RESEARCH ARTICLE | NOVEMBER 16 2020

A fast local embedded boundary met power electromagnetic sources

Mathialakan Thavappiragasam; Andrew Christlieb; Liang Li; Pearson Guthrey

AIP Advances 10, 115318 (2020)
https://doi.org/10.1063/5.0019210

A fast local embedded boundary method suitable for high power electromagnetic sources

Cite as: AIP Advances 10, 115318 (2020); doi: 10.1063/5.0019210
Submitted: 21 June 2020 • Accepted: 25 October 2020 •
Published Online: 16 November 2020

Mathialakan Thavappiragasam,[1,a] Andrew Christlieb,[1,b] John Luginsland,[1,c] and Pierson Guthrey[2]

AFFILIATIONS

[1] Department of Computational Mathematics Science and Engineering, Michigan State University, East Lansing, Michigan 48824, USA
[2] Lawrence Livermore National Laboratory, Livermore, California 94551, USA

[a] **Author to whom correspondence should be addressed:** mathialakan@gmail.com
[b] **Also at:** Department of Mathematics, Michigan State University, East Lansing, MI 48824, USA.
[c] **Also at:** Department of Electrical and Computer Engineering, Michigan State University, East Lansing, MI 48824, USA.

ABSTRACT

High power sources of electromagnetic energy often require complicated structures to support electromagnetic modes and shape electromagnetic fields to maximize the coupling of the field energy to intense relativistic electron beams. Geometric fidelity is critical to the accurate simulation of these High Power Electromagnetic (HPEM) sources. Here, we present a fast and geometrically flexible approach to calculate the solution to Maxwell's equations in vector potential form under the Lorenz gauge. The scheme is an implicit, linear-time, high-order, A-stable method that is based on the method of lines transpose (MOL^T). As presented, the method is fourth order in time and second order in space, but the A-stable formulation could be extended to both high order in time and space. An $\mathcal{O}(n)$ fast convolution is employed for space-integration. The main focus of this work is to develop an approach to impose perfectly electrically conducting (PEC) boundary conditions in MOL^T by extending our past work on embedded boundary methods. As the method is A-stable, it does not suffer from small time step limitations that are found in explicit finite difference time domain methods when using either embedded boundary or cut-cell methods to capture geometry. This is a major advance for the simulation of HPEM devices. While there is no conceptual limitation to develop this in 3D, our initial work has centered on 2D. The extension to 3D requires validation that the proposed fixed point iteration will converge and is the subject of our follow-up work. The eventual goal is to combine this method with particle methods for the simulations of plasma. In the current work, the scheme is evaluated for EM wave propagation within an object that is bounded by PEC. The consistency and performance of the scheme are confirmed using the ping test and frequency mode analysis for rotated square cavities—a standard test in the HPEM community. We then demonstrate the diffraction Q value test and the use of this method for simulating an A6 magnetron. The ability to handle both PEC and open boundaries in a standard device test problem, such as the A6, gives confidence on the robustness of this new method.

I. INTRODUCTION

The generation of coherent, high power electromagnetic (HPEM) radiation via the interaction of vacuum electromagnetic fields and intense relativistic electron beams continues to remain an active area of research, despite tracing its roots to the days of Hertz and Marconi and the intensive development of radar during the Second World War. Modern applications include radar and remote sensing, as well as novel schemes for high power, long-range communication, atmospheric and ionospheric modification, space propulsion, advanced electronic warfare, and the emerging fields of directed energy technology.[1] The power density of these novel sources has demonstrated Moore's law-like behavior for decades, with a doubling time of roughly 26 months since the 1930s. Critical to this long-term, and on-going, technological advance has been the advent of high fidelity and computationally efficient algorithms to simulate these devices.[2]

The introduction of these advanced modeling tools has re-invigorated the design of HPEM devices by allowing fully three-dimensional structures combining low frequency features,

to handle high DC applied fields, to co-exist with high frequency "radio frequency" (RF) features, such as slow wave structures, which combine to allow the kinetic energy stored in intense relativistic electron beams to be transferred to radiated electromagnetic fields with power in 100s of megawatts to gigawatts that can then be used to produce useful applications. It is this combination of the need to model both low frequency (DC) applied fields and high frequency (AC) fields that leads to both strict requirements on geometric fidelity as well as the need to handle multiple time scales in a time domain simulation. The current state of the art for these simulation tools is still predominantly the particle-in-cell method,[3] with field solutions based on a finite difference time domain (FDTD). These standard Particle In Cell (PIC) methods often require highly parallel implementation to reduce the effect of the structure ortho-normal grid and/or so-called "cut cells" where the geometry is approximated by linear interpolation through the cells. Care must be taken in the introduction of these cut cells to ensure that the resulting geometric description does not have new numerical "surface" modes associated with the cut cell algorithm. Additionally, it is often the case that these new "cut cells" still suffer from a Courant–Friedrichs–Lewy (CFL) condition on the maximum time step. Even though the FDTD based Yee scheme is a well-agreed method for Maxwell's equations and preserves divergence-free quantities, because of an explicit scheme, it suffers from CFL restrictions and is not suitable for curved surfaces due to the cut-cell staircasing approximation. Another approach, the conformal finite difference time domain (CFDTD) PIC method, treats cut-cells differently to maintain second-order accuracy for curved surfaces.[4] The CFDTD Dey–Mittra algorithm adjusts the area and lengths of cut cells in the equation of Faraday's law in Maxwell's equation, but its stability is limited by the CFL condition and it would not be scalable. The ADI-FDTD is one of the most powerful schemes to solve Maxwell's equations because it relies on simple, one-dimensional, tridiagonal system solvers in contrast to a single large system solver as is required by the Crank–Nicholson implicit method. However, the ADI-FDTD method is mostly used to solve non-complicated domains such as rectangular domain and is not applicable for complex geometries because of showing only first-order accuracy for stair-stepped curved boundaries. The ADI-FDTD method combined with the Dey–Mittra embedded boundary method can model the curved domains associated with complex structures and time step sizes beyond the CFL limit.[5] The efficiency of this method depends on the one-dimensional tridiagonal solvers used underneath and that will cause a major bottleneck issue and affects the scalability of the scheme. Furthermore, the order of accuracy is limited to second order.

These issues have motivated our use of an embedded boundary method that exploits a vector potential description for the time evolution of Maxwell's equations. The rest of this paper presents this new field solutions' method, tests it against standard test problems from the HPEM community, namely, rotated cavities, and the classic A6 relativistic magnetron in two-dimensions, and finally discusses the suitability for extension to a fully three-dimensional algorithm that is mated to particle methods. In this paper, we show a novel field advance algorithm.

Vector potential formulations of electromagnetism are widely used in classical and quantum physics. Maxwell's equations describe the time evolution of four fields: the magnetic flux density (**B**), the electric field intensity (**E**), the electric flux density (**D**), and the magnetic field intensity (**H**). It is often convenient to employ a formulation based on the magnetic vector potential **A** and electric scalar potential ϕ. Maxwell's equations then reduce to uncoupled wave equations for the vector potential **A** and scalar potential ϕ under the Lorenz gauge condition.[6] We use the recently developed implicit A-stable scheme[7–11] to solve these wave equations using the MOL^T approach (also known as the transverse method of lines or Rothe's method)[12–14] that first discretizes time and then obtains spatially varying solutions using boundary-integral methods,[15,16] which are coupled algebraically at discrete times. An $\mathcal{O}(N)$, recursive fast convolution algorithm is used for the line integration. Extension to multi-dimensions is done using an ADI scheme, with each line solved independently. The ADI method is extended to a fourth order unconditionally stable scheme using successive convolution.[8] The coupled set of boundary-value problems that result from the time integration yield a Partial Differential Equation (PDE), which is solved using a Green's function method.[7–11] This approach avoids the use of matrices, eliminating the main bottleneck in scaling implicit methods. Stable solutions result for a wide range of mesh sizes and potentials to leverage novel computing architectures, such as Graphics Processing Unit (GPU).

A perfectly electrically conducting (PEC) boundary condition is derived based on the continuity of the magnetic flux normal to the boundary and is applied to the magnetic vector potential. Our formulation avoids the additional compatibility conditions used in Ref. 17. The boundary condition is imposed using an embedded boundary method,[10] which supports complex boundaries. This embedded PEC boundary condition is a slightly modified embedded Neumann boundary condition. The fundamental difference is that one of the partial spatial derivatives in the PDE is negative. As presented, the current embedded boundary formulation is second order in space but could be extended to higher order.

In this paper, we first derive wave equations for vector and scalar potentials **A** and ϕ under the Lorenz gauge condition discussed in Sec. II, then we describe our two-dimensional high-order implicit scheme using ADI splitting and the higher order embedded PEC boundary conditions in Sec. III, and finally, we give numerical results for several test cases in Sec. V.

II. 2D ELECTRIC SCALAR AND MAGNETIC VECTOR POTENTIAL

The macroscopic Maxwell's equations are as follows:

$$\partial_t \mathbf{B} = -\nabla \times \mathbf{E}, \tag{1}$$

$$\partial_t \mathbf{D} = \nabla \times \mathbf{H} - \mathbf{J}, \tag{2}$$

$$\nabla \cdot \mathbf{B} = 0, \tag{3}$$

$$\nabla \cdot \mathbf{D} = \rho, \tag{4}$$

where **J** is the electric current density and ρ is the electric charge density. In a linear isotropic medium, $\mathbf{D} = \varepsilon\mathbf{E}$ and $\mathbf{B} = \mu\mathbf{H}$. The

dielectric constant $\varepsilon = \epsilon_0\epsilon_r$, with ϵ_0 and ϵ_r being the free space and relative permittivity, respectively, and the permeability $\mu = \mu_0\mu_r$, with μ_0 and μ_r being the free space and relative permeability, respectively.

The electric scalar potential $\boldsymbol{\phi}$ and magnetic vector potential $\mathbf{A}$ are related to $\mathbf{E}$ and $\mathbf{B}$ by

$$\mathbf{E} = -\nabla\phi - \partial_t\mathbf{A}, \tag{5}$$

$$\mathbf{B} = \nabla \times \mathbf{A}. \tag{6}$$

For free space, Ampere's law [Eq. (2)] is

$$\frac{1}{c^2}\partial_t\mathbf{E} = \nabla \times \mathbf{B} - \mu_0\mathbf{J}, \tag{7}$$

where the free space phase velocity $c = \frac{1}{\sqrt{(\mu_0\epsilon_0)}}$.

Substituting $\mathbf{E}$ and $\mathbf{B}$ using Eqs. (5) and (6) and imposing the Lorenz gauge condition

$$\frac{1}{c^2}\partial_t\phi = -\nabla\mathbf{A},$$

a wave equation in terms of the magnetic vector potential $\mathbf{A}$ results in

$$\frac{1}{c^2}\partial_t^2\mathbf{A} - \nabla^2\mathbf{A} = \mu_0\mathbf{J}. \tag{8}$$

Similarly, by imposing the Lorenz gauge condition, a wave equation for the scalar potential $\boldsymbol{\phi}$ in free space is obtained,

$$\frac{1}{c^2}\partial_t^2\boldsymbol{\phi} - \nabla^2\boldsymbol{\phi} = \frac{\rho}{\varepsilon_0}. \tag{9}$$

A. Perfectly conducting boundary

The electric field ($\mathbf{E}$) is continuous along the boundary, and the magnetic flux ($\mathbf{B}$) is continuous along the normal to the boundary. Since the Perfectly Conducting Boundary (PEC) has infinite electrical conductivity ($\sigma = \infty$), there will be no interior electric field ($\mathbf{E_2} = 0$) in the perfect conductor (Fig. 1). It also follows that there is no magnetic field ($\mathbf{H_2} = 0$). The boundary conditions for the PEC become

$$\mathbf{n} \times \mathbf{E_1} = 0, \tag{10}$$

$$\mathbf{n} \times \mathbf{H_1} = \mathbf{J_s}, \tag{11}$$

$$\mathbf{n} \cdot \mathbf{B_1} = 0, \tag{12}$$

$$\mathbf{n} \cdot \mathbf{D_1} = \boldsymbol{\rho_s}, \tag{13}$$

where $\mathbf{n}$ is the unit normal vector to the boundary and $\mathbf{J_s}$ and $\boldsymbol{\rho_s}$ are the surface current density and charge density, respectively.

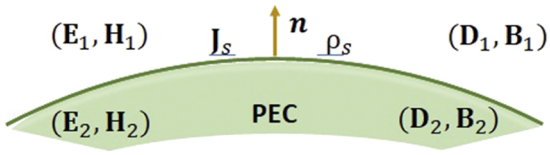

FIG. 1. Boundary surface between two regions with electric fields $\mathbf{E}_1$ and $\mathbf{E}_2$, magnetic fields $\mathbf{H}_1$ and $\mathbf{H}_2$, electric flux densities $\mathbf{D}_1$ and $\mathbf{D}_2$, and magnetic flux densities $\mathbf{B}_1$ and $\mathbf{B}_2$. Here, $\mathbf{J}_s$ is the surface current, ρ_s is the surface charge density, and $\mathbf{n}$ is the normal vector pointed out from the region two (PEC).

Consider a two-dimensional vector/scalar potential. If we choose to restrict B to the x–y plane, the vector potential $\mathbf{A}$ only has a z component,

$$\mathbf{A} = \mathbf{A}_z(x, y)\mathbf{z}. \tag{14}$$

Using Eq. (6) and the boundary condition represented by Eq. (12), we get

$$\mathbf{n} \cdot (\partial_y\mathbf{A}_z - \partial_x\mathbf{A}_z) = 0. \tag{15}$$

III. MOL^T BASED NUMERICAL SCHEME

A. Semi-discrete implicit scheme

The multi-dimensional wave equation for the vector potential $\mathbf{A}$ can be written as the initial boundary-value problem,

$$\frac{1}{c^2}\frac{\partial^2\mathbf{A}(\mathbf{k}, t)}{\partial t^2} - \nabla^2\mathbf{A}(\mathbf{k}, t) = \mu_0 J(\mathbf{k}, t), \quad \mathbf{k} \in \Omega, \quad t > 0, \tag{16}$$

$$\mathbf{A}(\mathbf{k}, 0) = f(\mathbf{k}), \quad \mathbf{k} \in \Omega,$$
$$\mathbf{A}_t(\mathbf{k}, 0) = g(\mathbf{k}), \quad \mathbf{k} \in \Omega,$$

with consistent boundary conditions.

Consider a scheme for two spatial dimensions. The two-dimensional implicit scheme uses an alternating direction implicit (ADI) scheme,[7] and each ADI line is solved independently.

Using MOL^T, we first perform a temporal discretization and then approximate the modified Helmholtz operator. The discretization begins with a time-centered finite different approximation,

$$\frac{\mathbf{A}^{n+1} - 2\mathbf{A}^n + \mathbf{A}^{n-1}}{(c\Delta t)^2} - \nabla^2\left(\mathbf{A}^n + \frac{\mathbf{A}^{n+1} - 2\mathbf{A}^n + \mathbf{A}^{n-1}}{\beta^2}\right) = \mu_0\mathbf{J}^n(\mathbf{k}, t), \tag{17}$$

where the averaging parameter $\beta > 0$. Without the β^2 term, the method would be central in time and thus an explicit method. To make the method implicit, we introduce an averaging parameter β. This new scheme is purely dispersive and fully implicit. The β should be big enough to achieve the targeted order of accuracy and should confirm the A-stability of the scheme. Based on local truncation errors of the Pth order scheme, $O((c\Delta t/\beta)^{2P+2})$,[7] the error constant will decrease with increasing β. However, the value of β must be below a limit for the scheme to maintain A-stability. For example, the second order scheme is A-stable for $0 < \beta \leq 2$ (proved in Ref.

7). An arbitral high order A-stable versions of the scheme are possible, accuracy $2P$, with $0 < \beta \le \beta_{max}$. As noted in Ref. 8, β_{max} depends on P. Since the truncation error decreases with increasing β, β_{max} should be an optimal choice for a specific order of accuracy in time. However, based on Ref. 10, we need to choose β slightly less than β_{max} for the embedded boundary scheme because of the artificial dissipation term, which was introduced to ensure the A-stability of the scheme. Reordering Eq. (17), we have the semi-discrete equation,

$$\left(1-\frac{1}{\alpha^2}\nabla^2\right)\left(\mathbf{A}^{n+1}+(\beta^2-2)\mathbf{A}^n+\mathbf{A}^{n-1}\right) = \beta^2\mathbf{A}^n+\frac{\mu_0\beta^2}{\alpha^2}\mathbf{J}^n(\mathbf{k},t), \tag{18}$$

where $\alpha = \frac{\beta}{c\Delta t}$. As in Refs. 8 and 10, we decompose the modified Helmholtz operator using ADI splitting in 2D,

$$1-\frac{1}{\alpha^2}\nabla^2 = 1-\frac{1}{\alpha^2}\left(\frac{\partial^2}{\partial x^2}+\frac{\partial^2}{\partial y^2}\right) = \left(1-\frac{1}{\alpha^2}\frac{\partial^2}{\partial x^2}\right)\left(1-\frac{1}{\alpha^2}\frac{\partial^2}{\partial y^2}\right)-\frac{1}{\alpha^4}\left(\frac{\partial^4}{\partial x^2\partial y^2}\right).$$

Hence, we make the temporal approximation,

$$1-\frac{1}{\alpha^2}\nabla^2 = \mathscr{L}_x\mathscr{L}_y+\mathscr{O}\left((c\Delta t)^4\right). \tag{19}$$

It results in a semi-discrete equation, with

$$\mathscr{L}_x = 1-\frac{\partial_{xx}}{\alpha^2} \quad and \quad \mathscr{L}_y = 1-\frac{\partial_{yy}}{\alpha^2},$$

allowing us to express Eq. (18) as

$$\mathscr{L}_x\mathscr{L}_y\left[\beta^2\mathbf{A}^n+\mathbf{A}^{n+1}-2\mathbf{A}^n+\mathbf{A}^{n-1}\right] = \beta^2\mathbf{A}^n+\frac{\mu_0\beta^2}{\alpha^2}\mathbf{J}^n(\mathbf{k},t). \tag{20}$$

Upon taking the inverse of the modified Helmholtz operators $\mathscr{L}_x$ and $\mathscr{L}_y$, we obtain the semi-discrete solution for two dimensions with second-order temporal accuracy,

$$\mathbf{A}^{n+1}-2\mathbf{A}^n+\mathbf{A}^{n-1} = -\beta^2\mathscr{D}_{xy}^{(1)}[\mathbf{A}^n]+\beta^2\mathscr{L}_y^{-1}\mathscr{L}_x^{-1}\left[\frac{\mu_0}{\alpha^2}\mathbf{J}^n\right](\mathbf{k},t), \tag{21}$$

where the two-dimensional operator is

$$\mathscr{D}_{xy}^{(1)}[\mathbf{A}] := \mathbf{A}-\mathscr{L}_y^{-1}\mathscr{L}_x^{-1}[\mathbf{A}] \tag{22}$$

with superscript 1, which denotes a second order scheme or level 1 computing (number of computing levels for $2p$th order scheme is p based on Refs. 8 and 11).

Using the free space Green's function, the inverse of the modified Helmholtz operator $\mathscr{L}_x^{-1}$ in the x direction can be defined as

$$\mathscr{L}_x^{-1}[\mathbf{A}] := \underbrace{\frac{\alpha}{2}\int_a^b e^{-\alpha|x-x'|}\mathbf{A}(x')dx'}_{\text{Particular solution}} + \underbrace{a_{x1}e^{-\alpha(x-x_a)}+b_{x1}e^{-\alpha(x_b-x)}}_{\text{Homogeneous solution}}, \tag{23}$$

where a_{x1} and b_{x1} are homogeneous boundary coefficients along the x direction. Since the subscripts of the coefficients denote the computing level, a_{x1} and b_{x1} are involved in the level 1 computing. We can define $\mathscr{L}_y^{-1}$ in a similar way using homogeneous boundary coefficients (a_{y1}, b_{y1}).

Higher order accurate solutions are obtained by exchanging the time derivative with the spatial derivative. As shown in Ref. 8, we can introduce a new operator $\mathscr{C}_{xy}$,

$$\mathscr{C}_{xy} = \mathscr{L}_y^{-1}\mathscr{D}_x+\mathscr{L}_x^{-1}\mathscr{D}_y.$$

We can combine $\mathscr{D}_{xy}$ [Eq. (22)] to get a symmetrized form

$$\mathscr{C}_{xy}^{(1)} = \mathscr{L}_x^{-1}+\mathscr{L}_y^{-1}-\mathscr{L}_x^{-1}\mathscr{L}_y^{-1}-\mathscr{L}_y^{-1}\mathscr{L}_x^{-1}.$$

As shown in Ref. 8, Taylor expanding $\mathbf{A}^{n+1}-2\mathbf{A}^n+\mathbf{A}^{n-1}$ in time and using a Lax–Wendroff procedure, and using what we know about the operator $\mathscr{C}$ and $\mathscr{D}$, we arrive at the high order method,

$$\mathbf{A}^{n+1}-2\mathbf{A}^n+\mathbf{A}^{n-1} = \sum_{p=1}^{\infty}\sum_{m=1}^{p}(-1)^m\frac{2\beta^{2m}}{(2m)!}\binom{p-1}{m-1}\mathscr{C}^m\mathscr{D}^{p-m}[\mathbf{A}^n]. \tag{24}$$

This scheme is unconditionally stable for all Δt and $0 \le \beta \le \beta_{max}$, with β_{max} obtained from Table I, as proved in Ref. 8. Therefore, the second and fourth order two-dimensional schemes can be expressed as

$$\mathbf{A}^{n+1}-2\mathbf{A}^n+\mathbf{A}^{n-1} = -\beta^2\mathscr{C}_{xy}^{(1)}[\mathbf{A}^n]. \tag{25}$$

$$\mathbf{A}^{n+1}-2\mathbf{A}^n+\mathbf{A}^{n-1} = -\beta^2\mathscr{C}_{xy}^{(1)}[\mathbf{A}^n] - \left(\beta^2\mathscr{D}_{xy}^{(2)}-\frac{\beta^4}{12}\mathscr{C}_{xy}^{(2)}\right)\mathscr{C}_{xy}^{(1)}[\mathbf{A}^n], \tag{26}$$

where superscripts on the operators $\mathscr{C}_{xy}$ and $\mathscr{D}_{xy}$ denote level numbers. The fourth order scheme is implemented on two levels ($p = 2$). At the first level, we compute $\mathscr{C}^{(1)}$ acting on $[\mathbf{A}^n]$ using the second order scheme (25) and then at the second level apply the operators $\mathscr{C}^{(2)}$ and $\mathscr{D}^{(2)}$ to the computed $\mathscr{C}^{(1)}[\mathbf{A}^n]$ to get the fourth order correction. Using these level numbers, we can define boundary coefficients related to level k computing, (a_{xk}, b_{xk}) and (a_{yk}, b_{yk}) along the x and y directions, respectively.

B. Boundary conditions

In this section, we discuss enforcing boundary conditions with the above method. For PEC boundary conditions, we extend our

TABLE I. β_{max} as a function of the method order P.

P	2	4	6	8	10
β_{max}	2	1.4840	1.2345	1.0795	0.9715

previous work on Neumann boundary conditions to PEC.[10] On a domain $\Omega = [x_a, x_b] \times [y_a, y_b]$, outflow boundary and PEC conditions can be defined by the following:

1. PEC: $\partial_y \mathbf{A}_z(x_a, t) = 0$, $-\partial_y \mathbf{A}_z(x_b, t) = 0$, $-\partial_x \mathbf{A}_z(y_a, t) = 0$, $\partial_x \mathbf{A}_z(y_b, t) = 0$.
2. Outflow boundary condition: $\partial_t \mathbf{A}(x_a, t) = c\partial_x \mathbf{A}(x_a, t)$, $\partial_t \mathbf{A}(x_b, t) = -c\partial_x \mathbf{A}(x_b, t)$, $\partial_t \mathbf{A}(y_a, t) = c\partial_y \mathbf{A}(y_a, t)$, $\partial_t \mathbf{A}(y_b, t) = -c\partial_y \mathbf{A}(y_b, t)$.

Unlike Neumann boundary conditions, the challenge here is that PEC needs y derivatives during the $x-$sweeps and x derivatives during the $y-$sweeps. This is challenging for an ADI method because of the separation of directional information.

For a rotated square, the PEC boundary condition can use a similar approach as used in the original method for Neumann boundaries in Ref. 10 because the directions are coupled at the boundary. Rotating the domain through any angle, say θ, as shown in Fig. 2, we have the following:

(a) Left: $\cos(\theta)\partial_y \mathbf{A}_z - \sin(\theta)\partial_x A\mathbf{A}_z = 0$.
(b) Right: $-\cos(\theta)\partial_y \mathbf{A}_z + \sin(\theta)\partial_x \mathbf{A}_z = 0$.
(c) Up: $\sin(\theta)\partial_y \mathbf{A}_z - \cos(\theta)\partial_x \mathbf{A}_z = 0$.
(d) Down: $-\sin(\theta)\partial_y \mathbf{A}_z + \cos(\theta)\partial_x \mathbf{A}_z = 0$.

In Sec. III B 1, we demonstrate how to construct a solution for $\mathbf{A}$ given the Dirichlet boundaries, followed by an extension of the fixed point map for Neumann boundaries in Ref. 10 to the case of PEC to generate the Dirichlet boundary conditions (including an adaptation for the mesh aligned case), and finally give a short review of our method for outflow, which we detailed in Ref. 11.

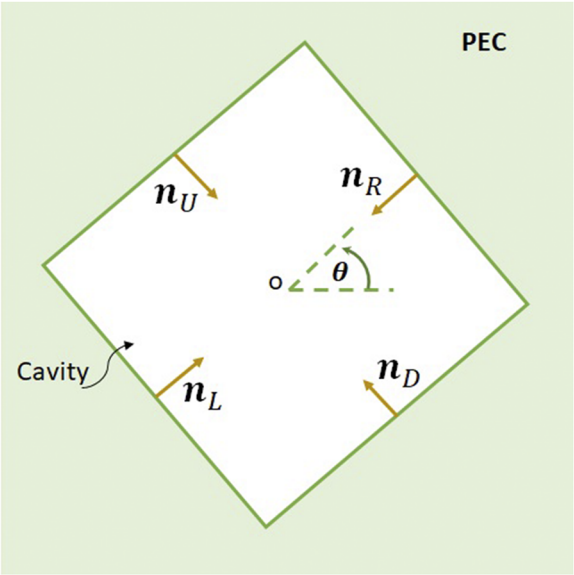

FIG. 2. A PEC square cavity rotated by an angle θ with normal vectors n_L, n_D, n_R, and n_U along the left, down, right, and up boundaries, respectively.

1. Using Dirichlet boundary condition for A to capture PEC boundary

In our embedded boundary method for Neumann, we constructed an efficient convergent Neumann to Dirichlet map, converting a boundary condition on the outward normal derivative of a function into a condition that constructs ghost points outside the domain. The method is designed such that the generated ghost points enforce the desired Neumann condition on the boundary. In Sec. III B 2, we adapt this idea to PEC, which is a condition on the tangential derivative.

We now show how to carry out the sweeps with the given values at the end of each line of the fourth order ADI method. In our 2D scenario, to construct the operator $\mathscr{L}_x^{-1}$ and $\mathscr{L}_y^{-1}$ and to build $\mathscr{C}^{(1)}$, $\mathscr{C}^{(2)}$, and $\mathscr{D}^{(2)}$, each line is treated as a 1D problem, where the lines are evaluated with $x-$sweeps and $y-$sweeps, respectively.[10] We note that the operators that make up $\mathscr{D}^{(k)}$ and $\mathscr{C}^{(k)}$ have their own boundary conditions to enforce at level k.

The objective is given the boundary condition, solve for each a_x and b_x along the $x-$direction and a_y and b_y along the $y-$direction. We will denote $\mathbf{A}(t_i, x_a) = A_L(t_i)$, $\mathbf{A}(t_i, x_b) = A_R(t_i)$, $\mathbf{A}(t_i, y_a) = A_D(t_i)$, and $\mathbf{A}(t_i, y_b) = A_U(t_i)$. Here, $[x_a, x_b]$ and $[y_a, y_b]$ are horizontal and vertical lines we make sweeps along. These permit the boundary conditions to be incorporated into the method.

As we did for wave solvers in our early work,[7] taking $A_L(t_i)$ and $A_R(t_i)$ as fixed, for the $\mathscr{L}_x^{-1}$ component of the second order term $\mathscr{C}^{(1)}$ along the line $\partial\Omega = [x_a, x_b]$, we arrive at two equations in two unknowns,

$$\begin{aligned}
&\mathbf{A}_L(t_{n+1}) + (\beta^2 - 2)\mathbf{A}_L(t_n) + \mathbf{A}_L(t_{n-1}) \\
&\quad = \beta^2\left(I\left[\mathbf{A}_L(t_n) + \frac{\mu_0}{\alpha^2}\mathbf{J}^n\right](x_a) + a_x - M_x b_x\right), \\
&\mathbf{A}_R(t_{n+1})(\beta^2 - 2)\mathbf{A}_R(t_n) + \mathbf{A}_R(t_{n-1}) \\
&\quad = \beta^2\left(I\left[\mathbf{A}_R(t_n) + \frac{\mu_0}{\alpha^2}\mathbf{J}^n\right](x_b) + M_x a_x + b_x\right),
\end{aligned}$$

where $M_x = e^{-\alpha(x_b - x_a)}$. We can rearrange the linear system into unknown and known values,

$$\begin{aligned}
a_x + M_x b_x &= w_a^P, \\
M_x a_x + b_x &= w_b^P.
\end{aligned}$$

Solving the linear system for the unknowns a_x and b_x gives

$$a_x = \frac{(w_a^P - M_x w_b^P)}{(1 - M_x^2)}, \quad b_x = \frac{(w_b^P - M_x w_a^P)}{(1 - M_x^2)}, \tag{27}$$

where

$$\begin{aligned}
w_a^P &= \frac{1}{\beta^2}\left(\mathbf{A}_L(t_{n+1}) + (\beta^2 - 2)\mathbf{A}_L(t_n) + \mathbf{A}_L(t_{n-1})\right) \\
&\quad - I\left[\mathbf{A}_L(t_n) + \frac{\mu_0}{\alpha^2}\mathbf{J}^n\right](x_a), \\
w_b^P &= \frac{1}{\beta^2}\left(\mathbf{A}_R(t_{n+1}) + (\beta^2 - 2)\mathbf{A}_R(t_n) + \mathbf{A}_R(t_{n-1})\right) \\
&\quad - I\left[\mathbf{A}_R(t_n) + \frac{\mu_0}{\alpha^2}\mathbf{J}^n\right](x_b).
\end{aligned}$$

Taking $A_D(t_i)$ and $A_U(t_i)$ as fixed for $\mathscr{L}_y^{-1}$ of the second order term $\mathscr{C}^{(1)}$ along the line $\partial\Omega = [y_a, y_b]$, we arrive at two equations in two unknowns for a_y and b_y,

$$\begin{aligned}
&\mathbf{A}_D(t_{n+1})(\beta^2-2)\mathbf{A}_D(t_n) + \mathbf{A}_D(t_{n-1}) \\
&\quad = \beta^2\left(I\left[\mathbf{A}_D + \frac{\mu_0}{\alpha^2}\mathbf{J}^n\right](y_a) + a_y + M_y b_y\right), \\
&\mathbf{A}_U(t_{n+1})(\beta^2-2)\mathbf{A}_U(t_n) + \mathbf{A}_U(t_{n-1}) \\
&\quad = \beta^2\left(I\left[\mathbf{A}_U + \frac{\mu_0}{\alpha^2}\mathbf{J}^n\right](y_b) + M_y a_y + b_y\right),
\end{aligned}$$

where $M_y = e^{-\alpha(y_b - y_a)}$. Solving the linear system for the unknowns a_y and b_y gives

$$a_y = \frac{(w_a^P - M_y w_b^P)}{(1-M_y^2)}, \quad b_y = \frac{(w_b^P - M_y w_a^P)}{(1-M_y^2)}, \tag{28}$$

where

$$\begin{aligned}
w_a^P =& \frac{1}{\beta^2}\left(\mathbf{A}_D(t_{n+1}) + (\beta^2-2)\mathbf{A}_D(t_n) + \mathbf{A}_D(t_{n-1})\right), \\
&- I\left[\mathbf{A}_D(t_n) + \frac{\mu_0}{\alpha^2}\mathbf{J}^n\right](y_a), \\
w_b^P =& \frac{1}{\beta^2}\left(\mathbf{A}_U(t_{n+1}) + (\beta^2-2)\mathbf{A}_U(t_n) + \mathbf{A}_U(t_{n-1})\right) \\
&- I\left[\mathbf{A}_U(t_n) + \frac{\mu_0}{\alpha^2}\mathbf{J}^n\right](y_b).
\end{aligned} \tag{29}$$

For the operator $\mathscr{C}^{(2)} + \mathscr{D}^{(2)}$ acting on $\mathscr{C}^{(1)}[A^n]$, which is the higher order correction, one can obtain similar equations, except that the values of $A_L(t_i)$, $A_R(t_i)$, $A_D(t_i)$, and $A_U(t_i)$ at the boundaries are explicitly zero. This is due to linearity, and the leading order operator satisfies the boundary condition exactly.

2. Embedded boundary method, an effective Neumann to Dirichlet map for creating ghost points

Given that we know how to solve for the Dirichlet boundary conditions for Eqs. (25) and (26), as in Ref. 8, we would like to extend this process to the PEC case. As discussed in Ref. 10, the reason to employ an embedded boundary approach for a differential boundary condition has to do with stability. The difference here is we need to develop an iterative map to find the homogeneous coefficients a and b for each piece of the operator instead of directly computing these coefficients.

Our embedded boundary method is based on our initial work in Ref. 10. In this paper, we developed a second order embedded boundary method for Neumann boundary conditions, which was second order accurate in time and space. Using the same ideas, we extended this method to PEC boundary conditions for the vector potential in 2D. However, in Ref. 10, direct application through the ADI of the operators is more direct. Extending this work to the fourth order involves developing an approach to solving for the boundary conditions for the operators $\mathscr{C}^{(1)}$, $\mathscr{C}^{(2)}$, and $\mathscr{D}^{(2)}$.

We use an iterative method to obtain the accurate values at the boundaries using an initial approximation and correct it by imposing our PEC boundary condition through a fixed point iteration. The fixed point iteration is a multi-step process that converts the derivative condition along the boundary to a Dirichlet boundary condition, to be set at the ghost point. These Dirichlet boundary values force the solution to satisfy the PEC condition at the boundary to within the tolerance of iteration of the fixed point method. It should be noted that the iteration is local at the boundary and does not involve re-computing the interior sweeps, making this update cost effective.

To understand the method, we start with considering a collection of uniform points with a boundary passing through (see Fig. 3). We define the ghost points to be the collection of points that are greater than one half of a grid spacing away, but less than two and a half grid spacings away, from the boundary. As in our work in Ref. 10, starting from the ghost point, we define a normal to the surface along which we will enforce the boundary condition. We define the fixed point method along this normal. In Fig. 3, the ghost point being considered is the red grid point labeled (x_G, y_G), the blue point labeled (x_B, y_B) is the boundary point, and there are three interior points needed that are related to the normal, at (x_I, y_I), (x_{II}, y_{II}), and (x_{III}, y_{III}). Their distances from the boundary point (x_B, y_B) are $\xi_I = |(x_I, y_I) - (x_B, y_B)| = \sqrt{2}\Delta s_I$, $\xi_{II} = |(x_{II}, y_{II}) - (x_B, y_B)| = \sqrt{2}\Delta s_I$, and $\xi_{III} = |(x_{III}, y_{III}) - (x_B, y_B)| = 2\Delta s_I$, where we will typically take $\Delta s_I = \sqrt{2}\Delta x$. The fixed point method starts by assuming we know $\mathbf{A}_I$, $\mathbf{A}_{II}$, $\mathbf{A}_{III}$, and $\partial_T \mathbf{A}_B$, which is taken as a solution at $\{(x_I, y_I), (x_{II}, y_{II})\text{and}(x_{III}, y_{III})\}$ and the tangential derivative of a solution at (x_B, y_B). Here, we enforce $\partial_T \mathbf{A}_B = 0$ by making $\partial_T P|_{(x_B, y_B)} = 0$, one of the conditions we use to solve for the coefficients of the Hermite–Birkhoff interpolating polynomial, $P(x, y) = c_0 + c_1 x + c_2 y + c_3 xy$. Enforcing that the Hermite–Birkhoff interpolates these points and explicitly solving for $\mathbf{A}_G$ give

$$\mathbf{A}_G^{l+1} = \Gamma_I \mathbf{A}_I^l + \Gamma_{II}\mathbf{A}_{II}^l + \Gamma_{III}\mathbf{A}_{III}^l + O(\Delta x^2), \tag{30}$$

where

$$\begin{aligned}
\Gamma_I &= 1 + \frac{\gamma_3(\gamma_4+\gamma_5) - \gamma_6(\gamma_1+\gamma_2)}{(\gamma_4\gamma_2 - \gamma_5\gamma_1)}, \\
\Gamma_{II} &= \frac{(\gamma_6\gamma_2 + \gamma_3\gamma_5)}{(\gamma_4\gamma_2 - \gamma_5\gamma_1)}, \\
\Gamma_{III} &= \frac{(\gamma_6\gamma_1 + \gamma_3\gamma_4)}{(\gamma_4\gamma_2 - \gamma_5\gamma_1)},
\end{aligned} \tag{31}$$

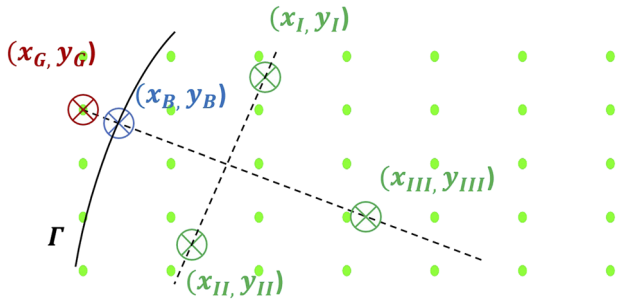

FIG. 3. Geometry of the 2D embedded PEC boundary method with a boundary point (x_B, y_B), ghost point (x_G, y_G), and interpolation points (x_I, y_I), (x_{II}, y_{II}), and (x_{III}, y_{III}).

where

$$\begin{aligned}\gamma_1 &= x_{II} - x_I + y_{II} - y_I,\\ \gamma_2 &= x_I - x_{III} + y_I - y_{III},\\ \gamma_3 &= x_G - x_I + y_G - y_I,\\ \gamma_4 &= (x_{II} - x_I)(x_B - y_B) + x_I y_I,\\ \gamma_5 &= (x_I - x_{III})(x_B - y_B) + x_{III} y_{III},\\ \gamma_6 &= (x_B - y_B)(x_G - x_1) + x_G(y_G - y_1).\end{aligned} \tag{32}$$

We have introduced additional notation of $l+1$ and l, which designate how we will set up a fixed point iteration using the Hermite–Birkhoff interpolant. The iteration itself is a simple first order fixed point method. Let $\mathbf{w}^{(l)}$ and $\mathbf{w}^{(l+1)}$ be the solutions at the lth and $(l+1)$th iterations. We choose a tolerance *tol*, and maximum number of allowed iterations *mit*, and define a stopping criterion as $|\mathbf{w}^{(l+1)} - \mathbf{w}^{(l)}|_\infty < tol$ or $nit > mit$, where *nit* is the current iteration number. We will now discuss the quasi-local process for obtaining $\mathbf{w}^{(l+1)}$ given the rest of the information.

For each term in the operator on the right-hand side of the second order solution given by Eq. (25) or fourth order in time solution given by Eq. (26), we must identify the correct ghost point that allows that term to guarantee that the tangential derivative is zero along the boundary. Taking the $\mathscr{C}^{(1)}[\mathbf{A}^n]$ operator and expanding the operator out, we have $\mathscr{C}^{(1)}[\mathbf{A}^n] = \mathscr{L}_x^{-1}[\mathbf{A}^n] + \mathscr{L}_y^{-1}[\mathbf{A}^n] - \mathscr{L}_y^{-1}\mathscr{L}_x^{-1}[\mathbf{A}^n] - \mathscr{L}_x^{-1}\mathscr{L}_y^{-1}[\mathbf{A}^n]$. The fixed point iteration identifies the ghost point for $\mathbf{w_x} = \mathscr{L}_x^{-1}[\mathbf{A}^n]$, $\mathbf{w_y} = \mathscr{L}_y^{-1}[\mathbf{A}^n]$, $\mathbf{w_{xy}} = \mathscr{L}_x^{-1}[\mathbf{w_y}]$, and $\mathbf{w_{yx}} = \mathscr{L}_y^{-1}[\mathbf{w_x}]$ such that when solving for the boundary correction terms, the operator satisfies $\partial_T \mathbf{w_x}|_{\partial\Omega} = 0$, $\partial_T \mathbf{w_y}|_{\partial\Omega} = 0$, $\partial_T \mathbf{w_{xy}}|_{\partial\Omega} = 0$, and $\partial_T \mathbf{w_{yx}}|_{\partial\Omega} = 0$.

Let us consider $\mathbf{w_x}$ and $\mathbf{w_{yx}} = \mathscr{L}_y^{-1}[\mathbf{w_x}]$, knowing that $\mathbf{w_y}$ and $\mathbf{w_{xy}}$ are similar. The iterative process for $\mathbf{w_x}$ starts by making an initial guess at the direct boundary values $\mathbf{w_x}|_{\partial\Omega}$ using an extrapolate in time $\mathbf{w_x}^{n+1,(0)} \approx 3\mathbf{w_x}^n - 3\mathbf{w_x}^{n-1} + \mathbf{w_x}^{n-2}$ at each boundary point (Algorithm 1).

Given the update on $\mathbf{w_x}$, the next step is to consider $\mathbf{w_{yx}} = \mathscr{L}_y^{-1}[\mathbf{w_x}]$. The process starts with the initial guess $\mathbf{w_x}^{n+1,(0)} \approx 3\mathbf{A}^n - 3\mathbf{A}^{n-1} + \mathbf{A}^{n-2}$ at the boundary (Algorithm 2).

The same process is done for $\mathbf{w_y}$ and $\mathbf{w_{xy}}$. The Hermite–Birkhoff interpolant enforces that the tangential derivative is zero for $\mathbf{w_x}$, $\mathbf{w_y}$, $\mathbf{w_{xy}}$, and $\mathbf{w_{yx}}$ such that $\mathscr{C}^{(1)}[\mathbf{A}^n]$ satisfies the tangential derivative condition on $\partial\Omega$ to within tolerance.

For the fourth order formulation, we compute the boundary conditions for $\mathscr{C}^{(1)}[\mathbf{A}^n]$, and then, we repeat the process for $\mathscr{C}^{(2)}$ and $\mathscr{D}^{(2)}$ acting on $\mathscr{C}^{(1)}[\mathbf{A}^n]$.

Further as detailed in Ref. 10, we need to include an artificial dissipation term in order to maintain stability for these embedded boundary domains. Thus, we have

$$\begin{aligned}\mathbf{A}^{n+1} - 2\mathbf{A}^n + \mathbf{A}^{n-1} &= -\beta^2 \mathscr{C}_{xy}^{(1)}[\mathbf{A}^n]\\ &\quad - \left(\beta^2 \mathscr{D}_{xy}^{(2)} - \frac{\beta^4}{12}\mathscr{C}_{xy}^{(2)}\right)\mathscr{C}_{xy}^{(1)}[\mathbf{A}^n] + \varepsilon \mathscr{C}_{xy}^{(3)}[\mathbf{A}^{n-1}],\end{aligned}$$

where ε is an artificial dissipation coefficient that satisfies $0 < \varepsilon < 1$. The value of $\mathscr{C}_{xy}^{(3)}[\mathbf{A}^n]$ at the previous time step can be used in place of $\mathscr{C}_{xy}^{(3)}[\mathbf{A}^{n-1}]$ at the current time step, and we need to go to one more computing level to obtain $\mathscr{C}_{xy}^{(3)}[\mathbf{A}^n]$ (see Ref. 11 for further details). We obtain $\mathbf{A}_I$, $\mathbf{A}_{II}$, and $\mathbf{A}_{III}$ approximately, using the bilinear interpolation. Suppose that the interpolation point $\mathbf{A}_I$ lies in a rectangular cell with corners (x_i, y_j), (x_{i+1}, y_j), (x_{i+1}, y_{j+1}), and (x_i, y_{j+1}). Then, we have the following approximation for $\mathbf{A}_I$:

$$\mathbf{A}_I = w_1 \mathbf{A}_{i,j} + w_2 \mathbf{A}_{i+1,j} + w_3 \mathbf{A}_{i+1,j+1} + w_4 \mathbf{A}_{i,j+1}, \tag{33}$$

where

$$w_1 = \frac{(x_{i+1} - x_I)(y_{j+1} - y_I)}{\Delta x \Delta y}, \qquad w_2 = \frac{(x_I - x_i)(y_{j+1} - y_I)}{\Delta x \Delta y},$$

ALGORITHM 1. Compute $\mathbf{w_x}\left(= L_x^{-1}[\mathbf{A}]\right)$.

1: Compute the interior sweep (particular solution) $I_x^{n+1} = I_x[\mathbf{A}^n]$
2: Initial guess $\mathbf{w_x}^{n+1(0)} \approx 3\mathbf{w_x}^n - 3\mathbf{w_x}^{n-1} + \mathbf{w_x}^{n-2}$
3: Initialize the iteration counter $nit = 0$
4: **repeat**
5: **for** $k = 1$ to n_y **do**
6: Compute $\mathbf{w_x}_{I(k)}$, and $\mathbf{w_x}_{II(k)}$, at the interpolation points $(x_{I(k)}, y_{(k)})$, and $(x_{II(k)}, y_{(k)})$ and as well as at the interpolation points along the right boundary using bilinear interpolation
7: Compute solution at the ghost points $(x_{aG(k)}, y_{(k)})$ and $(x_{bG(k)}, y_{(k)})$ using the Hermite–Birkhoff interpolant
8: Compute homogeneous coefficients $a_{1(k)}$ and $b_{1(k)}$ using the fact $\mathbf{w_x}^{n+1(l)} = I^{n+1} + a_{x(k)} e^{-\alpha(x - x_{aG(k)})} + b_{x(k)} e^{-\alpha(x_{bG(k)} - x)}$
9: Compute solution and update boundary stencil $\mathbf{w_x}^{n+1(l+1)} = I^{n+1} + a_{x(k)} e^{-\alpha(x - x_{aG(k)})} + b_{x(k)} e^{-\alpha(x_{bG(k)} - x)}$
10: **end for**
11: $nit = nit + 1$
12: **until** $(|\mathbf{w_x}^{n+1(l+1)} - \mathbf{w_x}^{n+1(l)}| < tol$ OR $nit > mit)$

ALGORITHM 2. Compute $\mathbf{w_{yx}}\left(= L_y^{-1}[\mathbf{w_x}]\right)$.

1: Compute the interior sweep (particular solution) $I_y^{n+1} = I_y[\mathbf{w_x}^n]$
2: Initial guess $\mathbf{w_{yx}}^{n+1(0)} \approx 3\mathbf{A}^n - 3\mathbf{A}^{n-1} + \mathbf{A}^{n-2}$
3: Initialize the iteration counter $nit = 0$
4: **repeat**
5: **for** $k = 1$ to n_x **do**
6: Compute $\mathbf{w_{yx}}_{I(k)}$, and $\mathbf{w_{yx}}_{II(k)}$, at the interpolation points $(x_{(k)}, y_{I(k)})$, and $(x_{(k)}, y_{II(k)})$ and as well as at the interpolation points along the right boundary using bilinear interpolation
7: Compute solution at the ghost points $(x_{(k)}, y_{aG(k)})$ and $(x_{(k)}, y_{bG(k)})$ using the Hermite–Birkhoff interpolant
8: Compute homogeneous coefficients $a_{1(k)}$ and $b_{1(k)}$ using the fact $\mathbf{w_{yx}}^{n+1(l)} = I^{n+1} + a_{y(k)} e^{-\alpha(y - y_{aG(k)})} + b_{y(k)} e^{-\alpha(y_{bG(k)} - y)}$
9: Compute solution and update boundary stencil $\mathbf{w_{yx}}^{n+1(l+1)} = I^{n+1} + a_{y(k)} e^{-\alpha(y - y_{aG(k)})} + b_{y(k)} e^{-\alpha(y_{bG(k)} - y)}$
10: **end for**
11: $nit = nit + 1$
12: **until** $(|\mathbf{w_{yx}}^{n+1(l+1)} - \mathbf{w_{yx}}^{n+1(l)}| < tol$ OR $nit > mit)$

$$w_3 = \frac{(x_I - x_i)(y_I - y_j)}{\Delta x \Delta y}, \qquad w_4 = \frac{(x_{i+1} - x_I)(y_I - y_j)}{\Delta x \Delta y}. \tag{34}$$

We show how the inverse operators are performed along 2D Cartesian lines for the 2D A6 magnetron in Fig. 4. The endpoints of these lines are not restricted to residing at mesh points and can always be chosen to lie on the boundary $\partial\Omega$. We first perform x-sweeps along with x-lines with actual boundary ends and obtain the intermediate solution and then apply y-sweeps over the intermediate solution, along with y-lines with actual boundary ends. The xy-sweeps form a complete boundary of the A6 magnetron [Fig. 4(c)]. Similarly, we can perform y-sweeps first, x-sweeps next, and take an average to obtain the solution.

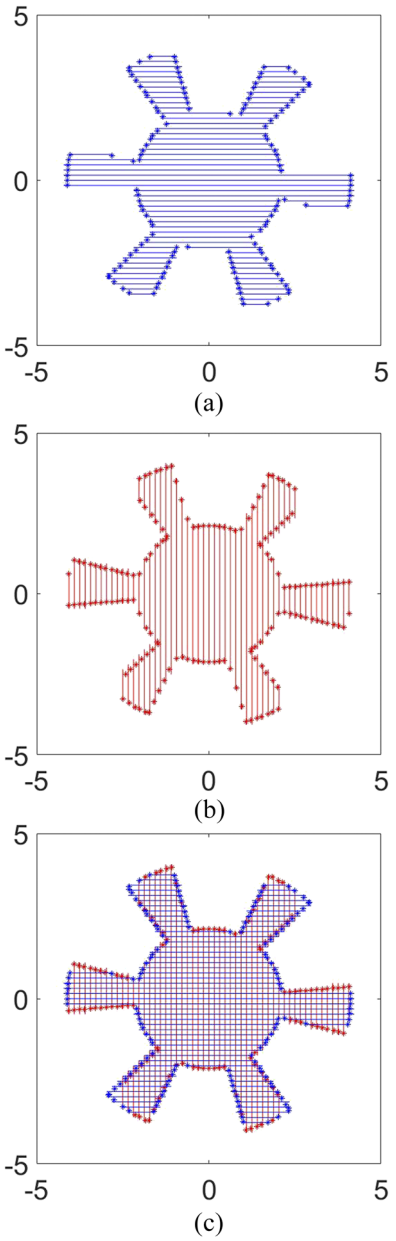

FIG. 4. (a) x-, (b) y-, and (c) xy-lines with exact A6 magnetron boundary points on the mesh lines that are used for the ADI x (a) and y (b) sweeps.

3. Outflow boundary condition

For the open boundary cases, we derive an outflow boundary condition by extending the domain $\Omega = [x_a, x_b] \times [y_a, y_b]$ into $\Omega = [x_a - ct_n, x_b + ct_n] \times [y_a - ct_n, y_b + ct_n]$ and switch the spatial integration with the time integration at the boundary points x_a, x_b, y_a, and y_b. Consider a one-dimensional scheme as explained in Ref. 11.

Let us work with the right boundary x_b first,

$$\begin{aligned} b_x^n &:= \frac{\alpha}{2}\int_{x_b}^{x_b+ct_n} e^{-\alpha(x'-x_b)}\mathbf{A}^n(x')dx' \\ &= \frac{\alpha}{2}\int_0^{ct_n} e^{-\alpha(x')}\mathbf{A}^n(x_b + x', t_n)dx' \\ &= \frac{\alpha c}{2}\int_0^{t_n} e^{-\alpha(cs)}\mathbf{A}^n(x_b, t_n - s)ds. \end{aligned} \tag{35}$$

This can be written in a recursive form,

$$b_x^n = \frac{\beta}{2}\int_0^1 e^{-\beta z}\mathbf{A}^n(x_b, t_n - z\Delta t)dz + e^{-\beta}b_x^{n-1}.$$

We construct a time interpolant at the right boundary ($x > x_b$) using a Taylor series expression and truncate higher order error terms to obtain fourth order accuracy using a five point backward finite difference stencil. We obtain

$$\begin{aligned} \mathbf{A}^n(x_b, t_n - z\Delta t) \approx{}& \mathbf{A}^n(x_b) - z\left(\frac{25}{12}\mathbf{A}^n(x_b) - 4\mathbf{A}^{n-1}(x_b)\right. \\ &\left. + 3\mathbf{A}^{n-2}(x_b) - \frac{4}{3}\mathbf{A}^{n-3}(x_b) + \frac{1}{4}\mathbf{A}^{n-4}(x_b)\right) \\ &+ \frac{z^2}{2}\left(\frac{35}{12}\mathbf{A}^n(x_b) - \frac{26}{3}\mathbf{A}^{n-1}(x_b) + \frac{19}{2}\mathbf{A}^{n-2}(x_b)\right. \\ &\left. - \frac{14}{3}\mathbf{A}^{n-3}(x_b) + \frac{11}{12}\mathbf{A}^{n-4}(x_b)\right) - \frac{z^3}{6}\left(\frac{5}{2}\mathbf{A}^n(x_b)\right. \\ &\left. - 9\mathbf{A}^{n-1}(x_b) + 12\mathbf{A}^{n-2}(x_b) - 7u\mathbf{A}^{n-3}(x_b)\right. \\ &\left. + \frac{3}{2}\mathbf{A}^{n-4}(x_b)\right) + \frac{z^4}{24}\left(\mathbf{A}^n(x_b) - 4\mathbf{A}^{n-1}(x_b)\right. \\ &\left. + 6\mathbf{A}^{n-2}(x_b) - 4\mathbf{A}^{n-3}(x_b) + \mathbf{A}^{n-4}(x_b)\right). \end{aligned} \tag{36}$$

We integrate this expression analytically using Lemma III.1 (see Ref. 7 for a proof).

Lemma III.1 *For integers $m \geq 0$ and real $v > 0$,*

$$\Psi_m := v\int_0^1 \frac{z^m}{m!}e^{-vz}dz = \frac{1}{v^m}\left(1 - e^{-v}P_m(v)\right),$$

where $P_m(v) = \sum_{l=0}^{m}\frac{v^l}{l!}$ is the Taylor series expansion of order m of e^v.

We arrive at

$$b^n = e^{-\beta}b^{n-1} + \gamma_0\mathbf{A}^n(x_b) + \gamma_1\mathbf{A}^{n-1}(x_b) + \gamma_2\mathbf{A}^{n-2}(x_b)$$

$$+ \gamma_3 \mathbf{A}^{n-3}(x_b) + \gamma_4 \mathbf{A}^{n-4}(x_b), \tag{37}$$

where

$$\begin{aligned}
\gamma_0 &= \Psi_0(\beta) - \frac{25}{12}\Psi_1(\beta) + \frac{35}{12}\Psi_2(\beta) - \frac{5}{2}\Psi_3(\beta) + \Psi_4(\beta),\\
\gamma_1 &= 4\Psi_1(\beta) - \frac{26}{3}\Psi_2(\beta) + 9\Psi_3(\beta) - 4\Psi_4(\beta),\\
\gamma_2 &= -3\Psi_1(\beta) + \frac{19}{2}\Psi_2(\beta) - 12\Psi_3(\beta) + 6\Psi_4(\beta),\\
\gamma_3 &= \frac{4}{3}\Psi_1(\beta) + \frac{14}{3}\Psi_2(\beta) + 7\Psi_3(\beta) - 4\Psi_4(\beta),\\
\gamma_4 &= -\frac{1}{4}\Psi_1(\beta) + \frac{11}{12}\Psi_2(\beta) - \frac{3}{2}\Psi_3(\beta) + \Psi_4(\beta).
\end{aligned} \tag{38}$$

Likewise, by considering the left boundary $x < a$, we get

$$\begin{aligned}
a^n &= e^{-\beta} a^{n-1} + \gamma_0 \mathbf{A}^n(x_a) + \gamma_1 \mathbf{A}^{n-1}(x_a) + \gamma_2 \mathbf{A}^{n-2}(x_a)\\
&\quad + \gamma_3 \mathbf{A}^{n-3}(x_a) + \gamma_4 \mathbf{A}^{n-4}(x_a).
\end{aligned} \tag{39}$$

IV. EXTENSION TO 3D

As in 2D, after we discretize the vector and scalar wave equations in time, we are left with a modified Helmholtz operator. The three-dimensional extension of the scheme can be derived by decomposing the modified Helmholtz operator using ADI splitting in 3D, as we did in 2D,

$$\begin{aligned}
1 - \frac{1}{\alpha^2}\nabla^2 &= 1 - \frac{1}{\alpha^2}\left(\frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} + \frac{\partial^2}{\partial z^2}\right)\\
&= \left(1 - \frac{1}{\alpha^2}\frac{\partial^2}{\partial x^2}\right)\left(1 - \frac{1}{\alpha^2}\frac{\partial^2}{\partial y^2}\right)\left(1 - \frac{1}{\alpha^2}\frac{\partial^2}{\partial z^2}\right)\\
&\quad + \frac{1}{\alpha^4}\left(\frac{\partial^4}{\partial x^2 \partial y^2} + \frac{\partial^4}{\partial x^2 \partial z^2} + \frac{\partial^4}{\partial y^2 \partial z^2}\right)\\
&\quad - \frac{1}{\alpha^6}\frac{\partial^6}{\partial x^2 \partial y^2 \partial z^2}\,.
\end{aligned}$$

Hence, the temporal approximation,

$$1 - \frac{1}{\alpha^2}\nabla^2 = \mathscr{L}_x \mathscr{L}_y \mathscr{L}_z + \mathcal{O}\left((c\Delta t)^4\right), \tag{40}$$

where, here, in addition to the modified Helmholtz operator $\mathscr{L}_x$ and $\mathscr{L}_y$, we use the operator $\mathscr{L}_z$ in the z direction. Now, we can express the semi-discrete equation,

$$\begin{aligned}
&\mathscr{L}_x \mathscr{L}_y \mathscr{L}_z \left[\beta^2 \mathbf{A}^n + \mathbf{A}^{n+1} - 2\mathbf{A}^n + \mathbf{A}^{n-1}\right]\\
&\quad = \beta^2 \mathbf{A}^n + \frac{\mu_0 \beta^2}{\alpha^2}\mathbf{J}^n(\mathbf{k}, t).
\end{aligned} \tag{41}$$

Upon inverting the modified Helmholtz operators $\mathscr{L}_x$, $\mathscr{L}_y$, and $\mathscr{L}_z$, we can obtain a 3D version of the semi-discrete solution,

$$\begin{aligned}
&\mathbf{A}^{n+1} - 2\mathbf{A}^n + \mathbf{A}^{n-1}\\
&\quad = -\beta^2 \mathscr{D}^{(1)}_{xyz}[\mathbf{A}^n] + \beta^2 \mathscr{L}_z^{-1}\mathscr{L}_y^{-1}\mathscr{L}_x^{-1}\left[\frac{\mu_0}{\alpha^2}\mathbf{J}^n\right](\mathbf{k}, t),
\end{aligned} \tag{42}$$

where the three-dimensional second order operator is

$$\mathscr{D}^{(1)}_{xyz}[\mathbf{A}] := \mathbf{A} - \mathscr{L}_z^{-1}\mathscr{L}_y^{-1}\mathscr{L}_x^{-1}[\mathbf{A}]\,.$$

As in 2D, there are high order extensions. The fourth order extension in 3D needs the addition of the 3D operator $\mathscr{C}_{xyz}$,

$$\mathscr{C}_{xyz} := \mathscr{L}_y^{-1}\mathscr{L}_z^{-1}\mathscr{D}_x + \mathscr{L}_z^{-1}\mathscr{L}_x^{-1}\mathscr{D}_y + \mathscr{L}_x^{-1}\mathscr{L}_y^{-1}\mathscr{D}_z.$$

The second and fourth order 3D schemes are as follows:

$$\mathbf{A}^{n+1} - 2\mathbf{A}^n + \mathbf{A}^{n-1} = -\beta^2 \mathscr{C}^{(1)}_{xyz}[\mathbf{A}^n], \tag{43}$$

$$\begin{aligned}
\mathbf{A}^{n+1} - 2\mathbf{A}^n + \mathbf{A}^{n-1} &= -\beta^2 \mathscr{C}^{(1)}_{xyz}[\mathbf{A}^n]\\
&\quad - \left(\beta^2 \mathscr{D}^{(2)}_{xyz} - \frac{\beta^4}{12}\mathscr{C}^{(2)}_{xyz}\right)\mathscr{C}^{(1)}_{xyz}[\mathbf{A}^n].
\end{aligned} \tag{44}$$

This fourth order scheme is implemented on two levels as explained in the 2D scheme and performs x–, y–, and z–sweeps as needed based on the operators used.

The general approach for implementation of the 3D ADI scheme follows the steps given below. Assume that the number of x, y, and z lines is n_x, n_y, and n_z, respectively.

At each time step,

1. Perform the x-sweep: Operate $\mathscr{L}_x^{-1}$ on $\mathbf{A}^n(x, y_i, z_k) + \frac{\mu_0}{\alpha^2}\mathbf{J}^n(x, y_i, z_k)$ along x and store the result into a temporary variable $\mathbf{w_x}(x, y_i, z_k)$ for $1 \le k \le n_z$ and $1 \le i \le n_y$. The boundary conditions are imposed at $x = x_{a(ik)}$ and $x_{b(ik)}$.
2. Perform the y-sweep: Operate $\mathscr{L}_y^{-1}$ on $\mathbf{w_x}(x_j, y, z_k)$ along y and store the result into a temporary variable $\mathbf{w_{yx}}(x_j, y, z_k)$ for $1 \le k \le n_z$ and $1 \le j \le n_x$. The boundary conditions are imposed at $y = y_{a(jk)}$ and $y_{b(jk)}$.
3. Perform the z-sweep: For $1 \le i \le n_y$ and $1 \le j \le n_x$ using $\mathbf{w_{yx}}(x_j, y_i, z)$, solve for the equation $\mathbf{A}^{n+1} = 2\mathbf{A}^n - \mathbf{A}^{n-1} - \beta^2 \mathscr{D}_{xyz}$, where $\mathscr{D}_{xyz} = \mathbf{A}^n - \mathscr{L}_z^{-1}[\mathbf{w_{yx}}]$. The boundary conditions are now applied at $z = z_{a(ij)}$ and $z_{b(ij)}$.

For an example of 3D modeling, We show the geometrical setup of the 3D A6 magnetron (Fig. 5), indicating key on/off mesh grid points that are involving the application of the scheme.

A. 3D PEC boundary condition

We start by noting that the 3D boundary condition for outflow is exactly the same as in 2D, so we will not go over that hear. However, PEC would require additional work, which we outline below.

In Sec. III, we exploited the fact that in 2D $\partial_T A_B = 0$, to develop our method for ghost points. In 3D, the conditions are more complicated because the boundary condition $\mathbf{B}_\perp = 0$ means that the boundary condition on the vector potential mixes the scalar components of $\mathbf{A}$. Here, we outline a possible method for extending these ideas by considering one boundary of a 3D PEC box as an example.

Consider a Cartesian box with PEC boundaries and one boundary set along $z = 0$. The $\mathbf{B}_\perp = 0$ at $z = 0$ gives

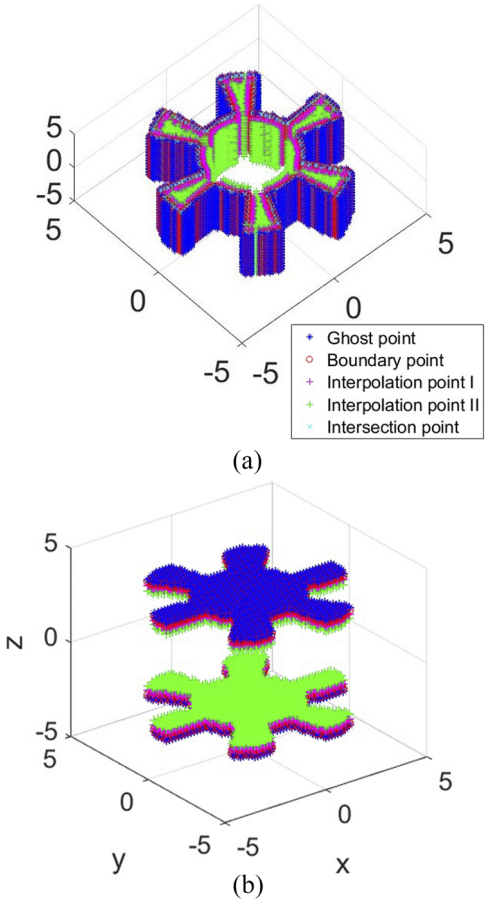

FIG. 5. Key points used for the simulation of the 3D A6 magnetron: intersection, boundary, ghost, and interpolation points required to obtain (a) $x-$ and $y-$, and (b) $z-$sweeps.

$$\partial_x A_2 - \partial_y A_1 = 0.$$

This means that the extension of the method proposed in Sec. IV would need to enforce $\partial_x A_2 = \partial_y A_1$ at the $z = 0$ boundary. Here, we develop the exact same process we did in Sec. III, except for the following key changes: we would approximate $\partial_x A_2$ and $\partial_y A_1$ by computing finite difference approximations to $D_x A_2 = \partial_x A_2$ and $D_y A_1 = \partial_y A_1$ at the boundary using interiors data; we set $g = \frac{1}{2}(D_x A_2 + D_y A_1)$; we would have our Hermite–Birkhoff interpolant pass through $\partial_T A_B = g$ instead of zero for both A_1 and A_2; we would compute ghost points in the z direction for both A_1 and A_2 using the Hermite–Birkhoff interpolant; we would update the solution A_1 and A_2 using the ghost points; in additional iterations, we would use successive over relaxation, i.e., $g = (1 - \tilde{\alpha})g + \tilde{\alpha}\frac{1}{2}(D_x A_2 + D_y A_1)$.

For a PEC box, this process would be done at each of the boundaries. The generalization of the method to other geometry would follow what we have done in this paper in 2D for geometry. The full development of this method in 3D is the subject of our next paper.

V. NUMERICAL EXPERIMENTS

A. Square cavity rotated through different angles

For this numerical experiment, we chose a square cavity bounded by a PEC, placed a point source at the center of the cavity, and performed a ping test and mode analysis. First, we computed the vector potential **A** at a point inside the box using our scheme, then computed the derived frequency by taking FFT over the time history of the measured **A**, and finally analyzed the frequency modes for varying CFL value, step size, and rotation angle.

We chose a square box $21 \times 21\ \text{cm}^2$ with PEC boundaries in a domain $(\Omega = [-21\ \text{cm}, 21\ \text{cm}]^2)$, placed a point source 1 at the center of the box, $(0, 0)$, and turned on it for a single time step $t = \Delta t$ ns. The vector potential is measured at the point $(3.36\ \text{cm}, 3.36\ \text{cm})$ for the time period $t = [\Delta t\ \text{ns}, T\ \text{ns}]$. The derived frequencies for different CFL values (0.5, 1.0, 2.0), rotation angles (0°, 31.42°, 45°), and resolutions are summarized in Tables II and III. Table III consists of frequencies obtained for the CFL value 1. Here, we set the wave speed $c = 30\ \text{Gcms}^{-1}$, averaging parameter $\beta = 1.4$, and dissipation coefficient $\varepsilon = 0.1$. Figure 6 shows the frequency distribution for $\theta = 31.42°$, and Fig. 7 shows a frequency mode computed for different resolutions with CFL 1. We see clear convergence to the analytically computed 1-1 mode fundamental frequency 1GHz.

Figure 8 shows the time evolution of the potential **A** generated by a point source $\sin(2\pi ft)$ with $f = 1$GHz placed at the center of the box. We chose the same domain as used in the previous test and set the grid size to be 100×100, time step size $\Delta t = 7$ ps.

TABLE II. Frequency (in GHz) obtained at the point $(3.36\ \text{cm}, 3.36\ \text{cm})$ using the ping test.

	$CFL = 0.5,\ T = 50$ ns			$CFL = 2,\ T = 200$ ns		
θ	45°	31.42°	0°	45°	31.42°	0°
n						
50^2	0.98	0.99	0.96	0.98	0.98	0.96
100^2	1.00	1.00	0.98	1.00	0.99	0.98
150^2	1.00	1.00	0.98	0.99	1.00	0.98
200^2	1.00	1.00	1.00	1.00	1.00	0.99
250^2	1.00	1.00	1.00	1.00	1.00	1.00

TABLE III. Frequency (in GHz) obtained at the point $(3.36\ \text{cm}, 3.36\ \text{cm})$ using the ping test for CFL 1.

θ	0°	31.42°	45°
n			
50^2	$9.599\,802 \times 10^{-1}$	$9.899\,800 \times 10^{-1}$	$9.899\,800 \times 10^{-1}$
100^2	$9.899\,800 \times 10^{-1}$	$9.999\,800 \times 10^{-1}$	$9.999\,800 \times 10^{-1}$
150^2	$9.899\,800 \times 10^{-1}$	$9.999\,800 \times 10^{-1}$	$9.999\,800 \times 10^{-1}$
200^2	$1.000\,015 \times 10^{0}$	$1.000\,115 \times 10^{0}$	$1.000\,015 \times 10^{0}$
250^2	$1.000\,008 \times 10^{0}$	$1.000\,108 \times 10^{0}$	$1.000\,008 \times 10^{0}$

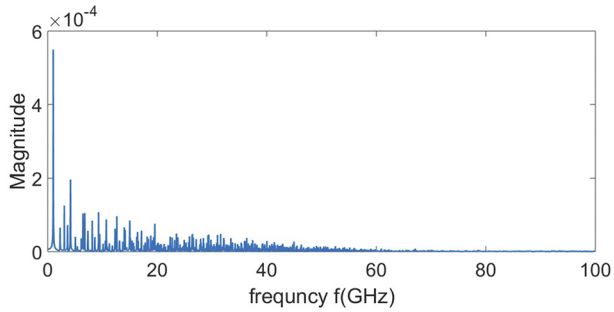

FIG. 6. Frequency distribution for 31.42° rotated 21 × 21 cm^2 square cavity computed using the measured vector potential at the point (3.36 cm, 3.36 cm) for the impulse response, h = 0.084 cm.

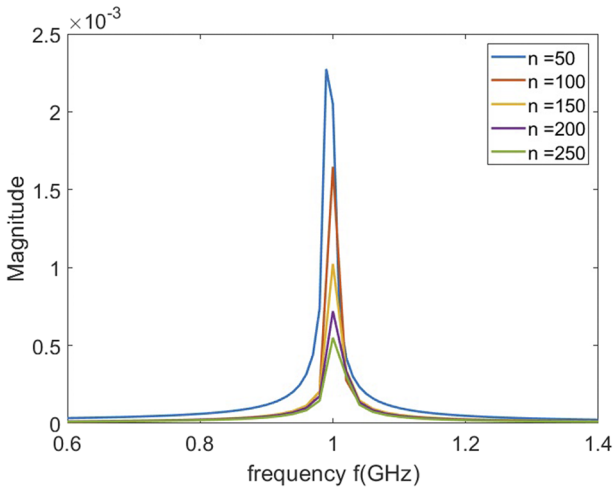

FIG. 7. A strong fundamental mode for 31.42° rotated 21 × 21 cm^2 square cavity computed for different resolutions.

1. Convergence studies and error analysis

We evaluate the consistency of our scheme via (1) the ping test and (2) space–time convergence studies for the magnetic vector potential **A** in the PEC square cavity.

First, we perform a convergence study given the initial condition $\sin\left(\frac{m\pi x}{L}\right)\sin\left(\frac{n\pi y}{H}\right)$ with frequency mode 3–2 ($m = 3, n = 2$) over a square domain $[0\text{ cm}, 21\text{ cm}]^2$ and compare our numerical solution to the exact solution given by $\mathbf{A}(t,x,y) = \cos\left(c\sqrt{(\frac{m\pi}{L})^2 + (\frac{n\pi}{H})^2}t\right)\sin\left(\frac{m\pi x}{L}\right)\sin\left(\frac{n\pi y}{H}\right)$, where the wave speed c is chosen to be 30 Gcms^{-1}. Our L_2 norm error plots of the solution at time $T = 1.0$ ns with varying resolutions and a fixed CFL(=1) for the cases of mesh aligned and nonaligned (rotated by 45°) square cavities show second-order accuracy [Fig. 9(a)]. We also note that the tangential derivative along the boundary converges with third order accuracy [Fig. 9(b)].

For the second evaluation, we compute error with our fundamental frequency computation using the ping test explained above

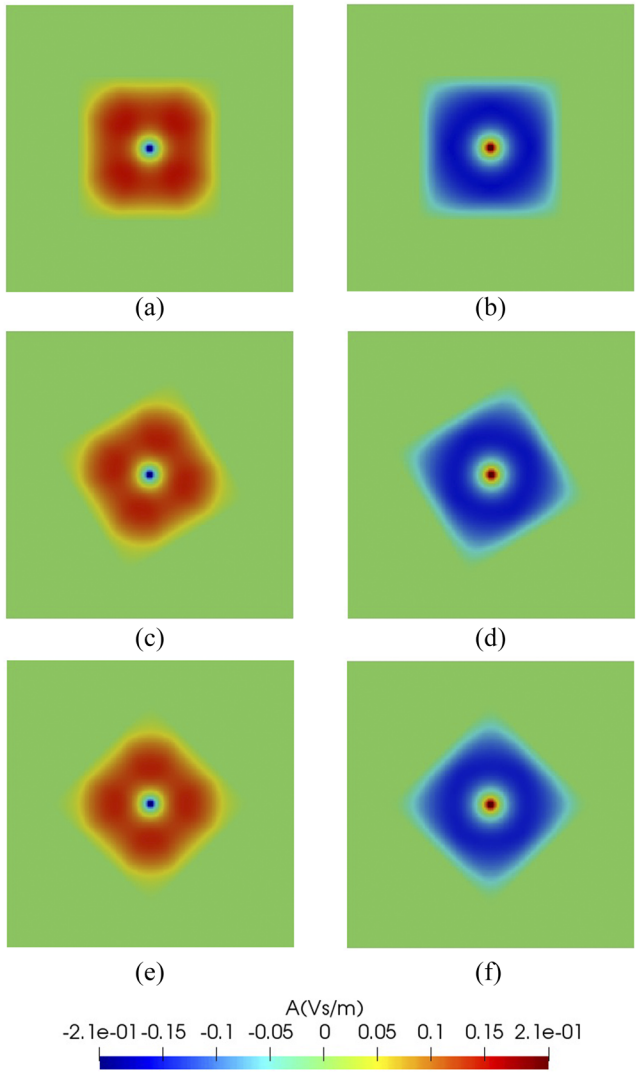

FIG. 8. Time evolution of a point source field $\sin(2\pi ft)$ with $f = 1$GHz placed at the center of a PEC square box of grid size 100 × 100 rotated by the angles 0° [shown in (a) and (b)], 31.42° [shown in (c) and (d)], 45° [shown in (e) and (f)], and time step size $\Delta t = 7$ ps.

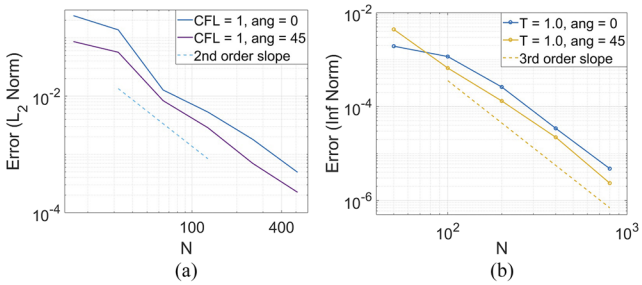

FIG. 9. Convergence plots for (a) space using entire solution and (b) space using boundary derivative on a PEC square domain $[0\text{ cm}, 21\text{ cm}]^2$. This study measures the L_2 norm of the error at time $T = 1.0$ ns compared with the analytical solution for the cases of mesh aligned and nonaligned (rotated by 45°).

TABLE IV. Numerical error in the frequency computation using the ping test at the point (3.36 cm, 3.36 cm).

	$CFL = 0.5$, $T = 50$ ns		$CFL = 2$, $T = 200$ ns	
θ	31.42°	0°	31.42°	0°
n				
50	$1.001\,980 \times 10^{-2}$	$4.501\,910 \times 10^{-2}$	$1.501\,970 \times 10^{-2}$	$4.501\,910 \times 10^{-2}$
100	$1.999\,960 \times 10^{-5}$	$2.001\,960 \times 10^{-2}$	$5.019\,900 \times 10^{-3}$	$2.001\,960 \times 10^{-2}$
150	$1.999\,960 \times 10^{-5}$	$2.001\,960 \times 10^{-2}$	$1.999\,960 \times 10^{-3}$	$1.501\,970 \times 10^{-2}$
200	$1.999\,960 \times 10^{-5}$	$1.500\,023 \times 10^{-5}$	$5.015\,075 \times 10^{-4}$	$4.985\,075 \times 10^{-4}$
250	$8.000\,006\,4 \times 10^{-6}$	$8.000\,064 \times 10^{-6}$	$5.008\,0040 \times 10^{-5}$	$8.000\,064 \times 10^{-6}$

and summarize it in Table IV. The error is the difference between the frequencies (1-1 mode) obtained analytically (= 1GHz) and numerically using our scheme.

Finally, we perform a self-refinement study for time and space convergence using a point source. Figure 10 shows the time and space convergence plots for the cases of mesh aligned and non-aligned (rotated by 45°) square cavities with the same configuration as used in the above study done for the time evolution of a point source field. For the time convergence test, we maintain the resolution at 320×320 and reduce the time step size $dt = 10.9$ ps–0.3418 ps. For the space convergence test, we maintain the CFL value as 1 and reduce the spatial step size $dx = 1.313$ cm–0.082 cm. We obtained fourth order convergence in time and second order in space.

B. Square cavity with a leak (diffraction Q)

In the second numerical experiment, an open boundary is placed along the center of the right edge of the square cavity, and the diffraction quality factor Q is obtained for an oscillating point source $[\sin(2\pi ft)$, $f = 1$GHz$]$ placed at the center. Every configuration was the same as above, except for imposing an outflow boundary condition along the open boundary and keeping the Gaussian pulse running for the entire time period $[\Delta t$ ns, T ns$]$. The vector potential $\mathbf{A}$ is measured at the point (3.36 cm, 3.36 cm), and the computation of Q is done for aligned and nonaligned mesh cases. For nonaligned meshes, we rotate the square by an angle $\theta = 45°$,

$$Q = -\frac{\pi f(t_2 - t_1)}{\ln\left(\frac{A_1 t_1}{A_2 t_2}\right)},$$

where A_1 and A_2 are the vector potentials at time t_1 and t_2. Analytic calculation[18] of the quality factor treating an open cavity like a transmission line with a load simulating free space (377 Ω) gives a value of Q = 24. The Q values obtained from our scheme (Q = 26.5768 for $\theta = 0°$ and Q = 25.7875 for $\theta = 31.42°$) are very close to the analytically obtained value. Figure 11 shows the time evaluation of vector potential $\mathbf{A}$ at the point (3.36 cm, 3.36 cm) for both cases.

C. A6 magnetron

We chose a 2D A6 magnetron to evaluate the applicability of our scheme to complicated geometry. For this study, a 2D domain of $\Omega = [0\text{ cm}, 5\text{ cm}]^2$ is chosen with the radius of the vane resonators $r_v = 4.11$ cm, anode radius $r_a = 2.11$ cm, cathode radius $r_c = 1.58$ cm, angular width of vane, 20°, and cavity angle, 40°. We set the grid size to be 128×128, time step size $\Delta t = 39$ ps, averaging parameter $\beta = 1.4$, and dissipation coefficient $\varepsilon = 0.1$. The embedded PEC boundary condition is imposed over the boundary stencil during the $x-$ and $y-$sweeps. Figure 12 shows the geometrical setup of the A6 magnetron, indicating key on/off mesh grid points used by the scheme for the $x-$sweeps.

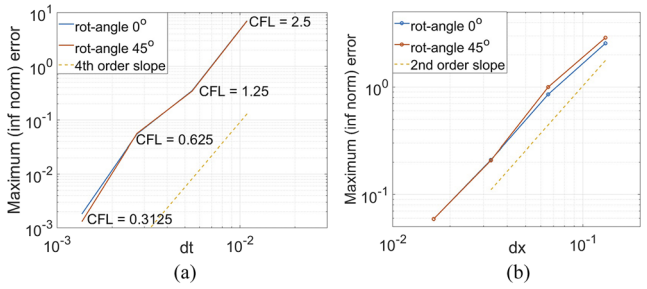

FIG. 10. Convergence plots for (a) time and (b) space on a PEC a square domain $[0\text{ cm}, 21\text{ cm}]^2$. This study measures the L_∞ norm of the error at time $T = 1.0$ ns compared with the analytical solution for the cases of mesh aligned and nonaligned (rotated by 45°).

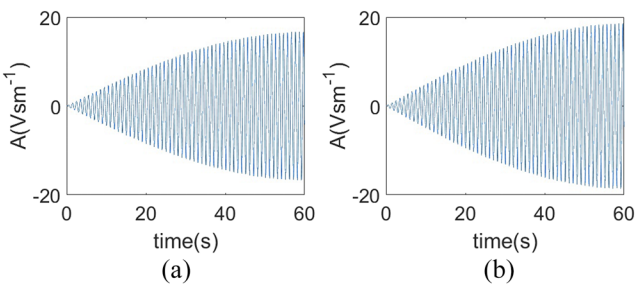

FIG. 11. Time evaluation of **A** in (a) mesh aligned and (b) $\theta = 31.42°$ rotated square cavities with a leak imposed by the outflow boundary condition and diffraction Q, which is obtained for an oscillating point source $[\sin(2\pi ft)$, $f = 1$GHz$]$.

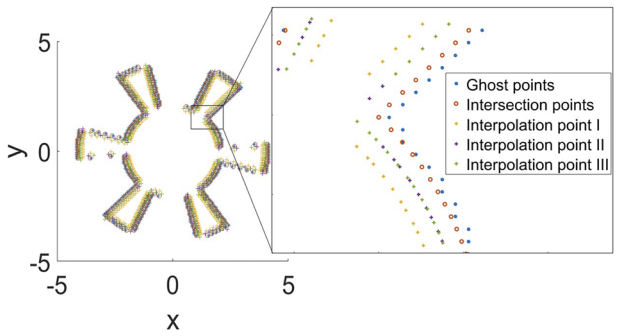

FIG. 12. Key points: intersection, boundary, ghost, and interpolation points required to obtain $x-$sweeps of the simulation of A6M.

This test was conducted for frequency mode analysis. A point source was placed in the center of the A6M and the frequency distribution examined using a ping test at the point $(1.4063\,\text{cm}, 0.8203\,\text{cm})$. We obtained six strong frequency modes as shown in Fig. 13, associated with the first two passbands, clearly showing the effects of all six symmetric resonances.

We further simulated the A6 magnetron with a transparent cathode that is mimicked by using six emitters/point sources and calculated the time evolution (Fig. 14) for the same configuration as above. The formulation maintains symmetry and high accuracy for relative coarse-grained solutions. For example, in the A6 magnetron results, there are only 12 points across the neck of the magnetron veins.

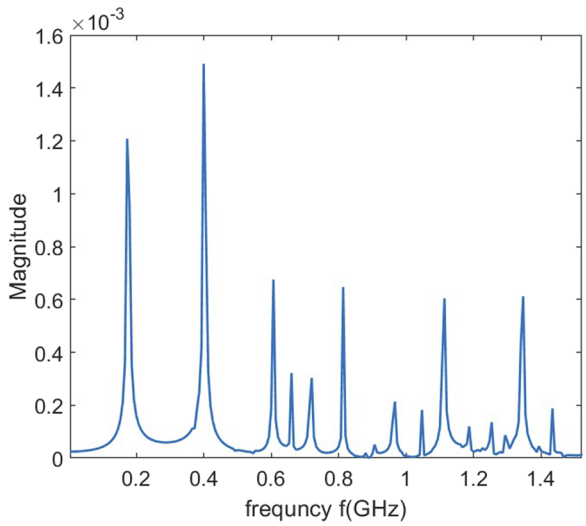

FIG. 13. Frequency spectrum of 2D A6M with vane resonators $r_v = 4.11$ cm, anode radius $r_a = 2.11$ cm, cathode radius $r_c = 1.58$ cm, angular width of vane, 20°, and cavity angle, 40°; the grid size 128 × 128, time step size $\Delta t = 39$ ps, averaging parameter $\beta = 1.4$, and dissipation coefficient $\varepsilon = 0.1$.

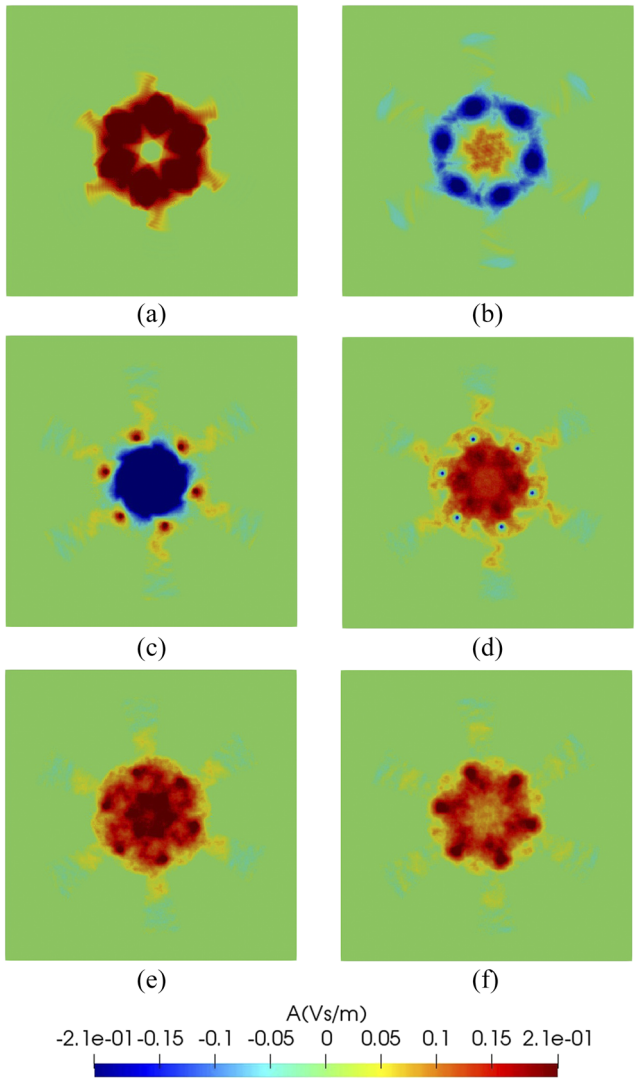

FIG. 14. Evolution of the transparent cathode A6M at different times, (a)–(f), with vane resonators $r_v = 4.11$ cm, anode radius $r_a = 2.11$ cm, cathode radius $r_c = 1.58$ cm, angular width of vane, 20°, and cavity angle, 40°; the grid size 128 × 128, time step size $\Delta t = 39$ ps, averaging parameter $\beta = 1.4$, and dissipation coefficient $\varepsilon = 0.1$.

VI. CONCLUSION

We developed a fast A-stable implicit scheme for the computation of electromagnetic potentials with an embedded PEC boundary condition. We successfully evaluated it using EM wave propagation in different shaped objects in 2D, especially an A6 magnetron. We obtained fourth order accuracy in time. Generalizing the stabilization of our scheme for PEC boundary conditions and developing a solver for 3D EM problems with complex geometries will be considered in the future.

ACKNOWLEDGMENTS

We would like to thank both Dr. W. Nicholas G. Hitchon and Dr. Matthew Causley for their helpful conversations related to this work. We would also like to thank AFOSR and NSF for their support of this work under Grant Nos. FA9550-19-1-0281 and FA9550-17-1-0394 and NSF Grant No. DMS 191218.

DATA AVAILABILITY

The data that support the findings of this study are available from the corresponding author upon reasonable request.

REFERENCES

[1]J. Zhang, D. Zhang, Y. Fan, J. He, X. Ge, X. Zhang, J. Ju, and T. Xun, "Progress in narrowband high-power microwave sources," Phys. Plasmas **27**, 010501 (2020).

[2]R. E. Peterkin and J. W. Luginsland, "A virtual prototyping environment for directed-energy concepts," Comput. Sci. Eng. **4**, 42–49 (2002).

[3]C. K. Birdsall and A. B. Langdon, *Plasma Physics Via Computer Simulation* (CRC Press, 2004).

[4]M.-C. Lin, C. Nieter, P. Stoltz, and D. Smithe, "Accurately and efficiently studying the RF structures using a conformal finite-difference time-domain particle-in-cell method," Open Plasma Phys. J. **3**, 48 (2010).

[5]T. M. Austin, J. R. Cary, D. N. Smithe, and C. Nieter, "Alternating direction implicit methods for FDTD using the Dey-Mittra embedded boundary method," Open Plasma Phys. J. **3**, 29–35 (2010).

[6]J. D. Jackson, *Classical Electrodynamics*, 3rd ed. (Wiley, New York, NY, 1999).

[7]M. F. Causley, A. Christlieb, Y. Güçlü, and E. Wolf, "Method of lines transpose: A fast implicit wave propagator," Math. Comput. (submitted) (2013); arXiv:1306.6902.

[8]M. F. Causley and A. J. Christlieb, "Higher order A-stable schemes for the wave equation using a successive convolution approach," SIAM J. Numer. Anal. **52**, 220–235 (2014).

[9]E. M. Wolf, M. Causley, A. Christlieb, and M. Bettencourt, "A particle-in-cell method for the simulation of plasmas based on an unconditionally stable field solver," J. Comput. Phys. **326**, 342–372 (2016).

[10]M. Causley, A. Christlieb, and E. Wolf, "Method of lines transpose: An efficient unconditionally stable solver for wave propagation," J. Sci. Comput. **70**, 896–921 (2017).

[11]M. Thavappiragasam, A. Viswanathan, and A. Christlieb, "MOL^T based fast high-order three dimensional A-stable scheme for wave propagation," J. Coupled Syst. Multiscale Dyn. **5**, 151–163 (2017).

[12]U. Ascher, R. Mattheij, and R. Russell, *Numerical Solution of Boundary Value Problems for Ordinary Differential Equations* (Society for Industrial and Applied Mathematics, 1995).

[13]R. Coifman, V. Rokhlin, and S. Wandzura, "The fast multipole method for the wave equation: A pedestrian prescription," IEEE Antennas Propag. Mag. **35**, 7–12 (1993).

[14]A. Mazzia and F. Mazzia, "High-order transverse schemes for the numerical solution of PDEs," J. Comput. Appl. Math. **82**, 299–311 (1997).

[15]Z. Gimbutas and V. Rokhlin, "A generalized fast multipole method for nonoscillatory kernels," SIAM J. Sci. Comput. **24**, 796–817 (2003).

[16]P. Li, H. Johnston, and R. Krasny, "A Cartesian treecode for screened Coulomb interactions," J. Comput. Phys. **228**, 3858–3868 (2009).

[17]W. D. Henshaw, "A high-order accurate parallel solver for Maxwell's equations on overlapping grids," SIAM J. Sci. Comput. **28**, 1730–1765 (2006).

[18]Y. J. Huang, L. H. Yeh, and K. R. Chu, "An analytical study on the diffraction quality factor of open cavities," Phys. Plasmas **21**, 103112 (2014).