Taming the CFL number for Discontinuous Galerkin Methods by Local Exponentiation

Daniel Appelö, Mingyu Hu and Maxim Zinchenko

Abstract The matrix valued exponential function can be used for time-stepping numerically stiff discretization, such as the discontinuous Galerkin method but this approach is expensive as the matrix is dense and necessitates global communication. In this paper, we propose a local low-rank approximation to this matrix. The local low-rank construction is motivated by the nature of wave propagation and costs significantly less to apply than full exponentiation. The accuracy of this time stepping method is inherited from the exponential integrator and the local property of it allows parallel implementation. The method is expected to be useful in design and inverse problems where many solves of the PDE are required. We demonstrate the error convergence of the method for the one-dimensional (1D) Maxwell's equation on a uniform grid.

1 Introduction

It was recognized by Kreiss and Oliger [11] almost half a century ago that high-order methods are superior in terms of accuracy for propagating waves over long distances. Since then, much research has been devoted to spectral and high-order methods and as a result, many highly accurate finite-difference, finite-element, spectral element and discontinuous Galerkin (DG) methods have been developed. Moreover, advances in computational hardware with its ever increasing level of parallelism have favored methods that are robustly stable, geometrically flexible and suitable to implement on parallel computers. The DG method posses all these qualities and has

Daniel Appelö
Michigan State University, East Lansing, USA, e-mail: appeloda@msu.edu

Mingyu Hu
University of Colorado, Boulder, USA, e-mail: mingyu.hu@colorado.edu

Maxim Zinchenko
University of New Mexico, Albuquerque, USA, e-mail: maxim@math.unm.edu

become popular among practitioners, for example in computational electromagnetics where it is gradually replacing the Yee - FDTD scheme. Although DG methods are spectrally convergent in the order q of the approximation, very high orders are rarely used in practice. A reason for this is that polynomials on bounded domains, used as bases in DG methods, have boundary layers at the element edges, [14]. These boundary layers cause the norms of matrices corresponding to differentiation to grow fast with the order, q resulting in numerical stiffness. In turn, this numerically induced stiffness forces the use of excessively small time steps, discouraging the use of very high order methods. Since polynomials vary slowly near the middle of the reference interval, methods such as co-volume filtering has been suggested to tame the CFL condition [14]. Similar ideas were also used in [12], where Li et al. formulated central DG methods for solving ideal magnetohydrodynamic (MHD) equations on overlapping meshes. Yet another way to allow for larger time step sizes is to modify the numerical flux as proposed in [1].

Another source of stiffness, not specific to DG methods, arises from small cells needed to resolve geometric features. For geometric stiffness, local time-stepping and implicit time stepping can be advantageous. Diaz and Grote [4] and Grote and Mitkova [7] proposed energy conserving explicit local time steppers for second order wave equation and Maxwell's equation, respectively. Local time-stepping using linear multistep methods has also been considered by Goedel and et al. [6]. Locally implicit energy conserving methods have been proposed in [3, 5, 13], and locally implicit upwind DG methods were considered by Hockbruck and Sturm [10].

Yet another class of time stepping methods, exponential time integrators, are capable of handling both numerically and geometrically induced stiffness. For a spatially discretized linear autonomous system of differential equations, the solution at each time step can be calculated exactly (up to machine precision) and the time stepping is stable for any choice of time step size. This is particularly beneficial for stiff problems. The literature on exponential integrators is extensive but summarized in the review paper by Hochbruck and Ostermann in [9]. In this paper we focus on the time evolution of a system of linear differential equations,

$$\mathbf{u}_t = A\mathbf{u},$$

arising from a spatial DG discretization of a linear, variable coefficient, hyperbolic PDE. We refer to entries in $\mathbf{u}$ as *degrees of freedom* or shorter, DOF. An exponential time-stepping scheme would then advance the DOF at time t_n by the use of the identity

$$\mathbf{u}(t_{n+1}) = e^{A\Delta t}\mathbf{u}(t_n) \equiv Q\mathbf{u}(t_n),$$

where thus Q is the matrix exponential solution operator corresponding to a timestep Δt. The main drawback with this approach is that the exponentiation of a matrix is an expensive and memory demanding operation. In addition as the matrix Q is in general dense, efficient parallelization can be difficult.

in this paper, to reduce the computational cost and memory requirements for finding and storing Q, we propose a local approximation to the exponential time integrator. This approximation still tames the CFL condition of DG schemes but is amenable to parallelization by domain decomposition techniques and is inexpensive to apply. The method is motivated by the finite-speed-of-propagation intrinsic to wave propagation. That is, the value of a solution at some location x and time t will only influence the solution nearby during a time interval Δt. This locality can be used to find an approximation, $\tilde{Q}$, to, Q, one row at a time by considering matrix exponentiation of low rank versions of $\Delta t A$. These low rank matrices will be chosen so that the time update of a DOF will only use nearby DOF.

The rest of the paper is organized as follows. In Section 2, we present the construction of our local exponential time integrator. We also present an error bound and discuss computational cost of the local approximation. In Section 3, we test the local exponential integrator for the one-dimensional Maxwell's equation discretized in space using DG. In Section 4, we conclude this work by pointing out potential applications of our local exponential time integrator and an outlook to future work.

2 Time-Stepping by Local Approximation to the Matrix Exponential

Consider a DG semi-discretization of a linear first order hyperbolic system on a mesh in one, two or three dimensions. This semi-discretization can be expressed as linear system of ordinary differential equations

$$\mathbf{u}_t = A\mathbf{u}, \ \ \mathbf{u} \in \mathbb{R}^M, \ \ A \in \mathbb{R}^{M \times M},$$

where M is the number of degrees of freedom in $\mathbf{u}$. To identify nearby elements and their degrees of freedom we will let $\rho(j,k)$ be the Euclidian distance between the element centers of elements j and k. The set

$$D(k,r) = \{j \in [1,\ldots,M] : \rho(k,j) \leq r\},$$

thus represents a ball of radius r in the physical space. We note that the matrix A acting on $\mathbf{u}$ can be expressed as

$$(A\mathbf{u})_k = \sum_{j=1}^{M} a_{kj} u_j, \quad k = 1,\ldots,M.$$

However, as we consider DG methods for hyperbolic problems most of the elements in A are zeros. In fact, for a degree of freedom k on a (finite) element E, only matrix elements $a_{k,j}$ with degree of freedom j belonging to the element E or its nearest neighbor elements would be nonzero. Let R_k be the radius of a circle with a center coinciding with the element center of E and let R_k be large enough for the circle to enclose the nearest neighbor elements (note that we may take R_k larger to improve the timestep constraints, see below). This allows us write the formula above as

$$(A\mathbf{u})_k = \sum_{j\in D(k,R_k)} a_{kj}u_j, \quad k = 1,\ldots,M.$$

2.1 Local Exponentiation

We now define the local exponentiation of the matrix A. For a given k, we fix R_k and form the $M\times M$ matrix $\tilde{A}^{(k)}$ by copying all the rows whose row-index belong to $D(k,R_k)$ into the $M\times M$ zero matrix one row at a time. We write this as a matrix operation $\tilde{A}^{(k)} = P^{(k)}A$ so that $P^{(k)}$ is the matrix that extracts the rows in A that corresponds to the indices in the set $D(k,R_k)$ and sets all the other rows to zero. In other words $P^{(k)}$ is a diagonal matrix with ones in the diagonal elements corresponding to the integers contained in the set $D(k,R_k)$.

With $\tilde{A}^{(k)}$ computed we can now build one part of the approximation, $\tilde{Q}\approx Q\equiv e^{\Delta tA}$. Precisely, the kth row in $\tilde{Q}$ is defined to be the kth row of the matrix

$$e^{\Delta t\tilde{A}^{(k)}}.$$

Expressing this construction in a formula we have

$$\tilde{Q}^T\delta_k = (e^{\Delta t\tilde{A}^{(k)}})^T\delta_k = e^{\Delta t\tilde{A}^{(k)T}}\delta_k = e^{\Delta tA^TP^{(k)}}\delta_k, \quad k\in\Omega, \tag{1}$$

where δ_k is the kth unit vector.

Remark 1 *The equation (1) is defined one row / DOF at a time and the set $D(k,R_k)$ above is defined so that it may contain a fraction of the number of DOF in an element as this makes the formulas short. However, in practice we work on one element at a time and process all rows of that element, one at a time, before moving to the next set of degrees of freedoms (rows).*

Remark 2 *In terms of matrix operations on the vector* $\mathbf{u}$*, the degrees of the freedom in A that are to be multiplied with the corresponding entries in* $\mathbf{u}$ *should be columns of A and one may consider the above approach but with the construction done column-by-column instead. However, our numerical experiments show that $\tilde{Q}$ constructed using rows of A yields better approximation to the full exponential time integrator Q.*

Remark 3 *Since the construction of the rows in $\tilde{Q}$ is local, depending only on degrees of freedom in A of the neighboring elements inside the circle of radius R_k of the center of the element that hold the degree of freedom k, the computation for each k can trivially be carried out in parallel.*

2.2 Exponential Locality and Decay of Discretizations of Hyperbolic Operators

The following theorem shows that for R_k large the locally exponentiated $\tilde{Q}$ is close to $Q = e^{\Delta t A}$ in the operator norm.

Theorem 1. *Let $\Omega = \{1,\dots,M\}$ and let $W = \max_{k\in\Omega} |D(k,R_k)|$ be the largest number of degrees of freedom included for the construction of any row. Further let $m = \lfloor R_k \rfloor$ and $T_m(z) = \sum_{k=m+1}^{\infty} \frac{z^k}{k!}$, then*

$$\|Q-\tilde{Q}\| \le (W+1)T_m(\|\Delta tA\|) \le (W+1)e^{\|\Delta tA\|}\frac{\|\Delta tA\|^{m+1}}{(m+1)!}.$$

Proof. It suffices to show the above norm estimate for the transposed matrices. First, note that A^T has the same nearest neighbor structure as A and hence $A^T\delta_k$ is supported on $D(k,R_k)$, that is, $(A^T\delta_k)_j = 0$ for all $j \in \Omega\backslash D(k,R_k)$. Using the triangle inequality for the metric ρ we obtain that $(A^T)^2\delta_k$ is supported on $D(k,2R_k)$ and by induction $(A^T)^n\delta_k$ is supported on $D(k,nR_k)$. Then for $n \le m$ we have $n \le R_k$ and hence $P_k(A^T)^n\delta_k = (A^T)^n\delta_k$. This implies

$$(\tilde{A}^{(k)T})^n\delta_k = (A^TP^{(k)})^n\delta_k = (A^T)^n\delta_k, \quad n \le m,\ k\in\Omega.$$

Next, consider an arbitrary vector $\mathbf{u}\in\mathbb{R}^M$ such that $\mathbf{u} = \sum_{k\in\Omega} u_k\delta_k$. Then

$$\big(Q^T-\tilde{Q}^T\big)\mathbf{u} = \sum_{k\in\Omega} u_k\big(Q^T-\tilde{Q}^T\big)\delta_k = \sum_{k\in\Omega} u_k\big(e^{\Delta tA^T}-e^{\Delta t\tilde{A}^{(k)T}}\big)\delta_k.$$

Using the Taylor series expansion for $e^{\Delta tA^T}$ and $e^{\Delta t\tilde{A}^{(k)T}}$, we get

$$\big(Q^T-\tilde{Q}^T\big)\mathbf{u} = \sum_{k\in\Omega} u_k \sum_{n=0}^{m} \frac{(\Delta tA^T)^n-(\Delta t\tilde{A}^{(k)T})^n}{n!}\delta_k + \sum_{k\in\Omega} u_k\big(T_m(\Delta tA^T)-T_m(\Delta t\tilde{A}^{(k)T})\big)\delta_k.$$

Since $(\Delta t\tilde{A}^{(k)T})^n\delta_k = (\Delta tA^T)^n\delta_k$ for all $n\le m$, the first double sum in the above equality is zero, so we have

$$\big(Q^T-\tilde{Q}^T\big)\mathbf{u} = T_m(\Delta tA^T)\mathbf{u} - \sum_{k\in\Omega} u_kT_m(\Delta t\tilde{A}^{(k)T})\delta_k.$$

Using the triangle inequality, the norm of the first term is easily estimated by $\|T_m(\Delta tA^T)u\| \le T_m(\|\Delta tA\|)\|u\|$. To estimate the norm of the second term, we first estimate the magnitude of its individual components. Noting that $(\Delta t\tilde{A}^{(k)T})^n\delta_k$ is supported on $D(k,R_k)$ for every $n\ge 0$ and that $\|\Delta t\tilde{A}^{(k)T}\| \le \|\Delta tA\|$ we have

$$\Big|\Big(\sum_{k\in\Omega} u_k T_m(\Delta t\tilde{A}^{(k)T})\delta_k\Big)_j\Big| = \Big|\sum_{k\in D(j,R_k)} u_k T_m(\Delta t\tilde{A}^{(k)T})\delta_k\Big|$$
$$\le T_m(\|\Delta tA\|)\sum_{k\in D(j,R_k)}|u_k|, \quad j\in\Omega.$$

Then, using the Cauchy-Schwarz inequality and the fact that each disk $D(j,R_k)$ contains at most W points and hence each point $k\in\Omega$ lies in at most W distinct disks $D(j,R_k)$ we get

$$\Big\|\sum_{k\in\Omega} u_k T_m(\Delta t\tilde{A}^{(k)T})\delta_k\Big\|^2 \le T_m(\|\Delta tA\|)^2\sum_{j\in\Omega}\Big(\sum_{k\in D(j,R_k)}|u_k|\Big)^2$$
$$\le T_m(\|\Delta tA\|)^2\sum_{j\in\Omega} W\sum_{k\in D(j,R_k)}|u_k|^2 \le T_m(\|\Delta tA\|)^2 W^2\|\mathbf{u}\|^2.$$

Combining the two estimates then yields

$$\big\|\big(Q^T-\tilde{Q}^T\big)\mathbf{u}\big\| \le T_m(\|\Delta tA\|)(1+W)\|\mathbf{u}\|.$$

Since u is an arbitrary vector we get

$$\|Q-\tilde{Q}\| = \|Q^T-\tilde{Q}^T\| \le T_m(\|\Delta tA\|)(1+W).$$

2.3 Cost and Complexity for Local Exponentiation

The spatial discretization matrix A of size $M\times M$ is a sparse matrix thus the local discretization matrix for the k-th DOF, $A^{(k)}$, will be even more sparse as it is composed of only a few of the entries in A. Moreover, as the vast majority of the eigenvalues of $A^{(k)}$ will be zero the degree of the minimal polynomial $\mathscr{M}_{A^{(k)}}(z)$ will be *very small* compared to M. It is well known, [2], that if the degree of $\mathscr{M}_{A^{(k)}}(z)$ is $m_{\min}$ then the matrix exponent can be explicitly computed by the formula

$$e^{\Delta tA^{(k)}} = \sum_{s=0}^{m_{\min}-1} f_s(\Delta t)\left(A^{(k)}\right)^s,$$

where $f_0(\Delta t), f_1(\Delta t),\ldots,f_{m_{\min}-1}(\Delta t)$ are analytical functions whose values can be computed by solving a linear system of equations.

Suppose $\tilde{A}^{(k)}$ has m non-zero columns with at most n non-zero entries in each column, where $m,n\ll M$. Then the cost of computing all of $e^{\Delta tA^{(k)}}$ would scale as $\mathscr{O}((\min\{m,n\})^3+\min\{m,n\}m_{\min})$, where the first term represents the cost of directly solving for $f_0,\ldots$ and the second term corresponds to the applications of the matrix powers of the sum to the kth unit vector. The cost of assembling all of the rows of $\tilde{Q}$ would increase this by a factor of M but this is still much less than computing $e^{\Delta tA}$ and, again, can be trivially parallelized.

3 Numerical Experiments in One Dimension: Maxwell's Equations

We consider example 2.6 from [8], Maxwell's equations in one-dimension domain $x \in [-\pi, \pi]$

$$\varepsilon(x)\frac{\partial E(x,t)}{\partial t} = -\frac{\partial H(x,t)}{\partial x}, \tag{2a}$$

$$\mu(x)\frac{\partial H(x,t)}{\partial t} = -\frac{\partial E(x,t)}{\partial x}, \tag{2b}$$

with homogenous boundary conditions on the electric field and and initial conditions

$$\begin{aligned} \mathrm{BC}: &\quad E(0,t) = E(2\pi,t) = 0, \\ \mathrm{IC}: &\quad E(x,0) = \sin(x),\ H(x,0) = 0. \end{aligned} \tag{3}$$

Here E and H are the electric and magnetic fields, and ε and μ are permittivity and permeability. To this end, we take $\varepsilon(x)$ and $\mu(x)$ to be 1. We note that this system assumes solutions to the three dimensional Maxwell system where the electric and magnetic fields are of the form $\mathbf{E}(x,y,z,t) = (0, E(x,t), 0)$, $\mathbf{H}(x,y,z,t) = (0,0,H(x,t))$. The exact solution to this particular problem is the standing wave $E(x,t) = \cos(t)\sin(x)$, $H(x,t) = -\sin(t)\cos(x)$.

We numerically approximate the solution with a modal DG method using Legendre polynomial bases on a uniform spatial grid with n elements. The standard upwind discontinuous Galerkin method takes the form

$$M_E\mathbf{E}_t = S\mathbf{H} + (H_R^- - H^*)\mathbf{L}_R - (H_L^- - H^*)\mathbf{L}_L, \tag{4a}$$

$$M_H\mathbf{H}_t = S\mathbf{E} + (E_R^- - E^*)\mathbf{L}_R - (E_L^- - E^*)\mathbf{L}_L, \tag{4b}$$

where M_E and M_H are the mass matrices for equations (2a) and (2b) receptively. The numerical fluxes are $E^* = \{\{E\}\} - [H]$ and $H^* = \{\{H\}\} - [E]$, where the $\{\{\cdot\}\}$ and $[\cdot]$ denotes the usual average and jump. The $-$ superscript indicates that the value is taken from the solution on the current element, and the subscripts indicate evaluation at the left and right end points of the element. The left and right lift vectors, $\boldsymbol{L}_L$ and $\boldsymbol{L}_R$, are composed of Legendre polynomials up to the q-th order evaluated at the left and right boundaries.

We refer to the modal coefficients as degrees of freedom of the problem. Then Eqn (4a) and (4b) is rewritten as

$$\mathbf{u}_t = A\mathbf{u}, \tag{5}$$

where $\hat{\mathbf{u}}$ contains all the degrees of freedom from both $\mathbf{E}$ and $\mathbf{H}$. The matrix A is the differentiation matrix for the entire system. It is observed that A is block-tridiagonal if the degrees of freedom in $\mathbf{u}$ are ordered by variables. If the degrees of freedom in $\mathbf{u}$ are ordered by elements, A is block-diagonal. For the rest of the paper, the former ordering of degrees of freedom is used. Since the system is autonomous, the exact numerical time-stepping solution is given by

$$\mathbf{u}^{(n+1)} = \exp(\Delta t A)\mathbf{u}^{(n)} = Q\mathbf{u}^{(n)}, \tag{6}$$

where Δt is the time step size, $\mathbf{u}^{(n)}$ is the numerical solution at time t_n, and Q is the matrix representing the exponential time integrator. All matrix exponentiation in our numerical examples are computed using the MATLAB built-in function `expm`. With the proposed local exponentiation construction in Section 2, stability of time-stepping with the local exponentiation matrix $\tilde{Q}$ is investigated and the convergence of error is compared with n_{Taylor}-stage Taylor time-stepping method. We also compare the performance of $\tilde{Q}_{col}$, which is constructed using column degrees of freedom of A, to that of $\tilde{Q}_{row}$, which is constructed using rows degrees of freedom of A. While $\tilde{Q}_{col}$ is more physically intuitive in terms of matrix-vector multiplication, however, we will show that $\tilde{Q}_{row}$ demonstrates better approximation.

The parameters in the tests are the number of elements, denoted by n, time step size Δt, the highest degree of Legendre polynomial q, and the number of neighbor elements (dependent element(s)) denoted by d.e. taken on each side of the k-th element. For the base case, the parameters take the values shown in Table 1.

Parameter	Value	Parameter (cont'd)	Value
n	20	# of d.e.	3
q	3	# of quadrature points	$q+4$
t_{final}	20	n_{Taylor}	4
h_t, uniform	$0.05hn_{\text{Taylor}}/q$		

Table 1 Base testing parameter values for 1D Maxwell's equations.

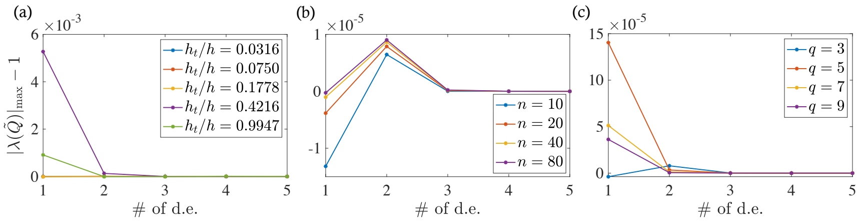

Fig. 1 Eigenvalue discrepancy of $\tilde{Q}_{row}$ v.s. # of d.e. for different (a) $\Delta t/h$, (b) n, (c) q. Note that $\Delta t = h_t$ in the figure.

3.1 Comparison of Spectra

In this section, we investigate the stability of the local exponentiation time-stepping based on the eigenvalue of the time-stepping operator $\tilde{Q}$. Eigenvalues with modulus greater than 1 will lead to instability during time-stepping. Figure 1 shows $\xi(.\mathrm{d.e}) = (|\lambda(\tilde{Q})|_{max} - 1)$ as a function of d.e. in the construction of $\tilde{Q}$ using row degree of freedom.

The eigenvalues with imaginary part falling in $(-0.01, 0.01)$ are excluded from the plot as those eigenvalues are not computed accurately numerically. Except for the varying parameters, all other parameters take the base values listed in Table 1. By varying different parameters, we also justify that our choices of base parameter values construct a stable local exponential time integrator $\tilde{Q}$ with minimum cost in its computation. In particular, Fig. 1(b) shows that the number of d.e., independent of the refinement of the spatial discretization, is the key variable for the stability of $\tilde{Q}$. Overall, the function ξ is small for high-order polynomial basis and relaxed CFL condition. Thus the local exponentiation time stepper is expected to be stable with appropriate combinations of d.e., Δt, and q.

When only one d.e. is taken on each side of each element in the construction of $\tilde{Q}$, Figure 2 shows that $\tilde{Q}_{row}$, in general, is a better approximation to Q than $\tilde{Q}_{col}$. The latter clearly would lead to an unstable scheme in this case but that also $\tilde{Q}_{col}$ will become stable if d.e. is large enough.

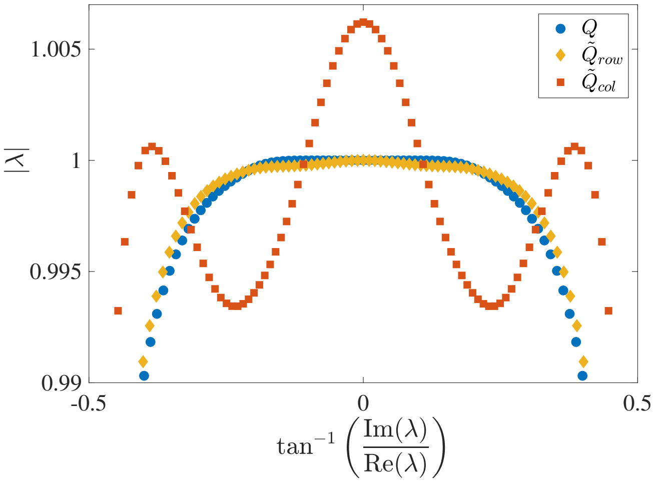

Fig. 2 Distribution of eigenvalues whose modulus is close to 1.

3.2 Convergence Properties of the Method

The convergence of error for both row neighbors and column neighbors is investigated in this section and is compared to fourth-order Taylor time-stepping. Fig. 3 shows that, compared to column degree of freedom, row degree of freedom gives smaller error and and the error converges to the 4−stage Taylor time-stepping error on a coarser grid. Fig. 4 shows that while both row-neighbor- and column-neighbor-constructed local exponential time integrators show advantage in stability, the errors are smaller with row degrees of freedom. In general, the local exponential time integrator demonstrates the possibility to take much larger time step size than that is allowed by other numerical time-stepping methods and the method is robust so high-order spatial approximation can be adopted.

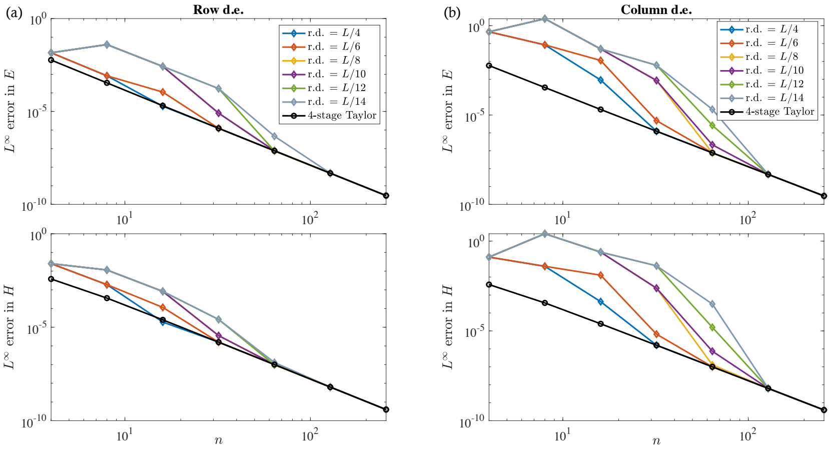

Fig. 3 Error convergence with respect to n on the uniform grid: (a) L^∞ errors, row degree of freedom; (b) L^∞ errors, column degree of freedom.

Base on the above observations, we conclude that in the construction of $\tilde{A}^{(k)}$, rows should be taken as dependent degrees of freedom in the assembly of $\tilde{Q}$. In Fig. 5, we display the error convergence versus the number of d.e. for different CFL numbers to demonstrate the efficiency of the local exponentiation time integrator in taming the CFL restriction which is inherited from the DG discretization. The number of elements used is $n = 100$ for more prominent effect. With less than 20% of total degrees of freedom, the local exponentiation time stepper is able to achieve the same or smaller error compared to the explicit 4-stage Taylor time-stepping but with 10 times larger CFL number than permitted by the Taylor time-stepping.

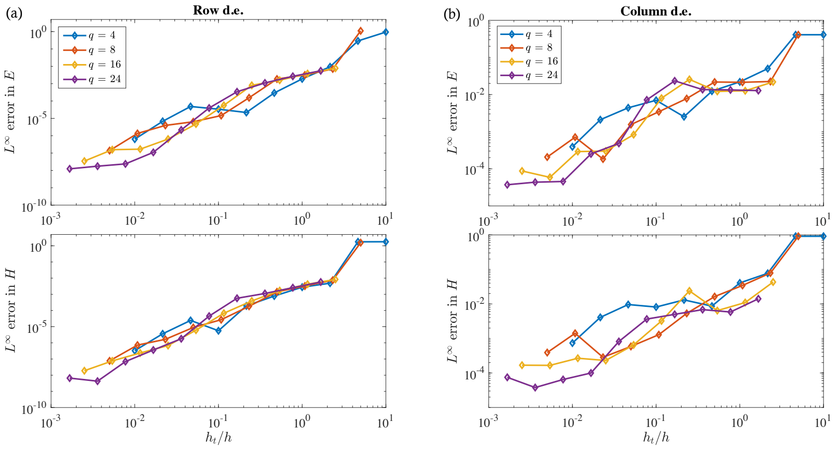

Fig. 4 Error convergence with respect to h_t/h on the uniform grid: (a) L^∞ errors, row degree of freedom; (b) L^∞ errors, column degree of freedom.

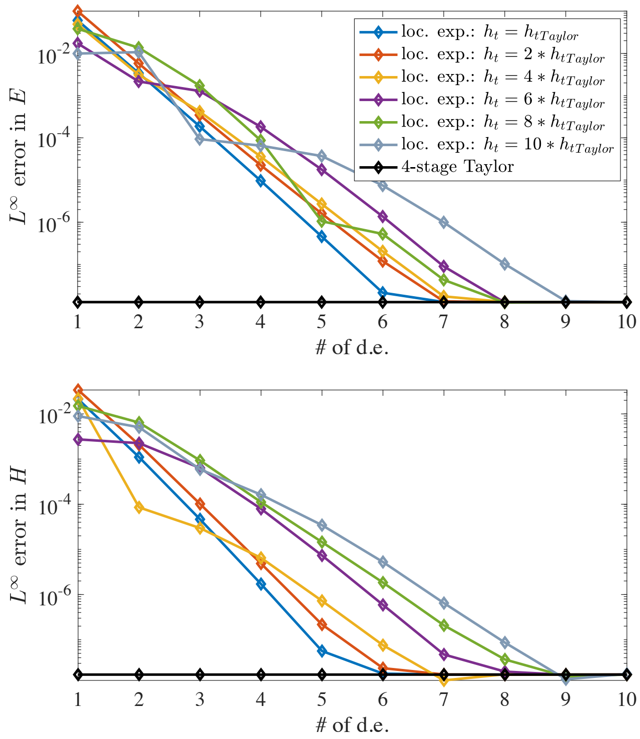

Fig. 5 Error convergence with respect to the number of d.e. with row degree of freedom on a uniform grid.

4 Conclusion

An approximate exponential time integrator is proposed and tested for Maxwell's equation in 1D. The construction of this approximate is local, leading to cheaper computation compared to the exponential integrator and potential parallel implementations. The size and structure of the domain of dependency in constructing the local exponential time integrator will be determined specifically based on different physical settings. Our numerical experiments have shown that this local exponentiation time-stepper is robust to high-order spatial discretization and is stable with at least 10 times larger the CFL number required by explicit time-stepping methods.

References

1. Noel Chalmers, Lilia Krivodonova, and Ruibin Qin. Relaxing the CFL number of the discontinuous Galerkin method. *SIAM Journal on Scientific Computing*, 36(4):A2047–A2075, 2014.
2. Hon-Wing Cheng and Stephen S.-T. Yau. More explicit formulas for the matrix exponential. *Linear Algebra and its Applications*, 262:131–163, 1997.
3. Stéphane Descombes, Stéphane Lanteri, and Ludovic Moya. Locally implicit time integration strategies in a discontinuous Galerkin method for Maxwell's equations. *Journal of Scientific Computing*, 56(1):190–218, 2013.
4. Julien Diaz and Marcus J. Grote. Energy conserving explicit local time stepping for second-order wave equations. *SIAM Journal on Scientific Computing*, 31(3):1985–2014, 2009.
5. Victorita Dolean, Hassan Fahs, Loula Fezoui, and Stéphane Lanteri. Locally implicit discontinuous Galerkin method for time domain electromagnetics. *Journal of Computational Physics*, 229(2):512–526, 2010.
6. N Gödel, S Schomann, T Warburton, and M Clemens. Local timestepping discontinuous Galerkin methods for electromagnetic RF field problems. In *Antennas and Propagation, 2009. EuCAP 2009. 3rd European Conference on*, pages 2149–2153. IEEE, 2009.
7. Marcus J Grote and Teodora Mitkova. Explicit local time-stepping methods for Maxwell's equations. *Journal of Computational and Applied Mathematics*, 234(12):3283–3302, 2010.
8. Jan S Hesthaven and Tim Warburton. *Nodal discontinuous Galerkin methods: algorithms, analysis, and applications*. Springer Science & Business Media, 2007.
9. Marlis Hochbruck and Alexander Ostermann. Exponential integrators. *Acta Numerica*, 19:209–286, 2010.
10. Marlis Hochbruck and Andreas Sturm. Upwind discontinuous Galerkin space discretization and locally implicit time integration for linear Maxwell's equations. *Mathematics of Computation*, 2018.
11. Heinz-Otto Kreiss and Joseph Oliger. Comparison of accurate methods for the integration of hyperbolic equations. *Tellus*, 24(3):199–215, 1972.
12. Fengyan Li, Liwei Xu, and Sergey Yakovlev. Central discontinuous Galerkin methods for ideal MHD equations with the exactly divergence-free magnetic field. *Journal of Computational Physics*, 230(12):4828–4847, 2011.
13. Serge Piperno. Symplectic local time-stepping in non-dissipative DGTD methods applied to wave propagation problems. *ESAIM: Mathematical Modelling and Numerical Analysis*, 40(5):815–841, 2006.
14. Timothy Warburton and Thomas Hagstrom. Taming the CFL number for discontinuous Galerkin methods on structured meshes. *SIAM Journal on Numerical Analysis*, 46(6):3151–3180, 2008.