Fast Analysis and Optimization of Sparsely Distributed Partial Modification Problems

Xiaoxing Fang, *Member, IEEE*, Alexander Heldring, Juan M. Rius, *Senior Member, IEEE*, and Qunsheng Cao

***Abstract*— This article addresses the efficient numerical analysis of sparsely distributed small modifications in a large structure, that is, the sparsely distributed partial modification problem (SDPMP). A hierarchical matrix (H-matrix)-based fast direct solver is adopted to solve the SDPMP in a very short time for each new set of modifications. The main idea of this method is to reuse the parts of the compressed inverse matrix operator that do not change and only recompute the modified parts. In contrast with already existing algorithms designed to tackle a single localized modification, this new approach is still very fast for sparsely distributed small modifications, so it becomes a very attractive option to solve optimization problems efficiently, due to the very cheap analysis of the modified structure at each iteration of the optimization procedure. The efficiency of the solution of SDPMP is demonstrated for several cases involving passive and active metasurfaces and a reconfigurable antenna. As a practical example, a programmable metasurface reflector is optimized with particle swarm optimization to obtain a prescribed reflection pattern.**

***Index Terms*— Fast direct solver, hierarchical matrix (H-matrix), method of moments (MoM), optimization problem, sparsely distributed partial modification problems (SDPMPs).**

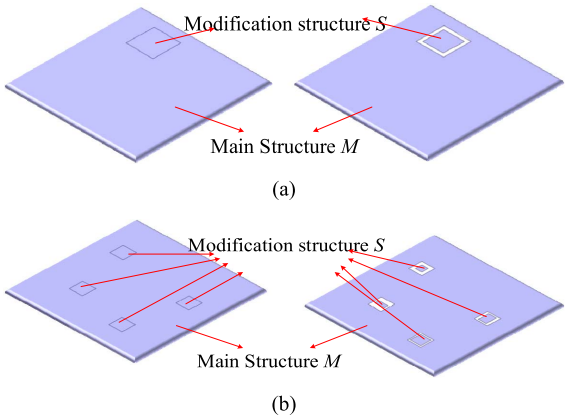

Fig. 1. Representation of the partial geometry modification problem. The modification structure S is much smaller than the main structure M. (a) Traditional PMP. (b) SDPMP, in which the partial modification structure S consists of a set of small modifications.

I. Introduction

COMPUTER aided design of electromagnetic devices using global optimization algorithms has always been a key objective of computational electromagnetics. In such optimization problems, it is often necessary to repeatedly analyze a given geometry including only small modifications introduced to the original object. The numerical analysis of the modified structure at each iteration of the optimizer is called a partial modification problem (PMP) [1].

The PMP is often solved using integral equations discretized by the method of moments (MoM) [2]. Since it is necessary to solve the PMP many times, direct or iterative fast solvers can be used to accelerate the solution time. Examples of fast iterative solvers include the fast multipole-based methods [3], the multilevel matrix decomposition algorithm [4], [5], and multilevel adaptive cross-approximation-based methods [6], [7].

The main drawback of iterative methods in this context, however, is that they need to be applied repeatedly, starting from zero, with every modification of the geometry. Alternatively, the PMP can be addressed by direct solvers, such as the partitioned-inverse formula-based methods [1], [8]–[11]. With these methods, when the problem geometry consists of a large main structure M and a much smaller "modification structure" S to be optimized, as illustrated in Fig. 1(a), only the part of the inverted matrix that represents the modification structure needs to be adapted with every modification. Nevertheless, the fact that the full MoM impedance matrix must be computed and inverted makes this approach inefficient in general.

Manuscript received September 27, 2021; revised January 10, 2022 and April 1, 2022; accepted May 27, 2022. This work was supported in part by the National Nature Science Foundation of China under Grant 61871219; in part by the China Scholarship Council under Grant 201906830042; in part by the Startup Foundation for Introducing Talent of the Nanjing University of Information Science and Technology (NUIST) under Grant 2022r079; in part by the Ministerio de Ciencia e Innovacion (MICINN) under Project PID2019-107885GB-C31/AEI/10.13039/501100011033, Project PID2020-113832RB-C21/AEI/10.13039/501100011033, and Project TEC2017-84817-C2-2-R/AEI/10.13039/501100011033; and in part by the Unidad de Excelencia Maria de Maeztu, which is financed by the Agencia Estatal de Investigación, Spain, under Grant MDM-2016-0600/AEI/10.13039/501100011033; and in part by the Catalan Agency for Management of University and Research Grants (AGAUR) under Research Group under Grant 2017 SGR 219. *(Corresponding author: Xiaoxing Fang.)*

Xiaoxing Fang is with the Research Center of Applied Electromagnetics, School of Electronic and Information Engineering, Nanjing University of Information Science and Technology, Nanjing 210044, China (e-mail: fang_xiaoxing@nuist.edu.cn).

Alexander Heldring and Juan M. Rius are with the CommSensLab, Department of Signal Processing and Telecommunications, Universitat Politècnica de Catalunya (UPC), 08034 Barcelona, Spain (e-mail: heldring@tsc.upc.edu; rius@tsc.upc.edu).

Qunsheng Cao is with the College of Electronic and Information Engineering, Nanjing University of Aeronautics and Astronautics, Nanjing 211106, China (e-mail: qunsheng@nuaa.edu.cn).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TMTT.2022.3182239.

Digital Object Identifier 10.1109/TMTT.2022.3182239

In recent years, several fast direct solvers have been proposed to accelerate the inverse process, such as hierarchical matrix (H-matrix) [12], [13], hierarchical off diagonal low-rank (HODLR) matrix [14], H^2 matrix [15], and hierarchically semiseparable (HSS) matrix [16]-based methods. These solvers can speed up the inverse of both the main structure M and the modification structure S, as long as the off-diagonal blocks are low-rank and therefore compressible.

In this article, the H-matrix format is adopted, and the multiscale compressed block decomposition (MSCBD) [13], [17] is used to accelerate the inverse process. This H-matrix-based fast direct solver is essentially a nested implementation of the partitioned inverse formulas. It can be applied to solve the PMP very efficiently as long as the modification structure is small with respect to the main one and concentrated in a small part of the object.

In H-matrix, the mutual-impedance matrix between M and S is compressed into a low rank matrix. However, unlike the case of Fig. 1(a), the rank of this mutual-impedance matrix will become very high when the modifications are sparsely distributed, as illustrated in Fig. 1(b). If this is the case, it will be very expensive to obtain and store the mutual-impedance matrix and the decomposition matrix. In this article, we propose a modified MSCBD process to address this sparsely distributed PMP (SDPMP) efficiently. After this, we present a general process to analyze structures loaded with sparsely distributed active devices. Finally, this process is combined with particle swarm optimization (PSO) [18] in order to optimize the active device loading of metasurfaces in a highly efficient manner.

This article is organized as follows. Section II first illustrates the PMP and its accelerated solution by the H-matrix. Then, the SDPMP and sparsely distributed active device loaded problem are introduced and solved with the improved H-matrix-based direct solver (MSCBD). Finally, the PSO is proposed in combination with SDPMP-H-matrix to accelerate the optimization problem. Some numerical results in Section III illustrate the performance of this algorithm. Finally, Section IV presents some conclusions.

II. Partial Modification Problems

A. Partitioned Impedance Matrix Equation Formula

Electromagnetic problems modeled with integral equations discretized with the MoM lead to a linear system of the form [2]

$$\mathbf{ZI} = \mathbf{E} \tag{1}$$

where $\mathbf{Z}$, $\mathbf{I}$, and $\mathbf{E}$ are the impedance matrix, the unknown current vector, and the field excitation vector, respectively.

As shown in Fig. 1(a), in PMPs, the object is divided into two parts: main structure M and modification structure S. The impedance matrix equation (1) for the object can be partitioned as follows:

$$\begin{bmatrix} \mathbf{Z}_{MM} & \mathbf{Z}_{MS} \\ \mathbf{Z}_{SM} & \mathbf{Z}_{SS} \end{bmatrix} \begin{bmatrix} \mathbf{I}_M \\ \mathbf{I}_S \end{bmatrix} = \begin{bmatrix} \mathbf{E}_M \\ \mathbf{E}_S \end{bmatrix} \tag{2}$$

where $\mathbf{Z}_{MM}$ and $\mathbf{Z}_{SS}$ are the self-impedance matrices of the main structure M and the modification structure S, respectively. $\mathbf{Z}_{MS}$ and $\mathbf{Z}_{SM}$ are the mutual-impedance matrices between M and S. $\mathbf{E}_M$ and $\mathbf{E}_S$ are the corresponding excitation vectors. $\mathbf{I}_M$ and $\mathbf{I}_S$ are the unknown vectors corresponding to structures M and S.

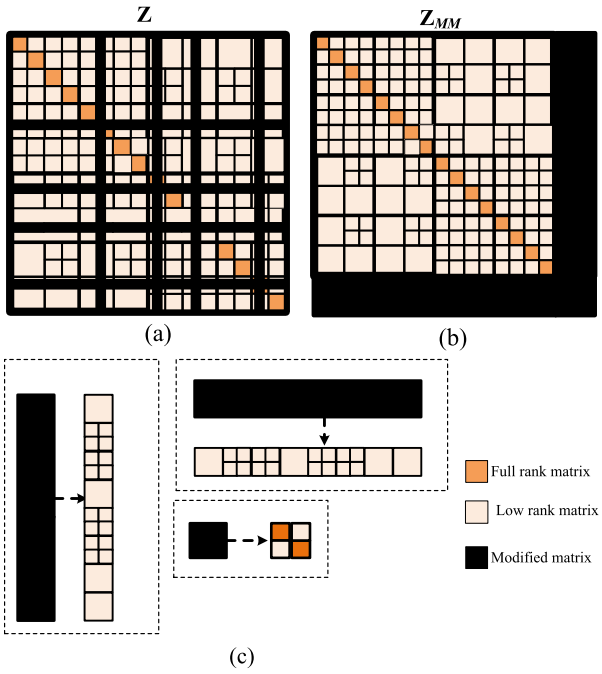

Fig. 2. Impedance matrix representation of SDPMP. (a) Conventional MSCBD. (b) SDPMP-MSCBD. (c) H-matrix type of $\mathbf{Z}_{SS}$, $\mathbf{Z}_{MS}$, and $\mathbf{Z}_{SM}$.

The inverse of $\mathbf{Z}_{MM}$ is calculated only once, since $\mathbf{Z}_{MM}$ does not change when the structure is modified

$$\widetilde{\mathbf{Z}}_{MM} = \mathbf{Z}_{MM}^{-1}. \tag{3}$$

The three, much smaller remaining parts of $\mathbf{Z}$, $\mathbf{Z}_{MS}$, $\mathbf{Z}_{SM}$, and $\mathbf{Z}_{SS}$ are recalculated with every modification of S. Then, the unknowns can be updated by the following equations:

$$\widetilde{\mathbf{Z}}_{SS} = \left(\mathbf{Z}_{SS} - \mathbf{Z}_{SM}\widetilde{\mathbf{Z}}_{MM}\mathbf{Z}_{MS}\right)^{-1} \tag{4}$$

$$\mathbf{I}_S = \widetilde{\mathbf{Z}}_{SS}\left(\mathbf{E}_S - \mathbf{Z}_{SM}\widetilde{\mathbf{Z}}_{MM}\mathbf{E}_M\right) \tag{5}$$

$$\mathbf{I}_M = \widetilde{\mathbf{Z}}_{MM}(\mathbf{E}_M - \mathbf{Z}_{MS}\mathbf{I}_S). \tag{6}$$

B. Accelerated PMP/SDPMP With H-Matrix-Based Direct Solver

Even though the inversion in (3) needs to be done only once, it quickly becomes prohibitively expensive for large problems. Therefore, in this section, we invoke the H-matrix format for the entire problem and compute the inverse of $\mathbf{Z}_{MM}$ using the H-matrix-based direct solvers [12], [13]. Here, the MSCBD [13] is chosen.

In the conventional MSCBD, the impedance matrix is constructed, as shown in Fig. 2(a). In each modification, we need to calculate the impedance matrix $\mathbf{Z}$ and its factorization matrix $\mathbf{B}$ repeatedly. $\mathbf{B}$ is also stored into H-matrix format after the factorization with MSCBD. If MSCBD process is combined with the partitioned impedance matrix in (2), this matrix will be changed into the structure shown in Fig. 2(b)

to solve PMPs. In this case, (3)–(6) can be rewritten as

$$\mathbf{Z}_{MM} \xrightarrow{\text{MSCBD}} \mathbf{B}_{MM} \tag{7}$$
$$\mathbf{I}'_M = \mathbf{B}_{MM}\mathbf{E}_M \tag{8}$$
$$\mathbf{B}_{MS} = \mathbf{B}_{MM}\mathbf{Z}_{MS} \tag{9}$$
$$\mathbf{B}_{SS} = (\mathbf{Z}_{SS} - \mathbf{Z}_{SM}\mathbf{B}_{MS})^{-1} \tag{10}$$
$$\mathbf{I}_S = \mathbf{B}_{SS}(\mathbf{E}_S - \mathbf{Z}_{SM}\mathbf{I}'_M) \tag{11}$$
$$\mathbf{I}_M = \mathbf{I}'_M - \mathbf{B}_{MS}\mathbf{I}_S \tag{12}$$

where $\mathbf{B}_{MM}$, $\mathbf{B}_{MS}$, and $\mathbf{B}_{SS}$ are the compressed block factorization matrices of $\mathbf{Z}_{MM}$, $\mathbf{Z}_{MS}$, and $\mathbf{Z}_{SS}$, respectively.

As the number of unknowns in M is much larger than that in S, the operation in (7) will be the most expensive in solving PMPs, but (7) and (8) need to be calculated only once. Equations (9)–(12) will be computed repeatedly depending on the number of modifications. In the most common case of optimization problems, the modification structure S is not separated well with the main structure M. Furthermore, the modification structure S is not always concentrated in just one block. The modifications in S are sometimes sparsely distributed on the object, as seen in Fig. 1(b). This kind of PMP is called an SDPMP. We can extract these sparsely distributed modified matrices or basis functions together, as shown in Fig. 2(b). With an increasing number of unknowns in S and a poor separation between M and S, $\mathbf{Z}_{SS}$, $\mathbf{Z}_{MS}$, and $\mathbf{Z}_{SM}$ will be exceedingly large. Accordingly, it will be very expensive to compute and save $\mathbf{B}_{SS}$, $\mathbf{B}_{MS}$, and $\mathbf{B}_{SM}$ when (9) and (10) are applied directly.

To solve this problem, the main structure M and modification structure S are subdivided using binary trees into 2^{L_M} and 2^{L_S} blocks, respectively. $\mathbf{Z}_{MM}$ is compressed into an L_M-level H-matrix, as shown in Fig. 2(b). Likewise, the modified matrices $\mathbf{Z}_{SS}$, $\mathbf{Z}_{MS}$, and $\mathbf{Z}_{SM}$ will be compressed into H-matrices according to the L_M and Ls binary tree subdivision, as seen in Fig. 3. Subsequently, the process of SDPMP-MSCBD can be divided into several steps, as shown in Fig. 3.

Step 1: Compress the partitioned impedance matrices $\mathbf{Z}_{MM}$, $\mathbf{Z}_{SS}$, $\mathbf{Z}_{MS}$, and $\mathbf{Z}_{SM}$ into H-matrix after the L_M and Ls binary tree subdivision, which can be seen in Fig. 3(a).

Step 2: As shown in Fig. 3(b), the factorization matrix $\mathbf{B}_{MM}$ is obtained with (7) by using the conventional H-matrix-based direct solver MSCBD. After that, $\mathbf{I}'_M$ can be calculated by (8). In SDPMP, $\mathbf{B}_{MM}$ and $\mathbf{I}'_M$ just need to be calculated once. Since $\mathbf{Z}_{SS}$, $\mathbf{Z}_{MS}$, and $\mathbf{Z}_{SM}$ change with each modification, we need to repeatedly recalculate their factorization matrices.

Step 3: Before factorization of $\mathbf{Z}_{MS}$, $\mathbf{B}_{MM}$ is partitioned into $2^{L_M-L_S}$ submatrices according to the L_M and Ls binary tree subdividing, as shown in Fig. 3(c)

$$\mathbf{B}_{MM} = \left[\, \mathbf{B}_{MM}^{(1)} \cdots \mathbf{B}_{MM}^{(i)} \cdots \mathbf{B}_{MM}^{(2^{L_M-L_S})} \,\right]^T. \tag{13}$$

Step 4: After partitioning of $\mathbf{B}_{MM}$, $\mathbf{B}_{MS}$ obtained by (9) can be rewritten as

$$\begin{aligned}\mathbf{B}_{MS} &= \mathbf{B}_{MM}\mathbf{Z}_{MS} \\ &= \left[\, \mathbf{B}_{MM}^{(1)} \cdots \mathbf{B}_{MM}^{(i)} \cdots \mathbf{B}_{MM}^{(2^{L_M-L_S})} \,\right]^T \mathbf{Z}_{MS}.\end{aligned} \tag{14}$$

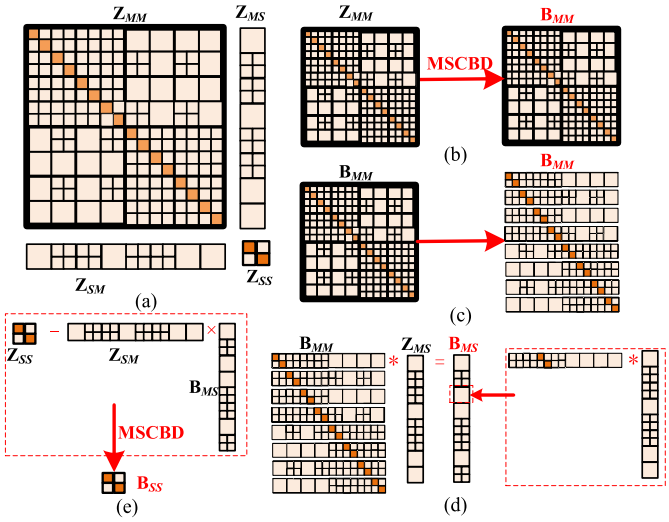

Fig. 3. Several steps of MSCBD-SDPMP. (a) H-matrix compression of $\mathbf{Z}_{MM}$, $\mathbf{Z}_{SS}$, $\mathbf{Z}_{MS}$, and $\mathbf{Z}_{SM}$. (b) Factorization of $\mathbf{Z}_{MM}$ with conventional MSCBD. (c) Factorization matrix $\mathbf{B}_{MM}$ partitioning. (d) Factorization matrix $\mathbf{B}_{MS}$ calculation. (e) Factorization matrix $\mathbf{B}_{SS}$ calculation with conventional MSCBD.

$\mathbf{Z}_{MS}$ is partitioned into $2^{L_M-L_S}$ submatrices

$$\mathbf{Z}_{MS} = \left[\, \mathbf{Z}_{MS}^{(1)} \cdots \mathbf{Z}_{MS}^{(i)} \cdots \mathbf{Z}_{MS}^{(2^{L_M-L_S})} \,\right]^T. \tag{15}$$

The ith submatrix of $\mathbf{B}_{MS}$ can be calculated by

$$\begin{aligned}&\mathbf{B}_{MS}^{(i)} \\ &= \mathbf{B}_{MM}^{(i)}\mathbf{Z}_{MS} \\ &= \mathbf{B}_{MM}^{(i)}\left[\, \mathbf{Z}_{MS}^{(1)} \cdots \mathbf{Z}_{MS}^{(i)} \cdots \mathbf{Z}_{MS}^{(2^{L_M-L_S})} \,\right]^T \\ &= \mathbf{B}_{MM}^{(i,1)}\mathbf{Z}_{MS}^{(1)} + \cdots + \mathbf{B}_{MM}^{(i,i)}\mathbf{Z}_{MS}^{(i)} + \cdots + \mathbf{B}_{MM}^{(i,2^{L_M-L_S})}\mathbf{Z}_{MS}^{(2^{L_M-L_S})} \\ &= \sum_{j=1}^{2^{L_M-L_S}} \mathbf{B}_{MM}^{(i,j)}\mathbf{Z}_{MS}^{(j)}\end{aligned} \tag{16}$$

where $\mathbf{B}_{MM}^{(i,j)}$ is the jth submatrix of $\mathbf{B}_{MM}^{(i)}$. After the calculation of each submatrix, $\mathbf{B}_{MS}$ follows as:

$$\mathbf{B}_{MS} = \left[\, \mathbf{B}_{MS}^{(1)} \cdots \mathbf{B}_{MS}^{(i)} \cdots \mathbf{B}_{MS}^{(2^{L_M-L_S})} \,\right]^T. \tag{17}$$

Step 5: $\mathbf{B}_{SS}$ is calculated with $\mathbf{Z}_{SS}$, $\mathbf{B}_{MS}$, and $\mathbf{Z}_{SM}$ by conventional MSCBD

$$\mathbf{Z}_{SS} - \mathbf{Z}_{SM}\mathbf{B}_{MS} \xrightarrow{\text{MSCBD}} \mathbf{B}_{SS}. \tag{18}$$

Step 6: Finally, (11) and (12) are applied to calculate the unknown vectors for the different modifications. The difference is that (11) should be replaced by

$$\mathbf{I}_S = \mathbf{B}_{SS}(\mathbf{E}_S - \mathbf{Z}_{SM}\mathbf{I}'_M). \tag{19}$$

In the steps above, the low rank characteristics of the submatrices are conserved by using the H-matrix additions and multiplications. If we consider n different configurations of the modification structure S, $\mathbf{B}_{MS}$ and $\mathbf{B}_{SS}$ will be calculated for each configuration. The unknown current vector of each configuration can be obtained by the following function.

TABLE I
COMPUTATIONAL COMPLEXITY COMPARISON OF MSCBD AND SDPMP-MSCBD

n different modifications			
Steps	SDPMP-MSCBD	Steps	MSCBD
$\mathbf{B}_{MM}$= *MSCBD* ($\mathbf{Z}_{MM}$)	$O(N_M^2)$	**B**= *MSCBD* (**Z**)	$n\times O(N^2)$
$\mathbf{I'}_M$=$\mathbf{B}_{MM}$ $\mathbf{E}_M$	$O(N_M^{1.5})$		
$\mathbf{B}_{MS}$=$\mathbf{B}_{MM}$ $\mathbf{Z}_{MS}$	$n\times O(N_S^{0.5}N_M^{1.5})$		
$\mathbf{B}_{SS}$= *MSCBD* ($\mathbf{Z}_{SS}$-$\mathbf{Z}_{SM}\mathbf{B}_{MS}$)	$n\times O(N_SN_M+N_S^2)$		
$\mathbf{I}^{(i)}{}_S$=$\mathbf{B}_{SS}$ ($\mathbf{E}s$-$\mathbf{Z}_{SM}\mathbf{I'}_M$)	$n\times O(N_S^{0.5}N_M+N_S^{1.5})$	**I**$^{(i)}$=**BE**	$n\times O(N^{1.5})$
$\mathbf{I}^{(i)}{}_M$=$\mathbf{I'}_M$-$\mathbf{B}_{MS}\mathbf{I}^{(i)}{}_S$	$n\times O(N_S^{0.5}N_M)$		
total	$O(N^2)$ +$(n$-1)$\times$ $O(N^{1.5})$	**total**	$n\times O(N^2)$

Function $\mathbf{I} = SDPMP\ (\mathbf{Z}_{MM}, \mathbf{E}_M, \mathbf{E}s)$
$\mathbf{B}_{MM} = MSCBD\ (\mathbf{Z}_{MM})$;
$\mathbf{I}'_M = \mathbf{B}_{MM}\ \mathbf{E}_M$;
for $i = 1{:}n$
  H-matrix compression: $\mathbf{Z}_{MS}$, $\mathbf{Z}_{SM}$, and $\mathbf{Z}_{SS}$;
  $\mathbf{B}_{MS} = \mathbf{B}_{MM}\ \mathbf{Z}_{MS}$;
  $\mathbf{B}_{SS} = MSCBD\ (\mathbf{Z}_{SS} - \mathbf{Z}_{SM}\mathbf{B}_{MS})$
  $\mathbf{I}_S^{(i)} = \mathbf{B}_{SS}\ (\mathbf{E}s - \mathbf{Z}_{SM}\mathbf{I}'_M)$
  $\mathbf{I}_M^{(i)} = \mathbf{I}'_M - \mathbf{B}_{MS}\mathbf{I}_S^{(i)}$
  $\mathbf{I}^{(i)} = [\mathbf{I}_M^{(i)}; \mathbf{I}_S^{(i)}]$
end
End

The output $\mathbf{I}$ of the function SDPMP contains the vectors $\mathbf{I}^{(1)}, \mathbf{I}^{(2)}, \ldots, \mathbf{I}^{(i)}, \ldots, \mathbf{I}^{(n)}$ for the n different modifications.

Let N, N_M, and N_S be the number of the unknowns in the whole structure, the main structure (M), and the modification structure (S), respectively. Since the modification part is small, $N_S \ll N_M$, $N_M = N - N_S \approx N$. The computational complexity of SDPMP-MSCBD for each step is listed in Table I and compared with that of conventional MSCBD.

The complexity of the MSCBD algorithm is $O(N^2)$, as shown in [13] and [19]. With SDPMP-MSCBD, the computational complexity for n different configurations is $O(N^2) + (n-1) \times O(N^{1.5})$, where the first term $O(N^2)$ is the initial solution of the whole problem and $O(N^{1.5})$ is the complexity of the $(n-1)$ additional solutions of the modified problems. For large n, the total operation count of SDPMP-MSCBD is much less than the $n \times O(N^2)$ operations required by repeating the whole analysis with MSCBD.

C. Sparsely Distributed Active Device Loaded Problems

A type of problem of practical interest for which the proposed method allows a further radical simplification and accelerated solution is that of active device loaded structures. Active metamaterials are a classic example of structures loaded with sparsely distributed active devices. The active metamaterial shown in Fig. 4 consists of a large number of unit cells, each of them including a perfect electric conductor (PEC) structure and an active device. The most used active devices

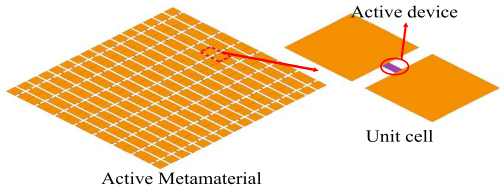

Fig. 4. Active metamaterial model and its unit cell.

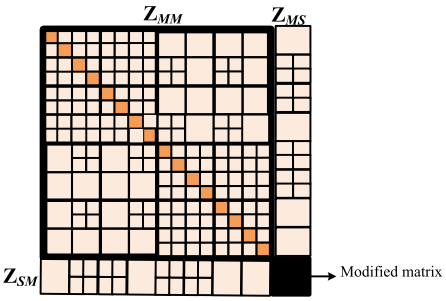

Fig. 5. Impedance matrix representation of sparsely distributed active device loaded problem.

are p-i-n diodes and varactor diodes. Since the impedance of these devices depends on the bias voltage applied to them, they are modeled as finite-width lumped loads [20].

Here, the PEC structures form the main structure M and the active devices are the modification structure S. Because in this case shown in Fig. 5, the only modification takes places in the self-impedance elements on the diagonal of $\mathbf{Z}_{SS}$, not in the mutual impedance blocks $\mathbf{Z}_{MS}$ and $\mathbf{Z}_{SM}$, we only need to calculate $\mathbf{B}_{ss}$ repeatedly. That means in (7)–(12), only (10)–(12) need to be recomputed for each modification and some intermediary matrices and vectors can be precomputed

$$\mathbf{Z}'_{SS} = \mathbf{Z}_{SM}\mathbf{B}_{MS} \tag{20}$$
$$\mathbf{I}'_S = \mathbf{E}_S - \mathbf{Z}_{SM}\mathbf{I}'_M. \tag{21}$$

This leads to a simplified version of the *SDPMP* function for problems involving only active loads.

Function $\mathbf{I} = SDPMP\_ActiveLoad\ (\mathbf{Z}_{MM}, \mathbf{Z}_{MS}, \mathbf{Z}_{SM}, \mathbf{E}_M, \mathbf{E}s)$
$\mathbf{B}_{MM} = MSCBD\ (\mathbf{Z}_{MM})$;
$\mathbf{I}'_M = \mathbf{B}_{MM}\ \mathbf{E}_M$;
$\mathbf{B}_{MS} = \mathbf{B}_{MM}\ \mathbf{Z}_{MS}$;
$\mathbf{Z}'_{SS} = \mathbf{Z}_{SM}\mathbf{B}_{MS}$;
$\mathbf{I}'_S = \mathbf{E}s - \mathbf{Z}_{SM}\mathbf{I}'_M$;
H-matrix compression of $\mathbf{Z}_{SS}$ without added impedance;
for $i = 1{:}n$
  add the impedance of active devices to update $\mathbf{Z}_{SS}$;
  $\mathbf{B}_{SS} = MSCBD\ (\mathbf{Z}_{SS} - \mathbf{Z}'_{SS})$;
  $\mathbf{I}_S^{(i)} = \mathbf{B}_{SS}\mathbf{I}'_S$;
  $\mathbf{I}_M^{(i)} = \mathbf{I}'_M - \mathbf{B}_{MS}\mathbf{I}_S^{(i)}$
  $\mathbf{I}^{(i)} = [\mathbf{I}_M^{(i)}; \mathbf{I}_S^{(i)}]$
end
End

The complexity for n different modifications using the approach proposed here is reduced from $n \times O(N^2)$ to

TABLE II

COMPUTATIONAL COMPLEXITY COMPARISON OF MSCBD AND SDPMP-MSCBD FOR ACTIVE DEVICE LOADED PROBLEMS

n different modifications			
Steps	SDPMP-MSCBD	Steps	MSCBD
$\mathbf{B}_{MM}$= *MSCBD* ($\mathbf{Z}_{MM}$)	$O(N_M^2)$		
$\mathbf{I'}_M$=$\mathbf{B}_{MM}$ $\mathbf{E}_M$	$O(N_M^{1.5})$	**B**= *MSCBD* (**Z**)	$n\times O(N^2)$
$\mathbf{B}_{MS}$=$\mathbf{B}_{MM}$ $\mathbf{Z}_{MS}$	$O(N_S^{0.5}N_M^{1.5})$		
$\mathbf{Z'}_{SS}$=$\mathbf{Z}_{SM}\mathbf{B}_{MS;}$	$O(N_S N_M)$		
$\mathbf{I'}_S$=$\mathbf{E}s$-$\mathbf{Z}_{SM}\mathbf{I'}_M$	$O(N_S^{0.5}N_M)$		
$\mathbf{B}_{SS}$= *MSCBD* ($\mathbf{Z}_{SS}$-$\mathbf{Z'}_{SS}$)	$n\times O(N_S^2)$	$\mathbf{I}^{(i)}$=**BE**	$n\times O(N^{1.5})$
$\mathbf{I}^{(i)}{}_S$=$\mathbf{B}_{SS}\mathbf{I'}_S$	$n\times O(N_S^{1.5})$		
$\mathbf{I}^{(i)}{}_M$=$\mathbf{I'}_M$-$\mathbf{B}_{MS}\mathbf{I}^{(i)}{}_S$	$n\times O(N_S^{0.5}N_M)$		
total	$O(N^2)+(n-1)\times O(N)$	**total**	$n\times O(N^2)$

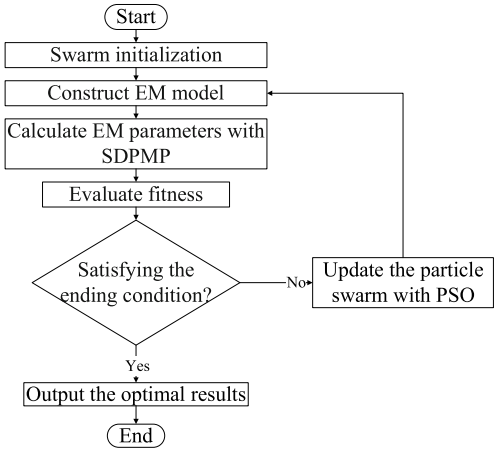

Fig. 6. Flowchart of the optimization with SDPMP-MSCBD and PSO.

$O(N^2) + (n-1) \times O(N)$. Table II shows the complexity of the different steps of the algorithm.

D. Optimization With SDPMP-MSCBD and PSO

The SDPMP-MSCBD algorithm described in Sections II-B and II-C can be combined with a stochastic optimization technique, such as genetic algorithms, simulated annealing, or PSO, to solve many electromagnetic optimization problems efficiently, such as the design of reconfigurable antennas and frequency selective surfaces (FSSs).

To obtain the results presented in this article, a fitness function is defined to optimize the electromagnetic parameters, the SDPMP-MSCBD is used to efficiently solve the SDPMP and evaluate the fitness function, and the PSO algorithm to search a large region in the solution space, as shown in the flowchart of Fig. 6.

III. NUMERICAL RESULTS

All the numerical experiments are performed on a PC with 64 GB of RAM and Intel[1] Xeon[1] CPU E5-2620 V4 processor at 2.10 GHz. The relative error in our radar cross section (RCS) computations is defined as

$$\tau = \frac{|\sigma - \sigma_{\text{ref}}|}{|\sigma_{\text{ref}}|} \tag{22}$$

where σ and σ_{ref} are the bistatic RCS in m^2 calculated by the proposed method and a reference method, respectively.

[1]Registered Trademark.

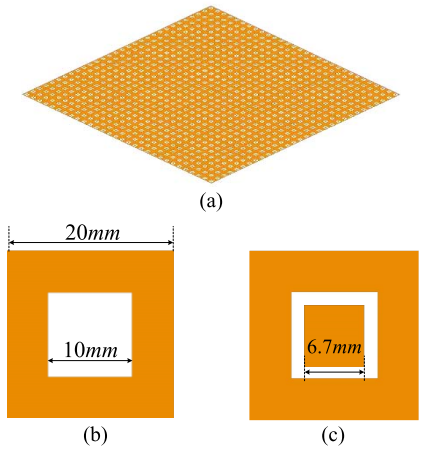

Fig. 7. FSS model. (a) Finite FSS consisting of 30 × 30 cells. (b) Unit cell of original model. (c) Unit cell of modified model.

A. Frequency Selective Surface

FSSs are another kind of metamaterial. First, we consider the FSS shown in Fig. 7 as an example. This FSS consists of 30 × 30 cells, and each unit cell has a size of 200 mm × 200 mm. It is located in the xy plane and centered at the origin. The main structure of the unit cell is shown in Fig. 7(b). The modified model has an additional structure at the center, as shown in Fig. 7(c). The incident x-polarized plane wave at 11.5 GHz impinges normally in the direction $(\theta, \varphi) = (0°, 0°)$. The original model and the modified one are discretized into 82 800 and 90 000 Rao- Wilton- Glisson (RWG) basis functions, respectively. That means there are seven 200 RWG basis functions added in the modifications.

In order to assess both the accuracy and the efficiency of the proposed SDPMP-MSCBD method, we compare the computed RCS with that of a straightforward MSCBD implementation that solves the original structure and the modified structure independently. The RCS results are shown in Fig. 8. The results are practically identical between the two approaches, demonstrating that the proposed approach does not deteriorate the accuracy.

The computation times and memory requirements are summarized in Table III. For the original structure, the two approaches are identical. However, by taking advantage of the work done for the original model, the proposed method reduces the computation time for the modified model by a factor 3.5. The proposed method does need some extra memory. This is inevitable when the modification structure is as sparsely distributed as in this example: all 900 unit cells are modified.

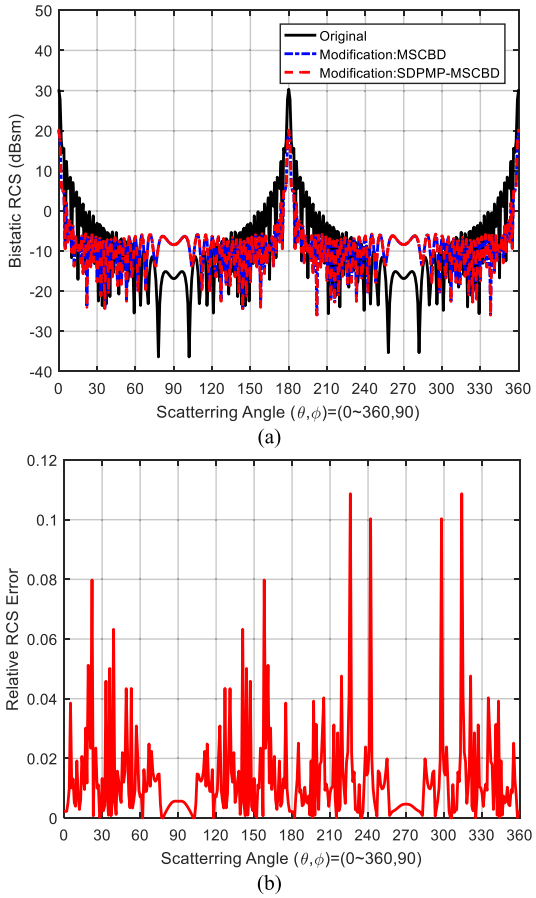

Fig. 8. Bistatic RCS of the FSS of Section III-A. (a) RCS comparison. (b) Relative RCS error.

TABLE III

TIME AND MEMORY COST COMPARISON OF MSCBD AND SDPMP-MSCBD

Original Model	MSCBD			
	Time(s)		Memory(MB)	
Z	529.3		3 890.0	
B	576.4		3 942.5	
solution	0.7		-	
Modification Model	MSCBD		SDPMP -MSCBD	
	Time (s)	Memory (MB)	Time (s)	Memory (MB)
Z	708.0	4428.4	134.5	1 800.5
B	743.9	4635.5	287.3	2 315.9
solution	0.9	-	0.4	-
Modification total	**1 452.8**	**4 635.5**	**413.2**	**6 258.4**

B. Active Frequency Selective Surface

In Fig. 9, an active FSS (AFSS) model [21] is taken as the example. This AFSS has 20 × 20 cells, and it is discretized into 56 480 RWG basis functions: 55 680 RWGs for the main structure and 800 for the active devices (two p-i-n diodes per cell). When the p-i-n diodes are switched ON, their resistance is about 2 Ω. When, on the contrary, no bias is provided to the p-i-n diodes, the configuration switches to a high resistance (2 kΩ). As in the previous example, there is an incident x-polarized plane wave at 3.4 GHz propagating in the direction $(\theta, \varphi) = (0°, 0°)$.

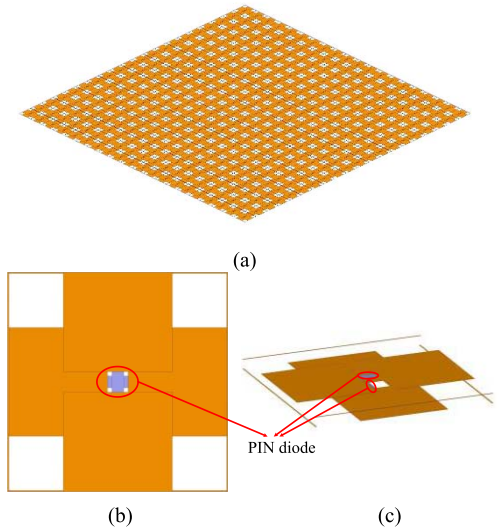

Fig. 9. AFSS model (similar to the model in [21]). (a) 20 × 20 finite AFSS. (b) Top view of its unit cell. (c) 3-D view of the unit cell.

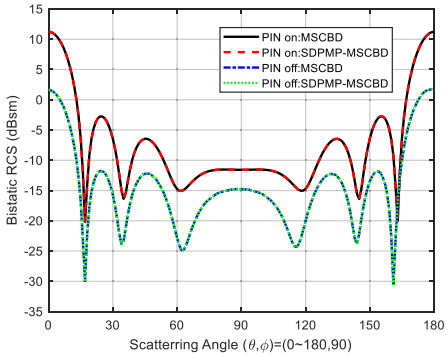

Fig. 10. Bistatic RCSs of the AFSS.

TABLE IV

TIME COST COMPARISON OF MSCBD AND SDPMP-MSCBD IN THE AFSS EXAMPLE

Time	PIN ON	PIN OFF
MSCBD	290.3 s	289.8 s
SDPMP-MSCBD	317.7 s	0.03 s

Fig. 10 shows the bistatic RCS of the AFSS when all the p-i-n diodes are ON and when they are OFF. The results of our method compare well with the MSCBD. The computation times are shown in Table IV. Our method reduces the time for the modified (p-i-n OFF) state by a factor of 9660 after the calculation of the p-i-n ON-state.

Subsequently, we study the effect on the RCS of the AFSS when several p-i-n diodes are damaged. The results when 20%, 40%, 60%, and 80% of the p-i-n diodes at random positions are damaged are shown in Fig. 11(a)–(d), respectively. A damage of 20% means, for instance, that 160 (800 × 20%) randomly chosen p-i-n diodes are modeled with a high resistance (2 kΩ) regardless if bias voltage is provided to them or not. In each figure of Fig. 11, the results are shown for ten different random configurations of the damaged p-i-n diodes. With our method,

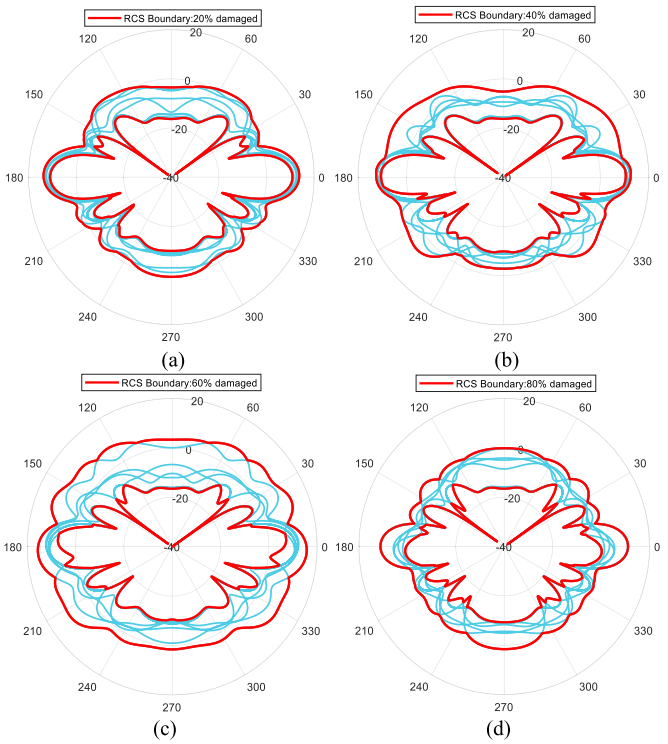

Fig. 11. Bistatic RCS of the AFSS with the p-i-n diodes damaged. (a) 20% p-i-n diodes damaged. (b) 40% p-i-n diodes damaged. (c) 60% p-i-n diodes damaged. (d) 80% p-i-n diodes damaged. The blue curves show the RCS with different random configurations of damaged p-i-n diodes. The red curves show their boundary.

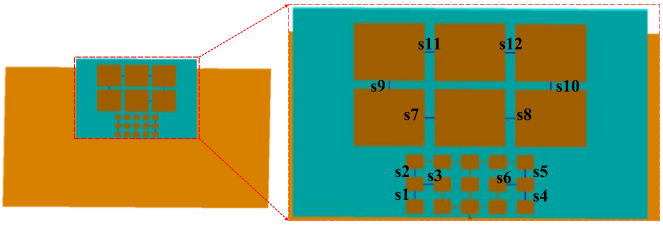

Fig. 12. Model of reconfigurable antenna. The details of the geometry can be found in [22].

the entire calculation takes only 318.9 s. With the conventional MSCBD, it takes 3.2 h.

C. Reconfigurable Antenna

The reconfigurable antenna designed and measured in [22] has been analyzed with SDPMP-MSCBD. It consists of a set of switchable patches, as shown in Fig. 12. The patches feeding is controlled by 12 p-i-n diodes for radiation pattern reconfiguration. The switch ON-state of each p-i-n diode is modeled as a serial resistance ($R_{\text{ON}} = 3.5\ \Omega$) and the OFF-state is modeled as an R–C shunt circuit ($R_{\text{OFF}} = 2.6\ \text{k}\Omega$ and $C_{\text{OFF}} = 0.17$ pF). The operating frequency is 2.45 GHz.

The antenna has been modeled with the volume integral equation (VIE)–surface integral equation (SIE) formulation [23]. The patches' surface is discretized into 3346 RWG and the substrate volume into 9868 Schaubert-Wilton-Glisson (SWG) basis functions. The active p-i-n diodes have 24 RWG basis functions, for a total of 3370 RWG.

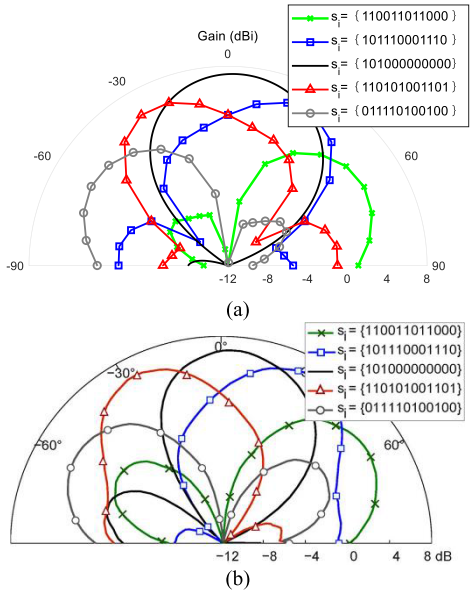

Fig. 13. Gain pattern of the configurations with maximum gain at angular directions $\theta = -60°$, $-30°$, $0°$, $30°$, and $60°$. (a) Simulated by SDPMP-MSCBD. (b) Measured results from [22].

TABLE V
TIME COST OF MSCBD AND SDPMP-MSCBD FOR DIFFERENT CONFIGURATIONS

Time	angular direction θ of maximum gain				
	0	-60°	-30°	30°	60°
MSCBD	88 s	91 s	90 s	91 s	87 s
SDPMP-MSCBD	107 s	0.02 s	0.02 s	0.01 s	0.02 s

Fig. 13 compares the simulated and measured radiation pattern gain for different configurations, in which the direction of the main lobe is $\theta_{\text{max}} = -60°$, $-30°$, $0°$, $30°$, and $60°$. The simulated results have been obtained with the SDPMP-MSCBD solver. The different patch configurations are coded as $[s_1, s_2, \ldots, s_{12}]$, where 1's and 0's, respectively, represent the diode ON- and OFF-states.

Simulated and measured results corresponding to the same configuration code have the same pattern maximum direction and gain. There are minor differences due to the usual discrepancies between the numerical model and the measured setup.

The computation time of the SDPMP-MSCBD and the conventional MSCBD for different configurations is shown in Table V. SDPMP-MSCBD takes about 20% longer for the analysis of the first configuration ($\theta_{\text{max}} = 0°$) but is extremely fast for the remaining ones, while MSCBD must repeat the whole computation.

D. Programmable Metasurface

As a final example, we consider a similar programmable metasurface from [24], shown in Fig. 14. It contains 16 × 16 cells, and it is discretized into 127 964 RWG basis functions: 127 708 for the main structure and 256 for the p-i-n diodes. The p-i-n diodes used here are the same as those in Section III-B. Fig. 14(b) and (c) shows the structure of a unit cell. Three kinds of programmed states are considered,

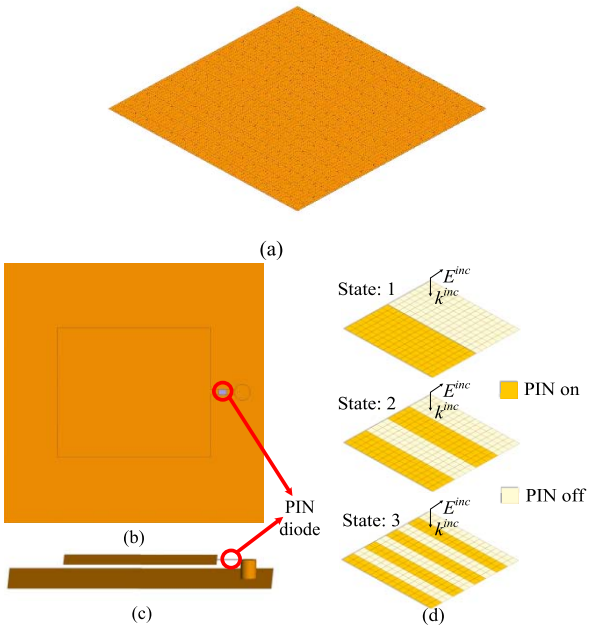

Fig. 14. Programmable metasurface model (similar to the model in [24]). (a) 16 × 16 finite programmable metasurface. (b) Top view of its unit cell. (c) 3-D view of the unit cell. (d) Three kinds of programmed states.

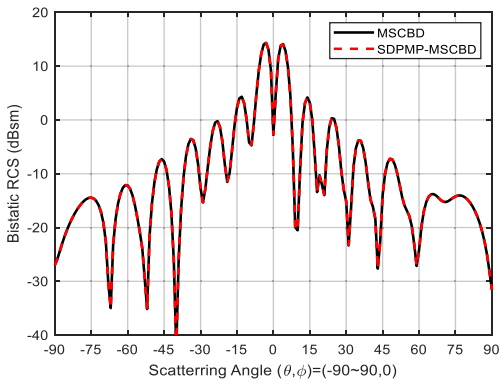

Fig. 15. Bistatic RCS comparison of the MSCBD and SDPMP-MSCBD, for the diodes in state 1.

as shown in Fig. 14(d). All of them are illuminated by an impinging x-polarized plane wave at 23 GHz with an incident direction of $(\theta, \varphi) = (0°, 0°)$. State 1 in Fig. 14(d) is chosen as the example to compare the results of MSCBD and SDPMP-MSCBD. The bistatic RCS, shown in Fig. 15, corresponds very well between the two approaches. Table VI summarizes the computation times for all three states. Although the proposed method costs a bit more time for state 1, the subsequent computation times for the other states are practically negligible. Fig. 16 shows how the reflected beam angles of the programmable metasurface are controlled by the different states.

Next, we consider the optimization of the reflected beam angle. The aim is to obtain the diode states that approximate as best as possible the desired reflected beam shown in Fig. 17 (black line). The fitness function of this optimization problem

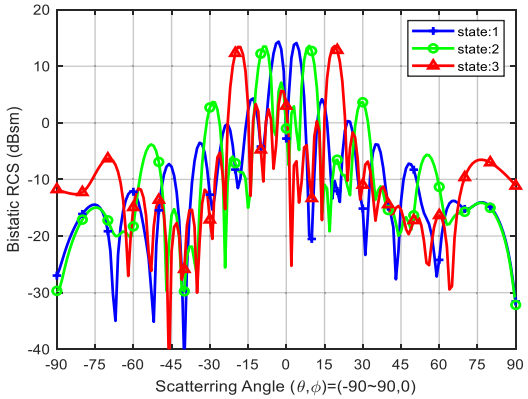

Fig. 16. Bistatic RCSs of the three states in Fig. 14(d).

TABLE VI
TIME COST COMPARISON OF MSCBD AND SDPMP-MSCBD

Time	state 1	state 2	state 3	total
MSCBD	29.2 min	28.9 min	29.6 min	87.7 min
SDPMP-MSCBD	33.3 min	0.04 s	0.03 s	33.3 min

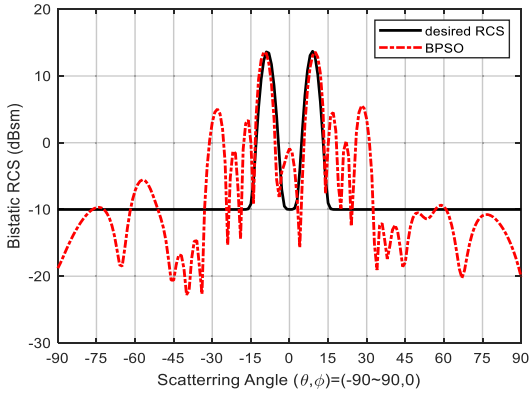

Fig. 17. Bistatic RCS optimized with the BPSO algorithm (red dashed line) compared to the desired RCS (black line).

is defined as

$$\text{Fitness} = \sum_{i=1}^{N_{\text{sample}}} |\sigma(i) - \sigma_{\text{des}}(i)| \tag{23}$$

where σ is the calculated bistatic RCS and σ_{des} are the desired RCS. N_{sample} is the total number of scattering angle samples, set to 181 in our example.

We chose the binary tree PSO (BPSO) algorithm [18] combined with our SDPMP-MSCBD method. Its learning factors are all set to 1.2. The inertia weight is 0.8 and the number of initial swarm particles is 10.

The optimized RCS is shown in Fig. 17 (red dashed line). The goal of the desired RCS is to obtain two beams toward $-10°$ and $10°$. It can be found that the optimized RCS has realized the goal of the desired RCS. The fitness function of the BPSO calculated with (23) and the final programmable metasurface state are shown in Fig. 18. The first iteration step costs about 33.3 min. The total time for the next 49 iteration

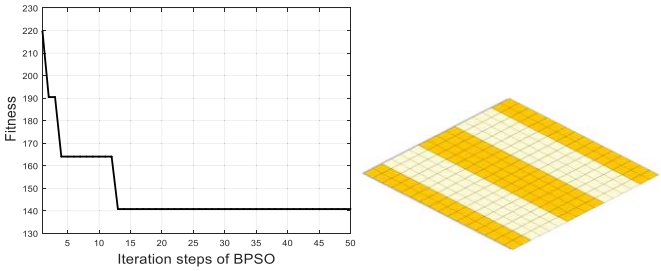

Fig. 18. (a) Evolution of the BPSO fitness function and (b) final state of the programmable metamaterial.

steps is 428.7 s. Every step takes only 8.75 s, while they would cost about 29.0 min with the conventional MSCBD.

IV. Conclusion

In this article, we have introduced the SDPMP and we have improved the H-matrix-based direct solver MSCBD to solve the SDPMP in a very short time for each new set of modifications. The improved version of MSCBD presented here is able to efficiently solve the problem when there is a large set of sparsely distributed small modifications.

The numerical results that we have obtained for active and passive metasurfaces and a reconfigurable antenna show that the combination of the SDPMP and the improved MSCBD leads to a great improvement in computation time, especially for metasurfaces loaded with sparsely distributed active devices.

We have also shown that the application of SDPMP-MSCBD with a global optimization algorithm like PSO becomes a very powerful tool for the design of reconfigurable reflective metasurfaces, improving the computation time by a factor 200 compared with standard MSCBD. Similar results are expected for other interesting electromagnetic optimization problems, such as the design of reconfigurable antennas.

References

[1] X. Chen, C. Gu, Y. Zhang, and R. Mittra, "Analysis of partial geometry modification problems using the partitioned-inverse formula and Sherman–Morrison–Woodbury formula-based method," *IEEE Trans. Antennas Propag.*, vol. 66, no. 10, pp. 5425–5431, Oct. 2018.

[2] R. F. Harrington, *Field Computation by Moment Methods*. New York, NY, USA: Macmillan, 1968.

[3] J. M. Song, C.-C. Lu, and W. C. Chew, "Multilevel fast multipole algorithm for electromagnetic scattering by large complex objects," *IEEE Trans. Antennas Propag.*, vol. 45, no. 10, pp. 1488–1493, Oct. 1997.

[4] E. Michielssen and A. Boag, "A multilevel matrix decomposition algorithm for analyzing scattering from large structures," *IEEE Trans. Antennas Propag.*, vol. 44, no. 8, pp. 1086–1093, Aug. 1996.

[5] J. M. Rius, J. Parrón, A. Heldring, J. M. Tamayo, and E. Ubeda, "Fast iterative solution of integral equations with method of moments and matrix decomposition algorithm–singular value decomposition," *IEEE Trans. Antennas Propag.*, vol. 56, no. 8, pp. 2314–2324, Aug. 2008.

[6] J. M. Tamayo, A. Heldring, and J. M. Rius, "Multilevel adaptive cross approximation (MLACA)," *IEEE Trans. Antennas Propag.*, vol. 59, no. 12, pp. 4600–4608, Dec. 2011.

[7] X. Chen, C. Gu, J. Ding, Z. Li, and Z. Niu, "Multilevel fast adaptive cross-approximation algorithm with characteristic basis functions," *IEEE Trans. Antennas Propag.*, vol. 63, no. 9, pp. 3994–4002, Sep. 2015.

[8] W. W. Hager, "Updating the inverse of a matrix," *SIAM Rev.*, vol. 31, no. 2, pp. 221–239, Jun. 1989.

[9] P. Parhami, Y. Rahmat-Samii, and R. Mittra, "Technique for calculating the radiation and scattering characteristics of antennas mounted on a finite ground plane," *Proc. Inst. Elect. Eng.*, vol. 124, no. 11, pp. 1009–1016, Nov. 1977.

[10] J. M. Johnson and Y. Rahmat-Samii, "Genetic algorithms and method of moments (GA/MOM) for the design of integrated antennas," *IEEE Trans. Antennas Propag.*, vol. 47, no. 10, pp. 1606–1614, Oct. 1999.

[11] J. Laviada, J. Gutierrez-Meana, M. R. Pino, and F. Las-Heras, "Analysis of partial modifications on electrically large bodies via characteristic basis functions," *IEEE Antennas Wireless Propag. Lett.*, vol. 9, pp. 834–837, 2010.

[12] H. Guo, Y. Liu, J. Hu, and E. Michielssen, "A butterfly-based direct integral-equation solver using hierarchical LU factorization for analyzing scattering from electrically large conducting objects," *IEEE Trans. Antennas Propag.*, vol. 65, no. 9, pp. 4742–4750, Sep. 2017.

[13] A. Heldring, J. M. Rius, J. M. Tamayo, J. Parrón, and E. Ubeda, "Multiscale compressed block decomposition for fast direct solution of method of moments linear system," *IEEE Trans. Antennas Propag.*, vol. 59, no. 2, pp. 526–536, Feb. 2011.

[14] X. Fang, Q. Cao, Y. Zhou, and Y. Wang, "Multiscale compressed and spliced Sherman–Morrison–Woodbury algorithm with characteristic basis function method," *IEEE Trans. Electromagn. Compat.*, vol. 60, no. 3, pp. 716–724, Jun. 2018.

[15] M. Ma and D. Jiao, "Accuracy controlled structure-preserving H^2-matrix-matrix product in linear complexity with change of cluster bases," *IEEE Trans. Microw. Theory Techn.*, vol. 68, no. 2, pp. 441–455, Feb. 2020.

[16] K. L. Ho and L. Greengard, "A fast direct solver for structured linear systems by recursive skeletonization," *SIAM J. Sci. Comput.*, vol. 34, no. 5, pp. 2507–2532, 2012.

[17] X. Fang, Q. Cao, Y. Zhou, and Y. Wang, "Multiscale compressed block decomposition method with characteristic basis function method and fast adaptive cross approximation," *IEEE Trans. Electromagn. Compat.*, vol. 61, no. 1, pp. 191–199, Feb. 2019.

[18] L. Zhang *et al.*, "Space-time-coding digital metasurfaces," *Nature Commun.*, vol. 9, no. 1, p. 4334, 2018.

[19] A. Heldring, J. M. Tamayo, E. Ubeda, and J. M. Rius, "Accelerated direct solution of the method-of-moments linear system," *Proc. IEEE*, vol. 101, no. 2, pp. 364–371, Feb. 2013, doi: 10.1109/JPROC.2012.2193369.

[20] Y. H. Lo, L. J. Jiang, and W. C. Chew, "Finite-width feed and load models," *IEEE Trans. Antennas Propag.*, vol. 61, no. 1, pp. 281–289, Jan. 2013.

[21] H. Li, Q. Cao, and Y. Wang, "A novel 2-B multifunctional active frequency selective surface for LTE-2.1 GHz," *IEEE Trans. Antennas Propag.*, vol. 65, no. 6, pp. 3084–3092, Jun. 2017.

[22] D. Rodrigo and L. Jofre, "Frequency and radiation pattern reconfigurability of a multi-size pixel antenna," *IEEE Trans. Antennas Propag.*, vol. 60, no. 5, pp. 2219–2225, May 2012.

[23] S. N. Makarov, S. D. Kulkarni, A. G. Marut, and L. C. Kempel, "Method of moments solution for a printed patch/slot antenna on a thin finite dielectric substrate using the volume integral equation," *IEEE Trans. Antennas Propag.*, vol. 54, no. 4, pp. 1174–1184, Apr. 2006.

[24] H. Yang *et al.*, "A programmable metasurface with dynamic polarization, scattering and focusing control," *Sci. Rep.*, vol. 6, no. 1, p. 35692, Dec. 2016.

Xiaoxing Fang (Member, IEEE) received the B.S. degree in electrical engineering from the Nanjing University of Information Science and Technology, Nanjing, China, in 2013, the M.E. degree in control theory and control engineering from the Jiangsu University of Science and Technology, Zhenjiang, China, in 2016, and the Ph.D. degree in communication and information system from the Nanjing University of Aeronautics and Astronautics (NUAA), Nanjing, in 2021.

From 2019 to 2020, he is a Visiting Research Fellow with the Universitat Politècnica de Catalunya (UPC), Barcelona, Spain. He is currently working as a Lecturer with the School of Electronic and Information Engineering, Nanjing University of Information Science and Technology. His research interest includes accelerated numerical methods for solving electromagnetic problems.

Alexander Heldring was born in Amsterdam, The Netherlands, in 1966. He received the M.S. degree in applied physics and the Ph.D. degree in electrical engineering from the Delft University of Technology, Delft, The Netherlands, in 1993 and 2002, respectively.

He is currently working as an Associate Professor with the Telecommunications Department, Universitat Politècnica de Catalunya (UPC), Barcelona, Spain. He has authored or coauthored more than 30 articles in international journals and 100 in international conference proceedings. His special research interest includes fast integral equation methods for electromagnetic problems.

Juan M. Rius (Senior Member, IEEE) was born in Barcelona, Spain, in 1963. He received the Ingeniero de Telecomunicación and Doctor Ingeniero degrees from Universitat Politècnica de Catalunya (UPC), Barcelona, in 1987 and 1991, respectively.

In 1985, he joined the Electromagnetic and Photonic Engineering Group, UPC, now the CommSensLab, where he currently holds a position of "Catedrático" (equivalent to a Full Professor). From 1985 to 1988, he developed a new inverse scattering algorithm for microwave tomography in cylindrical geometry systems. Since 1989, he has been engaged in the research for new and efficient methods for numerical computation of electromagnetic scattering and radiation. He was a Visiting Professor with EPFL, Lausanne, Switzerland, from May 1, 1996 to October 31, 1996; a Visiting Fellow with the City University of Hong Kong, Hong Kong, from January 3, 1997 to February 4, 1997; a CLUSTER Chair with EPFL, from December 1, 1997 to January 31, 1998; and a Visiting Professor with EPFL, from April 1, 2001 to June 30, 2001. He is the developer of the graphical electromagnetic computation (GRECO) approach for high-frequency radar cross section (RCS) computation, the integral equation formulation of the measured equation of invariance (IEMEI), and the multilevel matrix decomposition algorithm (MLMDA) in 3-D. He has authored or coauthored more than 70 articles published or accepted in refereed international journals (40 in IEEE TRANSACTIONS) and more than 200 in international conference proceedings.

His current interests are the numerical simulation of electrically large antennas and scatterers.

Qunsheng Cao received the Ph.D. degree in electrical engineering from The Hong Kong Polytechnic University, Hong Kong, in 2000.

From 2000 to 2005, he worked as a Research Associate with the Department of Electrical Engineering, University of Illinois at Urbana-Champaign, Champaign, IL, USA, and the Army High Performance Computing Research Center (AHPCRC), University of Minnesota, Minneapolis, MN, USA. In 2006, he joined the Nanjing University of Aeronautics and Astronautics (NUAA), Nanjing, China, as a Professor of electrical engineering. His research team is also engaged in high-speed circuit signal integrity, antenna, microwave components, and new method in radar signal processing. He has authored or coauthored more than 190 academic articles in refereed international journals and conference proceedings. His current research interests are in the computational electromagnetics, microwave and antennas technologies, and radar signal processing.