

*Communications in
Applied
Mathematics and
Computational
Science*

A REAL-SPACE GREEN'S FUNCTION METHOD
FOR
THE NUMERICAL SOLUTION OF MAXWELL'S
EQUATIONS

BORIS LO, VICTOR MINDEN AND PHILLIP COLELLA

vol. 11 no. 2 2016

A REAL-SPACE GREEN'S FUNCTION METHOD FOR THE NUMERICAL SOLUTION OF MAXWELL'S EQUATIONS

BORIS LO, VICTOR MINDEN AND PHILLIP COLELLA

A new method for solving the transverse part of the free-space Maxwell equations in three dimensions is presented. By taking the Helmholtz decomposition of the electric field and current sources and considering only the divergence-free parts, we obtain an explicit real-space representation for the transverse propagator that explicitly respects finite speed of propagation. Because the propagator involves convolution against a singular distribution, we regularize via convolution with smoothing kernels (B-splines) prior to sampling based on a method due to Beyer and LeVeque (1992). We show that the ultimate discrete convolutional propagator can be constructed to attain an arbitrarily high order of accuracy by using higher-order regularizing kernels and finite difference stencils and that it satisfies von Neumann's stability condition. Furthermore, the propagator is compactly supported and can be applied using Hockney's method (1970) and parallelized using the same observation as made by Vay, Haber, and Godfrey (2013), leading to a method that is computationally efficient.

1. Introduction

In this paper, we present a method for solving Maxwell's equations. Our approach will be based on the expression of the evolution of the magnetic and transverse electric fields in terms of a first-order, constant-coefficient hyperbolic system

$$\begin{aligned} \frac{\partial \mathbf{u}(\mathbf{x}, t)}{\partial t} &= L\mathbf{u}(\mathbf{x}, t) + \mathbf{f}(\mathbf{x}, t), & (\mathbf{x}, t) \in \mathbb{R}^D \times \mathbb{R}_+, \\ \mathbf{u}(\mathbf{x}, 0) &= \mathbf{u}_0(\mathbf{x}), & \mathbf{x} \in \mathbb{R}^D, \end{aligned} \quad (1)$$

where L is a constant-coefficient first-order linear differential operator in space and \mathbf{f} is some known source term. Formally, the solution to (1) can be written explicitly using Duhamel's formula:

$$\mathbf{u}(\mathbf{x}, t + \Delta t) = e^{L\Delta t} \mathbf{u}(\mathbf{x}, t) + \int_0^{\Delta t} e^{L(\Delta t-s)} \mathbf{f}(\mathbf{x}, t+s) ds. \quad (2)$$

MSC2010: primary 65M12, 65M80; secondary 65D05, 65D07, 78M25.

Keywords: Maxwell's equations, Green's function, high order.

This is the starting point for a broad class of time-integration schemes, known as exponential integrators [10], that use (2) to eliminate stiff terms contained in L , which, if treated explicitly, would impose unnecessary and undesirable time-step constraints. This constraint is removed by applying (2) to the spatially discretized equations and evaluating the action of the matrix corresponding to $e^{\eta L}$ on a vector using fast matrix-free methods. Such methods eliminate the stability constraint corresponding to the fast time scales in L .

In the present work, we use (2) as a starting point for eliminating the speed-of-light CFL stability condition in solving Maxwell's equations by explicitly discretizing an integral form of the propagator $e^{\eta L}$ for the original system of PDEs. This type of approach has been proposed previously in [2] and further examined in [13; 9]. In our approach, we use a Helmholtz decomposition to treat the divergence-free and curl-free parts of the solution separately, which allows us to express the propagator in terms of convolutions with weighted delta distributions over the sphere $|\mathbf{x}| = c\eta$, where c is the speed of light. We then discretize (2) in space by replacing the delta distributions with regularized approximate delta distributions defined on a rectangular grid, using the ideas in [19]. This leads to approximations of any order of spatial accuracy of the continuous propagator by discrete convolution operators on a rectangular grid. The discrete kernel satisfies a form of finite propagation speed; i.e., its support is contained in a bounded set of grid points of radius $\mathcal{O}(\sigma + P)$, where σ is the CFL number for the speed of light and P is the order of accuracy of the spatial approximation. This naturally leads to a domain-decomposition formulation of the problem, in which the convolution over the entire domain is replaced with a collection of convolutions over small patches that cover the domain. Due to boundedness of the support of the discrete propagator kernel, the resulting parallel application of the propagator is independent of the decomposition into patches. Finally, the evaluation of the time integral in (2) is approximated by quadratures, and the discrete convolutions are evaluated using FFTs with Hockney's method [11, pp. 180–181]. This method is closely related to the domain decomposition in [20] but differs from that method in that the starting point for our method is a discretization of a real-space propagator while the approach in [20] discretizes a propagator in Fourier space. We will discuss the relative merits of the two approaches in Section 6.

The remainder of this paper is organized as follows. In Section 2, we formalize our problem statement and present a high-level outline of our algorithm and its various components. In Section 3, we describe the discretization process in detail for a comprehensive presentation of each component of the algorithm. In Section 4, we perform a stability analysis of our procedure showing that under certain assumptions the von Neumann stability condition is satisfied. In Section 5, we present a number of numerical tests that show an implementation of our algorithm in action as

applied to the free-space Maxwell equations. Finally, in [Section 6](#), we make some concluding remarks.

2. Problem statement and derivation of propagators

Here, and in what follows, functions of space and/or time will frequently be written omitting their explicit spatial and temporal dependencies, e.g., $\psi = \psi(\mathbf{x}, t)$. We will also use the operator notation $\mathcal{P}^t(\mathbf{u}) \equiv e^{Lt}(\mathbf{u})$. In this notation, (2) is written as

$$\mathbf{u}(t + \Delta t) = \mathcal{P}^{\Delta t}(\mathbf{u}_0) + \int_0^{\Delta t} \mathcal{P}^{\Delta t-s}(\mathbf{f}_s) ds, \quad (3)$$

where $f_s(t) \equiv f(t + s)$.

2.1. The scalar wave equation propagator. To illustrate our approach, we will first derive a real-space propagator for the 3-D wave equation,

$$\begin{aligned} \frac{\partial^2 \phi}{\partial t^2} &= \Delta \phi, \\ \phi(\mathbf{x}, 0) &= \phi_0(\mathbf{x}), \quad \frac{\partial \phi(\mathbf{x}, 0)}{\partial t} = \psi_0(\mathbf{x}). \end{aligned} \quad (4)$$

We introduce unknowns $\mathbf{v} \equiv \nabla \phi$ and $p \equiv \frac{\partial \phi}{\partial t}$, permitting us to recast (4) as a first-order hyperbolic system for \mathbf{v} and p with initial-value constraints, i.e.,

$$\begin{aligned} \frac{\partial \mathbf{v}}{\partial t} &= \nabla p, & \frac{\partial p}{\partial t} &= \nabla \cdot \mathbf{v}, \\ \mathbf{v}(\mathbf{x}, 0) &= \mathbf{v}_0(\mathbf{x}) \equiv \nabla \phi_0(\mathbf{x}), & p(\mathbf{x}, 0) &= p_0(\mathbf{x}) \equiv \psi_0(\mathbf{x}). \end{aligned} \quad (5)$$

Because $\mathbf{v}_0 = \nabla \phi$ is the gradient of some function, we see that $\nabla \times \mathbf{v}_0 \equiv 0$. This implies that $\nabla \times \mathbf{v}_t = 0$ for all time $t > 0$. The curl-free constraint on \mathbf{v}_0 is a necessary and sufficient condition for the first-order system (5) to be equivalent to (4) as it is necessary for v_0 to be curl-free so that the second-order equation can be recovered from the first-order system.

Taking the Fourier transform in \mathbf{x} , we obtain

$$\frac{\partial}{\partial t} \begin{bmatrix} \tilde{\mathbf{v}}(\mathbf{k}, t) \\ \tilde{p}(\mathbf{k}, t) \end{bmatrix} = \begin{bmatrix} 0 & i\mathbf{k} \\ i\mathbf{k}^T & 0 \end{bmatrix} \begin{bmatrix} \tilde{\mathbf{v}}(\mathbf{k}, t) \\ \tilde{p}(\mathbf{k}, t) \end{bmatrix}, \quad (6)$$

where we interpret the Fourier variable \mathbf{k} as a column vector. The operator exponential of this system matrix scaled by Δt is $\tilde{\mathcal{P}}_W^{\Delta t}$, the Fourier transform of our desired propagator. Since we need consider only curl-free \mathbf{v} , we see that $\tilde{\mathcal{P}}_W^{\Delta t}$ can be written

$$\tilde{\mathcal{P}}_W^{\Delta t} = \begin{bmatrix} \cos|\mathbf{k}|\Delta t & i\mathbf{k}(\sin|\mathbf{k}|\Delta t)/|\mathbf{k}| \\ i\mathbf{k}^T(\sin|\mathbf{k}|\Delta t)/|\mathbf{k}| & \cos|\mathbf{k}|\Delta t \end{bmatrix}. \quad (7)$$

Note that, in full generality, the top-left block of $\widehat{\mathcal{P}}_W^{\Delta t}$ has terms involving $\widehat{\mathbf{k}}\widehat{\mathbf{k}}^T$ and $I - \widehat{\mathbf{k}}\widehat{\mathbf{k}}^T$, but the block reduces to $\cos|\mathbf{k}|\Delta t$ when restricted to curl-free input.

Taking an inverse Fourier transform and defining the kernels $G^{\Delta t}$ and $H^{\Delta t}$ via

$$G^{\Delta t}(\mathbf{z}) \equiv \frac{\delta(|\mathbf{z}| - \Delta t)}{4\pi\Delta t}, \quad (8)$$

$$H^{\Delta t}(\mathbf{z}) \equiv \frac{\partial}{\partial s} \left(\frac{\delta(|\mathbf{z}| - s)}{4\pi s} \right) \Big|_{s=\Delta t}, \quad (9)$$

we see that the action of the propagator on an arbitrary state vector $\mathbf{h}(\mathbf{x}) \equiv [\mathbf{f}(\mathbf{x}) \ g(\mathbf{x})]^T$ with \mathbf{f} curl-free is given by

$$\mathcal{P}_W^{\Delta t}(\mathbf{h}) = \begin{bmatrix} H^{\Delta t} * \mathbf{f} + G^{\Delta t} * \nabla g \\ G^{\Delta t} * (\nabla \cdot \mathbf{f}) + H^{\Delta t} * g \end{bmatrix}, \quad (10)$$

where convolutions are defined spatially as

$$(K * f)(\mathbf{x}) \equiv \int_{\mathbb{R}^3} K(\mathbf{y}) f(\mathbf{x} - \mathbf{y}) d\mathbf{y} \quad (11)$$

and convolution of a scalar quantity with a vector quantity is taken componentwise. Considering again (3) and noting the lack of sources, we see that we can obtain the solution to (5) for any final time $t_{\text{final}} = \Delta t$ by evaluating

$$\begin{bmatrix} \mathbf{v}(\Delta t) \\ p(\Delta t) \end{bmatrix} = \begin{bmatrix} H^{\Delta t} * \mathbf{v}_0 + G^{\Delta t} * \nabla p_0 \\ G^{\Delta t} * (\nabla \cdot \mathbf{v}_0) + H^{\Delta t} * p_0 \end{bmatrix}. \quad (12)$$

Here, the absence of sources obviates the need to treat the time integral in (3) and therefore reduces the problem entirely to discretely applying (10). We note that (10) can be derived from the classical solution starting with the second-order formulation as seen in [22]. However, we outline the approach starting with the first-order system as an analog to Maxwell's equations.

We see that for application of (10) it is necessary to approximate convolution against the singular kernels $G^{\Delta t}$ and $H^{\Delta t}$. To make $H^{\Delta t}$ more amenable to approximation, we use some calculus to reduce convolutions against $H^{\Delta t}$ to convolutions against $G^{\Delta t}$ combined with weights and spatial derivatives. We begin with the Fourier relationship

$$\frac{\partial}{\partial s} \left(\frac{\delta(|\mathbf{x}| - s)}{4\pi s} \right) = \mathcal{F}^{-1}[\cos|\mathbf{k}|s]. \quad (13)$$

It is not difficult to verify that if we write \mathbf{x} and \mathbf{k} in terms of their components then we have

$$\cos|\mathbf{k}|s = \frac{\sin|\mathbf{k}|s}{|\mathbf{k}|s} - i \sum_{d=1}^3 \frac{\partial}{\partial k_d} \left(\frac{\sin|\mathbf{k}|s}{|\mathbf{k}|s} \right) i k_d. \quad (14)$$

Defining the new convolutional kernels $G_d^{\Delta t}$ for $d = 1, 2, 3$ via

$$G_d^{\Delta t}(\mathbf{z}) \equiv \frac{z_d \delta(|\mathbf{z}| - \Delta t)}{4\pi \Delta t}, \quad (15)$$

i.e., convolution in space against the weighted distribution $x_d \delta(|\mathbf{x}| - \Delta t)/(4\pi)$, we see from standard rules of Fourier analysis that (14) is the Fourier transform of the operator that acts on a function $f : \mathbb{R}^3 \mapsto \mathbb{R}$ via

$$(H^{\Delta t} * f) = \frac{1}{\Delta t} G^{\Delta t} * f - \sum_{d=1}^3 G_d^{\Delta t} * \frac{\partial f}{\partial z_d}. \quad (16)$$

2.2. The Maxwell propagator. Similarly to Section 2.1, we can derive an expression for the propagator for the solution of Maxwell's equations written in terms of spatial derivatives and convolutions. We begin by writing the set of Maxwell's equations for $(\mathbf{x}, t) \in \mathbb{R}^3 \times \mathbb{R}_+$ as

$$\frac{\partial \mathbf{E}}{\partial t} = c \nabla \times \mathbf{B} - \mathbf{J}, \quad \frac{\partial \mathbf{B}}{\partial t} = -c \nabla \times \mathbf{E}, \quad (17)$$

$$\nabla \cdot \mathbf{E} = \rho, \quad \nabla \cdot \mathbf{B} = 0, \quad (18)$$

with appropriate initial conditions. Here, \mathbf{E} and \mathbf{B} are the electric and magnetic fields, respectively, \mathbf{J} is a known current source term, ρ is the bound current density, and c is the speed of light in vacuum.

To find a solution for Maxwell's equations, we first use a Helmholtz decomposition to break the electric field and current source into their longitudinal (curl-free) and transverse (divergence-free) parts

$$\mathbf{E} = \mathbf{E}_L + \mathbf{E}_T, \quad \mathbf{J} = \mathbf{J}_L + \mathbf{J}_T, \quad (19)$$

where $\nabla \times \mathbf{E}_L \equiv 0$ and $\nabla \cdot \mathbf{E}_T \equiv 0$ and similarly for \mathbf{J}_L and \mathbf{J}_T . This decomposition leads to a first-order system of hyperbolic PDEs describing the coupling between \mathbf{E}_T and \mathbf{B} (see (17))

$$\frac{\partial \mathbf{E}_T}{\partial t} = c \nabla \times \mathbf{B} - \mathbf{J}_T, \quad \frac{\partial \mathbf{B}}{\partial t} = -c \nabla \times \mathbf{E}_T, \quad (20)$$

$$\nabla \cdot \mathbf{E}_T = 0, \quad \nabla \cdot \mathbf{B} = 0. \quad (21)$$

The divergence-free conditions (21) are initial-value constraints, similar to the curl-free constraint on v_0 for the wave equation, and if satisfied at time $t = 0$, then they are satisfied at all later times according to (20). The longitudinal component of the electric field \mathbf{E}_L can be specified either in terms of Coulomb's law,

$$\mathbf{E}_L = -\nabla \phi, \quad \nabla \cdot \mathbf{E}_L = \rho, \quad (22)$$

or directly from applying the Helmholtz decomposition to the evolution equation for \mathbf{E} ,

$$\frac{\partial \mathbf{E}_L}{\partial t} = -\mathbf{J}_L. \quad (23)$$

These two specifications of \mathbf{E}_L are equivalent, provided that, at time $t = 0$, (22) is satisfied. In practice, the choice of which of these two formulations to use in discretizing \mathbf{E}_L depends on the details of how the evolution of ρ is specified. The risk is that, by using (23), accumulation of numerical error will cause (22) not to be satisfied. We will not address this issue here other than to note that the formulation given here will require the solution of Poisson's equation at least to compute the Helmholtz decomposition of \mathbf{J} and possibly to solve (22). For those problems, we can use fast Poisson solvers, the cost of which will be made up for by the ability to take larger time steps. Therefore, for the purposes of demonstrating the properties of the method described here, we will consider only examples in which $\mathbf{E}_L \equiv 0$ and $\mathbf{J}_L \equiv 0$. Given this and using reasoning similar to that in Section 2.1, we obtain the action of the Maxwell propagator on a state vector $\mathbf{h}(\mathbf{x}) = [\mathbf{E}_T(\mathbf{x}) \ \mathbf{B}(\mathbf{x})]^T$,

$$\mathcal{P}_M^{\Delta t}(\mathbf{h}) = \begin{bmatrix} H^{c\Delta t} * \mathbf{E}_T + G^{c\Delta t} * (\nabla \times \mathbf{B}) \\ -G^{c\Delta t} * (\nabla \times \mathbf{E}_T) + H^{c\Delta t} * \mathbf{B} \end{bmatrix}, \quad (24)$$

from which the action on a divergence-free current source can be inferred. Contrary to the source-free case we saw before, the appearance of \mathbf{J}_T in (20) will require the treatment of the integral in (3) to obtain the full solution.

3. Discretization

As seen in Section 2, the wave equation and Maxwell propagators involve convolution against kernels taking the form of (possibly weighted) delta distributions supported on spheres. For example, convolution against the kernel $G_d^{\Delta t}$ in (15) is given by

$$(G_d^{\Delta t} * w)(\mathbf{x}) \equiv \int \frac{z_d}{4\pi \Delta t} \delta(|\mathbf{z}| - \Delta t) w(\mathbf{x} - \mathbf{z}) d\mathbf{z} = \int_{\partial B_{\Delta t}} \frac{z_d}{4\pi \Delta t} w(\mathbf{x} - \mathbf{z}) d\mathbf{z}, \quad (25)$$

where $\partial B_{\Delta t}$ is the 3-D sphere of radius Δt . Rewriting these convolutions as integrals over singular surfaces as we have done above, we see that accurately computing these convolutions reduces to accurately evaluating integrals of the form

$$I \equiv \int_{\Gamma} g(\mathbf{z}) f(\mathbf{z}) d\mathbf{z}, \quad (26)$$

where Γ is a continuous and bounded surface and $g \in C(\mathbb{R}^d)$ is a weighting function.

Accurate discretization on a Cartesian grid of the integral I is treated succinctly by Tornberg and Engquist [19], who summarize a framework for replacing such

integrals with sums of samples of a regularized integrand based on work concerning singular source terms by Beyer and LeVeque [4]. We describe this in more detail below.

3.1. Regularized delta distributions. As a precursor to integration over multi-dimensional surfaces, consider integrating a function $f : \mathbb{R} \rightarrow \mathbb{R}$ against a 1-D delta distribution with arbitrary center $\bar{x} \in \mathbb{R}$, i.e., evaluating $f(\bar{x})$ via the sifting formula

$$f(\bar{x}) = \int_{\mathbb{R}} f(x)\delta(x - \bar{x}) dx. \tag{27}$$

The above integral can be thought of as integrating $f(x)$ over a singular surface of dimension 0, supported at the single point \bar{x} . Because of its singular nature, simply sampling $f(x)\delta(x - \bar{x})$ on a grid and approximating integration with summation is not a numerically well defined operation. Instead, given a grid spacing h , we consider sampling a *regularized* approximant $\delta_h(x - \bar{x})$. Beyer and LeVeque [4] introduce the set of discrete moment conditions for such an approximant, summarized succinctly in [19].

Definition 1 (discrete moment conditions [4; 19]). Given a grid spacing $h > 0$, we say a function $\delta_h : \mathbb{R} \rightarrow \mathbb{R}$ is in the function class Q^q if δ_h has compact support $[-mh, mh]$ for some $m > 0$ and

$$h \sum_{j \in \mathbb{Z}} (jh - \bar{x})^r \delta_h(jh - \bar{x}) = \begin{cases} 1, & r = 0, \\ 0, & 1 \leq r < q, \end{cases} \tag{28}$$

for any $\bar{x} \in \mathbb{R}$.

We note that the conditions in **Definition 1** are simply discrete analogues of the continuous moment conditions

$$\int_{-\infty}^{\infty} (x - \bar{x})^r \delta(x - \bar{x}) dx = \begin{cases} 1, & r = 0, \\ 0, & 1 \leq r < q, \end{cases} \tag{29}$$

which are satisfied by the delta distribution for arbitrarily large q .

For sufficiently regular functions f , the discrete moment conditions are sufficient for consistency of δ_h ; i.e., for $\delta_h \in Q^q$ and $f \in C^q$, we have the asymptotic error bound

$$f(\bar{x}) - h \sum_{j \in \mathbb{Z}} f(jh)\delta_h(jh - \bar{x}) = \mathcal{O}(h^q) \tag{30}$$

for any $\bar{x} \in \mathbb{R}$ [4]. In the **Appendix**, we give a review of how to find such 1-D approximants satisfying the discrete moment conditions to order q while attaining the minimum necessary support, which is essentially accomplished by piecewise Lagrange interpolation.

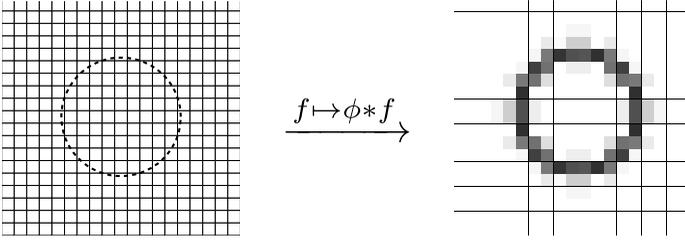


Figure 1. Left: a 2-D slice of the spherically supported delta distribution, where the distribution takes the value $+\infty$ on the dashed circle and is identically zero elsewhere. Right: with regularization, the support of the distribution is smoothed such that the distribution takes finite values and can be sampled on the underlying discrete grid.

To extend these 1-D ideas to multiple dimensions, we use the tensor product formulation of Peskin [16], which obeys the following consistency result.

Theorem 2 (consistency of multidimensional discrete deltas [19]). *Let Γ be a continuous and bounded surface, $g \in C(\mathbb{R}^d)$, and $\delta_{h_k} \in Q^q$ for $k = 1, \dots, d$. Define the multidimensional function*

$$\delta_h(\Gamma, g, \mathbf{x}) \equiv \int_{\Gamma} \prod_{k=1}^d \delta_{h_k}(x_k - z_k) g(\mathbf{z}) d\mathbf{z}, \quad (31)$$

where $\mathbf{x} = (x_1, \dots, x_d) \in \mathbb{R}^d$ and $\mathbf{z} = (z_1, \dots, z_d) \in \Gamma$. Suppose $f \in C^r(\mathbb{R}^d)$. Then

$$\left(\prod_{k=1}^d h_k \right) \sum_{\mathbf{j} \in \mathbb{Z}^d} \delta_h(\Gamma, g, \mathbf{x}_{\mathbf{j}}) f(\mathbf{x}_{\mathbf{j}}) - \int_{\Gamma} g(\mathbf{z}) f(\mathbf{z}) d\mathbf{z} = \mathcal{O}(h^q), \quad (32)$$

where $\mathbf{x}_{\mathbf{j}} = (j_1 h_1, \dots, j_d h_d)$ is a Cartesian grid point and $h = \max_{k=1, \dots, d} h_k$.

Intuitively, **Theorem 2** gives a method to evaluate integrals of the form (26) by regularizing the singular surface via convolution with the multidimensional smoothing kernel δ_h . For example, with $\Gamma = \partial B_{\Delta t}$ and $g(\mathbf{z}) \equiv 1$, we see that

$$\delta_h(\partial B_{\Delta t}, 1, \mathbf{x}) = \delta(|\mathbf{x}| - \Delta t) * \left(\prod_{k=1}^d \delta_{h_k}(x_k - z_k) \right), \quad (33)$$

so **Theorem 2** essentially permits accurate discrete convolution against $\delta(|\mathbf{x}| - \Delta t)$ by presmoothing the sphere with the kernel $\phi(\mathbf{x}) = \prod_{k=1}^d \delta_{h_k}(x_k)$ prior to sampling; see **Figure 1**. This is the key step permitting accurate discretization of the singular convolutional operators comprising our propagators.

3.2. Spherical quadrature. Discretization of the convolutional operators $G^{\Delta t}$ and $G_d^{\Delta t}$ necessitates the generation of samples of the function $\delta_h(\Gamma, g, \mathbf{x})$ in (31), where $\Gamma = \partial B_{\Delta t}$ and the weighting function g is defined by either $g(\mathbf{z}) \equiv 1$ or $g(\mathbf{z}) = z_i$,

depending on the kernel. In practice, we find that the integral in (31) has no simple analytical solution and therefore must be replaced with some form of quadrature scheme. We use the product Gaussian quadrature described in [3],

$$\begin{aligned} \delta_h(\partial B_{\Delta t}, g, \mathbf{x}) &= \int_{\partial B_{\Delta t}} \left(\prod_{k=1}^d \delta_{h_k}(x_k - z_k) g(\mathbf{z}) \right) d\mathbf{z}, \\ &\approx \frac{\pi(\Delta t)^2}{m} \sum_{j=1}^{2m} \sum_{i=1}^m w_i \left(\prod_{k=1}^d \delta_{h_k}(x_k - z_{ij,k}) g(\mathbf{z}_{ij}) \right), \end{aligned} \quad (34)$$

where \mathbf{z}_{ij} has polar coordinates $(\Delta t, \theta_i, \phi_j)$ with $\cos \theta_i$ and w_i the Gauss–Legendre nodes and weights on $[-1, 1]$ and ϕ_j evenly spaced on $[0, 2\pi]$.

We note that, typically, such numerical quadratures use assumptions on the smoothness of the integrand to prove convergence whereas the smoothness of the integrand in (31) is dependent on the smoothness of δ_{h_k} . However, we are not interested in the intermediate error in evaluating $\delta_h(\Gamma, g, \mathbf{x})$ on a grid but rather the operator error of $\delta_h(\Gamma, g, \mathbf{x})$ as a discrete convolutional operator when applied to a smooth function f . Up to machine precision, the numerical quadrature and the discrete convolution commute, and we thus achieve accuracy from our assumptions on the smoothness of f rather than that of δ_h .

3.3. Final construction of operators. Given a spline approximation to the 1-D delta distribution obeying moment conditions up to order q , we use the results from the previous section to construct discretizations $G^{\Delta t, h}$ and $H^{\Delta t, h}$ of the convolutional kernels $G^{\Delta t}$ and $H^{\Delta t}$ for a specified time step Δt as follows. Defining $G^{\Delta t, h}$ via

$$G^{\Delta t, h}(\mathbf{x}_i) \equiv \frac{1}{4\pi \Delta t} \left(\prod_{k=1}^d h_k \right) \delta_h(\partial B_{\Delta t}, 1, \mathbf{x}_i), \quad (35)$$

we see from Theorem 2 that the approximation $G^{\Delta t} * f \approx G^{\Delta t, h} * f_h$ is pointwise accurate to order q , where the second convolution is understood as discrete convolution of $G^{\Delta t, h}$ with f sampled on a Cartesian grid. As discussed in Section 3.2, we use product Gaussian quadrature to evaluate the spherical integral necessary to construct the discrete delta distribution. The accuracy of this scheme thus necessarily depends on the number of quadrature nodes used, but this can be taken to be sufficiently high since $G^{\Delta t, h}$ need be constructed only once as a precomputation. In practice, we find that the number of quadrature nodes is not a limiting factor; see Section 5.1.

To construct the discrete kernel $H^{\Delta t, h}$, we note from (16) that we require a discrete approximation of the weighted kernels $G_d^{\Delta t} = z_d \delta(|z| - \Delta t)$ for $d = 1, 2, 3$, which we construct similarly to the kernel in (35) by taking $g(\mathbf{z}) = z_d$. Then, approximating the spatial derivatives by precomposing the discrete kernels $G_d^{\Delta t, h}$

```

Initialize  $p_h^{(0)}$  and  $\mathbf{v}_h^{(0)}$ 
Compute  $G^{\Delta t, h}$ , and  $H^{\Delta t, h}$  based on step size in time and space
/* Begin time-stepping loop */
for  $n = 1, 2, \dots$ 
  /* Update velocities */
   $v_{x,h}^{(n)} \leftarrow H^{\Delta t, h} * v_{x,h}^{(n-1)} + (G^{\Delta t, h} * \Delta_x) * p_h^{(n-1)}$ 
   $v_{y,h}^{(n)} \leftarrow H^{\Delta t, h} * v_{y,h}^{(n-1)} + (G^{\Delta t, h} * \Delta_y) * p_h^{(n-1)}$ 
   $v_{z,h}^{(n)} \leftarrow H^{\Delta t, h} * v_{z,h}^{(n-1)} + (G^{\Delta t, h} * \Delta_z) * p_h^{(n-1)}$ 
  /* Update pressure */
   $p_h^{(n)} \leftarrow H^{\Delta t, h} * p_h^{(n-1)} + (G^{\Delta t, h} * \Delta_x) * v_{x,h}^{(n-1)} + (G^{\Delta t, h} * \Delta_y) * v_{y,h}^{(n-1)} + (G^{\Delta t, h} * \Delta_z) * v_{z,h}^{(n-1)}$ 
end for

```

Algorithm 1. Applying the wave equation propagator.

with corresponding finite difference stencils Δ_{x_d} , we obtain $H^{\Delta t, h}$ as

$$H^{\Delta t, h} \equiv \frac{1}{\Delta t} G^{\Delta t, h} - \sum_{d=1}^3 G_d^{\Delta t, h} * \Delta_{x_d} \quad (36)$$

using the equivalent expression (16). We choose central difference stencils accurate to order q for consistency. For example, if $q = 4$, then we use the typical fourth-order central difference [7]

$$(\Delta_{x_d} * f_h)(\mathbf{x}_i) = \frac{\frac{1}{12} f_h(\mathbf{x}_i - 2\mathbf{e}_d) - \frac{2}{3} f_h(\mathbf{x}_i - \mathbf{e}_d) + \frac{2}{3} f_h(\mathbf{x}_i + \mathbf{e}_d) - \frac{1}{12} f_h(\mathbf{x}_i + 2\mathbf{e}_d)}{h}, \quad (37)$$

where \mathbf{e}_d is the d -th unit coordinate vector.

We note that, just as the continuous kernels $G^{\Delta t}$ and $H^{\Delta t}$ are compactly supported, so too are the discrete kernels with only slightly larger support size dependent on the exact smoothing splines and finite difference stencils used.

3.4. A typical time step. For source-free applications, the Maxwell and wave equation propagators can in theory be constructed for $\Delta t = t_{\text{final}}$ and applied as a one-step method. However, for many applications of interest, such as particle-in-cell (PIC) methods, it is necessary to evolve the solution only by a small time increment so that sources can be computed and incorporated. Here we describe the computation loop for time-stepping using the discrete kernels $G^{\Delta t, h}$ and $H^{\Delta t, h}$ described in Section 3.3. Consider first the wave equation propagator of (10). Then, the solution of the source-free problem (5) is obtained using the time-stepping process in Algorithm 1. We use $v_{x,h}$, $v_{y,h}$, and $v_{z,h}$ to refer to the discrete representations of components of \mathbf{v} and Δ_x , Δ_y , and Δ_z to refer to finite difference operators in each coordinate direction as defined in Section 3.3. As mentioned, the discrete kernels $G^{\Delta t, h}$ and $H^{\Delta t, h}$ are compactly supported. This admits the use of Hockney’s domain-doubling method [11] to evaluate all discrete convolutions efficiently.

```

/* Electric field and source are transverse throughout */
Initialize  $E_h^{(0)}$  and  $B_h^{(0)}$ 
Initialize Newton–Cotes quadrature weights  $\{w_m\}_{m=0}^M$ 
Compute  $G^{c\Delta s, h}$  and  $H^{c\Delta s, h}$  based on step size in time and space
/* Begin time-stepping loop */
for  $n = 1, 2, \dots$ 
  /* Initialize for Newton–Cotes */
   $E_h^{(n)} \leftarrow E_h^{(n-1)}$ 
   $B_h^{(n)} \leftarrow B_h^{(n-1)}$ 
  for  $m = 0, 1, \dots, M-1$ 
    /* Add source term for node  $t_{n,m} = (n-1)\Delta t + m\Delta s$  */
     $E_h^{(n)} \leftarrow E_h^{(n)} - w_m J_h(t_{n,m})$ 
    /* Update electric field */
     $E_{x,h}^{(n)} \leftarrow H^{c\Delta s, h} * E_{x,h}^{(n)} + (G^{c\Delta s, h} * \Delta_y) * B_z^{(n)} - (G^{c\Delta s, h} * \Delta_z) * B_y^{(n)}$ 
     $E_{y,h}^{(n)} \leftarrow H^{c\Delta s, h} * E_{y,h}^{(n)} + (G^{c\Delta s, h} * \Delta_z) * B_x^{(n)} - (G^{c\Delta s, h} * \Delta_x) * B_z^{(n)}$ 
     $E_{z,h}^{(n)} \leftarrow H^{c\Delta s, h} * E_{z,h}^{(n)} + (G^{c\Delta s, h} * \Delta_x) * B_y^{(n)} - (G^{c\Delta s, h} * \Delta_y) * B_x^{(n)}$ 
    /* Update magnetic field */
     $B_{x,h}^{(n)} \leftarrow H^{c\Delta s, h} * B_{x,h}^{(n)} - (G^{c\Delta s, h} * \Delta_y) * E_{z,h}^{(n)} + (G^{c\Delta s, h} * \Delta_z) * E_{y,h}^{(n)}$ 
     $B_{y,h}^{(n)} \leftarrow H^{c\Delta s, h} * B_{y,h}^{(n)} - (G^{c\Delta s, h} * \Delta_z) * E_{x,h}^{(n)} + (G^{c\Delta s, h} * \Delta_x) * E_{z,h}^{(n)}$ 
     $B_{z,h}^{(n)} \leftarrow H^{c\Delta s, h} * B_{z,h}^{(n)} - (G^{c\Delta s, h} * \Delta_x) * E_{y,h}^{(n)} + (G^{c\Delta s, h} * \Delta_y) * E_{x,h}^{(n)}$ 
  end for
  /* Final Newton–Cotes node */
   $E_h^{(n)} \leftarrow E_h^{(n)} - w_M J_h(t_{n,M})$ 
end for

```

Algorithm 2. Applying the Maxwell propagator.

Application of the transverse Maxwell propagator for (20) is analogous to Algorithm 1 in the source-free case. In the presence of sources, however, we must combine the basic flavor of the previous algorithm with a time-integration scheme for treating the integral in (3). To accomplish this, we use a closed Newton–Cotes method in time of the appropriate order with equispaced nodes $\{s_m = m\Delta s\}_{m=0}^M$ ($\Delta s = \Delta t/M$):

$$\int_0^{\Delta t} \mathcal{P}^{\Delta t-s}(f_s) ds \approx \sum_{m=0}^M w_m \mathcal{P}^{\Delta t-s_m}(f_{s_m}) = \sum_{m=0}^M w_m \mathcal{P}^{(M-m)\Delta s}(f_{s_m}) \quad (38)$$

$$= \sum_{m=0}^M w_m \overbrace{\mathcal{P}^{\Delta s}(\mathcal{P}^{\Delta s}(\dots \mathcal{P}^{\Delta s}(f_{s_m}) \dots))}^{M-m \text{ times}}, \quad (39)$$

where, because the quadrature nodes are equispaced in time, we may make use of the fact that, analytically, application of the propagator $\mathcal{P}^{(M-m)\Delta s}$ corresponding to

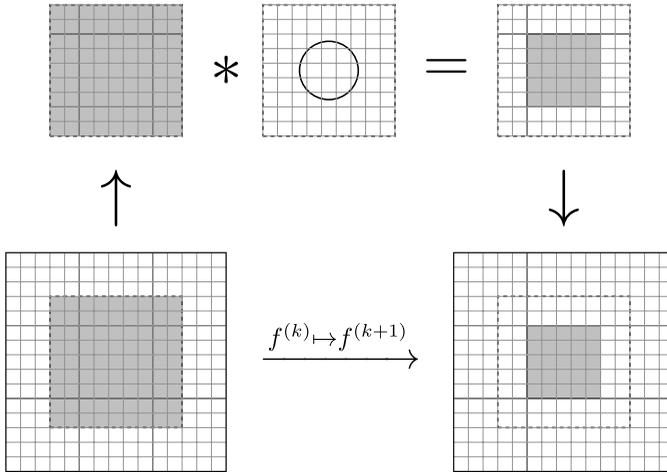


Figure 2. Suppose that the domain is decomposed across a 3×3 grid of processors such that the central processor owns the central block of unknowns as well as the halo region. Assuming the halo region is valid and sufficiently large, the local field values for time $n + 1$ can be calculated by convolving the current local-plus-halo field values with the propagator. In this figure, cells are shaded to show that the values of the corresponding unknowns are correct for the current time on the central processor.

advancing the solution in time by $(M - m)\Delta s$ is equivalent to $M - m$ successive applications of the single-step propagator $\mathcal{P}^{\Delta s}$, necessitating construction of only a single discrete propagator. This leads to [Algorithm 2](#) for the transverse Maxwell propagator with transverse current source \mathbf{J} .

Because we are considering using the short-time propagators in a time-stepping loop, we have to consider the stability properties of the repeated application of these propagators. We discuss this in [Section 4](#). While we do not discuss it in detail here, the number of nodes in the composite Newton–Cotes scheme described in (38) can also affect stability.

3.5. Parallelization. The use of a compactly supported convolutional kernel to regularize the 3-D delta distributions inherent in the propagators of (10) and (24) has the computational benefit of numerically preserving the locality inherent in the wave equation. In particular, in the same vein as Vay et al. [20], we can use standard domain decomposition to parallelize the time-stepping procedure as in [Figure 2](#). This is in contrast to methods that use spectral expansions or global Fourier transforms to obtain high accuracy.

Parallelization of our scheme follows the traditional communication-computation loop of standard finite difference schemes:

- (1) copy field values in halo region from neighboring processors, and then
- (2) apply propagator to update local field values, invalidating values in halo region.

Note that the width of the halo region here has a minimum bound dictated by the size of the time step Δt since the size of the support of the spherical delta distributions is dependent on how far in time the fields are to be advanced. For example, in the presence of sources as in [Algorithm 2](#), the size of the halo region should be such that communication is only necessary after M applications of the discrete propagator constructed with step size $\Delta s = \Delta t/M$. Thus, Δt is chosen based on the desired size of the halo region and Δs , which determines the CFL number, is chosen based on the necessary resolution to resolve variation in the source.

4. Stability analysis

Letting $\mathcal{P}_M^{\Delta s, h}$ denote the discrete Maxwell propagator described by [Algorithm 2](#), we see that the evolution of the discretized electromagnetic fields is given by

$$\begin{bmatrix} \mathbf{E}_h^{(n)} \\ \mathbf{B}_h^{(n)} \end{bmatrix} = \mathcal{P}_M^{\Delta s, h} \left(\begin{bmatrix} \mathbf{E}_h^{(n-1)} \\ \mathbf{B}_h^{(n-1)} \end{bmatrix} \right).$$

Taking a Fourier transform, we obtain the Fourier-space relation

$$\begin{bmatrix} \tilde{\mathbf{E}}_h^{(n)} \\ \tilde{\mathbf{B}}_h^{(n)} \end{bmatrix} = \tilde{\mathcal{P}}_M^{\Delta s, h} \begin{bmatrix} \tilde{\mathbf{E}}_h^{(n-1)} \\ \tilde{\mathbf{B}}_h^{(n-1)} \end{bmatrix},$$

where now $\tilde{\mathcal{P}}_M^{\Delta s, h}$ is a matrix in \mathbf{k} space (as opposed to a sum of convolutional operators in physical space). For a typical von Neumann analysis of ℓ_2 stability, we define $\rho(\mathbf{k})$ to be the spectral radius of $\tilde{\mathcal{P}}_M^{\Delta s, h}$ and show that the necessary condition (see, e.g., [\[17\]](#))

$$\rho(\mathbf{k}) \leq 1 + \mathcal{O}(\Delta s)$$

holds for all \mathbf{k} . For analysis purposes, we assume that the integrals used to construct the necessary regularized, spherically supported delta distributions are computed exactly, i.e., without the use of the quadrature described in [Section 3.2](#). Then, direct computation shows that $\tilde{\mathcal{P}}_M^{\Delta s, h}(\mathbf{k})$ is given by

$$\tilde{\mathcal{P}}_M^{\Delta s, h} = \begin{bmatrix} \tilde{H}^{c\Delta s, h} & 0 & 0 & 0 & -\tilde{G}^{c\Delta s, h} \tilde{\Delta}_z & \tilde{G}^{c\Delta s, h} \tilde{\Delta}_y \\ 0 & \tilde{H}^{c\Delta s, h} & 0 & \tilde{G}^{c\Delta s, h} \tilde{\Delta}_z & 0 & -\tilde{G}^{c\Delta s, h} \tilde{\Delta}_x \\ 0 & 0 & \tilde{H}^{c\Delta s, h} & -\tilde{G}^{c\Delta s, h} \tilde{\Delta}_y & \tilde{G}^{c\Delta s, h} \tilde{\Delta}_x & 0 \\ 0 & \tilde{G}^{c\Delta s, h} \tilde{\Delta}_z & -\tilde{G}^{c\Delta s, h} \tilde{\Delta}_y & \tilde{H}^{c\Delta s, h} & 0 & 0 \\ -\tilde{G}^{c\Delta s, h} \tilde{\Delta}_z & 0 & \tilde{G}^{c\Delta s, h} \tilde{\Delta}_x & 0 & \tilde{H}^{c\Delta s, h} & 0 \\ \tilde{G}^{c\Delta s, h} \tilde{\Delta}_y & -\tilde{G}^{c\Delta s, h} \tilde{\Delta}_x & 0 & 0 & 0 & \tilde{H}^{c\Delta s, h} \end{bmatrix},$$

where the ordering of the blocks corresponds to the vector $[\tilde{E}_x \ \tilde{E}_y \ \tilde{E}_z \ \tilde{B}_x \ \tilde{B}_y \ \tilde{B}_z]^T$ and the blocks of $\tilde{\mathcal{P}}_M^{\Delta s, h}$ are functions of the Fourier variable \mathbf{k} corresponding to the transforms of the discrete operators from [Section 3.3](#).

Lemma 3. *The eigenvalues of $\tilde{\mathcal{P}}_M^{\Delta s, h}$ as a function of \mathbf{k} are given by*

$$\begin{aligned}\lambda_{1,2}(\mathbf{k}) &= \tilde{H}^{c\Delta s, h}(\mathbf{k}) \pm \tilde{G}^{c\Delta s, h}(\mathbf{k}) \sqrt{\tilde{\Delta}_x^2(\mathbf{k}) + \tilde{\Delta}_y^2(\mathbf{k}) + \tilde{\Delta}_z^2(\mathbf{k})}, \\ \lambda_3(\mathbf{k}) &= \tilde{H}^{c\Delta s, h}(\mathbf{k}),\end{aligned}$$

each of which appears with multiplicity 2.

Proof. This follows from direct computation via block linear algebra. \square

To obtain explicit expressions for the eigenvalues in Lemma 3, we represent the sampling operator using:

Definition 4. Given the spatial step sizes h_1 , h_2 , and h_3 , the 3-D Dirac comb $\mathbb{I}\mathbb{I}_h$ is defined on \mathbb{R}^3 as

$$\mathbb{I}\mathbb{I}_h(\mathbf{x}) \equiv \sum_{\mathbf{l} \in \mathbb{Z}^3} \left(\prod_{d=1}^3 \delta(x_d - l_d h_d) \right).$$

In other words, $\mathbb{I}\mathbb{I}_h$ is a regular 3-D lattice of delta distributions.

For the methods described in this paper, the regularized 1-D delta distribution used to construct the regularized 3-D delta distributions in $G^{c\Delta s, h}$ and $H^{c\Delta s, h}$ is given by shifting and scaling of some fundamental kernel W , i.e.,

$$\delta_{h_d}(x) \equiv \frac{1}{h_d} W(x/h_d), \quad (40)$$

which we combine with the sampling operator to explicitly write the Fourier transforms $\tilde{G}^{c\Delta s, h}$ and $\tilde{H}^{c\Delta s, h}$ as

$$\tilde{H}^{c\Delta s, h}(\mathbf{k}) = \int_{\mathbb{R}^3} \mathbb{I}\mathbb{I}_{2\pi/h}(\mathbf{k} - \mathbf{k}') \left[\left(\prod_{d=1}^3 \tilde{W}(k'_d h_d) \right) \tilde{P}_1(\mathbf{k}') \right] d\mathbf{k}', \quad (41)$$

$$\tilde{G}^{c\Delta s, h}(\mathbf{k}) = \int_{\mathbb{R}^3} \mathbb{I}\mathbb{I}_{2\pi/h}(\mathbf{k} - \mathbf{k}') \left[\left(\prod_{d=1}^3 \tilde{W}(k'_d h_d) \right) \tilde{P}_2(\mathbf{k}') \right] d\mathbf{k}', \quad (42)$$

where the functions \tilde{P}_1 and \tilde{P}_2 are defined in Fourier space according to

$$\begin{aligned}\tilde{P}_1(\mathbf{k}) &\equiv \frac{\sin c|\mathbf{k}|\Delta s}{c|\mathbf{k}|\Delta s} - i \sum_{d=1}^3 \frac{\partial}{\partial k_d} \left[\frac{\sin c|\mathbf{k}|\Delta s}{c|\mathbf{k}|\Delta s} \right] \tilde{\Delta}_{x_d}(k_d), \\ \tilde{P}_2(\mathbf{k}) &\equiv \frac{\sin c|\mathbf{k}|\Delta s}{|\mathbf{k}|}.\end{aligned}$$

We are now ready to state and prove the fundamental stability result for the discrete Maxwell propagator.

Theorem 5. *In Algorithm 2, assume that the regularized, spherically supported delta distributions are computed exactly, i.e., without the use of quadrature. Suppose further that the 1-D delta distributions are constructed using a fundamental kernel W as in (40) satisfying the following properties:*

- (1) $\tilde{W}(k)$ is real and nonnegative for $k \in \mathbb{R}$.
- (2) $W(j) = 0$ for $j \in \mathbb{Z}$ except $W(0) = 1$; i.e., W behaves like the Kronecker delta on the lattice points.

In addition, suppose that for each d the finite difference stencil Δ_{x_d} has real coefficients and odd symmetry and has spectrum bounded for $|k_d h_d| \in (0, \pi]$ as

$$0 \leq \frac{\tilde{\Delta}_{x_d}(k_d)}{i k_d} \leq 1.$$

Furthermore, define the quantity

$$\tilde{R}(\mathbf{k}) \equiv \sum_{d=1}^3 \frac{-i k_d \tilde{\Delta}_{x_d}(k_d)}{|\mathbf{k}|^2},$$

and assume that there exists a bound $B < 1$ such that $|\tilde{R}(\mathbf{k})| \leq B < 1$ for all \mathbf{k} with any $|k_d| \geq \pi/h_d$. Then the time-stepping scheme satisfies the von Neumann condition for

$$\sigma \equiv \frac{c \Delta s}{h} \geq \frac{1 + B}{\pi(1 - B)},$$

where $h = \max_d h_d$.

Proof. Based on the assumption that Δ_{x_d} is a real and odd finite difference stencil, it has a purely imaginary Fourier transform. Using Lemma 3, we deduce that $\mathcal{P}_M^{\Delta s, h}(\mathbf{k})$ has a spectral radius ρ given by

$$\rho(\mathbf{k}) = \left| \tilde{H}^{c \Delta s, h}(\mathbf{k}) + \tilde{G}^{c \Delta s, h}(\mathbf{k}) \sqrt{\tilde{\Delta}_x^2(\mathbf{k}) + \tilde{\Delta}_y^2(\mathbf{k}) + \tilde{\Delta}_z^2(\mathbf{k})} \right|.$$

Using the triangle inequality and plugging in expressions (41) and (42) yields

$$\rho(\mathbf{k}) \leq \int_{\mathbb{R}^3} \mathbb{I}_{2\pi/h}(\mathbf{k} - \mathbf{k}') \left[\left(\prod_{d=1}^3 \tilde{W}(k'_d h_d) \right) \cdot |\tilde{S}(\mathbf{k}')| \right] d\mathbf{k}'$$

with the quantity $\tilde{S}(\mathbf{k})$ defined according to

$$\tilde{S}(\mathbf{k}) \equiv \tilde{P}_1(\mathbf{k}) + \tilde{P}_2(\mathbf{k}) \sqrt{\tilde{\Delta}_x^2(\mathbf{k}) + \tilde{\Delta}_y^2(\mathbf{k}) + \tilde{\Delta}_z^2(\mathbf{k})}. \tag{43}$$

We will show that \tilde{S} has magnitude bounded by 1 for all k and use this to show that ρ is bounded by 1 for all k .

Using some calculus and our definition of the quantity \tilde{R} , we write the term \tilde{P}_1 as

$$\tilde{P}_1(\mathbf{k}) = \frac{\sin c|\mathbf{k}|\Delta s}{c|\mathbf{k}|\Delta s}(1 - \tilde{R}(\mathbf{k})) + \tilde{R}(\mathbf{k}) \cos c|\mathbf{k}|\Delta s,$$

which we note is purely real. Furthermore, we see that the second term in (43) is purely imaginary. We proceed by breaking the argument across two separate cases.

First, assume that \mathbf{k} is such that $|k_d h_d| \in [0, \pi]$ for all d . Then by assumption, $\tilde{R}(\mathbf{k}) \in [0, 1]$. We use convexity to see that the real part of $\tilde{S}(\mathbf{k})$ squared is bounded according to

$$\begin{aligned} \operatorname{Re}[\tilde{S}(\mathbf{k})]^2 &= \left(\frac{\sin c|\mathbf{k}|\Delta s}{c|\mathbf{k}|\Delta s}(1 - \tilde{R}(\mathbf{k})) + \tilde{R}(\mathbf{k}) \cos c|\mathbf{k}|\Delta s \right)^2 \\ &\leq \left(\frac{\sin c|\mathbf{k}|\Delta s}{c|\mathbf{k}|\Delta s} \right)^2 (1 - \tilde{R}(\mathbf{k})) + \tilde{R}(\mathbf{k})(\cos c|\mathbf{k}|\Delta s)^2. \end{aligned}$$

On the other hand, the squared imaginary part of $\tilde{S}(\mathbf{k})$ is bounded according to

$$\operatorname{Im}[\tilde{S}(\mathbf{k})]^2 = \sin^2 c|\mathbf{k}|\Delta s \left(\frac{|\tilde{\Delta}_x^2(\mathbf{k}) + \tilde{\Delta}_y^2(\mathbf{k}) + \tilde{\Delta}_z^2(\mathbf{k})|}{|\mathbf{k}|^2} \right) \leq \tilde{R}(\mathbf{k}) \sin^2 c|\mathbf{k}|\Delta s.$$

Combining these two bounds and using the fact that $\cos^2(x) + \sin^2(x) = 1$, it is simple to show that $|\tilde{S}(\mathbf{k})|^2 \leq 1$ as desired.

Next, suppose that $|k_d| > \pi/h_d$ for some d . By assumption, $|\tilde{R}| \leq B$ and further it is evident that $|\mathbf{k}| \geq \pi/h_d$. In this case, we compute that the real part of $\tilde{S}(\mathbf{k})$ is given by

$$\begin{aligned} \operatorname{Re}[\tilde{S}(\mathbf{k})]^2 &= \left(\frac{\sin c|\mathbf{k}|\Delta s}{c|\mathbf{k}|\Delta s} \right)^2 (1 - \tilde{R}(\mathbf{k}))^2 + \tilde{R}^2(\mathbf{k}) \cos^2 c|\mathbf{k}|\Delta s \\ &\quad + 2(1 - \tilde{R}(\mathbf{k}))\tilde{R}(\mathbf{k}) \cos c|\mathbf{k}|\Delta s \frac{\sin c|\mathbf{k}|\Delta s}{c|\mathbf{k}|\Delta s}. \end{aligned}$$

Using the identity $2 \cos(x) \sin(x) = \sin(2x)$ and the bound B from our assumptions, we see

$$\operatorname{Re}[\tilde{S}(\mathbf{k})]^2 \leq \left(\frac{\sin c|\mathbf{k}|\Delta s}{c|\mathbf{k}|\Delta s} \right)^2 (1 + B)^2 + B \cos^2 c|\mathbf{k}|\Delta s + B(1 + B) \left| \frac{\sin 2c|\mathbf{k}|\Delta s}{c|\mathbf{k}|\Delta s} \right|.$$

We bound the sines by 1 and the magnitudes of \mathbf{k} by π/h_d to obtain

$$\begin{aligned} \operatorname{Re}[\tilde{S}(\mathbf{k})]^2 &\leq \left(\frac{1}{c\pi/h_d\Delta s} \right)^2 (1 + B)^2 + B \cos^2 c|\mathbf{k}|\Delta s + B(1 + B) \left| \frac{1}{c\pi/h_d\Delta s} \right| \\ &\leq \frac{1}{\sigma^2\pi^2} (1 + B)^2 + B \cos^2 c|\mathbf{k}|\Delta s + B(1 + B) \frac{1}{\sigma\pi}. \end{aligned}$$

The imaginary part of $\tilde{S}(\mathbf{k})$ is bounded trivially as before since

$$\text{Im}[\tilde{S}(\mathbf{k})]^2 \leq |\tilde{R}(\mathbf{k})| \sin^2 c|\mathbf{k}|\Delta s \leq B \sin^2 c|\mathbf{k}|\Delta s.$$

Combining these two bounds and assuming $\sigma \geq (1 + B)/(\pi(1 - B))$ again gives that $|\tilde{S}(\mathbf{k})|^2 \leq 1$.

Now we have shown that $|\tilde{S}(\mathbf{k})| \leq 1$ for all \mathbf{k} , from which it immediately follows that $\rho(\mathbf{k}) \leq 1$ since

$$\begin{aligned} \rho(\mathbf{k}) &\leq \int_{\mathbb{R}^3} \text{III}_{2\pi/h}(\mathbf{k} - \mathbf{k}') \left[\left(\prod_{d=1}^3 \tilde{W}(k'_d h_d) \right) \cdot |\tilde{S}(\mathbf{k}')| \right] d\mathbf{k}' \\ &\leq \int_{\mathbb{R}^3} \text{III}_{2\pi/h}(\mathbf{k} - \mathbf{k}') * \left(\prod_{d=1}^3 \tilde{W}(k'_d h_d) \right) d\mathbf{k}' \\ &= \prod_{d=1}^3 \left(\sum_{l \in \mathbb{Z}} \tilde{W}(k_d h_d + 2\pi l) \right) = 1, \end{aligned}$$

where the last equality comes from the fact that by Poisson summation we have that $\sum_{l \in \mathbb{Z}} \tilde{W}(kh + 2\pi l)$ is the discrete-space Fourier transform of W evaluated at kh , which is unity everywhere by the fact that W behaves like the Kronecker delta on the lattice points. This concludes the proof. \square

The main result of [Theorem 5](#) is somewhat odd in the sense that we have shown the von Neumann condition holds for all σ above a certain minimum value whereas in general we expect stability results to give an upper bound on σ . We believe this to simply be an artifact of the proof technique employed.

To extend [Theorem 5](#) to the wave equation propagator, we note that for this new propagator the eigenvalues in Fourier space are given by

$$\begin{aligned} \lambda_{1,2}^W(\mathbf{k}) &= \tilde{H}^{\Delta s, h}(\mathbf{k}) \pm \tilde{G}^{\Delta s, h}(\mathbf{k}) \sqrt{\tilde{\Delta}_x^2(\mathbf{k}) + \tilde{\Delta}_y^2(\mathbf{k}) + \tilde{\Delta}_z^2(\mathbf{k})}, \\ \lambda_3^W(\mathbf{k}) &= \tilde{H}^{\Delta s, h}(\mathbf{k}) \end{aligned}$$

with varying multiplicity. Analysis analogous to that in the proof of [Theorem 5](#) proceeds in a similar fashion by taking $c = 1$. For the source-free case, it is not necessary to use a quadrature scheme and thus one can take $\Delta t = \Delta s$.

We must caution that, in proving [Theorem 5](#), we have assumed that the spherical integrals are performed exactly, which in practice is not possible. It is unclear whether the use of a quadrature scheme such as described in [Section 3.2](#) might lead to unstable modes. Our numerical experiments, however, show no evidence of instability.

5. Numerical experiments

We have performed convergence studies using two tests: one where the exact solution is known, where we can compute the true absolute error, and another where the exact solution is unknown, where we estimate the convergence rate using Richardson error estimation. We also present numerical test results investigating any dependence the solution has on domain decomposition and number of points in the spherical quadrature of (34). Lastly, we present timing results demonstrating the weak scaling of our solver.

The following notation is used throughout the section:

N : number of grid points per spatial dimension,

h : grid spacing ($1/N$),

σ : CFL number ($c\Delta t/h$),

N_θ : number of points in the θ direction for the spherical quadrature,

N_t : number of time steps.

All tests are performed in a unit cube with N^3 points and $N_\theta = 16$ unless otherwise specified. All tests presented in this section were performed on the Edison machine at the NERSC facility.

5.1. Results. We implemented our method in C++ using the Chombo library [1]. Our implementation uses a sixth-order central difference stencil for the spatial derivatives, a sixth-order interpolation formula for the discrete delta distribution, and the sixth-order Boole’s rule for the source integration. All convolutions are performed via Hockney’s method with simple domain doubling using the FFTW library [8]. We note that it is in fact not necessary to fully double the domain to perform the convolutions for our solver—rather, we extend the domain with a number of grid cells equal to the support of the discrete propagator, which is usually much smaller than the local domain on each processor.

Plane wave. We begin by testing our code with no source ($\mathbf{J} = \mathbf{0}$ and $\rho = 0$) with periodic boundary conditions and initial conditions of the form

$$\begin{aligned} E_x(x, y, z) &= B_x(x, y, z) = 0, \\ E_y(x, y, z) &= B_z(x, y, z) = E_{y0} \sin 2\pi x, \\ E_z(x, y, z) &= E_{z0} \sin 2\pi x, \\ B_y(x, y, z) &= -E_z(x, y, z) \end{aligned}$$

in the domain $\Omega = [0, 1]^3$ m with $t_{\text{final}} = \frac{25}{8c}$ s. Solving Maxwell’s equations with these initial conditions yields a set of plane waves propagating in the x direction with velocity c . Figure 3 shows the absolute error for $\sigma = [1, 10, 100]$ for the sixth-order

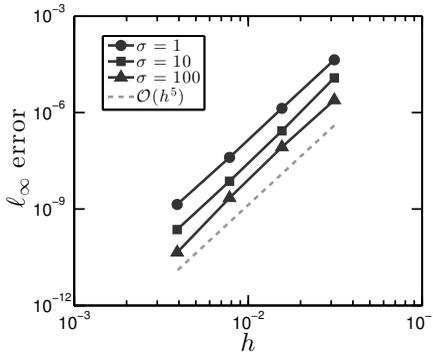


Figure 3. Max-norm error for the plane wave problem for $\sigma = [1, 10, 100]$ with $N = [32, 64, 128, 256]$ with $\sigma N_t = [100, 200, 400, 800]$, which corresponds to $t_{\text{final}} = \frac{25}{8c}$ s. The error scales as $h^{4.99}$ for $\sigma = 1$, $h^{5.22}$ for $\sigma = 10$, and $h^{5.28}$ for $\sigma = 100$. The $\sigma = 100$ problem uses $N_\theta = 128$, and the others use $N_\theta = 16$.

solver with $E_{y0} = E_{z0} = 1$. As expected with constant- σ tests, the results yield fifth-order convergence for the absolute error; i.e., we lose one order of accuracy since the number of time steps is inversely proportional to the spatial step size.

Divergence-free current source. For the second test, we begin with zero initial condition and zero charge density but with a divergence-free current density [5] of the form

$$\begin{aligned}
 J_x(x, y, z) &= -\frac{(y - y_0)}{r} \sin\left(\frac{\pi r}{2a}\right) \cos^{10}\left(\frac{\pi r}{2a}\right) \cos^{11}\left(\frac{\pi(z - z_0)}{d}\right) \sin(2\pi \nu t), \\
 J_y(x, y, z) &= \frac{x - x_0}{r} \sin\left(\frac{\pi r}{2a}\right) \cos^{10}\left(\frac{\pi r}{2a}\right) \cos^{11}\left(\frac{\pi(z - z_0)}{d}\right) \sin(2\pi \nu t), \\
 J_z(x, y, z) &= 0,
 \end{aligned}$$

where $r = r(x, y) \equiv \sqrt{(x - x_0)^2 + (y - y_0)^2}$. With this source, we solve Maxwell's equations in a unit cube with open boundary conditions and parameters $a = 0.25$ m, $d = x_0 = y_0 = z_0 = 0.5$ m, and $\nu = 149\,896\,229\text{ s}^{-1}$ in the domain $\Omega = [0, 1]^3$ m to $t_{\text{final}} = \frac{5}{32c}$ s. This frequency was chosen to match the low-frequency test of Chilton [5]. For this problem, the z component of the electric field is $E_z = 0$ for all time. Table 1 shows the Richardson error estimate for test problems with $\sigma = [\frac{1}{2}, 1, 10]$ using the sixth-order solver. For $\sigma = \frac{1}{2}$ and $\sigma = 1$, the error estimate for the nonzero components of the EM fields show fifth-order convergence as expected and for E_z it shows sixth-order convergence. For $\sigma = 10$, we see unexpected higher-order convergence for the error, which requires more investigation. However, as we see in Figure 4, the solution differences between corresponding points as the spatial resolution varies seem to indicate that our test cases sit within the asymptotic regime (a necessary condition for the validity of Richardson error estimates).

Component	$\sigma = 0.5$			$\sigma = 1$			$\sigma = 10$		
	ℓ_∞	ℓ_1	ℓ_2	ℓ_∞	ℓ_1	ℓ_2	ℓ_∞	ℓ_1	ℓ_2
E_x	4.96	4.99	4.99	4.97	4.99	5.00	8.12	7.46	7.87
E_y	4.96	4.99	4.99	4.97	4.99	5.00	8.12	7.46	7.87
E_z	5.88	5.89	5.90	5.82	5.85	5.84	4.89	5.22	5.05
B_x	5.12	5.12	5.15	5.14	5.14	5.17	6.63	7.03	6.96
B_y	5.12	5.12	5.15	5.14	5.14	5.17	6.63	7.03	6.96
B_z	5.07	5.09	5.11	5.08	5.10	5.13	6.53	7.03	6.95

Table 1. Richardson error estimate of asymptotic rate using ℓ_∞ , ℓ_1 , and ℓ_2 norms for the specified current test $\sigma = [\frac{1}{2}, 1, 10]$ with $N = [129, 257, 513]$ and $\sigma N_t = [20, 40, 80]$, respectively, corresponding to $t_{\text{final}} = \frac{5}{32c}$ s.

For the Richardson error estimation, we solve the problem to the same final time with increasingly finer discretizations h , $h/2$, and $h/4$ and let u_i denote the solution corresponding to the grid with spacing i . We then sample the solution on the $h/4$ grid onto the $h/2$ grid and the solution on the $h/2$ grid onto the h grid. Letting \mathcal{S}_i denote the sample operator that transfers a solution on a grid with spacing $i/2$ to a

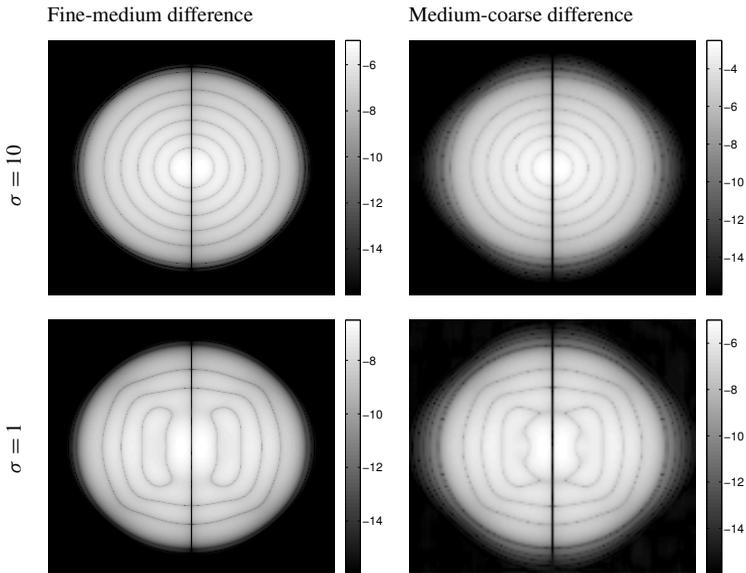


Figure 4. Visualizations of the logarithm (base 10) of the absolute difference between the solution on grids of different sizes, as used to obtain the antenna problem Richardson error estimates in Table 1. Our Richardson estimates use three grids at sizes $N = 513$ (fine), $N = 129$ (medium), and $N = 65$ (coarse). For example, to obtain the fine-medium difference, we subsample the values of E_x on the fine grid and measure the pointwise difference between the values at the corresponding locations on the medium grid. The figures here show a cross-sectional view of these quantities at the plane $z = 0.5$ m.

N_{patch}	N_θ	ℓ_∞ error	N_{patch}	N_θ	ℓ_∞ error
7	16	2.389×10^{-11}	9	32	6.343×10^{-12}
8	16	2.389×10^{-11}	9	64	3.543×10^{-11}
9	16	2.387×10^{-11}	9	128	4.175×10^{-11}
10	16	2.391×10^{-11}	9	256	1.031×10^{-11}

Table 2. Absolute ℓ_∞ error for the plane wave problem with $\sigma = 1$, $N = 513$, and $N_t = 160$, corresponding to $t_{\text{final}} = \frac{5}{16c}s$. The domain is decomposed into N_{patch}^3 number of subdomains.

grid with spacing i , the ℓ_p norm error rate estimate is given by

$$q = \frac{\log |u_{h/2} - \mathcal{S}_{h/2}(u_{h/4})|_p - \log |u_h - \mathcal{S}_h(u_{h/2})|_p}{\log \frac{1}{2}}. \tag{44}$$

For parallelization via domain decomposition, we break the domain into N_{patch}^3 subdomains and then solve the problem in parallel with a number of processors $N_{\text{proc}} = N_{\text{patch}}^3$. The error in our algorithm should not depend on N_{patch} , which we confirmed by solving the plane wave problem with fixed $\sigma = 1$, $N = 513$, and $N_t = 160$ and varying N_{patch} . The absolute ℓ_∞ error results can be seen in the left part of [Table 2](#). As expected, the error shows no significant dependence on the subdomain sizes. Further, for this same plane wave problem, we investigated the dependance of the error on the number of discretization points used for the spherical quadrature, N_θ , which shows a slight decreasing trend as expected; see the right half of [Table 2](#).

[Table 3](#) shows weak scaling results of our algorithm in parallel applied to the prescribed current-source problem with CFL parameter $\sigma = \frac{1}{2}$. As shown by the normalized τ factor, our solver exhibits reasonable scaling once communication is introduced in the problem while it is lower for the single-processor case where no communication is necessary. Note that, while the number of processors scales with

N	N_t	N_{proc}	t_{solve} (s)	t_{quad} (s)	τ
65	20	1	613.65	0.02	1.11×10^{-4}
129	40	8	1471.29	0.03	1.37×10^{-4}
257	80	64	2791.86	0.03	1.32×10^{-4}
513	160	512	5830.04	0.03	1.38×10^{-4}

Table 3. Timing data for the prescribed current problem with $\sigma = \frac{1}{2}$ with the domain subdivided into $N_{\text{patch}}^3 = N_{\text{proc}}$ total subdomains solved on N_{proc} processors. The time spent in the solver, t_{solve} , and time spent doing the spherical quadrature, t_{quad} , are taken from the timing data of a single processor. The factor $\tau = t_{\text{solve}}N_{\text{proc}}/(N^3N_t)$ is a normalized measure of time spent in the solver. In the perfect scaling case, τ would remain a constant.

the total number of points in the spatial discretization, the number of time steps increases by a factor of 2 between subsequent rows of the table, which is reflected in the way t_{solve} roughly doubles between rows.

6. Conclusion

Our numerical results demonstrate that we attain the desired order of accuracy through both a simple plane wave example (where the true solution is known) and a more complicated example with a time-dependent source (where we employ Richardson error estimates). The major advantage to this method is that it is easily parallelizable. This method does not have a CFL condition based on the wave speed so that the communication cost relative to an explicit method of same accuracy is much lower. The lack of a time step constraint is also a significant cost reduction when the field solver is used in PIC code. In addition, since the method is based on convolutions with compactly supported kernels, the Hockney algorithm is used on small patches of the domain; thus, it is well suited for multicore architectures with deep memory hierarchies. Our implementation has shown good weak scaling via parallelization in space. Our method is also computationally cheaper than explicit methods of the same accuracy for large CFL or large number of grid points. Suppose we would like to advance the solution by a time T using one time step of the present method. If we define $\sigma = \lceil cT/h \rceil$, then σ is the number of ghost points required in each direction per patch. The amount of work it takes to solve the problem per patch would then be the cost of the FFT and multiplication for Hockney, and therefore, it is $\mathcal{O}((N + \sigma)^3 \log(N + \sigma))$. On the other hand, for an explicit method, the CFL stability condition on the time step implies that the number of time steps required to advance the solution to time T is $\mathcal{O}(\sigma)$, and the total work required per patch would be $\mathcal{O}(N^3 \sigma)$. Thus, we expect a decrease in the time to solution of the present method relative to an explicit method to be $\mathcal{O}(\log(N + \sigma)(1/N + 1/\sigma))$. If the execution time is dominated by the time to read and write the field data to and from cache, then the present algorithm is performing $\mathcal{O}(1/\sigma)$ as many such communication steps as an explicit method, and the time to solution is reduced by that factor.

In essence, the method presented in this paper solves the free-space Maxwell equations by assuming that the fields have been separated into local and nonlocal parts via Helmholtz decomposition and solving the local portion in parallel by constructing a compactly supported discrete convolutional kernel via

- (1) finding an explicit analytic form for the Maxwell propagator,
- (2) regularizing the singularities with convolutional smoothing kernels,
- (3) replacing spatial derivatives with finite difference stencils, and
- (4) sampling the result on a Cartesian grid.

Because the resulting discretized propagator takes the form of discrete convolution against a compactly supported kernel, a regular decomposition of the domain across processors with a halo region whose size is driven by the support of the propagator (i.e., the size of Δt and the order of discretization accuracy) admits exact application of the discrete propagator in parallel; i.e., the error is independent of the domain decomposition. Furthermore, by appropriate choice of the smoothing kernel and finite difference stencils, our method can attain an arbitrarily high order of accuracy. We view the rigorous error analysis and demonstration of accuracy as a strength of our paper and method. The method of [20] employs a similar idea for parallelization, using linearity and finite propagation speed to justify domain decomposition in the solution of Maxwell's equations. In fact, the continuous propagator for our method (24) is the same as the one used in [20]. However, the method the authors present advances all fields in Fourier rather than physical space, which they approximate in parallel by taking local FFTs on each subdomain. While they assert spectral accuracy of their method, they do not provide any analysis of the method, nor convergence results that would support such an assertion. The authors do note that the finite number of modes used in this representation leads qualitatively to small nonlocal errors, of which they defer analysis for later work.

It is important to keep in mind that the approach we show here assumes that the electric field has been decomposed everywhere via the Helmholtz decomposition and that the divergence-free component is to be treated by other methods. As such, future work will focus on coupling of our method with a fast and accurate method for Poisson's equation in order to compute the solution to (22) and/or (23) in the specific application to PIC methods for the Maxwell–Vlasov equations. While it may appear that, by looking only at the case of $\mathbf{J}_L \equiv 0$, we are ignoring a large computational cost, the trend in PIC methods for charged systems is towards a large number of particles per cell, i.e., hundreds or thousands, for the purpose of minimizing numerical noise. For electrostatic problems, the need for such large numbers of particles per cell is indicated by the convergence theory for PIC methods [21]. In this regime, the field solve constitutes a small fraction of the cost, even with a Poisson solve included. However, for classical explicit time-stepping methods, the time step for the overall calculation is constrained by the CFL condition for the Maxwell solve so that the ability to use larger time steps provides an additional tool to improve overall performance. Another area we will investigate is the extension of this method to locally refined grids, using an approach analogous to that in [14] for Poisson's equation. Since our key motivation for this method is to use it in a PIC method to simulate free plasmas where the EM waves radiate out with open boundary conditions, we are not concerned with dealing with other boundary conditions with this method. However, incorporating boundary conditions in integral evolution methods for the wave equation has been examined [13].

Finally, in the present method, we use the special structure of Maxwell's equations and the wave equation in 3-D to compute an analytic form for the propagator with support on the surface of the sphere corresponding to the wave front. While the propagator in 3-D involves integrating on the surface of a sphere, the propagator does not take this form in general. For instance, the 2-D wave equation propagator requires integration on a disc [22]. Therefore, it is not obvious if we could extend our idea of regularization to other dimensions. Nonetheless, there are a number of ways in which we could attempt to generalize this approach. One approach would be to construct a discrete propagator directly, using iterates of an explicit method applied to discrete-delta-function initial data. This would be done at most once per time step on a small patch and then applied multiple times as a discrete convolution kernel, as above. Another approach would be to use geometrical optics [12] as a starting point for constructing a sufficiently accurate approximate propagator to represent the stiff wave propagation. The first problem to which to apply either of these approaches would be the wave equation in 2-D, followed by the linearized Euler equations in the low-Mach number limit or linearized MHD in the low-Alfvén number limit.

Appendix: Constructing compactly supported delta approximations

To derive a function δ_h satisfying the discrete moment conditions, we first define the unscaled approximant W such that

$$\delta_h(x) = \frac{1}{h} W(x/h). \quad (45)$$

Theorem 7.2.1 from [6] gives sufficient conditions for the discrete moment conditions in terms of the behavior of the Fourier transform of W , which we restate here without proof.

Theorem 6 (continuous moment conditions [6]). *Consider the approximation*

$$f_{\text{app}}(\bar{x}) = \sum_{j \in \mathbb{Z}} W(j - \bar{x}) f(j). \quad (46)$$

Suppose that W decays sufficiently quickly, i.e., $|W(x)| \leq A \exp(-B|x|)$ for some constants A and B . Then, the interpolation formula is of degree q if the following two conditions hold:

- (1) *The function $\tilde{W}(k) - 1$ has a zero of order q at $k = 0$.*
- (2) *The function $\tilde{W}(k)$ has a zero of order q at $k = 2\pi j$ for integer $j \neq 0$.*

We note that the first condition in **Theorem 6** is equivalent to the continuous moment conditions in (29) and the second condition arises from the periodic summation of the spectrum of W due to sampling.

For computation purposes, it is desirable that the support of W be as small as possible. Due to a theorem of Tornberg and Engquist, we have the following minimum bound on this support.

Theorem 7 (minimum support for an approximate delta function [19]). *There exists a function $W \in Q^q$ if and only if the support of W contains the interval $[-q/2, q/2]$. Furthermore, for each choice of q , there is a unique W that achieves this minimum support though it is not, in general, smooth (or even continuous).*

In other words, **Theorem 7** says that the support of W centered at 0 must cover at least $q + 1$ points in the discrete grid such that the support of W with arbitrary center covers at least q points in the discrete grid. This ensures an adequate number of degrees of freedom to satisfy the discrete moment conditions.

Define the B -splines via the recursion

$$M_q = M_{q-1} * M_1, \quad M_1 = \chi_{[-1/2, 1/2]}, \tag{47}$$

where $\chi_{[a,b]}$ is the indicator function of the interval $[a, b]$. It is evident that $M_q \in C^{q-2}$ is supported on the interval $[-q/2, q/2]$, and it is well known that its Fourier transform is given by

$$\tilde{M}_q(k) = \left(\frac{\sin(k/2)}{k/2} \right)^q, \tag{48}$$

which has zeros of order q at nonzero integer multiples of 2π . Unfortunately, $\tilde{M}_q(k) - 1$ has zeros of only order 2 at $k = 0$, restricting B -splines to only second-order approximations of the discrete delta distribution [15].

Based on these facts, we suppose for simplicity that q is even¹ and introduce the ansatz

$$\tilde{W}_q(k) = \sum_{p=0}^{q/2-1} a_{2p} k^{2p} \tilde{M}_q(k). \tag{49}$$

Because $\tilde{M}_q(k)$ decays as $1/k^q$, we see that $\tilde{W}_q(k)$ decays at least as fast as $1/k^2$, ensuring that $W_q(x)$ is continuous. Furthermore, it is evident that $\tilde{W}_q(k)$ still has zeros of order q at $j2\pi$ for integer $j \neq 0$ regardless of the choice of coefficients a_p . Finally, we see that $\tilde{W}_q(k)$ is real and even, therefore leading to a $W_q(x)$ that is real and symmetric. It remains to choose these coefficients such that $\tilde{W}_q(k) - 1$ has zeros of order q at $k = 0$.

Let the Taylor expansion of $\tilde{M}_q(k)$ about 0 be given by

$$\tilde{M}_q(k) = \sum_{p=0}^{q/2-1} b_{2p} k^{2p} + \mathcal{O}(k^q), \tag{50}$$

¹A similar argument holds for odd q , but the resulting approximant is not continuous.

where we note that $\tilde{M}_q(k)$ is an even function and thus all odd coefficients are necessarily zero. Then, the Taylor expansion of $\tilde{W}_q(k)$ about 0 is given by

$$\tilde{W}_q(k) = \sum_{m=0}^{q/2-1} \left(\sum_{p=0}^m a_{2p} b_{2m-2p} \right) k^{2m} + \mathcal{O}(k^q), \tag{51}$$

where we see that the coefficients are given by a convolutional formula. To ensure zeros of the appropriate order, we would like to choose a_p such $a_0 b_0 = 1$ and the rest of the coefficients are 0. This leads to $q/2$ equations in $q/2$ unknowns in a triangular system of the form

$$\begin{bmatrix} a_0 & 0 & 0 & \cdots & 0 \\ a_2 & a_0 & 0 & \cdots & 0 \\ a_4 & a_2 & a_0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ a_{q-2} & a_{q-4} & a_{q-6} & \cdots & a_0 \end{bmatrix} \begin{bmatrix} b_0 \\ b_2 \\ b_4 \\ \vdots \\ b_{q-2} \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix}. \tag{52}$$

It is easy to verify that, for the B -splines, $a_0 = 1$ and thus this system is nonsingular, yielding a unique set of coefficients that lead to a $\tilde{W}_q(k)$ satisfying the conditions of [Theorem 6](#).

Now that we see we can attain a Fourier representation of an appropriate kernel of the form in [\(49\)](#), it remains to transform back to the spatial domain. However, by simple properties of the Fourier transform, we have that

$$W_q(x) = \sum_{p=0}^{q/2-1} a_{2p} (-1)^p M_q^{(2p)}(x); \tag{53}$$

i.e., we are simply taking a linear combination of the B -spline and its derivatives, leading once again to a piecewise polynomial spline supported on $[-q/2, q/2]$.

For completeness, we give the approximants W_q for $q = 4, 6$ produced by this method:

$$W_4(x) = \begin{cases} \frac{1}{2}|x|^3 - |x|^2 - \frac{1}{2}|x| + 1, & |x| \in [0, 1], \\ -\frac{1}{6}|x|^3 + |x|^2 - \frac{11}{6}|x| + 1, & |x| \in [1, 2], \\ 0, & \text{else.} \end{cases} \tag{54}$$

$$W_6(x) = \begin{cases} -\frac{1}{12}|x|^5 + \frac{1}{4}|x|^4 + \frac{5}{12}|x|^3 - \frac{5}{4}|x|^2 - \frac{1}{3}|x| + 1, & |x| \in [0, 1], \\ \frac{1}{24}|x|^5 - \frac{3}{8}|x|^4 + \frac{25}{24}|x|^3 - \frac{5}{8}|x|^2 - \frac{13}{12}|x| + 1, & |x| \in [1, 2], \\ -\frac{1}{120}|x|^5 + \frac{1}{8}|x|^4 - \frac{17}{24}|x|^3 + \frac{15}{8}|x|^2 - \frac{137}{60}|x| + 1, & |x| \in [2, 3], \\ 0, & \text{else.} \end{cases} \tag{55}$$

We note that the first coincides with the “ k -point central interpolation formula” for $k = 4$ described, e.g., by Schoenberg [\[18\]](#). In fact, discrete delta distributions

matching moment conditions to order q correspond directly with interpolation kernels on uniform grids that exactly integrate polynomials of order less than q [19], so the fact that both methods achieve the minimum support size of $[-q/2, q/2]$ means they are one and the same by [Theorem 7](#). Finally, we remark that different forms of the ansatz in (49) can lead to kernels with slightly larger support but higher degrees of smoothness, if that is desired.

Acknowledgments

This research is supported by the Office of Advanced Scientific Computing Research of the U.S. Department of Energy under Contract Number DE-AC02-05CH11231. In addition, Minden is supported by a U.S. Department of Energy Computational Science Graduate Fellowship under grant number DE-FG02-97ER25308. This research used resources of the National Energy Research Scientific Computing Center (NERSC), a user facility supported by the Office of Science of the U.S. Department of Energy under Contract Number DE-AC02-05CH11231.

Minden would like to thank A. Benson, A. Damle, and L. Ying for useful comments on early drafts of this manuscript.

References

- [1] M. Adams, P. Colella, D. T. Graves, J. N. Johnson, H. S. Johansen, N. D. Keen, T. J. Ligoeki, D. F. Martin, P. W. McCorquodale, D. Modiano, P. O. Schwartz, T. D. Sternberg, and B. Van Straalen, *Chombo software package for AMR applications: design document*, Tech. Report LBNL-6616E, Lawrence Berkeley National Laboratory, 2014.
- [2] B. Alpert, L. Greengard, and T. Hagstrom, *An integral evolution formula for the wave equation*, *J. Comput. Phys.* **162** (2000), no. 2, 536–543. [MR 1774266](#) [Zbl 0966.65062](#)
- [3] K. Atkinson, *Numerical integration on the sphere*, *J. Austral. Math. Soc. (B)* **23** (1982), no. 3, 332–347. [MR 642631](#) [Zbl 0497.65010](#)
- [4] R. P. Beyer and R. J. LeVeque, *Analysis of a one-dimensional model for the immersed boundary method*, *SIAM J. Numer. Anal.* **29** (1992), no. 2, 332–364. [MR 1154270](#) [Zbl 0762.65052](#)
- [5] S. Chilton, *A fourth-order adaptive mesh refinement solver for Maxwell's Equations*, Ph.D. thesis, University of California, Berkeley, 2013. [MR 3232238](#)
- [6] G.-H. Cottet and P. D. Koumoutsakos, *Vortex methods: theory and practice*, Cambridge University, Cambridge, 2000. [MR 1755095](#)
- [7] B. Fornberg, *Generation of finite difference formulas on arbitrarily spaced grids*, *Math. Comp.* **51** (1988), no. 184, 699–706. [MR 935077](#) [Zbl 0701.65014](#)
- [8] M. Frigo and S. G. Johnson, *FFTW: an adaptive software architecture for the FFT*, Proceedings of the 1998 IEEE International Conference on Acoustics, Speech and Signal Processing (Seattle, 1988), vol. 3, IEEE, Piscataway, NJ, 1998, pp. 1381–1384.
- [9] T. Hagstrom, *High-resolution difference methods with exact evolution for multidimensional waves*, *Appl. Numer. Math.* **93** (2015), 114–122. [MR 3323449](#) [Zbl 1326.65107](#)
- [10] M. Hochbruck and A. Ostermann, *Exponential integrators*, *Acta Numer.* **19** (2010), 209–286. [MR 2652783](#) [Zbl 1242.65109](#)

- [11] R. W. Hockney, *The potential calculation and some applications*, Methods in computational physics (B. Alder, S. Fernbach, and M. Rotenberg, eds.), vol. 9: Plasma physics, Academic, New York, 1970, pp. 135–211.
- [12] P. D. Lax, *Asymptotic solutions of oscillatory initial value problems*, Duke Math. J. **24** (1957), 627–646. [MR 0097628](#) [Zbl 0083.31801](#)
- [13] J.-R. Li and L. Greengard, *High order marching schemes for the wave equation in complex geometry*, J. Comput. Phys. **198** (2004), no. 1, 295–309. [MR 2071396](#) [Zbl 1052.65075](#)
- [14] P. McCorquodale, P. Colella, G. T. Balls, and S. B. Baden, *A local corrections algorithm for solving Poisson's equation in three dimensions*, Commun. Appl. Math. Comput. Sci. **2** (2007), 57–81. [MR 2327083](#) [Zbl 1133.65106](#)
- [15] J. J. Monaghan, *Extrapolating B splines for interpolation*, J. Comput. Phys. **60** (1985), no. 2, 253–262. [MR 805872](#) [Zbl 0588.41005](#)
- [16] C. S. Peskin, *The immersed boundary method*, Acta Numer. **11** (2002), 479–517. [MR 2009378](#) [Zbl 1123.74309](#)
- [17] R. D. Richtmyer and K. W. Morton, *Difference methods for initial-value problems*, 2nd ed., Interscience Tracts in Pure and Applied Mathematics, no. 4, Interscience, New York, 1967. [MR 0220455](#) [Zbl 0155.47502](#)
- [18] I. J. Schoenberg, *Contributions to the problem of approximation of equidistant data by analytic functions, A: On the problem of smoothing or graduation*, I. J. Schoenberg selected papers (C. de Boor, ed.), vol. 2, Birkhäuser, Boston, 1988, pp. 3–57.
- [19] A.-K. Tornberg and B. Engquist, *Numerical approximations of singular source terms in differential equations*, J. Comput. Phys. **200** (2004), no. 2, 462–488. [MR 2095274](#) [Zbl 1115.76392](#)
- [20] J.-L. Vay, I. Haber, and B. B. Godfrey, *A domain decomposition method for pseudo-spectral electromagnetic simulations of plasmas*, J. Comput. Phys. **243** (2013), 260–268. [MR 3064167](#)
- [21] B. Wang, G. H. Miller, and P. Colella, *A particle-in-cell method with adaptive phase-space remapping for kinetic plasmas*, SIAM J. Sci. Comput. **33** (2011), no. 6, 3509–3537. [MR 2873252](#) [Zbl 1232.76046](#)
- [22] G. B. Whitham, *Linear and nonlinear waves*, Wiley, New York, 1974. [MR 0483954](#) [Zbl 0373.76001](#)

Received May 4, 2015. Revised January 28, 2016.

BORIS LO: bt.lo@berkeley.edu

Applied Science and Technology, University of California, Berkeley, Berkeley, CA 94720, United States

VICTOR MINDEN: vminden@stanford.edu

Institute for Computational and Mathematical Engineering, Stanford University, Stanford, CA 94305, United States

PHILLIP COLELLA: pcolella@lbl.gov

Lawrence Berkeley National Laboratory, Berkeley, CA 94720, United States

and

Electrical Engineering and Computer Sciences, University of California, Berkeley, Berkeley, CA 94720, United States

Communications in Applied Mathematics and Computational Science

msp.org/camcos

EDITORS

MANAGING EDITOR

John B. Bell
Lawrence Berkeley National Laboratory, USA
jbbell@lbl.gov

BOARD OF EDITORS

Marsha Berger	New York University berger@cs.nyu.edu	Ahmed Ghoniem	Massachusetts Inst. of Technology, USA ghoniem@mit.edu
Alexandre Chorin	University of California, Berkeley, USA chorin@math.berkeley.edu	Raz Kupferman	The Hebrew University, Israel raz@math.huji.ac.il
Phil Colella	Lawrence Berkeley Nat. Lab., USA pcolella@lbl.gov	Randall J. LeVeque	University of Washington, USA rjl@amath.washington.edu
Peter Constantin	University of Chicago, USA const@cs.uchicago.edu	Mitchell Luskin	University of Minnesota, USA luskin@umn.edu
Maksymilian Dryja	Warsaw University, Poland maksymilian.dryja@acn.waw.pl	Yvon Maday	Université Pierre et Marie Curie, France maday@ann.jussieu.fr
M. Gregory Forest	University of North Carolina, USA forest@amath.unc.edu	James Sethian	University of California, Berkeley, USA sethian@math.berkeley.edu
Leslie Greengard	New York University, USA greengard@cims.nyu.edu	Juan Luis Vázquez	Universidad Autónoma de Madrid, Spain juanluis.vazquez@uam.es
Rupert Klein	Freie Universität Berlin, Germany rupert.klein@pik-potsdam.de	Alfio Quarteroni	Ecole Polytech. Féd. Lausanne, Switzerland alfio.quarteroni@epfl.ch
Nigel Goldenfeld	University of Illinois, USA nigel@uiuc.edu	Eitan Tadmor	University of Maryland, USA etadmor@cscamm.umd.edu
		Denis Talay	INRIA, France denis.talay@inria.fr

PRODUCTION

production@msp.org

Silvio Levy, Scientific Editor

See inside back cover or msp.org/camcos for submission instructions.

The subscription price for 2016 is US \$95/year for the electronic version, and \$135/year (+\$15, if shipping outside the US) for print and electronic. Subscriptions, requests for back issues from the last three years and changes of subscribers address should be sent to MSP.

Communications in Applied Mathematics and Computational Science (ISSN 2157-5452 electronic, 1559-3940 printed) at Mathematical Sciences Publishers, 798 Evans Hall #3840, c/o University of California, Berkeley, CA 94720-3840, is published continuously online. Periodical rate postage paid at Berkeley, CA 94704, and additional mailing offices.

CAMCoS peer review and production are managed by EditFlow® from MSP.

PUBLISHED BY

 **mathematical sciences publishers**
nonprofit scientific publishing

<http://msp.org/>

© 2016 Mathematical Sciences Publishers

Communications in Applied Mathematics and Computational Science

vol. 11

no. 2

2016

- A real-space Green's function method for the numerical solution of
Maxwell's equations 143
BORIS LO, VICTOR MINDEN and PHILLIP COLELLA
- Analysis of estimators for Adaptive Kinetic Monte Carlo 171
DAVID ARISTOFF, SAMUEL T. CHILL and GIDEON SIMPSON
- Comparison of continuous and discrete-time data-based modeling for
hypoelliptic systems 187
FEI LU, KEVIN K. LIN and ALEXANDRE J. CHORIN
- Hydrodynamics of suspensions of passive and active rigid particles: a rigid
multiblob approach 217
FLORENCIO BALBOA USABIAGA, BAKYTZHAN KALLEMOV, BLAISE
DELMOTTE, AMNEET PAL SINGH BHALLA, BOYCE E. GRIFFITH and
ALEKSANDAR DONEV