Fast Scattering Matrix Computation for Complex Media and Metasurfaces

Ho-Chun Lin, Zeyu Wang, and Chia Wei Hsu*

Ming Hsieh Department of Electrical and Computer Engineering, University of Southern California, Los Angeles, USA
*corresponding author, E-mail: cwhsu@usc.edu

Abstract

We use the Schur complement and partial factorization to compute the scattering matrices of large-scale complex optical systems and large-area metasurfaces, achieving orders-of-magnitude speed-up and reduced memory usage over existing methods.

1. Introduction and Background

The linear transport of any static optical system is fully characterized by its scattering matrix $\mathbf{S}$, which maps any incident wavefront β (expressed as a vector) to the corresponding outgoing wavefront α through $\alpha_n = \sum_m S_{nm}\beta_m$. Scattering matrices are used extensively in wavefront shaping and imaging in scattering media [1], and they also fully describe the incident-angle-dependent transmission and reflection from metasurfaces [2] and the input-output relations in photonic circuits [3], with no approximation. However, numerical computation of the S-matrix requires significant computing time and memory, so existing methods can only compute the S-matrix for small systems.

Electromagnetic scattering problems in frequency domain can be written as a system of linear equations $\mathbf{A}x = b$ where matrix $\mathbf{A} = -(\omega/c)^2\varepsilon_\mathrm{r}(\omega,\mathbf{r}) + \nabla\times\mu_\mathrm{r}^{-1}(\omega,\mathbf{r})\nabla\times$ is the electric-field Maxwell operator, and x and b are the electric field profile $\mathbf{E}(\mathbf{r})$ and the equivalent source profile respectively written as column vectors. Given one input b_m, solving the scattering problem reduces to the standard problem of solving for $x_m = \mathbf{A}^{-1}b_m$. However, this only yields the m-th column of the S-matrix. To characterize the system's response to $M \gg 1$ distinct input states, such computation needs to be repeated M times.

Iterative methods for solving $x_m = \mathbf{A}^{-1}b_m$ can minimize memory usage but do not scale well with M. Direct methods can reuse the LU factorization of matrix $\mathbf{A}$ but take more memory. Also, both methods involve repetition over the M input states so they are slow when M is large, and they both compute unnecessary information such as field profiles in the interior of the system. The recursive Green's function method (RGF) and rigorous coupled-wave analysis (RCWA) can compute the full S-matrix without looping over inputs but scale poorly with the system size as they do not make use of the sparsity of matrix $\mathbf{A}$ in the transverse directions.

Here, we propose and implement a new frequency-domain method that can compute the entire S-matrix in one shot with no repetition while making full use of the sparsity of all matrices involved. Our method is applicable to complex systems with arbitrary $\varepsilon_\mathrm{r}(\omega,\mathbf{r})$ and $\mu_\mathrm{r}(\omega,\mathbf{r})$ tensors, and for large systems it is orders of magnitude faster than all of the above-mentioned methods while using less memory.

2. Methods

We start by recognizing that the scattering matrix $\mathbf{S}$ can be concisely written as

$$\mathbf{S} = \mathbf{C}\mathbf{A}^{-1}\mathbf{B} - \mathbf{D}, \tag{1}$$

where the m-th column of matrix $\mathbf{B} = [b_1, \ldots, b_M]$ is the equivalent source profile that generates the m-th input state, the n-th row of matrix $\mathbf{C}$ is the conjugated profile of the n-th output state on the surface or a near-to-far-field transformation, and matrix $\mathbf{D}$ optionally subtracts incident fields from the input/output configuration. Under spatial discretization, matrices $\mathbf{A}$, $\mathbf{B}$, $\mathbf{C}$, $\mathbf{D}$ are all sparse.

Traditional iterative and direct methods compute $\mathbf{X} = \mathbf{A}^{-1}\mathbf{B}$, which contains much more information than is needed. To compute only what is needed, we directly evaluate the much smaller matrix $\mathbf{C}\mathbf{A}^{-1}\mathbf{B}$ by constructing a new sparse matrix $\mathbf{K}$ and performing a $partial$ LU factorization on it,

$$\mathbf{K} \equiv \begin{bmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{C} & \mathbf{D} \end{bmatrix} = \begin{bmatrix} \mathbf{L} & \mathbf{0} \\ \mathbf{E} & \mathbf{I} \end{bmatrix} \begin{bmatrix} \mathbf{U} & \mathbf{F} \\ \mathbf{0} & \mathbf{H} \end{bmatrix}. \tag{2}$$

Here, $\mathbf{L}$ and $\mathbf{U}$ are lower-triangular and upper-triangular matrices, and $\mathbf{I}$ is the identity matrix. The factorization is partial as it stops after factorizing the upper-left block of $\mathbf{K}$ into $\mathbf{A} = \mathbf{L}\mathbf{U}$. Importantly, such $\mathbf{L}$ and $\mathbf{U}$ factors are *not* what we are interested in. By equating the middle and the right-hand sides of Eq. (2) for each of the four blocks, we can see that matrix $\mathbf{H}$, called the Schur complement, satisfies $\mathbf{H} = \mathbf{D} - \mathbf{C}\mathbf{A}^{-1}\mathbf{B}$ and gives the scattering matrix in Eq. (1). The Schur complement is often used in domain decomposition methods [4], but here we use it in an unconventional way to compute the scattering matrix $\mathbf{S}$ in a single step with no repetition and without evaluating $\mathbf{X} = \mathbf{A}^{-1}\mathbf{B}$.

The computing time of the proposed method only depends weakly on the number M of input states because the partial factorization time is dominated by the large matrix $\mathbf{A}$ instead of the smaller matrices $\mathbf{B}$ and $\mathbf{C}$; the number of non-zero elements in $\mathbf{B}$ and $\mathbf{C}$ can be further reduced through a change of basis. The sparsity patterns of all matrices are maintained and can be fully utilized in the partial factorization process. Furthermore, storing matrices $\mathbf{L}$

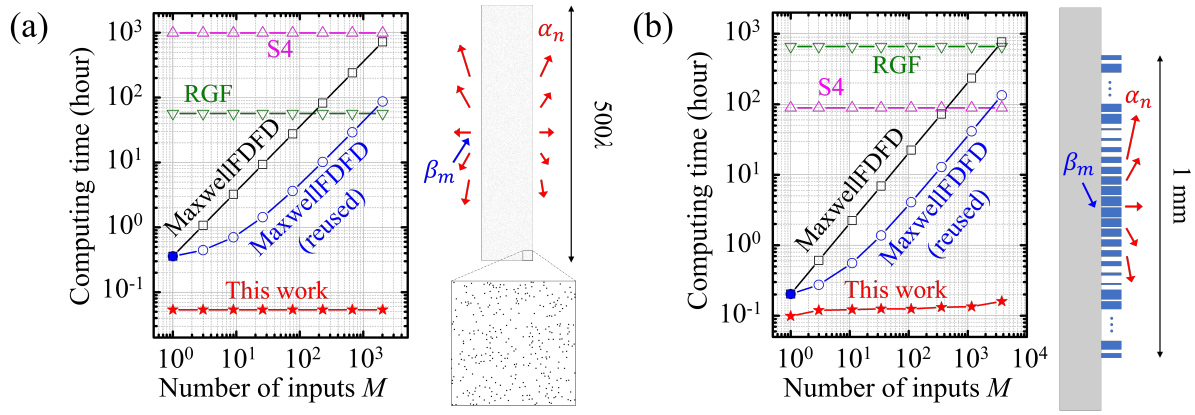

Figure 1: Computation time of the proposed method and other methods when solving for M columns of the scattering matrix, for (a) a $500\lambda \times 100\lambda$ disordered medium and (b) a 1 mm $\times$ 600 nm metalens. Solid symbols are raw data, and open symbols are data extrapolated from smaller M or smaller systems.

	This work	MaxwellFDFD	RGF	S4
Disordered medium	11.0	173.6	7.0	∼1200
Metalens	33.3	118.8	∼700	∼600

Table 1: Memory usage, in GB. The RGF and S4 values are extrapolated from smaller systems.

and $\mathbf{U}$ is typically the memory bottleneck for direct methods, but our method does not use $\mathbf{L}$ and $\mathbf{U}$ so we can drop them during the factorization process to significantly reduce memory usage.

3. Results

We implement this S-matrix solver for the transverse-magnetic waves in 2D under finite-difference discretization, with perfectly matched layer (PML) boundaries. We use the MUMPS package [5] with AMD ordering [6] to compute the partial factorization, with benchmarks done on an Intel Xeon Gold 6130 Processor with 184 GB memory, for two large-scale systems (Fig. 1). For the first system, we compute the full scattering matrix of a disordered medium with width $W = 500\lambda$ and thickness $L = 100\lambda$ using a coarse discretization ($\Delta x = \lambda_0/15$). For the second system, we compute the transmission matrix of a titanium dioxide metalens with parabolic phase profile [7, 8], with diameter $W = 1$ mm, thickness $L = 600$ nm, NA = 0.8, at wavelength $\lambda_0 = 532$ nm, using a finer discretization with $\Delta x = \lambda_0/40$.

Fig. 1 and Table 1 show the computation time and memory usage of the proposed method in comparison to (1) a conventional direct method implemented in MaxwellFDFD [9] where the full computation is repeated M times, (2) MaxwellFDFD modified to have the LU factors stored and reused for the M inputs, (3) RGF implemented in [10], and (4) RCWA implemented in S4 [11]. The proposed method achieves an almost M-independent computation time and is orders of magnitudes faster than the fastest alternative method, while also using much less memory.

The proposed method can readily apply to 3D vectorial simulations, with arbitrary discretization schemes, and can be generalized to compute other quantities such as force, torque, and differential scattering cross sections.

References

[1] S. Rotter and S. Gigan. *Rev. Mod. Phys.* **89**, 015005 (2017).

[2] Z. Lin *et. al. Appl. Phys. Lett.* **118**, 041104 (2021)

[3] W. Bogaerts *et. al. Nature* **586**, 207–216 (2020).

[4] F. Zhang. *The Schur Complement and Its Applications* (Springer, 2005).

[5] P. R. Amestoy, I. S. Duff, J.-Y. LExcellent, and J. Koster. *SIAM J. Matrix Anal. Appl.* **23**, 15–41 (2001).

[6] P. R. Amestoy, T. A. Davis, and I. S. Duff, SIAM J. Matrix Anal. Appl., **17**, 886–905 (1996).

[7] M. Pu *et. al. Opt. Express* **25**, 31471 (2017).

[8] A. Martins *et. al. ACS Photonics* **7**, 2073–2079 (2020).

[9] W. Shin and S. Fan. *Comput. Phys. Commun.* **231**, 3406–3431 (2012).

[10] C. W. Hsu, RGF, https://github.com/chiaweihsu/RGF.

[11] V. Liu and S. Fan. *Comput. Phys. Commun.* **183**, 2233–2244 (2012).