

## Research Article

# Application of a Sparsity Pattern and Region Clustering for Near Field Sparse Approximate Inverse Preconditioners in Method of Moments Simulations

**Carlos Delgado,<sup>1</sup> Javier Moreno,<sup>2</sup> and Felipe Cátedra<sup>1</sup>**

<sup>1</sup>Computer Science Department, University of Alcalá, Madrid, Spain

<sup>2</sup>NewFasant S.L., Guadalajara, Spain

Correspondence should be addressed to Carlos Delgado; carlos.delgado@uah.es

Received 18 May 2017; Revised 28 September 2017; Accepted 8 October 2017; Published 31 October 2017

Academic Editor: Nicolas Pinel

Copyright © 2017 Carlos Delgado et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

This document presents a technique for the generation of Sparse Inverse Preconditioners based on the near field coupling matrices of Method of Moments simulations where the geometry has been partitioned in terms of regions. A distance parameter is used to determine the sparsity pattern of the preconditioner. The rows of the preconditioner are computed in groups at a time, according to the number of unknowns contained in each region of the geometry. Two filtering thresholds allow considering only the coupling terms with a significant weight for a faster generation of the preconditioner and storing only the most significant preconditioner coefficients in order to decrease the memory required. The generation of the preconditioner involves the computation of as many independent linear least square problems as the number of regions in which the geometry is partitioned, resulting in very good scalability properties regarding its parallelization.

## 1. Introduction

Many of the modern approaches for electromagnetic analysis based on the Method of Moments (MoM) [1] rely on the idea of only storing the near field coupling terms of the impedance matrix, which typically extends about one-quarter of the wavelength under analysis. The coupling effects between distant parts of the geometry are taken into account in the matrix-vector product computation as a part of the iterative solution process, and the specific manner in which this is carried out depends on each approach. Some popular schemes are based on the use of the Multilevel Fast Multipole Algorithm (MLFMA) [2], which includes the processes of aggregation, translation, and disaggregation of multipole expansions in the computation of such products. Other schemes use matrix compression techniques [3, 4] to compute these multiplications efficiently. All these methods circumvent the burden of storing the full MoM matrix, which would easily surpass the memory capacity and cripple the efficiency for the analysis of even moderately sized problems.

Since the full coupling matrix is no longer to be calculated and due to the size of the system to be computed, iterative solvers play a major role in modern simulation methods, although it is worthwhile to mention that some approaches extend the range in which direct solvers can be applied, such as those based on the use of Macro Basis functions [5], which reduce the number of unknowns of the original problem. The necessity of iteration, however, is unavoidable for electrically large or very large problems and with the use of iterative solvers arises the problem of slow convergence for some cases, due to reasons of different nature, such as geometrical features of the model, multiscale problems, or to the intrinsic electromagnetic behavior, such as the presence of resonant regions within the geometry. In any case the use of preconditioners is very important in order to ease the iterative process, which, in many cases, accounts for most of the total simulation time.

It is noteworthy to mention a family of preconditioners that are based on the physical properties of the problem and more specifically on the idea that inverse matrix that is approximately represented by the preconditioner is

essentially an approximate solution of the electromagnetic problem. Previous works have been published for the efficient generation of physical-based preconditioners considering quasiplanar structures [6] or exploiting the quasiplanar electromagnetic behavior of conducting rough surfaces [7]. A method for the generation of a preconditioner for problems involving multiple scatterers, based on splitting the system matrix according to the types of material of the subdomains as well as currents on different parts, is shown in [8].

Another group of preconditioners that are also commonly used in the context mentioned above rely on the numerical manipulation of the system matrix rather than on physical properties of the problem under analysis. The Incomplete LU (ILU) decomposition [9–11] and the Sparse Approximate Inverse (SAI) [12–14] show a good behavior and are extensively used at the present time. The ILU preconditioner is considered slightly more efficient for the same amount of data [15], although the SAI approach is much more scalable, since it can be computed by solving independent linear least square (LLS) systems that generate the rows of the preconditioner. This property renders the SAI preconditioner very well suited to be used in very large problems that require a fair number of processing nodes. It should be noted, however, that modern ILU preconditioners based on multifrontal/multilevel schemes allow better performance than the conventional ILU approach while offering a good compromise between robustness and efficiency [16–18]. The application of multilevel ILU preconditioning techniques applied to electromagnetic problems involving the Electric Field Integral Equation has been documented in [19].

The preconditioners based on the ILU and SAI approaches are often generated considering only the near field part of the coupling matrix, which includes the strongest interactions between basis and testing functions. This allows a fast generation and reasonably reduced size, although the performance of such preconditioners may be lacking in the analysis of problems in which there are strong interactions between parts of the geometry that are physically distant, like the interaction between the feed of a reflector antenna and the main reflector or between parts of certain cavities.

The most common approach in the application of the SAI preconditioner involves the definition of a sparsity pattern, understood as the pattern of the elements of the preconditioner that are not null, which is the same as that of the near field coupling matrix [12]. In this work we propose an improved strategy based on the clustering of the least squares systems that fall under each near field region as well as a distance threshold parameter that allows a finer control of the sparsity pattern of the SAI preconditioner. We have obtained noticeable improvements on the convergence properties of the conventional use of the SAI preconditioner compared to the approach described here. Illustrative examples are given in the Numerical Results.

## 2. Description of the Approach

The application of the MoM is based on the solution of a linear system of equations as follows:

$$[Z][J] = [V], \quad (1)$$

where  $[J]$  and  $[V]$  are the current coefficient and the excitation vectors, respectively, and matrix  $[Z]$  contains the coupling terms between each basis function and each testing function of the problem. The expression of term  $(i, j)$  of  $[Z]$  can be written as

$$Z_{i,j} = \langle L(B_j), T_i \rangle, \quad (2)$$

where it is computed as the inner product of the  $i$ th testing function  $T_i$  and the field radiated by the  $j$ th active basis function  $B_j$  over the  $i$ th subdomain. This field is given by the operator  $L(B_j)$ . Consequently, each row of matrix  $[Z]$  represents the coupling terms between all the active basis functions of the geometry and one passive testing function and, in turn, a column of  $[Z]$  contains the coupling terms between one active basis function and all the passive testing functions. Since  $[Z]$  is dense and unmanageable for many realistic problems it is very common to compartmentalize the original geometry into regions and only calculate and store the coupling terms of the basis and testing functions contained within the same region, as well as those contained within adjacent regions. The typical region size is about  $\lambda/4$ , which makes the matrix containing such terms, denoted as  $[Z_{NF}]$  in this work, sparse and with a much more manageable size even for large problems. The rest of the coupling terms, noticeably weaker, are contained in the far field coupling matrix  $[Z_{FF}]$ . Many modern efficient numerical methods do not require calculating the far field coupling matrix [2–4] and consider those interactions using alternative approaches.

In order to incorporate the near field preconditioning scheme proposed in the present work, it is useful to write the MoM equation separating the coupling matrix into the near field term ( $[Z_{NF}]$ ) and the far field term ( $[Z_{FF}]$ ) as shown as follows:

$$[M]([Z_{NF}] + [Z_{FF}])[J] = [M][V], \quad (3)$$

where  $[M]$  is the preconditioner. This matrix should ideally be as close as possible to the inverse of  $[Z] = [Z_{NF}] + [Z_{FF}]$ . However, it is very important to keep the generation procedure scalable and efficient both in terms of size and CPU time.

Under the assumption that the near field part of the coupling matrix is stronger than the far field term, it is possible to rewrite (3) in the following fashion:

$$[M][Z_{NF}][J] \approx [M][V]. \quad (4)$$

**2.1. Region Clustering in the Generation of the Preconditioner.** A commonly used SAI preconditioner relies on the generation of  $[M]$  by minimizing the *Frobenius norm* of  $\{[I] - [M][Z_{NF}]\}$ , where  $[I]$  is the identity matrix. The scalability properties of this preconditioner arise from the possibility of decomposing such expression into the minimization of the norm of the difference between each row of the identity

matrix and the preconditioner multiplied by the near field coupling matrix as follows:

$$\min \| [I] - [M] [Z_{\text{NF}}] \|_F^2 = \sum_{k=1}^n \min \| \mathbf{e}_k - \mathbf{m}_k [Z_{\text{NF}}] \|_2^2, \quad (5)$$

where  $n$  indicates the total number of basis functions of the problem,  $\mathbf{e}_k$  stands for the  $k$ th row of the identity matrix, and  $\mathbf{m}_k$  denotes the  $k$ th row of the preconditioner  $[M]$ .

It should be remarked that the technique described in this work relies on the compartmentalization of the computational space in terms of *regions*, so that only the coupling terms between subdomain functions located in the same or in adjacent regions are to be computed. This is a widely adopted scheme by many modern methods. Under this context, it is possible to take advantage of the reutilization of the same LLS matrix to obtain as many rows of the preconditioner as the number of the unknowns contained in each region. This can be illustrated by rewriting (5) in the following manner:

$$\begin{aligned} \sum_{k=1}^n \min \| \mathbf{e}_k - \mathbf{m}_k [Z_{\text{NF}}] \|_2^2 \\ = \sum_{r=1}^R \min \| [I_r] - [M_r] [Z_{\text{NF}}] \|_F^2, \end{aligned} \quad (6)$$

where  $R$  indicates the total number of regions of the problem. Considering at this point that  $N_i$  is the number of subdomain functions contained in region  $i$  and that  $S_i^r$  represents the absolute index of the  $i$ th subdomain contained in region  $r$ , the  $[I_r]$  matrix in (6) consists of  $N_r$  rows of the unit matrix

$$[I_r] = \begin{bmatrix} \mathbf{e}_{S_1^r} \\ \vdots \\ \mathbf{e}_{S_{N_r}^r} \end{bmatrix} \quad (7)$$

and, analogously,  $[M_r]$  contains  $N_r$  rows of the preconditioner:

$$[M_r] = \begin{bmatrix} \mathbf{m}_{S_1^r} \\ \vdots \\ \mathbf{m}_{S_{N_r}^r} \end{bmatrix}. \quad (8)$$

**2.2. Parametric Sparsity Pattern.** Noting that  $[M_r]$  will be generally a full matrix it is necessary to define a sparsity pattern that enforces null values for some of its coefficients.

It is common to see approaches in the literature that consider such a sparsity pattern to be identical as that of the original near field coupling matrix [12]. We introduce an alternative approach by defining in this context the *sparsity distance parameter*  $\xi$  as the distance threshold that determines whether the subdomains contained in a region are to be included in the sparsity pattern. Let us denote for this purpose  $R_i$  as the region in which subdomain  $i$  is contained and  $D(R_i, R_j)$  as the distance between the centres of regions  $R_i$  and  $R_j$ , where  $i$  and  $j$  are two arbitrary subdomains. The sparsity mask matrix for region  $r$  is defined as

$$[\Omega]_r^\xi = \begin{bmatrix} \Omega_{1,1} & \cdots & \Omega_{1,n} \\ \vdots & \ddots & \vdots \\ \Omega_{n,1} & \cdots & \Omega_{n,n} \end{bmatrix}, \quad (9)$$

$$\text{where } \Omega_{i,j} = \begin{cases} 1, & \text{if } D(R_i, R_j) \leq \xi \\ 0, & \text{if } D(R_i, R_j) > \xi. \end{cases}$$

The  $[\Omega]_r^\xi$  matrix, applied for all the regions, defines the sparsity pattern used to generate the preconditioner. With this consideration, the right hand side of expression (6) can be approximated as

$$\begin{aligned} \sum_{r=1}^R \min \| [I_r] - [M_r] [Z_{\text{NF}}] \|_F^2 \\ \cong \sum_{r=1}^R \min \| [\widetilde{I}_r] - [\widetilde{M}_r] [\widetilde{Z}_{\text{NF}}] \|_F^2, \end{aligned} \quad (10)$$

where  $[\widetilde{M}_r]$  is an approximation of  $[M_r]$  obtained as the solution of the LLS where the matrices marked with the tilde overscore denote the application of the sparsity pattern by computing the Hadamard product with the *sparsity mask* and eliminating the null rows and columns of the result:

$$\begin{aligned} [\widetilde{I}_r] &= \mathfrak{N}([\Omega]_r^\xi \odot [I_r]), \\ [\widetilde{Z}_{\text{NF}}] &= \mathfrak{N}([\Omega]_r^\xi \odot \mathfrak{I}_r^\tau([Z_{\text{NF}}])). \end{aligned} \quad (11)$$

In this expression  $\odot$  denotes the element-wise matrix multiplication operator and  $\mathfrak{N}([A])$  represents the matrix that results from the elimination of the null rows and columns of  $[A]$ . The term  $\mathfrak{I}_r^\tau([Z_{\text{NF}}])$ , in turn, is an operator that returns a filtered version of  $[Z_{\text{NF}}]$  discarding the coefficients with a module lower than  $\tau$  times the largest self-impedance in the region where they belong. The  $\tau$  parameter, named *preprocessing threshold parameter* through the rest of this document, allows a reduction of the size of the LLS problems to be solved for the generation of the preconditioner by eliminating low-magnitude rows and columns, resulting in faster computation. Considering these comments  $\mathfrak{I}_r^\tau([Z_{\text{NF}}])$  can be written as follows:

$$\mathcal{Z}_r^\tau([Z_{\text{NF}}]) = \begin{bmatrix} \mathcal{Z}_{1,1} & \cdots & \mathcal{Z}_{1,n} \\ \vdots & \ddots & \vdots \\ \mathcal{Z}_{n,1} & \cdots & \mathcal{Z}_{n,n} \end{bmatrix}, \quad (12)$$

with  $0 \leq \tau \leq 1$ , where  $\mathcal{Z}_{i,j} = \begin{cases} 0, & \text{if } |[Z_{\text{NF}}]_{i,j}| \leq \tau \cdot \max(|[Z_{\text{NF}}]_{k,k}|) \text{ with } k \in \text{Region } r \\ [Z_{\text{NF}}]_{i,j}, & \text{if } |[Z_{\text{NF}}]_{i,j}| > \tau \cdot \max(|[Z_{\text{NF}}]_{k,k}|) \text{ with } k \in \text{Region } r. \end{cases}$

The preconditioner, therefore, can be computed by calculating the result of  $R$  independent LLS problems, which can be distributed among an arbitrary number of processing nodes, and assembling the preconditioner using the partial  $[\widetilde{M}_r]$  matrices. It is convenient, however, to filter these results to retain only the elements that pose a significant weight for each row of the preconditioner. Experimental results have shown that for typical applications the size of the preconditioner can be reduced at least by an order of magnitude without a compromise in the convergence performance. Expression

(12) illustrates the row filtering of  $[\widetilde{M}_r]$  introducing the *postprocessing threshold parameter*  $\varsigma$ :

$$[\widetilde{M}_r] = \Psi^\varsigma([\widetilde{M}_r]), \quad \text{with } 0 \leq \varsigma \leq 1, \quad (13)$$

where  $\Psi^\varsigma([\widetilde{M}_r])$  retains the elements with an absolute value over  $\varsigma$  times the largest element in each row of the preconditioner:

$$\Psi^\varsigma([\widetilde{M}_r]) = \begin{bmatrix} \Psi_{1,1} & \cdots & \Psi_{1,n} \\ \vdots & \ddots & \vdots \\ \Psi_{n,1} & \cdots & \Psi_{n,n} \end{bmatrix}, \quad \text{where } \Psi_{i,j} = \begin{cases} 0, & \text{if } |[\widetilde{M}_r]_{i,j}| \leq \varsigma \cdot \max(|[\widetilde{M}_r]_{i,j}|) \text{ within row } i \\ [\widetilde{M}_r]_{i,j}, & \text{if } |[\widetilde{M}_r]_{i,j}| > \varsigma \cdot \max(|[\widetilde{M}_r]_{i,j}|) \text{ within row } i. \end{cases} \quad (14)$$

The sparsity distance parameter  $\xi$  introduced in this section allows a fine control of the size of the preconditioning matrix. Large values (similar to the size of the geometry) imply no restrictions in the sparsity pattern of the resulting preconditioner, which can lead to the computation of the exact inverse of  $[Z_{\text{NF}}]$  with a prohibitive computational cost. Low values, around the size of the regions in which the geometry has been previously partitioned, render an approximate inverse with the same size as  $[Z_{\text{NF}}]$  that can be computed fast. These sizes can then be modified by the filtering threshold  $\varsigma$  that discards the least significant terms. Our experience, as shown in the next section, shows that sparsity distance parameter values ranging from  $0.5\lambda$  to  $\lambda$  present a good tradeoff between size and performance.

### 3. Numerical Results

Some test cases have been selected in this section in order to illustrate the performance of the approach described in the present document. All the simulations have been run on an HP Z820 workstation using 16 Intel Xeon E5-2660 2.2 GHz processing cores and 128 GB of RAM. In all the cases we have applied the Method of Moments combined with the Multilevel Fast Multipole Algorithm using  $0.25\lambda$  as the region size unless otherwise indicated. The SAI preconditioners have been obtained distributing the computation of  $[\widetilde{M}_r]$  shown in expression (13) for all the regions of the

problem between the computing nodes using the OpenMP paradigm [20]. The Generalized Minimal Residual (GMRES) iterative solver [21] with a restart parameter of 300 has been used for all the simulations presented in this section. The parallelization scheme of the SAI preconditioner followed in the simulations shown in this section consists of a loop that iterates over the regions in which the geometry has been previously partitioned. For each iteration the corresponding LLS matrix is assembled and the LAPACK numerical library [22] is used to obtain the solution. Since we use OpenMP to merely distribute the iterations of this loop over all the processing nodes, it is extremely important that the iterations are independent of each other, as is the case for the proposed preconditioner.

In the results presented in this section  $\theta$  represents the polar angle measured from the  $z$ -axis and  $\phi$  denotes the azimuthal angle measured from the  $x$ -axis, with  $0^\circ \leq \theta \leq 180^\circ$ ,  $0^\circ \leq \phi < 360^\circ$ . These angular coordinates have been illustrated in Figure 1 for the sake of clarification.

Figure 2 depicts the geometry of the PLACYL case, originally proposed for the JINA EM 2006 workshop [23], consisting of a cylinder with a radius of 0.2 m and a length of 1 m connected to a semispherical surface on one side and located on top of a plane plate that extends 1.8 m along the  $X$  direction and 1.2 m along the  $Y$  direction. The gap between the cylinder and the plate is 0.02 m. The monostatic RCS of this setup has been computed at a frequency of 10 GHz for

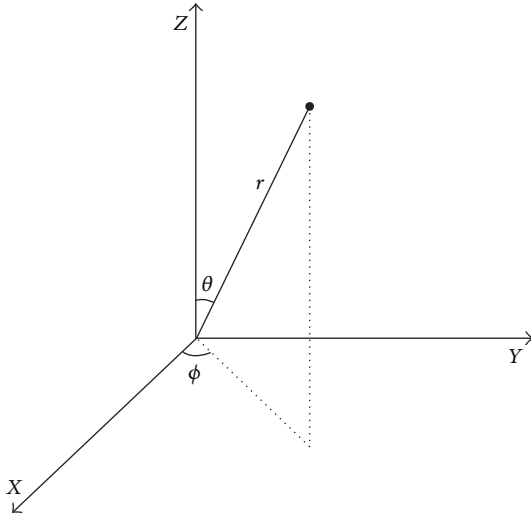


FIGURE 1: Depiction of the spherical coordinates considered.

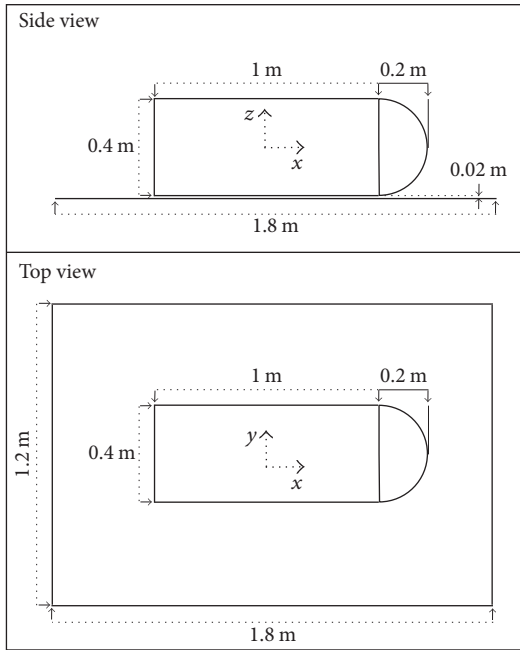
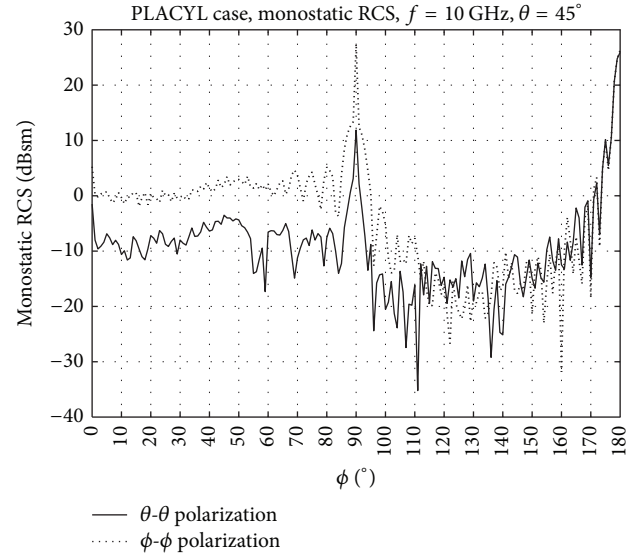


FIGURE 2: Geometrical model of the PLACYL case.

a set of 181 theta polarized incident waves corresponding to the directions given by  $\theta = 45^\circ$  and  $\phi$  ranging from  $0^\circ$  to  $180^\circ$ . The electrical length of the cylinder at this frequency is  $40\lambda$  and the radius is  $6.67\lambda$ , while the plate is  $60\lambda$  long and  $40\lambda$  wide. The Electric Field Integral Equation (EFIE) formulation has been considered for all the surfaces of this case instead of the Combined Field Integral Equation (CFIE) in order to obtain a moderately ill-conditioned problem useful to test the performance of the preconditioners. The geometry has been meshed using a sampling rate of 10 divisions per wavelength, resulting 831,734 unknowns, and the compartmentalization of the geometry has generated 73,590 regions. The iteration error threshold of the GMRES solver has been  $10^{-3}$ . Figure 3

FIGURE 3: Monostatic RCS results for the PLACYL case,  $f = 10$  GHz.

shows the results obtained for this case. Table 1 shows the CPU time required for the generation of the preconditioner as well as the memory for its storage considering different configurations and the average number of iterations per monostatic direction necessary to compute the solution. The values of the SAI parameters (sparsity distance  $\xi$ , preprocessing threshold  $\tau$ , and postprocessing threshold  $\varsigma$ ) defined in the approach presented in this work are also included in the table.

The results plotted in Figure 3 show that there is a certain amount of specular reflection on the hemisphere and between the hemisphere and the plate at  $\phi = 0^\circ$  that provides a RCS level that fades in a very narrow angular margin. The convergence for this direction is not especially slow because of the predominant specular effect over the high-order coupling reflections between plate and hemisphere. From  $\phi = 0^\circ$  to  $\phi = 90^\circ$  the backscattering RCS contributions are due to diffuse reflections on the plate, cylinder, and high-order effects between both bodies, leading to slow convergence due to the low RCS levels and strong coupling. The RCS increases very noticeably for angular values around  $\phi = 90^\circ$ , due to the larger cross section of the cylinder that provides specular reflection, as well as the high-order specular reflections between the side of the cylinder and the plate and creeping effects on the cylinder. From  $\phi = 90^\circ$  to  $\phi = 180^\circ$  the contribution of the interaction between the flat part of the cylinder and the plate becomes increasingly stronger and for the angular margin around  $\phi = 180^\circ$  the double reflection between the plate and the flat cap of the cylinder is overwhelmingly predominant, yielding high RCS values and easing the convergence rate.

It can be seen in Table 1 that the memory requirements for the storage of the preconditioner have a weak dependency on the sparsity distance parameter  $\xi$ . This is due to the use of the postprocessing threshold  $\varsigma$  that eliminates the least significant terms of the preconditioner. In general, it is normal to see a mild increase of the size of the preconditioner as  $\xi$  increases



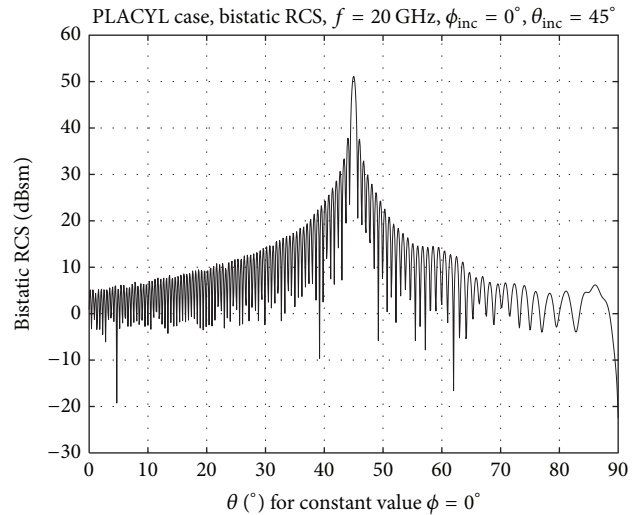
TABLE 1: Convergence performance for several SAI parameters.

	Precond. generation time (s)	Precond. size (MB)	Avg. number of iterations per direction ( $\theta$ pol.)	Avg. number of iterations per direction ( $\phi$ pol.)
<i>No preconditioner</i>	—	—	394.15	332.42
<i>Conventional SAI</i>	189	378.43	192.00	172.05
<i>Proposed SAI</i> Sparsity distance ( $\xi$ ) = $0.25 \lambda$ Preprocessing threshold ( $\tau$ ) = 0.03 Postprocessing threshold ( $\zeta$ ) = 0.03	43	284.57	181.94	158.73
<i>Proposed SAI</i> Sparsity distance ( $\xi$ ) = $0.5 \lambda$ Preprocessing threshold ( $\tau$ ) = 0.03 Postprocessing threshold ( $\zeta$ ) = 0.03	192	281.78	129.81	115.98
<i>Proposed SAI</i> Sparsity distance ( $\xi$ ) = $\lambda$ Preprocessing threshold ( $\tau$ ) = 0.03 Postprocessing threshold ( $\zeta$ ) = 0.03	1208	304.33	111.78	103.21

because, as more coefficients are included in the sparsity pattern of the preconditioner, a number of them are probably going to have a magnitude that exceeds the threshold, adding information to the computed preconditioner.

In addition to the monostatic results shown above, it is interesting to test the behavior of the convergence of the proposed approach for a different number of configurations. For the next example we have considered the PLACYL geometry increasing the frequency to 20 GHz, resulting in an electrical length of  $80 \lambda$  for the cylinder with a radius of  $13.35 \lambda$  and a plate size of  $120 \lambda \times 80 \lambda$ . Now the CFIE formulation with parameter  $\alpha = 0.5$  has been applied for the closed body (cylinder and semisphere), and the EFIE formulation has been applied for the plane plate. In this case, meshing with the standard rate of 10 samples per wavelength, we obtain 3,358,631 unknowns and 297,132 regions. A bistatic RCS analysis has been performed for the excitation incidence given by  $\theta = 45^\circ$  and  $\phi = 0^\circ$  with observation directions separated by  $0.1^\circ$  steps between  $\theta = 0^\circ$  and  $\theta = 90^\circ$ . The sparsity distance parameter considered in this case has been  $\xi = 0.5 \lambda$ , and the preprocessing and postprocessing thresholds have been, respectively,  $\tau = 0.05$  and  $\zeta = 0.03$ . Figure 4 shows the results obtained for the  $\theta$ -polarized incident field considering an iteration error threshold of  $10^{-3}$ . Figure 5 shows the convergence analysis of this case considering the use of the conventional SAI approach (with the same sparsity pattern as the near field coupling matrix), the SAI approach proposed in this document and without any preconditioner.

For the next example, the RCS of a realistic model of an Opel Astra car has been simulated at a frequency of 5.9 GHz. The geometry is shown in Figure 6 and has been modeled using 282 NURBS surfaces. This case consists of 2,255,500 unknowns and 200,936 regions using a sampling rate of 10 samples per wavelength. In order to analyze a more realistic problem, several surfaces have been treated as dielectric layers, defining their relative permeability  $\epsilon_r$ . The windows and lights have been modeled using a thin layer

FIGURE 4: Bistatic RCS results of the PLACYL case,  $f = 20$  GHz.

approximation with  $\epsilon_r = 4$ . The wheels have been modeled by means of a thin layer approximation using  $\epsilon_r = 7$ , and dielectric losses have been introduced by assigning the loss tangent  $\tan(\delta) = 0.04$ , considering these values as a realistic approach for the rubber compound of the tires [24]. The size of the car is approximately 4.14 meters long, 1.64 meters wide, and 1.35 meters tall, which at the working frequency corresponds to  $81.5 \lambda$  tall,  $32.25 \lambda$  wide, and  $26.55 \lambda$  high. The bistatic RCS has been computed for a  $\theta$ -polarized plane wave impinging from  $\theta = 80^\circ$  and  $\phi = 45^\circ$ . The observation directions are contained in the  $\phi = 45^\circ$  angular cut  $\theta$  ranging from  $0^\circ$  to  $90^\circ$  in  $0.1^\circ$  steps. The EFIE formulation has been applied, due to the geometrical and material features of this problem. Figure 7 shows the computed results. Convergence has been obtained for an error of  $10^{-3}$  in 281 iterations. Realistic cases often offer a slow convergence due to the more

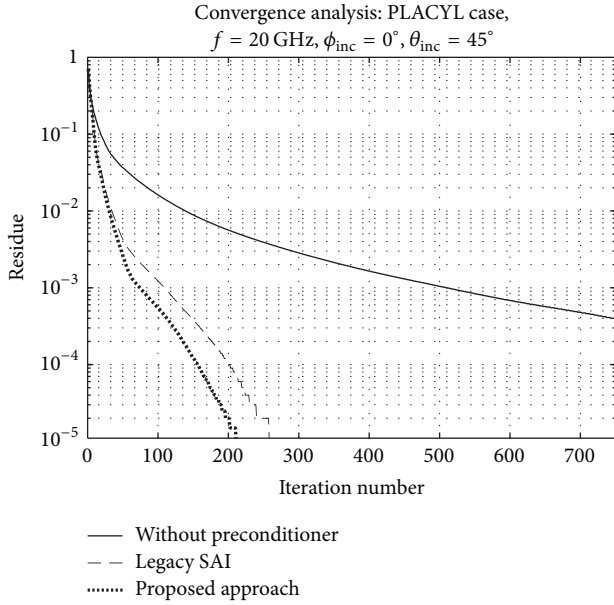
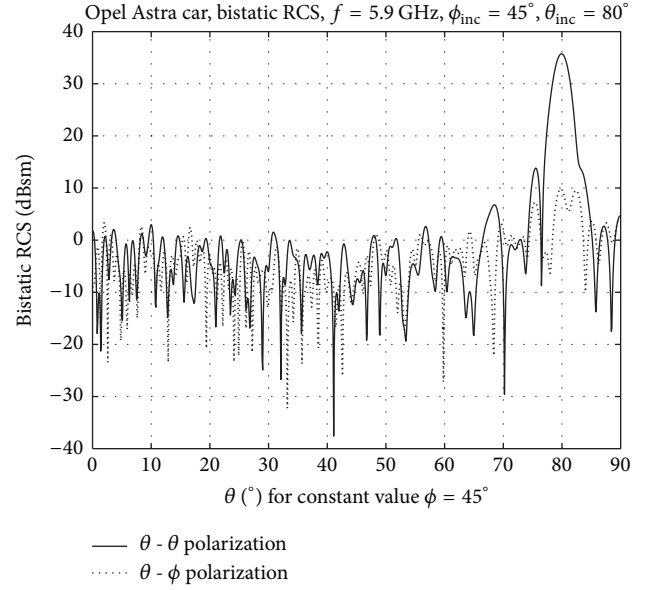
FIGURE 5: Convergence analysis for the PLACYL case,  $f = 20 \text{ GHz}$ .

FIGURE 6: Geometrical model of an Opel Astra car.

complex geometrical models. The inclusion of very small patches, degenerate surfaces, and other sublambda details produces an irregular mesh that favors slower convergence rates. In addition, the use of dielectric materials assigned to some of the surfaces of the model combined with PEC surfaces in the rest of the body also poses a burden to the convergence properties of the problem. In this case the conventional SAI approach did not reach convergence and stalled with an error value of  $2.2 \cdot 10^{-2}$ . The conventional SAI preconditioner uses the same sparsity pattern as the near field coupling matrix, which is in essence equivalent to generating the preconditioner using a sparsity distance of  $\xi = 0.25 \lambda$  in this case. However, due to the features of this geometrical model the resulting preconditioner does not adequately resemble the inverse of  $[Z_{\text{NF}}]$ . The extended value of the sparsity distance used with the proposed approach ( $\xi = 0.75 \lambda$ ) allows the computation of a denser preconditioner that provides better convergence. The total simulation time

FIGURE 7: Bistatic RCS results for the Opel Astra car,  $f = 5.9 \text{ GHz}$ .

has been 18,976 seconds. The preconditioner in this case has been generated with a sparsity distance parameter  $\xi = 0.75 \lambda$ , a preprocessing threshold  $\tau = 5 \cdot 10^{-3}$ , and the postprocessing threshold  $\varsigma = 2 \cdot 10^{-2}$  in 2,220 s and has required 5.9 GB of RAM.

#### 4. Conclusion

A new sparsity pattern and filtering approach that can be used to generate Sparse Approximate Inverse preconditioners considering the near field part for computations based on the Method of Moments have been presented in this document. The rows of the preconditioner are generated as groups that result from the solution of similar least squares problems. In addition, a parametric distance sparsity parameter allows a fine control of the amount of data to be used. The scalability properties of this preconditioner are excellent and make it well suited to solve large and realistic problems with a moderate computing time and RAM footprint. Some illustrative examples have been provided in order to test the performance of the proposed approach.

From the experience gathered by the authors, some recommended typical values of the parameters of the proposed preconditioner can be a sparsity distance ranging between  $0.5 \lambda$  and  $\lambda$ , as well as preprocessing and postprocessing thresholds between 0.01 and 0.03. For those cases with an especially difficult convergence the sparsity distance can be raised up to  $2 \lambda$  at the expense of higher CPU time and memory consumption. In cases where the memory consumption is a critical issue the postprocessing threshold can be increased to values around 0.05 in order to reduce the size of the preconditioner, with the possible consequence of a slightly slower convergence.

## Conflicts of Interest

The authors declare that they have no conflicts of interest.

## Acknowledgments

This work has been partially supported by the Spanish Ministry of Economy and Competitiveness (Project Ref. TEC 2013-46587-R and Grant Ref. PTQ-14-07060) and by the University of Alcalá (Project Ref. CCG2016/EXP-082).

## References

- [1] R. F. Harrington, *Field Computation by Moment Methods*, McMillan, New York, NY, USA, 1968.
- [2] J. M. Song and W. C. Chew, "Multilevel fast-multipole algorithm for solving combined field integral equations of electromagnetic scattering," *Microwave and Optical Technology Letters*, vol. 10, no. 1, pp. 14–19, 1995.
- [3] M. Bebendorf and S. Rjasanow, "Adaptive low-rank approximation of collocation matrices," *Computing: Archives for Scientific Computing*, vol. 70, no. 1, pp. 1–24, 2003.
- [4] K. Zhao, M. N. Vouvakis, and J.-F. Lee, "The adaptive cross approximation algorithm for accelerated method of moments computations of EMC problems," *IEEE Transactions on Electromagnetic Compatibility*, vol. 47, no. 4, pp. 763–773, 2005.
- [5] V. V. S. Prakash and R. Mittra, "Characteristic basis function method: a new technique for efficient solution of method of moments matrix equations," *Microwave and Optical Technology Letters*, vol. 36, no. 2, pp. 95–100, 2003.
- [6] P. Naenna and J. T. Johnson, "A physically-based preconditioner for quasi-planar scattering problems," *Institute of Electrical and Electronics Engineers. Transactions on Antennas and Propagation*, vol. 56, no. 8, part 1, pp. 2421–2426, 2008.
- [7] S. Tournier, P. Borderies, and J.-R. Poirier, "Integral equations physically-based preconditioner for two-dimensional electromagnetic scattering by rough surfaces," *Institute of Electrical and Electronics Engineers. Transactions on Antennas and Propagation*, vol. 59, no. 10, pp. 3764–3771, 2011.
- [8] J. Von Hagen and W. Wiesbeck, "Physics-based preconditioner for iterative algorithms in MoM-problems," *IEEE Transactions on Antennas and Propagation*, vol. 50, no. 9, pp. 1315–1316, 2002.
- [9] J. Lee, J. Zhang, and C.-C. Lu, "Incomplete LU preconditioning for large scale dense complex linear systems from electromagnetic wave scattering problems," *Journal of Computational Physics*, vol. 185, no. 1, pp. 158–175, 2003.
- [10] Y. Saad, "ILUT: A dual threshold incomplete LU factorization," *Numerical Linear Algebra with Applications*, vol. 1, no. 4, pp. 387–402, 1994.
- [11] T. Malas and L. Gürel, "Incomplete LU preconditioning with the multilevel fast multipole algorithm for electromagnetic scattering," *SIAM Journal on Scientific Computing*, vol. 29, no. 4, pp. 1476–1494, 2007.
- [12] J. Lee, J. Zhang, and C.-C. Lu, "Sparse inverse preconditioning of multilevel fast multipole algorithm for hybrid integral equations in electromagnetics," *Institute of Electrical and Electronics Engineers. Transactions on Antennas and Propagation*, vol. 52, no. 9, pp. 2277–2287, 2004.
- [13] E. Chow, "A priori sparsity patterns for parallel sparse approximate inverse preconditioners," *SIAM Journal on Scientific Computing*, vol. 21, no. 5, pp. 1804–1822, 2000.
- [14] B. Carpentieri, I. S. Duff, L. Giraud, and G. Sylvand, "Combining fast multipole techniques and an approximate inverse preconditioner for large electromagnetism calculations," *SIAM Journal on Scientific Computing*, vol. 27, no. 3, pp. 774–792, 2005.
- [15] M. Benzi and M. Tuna, "A comparative study of sparse approximate inverse preconditioners," *Applied Numerical Mathematics*, vol. 30, no. 2–3, pp. 305–340, 1999.
- [16] R. E. Bank and C. Wagner, "Multilevel ILU decomposition," *Numerische Mathematik*, vol. 82, no. 4, pp. 543–576, 1999.
- [17] Y. Saad and J. Zhang, "BILUM: Block Versions of Multielimination and Multilevel ILU Preconditioner for General Sparse Linear Systems," *SIAM Journal on Scientific Computing*, vol. 20, no. 6, pp. 2103–2121, 1999.
- [18] M. Bollhöfer and Y. Saad, "Multilevel preconditioners constructed from inverse-based ILUs," *SIAM Journal on Scientific Computing*, vol. 27, no. 5, pp. 1627–1650, 2006.
- [19] B. Carpentieri and M. Bollhöfer, "Symmetric inverse-based multilevel ilu preconditioning for solving dense complex non-hermitian systems in electromagnetics," *Progress in Electromagnetics Research*, vol. 128, pp. 55–74, 2012.
- [20] R. Chandra, R. Menon, L. Dagum, D. Kohr, D. Maydan, and J. McDonald, *Parallel Programming in OpenMP*, Morgan Kaufmann, 2000.
- [21] Y. Saad and M. H. Schultz, "GMRES: a generalized minimal residual algorithm for solving nonsymmetric linear systems," *SIAM Journal on Scientific and Statistical Computing*, vol. 7, no. 3, pp. 856–869, 1986.
- [22] E. Anderson, Z. Bai, C. Bischof et al., *LAPACK Users Guide*, SIAM, Philadelphia, PA, USA, 1999.
- [23] E. Kemptner and Erich, "Results to test case PLACYL at Workshop EM JINA," in *Proceedings of the Workshop Journées Internationales de Nice sur les Antennes (EM JINA)*, Nice, France, 2006.
- [24] M. D. Brzeska, *RF Modelling and Characterization of Tyre Pressure Sensors and Vehicle Access Systems*, KIT Scientific Publishing, 2015.



