ACS Photonics

pubs.acs.org/journal/apchd5

Article

Fast Inverse Design of 3D Nanophotonic Devices Using Boundary Integral Methods

Emmanuel Garza and Constantine Sideris*

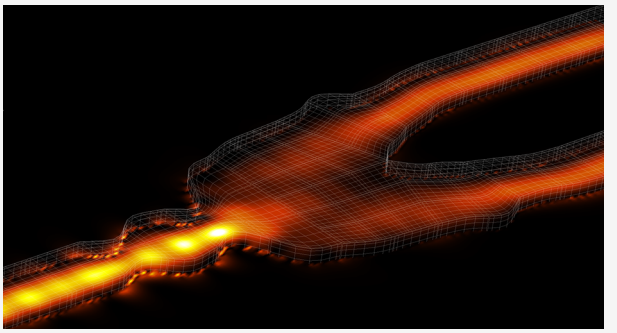
Cite This: https://doi.org/10.1021/acsphotonics.2c01072

Read Online

ACCESS | Metrics & More | Article Recommendations | Supporting Information

Downloaded via UNIV OF SOUTHERN CALIFORNIA on November 28, 2022 at 12:50:57 (UTC).
See https://pubs.acs.org/sharingguidelines for options on how to legitimately share published articles.

ABSTRACT: Recent developments in the computational automated design of electromagnetic devices, otherwise known as inverse design, have significantly enhanced the design process for nanophotonic systems. Inverse design can both reduce design time considerably and lead to high-performance, nonintuitive structures that would otherwise have been impossible to develop manually. Despite the successes enjoyed by structure optimization techniques, most approaches leverage electromagnetic solvers that require significant computational resources and suffer from slow convergence and numerical dispersion. Recently, a fast simulation and boundary-based inverse design approach based on boundary integral equations was demonstrated for two-dimensional nanophotonic problems. In this work, we introduce a new full-wave three-dimensional simulation and boundary-based optimization framework for nanophotonic devices also based on boundary integral methods, which achieves high accuracy even at coarse mesh discretizations while only requiring modest computational resources. The approach has been further accelerated by leveraging GPU computing, a sparse block-diagonal preconditioning strategy, and a matrix-free implementation of the discrete adjoint method. As a demonstration, we optimize three different devices: a 1:2 1550 nm power splitter and two nonadiabatic mode-preserving waveguide tapers. To the best of our knowledge, the tapers, which span 40 wavelengths in the silicon material, are the largest silicon photonic waveguiding devices to have been optimized using full-wave 3D solution of Maxwell's equations.



KEYWORDS: *inverse design, nanophotonic devices, fast maxwell simulation/optimization, integral equations, computational electromagnetics, GPU acceleration*

1. INTRODUCTION

Silicon photonics involves the precise harnessing and manipulation of light at the nanoscale and enables rich interaction with electronics via photonic integrated circuits. It is rapidly becoming a billion-dollar industry due to the large and diverse number of important applications it can enable, including high-speed optical interconnects,[1] biomedical and environmental sensing devices,[2,3] quantum computing,[4] and machine learning,[5] among many more. Despite offering tremendous potential, compact and high-performance silicon photonic devices unfortunately remain challenging to design and model today due to lack of analytical solutions and difficulties with numerical simulation. Recently, inverse design has emerged as a powerful paradigm for alleviating the design challenges by offering an automated computational design framework for nanophotonic devices which optimizes structures based on desired performance specifications and design constraints as inputs.[6−8] Inverse design can both appreciably reduce the design time required for developing new nanophotonic structures as well as produce nonintuitive, high-performance designs, which would otherwise have been impossible for a human to arrive at manually. Inverse design has been successfully applied toward the design of a plethora of nanophotonic devices, including power splitters,[8] mode and wavelength (de)multiplexers,[9] grating couplers,[10] and metamaterials,[11] among many more.

Although electromagnetic inverse design has enjoyed significant success,[6,12−14] most present-day implementations remain limited by the challenges associated with accurate simulation of the forward electromagnetic problem. Although some fast approximate or highly specialized simulation techniques have been leveraged in particular situations, such as 2D effective-index approximation,[15] rigorous coupled wave analysis (RCWA),[16] and beam-propagation method (BPM),[17] the majority of nanophotonics devices require full-wave 3D solution of Maxwell's solutions to be modeled accurately given that device geometrical features are of the order of the

Special Issue: Optimized Photonics and Inverse Design

Received: July 11, 2022

© XXXX American Chemical Society

operating wavelength. Finite-difference (FD) methods are the most commonly used 3D simulation algorithms for inverse design of nanophotonics, especially the finite-difference frequency-domain (FDFD) method. While FD methods are versatile and straightforward to implement, they suffer from a number of drawbacks which ultimately limit the size and complexity of devices that can be optimized via inverse design and often require significant computational power. In particular, due to their centered-difference approximation of the difference operators in Maxwell's equations, FD methods achieve only second-order convergence in the solution accuracy with respect to the meshing resolution in the best case, have difficulty representing structures with complex geometrical features and curvature accurately due to using orthogonal Cartesian grids, and suffer significant errors due to numerical dispersion due to accumulated local truncation errors. Furthermore, despite resulting in a highly sparse linear system, the FDFD method becomes very poorly conditioned as the problem size increases and mesh cell size decreases, requiring tens of thousands of iterations to achieve convergence using iterative linear solvers.[18] These limitations have hindered inverse design approaches that use the FDFD method from being used to optimize structures larger than a few wavelengths per side.

Recently, boundary integral equation (BIE) methods have been evaluated as an alternative solution approach for modeling nanophotonic devices.[8] In contrast to FD and FEM methods, BIE methods only require meshing and solving for unknowns on the surfaces between dielectric interfaces, rather than volumes. This often leads to a dramatic reduction in problem size. Furthermore, BIE formulations are known to result in dense linear systems which enjoy very favorable spectral and conditioning properties, often only requiring tens to hundreds of iterations to achieve convergence using Krylov subspace solvers. Due to their analytical propagation of sources to targets using Green's functions, integral methods are also virtually free of the numerical dispersion issues which plague FD methods. Although traditionally known as being open-domain methods and predominantly used for radar scattering and antenna simulation problems, the recent introduction of the Windowed Green's Function (WGF) method[19−23] has enabled the finite of truncation waveguides incoming from and outgoing to infinity, acting as an absorbing boundary condition. In ref 8, the application of BIE methods toward the simulation and inverse design of nanophotonic devices in two dimensions was demonstrated, and furthermore compared solution accuracy, computational time, and memory resources among various state-of-the-art solvers using different algorithms. The results achieved in ref 8 showed the potential of BIE methods for simulating and optimizing nanophotonic devices; however, the solver and examples presented were limited to two-dimensional problems.

In this work, we report on a new boundary integral and inverse design framework capable of efficient modeling of nanophotonic devices using full-wave 3D solution of Maxwell's equations. Due to the additional challenges involved in scaling from two to three dimensions, we have developed and utilized a number of techniques to accelerate our methods, including leveraging GPU computation, constructing and using sparse block-diagonal preconditioners, and deriving a matrix-free approach for applying the transpose of the electromagnetic system operator, which is necessary for device optimization using the discrete adjoint method. We first introduce the Chebyshev-based Boundary Integral Equation (CBIE) formulation, followed by detailing the aforementioned acceleration strategies. Finally, we demonstrate the potential of the new framework by designing three different silicon photonic devices: a 1:2 1550 nm power splitter and two nonadiabatic mode-preserving waveguide tapers. To the best of our knowledge, the tapers, which span 40 wavelengths in the silicon material, are among the largest silicon photonic waveguiding devices to have been optimized using full-wave 3D solution of Maxwell's equations, showcasing the capability of the approach for fast simulation and optimization of large, many wavelength devices.

2. FULL-WAVE ANALYSIS OF NANOPHOTONIC STRUCTURES USING BOUNDARY INTEGRAL METHODS

We consider the problem of simulating and optimizing nanophotonic structures composed of dielectric materials. In particular, the devices in question can support multiple open waveguides (incoming from or outgoing to infinity) and nonuniform geometrical features inside a bounded region. For simplicity, our description of the physical system involves only two dielectric domains, but the framework can be directly applied to any number of materials in a straightforward manner.

Let the device of interest consist of a single dielectric interior region Ω_i with permittivity ε_i, magnetic permeability μ_i, and an exterior region Ω_e with permittivity ε_e, magnetic permeability μ_e. Denote by Γ the interface between Ω_i and Ω_e with normal vector $\mathbf{n}$ pointing toward the exterior domain. In Figure 1, we illustrate a typical photonic device and the notation used. In general, Γ is an unbounded surface due to the presence of infinite input and output waveguides. In the frequency domain, the electric and magnetic fields, denoted by $\mathbf{E}$ and $\mathbf{H}$ respectively, satisfy Maxwell's equations:

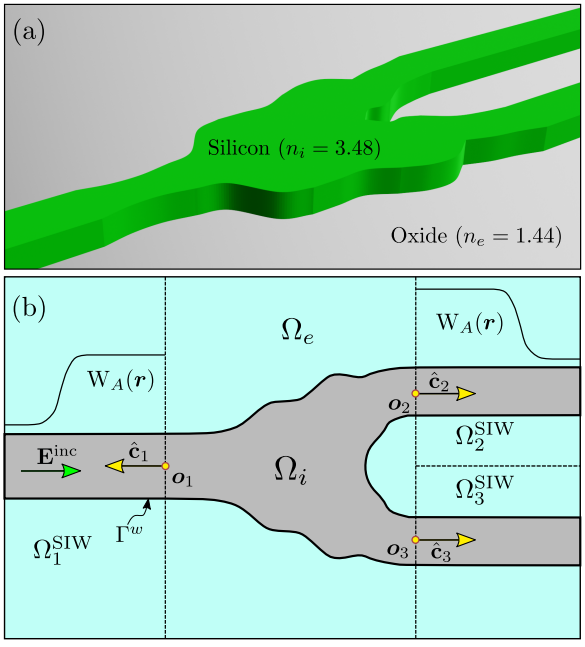


Figure 1. (a) Perspective view of a power splitter photonic device. (b) Illustration of the cross section view and notation used for the WGF method.

https://doi.org/10.1021/acsphotonics.2c01072
ACS Photonics XXXX, XXX, XXX−XXX

$$\begin{aligned} \nabla \times \mathbf{H}_l + i\omega\varepsilon_l \mathbf{E}_l &= 0 \\ \nabla \times \mathbf{E}_l - i\omega\mu_l \mathbf{H}_l &= 0 \end{aligned} \quad \text{in } \Omega_l \tag{1}$$

where ω is the angular frequency and the subscript l is used for either $l = i$ or $l = e$ to denote the interior and exterior domains, respectively. Additionally, the fields must satisfy the dielectric boundary conditions, which state that the tangential components of both the $\mathbf{E}$ and $\mathbf{H}$ must be continuous at Γ.

To solve the system of Maxwell's equations in eq 1, we use a three-dimensional boundary integral representation[24−26] for the electromagnetic fields. This approach has several advantages over volumetric methods (e.g., finite-differences), including the fact that only material interfaces need to be discretized (as opposed to the entire simulation volumes) and that the geometrical features of the devices are parametrically represented, avoiding material staircasing errors. The aforementioned integral representation is based on the free-space Green's function for the Helmholtz equation $G_l(\mathbf{r}, \mathbf{r}') = \exp(ik_l|\mathbf{r} - \mathbf{r}'|)/(4\pi|\mathbf{r} - \mathbf{r}'|)$ where k_l is the wavenumber in Ω_l, which is used to define the vector potentials acting on a surface tangential density $\mathbf{a}$:

$$\mathcal{A}_l[\mathbf{a}](\mathbf{r}) \equiv \nabla \times \int_\Gamma G_l(\mathbf{r}, \mathbf{r}')\mathbf{a}(\mathbf{r}')\, d\sigma(\mathbf{r}') \tag{2a}$$

$$\begin{aligned} \mathcal{B}_l[\mathbf{a}](\mathbf{r}) &\equiv \nabla \times \nabla \times \int_\Gamma G_l(\mathbf{r}, \mathbf{r}')\mathbf{a}(\mathbf{r}')\, d\sigma(\mathbf{r}'), \\ &= k_l^2 \int_\Gamma G_l(\mathbf{r}, \mathbf{r}')\mathbf{a}(\mathbf{r}')\, d\sigma(\mathbf{r}') \\ &\quad + \nabla \int_\Gamma G_l(\mathbf{r}, \mathbf{r}')\, \mathrm{div}_\Gamma\, \mathbf{a}(\mathbf{r}')\, d\sigma(\mathbf{r}') \end{aligned} \tag{2b}$$

Using the potentials in eq 2, the scattered components of the electromagnetic fields can be represented with either a direct[24−26] or indirect formulation.[27] In this work, we have used the indirect formulation for the electromagnetic fields given by

$$\begin{aligned} \mathbf{E}_l(\mathbf{r}) &= -\omega\mu_l \mathcal{A}_l[\mathbf{m}](\mathbf{r}) - i\mathcal{B}_l[\mathbf{j}](\mathbf{r}) \\ \mathbf{H}_l(\mathbf{r}) &= -\omega\varepsilon_l \mathcal{A}_l[\mathbf{j}](\mathbf{r}) + i\mathcal{B}_l[\mathbf{m}](\mathbf{r}) \end{aligned} \quad \text{for } \mathbf{r} \in \Omega_l \tag{3}$$

with tangential density currents $\mathbf{m}$ and $\mathbf{j}$, and where the coefficients in eq 3 multiplying the vector potentials are such that enforcing the continuity of the tangential components of the fields across Γ results in a Müller system of integral equations with only weakly singular integral operators.[28−30] The tangential components of eq 3 evaluated at the boundary Γ can be expressed in terms of the boundary integral operators that result from taking the cross product with the normal vector $\mathbf{n}(\mathbf{r})$ and the limit as $\mathbf{r} \to \mathbf{r}'$, namely

$$S_l[\mathbf{a}](\mathbf{r}) \equiv \mathbf{n}(\mathbf{r}) \times \int_\Gamma G_l(\mathbf{r}, \mathbf{r}')\mathbf{a}(\mathbf{r}')\, d\sigma(\mathbf{r}') \tag{4a}$$

$$R_l[\mathbf{a}](\mathbf{r}) \equiv \boldsymbol{n}(\mathbf{r}) \times \nabla \times \int_\Gamma G_l(\mathbf{r}, \mathbf{r}')\mathbf{a}(\mathbf{r}')\, d\sigma(\mathbf{r}') \tag{4b}$$

$$T_l[\mathbf{a}](\mathbf{r}) \equiv \mathbf{n}(\mathbf{r}) \times \nabla \int_\Gamma G_l(\mathbf{r}, \mathbf{r}')\, \mathrm{div}_\Gamma\, \mathbf{a}(\mathbf{r}')\, d\sigma(\mathbf{r}') \tag{4c}$$

with $\mathbf{r} \in \Gamma$. By defining the following weakly singular integral operators in terms of eq 4

$$R_\alpha^\Delta \equiv \frac{2}{\alpha_e + \alpha_i}(\alpha_e R_e - \alpha_i R_i) \tag{5a}$$

$$K_\alpha^\Delta \equiv \frac{2i}{\omega(\alpha_e + \alpha_i)}[(T_e - T_i) + (k_e^2 S_e - k_i^2 S_i)] \tag{5b}$$

where the subindex $\alpha = \varepsilon$ or $\alpha = \mu$ is used to denote the electric permittivity or magnetic permeability, respectively, then, the indirect Müller system of integral equations is

$$\begin{bmatrix} I + R_\mu^\Delta & K_\mu^\Delta \\ -K_\varepsilon^\Delta & I + R_\varepsilon^\Delta \end{bmatrix} \begin{bmatrix} \mathbf{m} \\ \mathbf{j} \end{bmatrix} = \begin{bmatrix} 2\omega^{-1}(\mu_e + \mu_i)^{-1} & \mathbf{n} \times (\mathbf{E}_e^{\mathrm{inc}} - \mathbf{E}_i^{\mathrm{inc}}) \\ 2\omega^{-1}(\varepsilon_e + \varepsilon_i)^{-1} & \mathbf{n} \times (\mathbf{H}_e^{\mathrm{inc}} - \mathbf{H}_i^{\mathrm{inc}}) \end{bmatrix} \tag{6}$$

To solve the scattering problem for given incident electromagnetic fields ($\mathbf{E}^{\mathrm{inc}}$, $\mathbf{H}^{\mathrm{inc}}$), we solve for the density currents $\mathbf{m}$ and $\mathbf{j}$ defined only along the boundary surface Γ. Subsequently, the scattered fields can be obtained anywhere in space using the potential representation eq 3. The numerical solution of eq 6 for waveguide problems requires the accurate truncation of integral operators over infinite boundaries, as well as discretizing the continuous operators to obtain a linear system of equations. These two components are explained in Sections 2.1 and 2.2, respectively.

2.1. Smooth Truncation of Integral Operators. Integral equation methods have traditionally been limited to bounded material interfaces Γ inside open domains, due to the low-order approximation and large error that results from truncating the slowly decaying integrals in eq 4 when an infinite boundary is present. This scenario happens to be the case for many photonic devices which consist of input/output waveguides that extend to infinity. However, there have been recent efforts in addressing this challenge, such as using a surface conductive absorber to avoid unwanted reflections from truncating waveguide boundaries,[31] or utilizing the Windowed Green Function (WGF) method to accurately evaluate integral operators along waveguide boundaries.[22,23,32] In this work, we use the WGF approach to handle waveguide simulations due to its ease of implementation as well as requiring relatively small window sizes (i.e., the length of the truncated waveguide) to achieve high accuracy.

The WGF method operates on the basis of smoothly windowing the integrals onto a truncated surface—the approach is highly effective, with superalgebraic convergence as the size of the window is increased.[22,23] We denote by Ω_q^{SIW} the domain that contains the q-th semi-infinite waveguide (SIW), $\mathbf{o}_q$ the corresponding origin of the SIW, and $\hat{\mathbf{c}}_q$ the respective optical axis pointing toward the infinite section of the SIW, see Figure 1 (b). Subsequently, we can define a suitable window function for the waveguide structure in the following way:

$$W_A(\boldsymbol{r}) \equiv \begin{cases} w_A(\hat{\mathbf{c}}_q \cdot (\mathbf{r} - \mathbf{o}_q)), & \mathbf{r} \in \Gamma \cap \Omega_q^{\mathrm{SIW}} \\ 1, & \text{otherwise} \end{cases} \tag{7}$$

where A is the window size, $w_A(x)$ is an infinitely smooth function with compact support from $x \in (-A, A)$ and which equals 1 on the interval $x \in (-\alpha A, \alpha A)$ for some $\alpha \in (0, 1)$. The function w_A can be, for example, the function defined in eq

https://doi.org/10.1021/acsphotonics.2c01072
ACS Photonics XXXX, XXX, XXX−XXX

14 from ref 23, or alternatively other smooth functions that may not have strict compact support can be used, such as window functions constructed on the basis of the hyperbolic tangent or the error function. In Figure 1 (b), the windowing strategy is illustrated for a nanophotonic device and the relevant domains are shown for a top view of the device.

For problems with illuminating fields such as Gaussian beams or dipole sources, applying the WGF involves "windowing" the integral operators in eq 6 by taking the windowed versions of eq 4:

$$S_l^W[\mathbf{a}](\mathbf{r}) \equiv \mathbf{n}(\mathbf{r}) \times \int_\Gamma G_l(\mathbf{r}, \mathbf{r}') W_A(\mathbf{r}') \mathbf{a}(\mathbf{r}')\, \mathrm{d}\sigma(\mathbf{r}') \tag{8a}$$

$$R_l^W[\mathbf{a}](\mathbf{r}) \equiv \mathbf{n}(\mathbf{r}) \times \nabla \times \int_\Gamma G_l(\mathbf{r}, \mathbf{r}') W_A(\mathbf{r}') \mathbf{a}(\mathbf{r}')\, \mathrm{d}\sigma(\mathbf{r}') \tag{8b}$$

$$T_l^W[\mathbf{a}](\mathbf{r}) \equiv \mathbf{n}(\mathbf{r}) \times \nabla \int_\Gamma G_l(\mathbf{r}, \mathbf{r}') W_A(\mathbf{r}')\, \mathrm{div}_\Gamma\, \mathbf{a}(\mathbf{r}')\, \mathrm{d}\sigma(\mathbf{r}') \tag{8c}$$

and the corresponding weakly singular operators $R_\alpha^{W,\Delta}$ and $T_\alpha^{W,\Delta}$ are obtained by using eq 8 for evaluating eq 5. Then, the windowed Müller system of integral equations is

$$\begin{bmatrix} I + R_\mu^{W,\Delta} & K_\mu^{W,\Delta} \\ -K_\varepsilon^{W,\Delta} & I + R_\varepsilon^{W,\Delta} \end{bmatrix} \begin{bmatrix} \mathbf{m} \\ \mathbf{j} \end{bmatrix} = \begin{bmatrix} 2\omega^{-1}(\mu_e + \mu_i)^{-1} & \mathbf{n} \times (\mathbf{E}_e^{\mathrm{inc}} - \mathbf{E}_i^{\mathrm{inc}}) \\ 2\omega^{-1}(\varepsilon_e + \varepsilon_i)^{-1} & \mathbf{n} \times (\mathbf{H}_e^{\mathrm{inc}} - \mathbf{H}_i^{\mathrm{inc}}) \end{bmatrix} \tag{9}$$

which is only enforced in the *finite* windowed boundary $\Gamma^W = \Gamma \cap \{W_A > 0\}$. The near-fields can be obtained using a similar windowed version of the vector potential representation in eq 2.

2.2. Numerical Discretization. The WGF formulation is agnostic to the numerical discretization utilized, and one can use any Galerkin, collocation, or Nyström method to solve for the unknown density currents in eq 9. In this work, we utilized the high-order Chebyshev-based Nyström method introduced in ref 33 for acoustic integral operators, and which has recently also been successfully implemented for the electromagnetic case,[30,32,34] including in the WGF waveguide context.[23] This Chebyshev-based boundary integral equation (CBIE) method relies on representing the surface boundary Γ by a set of nonoverlapping, curvilinear quadrilateral patches with parametrization $\mathbf{r}^p(u, v)$ for the p-th patch and with $(u, v) \in [-1,1]^2$. The tangential current densities $\mathbf{m}$ and $\mathbf{j}$ are expanded and approximated in terms of Chebyshev polynomials. To evaluate the integral operators when the target point $\mathbf{r}$ is far from the source point $\mathbf{r}'$, Fejér's first quadrature rule is used. On the other hand, for target and source points that are close to each other, which cause numerical issues due to the singularity of the Green's function, the CBIE method uses precomputed weights to accurately evaluate the weakly singular integrals. For a detailed description of the CBIE method, we refer the reader to refs 30, 33.

The discretized version of the integral operators in the left-hand side of eq 9 form matrices that are in general full and of size $(4N_p) \times (4N_p)$, where N_p is the total number of discretization points—the factor of 4 coming from having two vector densities, each with two components in the tangential basis. The number of points can range from tens of thousands to multiple millions depending on the size of the problem being considered, which makes explicitly constructing these matrices and storing them in memory intractable except for very small problems. In view of this, most BIE methods leverage iterative linear solvers such as the Generalized Minimum Residual (GMRES) algorithm,[35] for which only the action of the system matrix applied to vectors is required. Thus, there is typically no need to explicitly assemble the matrix system. However, in the optimization context, access to the action of the adjoint matrices is also needed to compute sensitivities in an efficient manner. Hence, in the Supporting Information, we include the nontrivial derivation of the CBIE matrices of the discretized integral operators, as well as the adjoint matrices. Both the direct and adjoint linear systems are solved using GMRES, therefore not requiring the formation of any matrix explicitly and significantly reducing the memory resources required. For shorthand notation, for the rest of this work, we denote by

$$A\mathbf{\Phi} = b \tag{10}$$

the CBIE discretized version of eq 9.

2.3. GPU Acceleration. The direct implementation of the action of the matrix in eq 10 involves a computational cost $O(N_p^2)$. In order to solve this system with an iterative method (e.g., GMRES), the total solution cost for N_i iterations is $O(N_i N_p^2)$. This computational complexity can severely cripple the application of BIE methods to large sized devices. One alternative is to incorporate algorithmic acceleration such as the fast multiple method (FMM),[36−39] equivalent source technique,[40] or the interpolated factored Green function method.[41] Although these acceleration techniques can greatly speed-up the simulation of large problems, they also introduce further difficulties that are particularly evident in the optimization context: (1) they are approximations to the full system and thus can introduce nonlinear errors that affect the computation of the gradient, and (2) small perturbations in the mesh can lead to changing the structure of the interactions between particles, thus requiring recomputation of the relevant multipole expansions or equivalent sources.

Alternatively, applying the action of eq 10 to a vector can be significantly sped up by instead using hardware acceleration with graphics processing units (GPUs). One of the great advantages of this approach is that it does not involve any approximation to the original matrix system and requires only programming implementation changes rather than any algorithmic changes. Although the computational complexity of the solution remains $O(N_i N_p^2)$, the proportionality constant decreases significantly with the incorporation of a GPU, making it an appealing approach for solving moderate to large sized problems up to several millions of unknowns.

In the GPU-accelerated implementation of the action of the matrix in eq 10, all the operations related to the singular and near-singular interactions are performed on the CPU due to the fact that the precomputed weights may require significant storage, potentially more than the available GPU memory. On the other hand, the far interactions are all computed on the GPU, which involve recomputing the pointwise distance between points and the sines and cosines associated with the Green's function in the discretized integral operators for each matrix-vector multiplication. This separation between operations in the CPU and GPU has the advantage that they can be

https://doi.org/10.1021/acsphotonics.2c01072
ACS Photonics XXXX, XXX, XXX−XXX

both done concurrently, hence the GPU speedup is not limited by the CPU computation of the singular and near-singular contributions. Furthermore, there are only $O(N_p)$ near-singular and singular interactions, whereas there are $O(N_p^2)$ far interactions, making the far interactions always the rate-limiting step.

In Table 1, we present the computing times (in seconds) for evaluating a single matrix-vector multiplication of the system in eq 10 using two different computers. One of the computers is a server with two Intel Xeon Gold 6154 (36 cores) and an NVIDIA Titan V GPU, and the other is a personal workstation with an Intel i9-9900KF (8 cores) and a consumer grade GPU (RTX 2080 SUPER). In both cases, the CPU only implementation used all of the available CPU cores, while the "CPU + GPU" case does the singular and near-singular interactions using the CPU and the rest of the contributions are performed on the GPU. We see that even on the system with 36 cores, the speedup from using the GPU close to 50 times faster for problems with more than one million unknowns. Furthermore, even on a modest workstation with a consumer-grade GPU we achieved 6 times faster computing times (still using double precision arithmetic). For the problems considered in this paper, namely optimizing devices, many simulations—direct and adjoint—as well as other matrix-vector multiplications from the system in eq 10 are needed; hence, the significant acceleration obtained with the help of the GPU is greatly magnified.

Table 1. Computing Times (in Seconds) for a Single Matrix-Vector Multiply of the System in Eq 10 for a Spherical Geometry[a]

	Dual Intel Xeon Gold 6154 (36 cores @3.0 GHz) + Titan V			Intel i9-9900KF (8 cores @4.7 GHz) + RTX 2080 SUPER		
N_{unk}	CPU only	CPU + GPU	speedup	CPU only	CPU + GPU	speedup
153 600	9.1	0.7	13.0×	14.9	2.4	6.2×
777 600	190	4.8	39.6×	359.7	60.6	5.9×
1 572 864	762	16.0	47.7×	1500	244.0	6.2×
4 153 344	5328	109.8	48.5×	–	–	–

[a]The CPU only implementation uses all cores of the system.

2.4. Block Diagonal Preconditioning. For silicon photonic applications, the waveguide structures typically have a high refractive index contrast, which, in the context of iterative BIE solvers, translates into potentially requiring a large number of iterations for the linear algebra solver (e.g., GMRES) to converge. This is a well-known issue of surface integral equations, and typically preconditioners are used to alleviate the increased number of iterations needed.[42−45] To mitigate this problem we developed and used an algebraic sparse block diagonal preconditioner, for which the nonzero

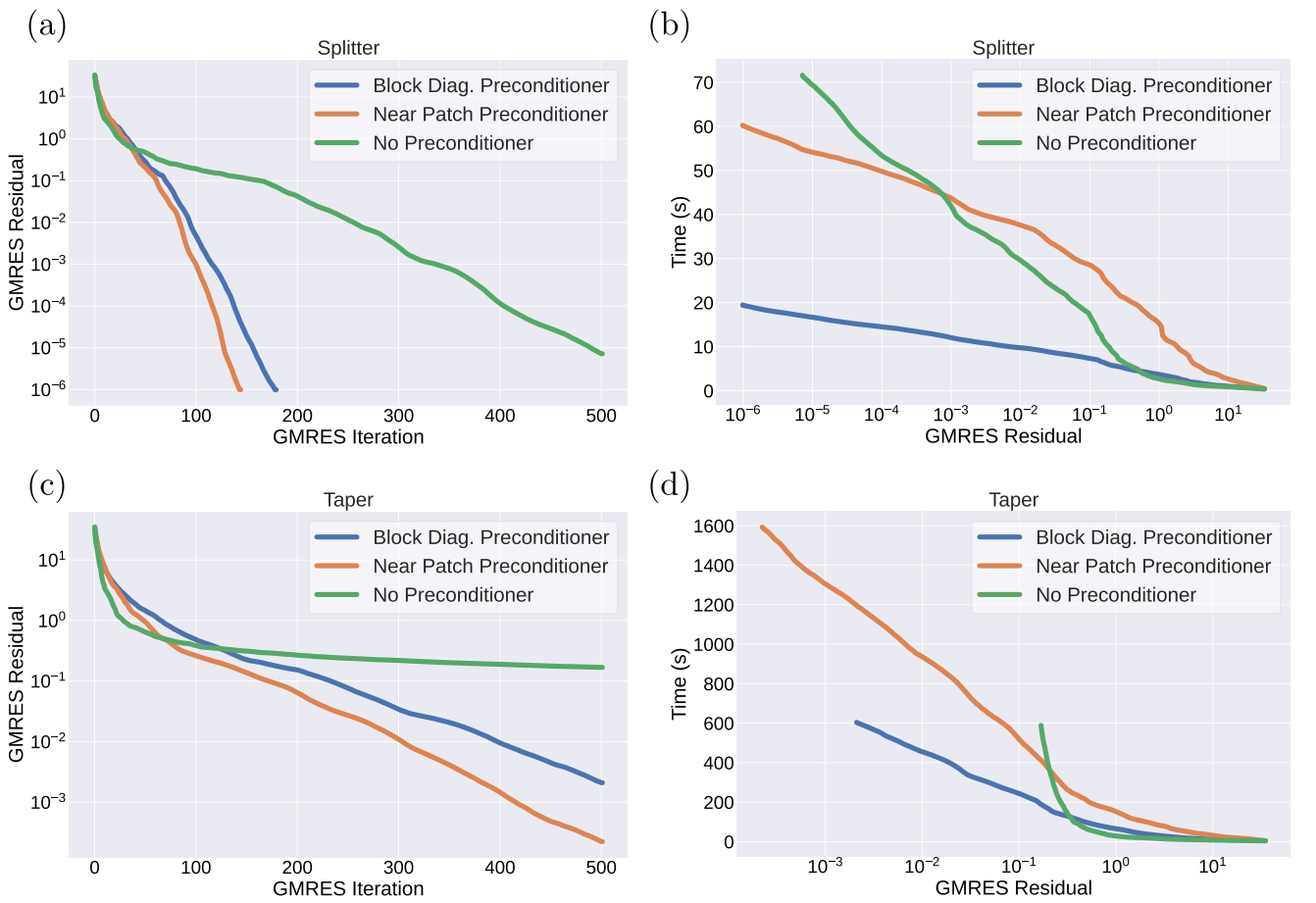


Figure 2. Effect of using a preconditioner for GMRES to solve the system in eq 10. Panels (a) and (b) correspond to the power splitter device from Section 4.1, while (c) and (d) correspond to the 9 μm waveguide taper from Section 4.2. In the left column, (a) and (c) present the residual as a function of the number of iterations, while (b) and (d) plot the computing time needed to achieve a given GMRES residual.

https://doi.org/10.1021/acsphotonics.2c01072
ACS Photonics XXXX, XXX, XXX−XXX

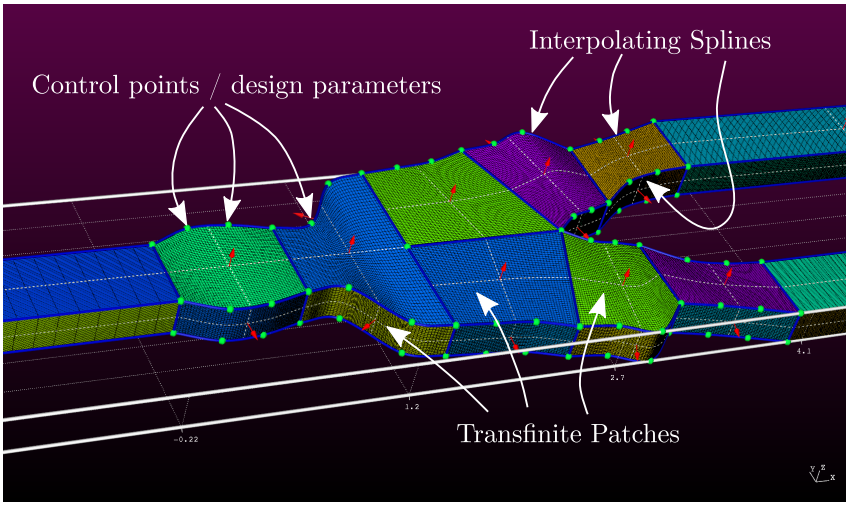


Figure 3. CAD design of a splitter geometry using Gmsh.[47] The green dots represent control points for the curves that form the wireframe of the design. The coordinates of some of those points are selected as design parameters for the optimization algorithm. The blue curves depict the wireframe of the device, which are interpolating splines that pass through the control points. The transfinite patches are then defined by selecting three or four curves that correspond to the edges of the patch.

blocks correspond to the self-patch interactions in the matrix in eq 10. To obtain a fully block diagonal matrix out of eq 10, the matrix must be reordered such that all of the unknowns corresponding to a particular discretization point are indexed consecutively, and the unknowns corresponding to each patch are grouped together and ordered sequentially. This leads to a block-diagonal structure with each nonzero block corresponding to the self-patch interactions for each patch. All of the blocks can be LU factorized efficiently in parallel and used as a left-sided preconditioner for GMRES.

Figure 2 shows the effect of preconditioning on the GMRES convergence rate when solving the BIE system in eq 10 for the unoptimized designs of the splitter and 9 μm taper geometries, which are described in Sections 4.1 and 4.2, respectively. The results of using the block preconditioner are also compared to those using no preconditioner and using a different preconditioner based on the sparse LU factorization of the BIE system which includes both the singular and near-singular contributions. The latter preconditioning approach (labeled as "Near Patch Preconditioner") utilizes MKL's Pardiso solver for matrix reordering and factorization, and despite the fact that, as shown in Figure 2 (a) and (c), it has the best rate of convergence in terms of the number of iterations, in general, it is not the most time efficient approach. Indeed, given that the sparsity pattern of the matrix that includes both the singular and near-singular terms can be quite complex, both the memory requirements and computational cost for the LU factorization are significantly higher than the simpler block-diagonal approach. As a practical example of this, the "Splitter" case requires approximately 15 s to achieve a GMRES residual of 10^{-4} when using the simpler block diagonal preconditioner, whereas solving the system using near patch preconditioner instead takes 50 s, a factor of 3.3 times slower. This may easily translate into hours when doing an optimization run which requires numerous solutions of the forward problem and adjoint.

3. SHAPE OPTIMIZATION USING BIE

3.1. Parametrization of Design Parameters. Representing the device geometry in a manner that is both flexible and robust and that is directly amenable to shape optimization using the CBIE method can pose significant challenges. To solve these difficulties, our approach to parametrizing the device is as follows. The cross-section wireframe of the device using a set of control points and splines that pass through those control points is first defined. Following this, the surface patches needed for the CBIE method are constructed via transfinite interpolation[46] from three or four bounding curves. As design optimization parameters, denoted by $\mathbf{p}$, a subset of the control points are selected that can be freely moved to change the shape of the device boundary. In Figure 3, we show a CAD design for a photonic power splitter device produced with the open-source meshing software, Gmsh,[47] whose API was used to form the transfinite patches and change the geometry during the optimization procedure.

3.2. Adjoint Computation of Sensitivities. Although a direct computation of the sensitivities (i.e., the gradient of objective function with respect to the optimization parameters) is possible via a finite-difference approximation of the objective function, the computational cost of such process grows proportionally to the product of the problem size and the number of design variables and quickly becomes prohibitive. An alternative approach is to use the adjoint method,[6,12,48,49] which can significantly reduce the computational cost of sensitivity calculations to a single extra solution of the forward problem using the adjoint of the system matrix operator and some additional sparse matrix-vector operations.

For our BIE-based shape optimization problem, we define a real-valued, scalar objective function J, such as, for example, the field intensity at a given point, constrained to satisfy Maxwell's equations in their BIE discretized form eq 10:

$$\begin{cases} \max\limits_{\mathbf{p}} & J(\mathbf{p}, \mathbf{\Phi}) \\ \text{s. t.} & A(\mathbf{p})\mathbf{\Phi} - b(\mathbf{p}) = 0 \end{cases} \tag{11}$$

The objective function considered here is of the form

$$\begin{aligned} J &= \langle O_1(F^{\text{inc}} + M\mathbf{\Phi}),\, O_2(F^{\text{inc}} + M\mathbf{\Phi})\rangle \\ &= (\mathbf{\Phi}^{\dagger}M^{\dagger} + F^{\text{inc},\dagger})O_1^{\dagger}O_2(F^{\text{inc}} + M\mathbf{\Phi}) \end{aligned} \tag{12}$$

https://doi.org/10.1021/acsphotonics.2c01072
ACS Photonics XXXX, XXX, XXX−XXX

where $M = M(\mathbf{p})$ is a field matrix which maps the densities $\mathbf{\Phi}$ to the evaluated fields at the points of interest, and F^{inc} are the incident field values also at the points of interest. The matrices O_1 and O_2 can be used to select individual polarizations or modes of interest for the specific objective function. This form is very general and can handle a number of very different cases, such the intensity at a given point or power flux across a given area. Another possible form for the objective function suitable for mode-expansion coefficients is

$$J = |\langle F^{\text{mode}}, (F^{\text{inc}} + M\mathbf{\Phi})\rangle|^2 \tag{13}$$

where J is the square of the modal coefficient for the mode specified by the field values F^{mode} and is proportional to the power in the mode.

Let the solution mapping that takes the design parameter vector $\mathbf{p}$ and returns the solution current densities be defined by

$$\mathbf{S}(\mathbf{p}) \equiv [A(\mathbf{p})]^{-1} b(\mathbf{p}) \tag{14}$$

then the reduced form objective function can be written as[50]

$$f(\mathbf{p}) \equiv J(\mathbf{p}, \mathbf{S}(\mathbf{p})) \tag{15}$$

for which the gradient of the objective function w.r.t. the design parameters is

$$[\nabla f]_i = \frac{\partial f}{\partial \mathbf{p}_i}(\mathbf{p}) \tag{16}$$

In terms of J, and including the explicit dependence on the density solution $\mathbf{\Phi}$

$$\frac{\partial f}{\partial \mathbf{p}_i}(\mathbf{p}) = \frac{\partial J}{\partial \mathbf{p}_i}(\mathbf{p}, \mathbf{\Phi}) + \left\langle K(\mathbf{p}, \mathbf{\Phi}), \frac{\partial \mathbf{S}}{\partial \mathbf{p}_i}(\mathbf{p}) \right\rangle \tag{17}$$

with

$$K(\mathbf{p}, \mathbf{\Phi}) = \frac{\partial J}{\partial \mathbf{\Phi}}(\mathbf{p}, \mathbf{\Phi}) \tag{18}$$

On the other hand, the system of integral equations can be written as

$$T(\mathbf{p}) \equiv A(\mathbf{p})\mathbf{S}(\mathbf{p}) - b(\mathbf{p}) = 0 \tag{19}$$

Then, the derivative w.r.t. each of the design parameters has to be equal to zero

$$\frac{\partial T}{\partial \mathbf{p}_i} = \left(\left[\frac{\partial A}{\partial \mathbf{p}_i}\right]\mathbf{S} - \frac{\partial b}{\partial \mathbf{p}_i}\right) + A\frac{\partial \mathbf{S}}{\partial \mathbf{p}_i} = 0 \tag{20}$$

where we have dropped in the notation the explicit dependence on the design parameters $\mathbf{p}$. Since eq 20 is a vector equation, an inner product with an arbitrary vector ψ can be taken to obtain the scalar identity:

$$\left\langle \psi, \left(\left[\frac{\partial A}{\partial \mathbf{p}_i}\right]\mathbf{S} - \frac{\partial b}{\partial \mathbf{p}_i}\right) + A\frac{\partial \mathbf{S}}{\partial \mathbf{p}_i} \right\rangle = 0 \tag{21}$$

Using some fundamental distributive properties of the inner product, as well as the definition of the adjoint operator in an inner product space, the following identity holds:

$$\left\langle \psi, \left[\frac{\partial A}{\partial \mathbf{p}_i}\right]\mathbf{S} - \frac{\partial b}{\partial \mathbf{p}_i} \right\rangle + \left\langle A^{\dagger}\psi, \frac{\partial \mathbf{S}}{\partial \mathbf{p}_i} \right\rangle = 0 \tag{22}$$

Adding eq 22 to eq 17 results in

$$\frac{\partial f}{\partial \mathbf{p}_i} = \left\{\frac{\partial J}{\partial \mathbf{p}_i} + \left\langle \psi, \left[\frac{\partial A}{\partial \mathbf{p}_i}\right]\mathbf{S} - \frac{\partial b}{\partial \mathbf{p}_i} \right\rangle\right\} + \left\langle K + A^{\dagger}\psi, \frac{\partial \mathbf{S}}{\partial \mathbf{p}_i} \right\rangle \tag{23}$$

Finding the adjoint density ψ^* which satisfies

$$A^{\dagger}\psi^* = -K \tag{24}$$

allows us to avoid computing the change in the current densities when changing each individual parameter $\mathbf{p}_i$, which is computationally expensive since it requires solving an additional linear system per design parameter. Thus, the gradient can be obtained in terms of only geometrical changes, which are significantly cheaper (computationally speaking) to obtain. By combining eq 23 and eq 24, the gradient can be computed as

$$\frac{\partial f}{\partial \mathbf{p}_i} = \frac{\partial J}{\partial \mathbf{p}_i} + \left\langle \psi^*, \left[\frac{\partial A}{\partial \mathbf{p}_i}\right]\mathbf{S} - \frac{\partial b}{\partial \mathbf{p}_i} \right\rangle \tag{25}$$

The most computationally intensive term in eq 25 is the partial derivative of the integral equation system with respect to each optimization parameter. When each optimization parameter only locally affects the device shape, this term is a sparse matrix-vector product that can be efficiently computed, as explained in the following Section 3.3.

3.3. Sparse Computation of Geometrical Contributions. Each optimization parameter $\mathbf{p}_i$ affects a portion of the discretized geometry. Although a single design parameter can, in general, affect the whole parametrization globally, it can also be localized to a specific feature of the design. Consider a parametrization of the design which consists of M nonoverlapping rectangular curvilinear patches. Define the sets P_i to consist of all the patches that are affected when $\mathbf{p}_i$ is changed. This set can be obtained automatically by means of a simple preprocessing step: each parameter $\mathbf{p}_i$ is perturbed in sequence, and the patches which change coordinates for the given discretization are detected and recorded.

Denote by

$$\left[\frac{\partial A}{\partial \mathbf{p}_i}\right]_{m,n} \tag{26}$$

the block of the matrix for the m-th target patch, and the n-th source patch. Numerically, this matrix can be approximated by

$$\left[\frac{\partial A}{\partial \mathbf{p}_i}\right]_{m,n} = \begin{cases} \dfrac{[A(\mathbf{p} + \delta\mathbf{e}_i)]_{m,n} - [A(\mathbf{p} - \delta\mathbf{e}_i)]_{m,n}}{2\delta}, & m \text{ or } n \in P_i \\ 0, & \text{otherwise} \end{cases} \tag{27}$$

where, as before, the matrix entries are never explicitly computed, but rather just the action of the matrix.

Combining eq 25 and eq 27, the sensitivities for the objective function can be computed at the cost of two system

https://doi.org/10.1021/acsphotonics.2c01072
ACS Photonics XXXX, XXX, XXX−XXX

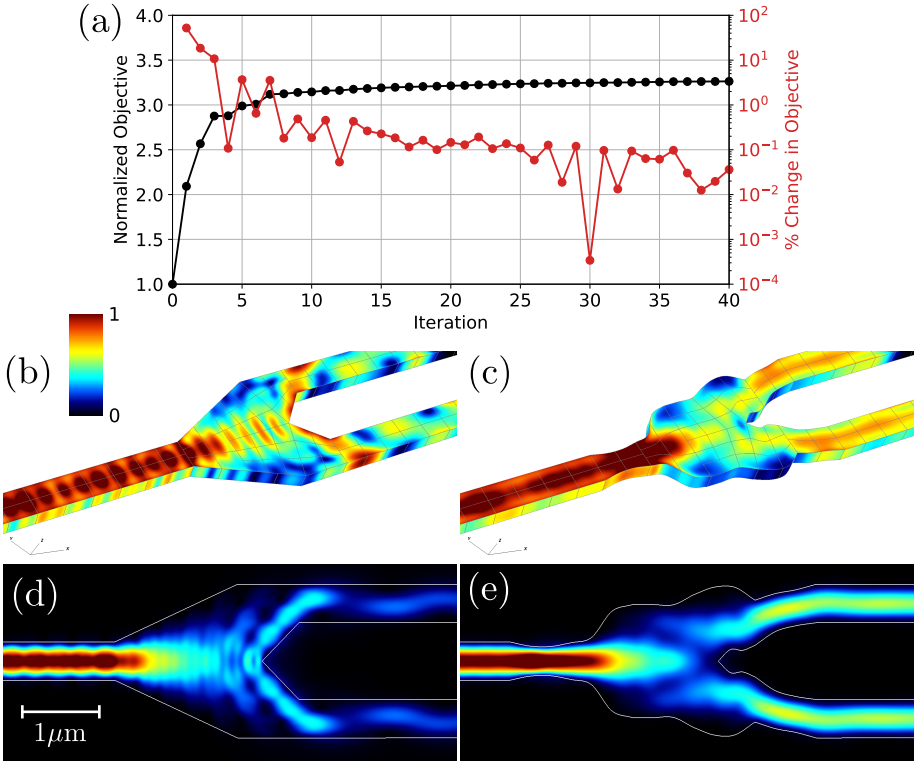


Figure 4. A 1:2 power splitter optimized using the 3D CBIE framework. Panel (a) plots the normalized objective function as a function of the optimization iteration number, as well as the percent change with at each iteration. Panels (b) and (c) plot the BIE **m** surface density at the device boundaries for the initial and optimized designs, respectively. Panels (d) and (e) show the final verification using Lumerical, plotting the absolute value of the Poynting vector for the initial and optimized designs.

solves (using GMRES), one direct and one adjoint, plus P sparse operations eq 27, with P being the total number of design parameters. This computational cost is greatly advantageous when compared to the direct finite-difference approximation of the gradient which requires $P + 1$ system solves.

4. RESULTS

In order to demonstrate the proposed optimization scheme described in the previous sections, we used the framework to inverse design a 1:2 power splitter and two different nonadiabatic silicon taper devices as design examples.

4.1. Power Splitter. The first device optimized was a 1:2 power splitter with a 250 nm silicon (n_{Si} = 3.48) device layer, background medium of SiO_2(n_{SiO_2} = 1.44), and a free-space wavelength of 1550 nm. The design parameters are 19 y-coordinates of the spline control points of the waveguide boundaries (see Figure 3), and a mirror symmetry constraint is enforced on the bottom curves. Gradient ascent was used for optimization, with a line search for the step size which enforces improvement in the objective function at each iteration. Note that any other gradient-based optimization can be readily used with the framework presented in this work. The optimization history, initial and optimized designs are shown in Figure 4.

The initial design consists of straight lines connecting the input to the output waveguides, and a set of TE dipole sources was used to excite the fundamental mode of the waveguide. The objective function is the $|E_y|^2$ value at a point inside the top output waveguide. Due to the symmetry constraints, this objective function results in a design that splits the fields evenly between the top and bottom waveguides. In the optimization history, Figure 4 (a), it can be seen that the optimizer improves the design drastically after only 7 iterations, after which, all changes from one iteration to the next one are under 1% (relative to the objective at the current iteration). The overall optimization of the structure, which was discretized using 224 224 unknowns, took 4 h and 12 min using a server with two Intel Xeon Gold 6154 (36 cores total) and a single NVIDIA Titan V GPU. A total of 102 linear system solutions were required—including the forward and adjoint problems as well as forward solutions needed for the line search of the optimizer—for the 40 iterations it took to converge. In order to reduce the number of GMRES iterations needed per solve, we used the solution from the previous optimization iteration as an initial guess for GMRES. Figure 4 (b,c) displays the absolute value of the **m** density at the boundaries for the initial and optimized designs, respectively. To verify the efficiency of the final design, we used the commercial software Lumerical[51] to simulate the device by launching the fundamental mode in the input (left) waveguide and monitoring the output modal and total power at the top and bottom output waveguides. Figure 4 (d,e) plots the absolute value of the Poynting vector of the initial and optimized designs. The initial mode efficiency, calculated as the output power coupling to the fundamental mode on both of the output waveguides, was only 34.09%, while the optimized design has a 96.52% modal efficiency. Table 2 summarizes the efficiency of the initial and optimized designs for the splitter, as well as for the rest of the examples.

4.2. Nonadiabatic Waveguide Tapers. The second example consists of a nonadiabatic waveguide taper, whose purpose is to efficiently transition from a small width

https://doi.org/10.1021/acsphotonics.2c01072
ACS Photonics XXXX, XXX, XXX−XXX

Table 2. Efficiency of the Initial and Optimized Designs

	initial design		optimized design	
model	mode efficiency (%)	total power (%)	mode efficiency (%)	total power (%)
splitter	34.09	51.68	96.52	97.20
6 μm taper	82.97	98.83	93.57	98.11
9 μm taper	17.65	73.12	91.01	96.22

waveguide to a larger one while maintaining the energy in the fundamental mode. Such waveguide tapers are typically designed in an adiabatic way, such that the width of the tapered region increases slowly to minimize modal conversion and scattering power losses. This can result in very large designs, and in order to reduce the size of these devices, nonadiabatic designs can be employed. By using complex geometrical taper boundaries, high efficiencies can be achieved over taper lengths that are significantly shorter than the lengths needed for the adiabatic approximation to be valid.[52] We used our CBIE optimization framework to design two nonadiabatic tapers consisting of a 220 nm silicon device layer, also embedded in a SiO_2 background medium, and free-space wavelength of 1550 nm. The design region consisted of a length of 18 μm (40.4 wavelengths in the Silicon), with design parameters being the y-coordinate of the control points for the top bounding curve of the taper (resulting in 19 design parameters). Similar to the power splitter example, mirror symmetry constraints are enforced for the bottom bounding curve.

The first taper consists of an input waveguide, corresponding to the left waveguide in Figure 5, with a width of 0.5 μm and an output waveguide with 6 μm width. In this case, the optimization problem was prescribed by using a group of incident TE dipoles distributed on a line on the right (originally the output) waveguide and maximizing the $|E_y|^2$ value at the center of the left (smaller) waveguide. (An alternative approach for this problem would be to launch the input field on the left waveguide, and use the fundamental mode power of the output waveguide as the objective.) The initial design consists of a linearly tapering waveguide to connect the left and right waveguides. To verify the results against a commercial solver, Lumerical[51] was used to compute the modal efficiencies of the initial and final structures, which resulted in 82.97% for the initial design, and 93.57% for the optimized design. Figure 5 plots (a) the optimization history, and (b) the initial and (c) optimized values of the absolute value of the Poynting vector at the midplane of the silicon device, obtained by the Lumerical simulation. The optimization of this taper, discretized with 401 600 unknowns, required 18 h and 47 min on the same computer as the splitter. For this case, a total of 17 optimization iterations were needed for the optimization, which accounted for 53 linear system solutions. The increase in time per solve, when compared to the splitter, is due to the larger propagation size of the device, which translates into GMRES requiring a significantly larger amount of iterations to converge to the prescribed residual tolerance of 1×10^{-3}—about 600 iterations per solve. Additionally, the total number of unknowns is almost twice what was used for the splitter.

The second taper differs from the first one by having a larger output width of 9 μm (50% larger than the first one), while the length of tapered region is kept the same at 18 μm (40.4 wavelengths inside the silicon). This device was more

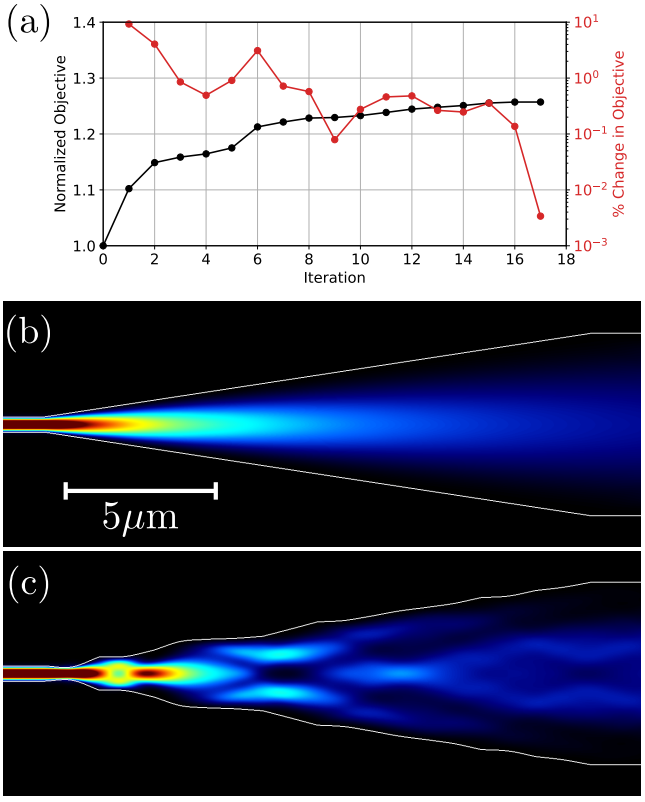


Figure 5. CBIE optimization of a taper with an input waveguide width of 0.5 μm and an output width of 6 μm over an 18 μm design region. Panel (a) shows the optimization history, and (b) and (c) show the absolute value of the Poynting vector for the initial and optimized designs, respectively. The fundamental mode efficiency went from 82.97% for the initial device, to 93.57% for the optimized device.

challenging to optimize due to the existence of multiple local minima and required starting the optimization process with different randomized initial conditions. In Figure 6, we show the result of the best optimization result obtained, in which the final design achieved a mode efficiency of 91.01%. In this case, the initial condition, shown in Figure 6 (b), had a mode efficiency of only 17.65%. For reference, a straight taper has a 48.71% mode efficiency. The optimized design therefore achieved a 42.3% improvement with respect to the straight taper, and a 73.36% improvement with respect to the initial random condition. In Figure 6 (d), the CBIE patch and mesh structure used for this simulation are displayed, where the total number of system unknowns was 625 600. Note that all patches represent curvilinear meshes that conform to the design boundaries. The total optimization time for this problem was 48 h and 5 min (as before, using the same hardware as that for the splitter), with a total of 87 linear system solves and 34 optimization iterations.

5. DISCUSSION AND CONCLUSION

This Article has introduced a new simulation and boundary-based inverse design methodology for nanophotonic devices. We have shown that the Chebyshev-based Boundary Integral Equation (CBIE) method can be used to model photonic devices by solving the fully vectorial 3D Maxwell equations efficiently. Representing the device geometries via curvilinear logically quadrilateral patches and the corresponding unknown surface densities using a high-order polynomial basis leads to

https://doi.org/10.1021/acsphotonics.2c01072
ACS Photonics XXXX, XXX, XXX−XXX

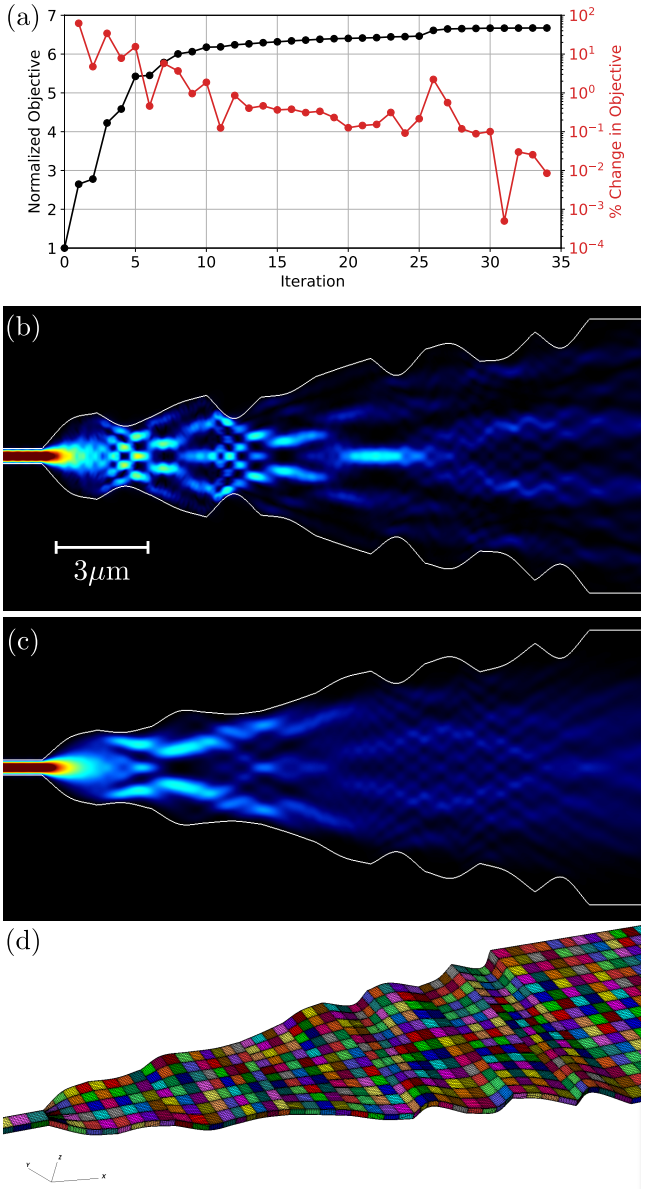


Figure 6. Optimization of a nonadiabatic taper with an input waveguide width of 0.5 μm and an output width of 9 μm over an 18 μm design region. Panel (a) plots the optimization history, while (b) and (c) show the absolute value of the Poynting vector for the initial and optimized designs, respectively. In (d), the CBIE patch and mesh structure of the final design is displayed.

accurate frequency-domain field solutions while only requiring coarse mesh discretizations. Due to the significant added complexity and increased degrees of freedom required for full 3D simulation of nanophotonic devices, we have developed and incorporated three crucial advancements to the original CBIE approach: (1) the expensive source-to-target far interactions are calculated using GPU computing, (2) we have designed and implemented a sparse block-diagonal matrix preconditioner, which can be rapidly constructed and applied, leading to dramatic reductions in the number of iterations required for solution convergence, and (3) we have derived and implemented a matrix-free approach for computing a matrix-vector product with the adjoint of the system matrix, which is necessary for implementation of the discrete adjoint method without requiring explicit formation of any matrices.

As design examples, we demonstrated the inverse design of a 1550 nm power splitter and two nonadiabatic mode-preserving tapers. To the best of our knowledge, the taper designs represent the largest silicon photonic tapers inverse designed using full-wave solution of Maxwell's equations in three dimensions. Current work seeks to leverage algorithmic acceleration techniques (such as the fast multipole method) to speed-up solution of the forward problem further.

■ ASSOCIATED CONTENT

Supporting Information

The Supporting Information is available free of charge at https://pubs.acs.org/doi/10.1021/acsphotonics.2c01072.

Matrix discretization of Chebyshev-based integral operators, details on the Chebyshev-based integral equation dielectric matrices (PDF)

■ AUTHOR INFORMATION

Corresponding Author

Constantine Sideris − *Department of Electrical and Computer Engineering, University of Southern California, Los Angeles, California 90089, United States;* orcid.org/0000-0002-3042-4889; Email: csideris@usc.edu

Author

Emmanuel Garza − *Department of Electrical and Computer Engineering, University of Southern California, Los Angeles, California 90089, United States*

Complete contact information is available at: https://pubs.acs.org/10.1021/acsphotonics.2c01072

Funding

This work was supported in part by the National Science Foundation under Grants CCF-1849965 and CCF-2047433, in part by the Air Force Office of Scientific Research under Grant FA9550-20-1-0087, and in part by the National Science Foundation through the Computing Research Association for the CIFellows Project under Grant #2030859.

Notes

The authors declare no competing financial interest.

■ REFERENCES

(1) Heck, M. J.; Chen, H.-W.; Fang, A. W.; Koch, B. R.; Liang, D.; Park, H.; Sysak, M. N.; Bowers, J. E. Hybrid silicon photonics for optical interconnects. *IEEE J. Sel. Top. Quantum Electron.* **2011**, *17*, 333−346.

(2) Luan, E.; Shoman, H.; Ratner, D. M.; Cheung, K. C.; Chrostowski, L. Silicon photonic biosensors using label-free detection. *Sensors* **2018**, *18*, 3519.

(3) Tombez, L.; Zhang, E.; Orcutt, J.; Kamlapurkar, S.; Green, W. Methane absorption spectroscopy on a silicon photonic chip. *Optica* **2017**, *4*, 1322−1325.

(4) Wang, J.; Sciarrino, F.; Laing, A.; Thompson, M. G. Integrated photonic quantum technologies. *Nat. Photonics* **2020**, *14*, 273−284.

(5) Shastri, B. J.; Tait, A. N.; Ferreira de Lima, T.; Pernice, W. H.; Bhaskaran, H.; Wright, C. D.; Prucnal, P. R. Photonics for artificial intelligence and neuromorphic computing. *Nat. Photonics* **2021**, *15*, 102−114.

(6) Molesky, S.; Lin, Z.; Piggott, A. Y.; Jin, W.; Vucković, J.; Rodriguez, A. W. Inverse design in nanophotonics. *Nat. Photonics* **2018**, *12*, 659−670.

(7) Piggott, A. Y.; Petykiewicz, J.; Su, L.; Vučković, J. Fabrication-constrained nanophotonic inverse design. *Sci. Rep.* **2017**, *7*, 1−7.

(8) Sideris, C.; Garza, E.; Bruno, O. P. Ultrafast simulation and optimization of nanophotonic devices with integral equation methods. *ACS Photonics* **2019**, *6*, 3233−3240.
(9) Piggott, A. Y.; Lu, J.; Lagoudakis, K. G.; Petykiewicz, J.; Babinec, T. M.; Vučković, J. Inverse design and demonstration of a compact and broadband on-chip wavelength demultiplexer. *Nat. Photonics* **2015**, *9*, 374−377.
(10) Sideris, C.; Khachaturian, A.; White, A. D.; Bruno, O. P.; Hajimiri, A. Foundry-fabricated grating coupler demultiplexer inverse-designed via fast integral methods. *Commun. Phys.* **2022**, *5*, 1−8.
(11) So, S.; Badloe, T.; Noh, J.; Bravo-Abad, J.; Rho, J. Deep learning enabled inverse design in nanophotonics. *Nanophotonics* **2020**, *9*, 1041−1057.
(12) Niederberger, A. C. R.; Fattal, D. A.; Gauger, N. R.; Fan, S.; Beausoleil, R. G. Sensitivity analysis and optimization of sub-wavelength optical gratings using adjoints. *Opt. Express* **2014**, *22*, 12971.
(13) Deng, Y.; Korvink, J. G. Topology optimization for three-dimensional electromagnetic waves using an edge element-based finite-element method. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Science* **2016**, *472*, 20150835.
(14) Nikolova, N. K.; Safian, R.; Soliman, E. A.; Bakr, M. H.; Bandler, J. W. Accelerated Gradient Based Optimization Using Adjoint Sensitivities. *IEEE Transactions on Antennas and Propagation* **2004**, *52*, 2147−2157.
(15) Hammer, M.; Ivanova, O. V. Effective index approximations of photonic crystal slabs: a 2-to-1-D assessment. *Optical and quantum electronics* **2009**, *41*, 267−283.
(16) Liu, V.; Fan, S. S4: A free electromagnetic solver for layered periodic structures. *Comput. Phys. Commun.* **2012**, *183*, 2233−2244.
(17) Di Domenico, G.; Weisman, D.; Panichella, A.; Roitman, D.; Arie, A. Large-Scale Inverse Design of a Planar On-Chip Mode Sorter. *ACS Photonics* **2022**, *9*, 378−382.
(18) Shin, W.; Fan, S. Choice of the perfectly matched layer boundary condition for frequency-domain Maxwell's equations solvers. *J. Comput. Phys.* **2012**, *231*, 3406−3431.
(19) Bruno, O. P.; Lyon, M.; Pérez-Arancibia, C.; Turc, C. Windowed Green Function Method for Layered-Media Scattering. *SIAM Journal on Applied Mathematics* **2016**, *76*, 1871−1898.
(20) Pérez-Arancibia, C.*Windowed integral equation methods for problems of scattering by defects and obstacles in layered media*. Ph.D. thesis, California Institute of Technology, 2016.
(21) Bruno, O. P.; Pérez-Arancibia, C. Windowed Green function method for the Helmholtz equation in the presence of multiply layered media. *Proc. R. Soc. A Math. Phys. Eng. Sci.* **2017**, *473*, 20170161.
(22) Bruno, O. P.; Garza, E.; Pérez-Arancibia, C. Windowed Green function method for nonuniform open-waveguide problems. *IEEE Trans. Antennas Propag.* **2017**, *65*, 4684−4692.
(23) Garza, E.; Sideris, C.; Bruno, O. P.A boundary integral method for 3D nonuniform dielectric waveguide problems via the windowed Green function. *arXiv*, October 21, 2021. DOI: 10.48550/arXiv.2110.11419 (accessed on September 30, 2022).
(24) Volakis, J.; Kubilay, S.*Integral Equation Methods for Electromagnetics*; SciTech Publishing, Inc.: Raleigh, NC, 2012.
(25) Nédélec, J.-C.*Acoustic and Electromagnetic Equations: Integral Representations for Harmonic Problems*, 1st ed.; Springer, 2001.
(26) Colton, D. L.; Kress, R.; Kress, R.*Inverse Acoustic and Electromagnetic Scattering Theory*, 3rd ed.; Springer, 2013.
(27) Bruno, O.; Elling, T.; Paffenroth, R.; Turc, C. Electromagnetic integral equations requiring small numbers of Krylov-subspace iterations. *J. Comput. Phys.* **2009**, *228*, 6169−6183.
(28) Müller, C.*Foundations of the Mathematical Theory of Electromagnetic Waves*, 1st ed.; Springer-Verlag: Berlin, NY, 1969.
(29) Ylä-Oijala, P.; Taskinen, M. Well-conditioned Müller formulation for electromagnetic scattering by dielectric objects. *IEEE Trans. Antennas Propag.* **2005**, *53*, 3316−3323.
(30) Hu, J.; Garza, E.; Sideris, C. A Chebyshev-Based High-Order-Accurate Integral Equation Solver for Maxwell's Equations. *IEEE Trans. Antennas Propag.* **2021**, *69*, 5790−5800.
(31) Zhang, L.; Lee, J. H.; Oskooi, A.; Hochman, A.; White, J. K.; Johnson, S. G. A Novel Boundary Element Method Using Surface Conductive Absorbers for Full-Wave Analysis of 3-D Nanophotonics. *J. Lightw. Technol.* **2011**, *29*, 949−959.
(32) Garza, E.*Boundary integral equation methods for simulation and design of photonic devices*. Ph.D. thesis, California Institute of Technology, 2020.
(33) Bruno, O. P.; Garza, E. A Chebyshev-based rectangular-polar integral solver for scattering by geometries described by non-overlapping patches. *J. Comput. Phys.* **2020**, *421*, 109740.
(34) Garza, E.; Hu, J.; Sideris, C.High-order Chebyshev-based Nyström Methods for Electromagnetics. *2021 International Applied Computational Electromagnetics Society Symposium (ACES)*; ACES, 2021; pp 1−4.
(35) Saad, Y.; Schultz, M. H. GMRES: A generalized minimal residual algorithm for solving nonsymmetric linear systems. *SIAM J. Sci. Stat. Comput.* **1986**, *7*, 856−869.
(36) Greengard, L.; Rokhlin, V. A fast algorithm for particle simulations. *J. Comput. Phys.* **1987**, *73*, 325−348.
(37) Greengard, L.; Huang, J.; Rokhlin, V.; Wandzura, S. Accelerating fast multipole methods for the Helmholtz equation at low frequencies. *IEEE Comput. Sci. Eng.* **1998**, *5*, 32−38.
(38) Benson, A. R.; Poulson, J.; Tran, K.; Engquist, B.; Ying, L. A Parallel Directional Fast Multipole Method. *SIAM Journal on Scientific Computing* **2014**, *36*, C335−C352.
(39) Gumerov, N. A.; Duraiswami, R.*Fast Multipole Methods for the Helmholtz Equation in Three Dimensions*, 1st ed.; Elsevier Ltd.: Kidlington, Oxford, 2004.
(40) Bruno, O. P.; Kunyansky, L. A. Surface scattering in three dimensions: an accelerated high-order solver. *Proc. R. Soc. London A* **2001**, *457*, 2921−2934.
(41) Bauinger, C.; Bruno, O. P. Interpolated Factored Green Function" method for accelerated solution of scattering problems. *J. Comput. Phys.* **2021**, *430*, 110095.
(42) Ylä-Oijala, P.; Taskinen, M.; Järvenpää, S. Surface integral equation formulations for solving electromagnetic scattering problems with iterative methods. *Radio Science* **2005**, *40*, 1−19.
(43) Carpentieri, B.*New Trends in Computational Electromagnetics*; Institution of Engineering and Technology, 2019; pp 535−566.
(44) Adrian, S. B.; Dely, A.; Consoli, D.; Merlini, A.; Andriulli, F. P. Electromagnetic Integral Equations: Insights in Conditioning and Preconditioning. *IEEE Open Journal of Antennas and Propagation* **2021**, *2*, 1143−1174.
(45) Bruno, O. P.; Pandey, A.Fast, higher-order direct/iterative hybrid solver for scattering by Inhomogeneous media−with application to high-frequency and discontinuous refractivity problems. *arXiv*, July 12, 2019. DOI: 10.48550/arXiv.1907.05914 (accessed on September 30, 2022).
(46) Gordon, W. J.; Hall, C. A. Construction of curvilinear co-ordinate systems and applications to mesh generation. *International Journal for Numerical Methods in Engineering* **1973**, *7*, 461−477.
(47) Geuzaine, C.; Remacle, J.-F. Gmsh: A 3-D finite element mesh generator with built-in pre-and post-processing facilities. *International journal for numerical methods in engineering* **2009**, *79*, 1309−1331.
(48) Giles, M. B.; Pierce, N. A. An Introduction to the Adjoint Approach to Design. *Flow, Turbulence and Combustion* **2000**, *65*, 393−415.
(49) Lalau-Keraly, C. M.; Bhargava, S.; Miller, O. D.; Yablonovitch, E. Adjoint shape optimization applied to electromagnetic design. *Opt. Express* **2013**, *21*, 21693−21701.
(50) De los Reyes, J. C.*Numerical PDE-Constrained Optimization*, 1st ed.; Springer: New York, 2015.
(51) *FDTD: 3D Electromagnetic Simulator*; Lumerical Inc.https://www.lumerical.com/ (accessed on September 30, 2022).

https://doi.org/10.1021/acsphotonics.2c01072
ACS Photonics XXXX, XXX, XXX−XXX

(52) Luyssaert, B.; Bienstman, P.; Vandersteegen, P.; Dumon, P.; Baets, R. Efficient nonadiabatic planar waveguide tapers. *Journal of Lightwave Technology* **2005**, *23*, 2462−2468.

Recommended by ACS

Manifold Learning for Knowledge Discovery and Intelligent Inverse Design of Photonic Nanostructures: Breaking the Geometric Complexity

Mohammadreza Zandehshahvar, Ali Adibi, *et al.*

JANUARY 24, 2022
ACS PHOTONICS

READ

Elucidating the Behavior of Nanophotonic Structures through Explainable Machine Learning Algorithms

Christopher Yeung, Aaswath P. Raman, *et al.*

JULY 20, 2020
ACS PHOTONICS

READ

Structural Optimization of a One-Dimensional Freeform Metagrating Deflector via Deep Reinforcement Learning

Dongjin Seo, Min Seok Jang, *et al.*

DECEMBER 30, 2021
ACS PHOTONICS

READ

Empowering Metasurfaces with Inverse Design: Principles and Applications

Zhaoyi Li, Federico Capasso, *et al.*

JUNE 09, 2022
ACS PHOTONICS

READ

Get More Suggestions >

https://doi.org/10.1021/acsphotonics.2c01072
ACS Photonics XXXX, XXX, XXX−XXX