$\mathcal{O}(N)$ Nested Skeletonization Scheme for the Analysis of Multiscale Structures Using the Method of Moments

Mario Alberto Echeverri Bautista, *Member, IEEE*, Matteo Alessandro Francavilla, *Member, IEEE*, Per-Gunnar Martinsson, and Francesca Vipiana, *Senior Member, IEEE*

***Abstract*—We present a scheme to compress the method of moments (MoM) matrix, which is linear in complexity for low to intermediate frequency problems in electromagnetics. The method is fully kernel independent and easy to implement using existing codes. The $\mathcal{O}(N)$ complexity for both memory and time (setup and matrix–vector product) is achieved thanks to the application of a recursive skeletonization to the $\mathcal{H}^2$ matrix structure of the MoM matrix that uses the nested nature of the far interactions. The interpolative decomposition is applied in a novel manner in order to compress the "far-field signature" of the groups of basis functions. Moreover the scheme is fully characterized and it proves itself well suited for the analysis of multiscale structures.**

***Index Terms*—Fast solvers, integral equations (IEs), method of moments (MoM).**

I. Introduction

Surface integral equations (SIEs) are among the most used formulations for the analysis of problems in electromagnetics due to their accuracy and favorable scaling properties. The method of moments (MoM) discretization of integral equation (IE) operators is conveniently done via subsectional elements, such as the Rao–Wilton–Glisson (RWG) basis [1] for surface discretizations. The electric field integral equation (EFIE) has proven extremely accurate, and is therefore, the usual choice from low to intermediate frequency cases. As frequency increases, it is often combined with the magnetic field integral equation (MFIE) yielding the combined field integral equation (CFIE), a resonance-free formulation. Although the number of unknowns is relatively small (compared to volume discretizations), the well-known drawback of MoM discretizations is that they result in a dense matrix system. Many strategies have been explored to reduce the cost of matrix filling and inversion, typically labeled as *fast solvers* [2]: these techniques are aimed at reducing storage requirements and speeding up matrix–vector (MV) multiplications in iterative solutions. Fast solvers can be classified in two main classes: kernel-based factorizations and algebraic factorizations. Historically, kernel-dependent techniques were addressed first, starting from the revolutionary work by Rokhlin [3] paving the way for fast multipole methods (FMM) techniques [4]–[6]. Kernel-based factorizations, however, rely on analytic expansions of the integral kernel: if the kernel (i.e., the problem, or simply the formulation) is varied, the expansion needs be recomputed (assuming such expansion exists). Conversely, kernel-independent techniques are highly desirable due to their ability to solve a wide plethora of problems without substantial modifications to the code: In the following, we will limit our focus to this latter class of fast solvers.

All techniques are generally based on the following observation: subblocks of the system matrix representing interactions between two clustered groups of "far" basis functions are (numerically) rank deficient. This property guarantees that a compression of such matrix blocks is possible, although it does not tell how to construct it: The difference among different fast solvers is indeed the way this factorization is constructed. Because of the need of subdividing the geometry into clusters of basis functions, clustering algorithms (e.g., Octree) are a key ingredient of most fast solvers. Once the computational domain has been geometrically partitioned, the same partitioning can be related to the matrix itself, highlighting blocks which can be compressed by means of some factorization scheme. Because of the fact that compressible blocks correspond to distant clusters, this geometrical distance terminology is often associated to the matrix itself, which is then split in a *near-field* (incompressible) portion, and its *far-field* (compressible) complement.

The achievable computational gain depends on the structure employed for representing the *far-field* portion of the matrix. In the $\mathcal{H}$-matrix representation [7], at each level of the clustering far-field interactions are independently compressed: This means that compression is carried out separately at each *scale*,

Manuscript received May 10, 2016; revised November 22, 2016; accepted December 11, 2016. Date of publication December 28, 2016; date of current version January 18, 2017. This work was supported by COST Action IC1406 High-Performance Modelling and Simulation for Big Data Applications (cHiPSet) supported by COST (European Cooperation in Science and Technology).

M. A. Echeverri Bautista is with the Department of Electronics and Telecommunications, Istituto Superiore Mario Boella, Turin, 10138, Italy (e-mail: mechhz@gmail.com).

M. A. Francavilla is with the Istituto Superiore Mario Boella, Turin, 10138, Italy (e-mail: francavilla@ismb.it).

P.-G. Martinsson is with the Applied Mathematics, University of Colorado Boulder, Boulder, CO 80309 USA (e-mail: Per-gunnar.Martinsson @colorado.edu).

F. Vipiana is with Department of Electronics and Telecommunications, Politecnico di Torino, Turin, 10129, Italy (e-mail: francesca.vipiana@ polito.it).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/JMMCT.2016.2645838

2379-8793 © 2017 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission.
See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

and no communication between different scales is required. As one might guess, this is not the most efficient strategy, especially in the presence of multiscale discretizations. $\mathcal{H}$-matrix compression techniques are very well known and employed due to their relative simplicity: Classic examples can be found in the adaptive cross approximation (ACA) [8], [19]. Nested compression schemes, in which *bases* are recursively expressed in terms of the bases at finer level of the clustering, are introduced to mitigate the inefficiencies of $\mathcal{H}$-matrix representations: These matrix representations, formally a subset of $\mathcal{H}$-matrices, are known as $\mathcal{H}^2$-matrices [9].

A second key factor for the computational gain is played by the admissibility condition [7], which can be either *weak* or *strong*. In short, the admissibility condition fixes a distance between clusters for the corresponding matrix block to be compressed. A weak admissibility condition yields matrices in which only interactions of a cluster with itself are computed and stored without compression: With the proper ordering, this induces a block diagonal near-field matrix. Conversely, if a strong admissibility condition is assumed, interactions between neighboring (touching) clusters are not compressed: this obviously increases storage requirements for the near-field matrix, but this is largely compensated by the significantly higher compression rates achievable on far-field couplings. If N denotes the dimension of the linear system, using the strong admissibility condition allows to solve static problems with provable $\mathcal{O}(N \log N)$ and $\mathcal{O}(N)$ complexities, for $\mathcal{H}$- and $\mathcal{H}^2$-matrix representations, respectively.

Matrices with weak admissibility condition applied to $\mathcal{H}^2$-matrix format are known as hierarchically semiseparable (HSS) matrices; HSS matrices have been successfully used for the development of fast direct methods using Nystrom discretizations [10], [12] and collocation schemes [13], [14]. Previous attempts using the weak admissibility for MoM matrices resulted in complexities higher than linear for the compression stages [15]. The compression stage is accelerated thanks to the concept of proxy surfaces, auxiliary surface introduced in [10]. A similar idea was already foreseen in [11], where auxiliary sources were place around a group of basis functions for the construction of a reduced representation of the scattered field.

In this paper, we propose a novel scheme, labeled in the following as *nested skeletonization scheme* (*NSS*), directly inspired by the work in [12] where a direct solver for HSS matrices is introduced for 1-D problems. Our scheme is used for the compression of the MoM matrix with a strong admissibility condition. We demonstrate that the solver has linear complexity for fully 3-D problems, thanks to the control of the rank during compression stages provided by the *strong* admissibility condition. The solver is well suited to problems in the low and intermediate frequency regimes. The kernel free nature of this paper makes it applicable to a broad range of formulations. Also, a thorough derivation and discussion of the computational cost of the algorithm is presented; to the best of the authors' knowledge, this systematic study for the compression of the entire MoM matrix had not been addressed previously.

The use of $\mathcal{H}^2$ matrices with strong admissibility in electromagnetics can be tracked back to kernel interpolation-based solvers [16]–[18], where the degenerate nature of the Green's function is cleverly exploited to factor the kernel using Lagrange polynomials. The solver proposed in this paper, on the other hand, does not explicitly require an analytic expansion of the kernel; rather it directly compresses matrix blocks. Such compression techniques are often referred to as kernel free or algebraic, and include the ACA [19], multilevel ACA [20], nested equivalent sources approximation (NESA) [21], multilevel matrix decomposition algorithm [22].

In [23], the strong admissibility condition was used in the compression of the near-field matrix of a standard multilevel fast multipole algorithm with $\mathcal{O}(N \log N)$ complexity, obtaining however a kernel-dependent solver; the same authors successfully used a similar scheme for generating a preconditioner for multiscale structures [24]. Instead our solver is completely kernel free and achieves a $\mathcal{O}(N)$ complexity. Another work using strong admissibility is the fast summation technique for 1-D problems presented in [25]; focus of that work is the analysis of potential problems for 1-D geometries, while our scheme is demonstrated to efficiently deal with 3-D vector problems.

At a difference from [15] and [23], our approach discards the use of fully discretized proxy surfaces using RWG vector functions, which can be cumbersome in practical codes. Instead, the proposed approach employs fields scattered by the sources and measured in a finite set of points located in the artificial proxy surface. Our work shares the objectives of the recent kernel-independent solver for MoM matrices presented in [21]; nevertheless the present approach is much simpler as it avoids the use of inverse source procedures (e.g., using truncated SVD pseudoinverses), consequently, reducing the number of user-defined parameters to set.

To summarize, we present a state of the art kernel-independent solver with linear complexity for MoM solutions in the low and intermediate frequency regimes. A brief background is presented in Section II, a detailed description of the proposed scheme is presented in Section III. The solver is fully characterized and its efficiency and accuracy addressed in Section IV; finally, Section V summarizes our findings.

II. Background

We consider a perfect electric conductor body on which the unknown current density $\mathbf{J}$ is approximated as a linear combination of basis functions

$$\mathbf{J}(\mathbf{r}) \simeq \sum_{n=1}^{N} I_n \mathbf{f}_n(\mathbf{r}) \tag{1}$$

where I_n corresponds to the current expansion coefficients and $\mathbf{f}_n(\mathbf{r})$ to the RWG basis functions [1]; applying the MoM and Galerkin testing to the EFIE, the problem is transformed into the linear system as

$$Z_{\mathrm{EFIE}} I = (Z^{A}_{\mathrm{EFIE}} + Z^{\phi}_{\mathrm{EFIE}}) I = V_{\mathrm{EFIE}} \tag{2}$$

where

$$
\begin{aligned}
Z^{A}_{\mathrm{mnEFIE}} &= \frac{j\omega\mu_0}{4\pi}\iint_{S_m} dS \\
&\quad \cdot \mathbf{f}_m(\mathbf{r}) \iint_{S_n} dS'\, G(\mathbf{r},\mathbf{r}')\mathbf{f}_n(\mathbf{r}') \qquad (3)\\
Z^{\phi}_{\mathrm{mnEFIE}} &= \frac{1}{j4\pi\omega\epsilon_0}\iint_{S_m} dS\, \nabla_s \\
&\quad \cdot \mathbf{f}_m(\mathbf{r}) \iint_{S_n} dS' G(\mathbf{r},\mathbf{r}')\nabla_s \cdot \mathbf{f}_n(\mathbf{r}') \qquad (4)\\
V_{m\,\mathrm{EFIE}} &= \iint_{S_m} dS\, \mathbf{f}_m(\mathbf{r}) \cdot \mathbf{E}_i(\mathbf{r}) \qquad (5)
\end{aligned}
$$

with $G(\mathbf{r},\mathbf{r}') = e^{-jk_0|\mathbf{r}-\mathbf{r}'|}/|\mathbf{r}-\mathbf{r}'|$, $k_0 = \omega\sqrt{\epsilon_0\mu_0}$, S_m and S_n correspond to the supports of the functions $\mathbf{f}_m$ and $\mathbf{f}_n$, respectively, and $\mathbf{E}_i$ to the incident electric field. In (2), we have explicitly denoted the contributions of the scalar (ϕ) and vector (A) potentials.

The same discretization strategy is applied to the MFIE, yielding

$$
\begin{aligned}
Z_{\mathrm{mnMFIE}} &= \frac{1}{2}\iint_{S_m} dS\, \mathbf{f}_m(\mathbf{r}) \cdot \mathbf{f}_n(\mathbf{r}) \\
&\quad + \frac{1}{4\pi}\iint_{S_m} dS\, \mathbf{f}_m(\mathbf{r}) \\
&\quad \times \hat{n} \iint_{S_n} dS'\, \mathbf{f}_n(\mathbf{r}') \times \nabla G(\mathbf{r},\mathbf{r}') \qquad (6)\\
V_{m\,\mathrm{MFIE}} &= \iint_{S_m} dS\, \mathbf{f}_m(\mathbf{r}) \cdot \hat{n} \times \mathbf{H}_i(\mathbf{r}) \qquad (7)
\end{aligned}
$$

where $\hat{n}$ is the outward unit normal to the surface, and $\mathbf{H}_i$ is the incident magnetic field. The CFIE is then expressed as a linear combination of EFIE and MFIE as follows:

$$
\begin{aligned}
&(\alpha Z_{\mathrm{EFIE}} + (1-\alpha) Z_{\mathrm{MFIE}}) I \\
&\quad = (\alpha V_{\mathrm{EFIE}} + (1-\alpha) V_{\mathrm{MFIE}}) \qquad (8)
\end{aligned}
$$

where $0 < \alpha < 1$ is the weight controlling the contribution of the EFIE and MFIE operators. In this paper, we use Z as the MoM matrix, defined as $Z = \alpha Z_{\mathrm{EFIE}} + (1-\alpha) Z_{\mathrm{MFIE}}$.

III. NSS

In this section the main stages of the proposed scheme, NSS are described. First, the matrix structure is summarized in Section III-A; then the compression scheme based on field sampling is elaborated in Section III-B. The memory complexity scaling is described in Section III-C.

A. Fast Solver Matrix Structure

The NSS starts clustering the basis functions in different groups in $L+1$ levels (e.g., using an Octree with root level equal to zero), where L is the leafs' index. For each group, the far field is defined as the groups complying with the admissibility condition [9]

$$\mathrm{diam}(\tau') \leq \eta \cdot \mathrm{dist}(\tau', \tau) \qquad (9)$$

where τ' and τ are two groups diam stands for the size of the group and dist is the distance between the centroids of two groups; η is a constant usually around 1 [26]. Here, the clustering is performed via an Octree, therefore, the diam of groups τ' and τ is equal.

Once this formal description is given, the proposed solver uses the classical definition for the far field used in the FMM, where, for each group, all the "touching" groups and the group itself are defined as the near field (e.g., $\eta = 1.1$), while the rest of the groups in the cluster tree are considered in the far field. The clustering of the geometry induces a tessellation of the matrix, while the admissibility condition split the contributions in near and far field (low rank). NSS computes the near-field matrix at the leafs' level (level L) $Z^{(L)}_{\mathrm{Near}}$, then each group of functions at level L is compressed using the proxy fields scheme (explained in Section III-B). The compression of each group generates three outputs: an anterpolation matrix ${V^{(L)}}^{\mathrm{T}}$, an interpolation matrix $U^{(L)}$ (these are the transpose of each other due to reciprocity), and a set of the principal radiators/receivers in the group (e.g., a list containing the indices of the dominant basis functions). Notice that the matrix names interpolation and anterpolation are used for notation purposes only.

The far-field interaction between two groups (e.g., τ and τ') that have been compressed is described in terms of the factorization

$$Z_{\tau,\tau'}{}^{(L)} \approx U_{\tau}{}^{(L)} \tilde{Z}^{(L)}_{\tau,\tau'} V_{\tau'}{}^{(L)\mathrm{T}} \qquad (10)$$

where $Z_{\tau,\tau'}{}^{(L)}$ is the low rank interaction matrix between groups τ and τ' at level L; the anterpolation and interpolation matrices are ${V^{(L)}_{\tau'}}^{\mathrm{T}}$ and $U_{\tau}{}^{(L)}$, respectively. Notice that $U_{\tau}{}^{(L)}$ depends on τ only, while ${V^{(L)}_{\tau'}}^{\mathrm{T}}$ depends on τ' only: this is one of the characteristics of the $\mathcal{H}^2$ structure. The matrix $\tilde{Z}^{(L)}_{\tau,\tau'}$ contains the interactions between the principal functions (e.g., is a submatrix of $Z_{\tau,\tau'}{}^{(L)}$). In MATLAB notation $\tilde{Z}^{(L)}_{\tau,\tau'} = Z_{\tau,\tau'}{}^{(L)}(I_s, J_s)$, where I_s is the set of indices obtained from the compression of group τ, while J_s is the set of indices obtained from the compression of group τ'; these indices are usually known as *skeletons*. As in all low-rank factorizations, the number of skeletons is much lower than the size of the groups; details on the low rank factorization are given in Section III-B.

Once all the low rank factorizations are computed (10), the structure of the MoM matrix Z can be written as

$$Z \approx U^{(L)} \tilde{Z}^{(L-1)} {V^{(L)}}^{\mathrm{T}} + Z^{(L)}_{\mathrm{Near}} \qquad (11)$$

that describes the leafs' level compression of the MoM matrix; $U^{(L)}$ and ${V^{(L)}}^{\mathrm{T}}$ contain the basis matrices spanning the skeletons (as receivers and radiators) for all the far interacting groups; the matrix $\tilde{Z}^{(L-1)}$ contains the interactions between the skeletons at level $L-1$, where we have regrouped the leaf groups into their parents (in level $L-1$), reintroducing rank deficiencies in $\tilde{Z}^{(L)}$. The structured nature of matrix $\tilde{Z}^{(L-1)}$ allows to define a new near field for level $L-1$, called simply $Z^{(L-1)}_{\mathrm{Near}}$, while far interactions in level $L-1$ can be compressed through the same procedure described previously. This is a recursive procedure described by

$$\tilde{Z}^{(L-i)} \approx U^{(L-i)} \tilde{Z}^{(L-i-1)} V^{(L-i)^{\mathrm{T}}} + Z_{\mathrm{Near}}^{(L-i)} \tag{12}$$

where $i = 0, \ldots, L-2$ for $L \geq 2$. It is remarked that matrices $\tilde{Z}^{(L-i-1)}$ are never computed except for $i = L-2$ (e.g., the first Octree subdivision); these matrices are used here just for illustrative purposes of the method (we use the indices obtained from the skeletonization in each level only).

Regarding the nature of matrices $Z_{\mathrm{Near}}^{(L-i)}$ in (12), we point out that, although they are considered full rank in level $(L-i)$, they are low rank in level $(L-i+1)$; this simple observation allows to further compress these matrices. Indeed, going one level down (e.g., from $(L-i)$ to $(L-i+1)$), it can be seen that the interaction blocks are rank deficient ($\mathcal{R}(k)$ with rank k, in hierarchical matrices terminology [26]). It is important to highlight that this optional step is not necessary in any way for obtaining the linear complexity; it just decreases the necessary memory of the method, keeping the complexity unchanged. Notice that $Z_{\mathrm{Near}}^{(L-i)}$ is sparse with a few $\mathcal{R}(k)$ blocks, so they are suitable for a simple block-by-block compression; the ACA [8] with the recompression proposed in [26] is used here for this step.

Combining (11) and (12), we obtain the generalized structure of the NSS as

$$\begin{aligned} Z \approx\ & U^{(L)} \{ U^{(L-1)} [\ldots (U^{(2)} \tilde{Z}^{(1)} V^{(2)^{\mathrm{T}}} + Z_{\mathrm{Near}}^{(2)}) \ldots] \\ & V^{(L-1)^{\mathrm{T}}} + Z_{\mathrm{Near}}^{(L-1)} \} V^{(L)^{\mathrm{T}}} + Z_{\mathrm{Near}}^{(L)}. \end{aligned} \tag{13}$$

The recursion for the compression stops at $\tilde{Z}^{(2)} = U^{(2)} \tilde{Z}^{(1)} V^{(2)^{\mathrm{T}}} + Z_{\mathrm{Near}}^{(2)}$ as in level 1, no group is admissible according to the admissibility condition in (9). The nestedness of the method resides in the fact that the hierarchical structure is constructed from the leafs information only.

The previous concepts are elucidated with a simple example in Fig. 1(a), where a 1-D geometry is used for ease to illustrate the key points; nonetheless, it is highlighted that our solver works for general 3-D cases as will be demonstrated in the numerical results section. Fig. 1(a) shows the 1-D discretization (in triangles); the different groups are numbered according to the tree presented in Fig. 1(b).

The solver starts at the leafs' level [$L = 3$, last level in the tree in Fig. 1(b)], where for each group, the near-field interactions are computed and stored in the near-field matrix [$Z_{\mathrm{Near}}^{(3)}$ in Fig. 1(c)], while the far-field interactions are compressed through the local compression explained in the next section. The far-field interactions for each group are factorized both as receiver and radiator in order to construct the basis matrices $U^{(3)}$ and $V^{(3)^{\mathrm{T}}}$, respectively, and a set of indices that represent the dominant basis/testing functions selected through the skeletonization; the dominant indices interactions are represented by the matrix $\tilde{Z}^{(2)}$, although this matrix is never build explicitly. In the next coarser level $L-1$ [e.g., level 2 in Fig. 1(b)], the dominant interactions that survived at level L are regrouped and recompressed through the same compression scheme used at level L.

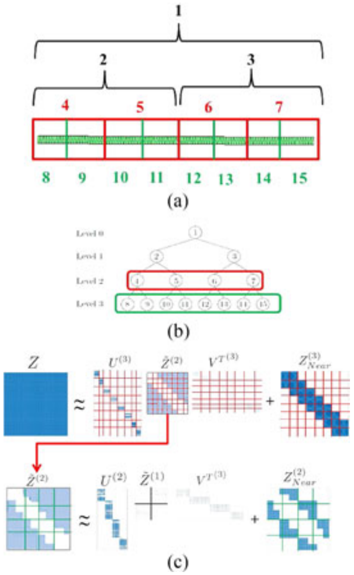

Fig. 1. 1-D geometry example. (a) 1-D geometry clustering. (b) Cluster tree. (c) Matrix tessellation.

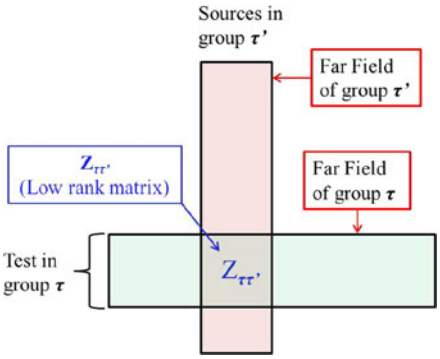

Fig. 2. Far-field signature of a given group by computing the entire far field.

B. Local Compression Scheme

For the sake of simplicity, in this subsection, the terms *source* or *test* will be used referring to vector functions (e.g., the RWG) that are either radiating or receiving; therefore, when referring to blocks of the matrix, block row is the block of the MoM matrix containing the interactions between a test group and its far field, a block column is the block of the MoM matrix containing the interactions between a source group and its far field.

The compression in this paper uses the idea of proxy surfaces [10] where, instead of building the entire block row and block column of a group in order to compress it (as shown in Fig. 2), only the interactions between the sources (test) of the group and a set of test (sources) auxiliary functions is computed. By doing this, the computation is now local, consequently, bounded by the number of sources (test) in the group. Up-to-date schemes working with proxy surfaces first discretize the proxy that is generally a sphere or a cube enclosing the group [15], [23]. Instead, here, we use samples of the field scattered by the group being compressed in the artificial sphere around it. In practice,

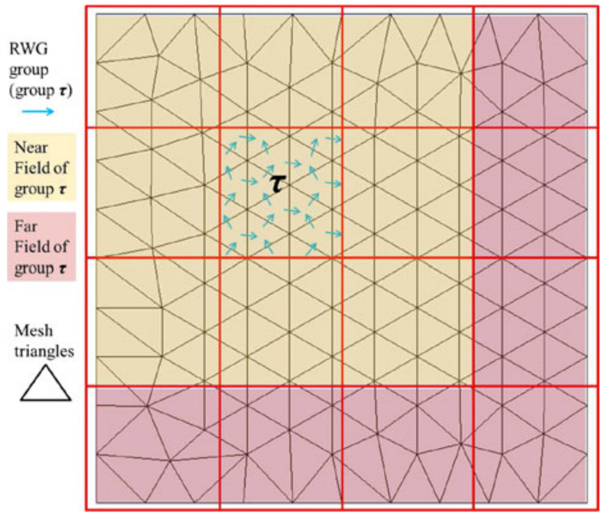

Fig. 3. Near- and far-field zones for group τ.

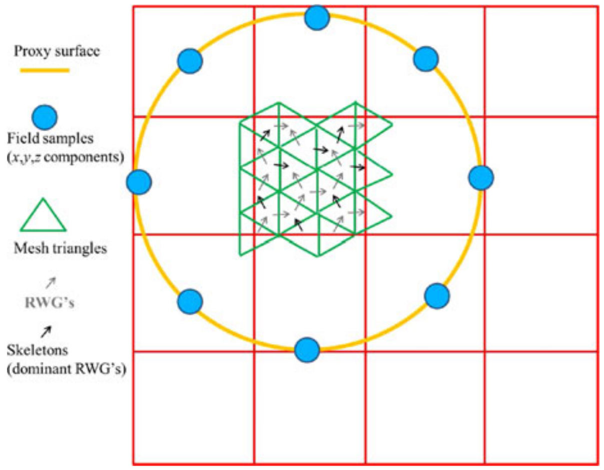

Fig. 4. Skeletonization using proxy field sampling for group τ.

we use the routines that compute the scattered field (electric field, the magnetic field can be used as well); the field is obtained using the mixed potentials formulation used for the MoM interactions [e.g., for the electric field, we use (3) and (4)], but instead of a Galerkin testing, we use vector Dirac deltas for sampling each of the components of the field in each of the sampling points. Fig. 3 shows graphically the near and far parts for a given group (group τ), where the boundaries of the tree are highlighted in order to ease the comprehension.

The NSS scheme generates Q points (set according to the desired accuracy, see Section IV ahead) around each group, then the electric field radiated by the RWG's in the group is sampled at those points (three components for each of the Q points); this procedure generates a matrix Z_S, that is compressed using the interpolative decomposition (ID) [27]. The geometrical interpretation of the scheme is depicted in Fig. 4. Fig. 5 presents the matrix version of the factorization of a far-field group using the field sampling compression; each row of matrix Z_S contains one component of the field due to the contributions of all the sources in the group; therefore, Z_S have $3Q$ rows.

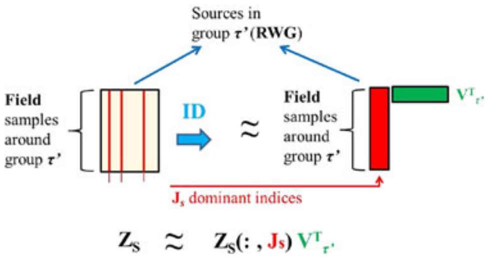
Sources in group τ'(RWG)
Field samples around group τ'
ID
≈
Field samples around group τ'
V^T_τ'
J_s dominant indices
Z_S ≈ Z_S(: , J_S) V^T_τ'

Fig. 5. Construction of anterpolation matrices via interpolative decomposition.

The electric (magnetic can be used as well) field, scattered or radiated by generic vector functions $\mathbf{f}_n(\mathbf{r}')$ sampled at points $\mathbf{r}_m$ can be written as

$$Z_{S_{\mathrm{m\,n}}} = -\frac{j\omega\mu_0}{4\pi}\iint_{S_n} dS'\hat{\gamma}\cdot\left[(\underline{\underline{I}}+\frac{\nabla\nabla}{k^2})G(\mathbf{r}_m,\mathbf{r}')\right]\cdot\mathbf{f}_n(\mathbf{r}') \quad (14)$$

where we have used $\hat{\gamma}$ for indicating either $\hat{x}$, $\hat{y}$, or $\hat{z}$, while $\mathbf{r}_m$ is the position vector of the *m*th proxy point where the field component is sampled. Each point where the field is sampled produce then three values (one for each component of the field). The ID takes as input the low rank matrix Z_S and a user-defined tolerance, then it outputs the rank of the matrix, a set of the dominant indices, J_s (that corresponds to the skeletons, defined as the vector basis functions $\mathbf{f}_{J_s}$), and a basis matrix ${V_{\tau'}}^{\mathrm{T}}$ for a generic group τ'. The matrix ${V_{\tau'}}^{\mathrm{T}}$ maps (anterpolates) the sources $\mathbf{f}(\mathbf{r}')_n$ contributions to a reduced set of dominant ones, J_s, after which this reduced set is able to radiate any field (accurate up to the user given tolerance) in the far field of the group. The outputs of the ID are graphically shown in Fig. 5 (right part). The obtained ${V_{\tau'}}^{\mathrm{T}}$ is also used to find the basis as $U_{\tau'} = V_{\tau'}$.

If we take two far interacting groups, as the ones described in Section III-A (e.g., τ and τ'), we perform the ID to both groups obtaining ${V_{\tau'}}^{\mathrm{T}}$ and J_s for the source group and U_τ and I_s for the test group (10), this is pictorially represented in Fig. 6. Even if $U_\tau = V_\tau$ for any group τ, we will keep the separated representation for the sake of clarity in the discussion. It is also highlighted that for the construction of the anterpolation/interpolation matrices (V and U) that either the electric or magnetic field can be used, because, once the field is matched in the proxy surface, the solutions outside the proxy are unique and so their representations. We point out that in view of the discrete nature of the MoM representation, a relaxation of the requirement of a completely closed surface is prescribed; however, this is common to all proxy-based solvers, and as demonstrated in the numerical results, it does not pose a limitation to the applicability and accuracy of the scheme. As stated in the introduction, the method is entirely kernel independent, as the translations between far groups are represented by the MoM matrix interactions between the skeletons (e.g., through the $Z_{\mathrm{Near}}^{(L-i)}$ with $i = 0, \ldots, L-2$ and $\tilde{Z}^{(1)}$ matrices, as shown in (13)].

C. *Complexity Scaling*

The matrix structure used in this work belongs to the so called $\mathcal{H}^2$ matrices; both the strong admissibility condition and nestedness used are responsible for the linear complexity obtained as

$$C_{\text{memory}} \approx QN + 8Q^2 \log_2(N) + 27\ \text{K}N + 4^d QN + Q^2 \tag{15}$$

where C_{memory} stands for the computational complexity in memory with Q constant sampling points around each group, while N is the total number of unknowns and K is a constant number of basis functions in the near field for each group in the leafs' level. The different contributions are summarized as follows. The QN term is related to the memory used for the interpolation and anterpolation matrices U and V at the leafs' level, while the near-field matrix contribution (near field at leafs' level) is expressed in the $27\ \text{K}N$ term. The $8Q^2 \log_2(N)$ term describes the scaling of U and V at coarser levels (e.g., different from leafs' level); the near-field matrices for coarser levels scale as $4^d QN$, for d the dimension of the clustering algorithm ($d = 3$ for the Octree, $d = 2$ for a Quadtree, etc.). The last contribution Q^2 is related to the coupling matrix $\tilde{Z}^{(1)}$, which is computed in the coarsest level only. It is evident that the linear trend is the dominant one. A detailed discussion of the terms in this complexity estimation in (15) is reported in the Appendix A.

As the MV product is done from right to left, as described in the Appendix B, the MV product complexity follows that of the memory times the number of iterations.

The setup time complexity depends on the ID applied to each of the $Q \cdot n_i$ matrices in order to generate V (n_i being the number of basis functions in the group i). Each ID costs $mn \log k + k^2 n$ [14] for each $m \cdot n$ matrix with rank k; in our analysis, this becomes $n_i Q \log Q + Q^3$ using our maximum bound for the rank. In the leafs' level, adding the cost of all groups, this cost becomes $NQ \log Q + Q^3$. The previous is the upper bound for the cost of the ID, as in the coarser levels (e.g., after the skeletonization is made in the leafs' level), the cost is bounded by applying the ID to $Q \cdot Q = Q^2$ matrices.

Finally, the time complexity for all the near-field matrices ($Z_{\text{Near}}^{(l)}$ for $l = 2, \ldots, L$) and the coupling matrix $\tilde{Z}^{(1)}$ follows the memory trend in (15) (the time is proportional to the number of elements of these sparse matrices).

IV. Numerical Results

This section assesses the behavior of the proposed strategy. The accuracy of the proposed field sampling technique is assessed in Sections IV-A and IV-B in the low-frequency regime with respect to the user-defined tolerance. Second, in Section IV-C, we evaluate the numerical complexities of the method; then the intermediate frequency error is determined in Section IV-D, in order to bound properly the limitations of our solver. Finally, realistic multiscale structures are simulated in Sections IV-E and IV-F. In all the tests, the integrals for obtaining the scattered fields are computed with a three-points Gaussian rule [28]. The mean discretization size is referred to as h. Our proposed scheme will be referred to as NSS. All the simulations were performed with an Intel Xeon ES-2670 (2.6 GHz), 64-Bit server with 256 GB of RAM.

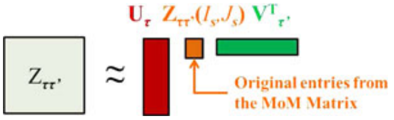
U_τ $Z_{\tau\tau'}(I_s,J_s)$ $V^T_{\tau'}$
$Z_{\tau\tau'}$
≈
Original entries from the MoM Matrix

Fig. 6. Compression of a far-field interaction $Z_{\tau,\tau'}$ by proxy field sampling.

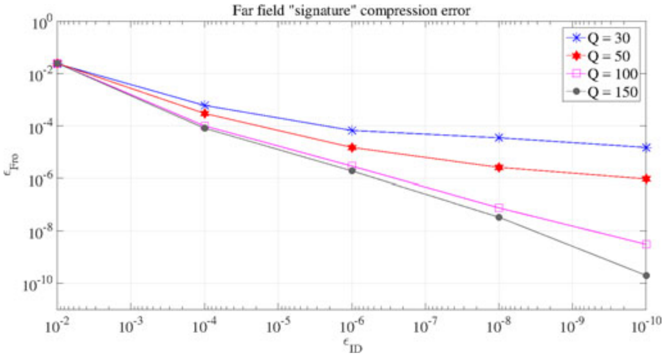

Fig. 7. Far-field signature compression error varying the number of auxiliary test points Q in the proxy surface.

A. *Error Assessment*

In order to assess the accuracy of the proposed field sampling technique, the normalized Frobenius error is computed. A PEC square plate with side 0.1λ is discretized with $h/\lambda = 0.005$, and an Octree with four levels is generated such that there are 64 nonempty groups in the last level. A group in the middle of the plate in the last level is taken as reference; the group's far-field block column (Z_{block}) as shown in Fig. 2 is fully assembled for reference using a 61-points Gaussian rule for the MoM reaction integrals [28], both for internal and external integrals. Such a high number of integration points intends to separate the error due to the accuracy of the MoM integrals evaluation from the error of the used fast factorization. The group compression is performed as explained in Section III-B using Q sampling points, which will be the varying parameter in this test; both V^T and the set of indices J_s are obtained and finally the normalized Frobenius error is evaluated as

$$\epsilon_{\text{Fro}} = \frac{||Z_{\text{block}} - Z_{\text{block}}(:, J_s) \cdot V^T||}{||Z_{\text{block}}||}. \tag{16}$$

Fig. 7 demonstrates the general trend when Q changes; the almost one-to-one relation between the user's input tolerance ε_{ID} and the effective accuracy obtained allow us to be confident in the obtained results. Once Q is defined using Fig. 7 as a guide, the only parameter, which is user dependent is the compression tolerance itself, being much more simple than the recently proposed solver in [21]. From the rest of the numerical results a Gaussian rule of three points for the exterior integral and seven for the inner integral is used (except for near interactions, where singularity treatments are used as proposed in [29] and [30]), which provides an accuracy of 10^{-4} [21]. The ID and the number of proxy samples rely on the accurate calculation of

TABLE I
ACCURACY AND COMPUTATIONAL DATA FOR THE PEC SPHERE DISCRETIZED WITH 9060 UNKNOWNS, FOR DIFFERENT Q SAMPLING POINTS; $\varepsilon_{\mathrm{ID}}$ IS SET TO $1e-4$

Q	Near/Far-Field Memory [MB]	Far-Field Approximation Time [m:s]	Iterative Solution Time [m:s]	η
30	68.34/346	12:45	01:26	0.0035
50	68.34/383.2	14:00	01:20	0.0021
100	68.34/397	15:11	01:27	0.0018

TABLE II
ACCURACY AND COMPUTATIONAL DATA FOR THE PEC SPHERE DISCRETIZED WITH 9060 UNKNOWNS, FOR DIFFERENT ID TOLERANCES ($\varepsilon_{\mathrm{ID}}$); THE NUMBER OF SAMPLING POINTS Q IS SET TO 50

$\varepsilon_{\mathrm{ID}}$	Near- and Far-Field Memory No Recompression/With Recompression [MB]	Far-Field Approximation Time No Recompression/With Recompression [m:s]	Iterative Solution Time No Recompression/With Recompression [m:s]	η
$1e-2$	68.34 and 79.11/44.86	03:06/02:59	00:29/00:43	0.122
$1e-3$	68.34 and 206.24/107.66	07:42/06:40	01:03/00:58	0.011
$1e-4$	68.34 and 383.2/204.66	14:00/11:40	01:20/01:21	0.0021

the sampling matrix (Z_s). Our implementation follows careful integration rules [29], [30], in order to guarantee the correct sampling of the radiated fields.

B. Tradeoff Between Accuracy and Computational Burden

The tradeoff between accuracy and computational burden for the NSS is assessed. A PEC sphere is used as test case (radius $r = 0.5$ m, frequency 300 MHz); the sphere is discretized using 9060 RWG functions. The l^2 error of the current density is defined as

$$\eta = \frac{||J_{\mathrm{ref}} - J||}{||J_{\mathrm{ref}}||} \tag{17}$$

where J_{ref} is the reference solution (vector of coefficients) obtained with the LU decomposition of the full MoM matrix without approximations; J is the corresponding solution obtained with the NSS. First, we address the accuracy/computational burden versus the number of sampling points Q: A summary of the results is presented in Table I. The solution is obtained using a BiCGStab with a residual set to $1e-6$. A simple diagonal preconditioner is applied in order to minimize its effect on the MV times. From the data in Table I, it is noted that increasing the number of points Q, the MV time remains almost unchanged (the MV here is measured by the iterative solution time, due to the very short duration of each MV); the far-field memory increases just slightly with the number of points. The error of the current remains almost constant.

Next, we keep Q constant to 50 points and we vary the $\varepsilon_{\mathrm{ID}}$: the results are displayed in Table II. The obtained results follow the expected trend for an increasing the accuracy (e.g., decreasing $\varepsilon_{\mathrm{ID}}$) required to the fast solver; recompression/no recompression stands for the use of the ACA compression of the near-field matrices as described in Section III-A. For the rest of the numerical results section Q is set to 50.

Finally, we benchmark our implementation's accuracy using the Mie series; the bistatic radar cross section (RCS) of a PEC sphere with radius 0.5λ is computed with our method (using two different kernels, being the EFIE and the CFIE with $\alpha = 0.8$) and compared to the Mie series. The $\varepsilon_{\mathrm{ID}}$ is set to $1e^{-3}$ and a BiCGStab with a residual of $1e^{-4}$ is used. The results are shown in Fig. 8.

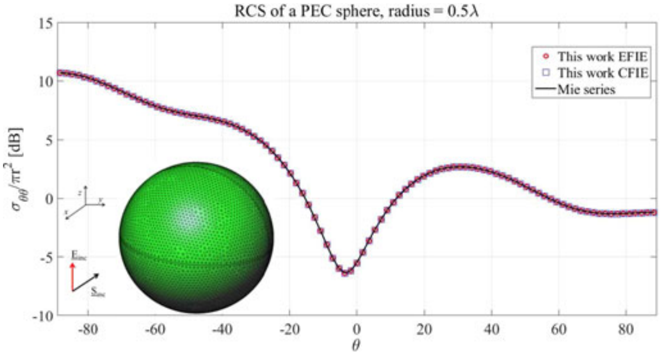

Fig. 8. PEC sphere benchmark.

The excellent agreement between the NSS solution and the Mie series confirm that the proposed method fulfills the standard accuracy requirements for fast solvers; moreover, the kernel-independent nature of the solver is clearly established.

C. Numerical Complexity

In this section, we numerically verify that the NSS is linear in setup time, memory, and MV product time as discussed in Section III-C. The test case is a PEC sphere (radius equal to 0.5 m), the frequency is set to 3 MHz, and the number of unknowns (N) is increased by increasing the mesh density (decreasing the h parameter). The $\varepsilon_{\mathrm{ID}}$ is set to $1e^{-3}$. Four cases are tested (10 464, 39 696, 155529, 642936 unknowns); this is usually known as fixed discretization complexity. The MoM matrix setup time complexity is shown in Fig. 9, the memory complexity in Fig. 10, and finally, the MV product time complexity is in Fig. 11.

The linear scaling is clear in all the three computational complexities, demonstrating the feasibility of the proposed method for the analysis of large 3-D surface MoM computations.

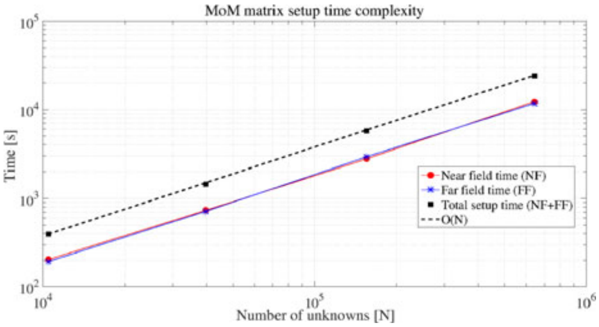

Fig. 9. MoM matrix setup complexity.

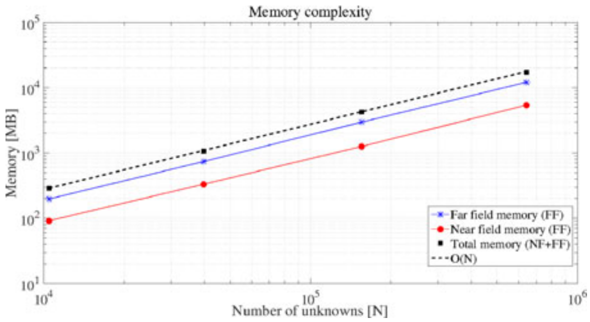

Fig. 10. MoM matrix memory complexity.

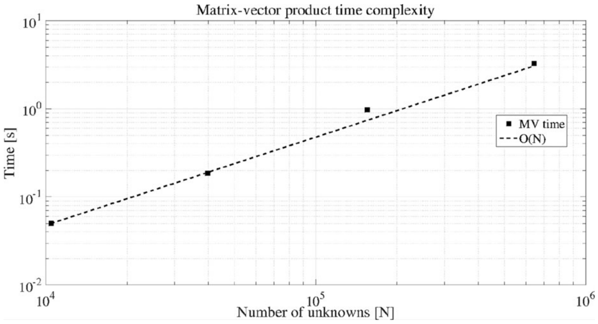

Fig. 11. MoM MV product time complexity.

D. *Intermediate Frequency Limit*

The NSS is well suited for simulations from very low frequencies to intermediate frequencies (as demonstrated in the multiscale cases in Sections IV-E and IV-F); the purpose of this section is to assess the intermediate frequency limit. Nonetheless it is important to remark that by modifying the compression stages by including directional rank properties [31], [32], the solver can be extended to work in high-frequency regimes, but this is outside the scope of this paper. A square PEC plate (0.3 m side) is discretized ($h = 0.015$ m); a four levels Octree is generated (e.g., leafs' level with 64 groups), with each group having a box side of 0.0375 m and 543 RWG. The frequency is varied from 1 to 16 GHz (the box side in λ grows from $\lambda/8$ to 2λ); the $\varepsilon_{\mathrm{ID}}$ tolerance is fixed to $1e^{-3}$. The test is repeated for different number of proxy field sampling points (Q). The test evaluates the rank obtained at each frequency, as reported in Fig. 12.

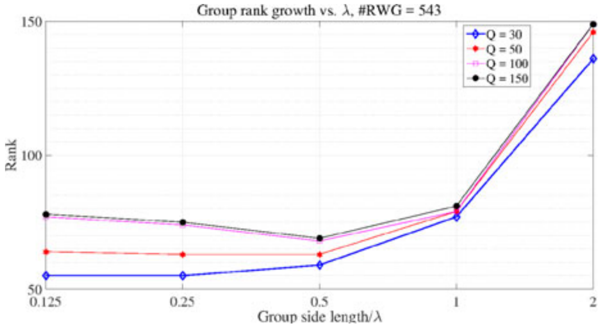

Fig. 12. Rank versus λ.

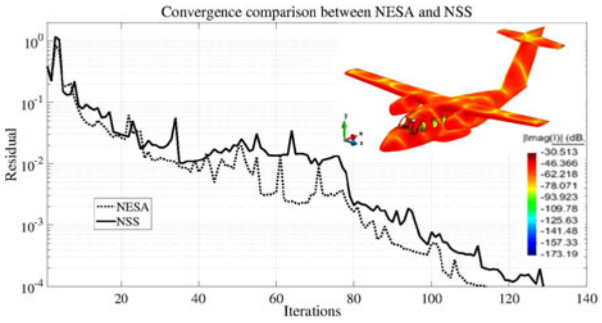

Fig. 13. Comparison of the convergences of NESA and NSS.

TABLE III
MORPHED EV-55 AIRCRAFT, $\varepsilon_{\mathrm{ID}} = 1e-4$, $Q = 100$

Solver	Near- / Far-Field Memory [MB]	MoM Matrix Setup Time [m:s]	MV Product [s]
NESA	3500 / 2300	55:56	7.8
NSS	3774 / 8643	49:35	2.8

From Fig. 12, it is noticed that the rank growth explodes for groups sizes larger than 1λ; therefore, we can state that the solver presented in this paper deals efficiently with structures of up to 4 λ in size (as for the boxes in the coarsest level using an Octree, where the admissibility condition is fulfilled, have 1λ in length).

E. *Real-Life Multiscale Test Case at Intermediate Frequency*

The multiscale capabilities of the proposed scheme are tested using a mockup version of the Evektor EV-55 airplane[1] (seen in the insert in Fig. 13), discretized with 171.763 unknowns. A 100-MHz plane wave impinges on the structure from the $\theta = 90^\circ$, $\phi = 225^\circ$ direction; the wing span of the airplane is 5.3 λ at the simulation frequency and the discretization parameter h/λ varies as $1.6e-3 \leq h/\lambda \leq 2.8e-2$. Our scheme is then compared to the one proposed in [21], in the following indicated as *NESA*, in Table III.

For NESA, 100 *equivalent sources* are used, $\varepsilon_{\mathrm{SVD}} = 1e-12$, $\varepsilon_{\mathrm{ACA}} = 1e-4$ and $R_\tau/R_0 = 1$, which should provide an accuracy $\sim 1e-4$ according to the tables presented in [21]. For the NSS, we use $Q = 100$ and $\varepsilon_{\mathrm{ID}} = 1e-4$, setting less user-

[1] [Online]. Available: http://www.evektoraircraft.com/en/aircraft/ev-55-outback/overview

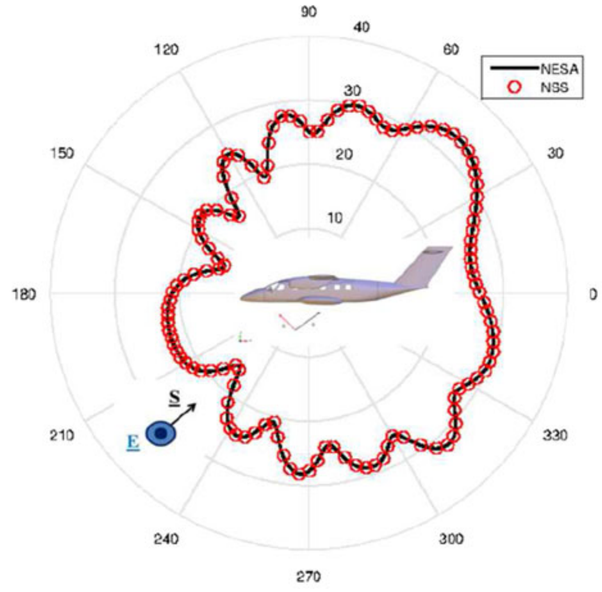

Fig. 14. Comparison between the RCS of NESA and NSS.

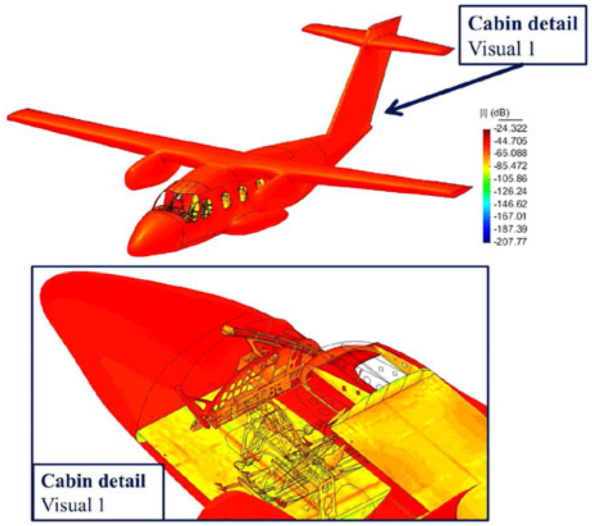

Fig. 15. Morphed ev55 current density at 300 kHz

defined parameters than NESA as previously discussed. Both solutions are obtained with a BiCGStab with a relative residual of $1e-4$; both solvers are preconditioned with the MR + ILU preconditioner [33].

Table III shows a decrease of the MV product time (Column 4) of the NSS respect to the NESA solver of a factor of almost 3. This is due to the fact that NESA needs to compute the interactions between equivalent sources in far interactions when doing the MV, which depends on the number of equivalent sources (the higher the desired accuracy the higher the MV time). NSS instead performs the coupling between far groups using the compressed representation, involving the numerical ranks only (which are always lower than the maximum bound given by the number of field sampling points Q).

On the memory side, however, there is an increase on the far-field memory of the NSS respect to the NESA solver (Column 2). This is because NESA communicates between far groups by using the equivalent sources defined around them, this is uniquely defined for each level, therefore, a single set of equivalent sources is used, reducing the memory required. The NSS instead, transfers this information between far groups by using interactions between skeletons rather than equivalent sources, in this way, the necessary memory is higher. Nevertheless this increase in memory is not directly reflected in the MoM matrix setup times (Column 3) that are instead comparable (this is of course implementation dependent). Finally, the iterations count for both solvers is shown in Fig. 13, where the imaginary part of the current density with the NSS is shown as well; the bistatic radar cross section (RCS) in dB is computed for both approaches and compared in Fig. 14.

F. Real-Life Multiscale Tests Case at Low Frequency

In this section, the multiscale capabilities at low frequencies and dense meshes of the solver are demonstrated simulating complex, realistic structures. We use the same mockup version of the Evektor EV-55 airplane of the previous subsection with a denser discretization (the number of unknowns is 2.799.662); the simulation frequency is set to 300 kHz (the aircraft's wingspan is 0.016λ); in the following, the Q sampling points are set to 50, while the ε_{ID} is set to $5e^{-3}$. A BiCGStab is used for the iterative solution with a residual set to $1e^{-4}$. The used preconditioner is the multiresolution preconditioner [34]; the low-frequency stabilization in [35] is used for dealing with the convergence problems caused at very low frequencies with dense discretizations. Fig. 15 shows the current density obtained; a detail on the realistic nature of the model is presented for illustration purposes.

This test case demonstrates the ability of our solver to deal with large structures; in particular, this structure uses 155 GB of memory, while the MV product takes 45 s. The target residual is reached after 197 iterations.

V. Conclusion and Perspectives

A fully characterized fast compressive solver was presented. The linear complexities obtained allow us to claim that this scheme belongs to the state of the art kernel-independent fast solvers. The extension of the HSS structure with strong admissibility is demonstrated to be a suitable option, which is simpler than other kernel-independent solvers. Moreover the presented scheme was analyzed and discussed in a comprehensive manner and its advantages demonstrated numerically.

Further improvements that are being considered are the extension to high-frequency analysis by including directional properties in the compression stage. In addition, the developed matrix structure can be used to develop a fast direct solver.

Appendix

A. Complexity

We start out this sketch by analyzing the memory trends of the scheme that results in the complexity in (15). We recall that Q is the number of sampling points and N is the number of unknowns. Q is used as the upper bound for the numerical rank of the far-field interactions for low to intermediate frequencies, which is a well-known fact [19], [22]. Anterpolation and interpolation matrices V and U are the same, hence, we refer only to V in order to facilitate the discussion, also constants, which are of no use will be overlooked for clarity in the presentation.

At the leafs' level L, each group i has n_i basis functions, $i = 1, 2, \ldots, N_g$ with N_g number of groups at the leafs' level, therefore, anterpolation matrices ${V_i^{(L)}}^{\mathrm{T}}$ contain $n_i \cdot Q$ elements. Adding up all the N_g groups, at the leafs' level, results in $\sum_{i=1}^{N_g} Q \cdot n_i = Q \cdot N$.

In the next coarser levels ($l = L-1, \ldots, 2$), the memory of the matrices ${V^{(l)}}^{\mathrm{T}}$ is bounded by $Q \cdot Q = Q^2$ times the number of children of each group, that is constant and equal to 2^d (where $d = 3$ for an Octree), times the number of levels L, which scales as $\log_2(N)$; this results in $8Q^2 \cdot \log_2(N)$.

For each group at the leafs' level, we define a near-field made of the neighbors and self-terms; this near field is bounded by 27 groups (for an Octree), each of which contains a constant number of functions K so that each group i will have 27 K functions to interact with (MoM blocks of 27 $\mathrm{K} \cdot n_i$). Adding all groups contributions, we end up with 27 $\mathrm{K} \cdot N$.

In the coarser levels ($l = L-1, \ldots, 2$), the near field comes from an skeletonization performed in the previous child level; we then regroup the children of each group in this level (level l) but we keep constant our upper bound Q. This scales as $2^d Q$. The number of groups decreases as $N_g/2^{d(L-l)}$ (where N_g is the number of groups at the leafs' level). These near-field matrices then scale as $2^d \cdot Q N_g / 2^{d(L-l)}$, where Q is constant and N_g grows as N. We concentrate now in the term $2^d/2^{d(L-l)}$. Adding throughout the levels and simplifying, we obtain $2^d \sum_l (2^{(l-L)})^d$.

Our interest lies on the trend of the summation regarding the $(l-L)$ terms as d is a constant, and therefore, we tackle the term $\sum_{l=1}^{L} 2^{(l-L)}$. Notice that we have assumed the sum to go up to level $l = 1$, though in practice we sum up to level $l = 2$, this does not constitute any change in terms of the trend, but it allows us to obtain a closed form for the series. The series can be written in a geometric series form by a change of variables $n = l - L$, so the series becomes $\sum_{n=1-L}^{0} 2^n$. Then, the sum is recast in

$$\sum_{n=1-L}^{0} 2^n = \sum_{n=0}^{L-1} (1/2)^n = \frac{1-(1/2)^L}{1-1/2}. \qquad (18)$$

The term $(1/2)^L$ tends to N^{-1}, which goes to zero for growing N, therefore the series converge to a constant value of 2.

We conclude that the scaling for the near-field matrices at coarser levels is $4^d \cdot Q \cdot N$.

Finally, the coupling matrix $\tilde{Z}$ is computed at level $l = 1$ only, scaling as a constant number of groups (e.g., 8) times Q^2. Adding the cost of all the steps involved in the construction of the compressed MoM matrix results in (15).

B. MV Product

In this section, the implementation of the MV product is described in details; this description is strictly related to the matrix structure in (13). For the sake of clarity, the input vector will be called q, while the output vector is referred to as u [e.g., $u = Zq$, with Z the MoM matrix in (13)]. Any given subset of functions of a vector is written directly using the subscript of the given group (e.g., q_τ are the functions of q that belong to group τ); vectors with $\wedge$ indicate intermediate results at different levels. The algorithm uses the Octree data structure from level 1 (eight groups) to level L (leafs' level).

Algorithm 1: MV Algorithm.

```
At level $l = L$ (leafs' level)
for all leaf groups $\tau$ do
    $\hat{q}_\tau^{(l)} \leftarrow {V^{(l)}}_\tau^{\mathrm{T}} q_\tau$
end for
for all levels, finer to coarser, $l = L-1, L-2, \ldots, 2$ do
    for all parent groups $\tau$ on level $l$ do
        Let $\sigma_1, \sigma_2, \ldots \sigma_8$ be the children of group $\tau$
        $\hat{q}_\tau^{(l)} \leftarrow {V^{(l)}}_\tau^{\mathrm{T}} \begin{bmatrix} \hat{q}_{\sigma_1} \\ \hat{q}_{\sigma_2} \\ \vdots \\ \hat{q}_{\sigma_8} \end{bmatrix}^{(l)}$
    end for
end for
At level $l = 1$ couple the skeletons that survived
throughout the finer level skeletonizations
Let $\sigma_1, \sigma_2, \ldots \sigma_8$ be the children of group $\tau$
$\begin{bmatrix} \hat{u}_{\sigma_1} \\ \hat{u}_{\sigma_2} \\ \vdots \\ \hat{u}_{\sigma_8} \end{bmatrix}^{(l)} \leftarrow \tilde{Z}^{(l)} \begin{bmatrix} \hat{q}_{\sigma_1} \\ \hat{q}_{\sigma_2} \\ \vdots \\ \hat{q}_{\sigma_8} \end{bmatrix}^{(l)}$
for all levels, coarser to finer, $l = 2, 3, \ldots, L-1$ do
    for all parent groups $\tau$ on level $l$ do
        Let $\sigma_1, \sigma_2, \ldots \sigma_8$ be the children of group $\tau$
        $\begin{bmatrix} \hat{u}_{\sigma_1} \\ \hat{u}_{\sigma_2} \\ \vdots \\ \hat{u}_{\sigma_8} \end{bmatrix}^{(l)} \leftarrow U_\tau^{(l)} \hat{u}_\tau^{(l)} + Z_{\text{Near}}^{(l)} \begin{bmatrix} \hat{q}_{\sigma_1} \\ \hat{q}_{\sigma_2} \\ \vdots \\ \hat{q}_{\sigma_8} \end{bmatrix}^{(l)}$
    end for
end for
At level $l = L$ couple the near field at leafs' level with the
far field
for all leaf groups $\tau$ do
    $u_\tau \leftarrow U_\tau^{(l)} \hat{u}_\tau^{(l)} + Z_{\text{Near}}^{(l)} q_\tau$
end for
End MV Algorithm
```

Acknowledgment

The authors would like to thank Evektor for providing the model of the (morphed) EV-55 aircraft.

References

[1] S. M. Rao, D. R. Wilton, and A. W. Glisson, "Electromagnetic scattering by surfaces of arbitrary shape," *IEEE Trans. Antennas Propag.*, vol. AP-30, no. 3, pp. 409–418, May 1982.

[2] W. C. Chew, J. M. Jin, E. Michielssen, and J. M. Song, *Fast and Efficient Algorithms in Computational Electromagnetics*. Boston, MA, USA: Artech House, 2001.

[3] V. Rokhlin, "Rapid solution of integral equations of classical potential theory," *J. Comput. Phys.*, vol. 60, no. 2, pp. 187207, Sep. 1985.

[4] J. M. Song and W. C. Chew, "Multilevel fast multipole algorithm for solving combined field integral equation of electromagnetic scattering," *Microw. Opt. Technol. Lett.*, vol. 10, pp. 14–19, Sep. 1995.

[5] L. Greengard and V. Rokhlin, "A fast algorithm for particle simulations," *J. Comput. Phys.*, vol. 73, pp. 325–348, Dec. 1987.

[6] J. Song, C. Lu, and W. 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.

[7] W. Hackbusch, "A sparse matrix arithmetic based on $\mathcal{H}$-matrices," *Computing*, vol. 62, pp. 89–108, 1999.

[8] M. Bebendorf, "Approximation of boundary element matrices," *Numerische Mathematik*, vol. 86, pp. 565–589, 2000.

[9] W. Hackbusch, B. Khoromskij, and S. Sauter, On $\mathcal{H}^2$-Matrices (Lectures Appl. Math.). H. Bungartz, R.H.W. Hoppe, and C. Zenger, Eds. Berlin, Germany: Springer, pp. 9–29, 2000.

[10] P. G. Martinsson and V. Rokhlin, "A fast direct solver for integral equations in two dimensions," *J. Comput. Phys.*, no. 205, pp. 1–23, 2005.

[11] E. Michielssen, A. Boag, and W. C. Chew, "Scattering from elongated objects: Direct solution in O(N log2 N) operations" *Proc. IEE*, vol. 143, no. 4, pp. 277–283, Aug. 1996.

[12] A. Gillman, P. Young, and P. G. Martinsson, "A direct solver with o(n) complexity for integral equations on one-dimensional domains," *Front. Math., China*, vol. 7, pp. 217–247, 2012.

[13] L. Greengard, D. G. P. G. Martinsson, and V. Rokhlin, "Fast direct solvers for integral equations in complex three-dimensional domains," *Acta Numer.*, no. 18, pp. 243–275, 2009.

[14] K. L. Ho and L. Greengard, "A fast direct solver for structured linear systems by recursive skeletonization," *SIAM J. Sci. Stat. Comput.*, vol. 34, pp. 2507–2532, 2012.

[15] J. G. Wei, Z. Peng, and J. F. Lee, "A fast direct matrix solver for surface integral equation methods for electromagnetic wave scattering from non-penetrable targets," *Radio Sci.*, vol. 47, pp. 1–9, 2012.

[16] W. Chai and D. Jiao, "An $\mathcal{H}^2$ matrix-based integral equation solver of reduced complexity and controlled accuracy for solving electrodynamic problems," *IEEE Trans. Antennas Propag.*, vol. 57, no. 10, pp. 3147–3159, Oct. 2009.

[17] W. Chai and D. Jiao, "Linear-complexity direct and iterative integral equation solvers accelerated by a new rank-minimized $\mathcal{H}^2$ representation for large-scale 3-D interconnect extraction," *IEEE Trans. Microw. Theory Techn.*, vol. 61, no. 8, pp. 2792–2805, Aug. 2013.

[18] S. Omar and D. Jiao, "Linear complexity direct volume integral equation solver for full-wave 3-D circuit extraction in inhomogeneous materials" *IEEE Trans. Microw. Theory Techn.*, vol. 63, no. 3, pp. 897–912, Mar. 2015.

[19] K. Zhao, M. N. Vouvakis, and J. F. Lee, "The adaptive cross approximation algorithm for accelerated method of moments of EMC problems," *IEEE Trans. Electromagn. Compat.*, vol. 47, no. 4, pp. 763–773, Nov. 2005.

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

[21] M. Li, M. A. Francavilla, F. Vipiana, G. Vecchi, and R. Chen, "Nested equivalent source approximation for the modeling of multiscale structures," *IEEE Trans. Antennas Propag.*, vol. 62, no. 7, pp. 3664–3678, Jul. 2014.

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

[23] X. Pan and X. Sheng, "Hierarchical interpolative decomposition multilevel fast multipole algorithm for dynamic electromagnetic simulations," *Progress Electromagn. Res.*, vol. 134, pp. 79–94, 2013.

[24] X. M. Pan and X. Q. Sheng, "Preconditioning technique in the interpolative decomposition multilevel fast multipole algorithm" *IEEE Trans. Antennas Propag.*, vol. 61, no. 6, pp. 3373–3377, Mar. 2013.

[25] P. G. Martinsson and V. Rokhlin, "An accelerated kernel-independent fast multipole method in one dimension," *SIAM J. Sci. Comput.*, vol. 29, no. 3, pp. 1160–1178, 2007.

[26] L. Grasedyck and W. Hackbusch, "Construction and arithmetics of $\mathcal{H}$-matrices," *Computing*, vol. 70, pp. 295–334, 2003.

[27] H. Cheng, Z. Gimbutas, P. G. Martinsson, and V. Rokhlin, "On the compression of low rank matrices," *SIAM J. Sci. Comput.*, vol. 26, pp. 1389–1404, Mar. 2005.

[28] A. H. Stroud, *Approximate Calculation of Multiple Integrals*. Englewood Cliffs, NJ, USA: Prentice-Hall, 1971.

[29] M. A. Khayat, D. R. Wilton, and P. W. Fink, "An improved transformation and optimized sampling scheme for the numerical evaluation of singular and near-singular potentials," *IEEE Antennas Wireless Propag. Lett.*, vol. 7, pp. 377–380, 2008.

[30] F. Vipiana and D. R. Wilton, "Numerical evaluation via singularity cancellation schemes of near-singular integrals involving the gradient of Helmholtz-type potentials," *IEEE Trans. Antennas Propag.*, vol. 61, no. 3, pp. 1255–1265, Mar. 2013.

[31] B. Enquist and L. Ying, "Fast directional multilevel algorithms for oscillatory kernels," *SIAM J. Sci. Stat. Comput.*, vol. 29, pp. 1710–1737, Aug. 2007.

[32] M. Li, M. A. Francavilla, R. Chen, and G. Vecchi, "Wideband fast kernel-independent modeling of large multiscale structures via nested equivalent source approximation," *IEEE Trans. Antennas Propag.*, vol. 63, no. 5, pp. 2122–2134, May 2015.

[33] F. Vipiana, M. A. Francavilla, and G. Vecchi, "EFIE modeling of high-definition multi-scale structures," *IEEE Trans. Antennas Propag.*, vol. 58, no. 7, pp. 2362–2374, Jul. 2010.

[34] F. Vipiana, G. Vecchi, and P. Pirinoli, "A multi-resolution system of Rao-Wilton-Glisson functions," *IEEE Trans. Antennas Propag.*, vol. 55, no. 3, pp. 924–930, Mar. 2007.

[35] M. A. Echeverri, M. A. Francavilla, F. Vipiana, and G. Vecchi, "A hierarchical fast solver for EFIE-MoM analysis of multiscale structures at very low frequencies," *IEEE Trans. Antennas Propag.*, vol. 62, no. 3, pp. 1523–1528, Mar. 2014.

Mario Alberto Echeverri Bautista (GS'13–M'16) received the B.Sc. degree from the Universidad Nacional de Colombia (UNAL), Bogotá, Colombia, in 2008, and the M.Sc. degree in electronics and telecommunications and Ph.D. degree from the Politecnico di Torino, Turin, Italy, in 2012 and 2016, respectively.

From 2009 to 2010, he was with the EMC Research Group, UNAL, involved in high power electromagnetic fields measurement in the microwaves regime. He received a research scholarship from the Politecnico di Torino and spent six months doing research in numerical electromagnetics during 2012–2013. During 2014–2015, he spent eight months as a Visiting Scholar with the Applied Mathematics Department, University of Colorado, Boulder, CO, USA, where he was involved with structured matrices for integral equations. He recently joined to ASML. His main research interests include fast solvers, preconditioners, domain decomposition analysis for integral equations, and electromagnetic scattering applications.

Matteo Alessandro Francavilla (M'15) received the Laurea degree in telecommunication engineering and Ph.D. degree in applied electromagnetics from the Politecnico di Torino, Turin, Italy, in 2007 and 2011, respectively.

During 2007, he spent six months with the Netherlands Organization for Applied Scientific Research (TNO), The Hague, The Netherlands. During 2010, he carried out his doctoral research at the University of Houston, Houston, TX, USA. Since January 2011, he has been a Researcher with the Antenna and EMC Lab (LACE), Istituto Superiore Mario Boella, Turin, Italy. His research interests include integral equations, numerical techniques for antennas, fast solvers, preconditioners, periodic structures, and layered media analysis.

Per-Gunnar Martinsson received the M.S. degree in engineering physics and Licenciate degree in mathematics from the Chalmers University of Technology, Gothenburg, Sweden, in 1996 and 1998, respectively, and the Ph.D. degree in computational and applied mathematics from The University of Texas at Austin, Austin, TX, USA, in 2002.

From 2002 to 2005, he was a Gibbs Assistant Professor with Yale University, and is currently a Professor of applied mathematics with the University of Colorado at Boulder, Boulder, CO, USA, where he has been a faculty member since 2005. His research interests include numerical methods for solving partial differential equations with a particular focus on the development of direct solvers for elliptic problems with linear or close to linear complexity. He is also conducting research in numerical linear algebra with an emphasis on randomized methods.

Dr. Martinsson was the recipient of an NSF Career Award in 2008. He served as the Principal Lecturer at a CBMS conference on direct solvers at Dartmouth College in 2014.

Francesca Vipiana (M'07–SM'13) received the Laurea and Ph.D. (Dottorato di Ricerca) degrees in electronic engineering from the Politecnico di Torino, Turin, Italy, in 2000 and 2004, respectively.

Her doctoral research was carried out partly at the European Space Research Technology Centre, Noordwijk, The Neterlands. From 2005 to 2008, she was a Research Fellow with the Department of Electronics, Politecnico di Torino. Since 2008, she has had several visiting periods with the Applied Electromagnetics Laboratory, University of Houston, Houston, TX, USA. From 2009 to 2012, she was with the Istituto Superiore Mario Boella (ISMB), Turin, Italy, as the Head of the Antenna and EMC Lab (LACE). Since 2012, she has been with the Department of Electronics and Telecommunications, Politecnico di Torino, as an Assistant Professor, and since 2014, as an Associate Professor. Her research interests include numerical techniques based on the integral equation and method of moments approaches with a focus on multiresolution and hierarchical schemes, domain decomposition, preconditioning and fast solution methods, Greens function regularization, and advanced quadrature integration schemes. She is also involved in the analysis, synthesis, and optimization of multiband reconfigurable compact antennas, and more recently, in the modeling and design of microwave imaging systems for medical applications.

Prof. Vipiana was the recipient of the Young Scientist Award of the Union of Radio Science General Assembly in 2005, the First Prize in the Poster Competition at the First IEEE Women in Electromagnetics Workshop in 2009, and the ISMB Best Paper Award in 2011.