





Accelerated Evaluation of Quasi-Static Interaction Integrals via Cubic Spline Interpolation in the Framework of the PEEC Method

Daniele Romano , Ivana Kovacevic-Badstuebner , *Senior Member, IEEE*,
Giulio Antonini , *Senior Member, IEEE*, and Ulrike Grossner , *Senior Member, IEEE*

Abstract—Fast Fourier transform (FFT)-based techniques are used in conjunction with iterative solvers to accelerate matrix–vector products in the framework of integral-equation-based methods. Even using this type of technique, the interaction matrix fill-in can be time consuming because of the huge number of coefficients to be computed. In this work, a cubic spline interpolation in space of the coefficients of potential is proposed in the framework of the partial element equivalent circuit method. The presented method can be used in the framework of FFT-based volume integral techniques, such as the partial element equivalent circuit technique. Two numerical tests are presented confirming a significant speedup while preserving the accuracy.

Index Terms—Cloud computing system, cubic spline interpolation, integral equations, partial element equivalent circuit (PEEC) method, partial element evaluation.

I. INTRODUCTION

NOWADAYS, integral-equation-based methods are recognized as powerful numerical methods for solving electromagnetic (EM) radiation and coupling problems. The most popular technique in the antenna area is the method of moments [1], [2], while in the EM compatibility/EM interference area, the partial element equivalent circuit (PEEC) method [3] has become increasingly popular because of its ability to provide a circuit interpretation to the mixed-potential integral equation, thus allowing to handle mixed circuit/EM problems. The application of all the integral-equation-based methods ends up with the solution of a linear system in the frequency domain and has $O(N^2)$ memory requirements to store the system matrix and $O(N^2)$ operations to perform the matrix–vector products via an iterative solver, where N is the number of unknowns. Over the

last 30 years, many algorithms have been proposed to reduce the memory requirement and central processing unit (CPU) time for solving linear systems iteratively: the fast multipole method in its single- and multilevel versions [4], [5], [6], algebraic methods like the adaptive cross approximation [7], [8], multiscale decomposition techniques [9], and hierarchical matrices [10].

Despite these advancements in the iterative solution of linear systems for the integral-equation-based methods, the CPU time for calculating the interaction matrix is still significant because the number of fill-in coefficients is extremely large, and thus, their computation is time consuming.

In the case of the PEEC method, the coefficients to be computed describing the magnetic and electric field couplings are known as partial inductances and coefficients of potential [11], [12]. Furthermore, when the full-wave Green’s function is considered, these coefficients have to be recomputed at each frequency in the range of interest. To mitigate this computational burden, a new accurate interpolation methodology is proposed in [13] for fast frequency sweeps taking advantage of the smooth behavior of these coefficients with the frequency.

On the other hand, the PEEC method is based on the source-contrast formulation, the magnetic and electric field couplings are assumed to take place in the background medium, and thus, the free-space Green’s function is used. This implies that there is a dependence of the PEEC coefficients on distance R , while the size of the elementary volumes and surfaces is less and less important as the distance increases. This results in an almost $1/R$ behavior of coefficients in space, which is, in principle, well suited for the interpolation when a large number of elementary volumes and surfaces can have the same relative geometrical configuration, i.e., their absolute geometrical positions result from only rotational and/or translation transformations.

A way to exploit such geometrical configurations was presented in [14] and [15] using an efficient identification of the elementary cell positions allowing a significant speedup of the partial element matrices filling-in process.

More recently, in the framework of integral-equation-based techniques, the translational invariance of elementary magnetic and electric field interactions has been exploited in [16] to accelerate the matrix–vector product of iterative solvers. This leads to significant memory saving and improvements in computation performances since it is shown that only one row of the

Manuscript received 20 July 2023; revised 15 December 2023; accepted 12 February 2024. Date of publication 29 February 2024; date of current version 13 June 2024. This work was supported by the Swiss National Science Foundation under Grant 209501. (*Corresponding author: Daniele Romano.*)

Daniele Romano and Giulio Antonini are with the Electromagnetic Compatibility Laboratory, Department of Industrial and Information Engineering and Economics, University of L’Aquila, I-67100 L’Aquila, Italy (e-mail: daniele.romano@univaq.com; giulio.antonini@univaq.it).

Ivana Kovacevic-Badstuebner and Ulrike Grossner are with the Advanced Power Semiconductor Laboratory, ETH Zürich, 8092 Zurich, Switzerland (e-mail: kovacevic@aps.ee.ethz.ch; ulrike.grossner@ethz.ch).

Color versions of one or more figures in this article are available at <https://doi.org/10.1109/TEMC.2024.3366985>.

Digital Object Identifier 10.1109/TEMC.2024.3366985

interaction matrices of the voxelized geometry has to be computed. Such an approach was applied to PEEC models in [17] and [18] in the so-called fast Fourier transform (FFT)-PEEC method. Nevertheless, the number of coefficients can be huge.

Therefore, a practical methodology for the interpolation of the partial inductances based on a cubic voxelization was presented in [19] for the quasi-static PEEC method.

This article presents an extended interpolation method for the coefficients of potential and an efficient FFT-PEEC solver with a feature of fast computation of large inductance and coefficient of potential matrices.

II. BASIC PEEC FORMULATION

The standard PEEC method [20] discretizes volumes and surfaces into hexahedra and patches, respectively. Current and charge densities are expanded into a series of basis functions. Current and charge densities are typically assumed to be constant over the elementary volume and surface cells, and thus, rectangular basis functions are chosen. By applying standard Galerkin's testing procedure to the electric field integral equation and continuity equation, electrical lumped elements are identified modeling both the magnetic and electric field couplings. Dielectrics are modeled by their excess capacitance taking the dielectric polarization into account [21], while conductor losses are modeled through their ohmic resistance or the introduction of a surface impedance [22]. Magnetic and electric field couplings are modeled by partial inductances and coefficients of potential, respectively.

Enforcing Kirchhoff voltage and current laws to the PEEC circuit yields the following system of equations in the frequency domain:

$$\underbrace{\begin{bmatrix} \mathbf{Z}_s + j\omega\mathbf{L}_p & \mathbf{0} & \mathbf{A} \\ \mathbf{0} & \mathbf{P} & -\mathbf{M} \\ -\mathbf{A}^T & j\omega\mathbf{M} & \mathbf{Y}_{\ell e} \end{bmatrix}}_{\text{MNA}} \cdot \underbrace{\begin{bmatrix} \mathbf{I} \\ \mathbf{Q} \\ \phi \end{bmatrix}}_{\mathbf{X}} = \underbrace{\begin{bmatrix} \mathbf{0} \\ \mathbf{0} \\ \mathbf{I}_s \end{bmatrix}}_{\mathbf{U}} \quad (1)$$

where \mathbf{P} accounts for the coefficients of potential, \mathbf{L}_p is the partial inductance matrix, $\mathbf{Y}_{\ell e}$ is the lumped elements matrix, \mathbf{A} is the incidence matrix, and \mathbf{M} is a selection matrix introduced in [7]. Matrix \mathbf{Z}_s is a diagonal matrix with the self-impedances of elementary volumes, which reduce to resistances for conductors [3] and the impedances of the excess capacitance [21] for dielectrics. Finally, \mathbf{I}_s represents the independent current source, which is assumed to excite the system.

When system (1) is large, the fill-in of the dense matrices \mathbf{P} and \mathbf{L}_p is very time consuming even if closed-form formulas can be used under the quasi-static hypothesis and assuming a Manhattan mesh type. In [23] and [24], the FFT circulant tensors for \mathbf{P} and \mathbf{L}_p are computed once, compressed with the Tucker singular value decomposition, and then stored in a file. After that, at each iteration of the iterative solver, the compressed circulant tensors are first loaded and then decompressed. Then, they are properly scaled during the matrix–vector products with a factor of Δ_v^{-1} for matrix \mathbf{P} and with a factor Δ_v for matrix \mathbf{L}_p , being Δ_v the size of the cubes in each direction.

It is evident that this approach introduced in [23] and [24] is very useful when operating on a single computer in conjunction with the FFT acceleration technique, while in the case of a parallelized environment, it is better to fast compute the partial elements directly on each thread and/or a single computer of the network.

In addition, if an iterative solver is used in a parallelized environment, the storage and the transfer of a portion of these matrices can be prohibitive, and for this reason, the matrix coefficients are repeatedly computed directly on several threads and/or computers of the network skipping in this way the data exchange.

Under the same hypothesis of [23] and [24], in this work, we propose a similar strategy to fast-filling of matrix \mathbf{P} . A coefficient of potential between two capacitive surface cells γ and δ , under the quasi-static hypothesis, requires the evaluation of the following double-folded surface integrals:

$$P_{\gamma\delta} = \frac{1}{4\pi\epsilon_0} \frac{1}{S_\gamma S_\delta} \int_{S_\gamma} \int_{S_\delta} \frac{1}{R_{\gamma\delta}} dS_\gamma dS_\delta \quad (2)$$

where $R_{\gamma\delta}$ is the distance between any two points on surfaces γ and δ , while S_γ and S_δ denote the area of their respective surfaces. The closed-form formulas for parallel and orthogonal surfaces to compute the integrals in (2) can be found in [20].

III. P-MATRIX FILL-IN ACCELERATION VIA CUBIC SPLINE INTERPOLATION

In the following, we propose a method for fast evaluation of \mathbf{P} by using the cubic spline interpolation [25], [26] with a similar approach presented in [19] for matrix \mathbf{L}_p . Nevertheless, the interpolation of the \mathbf{P} -matrix entries is more challenging as it requires both the parallel and orthogonal geometrical positions to be distinguished in comparison to the inductance matrix.

A. Minimal Domain Identification

For accelerating the computation of \mathbf{P} -matrix elements, a starting point is a voxelized mesh made by $N_x \cdot N_y \cdot N_z$ cubic voxels of unit size, where N_x is the number of voxels along x , N_y is the number of voxels along y , and N_z is the number of voxels along z . Let $N_x = N_y = N_z = 1000$. In the next step, the minimal set of configurations inside the modeling domain is identified. The minimal set of geometrical configurations represents a set of surface pairs having a unique relative position between them. For a case of $N_x = N_y = 10$ and $N_z = 9$ shown in Fig. 1, the corresponding minimal sets of geometrical configurations are shown in Figs. 2 and 3 for parallel and orthogonal surfaces, respectively.

Namely, the mutual couplings between the blue surface (the surface closer to the origin of the Cartesian axes) with itself and with all the other surfaces are considered. The surfaces marked with a dot are distinguished, as discussed in Section III-C.

B. Near and Far Interaction Regions

To preserve higher accuracy for the mutual couplings for which the surfaces are physically close to each other, we can

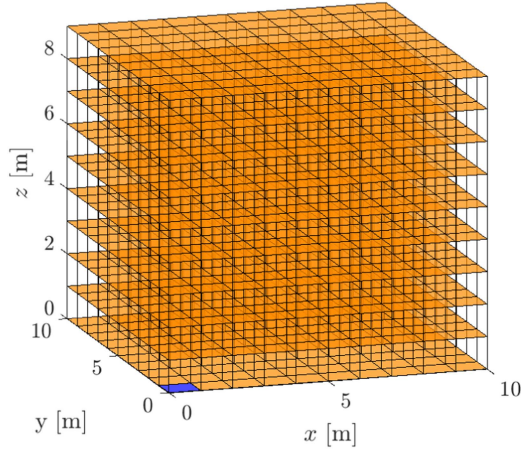


Fig. 1. All possible cases for parallel surfaces for a cubic voxelization made by $N_x = N_y = 10$ and $N_z = 9$.

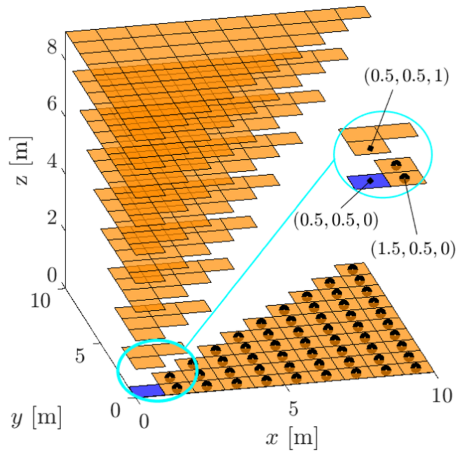


Fig. 2. Reduced cases for parallel surfaces for a cubic voxelization made by $N_x = N_y = 10$ and $N_z = 9$.

split the entire domain as follows. First, all the configurations shown in Figs. 2 and 3 are sorted by using their center-to-center distance R_{cc} . Then, we define:

- 1) a near interaction region that contains all the configurations for which the surfaces are physically close. The used criterion is a condition $R_{cc} \leq 100$ m with the unit size equals 1 m;
- 2) a far interaction region that contains all the configurations for which the surfaces are physically distant. The used criterion is a condition $R_{cc} > 100$ m.

C. Interpolation Strategy

The cubic spline interpolation, like other interpolation techniques, is a method to find/estimate new data points based on the range of a discrete set of known data points. A detailed description of the technique can be found in [25]. To apply the cubic spline interpolation for the parallel surfaces shown in Fig. 2, we define two subsets of configurations.

- 1) The first subset includes the blue surface (the closest to the origin of the Cartesian axes) and all the surfaces without

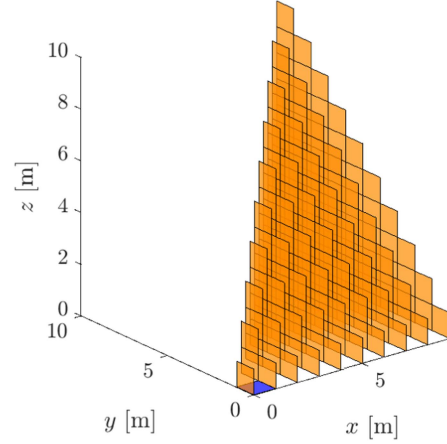


Fig. 3. Reduced cases for orthogonal surfaces for a cubic voxelization made by $N_x = 9$ and $N_y = N_z = 10$.

the dot marker. The total number of these configurations is

$$\sum_{i=1}^N [N - (i - 1)] i \quad (3)$$

being $N = N_x = N_y = N_z$.

- 2) The second subset includes the blue surface (the closest to the origin of the Cartesian axes) and all the surfaces with the dot marker. The total number of these configuration is $N(N + 1)/2$.

This separation is performed in order to consider that the mutual coefficient between some pairs of surfaces having the same R_{cc} can be quite different. For example, the \mathbf{P} coefficient calculated by (2) between the blue surface centered in the point $(0.5, 0.5, 0)$ and the surface centered in the point $(1.5, 0.5, 0)$ marked in the zoomed part of Fig. 2 is quite different to the coefficient computed between the blue surface and the surface centered in the point $(0.5, 0.5, 1)$ even though the center-to-center distance is the same.

In addition, these two subsets of parallel surfaces are further split for both the near and far interaction regions defined in Section III-B.

Finally, in each subset, 40 logarithmically spaced values of R_{cc} are selected, and for these samples, we compute the corresponding values of \mathbf{P} , and we use only these points for the interpolation. Accordingly, for parallel surfaces, there are four subsets in total, i.e., two subsets previously defined for the near and far interactions, each consisting of 40 samples used to perform the interpolation.

For the orthogonal surfaces, the same interpolation strategy is applied, but, differently from the parallel case, in this case, it is not possible to have pairs of surfaces having the same center-to-center distance with a different mutual coupling coefficient. For this reason, for orthogonal surfaces, we distinguish only the near region from the far region to set the samples required for the interpolation.

IV. NUMERICAL TESTS

In this section, the performance and the accuracy of the proposed interpolation strategy for the coefficients of potential for the near and far interaction regions are first presented. In the next step, the interpolation strategy is applied in the framework of the PEEC method for two numerical examples, in which the accuracy and efficiency of the matrix \mathbf{L}_p coefficient interpolation, introduced in [19], are investigated. The PEEC solver is an iterative solver based on the generalized minimal residual method (GMRES), in which the matrix–vector product involving \mathbf{L}_p and \mathbf{P} is accelerated through the FFT-based approach [16], [17]. The FFT acceleration can be summarized as follows. In each iteration step of the PEEC iterative solver, the vector \mathbf{I}_{est} representing the estimated solution for currents is required to perform the matrix–vector product $\mathbf{L}_p \mathbf{I}_{\text{est}}$ and find a new estimated solution \mathbf{X}_{est} , i.e., $\mathbf{X}_{\text{est}} = \mathbf{L}_p \mathbf{I}_{\text{est}}$. In this case, first, two 3-D matrices (circulant tensors) are stored for both \mathbf{L}_p and \mathbf{I}_{est} through their multidimensional FFT, and then, a 3-D matrix $\mathbf{X}_{\text{est}}^{3D}$ is computed as the result of the member-to-member product between the two circulant tensors. Finally, the vector \mathbf{X}_{est} is restored from $\mathbf{X}_{\text{est}}^{3D}$ through its inverse FFT. In this process, it is sufficient to compute only one row of matrix \mathbf{L}_p . A detailed description can be found in [17]. The proposed cubic spline interpolation is performed using the `spline` MATLAB built-in function, while the standard computation of matrices \mathbf{L}_p and \mathbf{P} is carried out by using precompiled C++ MATLAB MEX files. In addition, the fill-in of these matrices is performed by using the technique proposed in [27] to avoid numerical errors due to the precision digits. All the simulations have been carried out on a computer equipped with 64 GB of RAM memory and a quad-core Intel processor operating at 2.4 GHz.

A. Interpolation Validation

The accuracy of the proposed interpolation strategy of coefficients of potential is validated on the entire domain made by $N_x = N_y = N_z = 1000$ cubic voxels. The relative error between the coefficients computed by using the closed formula (2), denoted as h_{closed} , and interpolated coefficients, denoted as h_{interp} , is defined as

$$\text{err} = \frac{|h_{\text{closed}} - h_{\text{interp}}|}{|h_{\text{closed}}|}. \quad (4)$$

The coefficients and relative error are shown in Figs. 4 and 5 for parallel and orthogonal surfaces, respectively.

In the near region, for parallel surfaces, the maximum error is around 0.01, while for orthogonal surfaces, it is around 0.003. In the far region, for both the parallel and orthogonal surfaces, the maximum error is around 0.027. In addition, for both the parallel and orthogonal surfaces, the closed-form evaluation required around 118 s, while the cubic spline interpolation required 8 s (thus a $15\times$ speedup).

B. Copper Horseshoe Example

To evaluate the effectiveness of the proposed approach, a copper horseshoe structure has been modeled. Its geometry is

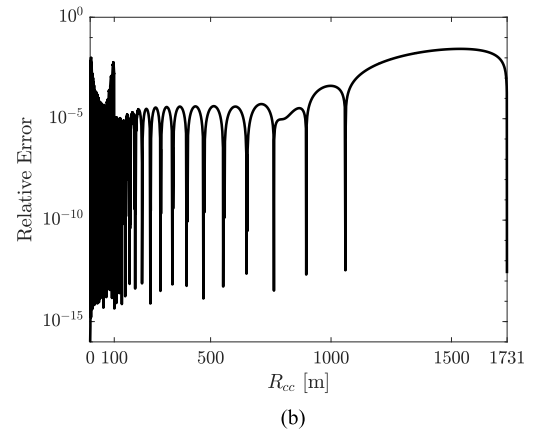
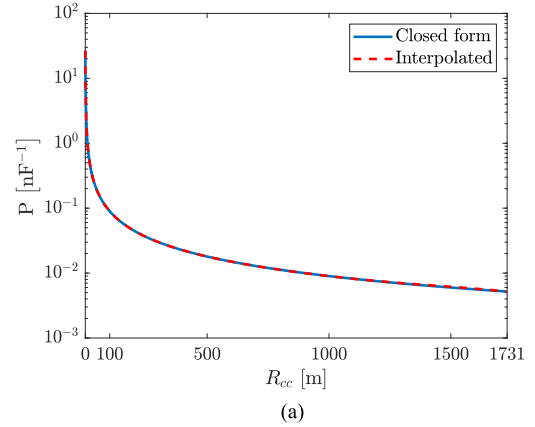


Fig. 4. Coefficients of matrix \mathbf{P} and interpolation relative error in the case of parallel surfaces on the entire domain made by $N_x = N_y = N_z = 1000$ cubic voxels. (a) Rigorous and interpolated coefficients of potential. (b) Interpolation relative error.

shown in Fig. 6. The PEEC analysis, from 10 Hz to 100 MHz, has been performed by using a voxelization with $N_x = 120$, $N_y = 60$, and $N_z = 10$, leading to 148 000 inductive branches, 16 000 external surfaces, and 52 000 nodes. With this mesh, the edge size of the cubic voxels is 0.2 mm. It satisfies the $\lambda/40$ criterion condition, i.e., $0.2 \times 10^{-3} < \frac{c_0}{40 \times 10^8}$, while the skin effect is properly modeled by using the surface impedance introduced in [18]. In this example, system (1) is solved with:

- 1) the coefficients of matrices \mathbf{L}_p and \mathbf{P} computed by using the closed formulas. The solution obtained in this way is denoted as Reference;
- 2) the coefficients of matrices \mathbf{L}_p and \mathbf{P} computed by using the cubic spline interpolation. The solution obtained in this way is denoted as Proposed.

For both the methods, the GMRES threshold for the convergence is set to 10^{-5} . The probe impedance, computed with Reference and Proposed approaches, is shown in Fig. 7(a), where it can be seen that the results are practically the same for both the methods. The relative error between Reference and Proposed is shown in Fig. 7(b), where it can be seen that the error is always less than 0.1% on the entire frequency range.

The CPU time for all the PEEC steps required by Reference and Proposed methods is reported in Table I. In particular, the

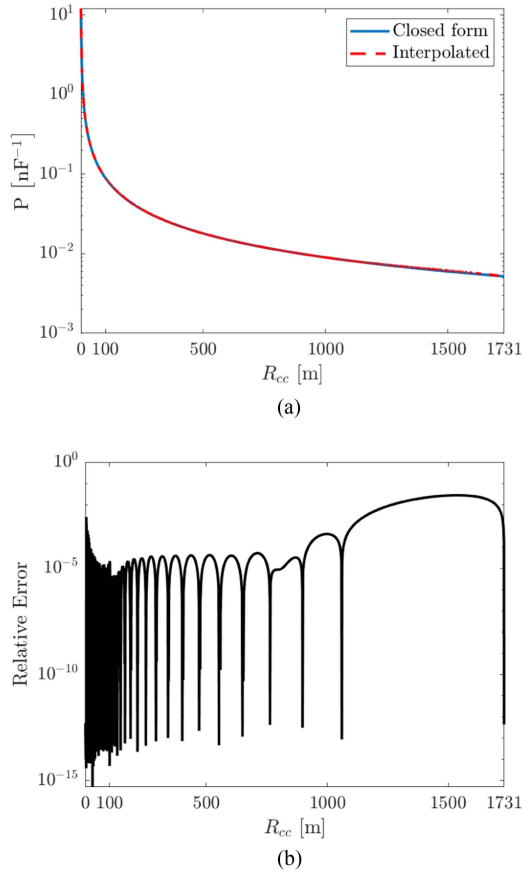


Fig. 5. Coefficients of matrix \mathbf{P} and interpolation relative error in the case of orthogonal surfaces on the entire domain made by $N_x = N_y = N_z = 1000$ cubic voxels. (a) Rigorous and interpolated coefficients of potential. (b) Interpolation relative error.

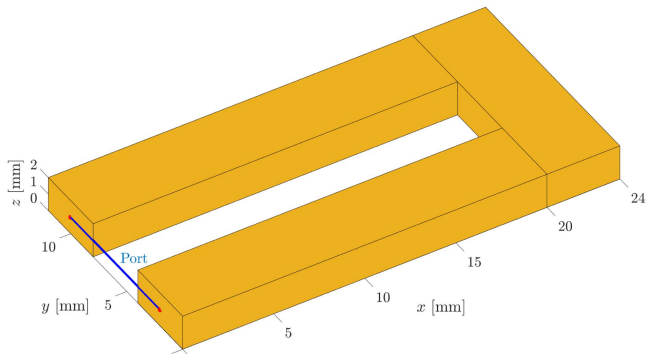


Fig. 6. Geometry of the copper horseshoe example.

TABLE I
CPU TIME REQUIRED BY REFERENCE AND PROPOSED TO FILL THE ROWS OF
MATRICES \mathbf{L}_p AND \mathbf{P} , REQUIRED BY THE FFT-BASED ACCELERATION
TECHNIQUE, FOR THE COPPER HORSESHOE EXAMPLE

	Reference	Proposed
mesh time	0.8 s	0.8 s
\mathbf{L}_p filling time	5 s	50 ms
\mathbf{P} filling time	4 s	0.2 s
solver time	280 s	280 s

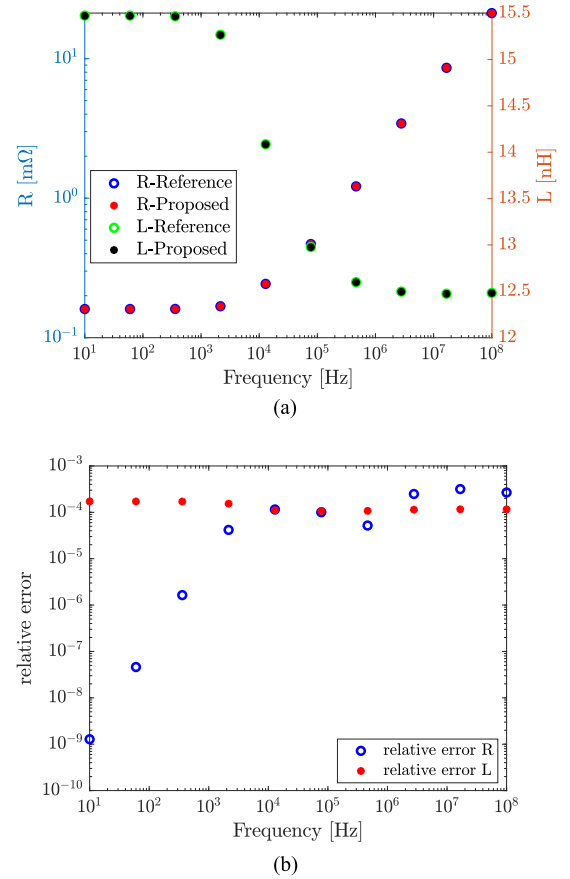


Fig. 7. Input resistance and inductance and relative error for the copper horseshoe example. (a) Input resistance and inductance. (b) Relative error.

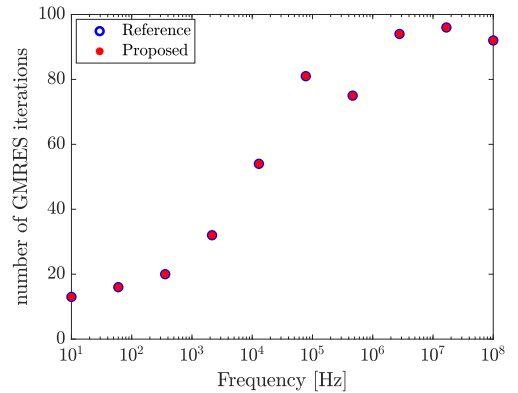


Fig. 8. Number of iterations for the copper horseshoe example.

filling time for the partial coefficients is the time to compute just one row of matrices \mathbf{L}_p and \mathbf{P} since the FFT-based acceleration technique is applied. The results in Table I confirm that the interpolation strategy leads to a speedup of $100\times$ for \mathbf{L}_p and a speedup of $20\times$ for \mathbf{P} . Finally, the number of GMRES iterations for both the Reference and Proposed methods is shown in Fig. 8. As clearly can be seen, the proposed interpolation strategy does not impact the convergence of the iterative solver.

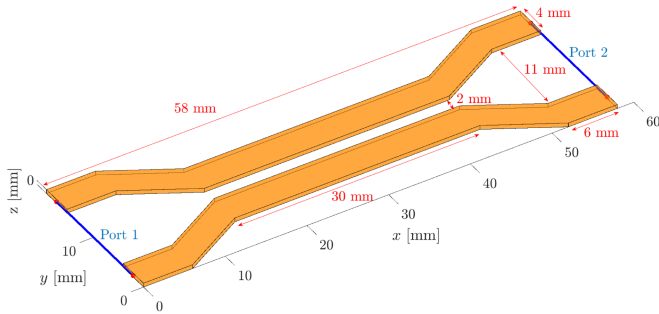


Fig. 9. Geometry of the copper interconnect structure example.

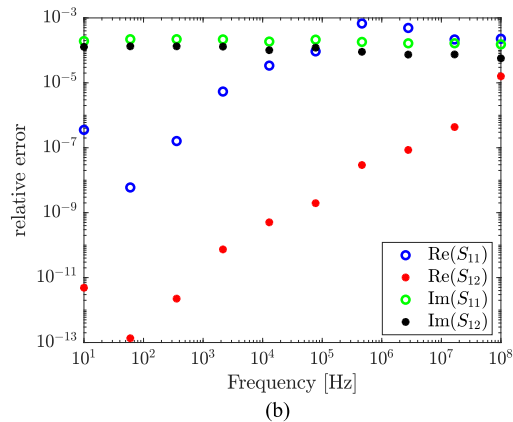
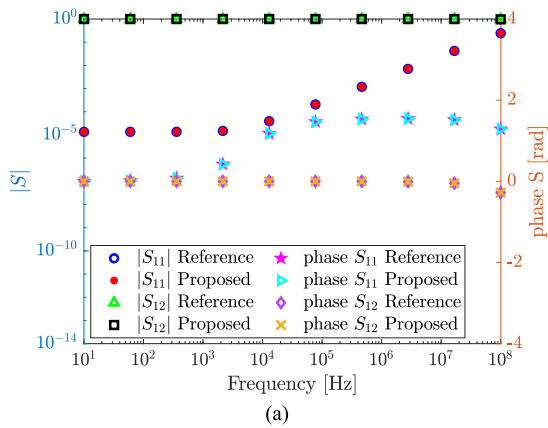


Fig. 10. S -parameters and relative error for the copper interconnect structure example. (a) Scattering parameters. (b) Relative error.

C. Copper Interconnect Structure Example

In this last example, a copper interconnect structure example is analyzed. Its geometry is shown in Fig. 9. The PEEC analysis, from 10 Hz to 100 MHz, is performed by using a voxelization with $N_x = 580$, $N_y = 190$, and $N_z = 5$, leading to have 642 962 inductive branches, 107 066 external surfaces, and 232 165 nodes. With this mesh, the edge size of the cubic voxels is 0.1 mm. Also, in this case, the mesh size of 0.1 mm satisfies the $\lambda/40$ criterion condition since $0.1 \times 10^{-3} < \frac{c_0}{40 \times 10^8}$, while the skin effect is properly modeled by using the surface impedance introduced in [18]. Also, in this case, system (1) is solved with the Reference and Proposed methods defined in Section IV-B,

TABLE II
CPU TIME REQUIRED BY REFERENCE AND PROPOSED TO FILL THE ROWS OF MATRICES \mathbf{L}_p AND \mathbf{P} , REQUIRED BY THE FFT-BASED ACCELERATION TECHNIQUE, FOR THE COPPER INTERCONNECT STRUCTURE EXAMPLE

	Reference	Proposed
mesh time	3 s	3 s
\mathbf{L}_p filling time	9 s	80 ms
\mathbf{P} filling time	10 s	0.6 s
solver time	39 m	39 m

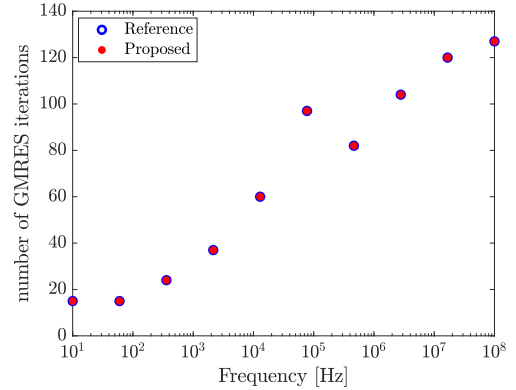


Fig. 11. Number of iterations for the copper interconnect structure example.

and the GMRES threshold for the convergence is set to 10^{-5} . The magnitude and phase spectra of the scattering parameters obtained with both the methods are shown in Fig. 10(a) confirming a good agreement between the results obtained with both the methods. The relative error between Reference and Proposed, shown in Fig. 10(b), is always less than 0.1% on the entire frequency range of interest.

The CPU time for all the PEEC steps required by Reference and Proposed methods is reported in Table II.

As can be clearly seen from Table II, the interpolation strategy allows achieving a speedup of $112\times$ for \mathbf{L}_p and a speedup of $16\times$ for \mathbf{P} .

Finally, the number of GMRES iterations for both the Reference and Proposed methods is shown in Fig. 11. Also, in this case, the proposed interpolation strategy does not impact the convergence of the iterative solver.

V. CONCLUSION

In this article, the cubic spline interpolation method used to accelerate the computation of partial inductances, introduced in previous work, is extended for the computation of the coefficients of potentials for geometries meshed in cubic subregions in the framework of the PEEC method under the quasi-static hypothesis. To this purpose, a minimal set of geometrical configurations is first identified, and then, the cubic spline interpolation is applied to speed up the computation of the mutual coupling coefficients. The numerical tests demonstrate the accuracy of the proposed interpolation scheme and a significant computation speedup. The proposed method is particularly useful if used in conjunction with an iterative solver implemented within a parallel computing environment, where the rows of matrices must be computed as fast as possible to avoid data exchange

Algorithm 1: Pseudocode of Procedure PREPARE-P-SAMPLES.

Output: $\mathbf{R}_{sa}, \mathbf{P}_{sa}$

- 1: Compute the centers \mathbf{C} for all the surfaces
 - 2: Compute all the distances \mathbf{R} between the point $C_o = (0.5, 0.5, 0)$ and the centers \mathbf{C}
 - 3: Sort the vector \mathbf{R} in ascending order
 - 4: Store N_{sa} logarithmically spaced points of \mathbf{R} into \mathbf{R}_{sa}
 - 5: Compute the N_{sa} coefficients of potentials \mathbf{P}_{sa} by using the geometrical configurations belonging to \mathbf{R}_{sa} .
-

between computers and/or threads. As a future step, the authors will investigate the extension of the technique to the computation of the full-wave coefficients.

APPENDIX

In what follows, the implementation of the proposed interpolation strategy is discussed in detail. As the first step, it is needed to create six datasets of the samples required to interpolate the coefficient of matrix \mathbf{P} , as reported in Section III: three datasets for the near-field interactions and three datasets for the far-field interactions:

- 1) one dataset for the parallel surfaces related to the dotted surfaces (see Fig. 2);
- 2) one dataset for the parallel surfaces related to the nondotted surfaces (see Fig. 2);
- 3) one dataset for the orthogonal surfaces (see Fig. 3).

To build a generic dataset, the procedure PREPARE-P-SAMPLES can be used. Its pseudocode is given in Algorithm 1. At line 1, the centers \mathbf{C} for the surfaces are disposed, as shown in Figs. 2 and 3. Then, the distances \mathbf{R} between the centers \mathbf{C} and the point $C_o = (0.5, 0.5, 0)$ are computed, being C_o the center of the blue surface in Figs. 2 and 3. At line 3, the vector \mathbf{R} is sorted in the ascending order, and then, at line 4, N_{sa} logarithmically spaced points of \mathbf{R} are selected and stored into a vector \mathbf{R}_{sa} (we suggest to use at least $N_{sa} = 40$ since keeping fewer samples leads to an inaccurate interpolation). Finally, at line 5, the vector \mathbf{P}_{sa} is filled with N_{sa} coefficients of potentials, computed through the formula (2), by using the geometrical configuration of the surfaces belonging to \mathbf{R}_{sa} . It is important to underline that this procedure is invoked just one time, and then, the pairs of vectors \mathbf{R}_{sa} and \mathbf{P}_{sa} (one pair for each dataset) must be kept in RAM memory or must be directly hard coded into the procedure COMPUTE-P-INTERPOLATED. Such a procedure, whose pseudocode is shown in Algorithm 2, is the only function used to compute a generic coefficient P_{ij} of the matrix \mathbf{P} .

At line 1, the center-to-center distance R_{ij} between the surfaces S_i and S_j is computed, and then, R_{ij} , \mathbf{R}_{sa} , and \mathbf{P}_{sa} are used to compute the interpolated P_{ij} by exploiting the cubic spline interpolation [25]. Clearly, \mathbf{R}_{sa} and \mathbf{P}_{sa} must be properly chosen by checking the distance R_{ij} (near or far field) and distinguishing between the parallel (dotted surfaces or not) and orthogonal surface cases. Finally, being Δ_v the edge size of the cubes of the mesh, the coefficient P_{ij} is properly scaled by the factor Δ_v^{-1} .

Algorithm 2: Pseudocode of Procedure COMPUTE-P-INTERPOLATED.

Procedure: COMPUTE-P-INTERPOLATED($S_i, S_j, \mathbf{R}_{sa}, \mathbf{P}_{sa}, \Delta_v$)

Output: P_{ij}

- 1: Compute the center-to-center distance R_{ij} between the surfaces S_i and S_j
 - 2: Compute the interpolated coefficient P_{ij} by using R_{ij} , $\mathbf{R}_{sa}, \mathbf{P}_{sa}$
 - 3: Set $P_{ij} = 1/\Delta_v P_{ij}$.
-

REFERENCES

- [1] R. F. Harrington, *Field Computation by Moment Methods*, Melbourne, FL, USA: Krieger, 1982.
- [2] 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.
- [3] A. Ruehli, "Equivalent circuit models for three-dimensional multiconductor systems," *IEEE Trans. Microw. Theory Techn.*, vol. MTT-22, no. 3, pp. 216–221, Mar. 1974.
- [4] R. Coifman, V. Rokhlin, and S. Wandzura, "The fast multipole method for the wave equation: A pedestrian prescription," *IEEE Antennas Propag. Mag.*, vol. 35, no. 3, pp. 7–12, Jun. 1993.
- [5] J. Song, C.-C. Lu, and W. C. Chew, "Multilevel fast multipole algorithm for electromagnetic scattering by large complex objects," *IEEE Trans. Antennas Propag.*, vol. 45, no. 10, pp. 1488–1493, Oct. 1997.
- [6] W. C. Chew, J.-M. Jin, E. Michielssen, and J. Song, *Fast and Efficient Algorithms in Computational Electromagnetics*, Norwood, MA, USA: Artech House, 2001.
- [7] D. Romano and G. Antonini, "Adaptive-cross-approximation-based acceleration of transient analysis of quasi-static partial element equivalent circuits," *IET Microw., Antennas Propag.*, vol. 9, pp. 700–709, May 2015.
- [8] D. Voltolina, R. Torchio, P. Bettini, R. Cavazzana, and M. Moresco, "PEEC modeling of planar spiral resonators," *IEEE Trans. Magn.*, vol. 56, no. 1, Jan. 2020, Art. no. 6700404.
- [9] G. Antonini and D. Romano, "Efficient frequency-domain analysis of PEEC circuits through multiscale compressed decomposition," *IEEE Trans. Electromagn. Compat.*, vol. 56, no. 2, pp. 454–465, Apr. 2014.
- [10] D. Voltolina, P. Bettini, P. Alotto, F. Moro, and R. Torchio, "High-performance PEEC analysis of electromagnetic scatterers," *IEEE Trans. Magn.*, vol. 55, no. 6, Jun. 2019, Art. no. 7201204.
- [11] A. E. Ruehli, "Inductance calculations in a complex integrated circuit environment," *IBM J. Res. Develop.*, vol. 16, no. 5, pp. 470–481, 1972.
- [12] A. Ruehli and P. Brennan, "Efficient capacitance calculations for three-dimensional multiconductor systems," *IEEE Trans. Microw. Theory Techn.*, vol. 21, no. 2, pp. 76–82, Feb. 1973.
- [13] G. Antonini and D. Romano, "An accurate interpolation strategy for fast frequency sweep of partial element equivalent circuit models," *IEEE Trans. Electromagn. Compat.*, vol. 56, no. 3, pp. 653–658, Jun. 2014.
- [14] D. Romano, L. Lombardi, and G. Antonini, "Acceleration of the partial element equivalent circuit method with uniform tessellation—Part I: Identification of geometrical signatures," *Int. J. Numer. Model.: Electron. Netw., Devices, Fields*, vol. 31, no. 6, 2018, Art. no. e2307, doi: 10.1002/jnm.2307.
- [15] D. Romano, L. Lombardi, and G. Antonini, "Acceleration of the partial element equivalent circuit method with uniform tessellation—Part II: Efficient frequency domain solver with interpolation and reuse of partial elements," *Int. J. Numer. Model.: Electron. Netw., Devices, Fields*, vol. 31, no. 6, 2018, Art. no. e2306, doi: 10.1002/jnm.2306.
- [16] A. Polimeridis, J. Villena, L. Daniel, and J. White, "Stable FFT-JVIE solvers for fast analysis of highly inhomogeneous dielectric objects," *J. Comput. Phys.*, vol. 269, pp. 280–296, 2014.
- [17] R. Torchio, F. Lucchini, J.-L. Schanen, O. Chadebec, and G. Meunier, "FFT-PEEC: A fast tool from cad to power electronics simulations," *IEEE Trans. Power Electron.*, vol. 37, no. 1, pp. 700–713, Jan. 2022.
- [18] D. Romano, I. Kovacevic-Badstuebner, G. Antonini, and U. Grossner, "Efficient PEEC iterative solver for power electronic applications," *IEEE Trans. Electromagn. Compat.*, vol. 65, no. 2, pp. 546–554, Apr. 2023.

- [19] D. Romano, F. Loreto, G. Antonini, I. Kovacevic-Badstuebner, and U. Grossner, "Accelerated partial inductance evaluation via cubic spline interpolation for the PEEC method," in *Proc. 52nd Eur. Microw. Conf.*, 2022, pp. 357–360.
- [20] A. E. Ruehli, G. Antonini, and L. Jiang, *Circuit Oriented Electromagnetic Modeling Using the PEEC Techniques*. Hoboken, NJ, USA: Wiley, 2017.
- [21] A. Ruehli and H. Heeb, "Circuit models for three-dimensional geometries including dielectrics," *IEEE Trans. Microw. Theory Techn.*, vol. 40, no. 7, pp. 1507–1516, Jul. 1992.
- [22] M. Bandinelli et al., "A surface PEEC formulation for high-fidelity analysis of the current return networks in composite aircrafts," *IEEE Trans. Electromagn. Compat.*, vol. 57, no. 5, pp. 1027–1036, Oct. 2015.
- [23] M. Wang, C. Qian, J. K. White, and A. C. Yucel, "VoxCap: FFT-accelerated and Tucker-enhanced capacitance extraction simulator for voxelized structures," *IEEE Trans. Microw. Theory Techn.*, vol. 68, no. 12, pp. 5154–5168, Dec. 2020.
- [24] C. Qian and A. C. Yucel, "On the compression of translation operator tensors in FMM-FFT-accelerated SIE simulators via tensor decompositions," *IEEE Trans. Antennas Propag.*, vol. 69, no. 6, pp. 3359–3370, Jun. 2021.
- [25] C. De Boor, *A Practical Guide to Splines* (Ser. Applied Mathematical Sciences). New York, NY, USA: Springer-Verlag, 1978.
- [26] A. Quarteroni, R. Sacco, and F. Saleri, *Numerical Mathematics—Polynomial Interpolation* (Ser. Texts in Applied Mathematics). New York, NY, USA: Springer, 2006.
- [27] I. Kovacevic-Badstuebner, D. Romano, L. Lombardi, U. Grossner, J. Ekman, and G. Antonini, "Accurate calculation of partial inductances for the orthogonal PEEC formulation," *IEEE Trans. Electromagn. Compat.*, vol. 63, no. 1, pp. 82–92, Feb. 2021.



Daniele Romano was born in Campobasso, Italy, in 1984. He received the Laurea degree in computer science and automation engineering and the Ph.D. degree in industrial engineering from the University of L'Aquila, L'Aquila, Italy, in 2012 and 2018, respectively.

Since 2012, he has been with the Electromagnetic Compatibility (EMC) Laboratory, University of L'Aquila, where he is currently a Researcher, focusing on EMC modeling and analysis, algorithm engineering, and speedup techniques applied to EMC problems.



Ivana Kovacevic-Badstuebner (Senior Member, IEEE) received the Ph.D. degree in electrical engineering from ETH Zürich, Zürich, Switzerland, in 2012.

From 2008 to 2015, she was with the Power Electronics Systems Laboratory, ETH Zürich, focusing on the prediction of electromagnetic behavior of power electronics systems based on the developed numerical techniques and the lifetime modeling of power semiconductor modules. In 2016, she joined the Advanced Power Semiconductor Laboratory, ETH Zürich. Her research interests include novel packaging technologies for SiC power devices, the optimization of power module layout with respect to electromagnetic interference, and multidomain modeling of power semiconductor devices and their modules.



Giulio Antonini (Senior Member, IEEE) received the Laurea degree (*cum laude*) in electrical engineering from the University of L'Aquila, L'Aquila, Italy, in 1994, and the Ph.D. degree in electrical engineering from the University of Rome "La Sapienza," Rome, Italy, in 1998.

Since 1998, he has been with the Electromagnetic Compatibility Laboratory, University of L'Aquila, where he is currently a Professor. He has coauthored the book titled *Circuit Oriented Electromagnetic Modeling Using the PEEC Techniques* (Hoboken, NJ, USA: Wiley/IEEE Press, 2017). He has authored more than 300 papers published on international journals and in the proceedings of international conferences. His research interests include computational electromagnetics.



Ulrike Grossner (Senior Member, IEEE) received the Dipl.-Phys. and Dr.rer.nat. degrees in physics from the Friedrich-Schiller-University, Jena, Germany, in 1997 and 2000, respectively.

In 2014, she became a Full Professor with ETH Zürich, Zürich, Switzerland, where she established the Advanced Power Semiconductor Laboratory, working on devices and packaging for advanced power semiconductors.