k1}(-\zeta_j)}{P_r'(\zeta_j)} (\tau_n \mathbb{D})\, \mathbf{b}_{k-1}.$$

2° *Solve $(\tau_n \mathbb{D} + \zeta_j \mathbb{I})\mathbf{w}_j = \mathbf{v}_j$, $j = 1, \ldots, r$, in parallel.*
3° *Compute*

$$\mathbf{Y}_r(t_{n+1}) = \mathbf{Y}_r(t_n) + \sum_{j=0}^{r} \mathbf{w}_j + \tau_n \mathbf{R}_0.$$

The following algorithm computes the solution of problem (2.2) inside each time interval.

ALGORITHM 3.2. *Given $\mathbf{Y}_r(t_0) = \mathbf{Y}_0$. For $n = 0, \ldots, N-1$, compute the coefficients $\mathbf{a}_0, \ldots, \mathbf{a}_r$ of $\mathbf{Y}_r$ in each time interval I_n as follows:*
(i) *Compute $\mathbf{v}_{ij} \in \mathbb{R}^M$, $i = 1, \ldots, r+1$, $j = 1, \ldots r$, in parallel, where*

$$\mathbf{v}_{ij} = \sum_{k=1}^{r+1} (-1)^{i+k+1} \frac{\phi_{ki}(-\zeta_j)}{P_r'(\zeta_j)}\, \mathbf{b}_{k-1}.$$

(ii) *Solve $(\tau_n \mathbb{D} + \zeta_j \mathbb{I})\mathbf{w}_{ij} = \mathbf{v}_{ij}$, $i = 1, \ldots, r+1, j = 1, \ldots, r$, in parallel.*
(iii) *Compute $\mathbf{a}_{i-1} = (-1)^r \delta_{i,r+1} \mathbf{b}_r + \sum_{j=1}^{r} \mathbf{w}_{ij}$, $i = 1, \ldots, r+1$, in parallel.*

Remark 3.1. In [12], it is observed that the zeros of $P_r(z)$ come in complex conjugate pairs if they are complex. If $\zeta_{j'} = \bar{\zeta}_j$, $j, j' = 1, \ldots, r$, then $\mathbf{v}_j = \bar{\mathbf{v}}_{j'}$, and

$$\mathbf{w}_j + \mathbf{w}_{j'} = \frac{\mathbf{v}_j}{\tau_n \mathbb{D} + \zeta_j \mathbb{I}} + \frac{\bar{\mathbf{v}}_j}{\tau_n \mathbb{D} + \bar{\zeta}_j \mathbb{I}} = 2\,\mathrm{Re}\left[\frac{\mathbf{v}_j}{\tau_n \mathbb{D} + \zeta_j \mathbb{I}}\right].$$

Thus one need only solve k complex matrix problems instead of $2k$ in Algorithm 3.1 (2°) and $k(r+1)$ complex matrix problems instead of $2k(r+1)$ in Algorithm 3.2 (2°), where $2k$, $0 \le k \le r/2$, are the number of complex zeros of $P_r(z)$.

Remark 3.2. If $\zeta = a + \mathbf{i}b$ is a complex zero of $P_r(z)$, then $a \le -2$ by Lemma 2.5. By Remark 3.1, without loss of generality, we can choose one of the zeros such that

$b < 0$. Let $\mathbf{w} = \mathbf{w}_1 + \mathbf{i}\mathbf{w}_2$, $\mathbf{v} = \mathbf{v}_1 + \mathbf{i}\mathbf{v}_2$, where $\mathbf{v}_i, \mathbf{w}_i \in \mathbb{R}^M$, $i = 1, 2$, satisfy $(\tau_n \mathbb{D} + \zeta \mathbb{I})\mathbf{w} = \mathbf{v}$. Then

$$\widetilde{\mathbb{D}} \begin{pmatrix} \mathbf{w}_1 \\ \mathbf{w}_2 \end{pmatrix} := \begin{pmatrix} \tau_n \mathbb{D} + a\mathbb{I} & -b\,\mathbb{I} \\ -b\,\mathbb{I} & -(\tau_n \mathbb{D} + a\mathbb{I}) \end{pmatrix} \begin{pmatrix} \mathbf{w}_1 \\ \mathbf{w}_2 \end{pmatrix} = \begin{pmatrix} \mathbf{v}_1 \\ -\mathbf{v}_2 \end{pmatrix}.$$

Let $\mathbb{F} = \operatorname{diag}(\tau_n \mathbb{D} + (a+b)\,\mathbb{I}, \tau_n \mathbb{D} + (a+b)\,\mathbb{I}) \in \mathbb{R}^{2M \times 2M}$ be the diagonal matrix. It is shown in Chen et al. [5, Lemma 4.1] that the condition number $\kappa(\mathbb{F}^{-1}\widetilde{\mathbb{D}}) \leq \sqrt{2}$. Therefore, the complex system $(\tau_n \mathbb{D} + \zeta \mathbb{I})\mathbf{w} = \mathbf{v}$ can be efficiently solved if one has the efficient solver for the real matrix $\tau_n \mathbb{D} + (a+b)\,\mathbb{I}$, where $a + b \leq -2$. Notice that the eigenvalues of $\mathbb{D}$ lie in the left half-plane due to the assumption $\mathbb{D} + \mathbb{D}^T \leq 0$.

Remark 3.3. The numerator of the $[r/r]$ Padé approximation $P_r(z)$ has $2k$ complex zeros and 1 real root if $r = 2k+1, k \geq 1$, and has $2k$ complex zeros if $r = 2k$, $k \geq 1$. Denote by $C(2M)$ the costs of solving the matrix problem $\tau_n \mathbb{D} + \zeta_j \mathbb{I}$ with ζ_j being complex, and denote by $C(M)$ the costs of solving the matrix problem $\tau_n \mathbb{D} + \zeta_j \mathbb{I}$ with ζ_j real, where ζ_j, $j = 1, \ldots, r$, are zeros of $P_r(z)$. Let $A(M)$ and $B(M)$ stand for the costs of a vector-vector addition operation and a matrix-vector product operation, respectively. Then the wall time of Algorithm 3.1 in the serial mode is proportional to T_s, where

$$T_s = N \left[\overbrace{r((r+1)B(M) + rA(M))}^{\text{Step } 1^\circ} + \overbrace{kC(2M) + (r-2k)C(M)}^{\text{Step } 2^\circ} + \overbrace{(r+2)A(M)}^{\text{Step } 3^\circ} \right].$$

On the other hand, the wall time of Algorithm 3.1 in the parallel mode is proportional to T_p when using $r - k$ cores, where

$$T_p = N \left[\overbrace{\frac{r}{r-k}((r+1)B(M) + rA(M))}^{\text{Step } 1^\circ} + \overbrace{C(2M)}^{\text{Step } 2^\circ} + \overbrace{(r+2)A(M)}^{\text{Step } 3^\circ} \right] + T_{\text{com}}.$$

Here T_{com} stands for the parallel communication time. Thus the theoretical parallel speedup ratio of Algorithm 3.1 is $S_p = T_s/T_p = O(k)$ if the computational time of the matrix problem is dominant.

For Algorithm 3.2, the wall time in the serial mode is proportional to

$$T_s^{\text{total}} = N \left[\overbrace{r^2(r+1)A(M)}^{\text{Step (i)}} + \overbrace{(r+1)(kC(2M) + (r-2k)C(M))}^{\text{Step (ii)}} + \overbrace{r(r+1)A(M)}^{\text{Step (iii)}} \right].$$

The wall time of Algorithm 3.2 in the parallel mode when using $(r+1)(r-k)$ cores is proportional to

$$T_p^{\text{total}} = N \left[\overbrace{\frac{r^2}{r-k}A(M)}^{\text{Step (i)}} + \overbrace{C(2M)}^{\text{Step (ii)}} + \overbrace{rA(M)}^{\text{Step (iii)}} \right] + T_{\text{com}}.$$

Thus the theoretical parallel speedup ratio of Algorithm 3.2 is $S_p^{\text{total}} = T_s^{\text{total}}/T_p^{\text{total}} = O((r+1)k)$ when the computational time of the matrix problem is dominant.

Remark 3.4. For nonstandard ODE systems of the form

$$\mathbb{M}\mathbf{Y}' = \mathbb{D}\mathbf{Y} + \mathbf{R} \quad \text{in } (0,T), \quad \mathbf{Y}(0) = \mathbf{Y}_0, \tag{3.18}$$

one can use the transformations $\widetilde{\mathbf{Y}} = \mathbb{M}^{\frac{1}{2}}\mathbf{Y}$, $\widetilde{\mathbb{D}} = \mathbb{M}^{-\frac{1}{2}}\mathbb{D}\mathbb{M}^{-\frac{1}{2}}$, $\widetilde{\mathbf{R}} = \mathbb{M}^{-\frac{1}{2}}\mathbf{R}$ to transform problem (3.18) into (1.1) and use the above algorithms to solve the transformed problem. This leads to the following algorithm, which is similar to Algorithm 3.1, to find the nodal values of the solution of the continuous time Galerkin method for solving (3.18).

ALGORITHM 3.3. *Given* $\mathbf{Y}_r(t_0) = \mathbf{Y}_0$. *For* $n = 0, \dots, N-1$, *do the following:*
1° *Compute* $\mathbf{v}_j \in \mathbb{R}^M$, $j = 1, \dots, r$, *in parallel, where*

$$\mathbf{v}_j = \sum_{k=1}^{r+1} (-1)^k \frac{\phi_{k1}(-\zeta_j)}{P_r'(\zeta_j)} (\tau_n \mathbb{D}) \mathbb{M}^{-1}\, \mathbf{b}_{k-1}.$$

2° *Solve* $(\tau_n \mathbb{D} + \zeta_j \mathbb{M})\mathbf{w}_j = \mathbf{v}_j$, $j = 1, \dots, r$, *in parallel.*
3° *Compute*

$$\mathbf{Y}_r(t_{n+1}) = \mathbf{Y}_r(t_n) + \sum_{j=1}^{r} \mathbf{w}_j + \tau_n \mathbb{M}^{-1}\mathbf{R}_0.$$

An algorithm similar to Algorithm 3.2 can also be formulated.

To conclude this section, we prove the following theorem for finding the nodal values of the solution (2.2) that extends (1.3) for solving the ODE system (1.1) when $\mathbf{R} = \mathbf{0}$.

THEOREM 3.3. *Let* $\mathbf{Y}_r \in \mathbf{V}_\tau^r$, $r \ge 1$, *be the solution of problem* (2.2). *Then for* $n = 0, \dots, N-1$,

$$\mathbf{Y}_r(t_{n+1}) = \frac{P_r(\tau_n \mathbb{D})}{P_r(-\tau_n \mathbb{D})} \mathbf{Y}_r(t_n) + \sum_{k=1}^{r} (-1)^{k+1} \frac{\phi_{k1}(\tau_n \mathbb{D})}{P_r(-\tau_n \mathbb{D})}\, \mathbf{b}_{k-1} + \tau_n \mathbf{R}_0,$$

where $\phi_{k1}(\lambda) = \det[\mathbb{E}_{r+1}(\lambda)_{k,1}]$.

Proof. By Lemma 3.2 and (3.17) we have

$$P_r(-\tau_n \mathbb{D})\mathbf{a}_0 = \sum_{k=1}^{r+1} (-1)^{k+1} \phi_{k1}(\tau_n \mathbb{D}) \mathbf{b}_{k-1}.$$

Since $\mathbf{b}_r = \mathbf{Y}_r(t_n)$, by Lemma 3.5,

$$\mathbf{Y}_r(t_{n+1}) = \left[\mathbb{I} + (-1)^r \frac{\tau_n \mathbb{D} \phi_{r+1,1}(\tau_n \mathbb{D})}{P_r(-\tau_n \mathbb{D})} \right] \mathbf{Y}_r(t_n) + \sum_{k=1}^{r} (-1)^{k+1} \frac{\phi_{k1}(\tau_n \mathbb{D})}{P_r(-\tau_n \mathbb{D})} \mathbf{b}_{k-1} + \tau_n \mathbf{R}_0.$$

Denote by $\psi_r(\lambda) = (-1)^r \lambda \phi_{r+1,1}(\lambda) = (-1)^r \lambda \det[\mathbb{E}_{r+1}(\lambda)_{r+1,1}]$. By Theorem 3.2 (4°), we know that $\psi_r(\lambda)$ satisfies

$$\psi_r(\lambda) = \psi_{r-1}(\lambda) + \frac{\lambda^2}{4} \frac{1}{(2r-1)(2r-3)} \psi_{r-2}(\lambda), \quad r \ge 3.$$

On the other hand, by (3.14), we have $\psi_1(\lambda) = \lambda, \psi_2(\lambda) = \lambda$. This implies by Lemma 2.5 that $\psi_r(\lambda) = P_r(\lambda) - P_r(-\lambda)$. Thus

$$\mathbb{I} + (-1)^r \frac{\tau_n \mathbb{D} \phi_{r+1,1}(\tau_n \mathbb{D})}{P_r(-\tau_n \mathbb{D})} = \mathbb{I} + \frac{P_r(\tau_n \mathbb{D}) - P_r(-\tau_n \mathbb{D})}{P_r(-\tau_n \mathbb{D})} = \frac{P_r(\tau_n \mathbb{D})}{P_r(-\tau_n \mathbb{D})}.$$

This completes the proof. ∎

4. Optimal stability and error estimates. In this section, we show optimal stability and error estimates of the continuous time Galerkin method (2.2) in terms of r when $\mathbb{D}$ is a symmetric or skew-symmetric matrix. This will be achieved by using the explicit formulas in Theorem 3.1. We start by studying further properties of the minors of the stiffness matrix of the continuous time Galerkin method $\mathbb{A}=\mathbb{E}_{r+1}(\tau_n\mathbb{D})$.

Let $\chi_{r+1,j}(\lambda)=(-1)^{r+1}\det[\mathbb{E}_{r+1}(\lambda)_{r+1,j}]$. Then by (3.14) we have

$$\chi_{2,1}=-1,\chi_{2,2}=a_1,\chi_{3,1}=-1,\chi_{3,2}=a_1,\chi_{3,3}=-a_1a_2. \tag{4.1}$$

For $r\geq 3$, by 4° in Theorem 3.2, we have the following recursive formulas:

$$\chi_{r+1,j}(\lambda)=\chi_{r,j}(\lambda)+a_ra_{r-1}\chi_{r-1,j}(\lambda),\quad 1\leq j\leq r-2, \tag{4.2}$$

$$\chi_{r+1,j}(\lambda)=(-1)^j\prod_{k=1}^{j-1}a_k,\quad j=r-1,r,r+1. \tag{4.3}$$

Let $\varphi_0=1$ and $\varphi_r(\lambda)=\det\ \mathbb{E}_{r+1}(\lambda)$, $r\geq 1$. Then by Lemma 3.2, $\varphi_1=1-a_1$,

$$\varphi_{r+1}=\varphi_r+a_ra_{r+1}\varphi_{r-1},\quad r\geq 1. \tag{4.4}$$

LEMMA 4.1. *For any $r\geq 1$ and $\lambda\in\mathbb{R}$, we have*

$$\sum_{j=1}^{r-1}[\chi_{r,j}(-\lambda)\chi_{r+1,j}(\lambda)+\chi_{r,j}(\lambda)\chi_{r+1,j}(-\lambda)]\frac{1}{2j-1}$$
$$=\varphi_{r-1}(-\lambda)\varphi_r(\lambda)+\varphi_{r-1}(\lambda)\varphi_r(-\lambda)+\frac{(-1)^r2r}{2r-1}(a_1\cdots a_{r-1})^2, \tag{4.5}$$

$$\sum_{j=1}^{r+1}\chi_{r+1,j}(-\lambda)\chi_{r+1,j}(\lambda)\frac{1}{2j-1}=\varphi_r(-\lambda)\varphi_r(\lambda)+\frac{(-1)^{r+1}2r}{2r+1}(a_1\cdots a_r)^2. \tag{4.6}$$

Proof. We denote

$$\begin{aligned}A_r:=&\sum_{j=1}^{r-1}\left[\chi_{r,j}(-\lambda)\chi_{r+1,j}(\lambda)+\chi_{r,j}(\lambda)\chi_{r+1,j}(-\lambda)\right]\frac{1}{2j-1}\\&-\left[\varphi_{r-1}(-\lambda)\varphi_r(\lambda)+\varphi_{r-1}(\lambda)\varphi_r(-\lambda)\right],\\B_{r+1}:=&\sum_{j=1}^{r+1}\chi_{r+1,j}(-\lambda)\chi_{r+1,j}(\lambda)\frac{1}{2j-1}-\varphi_r(-\lambda)\varphi_r(\lambda).\end{aligned}$$

We will argue by induction. First (4.5)–(4.6) are obvious for $r=1,2$ by (4.1). Now we assume (4.5)–(4.6) are valid for all $r\leq n$, $n\geq 2$. Since by (4.3), $\chi_{n+2,n}(\lambda)=\chi_{n+1,n}(\lambda)$, we have by (4.2) and (4.4) that

$$A_{n+1}=2B_{n+1}-2\chi_{n+1,n+1}(-\lambda)\chi_{n+1,n+1}(\lambda)\frac{1}{2n+1}+a_na_{n+1}A_n.$$

Now by (4.3) and the induction assumption that (4.5)–(4.6) are valid for $r=n$, we obtain

$$A_{n+1}=\frac{(-1)^{n+1}2(n+1)}{2n+1}(a_1\cdots a_n)^2.$$

This shows (4.5) for $r=n+1$. Similarly, we can prove by (4.2)–(4.4) that

$$\begin{aligned}B_{n+2}&=B_{n+1}+(a_na_{n+1})^2B_n+a_na_{n+1}A_n\\&\quad+\chi_{n+2,n+2}(-\lambda)\chi_{n+2,n+2}(\lambda)\frac{1}{2n+3}-(a_na_{n+1})^2\chi_{n,n}(-\lambda)\chi_{n,n}(\lambda)\frac{1}{2n-1}.\end{aligned}$$

Now by the induction assumption (4.6) for $r=n,n-1$ and (4.5) for $r=n$, we obtain by using (4.3) that

$$B_{n+2}=(-1)^{n+2}\frac{2(n+1)}{2n+3}(a_1\cdots a_{n+1})^2.$$

This completes the proof. ∎

LEMMA 4.2. *Let $r\ge 1$. For any $\lambda\le 0$, we have*

$$\sum_{j=1}^{r-1}\chi_{r,j}(\lambda)\chi_{r+1,j}(\lambda)\frac{1}{2j-1}\le\varphi_{r-1}(\lambda)\varphi_r(\lambda),\quad\sum_{j=1}^{r+1}\chi_{r+1,j}(\lambda)^2\frac{1}{2j-1}\le\varphi_r(\lambda)^2. \tag{4.7}$$

Proof. For any $r\ge 1$, we denote

$$C_r:=\sum_{j=1}^{r-1}\chi_{r,j}\chi_{r+1,j}\frac{1}{2j-1}-\varphi_{r-1}\varphi_r,\quad D_{r+1}:=\sum_{j=1}^{r+1}\chi_{r+1,j}^2\frac{1}{2j-1}-\varphi_r^2.$$

We again argue by induction. First (4.7) is obvious for $r=1,2$ by (4.1) since $\lambda\le 0$. Now we assume (4.7) is valid for all $r\le n$, $n\ge 2$. By (4.2)–(4.4), it is easy to see that

$$C_{n+1}=-\chi_{n+1,n+1}^2\frac{1}{2n+1}+D_{n+1}+a_na_{n+1}C_n,$$

where we have used $\chi_{n+2,n}(\lambda)=\chi_{n+1,n}(\lambda)$. Thus if $C_n\le 0, D_{n+1}\le 0$, then $C_{n+1}\le 0$.

On the other hand, by (4.2)–(4.4), we have

$$D_{n+2}=D_{n+1}+a_na_{n+1}D_n+2a_na_{n+1}C_n+(a_1\cdots a_{n+1})^2\left(\frac{1}{2n+3}-\frac{1}{2n-1}\right).$$

Thus $D_{n+2}\le 0$ if $D_n\le 0, D_{n+1}\le 0$, and $C_n\le 0$. This completes the proof. ∎

The following theorem is the main result of this section.

THEOREM 4.1. *Let $\mathbb{D}$ be a symmetric or skew-symmetric matrix, and $\mathbf{Y}_r\in\mathbf{V}_\tau^r$ is the solution of problem* (2.2). *Then we have*

$$\|\mathbf{Y}_r\|_{L^2(0,T)}\le T^{1/2}\|\mathbf{Y}_0\|_{\mathbb{R}^M}+CT\|\mathbf{R}\|_{L^2(0,T)}, \tag{4.8}$$

$$\max_{0\le t\le T}\|\mathbf{Y}_r\|_{\mathbb{R}^M}\le Cr(\|\mathbf{Y}_0\|_{\mathbb{R}^M}+T^{1/2}\|\mathbf{R}\|_{L^2(0,T)}), \tag{4.9}$$

where the constant C is independent of $\tau,r,\mathbb{D}$, and $\mathbf{R}$.

Proof. Letting $\hat{\mathbf{Y}}_r\in[P^r]^M$ be defined as in (2.6) of Lemma 2.1, we claim that

$$\|\hat{\mathbf{Y}}_r\|_{L^2(I_n)}\le\tau_n^{1/2}\|\mathbf{Y}_r^n\|_{\mathbb{R}^M}, \tag{4.10}$$

which improves the bound (2.10) in the proof of Lemma 2.1. To show (2.6), by Theorem 3.1, we have

$$\hat{\mathbf{Y}}_r(t)=\sum_{j=1}^{r+1}\mathbf{a}_{j-1}\tilde{L}_{j-1}(t)=\sum_{j=1}^{r+1}(-1)^j\frac{\chi_{r+1,j}(\tau_n\mathbb{D})}{\varphi_r(\tau_n\mathbb{D})}\mathbf{Y}_r^n\tilde{L}_{j-1}(t)\quad\forall t\in I_n.$$

Thus by (2.5),

$$\|\hat{\mathbf{Y}}_r\|^2_{L^2(I_n)} = \sum_{j=1}^{r+1} \|\chi_{r+1,j}(\tau\mathbb{D})\varphi_r(\tau_n\mathbb{D})^{-1}\mathbf{Y}_r^n\|^2_{\mathbb{R}^M}\frac{\tau_n}{2j-1}. \tag{4.11}$$

Denote $\mathbf{Z}_r^n = \varphi_r(\tau_n\mathbb{D})^{-1}\mathbf{Y}_r^n$. If $\mathbb{D}$ is skew-symmetric $\mathbb{D}^T = -\mathbb{D}$, by (2.5), (4.6), we have

$$\begin{aligned}
\|\hat{\mathbf{Y}}_r\|^2_{L^2(I_n)} &= \sum_{j=1}^{r+1} \|\chi_{r+1,j}(\tau_n\mathbb{D})\mathbf{Z}_r^n\|^2_{\mathbb{R}^M}\frac{\tau_n}{2j-1} \\
&= \tau_n\|\varphi_r(\tau_n\mathbb{D})\mathbf{Z}_r^n\|^2_{\mathbb{R}^M} - \tau_n\left\|\sqrt{\frac{2r}{2r+1}}\frac{r!}{(2r)!}(\tau_n\mathbb{D})^r\mathbf{Z}_r^n\right\|^2_{\mathbb{R}^M} \\
&\le \tau_n\|\mathbf{Y}_r^n\|^2_{\mathbb{R}^M}.
\end{aligned}$$

This shows the claim (4.10) when $\mathbb{D}$ is skew-symmetric.

If $\mathbb{D}$ is symmetric, the eigenvalues of $\mathbb{D}$ are nonpositive since $\mathbb{D}+\mathbb{D}^T \le 0$. By Lemma 4.2, it is easy to show that

$$\sum_{j=1}^{r+1} \|\chi_{r+1,j}(\tau_n\mathbb{D})\mathbf{Z}_r^n\|^2_{\mathbb{R}^M}\frac{\tau_n}{2j-1} - \tau_n\|\varphi_r(\tau_n\mathbb{D})\mathbf{Z}_r^n\|^2_{\mathbb{R}^M} \le 0,$$

which yields

$$\sum_{j=1}^{r+1} \|\chi_{r+1,j}(\tau_n\mathbb{D})\varphi_r(\tau_n\mathbb{D})^{-1}\mathbf{Y}_r^n\|^2_{\mathbb{R}^M}\frac{\tau_n}{2j-1} \le \tau_n\|\mathbf{Y}_r^n\|^2_{\mathbb{R}^M}.$$

Now it follows from (4.11) that $\|\hat{\mathbf{Y}}_r\|_{L^2(I_n)} \le \tau_n^{1/2}\|\mathbf{Y}_r^n\|_{\mathbb{R}^M}$. This shows the claim (4.10) when $\mathbb{D}$ is a symmetric matrix.

It follows from (4.10), (2.12), and (2.3) that

$$\begin{aligned}
\|\mathbf{Y}_r\|_{L^2(I_n)} &\le \tau_n^{1/2}\|\mathbf{Y}_r^n\|_{\mathbb{R}^M} + 2\tau_n\|\mathbf{R}\|_{L^2(I_n)} \\
&\le \tau_n^{1/2}\|\mathbf{Y}_0\|_{\mathbb{R}^M} + C\tau_n^{1/2}T^{1/2}\|\mathbf{R}\|_{L^2(0,T)}.
\end{aligned}$$

This implies (4.8) easily. Now by the *hp* inverse estimate,

$$\max_{t_n\le t\le t_{n+1}} \|\mathbf{Y}_r\|_{\mathbb{R}^M} \le C\tau_n^{-1/2}r\|\mathbf{Y}_r\|_{L^2(I_n)} \le Cr(\|\mathbf{Y}_0\|_{\mathbb{R}^M} + T^{1/2}\|\mathbf{R}\|_{L^2(0,T)}).$$

This shows (4.9). This completes the proof. ∎

The following theorem improves the error estimates in Theorem 2.3. It can be proved by the standard argument by using Theorem 4.1 instead of Lemma 2.1. Here we omit the details.

THEOREM 4.2. *Let $\mathbb{D}$ be a symmetric or skew-symmetric matrix. Assuming that $\mathbf{R} \in [H^s(0,T)]^M$, $\mathbf{Y} \in [W^{1+s,\infty}(0,T)]^M$, $s \ge 1$, and $\mathbf{Y}_r \in \mathbf{V}_\tau^r$ is the solution of the problem* (2.2)*, we have*

$$\|\mathbf{Y}-\mathbf{Y}_r\|_{L^2(0,T)} \le C(1+T)\frac{\tau^{\min(r+1,s)}}{r^s}(\|\mathbf{Y}\|_{H^s(0,T)} + \|\mathbb{D}\mathbf{Y}\|_{H^s(0,T)}),$$

$$\max_{0\le t\le T}\|\mathbf{Y}-\mathbf{Y}_r\|_{\mathbb{R}^M} \le C(1+T^{1/2})\frac{\tau^{\min(r+1,s)}}{r^{s-1}}(T^{1/2}\|\mathbf{Y}\|_{W^{s+1,\infty}(0,T)} + \|\mathbf{R}\|_{H^s(0,T)}),$$

where the constant C is independent of $\tau, r, \mathbb{D}$, and $\mathbf{R}$ but may depend on s.

We remark that the first estimate in Theorem 4.2 is optimal both in τ and r.

5. Numerical examples. In this section, we provide some numerical examples to confirm the theoretical results in this paper. The computations are (partly) done on the high performance computers of State Key Laboratory of Scientific and Engineering Computing, Chinese Academy of Sciences.

Example 1 (convection-diffusion problem). Let $\Omega=(0,1)\times(0,1)$ and $T=1$. We consider the following constant coefficient convection-diffusion problem:

$$\begin{cases} u_t+\nabla\cdot(\boldsymbol{\beta}u-\epsilon\nabla u)=f & \text{in } \Omega\times(0,T),\\ u(\mathbf{x},0)=u_0(\mathbf{x}) & \text{in } \Omega. \end{cases} \tag{5.1}$$

The boundary condition is set to be periodic. The source term f is chosen such that the exact solution is $u(\mathbf{x},t)=\exp(-t)\sin(4\pi(x_1-t))\cos(4\pi(x_2-t))$.

We choose $\boldsymbol{\beta}=(1,1)^T$ and $\epsilon=1$ in (5.1). For spatial discretizations, we apply the local discontinuous Galerkin (LDG) method in Cockburn and Shu [9] by using purely upwind fluxes for convection terms and alternating fluxes for diffusion terms. For the sake of completeness, we recall the method for solving (5.1) here.

Let $\mathcal{M}$ denote a uniform Cartesian mesh of Ω with h the length of the sides of the elements. $\mathcal{E}=\mathcal{E}^{\mathrm{side}}\cup\mathcal{E}^{\mathrm{bdy}}$, where $\mathcal{E}^{\mathrm{side}}:=\{e=\partial K\cap\partial K' : K,K'\in\mathcal{M}\}$, $\mathcal{E}^{\mathrm{bdy}}:=\{e=\partial K\cap\partial\Omega : K\in\mathcal{M}\}$. For any subset $\widehat{\mathcal{M}}\subset\mathcal{M}$ and $\widehat{\mathcal{E}}\subset\mathcal{E}$, we use the notation

$$(u,v)_{\widehat{\mathcal{M}}}=\sum_{K\in\widehat{\mathcal{M}}}(u,v)_K,\quad \langle u,v\rangle_{\widehat{\mathcal{E}}}=\sum_{e\in\widehat{\mathcal{E}}}\langle u,v\rangle_e,$$

where $(\cdot,\cdot)_K$ and $\langle\cdot,\cdot\rangle_e$ denote the inner products of $L^2(K)$ and $L^2(e)$, respectively.

For any $e\in\mathcal{E}$, we fix a unit normal vector $\mathbf{n}_e$ of e with the convention that $\mathbf{n}_e$ is the unit outer normal to $\partial\Omega$ if $e\in\mathcal{E}^{\mathrm{bdy}}$. For any $v\in H^1(\mathcal{M}):=\{v: v\in H^1(K), K\in\mathcal{M}\}$, we define the jump operator of v across e:

$$[\![v]\!]_e:=v^- - v^+\quad\forall e\in\mathcal{E}^{\mathrm{side}},\qquad [\![v]\!]_e:=v^-\quad\forall e\in\mathcal{E}^{\mathrm{bdy}},$$

where $v^{\pm}(\mathbf{x}):=\lim_{\varepsilon\to 0^+}v(\mathbf{x}\pm\varepsilon\mathbf{n}_e)$ for all $\mathbf{x}\in e$. For any integer $p\geq 0$, we define the finite element space

$$V_h^p:=\{v\in L^2(\Omega): v|_K\in Q^p(K), K\in\mathcal{M}\},$$

where $Q^p(K)$ denotes the space of polynomials of degree at most p in each variable in K.

The semidiscrete problem is to find $(u_h,\mathbf{q}_h)\in[V_h^p]^3$ such that, for all test functions $(v_h,\mathbf{r}_h)\in[V_h^p]^3$,

$$\begin{aligned} &(\partial_t u_h,v_h)_{\mathcal{M}}+\mathcal{G}(\boldsymbol{\beta}u_h,v_h)=\sqrt{\epsilon}\left[-(\mathbf{q}_h,\nabla v_h)_{\mathcal{M}}+\langle\mathbf{q}_h^-\cdot\mathbf{n},[\![v_h]\!]\rangle_{\mathcal{E}}\right]+(f,v_h)_{\mathcal{M}},\\ &(\mathbf{q}_h,\mathbf{r}_h)_{\mathcal{M}}=\sqrt{\epsilon}\left[-(u_h,\operatorname{div}\mathbf{r}_h)_{\mathcal{M}}+\langle u_h^+,[\![\mathbf{r}_h]\!]\cdot\mathbf{n}\rangle_{\mathcal{E}}\right],\\ &u_h(\mathbf{x},0)=(\mathcal{P}_h u_0)(\mathbf{x})\quad\text{in } \Omega. \end{aligned}$$

Here $\mathcal{P}_h: L^2(\Omega)\to V_h^p$ is the standard L^2 projection operator, and $\mathcal{G}(\boldsymbol{\beta}u_h,v_h)=-(\boldsymbol{\beta}u_h,\nabla v_h)_{\mathcal{M}}+\langle\check{u}_h\boldsymbol{\beta}\cdot\mathbf{n},[\![v_h]\!]\rangle_{\mathcal{E}}$, where $\check{u}_h$ is chosen as the upwind flux: $\check{u}_h=u_h^-$ if $\boldsymbol{\beta}\cdot\mathbf{n}>0$, $\check{u}=u_h^+$ if $\boldsymbol{\beta}\cdot\mathbf{n}<0$. For $e\in\mathcal{E}^{\mathrm{bdy}}$, we use the periodic boundary condition to define u_h^+.

The optimal L^2 norm error estimate of order $p+1$ of the semidiscrete scheme for quasi-uniform Cartesian meshes can be found in Cheng, Meng, and Zhang [8, Theorem 2.4], where it is shown that $\max_{0\le t\le T}\|u-u_h\|_{L^2(\Omega)} \le C(1+T)h^{p+1}$. Therefore, combined with the continuous time Galerkin scheme, we know that the fully discrete scheme has $O(h^{p+1}+\tau^{r+1})$ accuracy in the norm $\|\cdot\|_{L^\infty(0,T;L^2(\Omega))}$ and $O(h^{p+1}+\tau^{2r})$ in the L^2 norm at nodes $t=t_n$, $n=1,\dots,N$.

To test the accuracy at the nodes, we set $\tau = h^{\frac{p+1}{2r}}$ and thus $N=T/\tau=T\beta^{\frac{1}{r}}$, where $\beta = h^{-\frac{p+1}{2}}$. According to Remark 3.3, the wall time of using Algorithm 3.1 in the sequential computation is proportional to $rN = Tr\beta^{\frac{1}{r}}$ that minimizes at $r=\ln\beta$ for $r>0$. This implies that the optimal choice of the order for the sequential computation is $r=\lfloor \ln\beta\rfloor+1$, where $\lfloor a\rfloor$ is the maximum integer strictly less than $a>0$. Table 5.1 shows the error $\|(u-u_h)(\cdot,T)\|_{L^2(\Omega)}$ at the terminal time when $r=\lfloor\ln\beta\rfloor+1$. The optimal $(p+1)$th order is observed that confirms our theoretical results. We observe that the errors of high-order schemes are significantly smaller than the low order schemes.

To test the accuracy in the $\|\cdot\|_{L^\infty(0,T;L^2(\Omega))}$ norm, we set $\tau = h^{\frac{p+1}{r+1}}$ and thus $N=T/\tau = T\gamma^{\frac{1}{r+1}}$, where $\gamma = h^{-(p+1)}$. The wall time of using Algorithm 3.2 in the sequential computation is proportional to $r(r+1)N = Tr(r+1)\gamma^{\frac{1}{r+1}}$ that is increasing in r if $\ln\gamma\le 6$ and minimizes at $r^* = [-(3-\ln\gamma)+\sqrt{(3-\ln\gamma)^2-8}]/4$ if $\ln\gamma\ge 6$. Since $r^*\ge 1$ is equivalent to $\ln\gamma\ge 6$, the optimal choice of the order for minimizing the computation wall time is $r=\max(1,\lfloor r^*\rfloor+1)$. Table 5.2 shows the error

$$\max_{0\le n\le N-1,1\le k\le 10}\|(u-u_h)(\cdot,t_n+0.1k\tau_n)\|_{L^2(\Omega)}$$

as the approximation of $\|u-u_h\|_{L^\infty(0,T;L^2(\Omega))}$ when $r=\max(1,\lfloor r^*\rfloor+1)$. We again observe the optimal $(p+1)$th order convergence and that high-order methods perform much better than low order methods.

Example 2 (wave propagation problem). Let $\Omega=(-2,2)\times(-2,2)$ and $T=1$. We consider the following wave equation with discontinuous coefficients:

Table 5.1
Example 1: numerical errors of $\|(u-u_h)(\cdot,T)\|_{L^2(\Omega)}$ and orders.

	$p=3$			$p=4$			$p=5$		
h	r	error	order	r	error	order	r	error	order
1/4	3	8.06E-03	–	4	1.29E-03	–	5	1.71E-04	–
1/8	5	5.64E-04	3.84	6	4.40E-05	4.88	7	2.84E-06	5.91
1/16	6	3.55E-05	3.99	7	1.37E-06	5.01	9	4.33E-08	6.04
1/32	7	2.03E-06	4.13	9	4.26E-08	5.00	11	6.93E-10	5.96

Table 5.2
Example 1: numerical errors in $\|\cdot\|_{L^\infty(0,T;L^2(\Omega))}$ norm and orders.

	$p=3$			$p=4$			$p=5$		
h	r	error	order	r	error	order	r	error	order
1/4	1	2.77E-02	–	2	3.99E-03	–	3	5.81E-04	–
1/8	3	2.00E-03	3.79	4	1.36E-04	4.87	5	1.02E-05	5.83
1/16	4	1.14E-04	4.14	6	4.40E-06	4.95	7	1.69E-07	5.91
1/32	6	7.18E-06	3.98	8	1.33E-07	5.05	9	2.74E-09	5.95

$$
\begin{cases}
\dfrac{1}{\rho c^2}\partial_t u = \operatorname{div}\mathbf{q} + f, \quad \rho\partial_t \mathbf{q} = \nabla u \quad \text{in } \Omega\times(0,T),\\
[\![u]\!] = 0, \quad [\![\mathbf{q}\cdot\mathbf{n}]\!] = 0 \quad \text{on } \Gamma\times(0,T),\\
u = 0 \quad \text{on } \partial\Omega\times(0,T),\\
u(\mathbf{x},0) = u_0(\mathbf{x}), \quad \mathbf{q}(\mathbf{x},0) = \mathbf{q}_0(\mathbf{x}) \quad \text{in } \Omega.
\end{cases}
\tag{5.2}
$$

We assume the interface Γ is the union of two closely located ellipses. We take $\Omega_1 = \{\mathbf{x}\in\Omega : \frac{(x_1-d_1)^2}{a^2} + \frac{x_2^2}{b^2} < 1 \text{ or } \frac{(x_1-d_2)^2}{a^2} + \frac{x_2^2}{b^2} < 1\}$, which is the union of two disks, and $\Omega_2 = \Omega\backslash\bar{\Omega}_1$. Here $d_1 = -0.82$, $d_2 = 0.82$, $a = 0.81$, and $b = 0.51$. The distance between two ellipses is 0.02. We consider the wave equation (5.2) with $\rho_1 = 1/2$, $\rho_2 = 1$, $c_1 = c_2 = 1$, and the source f is chosen such that the exact solution is

$$
u(\mathbf{x},t) = \begin{cases}
\cos(3t)\sin(r_1-1)\sin(r_2-1)\sin(3\pi x_1)\sin(3\pi x_2) & \text{in } \Omega_1,\\[2ex]
2\cos(3t)\sin(r_1-1)\sin(r_2-1)\sin(3\pi x_1)\sin(3\pi x_2) & \text{in } \Omega_2,
\end{cases}
$$

where $r_1 = \frac{(x_1-d_1)^2}{a^2} + \frac{x_2^2}{b^2}$, $r_2 = \frac{(x_1-d_2)^2}{a^2} + \frac{x_2^2}{b^2}$. The exact solution $\mathbf{q}(\mathbf{x},t)$ is computed by (5.2) with the initial condition $\mathbf{q}_0 = 0$.

We use the unfitted finite element method in Chen, Liu, and Xiang [7] to discretize the problem in space. Let $\mathcal{M}$ be an induced mesh, which is constructed from a Cartesian partition $\mathcal{T}$ of the domain Ω with possible local refinements and hanging nodes so that the elements are large with respect to both domains Ω_1, Ω_2. A reliable algorithm to generate the induced mesh from the Cartesian mesh $\mathcal{T}$ is constructed in Chen, Liu, and Xiang [7] for any C^2-smooth interface. Let $\mathcal{M}^\Gamma := \{K\in\mathcal{M} : K\cap\Gamma\neq\emptyset\}$ and $\mathcal{E} = \mathcal{E}^{\text{side}}\cup\mathcal{E}^\Gamma\cup\mathcal{E}^{\text{bdy}}$, where $\mathcal{E}^\Gamma := \{\Gamma_K = \Gamma\cap K : K\in\mathcal{M}\}$.

For any $K\in\mathcal{M}^\Gamma$, $i = 1,2$, let $K_i = K\cap\Omega_i$ and let K_i^h be the polygonal approximation of K_i bounded by the sides of K and Γ_K^h, which is the line segment connecting two intersection points of $\Gamma_K\cap\partial K$. K_i^h is the union of shape regular triangles K_{ij}^h, $1\le J_i^K\le 3$, whose sides are the sides of K_i^h and Γ_K^h. We always set K_{i1}^h to be the element having Γ_K^h as one of its sides. From K_{ij}^h we define the curved element $\widetilde{K}_{ij}^h$ by

$$
\widetilde{K}_{i1}^h = (K_i\cap K_{i1}^h)\cup(K_i\backslash\bar{K}_{i1}^h), \quad \widetilde{K}_{ij}^h = K_i\cap K_{ij}^h, \quad j = 2,\dots,J_i^K.
$$

Then we know that K is the union of curved triangles $\widetilde{K}_{ij}^h$, $i = 1,2, j = 1,\dots,J_i^K$.

For any integers $p,q\ge 1$, the space $P^p(K)$ denotes the space of polynomials of degree at most p in K and $Q^{p,q}(K)$ denotes the space of polynomials of degree at most p for the first variable and q for the second variable in K. For any $K\in\mathcal{M}^\Gamma$, we define the interface finite element spaces

$$
W_p(K) = \{\varphi : \varphi|_{\widetilde{K}_{ij}^h}\in P^p(\widetilde{K}_{ij}^h),\ i = 1,2,\ j = 1,\dots,J_i^K\},
$$

and $X_p(K) = W_p(K)\cap H^1(K_1\cup K_2)$. Notice that the functions in $X_p(K)$ are conforming in each $K_i, i = 1,2$. Now we define the following unfitted finite element spaces:

$$
\begin{aligned}
X_p(\mathcal{M}) &:= \{v\in H^1(\Omega_1\cup\Omega_2) : v|_K\in X_p(K) \ \ \forall K\in\mathcal{M}^\Gamma,\\
&\qquad v|_K\in Q^p(K) \ \ \forall K\in\mathcal{M}\backslash\mathcal{M}^\Gamma\},\\
\mathbf{W}_p(\mathcal{M}) &:= \{\boldsymbol{\psi} : \boldsymbol{\psi}|_K\in[W_p(K)]^2 \ \ \forall K\in\mathcal{M}^\Gamma,\\
&\qquad \boldsymbol{\psi}|_K\in Q^{p-1,p}(K)\times Q^{p,p-1}(K) \ \ \forall K\in\mathcal{M}\backslash\mathcal{M}^\Gamma\}.
\end{aligned}
$$

Let $X_p^0(\mathcal{M}) = X_p(\mathcal{M}) \cap H_0^1(\Omega_1 \cup \Omega_2)$, where $H_0^1(\Omega_1 \cup \Omega_2) = \{v \in H^1(\Omega_1 \cup \Omega_2) : v = 0 \text{ on } \partial\Omega\}$.

The semidiscrete unfitted finite element method for solving (5.2) is then to find $(u_h, \mathbf{q}_h) \in X_p^0(\mathcal{M}) \times \mathbf{W}_p(\mathcal{M})$ such that for all $(\varphi_h, \boldsymbol{\psi}_h) \in X_p^0(\mathcal{M}) \times \mathbf{W}_p(\mathcal{M})$,

$$\left(\frac{1}{\rho c^2}\partial_t u_h, \varphi_h\right)_{\mathcal{M}} = -(\mathbf{q}_h, \nabla\varphi_h)_{\mathcal{M}} + \langle \mathbf{q}_h^- \cdot \mathbf{n}, [\![\varphi_h]\!]\rangle_{\mathcal{E}^\Gamma} + (f, \varphi_h)_{\mathcal{M}}, \tag{5.3}$$

$$(\rho\partial_t \mathbf{q}_h, \boldsymbol{\psi}_h)_{\mathcal{M}} = -(u_h, \mathrm{div}\boldsymbol{\psi}_h)_{\mathcal{M}} + \langle u_h^+, [\![\boldsymbol{\psi}_h]\!] \cdot \mathbf{n}\rangle_{\mathcal{E}}, \tag{5.4}$$

$$u_h(\mathbf{x}, 0) = (\mathcal{P}_h u_0)(\mathbf{x}), \quad \mathbf{q}_h(\mathbf{x}, 0) = (\boldsymbol{P}_h \mathbf{q}_0)(\mathbf{x}) \ \text{ in } \Omega, \tag{5.5}$$

where $\mathcal{P}_h : L^2(\Omega) \to X_p^0(\mathcal{M})$ and $\boldsymbol{P}_h : [L^2(\Omega)]^2 \to \mathbf{W}_p(\mathcal{M})$ are the standard L^2 projection operators.

It is shown in [7, Theorem 2.2] that the following energy error of the semidiscrete scheme has pth order convergence:

$$E_{en}(t) := (\|(u-u_h)(\cdot,t)\|^2_{L^2(\Omega)} + \|(\mathbf{q}-\mathbf{q}_h)(\cdot,t)\|^2_{L^2(\Omega)})^{1/2}.$$

The semidiscrete problem (5.3)–(5.5) is an ODE system, which is solved by the continuous time Galerkin method in this paper. By Theorems 2.3 and 2.4, we know that the energy error has $O(h^p + \tau^{r+1})$ convergence rate in the norm $\max_{0\le t\le T} E_{en}(t)$ and $O(h^p + \tau^{2r})$ in $E_{en}(t)$ at nodes $t = t_n$, $n = 1, \ldots, N$.

Note that these two ellipses are close but not tangent. To resolve the interface Γ well, we locally refine the mesh near the interface such that the interface deviation $\eta_K \le \eta_0 = 0.05$ for all $K \in \mathcal{M}^\Gamma$. For the concept of the interface deviation we refer the reader to [7, Definition 2.2]; see also Chen, Li, and Xiang [6, Definition 2.2]. As an illustration, we show the computational mesh for $h = 1/4$ in Figure 5.1.

In this example, we test the accuracy of the error in the $\|\cdot\|_{L^\infty(0,T;L^2(\Omega))}$ norm and $E_{en}(T)$. As in Example 1, to test the accuracy at nodes, we set $\tau = h^{\frac{p}{2r}}$, and thus the wall time of using Algorithm 3.1 for the sequential computation is proportional to $Tr\nu^{\frac{1}{r}}$ that minimizes at $r = \ln\nu$ for $r > 0$, where $\nu = h^{-\frac{p}{2}}$. Table 5.3 shows the error $E_{en}(T)$ at the terminal time when $r = \lfloor \ln\nu \rfloor + 1$.

To test the accuracy in the $\|\cdot\|_{L^\infty(0,T;L^2(\Omega))}$ norm, we set $\tau = h^{\frac{p}{r+1}}$ and thus $N = T/\tau = T\mu^{\frac{1}{r+1}}$, where $\mu = h^{-p}$. As in Example 1, the wall time of using Algorithm 3.2 for the sequential computation is proportional to $r(r+1)N = Tr(r+1)\mu^{\frac{1}{r+1}}$.

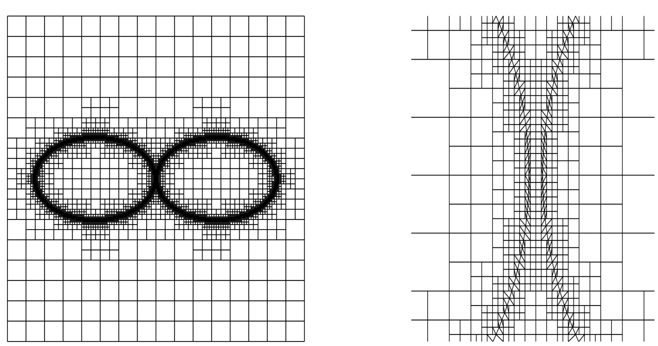

FIG. 5.1. *Illustration of the computational domain and the mesh (left) and the corresponding zoomed local mesh (right) with $h = 1/4$ in Example 2.*

Table 5.3
Example 2: numerical errors of $E_{en}(T)$ and orders.

	$p=3$			$p=4$			$p=5$		
h	r	error	order	r	error	order	r	error	order
1/4	3	4.22E-01	–	3	2.21E-01	–	4	1.25E-01	–
1/8	4	8.97E-02	2.24	5	2.64E-02	3.06	6	6.20E-03	4.33
1/16	5	1.32E-02	2.76	6	1.81E-03	3.87	7	3.82E-05	4.96
1/32	6	1.66E-03	2.99	7	1.13E-04	4.00	9	1.19E-06	5.00

Table 5.4
Example 2: numerical errors of $\max_{0\le t\le T} E_{en}(t)$ and orders.

	$p=3$			$p=4$			$p=5$		
h	r	error	order	r	error	order	r	error	order
1/4	1	5.23E-01	–	1	2.95E-01	–	2	1.69E-01	–
1/8	2	1.26E-01	2.05	3	3.70E-02	3.00	4	8.87E-03	4.25
1/16	3	1.90E-02	2.72	4	2.58E-03	3.84	6	2.93E-04	4.92
1/32	4	2.45E-03	2.96	6	1.66E-04	3.95	8	9.21E-06	4.99

The optimal choice of the order for minimizing the computation wall time is $r = \max(1, \lfloor r^{**}\rfloor + 1)$, where $r^{**} = [-(3-\ln\mu) + \sqrt{(3-\ln\mu)^2 - 8}]/4$. Table 5.3 shows the error

$$\max_{0\le n\le N-1, 1\le k\le 10} E_{en}(t_n + 0.1k\tau_n)$$

as the approximation of $\max_{0\le t\le T} E_{en}(t)$ when $r = \max(1, \lfloor r^{**}\rfloor + 1)$. We clearly observe the optimal pth order convergence and the superior performance of high-order methods from Tables 5.3 and 5.4.

Example 3. We consider the parallel efficiency of Algorithms 3.1 and 3.2 in this example. We use the same setting as that of Example 1, but the exact solution is $u(\mathbf{x}, t) = \exp(-t)\sin(16\pi(x_1))\cos(16\pi(x_2))$. We set the mesh size $h = 1/128$ and $p = 3, 4, 5$.

In this example, we use the Fortran language with MPI parallelism and solve the linear systems of equations by using the Intel direct solver MKL_Pardiso function. We remark that one can also use the parallel mode to solve the linear systems of equations to reduce the total computational time. Here we only use the serial mode for solving the linear systems of equations because we want to compare the efficiency of our algorithms with and without parallelization.

We first consider Algorithm 3.1. We take the time step size as $\tau = 1$ to treat the continuous time Galerkin method as the spectral method, that is, we only take one time step to the final time $T = 1$ and reduce the errors of numerical solutions through increasing the order r. Figure 5.2 shows the errors of the numerical solutions converge exponentially to the error level of the spatial discretization as r increases, which conforms with the convergence rates of the spectral method. Figure 5.3 shows the efficiency of the parallel mode compared to the serial mode. We observe clearly that the computational time is nearly constant in the parallel mode but increases linearly in the serial mode as r increases. Table 5.5 shows the parallel speedup ratio S_p that confirms our theoretical prediction in Remark 3.3.

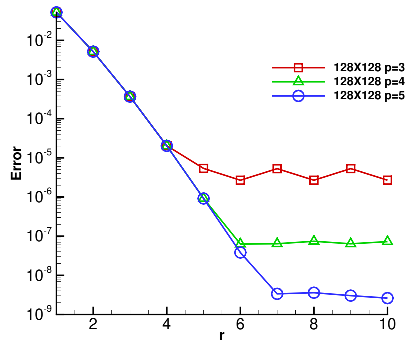

FIG. 5.2. *The errors of numerical solutions versus r.*

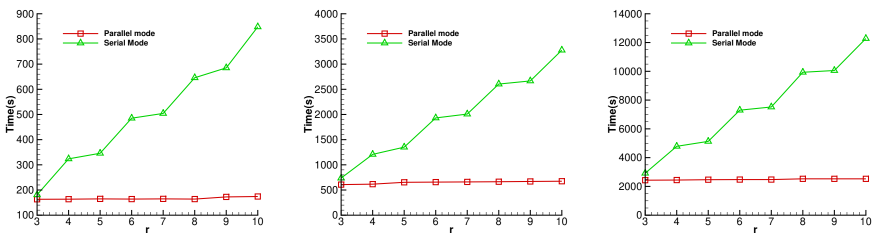

FIG. 5.3. *The computational time (seconds) in the serial mode and parallel mode for $p = 3$ (left), 4 (middle), 5 (right).*

TABLE 5.5
Example 3: the parallel speedup ratio S_p for $r = 3, \ldots, 10$.

	M	$r=3$	$r=4$	$r=5$	$r=6$	$r=7$	$r=8$	$r=9$	$r=10$
Cores	–	2	2	3	3	4	4	5	5
$p=3$	262,144	1.11x	1.98x	2.10x	2.96x	3.06x	3.94x	3.97x	4.87x
$p=4$	409,600	1.21x	1.95x	2.07x	2.94x	3.04x	3.92x	3.98x	4.86x
$p=5$	589,824	1.20x	1.96x	2.08x	2.95x	3.03x	3.93x	3.98x	4.86x

TABLE 5.6
Example 3: the wall time (seconds) and total parallel speedup ratio $S_p^{\rm total}$ when the mesh size $h = 1/128$, $p = 5$, the number of unknowns $M = 589{,}824$, and the number of time steps $N = 10$.

	$r=3$	$r=6$	$r=10$
$T_s^{\rm total}$	89,828.94	411,746.82	1,076,034.29
Cores	8	21	55
$T_p^{\rm total}$	19,943.44	20,226.67	21,581.50
$S_p^{\rm total}$	4.50x	20.35x	49.85x

Now we consider Algorithm 3.2 using $N = 10$ time steps. The total parallel speedup ratio $S_p^{\rm total}$ is shown in Table 5.6 for $p = 5$, $h = 1/128$, $r = 3, 6, 10$, which again confirms our analysis in Remark 3.3.

REFERENCES

[1] G. Akrivis, C. Makridakis, and R. H. Nochetto, *Galerkin and Runge-Kutta methods: Unified formulation, a posteriori error estimates and nodal superconvergence*, Numer. Math., 118 (2011), pp. 429–456.

[2] A. K. Aziz and P. Monk, *Continuous finite elements in space and time for the heat equation*, Math. Comp., 52 (1989), pp. 255–274.

[3] C. Bernardi and Y. Maday, *Spectral methods*, Handb. Numer. Anal., 5 (1997), pp. 209–485.
[4] W. C. Brown, *Matrices over Commutative Rings*, Marcel Dekker, New York, 1993.
[5] J. Chen, Z. Chen, T. Cui, and L.-B. Zhang, *An adaptive finite element method for the eddy current model with circuit/field couplings*, SIAM J. Sci. Comput., 32 (2010), pp. 1020–1042, https://doi.org/10.1137/080713112.
[6] Z. Chen, K. Li, and X. Xiang, *An adaptive high-order unfitted finite element method for elliptic interface problems*, Numer. Math., 149 (2021), pp. 507–548.
[7] Z. Chen, Y. Liu, and X. Xiang, *A High Order Explicit Time Finite Element Method for the Acoustic Wave Equation with Discontinuous Coefficients*, https://arxiv.org/abs/2112.02867v2, 2023.
[8] Y. Cheng, X. Meng, and Q. Zhang, *Application of generalized Gauss-Radau projections for the local discontinuous Galerkin method for linear convection-diffusion equations*, Math. Comp., 86 (2017), pp. 1233–1267.
[9] B. Cockburn and C.-W. Shu, *The local discontinuous Galerkin method for time-dependent convection-diffusion systems*, SIAM J. Numer. Anal., 35 (1998), pp. 2440–2463, https://doi.org/10.1137/S0036142997316712.
[10] J. Douglas Jr., T. Dupont, and M. F. Wheeler, *A quasi projection analysis of Galerkin methods for parabolic and hyperbolic equations*, Math. Comp., 32 (1978), pp. 345–362.
[11] D. A. French and T. E. Peterson, *A continuous space-time finite element method for the wave equation*, Math. Comp., 65 (1996), pp. 491–506.
[12] E. Gallopoulos and Y. Saad, *On the parallel solution of parabolic equations*, in Proceedings of the 3rd International Conference on Supercomputing, ACM, 1989, pp. 17–28.
[13] M. J. Gander, 50 *years of time parallel time integration*, in Multiple Shooting and Time Domain Decomposition, T. Carraro, M. Geiger, S. Körkel, and R. Rannacher, eds., Springer-Verlag, Berlin, 2015, pp. 69–114.
[14] R. Griesmaier and P. Monk, *Discretization of the wave equation using continuous elements in time and a hybridizable discontinuous Galerkin method in space*, J. Sci. Comput., 58 (2014), pp. 472–498.
[15] E. Hairer and G. Wanner, *Solving Ordinary Differential Equations* II, Springer-Verlag, Berlin, Heidelberg, 1996.
[16] B. L. Hulme, *One-step piecewise polynomial Galerkin methods for initial value problems*, Math. Comp., 26 (1972), pp. 415–426.
[17] MathWorks Inc., *The Student Edition of MATLAB*, Version 4, User's Guide, Prentice-Hall, 1995, pp. 185–186, 559-560.
[18] T. Richter, A. Springer, and B. Vexler, *Efficient numerical realization of discontinuous Galerkin methods for temporal discretization of parabolic problems*, Numer. Math., 124 (2013), pp. 151–182.
[19] E. B. Saff and R. S. Varga, *On the zeros and Poles of Padé Approximation to* e^z, Numer. Math., 25 (1975), pp. 1–14.
[20] D. Schötzau and Ch. Schwab, *p- and hp- Finite Element Methods*, Oxford Science Publications, New York, 1998.
[21] D. Schötzau and Ch. Schwab, *Time discretization of parabolic problems by the hp-version of the discontinuous Galerkin finite element method*, SIAM J. Numer. Anal., 38 (2000), pp. 837–875, https://doi.org/10.1137/S0036142999352394.
[22] B. S. Southworth, O. Krzysik, W. Pazner, and H. De Sterck, *Fast solution of fully implicit Runge–Kutta and discontinuous Galerkin in time for numerical PDEs, part* I*: The linear setting*, SIAM J. Sci. Comput., 44 (2022), pp. A416–A443, https://doi.org/10.1137/21M1389742.
[23] G. Szegö, *Orthogonal Polynomials*, American Mathematical Society, New York, 1939.