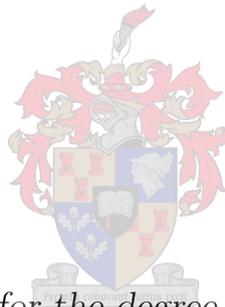


# Adaptive Cross Approximation Methods for Fast Analysis of Antenna Arrays

by

Keshav Sewraj



*Dissertation presented for the degree of Doctor of Philosophy  
in Electronic Engineering in the Faculty of Engineering at  
Stellenbosch University*

Supervisor: Prof. M. M. Botha

March 2021



# Declaration

By submitting this dissertation electronically, I declare that the entirety of the work contained therein is my own, original work, that I am the sole author thereof (save to the extent explicitly otherwise stated), that reproduction and publication thereof by Stellenbosch University will not infringe any third party rights and that I have not previously in its entirety or in part submitted it for obtaining any qualification.

Date: ..... March 2021 .....

Copyright © 2021 Stellenbosch University  
All rights reserved.



# Abstract

## Adaptive Cross Approximation Methods for Fast Analysis of Antenna Arrays

K. Sewraj

*Department of Electrical and Electronic Engineering,  
University of Stellenbosch,  
Private Bag X1, Matieland 7602, South Africa.*

Dissertation: Ph.D

March 2021

This work is focused on developing efficient numerical electromagnetic algorithms for the analysis of large antenna arrays, such as being considered as part of the international Square Kilometre Array (SKA) radio astronomy project currently under development. Numerical electromagnetic simulation is a vital tool to evaluate performance during the antenna design process, and it is common to iterate through thousands of simulations to fine-tune parameters. However, these simulations are often expensive and can be a limiting factor in the available design choices.

The Method of Moments (MoM) is a numerical technique used to solve electromagnetic field problems, and is highly suitable for radiation problems such as the analysis of antenna arrays. However, the memory and runtime requirement of the MoM scale as  $\mathcal{O}(N^2)$  and  $\mathcal{O}(N^3)$ , respectively, where  $N$  is the number of Degrees of Freedom. As such, electromagnetic analysis performed by the MoM is limited by the electrical size of the problem. For larger structures, fast MoM-based techniques tailored to specific problems need to be devised.

In this work, a variety of techniques based on cross approximation is explored and implemented, for array analysis. In this context, two Directional Cross Approximation (DCA)-based solvers are devised. The DCA is a nested multilevel algorithm which efficiently compresses MoM sub-blocks due to far interactions as a product of low-rank factors. During the computation of these factors, the far-field is segmented in angular sectors to ensure the numerical rank is limited irrespective of the cluster size.

Firstly, the DCA is combined with the Macro Basis Function (MBF) method. In this technique, physics-based MBFs are defined over each antenna element, through linear combinations of the low-level basis functions defined on that domain, in order to create a reduced matrix system, which can then be solved directly. However, one of the MBF solvers' bottlenecks is the high cost associated with the computation of reaction terms during the fill-in of the reduced matrix. As such, the DCA algorithm is used to efficiently represent and compute the reaction terms in MBF solvers. The accuracy of using the MBF-DCA solver is validated, and a favorable memory scaling is obtained.

Secondly, the single-level version of the DCA is formulated together with a sparse direct solver scheme, based on the Inverse Fast Multipole Method (IFMM), to solve for

antenna array MoM solution directly. The original IFMM formulation is extended for the directional case, and a new procedure to eliminate and redirect compressible fill-ins during the Gaussian elimination of the sparse matrix is devised.

Lastly, a hybrid single-level compression scheme is devised to accelerate the Iterative Radius-Based Domain Green's Function Method (IRB-DGFM) solver, for array analysis. The compression algorithm combines the standard Adaptive Cross Approximation (ACA) to compress intermediate interactions, and the single-level Nested Cross Approximation (NCA) to represent far interactions efficiently.

# Opsomming

## Aanpassingsvaardige Kruis-Benadering Metodes vir Vinnige Analise van Antenna Samestellings

K. Sewraj

*Departement van Elektriese en Elektroniese Ingenieurswese,  
Universiteit van Stellenbosch,  
Privaatsak X1, Matieland 7602, Suid-Afrika.*

Proefskrif: Ph.D

Maart 2021

Hierdie werk fokus op die ontwikkeling van doeltreffende numeriese elektromagnetika algoritmes vir die analise van groot antenna samestellings, soos oorweeg word as deel van die internasionale “Square Kilometre Array (SKA)” radiosterrekunde projek tans onder ontwikkeling. Numeriese elektromagnetika simulاسie is ’n onontbeerlike gereedskapstuk om antenna werkverrigting gedurende die ontwerpsproses te bepaal, aangesien duisende simulاسies dikwels benodig word tydens veelvuldige ontwerpsiterاسies. Sulke simulاسies is egter tipies duur en kan ’n beperkende faktor tot ontwerpskeuses wees.

Die moment metode (MoM) is ’n numeriese tegniek om elektromagnetiese veldprobleme op te los en is hoogs gepas vir stralingsprobleme soos die analise van antenna samestellings. Die geheue en looptyd vereistes van die MoM skaleer egter as  $\mathcal{O}(N^2)$  en  $\mathcal{O}(N^3)$ , onderskeidelik, waar  $N$  die aantal vryheidsgrade aandui. In die lig hiervan word elektromagnetiese analise met die MoM beperk deur die elektriese grootte van die probleem. Vir groter strukture moet versnelde MoM tegnieke wat pasgemaak is vir die spesifieke probleem, geskep word.

’n Verskeidenheid van tegnieke gebaseer op kruis-benadering word in hierdie werk ondersoek, vir samestellingsanalise. Binne hierdie konteks word twee Rigtingsgewyse Kruis-Benadering (RKB)-gebaseerde oplossers daargestel. Die RKB is ’n geneste multivlak algoritme wat MoM sub-blokke doeltreffend saampers as produkte van lae-rang faktore vir vêr-interاسies. Tydens die bepaling van hierdie faktore word die vêrveld gesegmenteer in hoeksgewyse sektore wat verseker dat die numeriese range beperk word, ongeag die tros-grootte.

Eerstens word die RKB gekombineer met die Makro Basis Funksie (MBF) metode. Met hierdie tegniek word fisika-gefundeerde MBFs gedefinieer oor elke antenna element, as linieêre kombinasies van die laevlak basis funksies op elke domein, om sodoende ’n gereduseerde matriks stelsel te skep, wat op direkte wyse opgelos kan word. ’n Bottelnek van die MBF-oplosser is egter die hoë koste verbonde aan die berekening van reaksie-terme met die vul van die gereduseerde matriks. In hierdie verband word die RKB algoritme gebruik om die MBF reaksie-terme doeltreffend te bereken. Die akkuraatheid van die MBF-RKB oplossing word nagegaan en gunstige geheue-skalering word gedemonstreer.

Tweedens word 'n yl-matriks direkte oplossingskema geformuleer wat geskoei is op die enkel-vlak weergawe van die RKB, vir direkte oplossing van antenna samestelling MoM matriksvergelykings. Dit is 'n aangepaste weergawe van die Inverse Vinnige Multipool Metode (IVMM). Die oorspronklike IVMM formulering word uitgebrei na die rigtingsge-wyse geval, en nuwe prosedures vir die eliminasië en herkanalisering van saampersbare invullings gedurende die Gaussiese eliminasië proses, word daargestel.

Laastens word 'n hibriede enkel-vlak saampersingskema daargestel om die Iteratiewe Radius-Gebaseerde Domein Green Funksie Metode (IRG-DGFM) oplosser te vermen, vir samestellingsanalise. Die saampersingskema kombineer die standaard Aanpassingsvaardige Kruis-Benadering (AKB) vir saampersing van intermediêre interaksies, met die enkel-vlak Geneste Kruis-Benadering (GKB) vir doeltreffende saampersing van vêr-interaksies.

# Acknowledgements

I would like to express my sincere gratitude to the following people:

- My supervisor, Prof. Matthys Botha, for his consistent support and encouragement throughout my years at Stellenbosch University. I am very grateful to have had the opportunity to learn from his expertise in computational electromagnetics.
- Prof. Christophe Craeye, for all the valuable advises on Macro Basis Functions, during my visit to the ICTEAM Institute.
- All the friends and colleagues in the electromagnetic research group at Stellenbosch University. I would like to thank Matthews for the various discussions regarding array analysis and collaborative work. Thanks to Robey, who is the author of the MoM library used in this thesis.
- My parents and sister, Deeya, for everything they have done and for always being here.
- My girlfriend, Tayla, for all her support during the writing up of this thesis.

This research was supported by the South African Radio Astronomy Observatory, which is a facility of the National Research Foundation, an agency of the Department of Science and Technology.

# Contents

<b>Declaration</b>	<b>iii</b>
<b>Abstract</b>	<b>v</b>
<b>Opsomming</b>	<b>vii</b>
<b>Acknowledgements</b>	<b>ix</b>
<b>Contents</b>	<b>x</b>
<b>List of Figures</b>	<b>xii</b>
<b>List of Tables</b>	<b>xiv</b>
<b>1 Introduction</b>	<b>1</b>
1.1 Motivation . . . . .	1
1.2 Computational Electromagnetics for Array Analysis . . . . .	2
1.2.1 Contributions . . . . .	3
1.2.2 Outline of the thesis . . . . .	3
1.3 List of appended papers . . . . .	4
1.3.1 Journal papers . . . . .	4
1.3.2 Conference papers . . . . .	4
<b>2 Electromagnetic Field Theory and the Method of Moments</b>	<b>5</b>
2.1 Introduction . . . . .	5
2.2 Electromagnetic Field Theory . . . . .	5
2.2.1 Boundary Conditions . . . . .	6
2.2.2 Relationship Between Radiating Field and Electric Current . . . . .	6
2.3 Electric Field Integral Equation . . . . .	8
2.4 Method of Moments . . . . .	8
2.4.1 Background . . . . .	8
2.4.2 Rao-Wilton-Glisson Basis Function . . . . .	9
2.4.3 Applying Galerkin Testing to the EFIE . . . . .	10
2.5 Conclusion . . . . .	11
<b>3 Macro Basis Functions</b>	<b>13</b>
3.1 Introduction . . . . .	13
3.2 Generation of MBFs . . . . .	13
3.3 Creating the Reduced Matrix . . . . .	14
3.3.1 Fast Computation of Reaction Terms . . . . .	15

3.4	Solving the Reduced Impedance Matrix . . . . .	17
3.5	Conclusion . . . . .	17
<b>4</b>	<b>Hierarchical Matrices</b>	<b>19</b>
4.1	Introduction . . . . .	19
4.1.1	HODLR and HSS . . . . .	21
4.1.1.1	Application to MBF Solvers . . . . .	21
4.1.2	$\mathcal{H}$ -Matrix . . . . .	22
4.1.2.1	Application to MBF Solvers . . . . .	23
4.1.3	$\mathcal{H}^2$ -Matrix . . . . .	23
4.2	Nested Cross Approximation . . . . .	24
4.2.1	Partitioning and Admissibility Condition . . . . .	24
4.2.2	Generation of NCA Factors . . . . .	25
4.2.2.1	Selecting Basis and Testing Functions for NCA Factors . . . . .	27
4.3	Directional Cross Approximation . . . . .	29
4.4	Conclusion . . . . .	31
<b>5</b>	<b>Fast Computation of MBF Reaction Terms with DCA Factors</b>	<b>33</b>
5.1	Introduction . . . . .	33
5.2	Memory requirement . . . . .	33
5.3	Using DCA for Fast Computation of MBF Reaction Terms . . . . .	34
5.4	Numerical Experiment Setup . . . . .	34
5.5	Numerical results . . . . .	36
5.6	Conclusion . . . . .	38
<b>6</b>	<b>Inverse Fast Multipole Method Based on DCA Factors</b>	<b>41</b>
6.1	Introduction . . . . .	41
6.2	IFMM . . . . .	41
6.2.1	Application of Non-Directional IFMM to the MoM in Combination with MBFs . . . . .	42
6.2.2	Application of Directional IFMM to the MoM . . . . .	42
6.3	Hybrid Single-Level ACA Compression for the IRB-DGFM . . . . .	43
6.4	Conclusion . . . . .	44
<b>7</b>	<b>Conclusion</b>	<b>47</b>
<b>A</b>	<b>Journal paper — DCA-based IFMM solver [1]</b>	<b>49</b>
<b>B</b>	<b>Journal paper — ACA acceleration for the IRB-DGFM [2]</b>	<b>61</b>
<b>C</b>	<b>Conference paper — Reaction Matrix Computation with NCA and DCA [3]</b>	<b>73</b>
<b>D</b>	<b>Conference paper — Reaction Matrix Computation with DCA [4]</b>	<b>79</b>
<b>E</b>	<b>Conference paper — Non-Directional IFMM with MBFs [5]</b>	<b>85</b>
<b>F</b>	<b>Conference paper — Overview of Group’s Results [6]</b>	<b>89</b>
	<b>List of References</b>	<b>93</b>

# List of Figures

1.1	Artist impression of the SKA MFAA. <b>Source:</b> <a href="http://www.skatelescope.org/aperture-arrays">www.skatelescope.org/aperture-arrays</a> . . . . .	2
2.1	Triangle pair with common edge, over which the $n$ -th RWG function is defined. . . . .	10
3.1	(a) Normalized singular values of MoM impedance matrix sub-blocks, generated by a linear array of 9 blade dipole antennas. Each sub-block represents the interaction between an antenna pair (self- or non-self-interaction). Self-interaction blocks are full-rank, while non-self-interaction blocks are rank deficient. (b) Blade dipole array configuration used to generate the MoM impedance matrix. The inter-element spacing is half wavelength. . . . .	16
4.1	Representation of a flat/non-hierarchical block matrix structure. . . . .	19
4.2	Structure of various hierarchical matrices: (a) HODLR, (b) HSS, (c) $\mathcal{H}$ matrix and (d) $\mathcal{H}^2$ matrix. . . . .	20
4.3	(a) Decomposition of a geometrical structure. (b) Block cluster tree which depicts the partitioning of source and observer clusters. Green dotted arrow represents interaction between admissible blocks (Low-ranked blocks); Red dotted arrow represents interaction between inadmissible blocks (Non-compressed blocks). . . . .	22
4.4	Binary partitioning of an irregular antenna array containing 256 elements. Each dot represents the position of an antenna element. The radius of the circular array is $15\lambda$ . . . . .	24
4.5	Depiction of RWG indices described in Table 4.2 at leaf level. Red RWG functions: Subset of RWGs within the local domain, the set of indices is denoted as $\tau_t$ (observer) or $\sigma_s$ (source). Blue RWG functions: Subset of RWGs within the far-field domain, the set of indices is denoted as $\sigma_t$ (observer) or $\tau_s$ (source). The set of indices on all the RWGs on the local domain is denoted as $t$ (observer) or $s$ (source). The antennas in the neighboring region are omitted for simplicity. . . . .	26
4.6	Local and far-field domains for the source and observer clusters. . . . .	26
4.7	Local domain of the child and parent clusters, and the far-field domain of the parent cluster. . . . .	27
4.8	Meshed bow-tie antenna. Red dots represent the midpoints of RWG functions; Blue dots represent the Chebyshev nodes. . . . .	28
4.9	Inverse source process to obtain directional cluster bases. . . . .	29
5.1	Mesh of bent LPDA antenna at the frequency of 750 MHz, resulting in 848 Dofs. . . . .	35
5.2	Antenna array configuration containing 256 elements with minimum inter-element distance of 0.9m. . . . .	36

5.3	MoM current coefficient for antenna array of 32 elements, and the current errors using MBF only and MBF-DCA solvers. . . . .	37
5.4	Electric far-field of antenna array of 32 elements using the MoM, and MBF only and MBF-DCA solvers. . . . .	37
5.5	Average number of neighbours and ranks for varying array sizes. . . . .	38
5.6	Memory requirement for the DCA, before and after combining with MBFs. Memory requirement for the reduced MBF matrix is also shown. . . . .	39
6.1	Mesh at 300 MHz, of a bent log-periodic dipole antenna, with dimensions $l = 468.05$ mm, $w = 926.10$ mm, $h = 270.23$ mm and $\theta = 120^\circ$ . . . . .	45
6.2	Storage requirements for the ACA IRB-DGFM solution of arrays with varying numbers of elements, of the bent log-periodic dipole antenna shown in Fig. 6.1, at 300 MHz. Each trace is obtained by varying $R_\tau$ and recording the total memory required for the ACA factors and then normalizing the data for each array with respect to the minimum storage achieved for that array. The actual minimum values are listed in Table 6.2. . . . .	45

# List of Tables

4.1	Description of the four hierarchical matrices. . . . .	20
4.2	Description of indices during the construction of cluster bases. . . . .	25
4.3	Description of indices during the construction of directional cluster bases. . . . .	30
5.1	Memory requirement for various antenna array sizes. . . . .	40
6.1	Runtime and memory requirements for Gaussian elimination of extended sparse matrices. . . . .	43
6.2	Memory requirements for solving bent log-periodic dipole antenna arrays of various sizes, at 300 MHz. For the ACA IRB-DGFM solutions the optimal value of $R_\tau = 8\lambda$ is used. . . . .	44

# Chapter 1

## Introduction

### 1.1 Motivation

The aim of this thesis is to develop fast and efficient numerical electromagnetic algorithms to analyze large antenna arrays. The primary motivation for fast array analysis in this research is due, but not limited, to the radio astronomy project, the Square Kilometre Array (SKA) [7]. The SKA is an international collaboration to build the next generation of highly sensitive radio telescopes in South Africa, Australia and other partner African countries listed in [8] to form part of the Very Long Baseline Interferometry (VLBI). Part of this project consists of arrays of individual antennas, called Aperture Arrays (AA), which will be used as a collective to act as a single radio telescope through the principle of interferometry such as in the Low Frequency Aperture Array (LFAA) [9] and the Mid Frequency Aperture Array (MFAA) [10] during the first and second phases of the SKA, respectively. The LFAA will operate on frequencies ranging from 50MHz to 350MHz and consisting of 1024 stations of approximately 35m diameter, each with 256 wideband LPDA antennas [11] arranged in an irregular configuration. The MFAA is expected to operate on a frequency range of 450MHz to 1450MHz, and the design is still in the research phase at the time of writing. Depending on the design, the inter-element spacing will be dense (less than a wavelength) or sparse, and the array will either have a regular or irregular configuration. Some of the proposed antenna elements for the MFAA are the Vivaldi [12], Orthogonal Ring Antenna (ORA) [13], Dense Dipole [14], and Log Periodic Dipole Array [15]. The number of antennas per station will vary depending on the choice of antenna and array configuration, however, a few thousands of antennas per station can be expected because of the high sensitivity requirement. An artist impression of the SKA MFAA stations is shown in Fig. 1.1.

To design antenna arrays, numerical electromagnetic simulation is a vital tool to validate the performance of the design. It is very common to iterate through thousands of simulations to fine tune the parameters of both the antenna geometry and array synthesis, to meet the desired criteria. However, electromagnetic solvers can often be a limiting factor in the available design choices since computational costs in terms of runtime and memory grow rapidly with problem size. The geometries of the antenna arrays to be used for the SKA are both electrically very large, that is in terms of wavelengths, and consist of fine structures. A single simulation of such an array with a naive approach might be expected to run for many months on a desktop workstation. As such, there is the need to devise tailored and efficient algorithms to deal with these large arrays, for instance, an algorithm targeted to analyze the stations of the LFAA is developed in



**Figure 1.1:** Artist impression of the SKA MFAA. **Source:** [www.skatelescope.org/aperture-arrays](http://www.skatelescope.org/aperture-arrays)

[16]. The number of antennas in an MFAA station is expected to be significantly larger, hence the continuing drive to improve fast algorithms and develop new ideas. Despite the SKA being the motivation for this research, it should be noted that developing these fast electromagnetic algorithms is beneficial to other applications of antenna arrays, such as for example telecommunications, radar systems and remote sensing.

## 1.2 Computational Electromagnetics for Array Analysis

The Method of Moments (MoM) [17] is a full-wave integral equation technique used to solve electromagnetic field problems. The MoM converts the integral equation, derived from Maxwell's equations and electromagnetic boundary conditions, into a linear matrix equation which can be solved numerically. The MoM matrix equation is commonly written as  $ZI = V$ , where  $Z$  is the impedance matrix,  $V$  is the excitation vector and  $I$  is the unknown current coefficient vector. Once  $I$  is known, the responses of interest such as the radiation pattern and input impedance can be computed. The MoM is highly suitable to radiation problems such as the analysis of antenna arrays due to the nature of the integral equation. Moreover, unlike differential equation based methods such as the Finite Element Method (FEM) [18], and Finite Difference Time Domain (FDTD) method [19], the MoM only requires surface discretization hence requiring fewer Degrees of Freedom (DoFs). However, the MoM produces a fully populated impedance matrix. As such, the fill-in time and memory requirement of the impedance matrix scale as  $\mathcal{O}(N^2)$  and the computational time to solve the MoM matrix equation through Gaussian elimination scales as  $\mathcal{O}(N^3)$ , where  $N$  refers to the number of DoFs of unknown current coefficients. Storing and solving the linear equation rapidly becomes impractical and therefore electromagnetic analysis performed by the MoM is limited by the electrical size of the problem.

For larger structures, a plethora of MoM-based fast techniques have been developed specialized to address different problems. Non-asymptotic fast methods can be classified as either being iterative or direct solvers. The use of Krylov subspace iterative methods, together with fast techniques to accelerate matrix-vector products in the solver such as the Fast Multipole Method (FMM) [20] and its multilevel version [21], among others, have

successfully been used in computational electromagnetics, especially for large scattering problems.

For antenna array analysis, direct methods are preferred when simulating multiple excitation schemes. In iterative solvers, we have to start anew for each excitation scheme. One such direct solution which has successfully been used to analyze large antenna arrays, is the Macro Basis Function (MBF) [22] method. In this technique, physics-based MBFs are defined over each antenna element, through linear combinations of the low-level basis functions defined on that domain. Since the number of MBFs defined can be fewer than the initial DoFs by a few orders of magnitude, the MBFs are used to create a reduced matrix system, which can then be solved directly. However, computing reaction terms to fill the reduced matrix system is still costly, and has been the focus of various research works. In this thesis, new methods to efficiently create the reduced MBF system will be explored. As an alternative approach to fast direct array analysis, a sparse direct solver is formulated, which is based upon the recently-proposed Inverse Fast Multipole Method (IFMM) scheme [23, 24].

A general advancement for fast integral-equation based solvers are the hierarchical matrix structures. Hierarchical matrices exploit the data-sparse structure of the MoM impedance matrix, and block entries due to far-interactions are compressed efficiently into low-rank factors. These representations broadly form the cornerstone of all the work presented in this thesis. The standard Adaptive Cross Approximation (ACA) [25, 26], the Nested Cross Approximation (NCA) [27] and Directional Cross Approximation (DCA) [28], which are hierarchical matrices, are employed.

### 1.2.1 Contributions

The following research contributions to fast array analysis have been made:

- Use of the DCA to compute reaction terms for MBF solvers. This work is partly documented in [3, 4]. The author intends to write a journal paper on this work in the near future.
- A sparse direct solver based on the single level version of the DCA is formulated and evaluated. This work is documented in [1, 5, 6].
- A single-level omnidirectional cross approximation scheme is formulated to accelerate an Iterative Radius-Based Domain Green's Function Method (IRB-DGFM) for array analysis, developed by M. Chose. This work is documented in [2, 6].

### 1.2.2 Outline of the thesis

- **Chapter 2:** Electromagnetic field theory and MoM formulation relevant to this thesis are reviewed. The Electric Field Integral Equation (EFIE) which relates the incident electric field to the unknown surface current is derived. The EFIE is discretized and the MoM matrix equation is formulated.
- **Chapter 3:** The background and formulation of the MBF solver for array analysis are reviewed.
- **Chapter 4:** Various hierarchical matrices from the literature are reviewed. Their application to accelerate the computation of reaction terms in the MBF solver is discussed. The DCA is found suitable for this task.

- **Chapter 5:** The DCA algorithm is used for fast computation of reaction terms in the MBF solver, and the reduced impedance matrix is efficiently created. Numerical results are presented to assess the performance.
- **Chapter 6:** The single-level DCA factors are used as inputs to a sparse direct solver, based on the Inverse Fast Multipole Method (IFMM) [23, 24] to solve for the MoM solution. The formulation of the original IFMM is extended to the directional case. Related to the above single-level compression scheme (but not in a direct solver context), a hybrid single-level ACA-based compression scheme is formulated to accelerate the IRB-DGFM. The compression scheme is based on the single-level NCA and the ACA.
- **Chapter 7:** Conclusion.

### 1.3 List of appended papers

The details of this research work is presented in two journal papers [1, 2] and four conference papers [3, 4, 5, 6] and are included as Appendices. The appended papers are:

#### 1.3.1 Journal papers

1. **Appendix A: A Sparse Direct Solver Based on Directional Cross Approximation for Antenna Array Analysis [1]:** The single-level DCA is combined with a newly formulated directional version of the IFMM, to solve for the MoM solution directly.
2. **Appendix B: Iterative Radius-Based Domain Green's Function Method with ACA for Antenna Array Analysis [2]:** This paper is primarily the work of M. Chose to formulate an IRB-DGFM algorithm for array analysis. The author's contribution is the formulation of a hybrid compression scheme to accelerate the IRB-DGFM algorithm. The hybrid scheme consist of the ACA and the single-level NCA. The single-level NCA is termed as translation-based ACA (TACA) in the paper.

#### 1.3.2 Conference papers

1. **Appendix C: Computation of MBF reaction matrices for antenna array analysis, with a directional method [3]:** A multilevel DCA algorithm is used to rapidly compute the reaction terms in MBF solvers.
2. **Appendix D: Directional method to compute reduced matrix system in MBF solvers [4]:** Further results of the DCA-MBF solver are presented.
3. **Appendix E: Application of the inverse fast multipole method to antenna array analysis [5]:** The factors of the single-level NCA are used in an IFMM solver to solve for the MoM solution.
4. **Appendix F: Developments in the analysis of large antenna arrays with disjoint elements [6]:** Overview conference paper of work on array analysis from the authors. The factors of the single-level NCA are reduced with MBF definitions, and then used in an IFMM solver to solve for the MBF solution.

## Chapter 2

# Electromagnetic Field Theory and the Method of Moments

### 2.1 Introduction

In this chapter, the background of electromagnetic field theory necessary for the numerical analysis of antenna arrays is introduced. The EFIE is derived, which describes the relationship between the surface current over the antennas (assumed Perfect Electric Conductor (PEC)) and the incident electromagnetic field. The aim is to solve for the unknown surface current in the integrand of the EFIE. However, since no analytical solution exists for arbitrary shapes, the EFIE is solved numerically. As such, the Method of Moments (MoM) [17] is introduced. The MoM is a powerful numerical technique that converts the EFIE through discretization and weighting, into a linear matrix equation that can be solved numerically. In this thesis, an inhouse MoM library [29] is used. The MoM function is called when MoM matrix entries are needed in the solvers discussed in later chapters.

### 2.2 Electromagnetic Field Theory

Maxwell's set of equations form the foundation of the classical electromagnetic theory. These equations describe the electric and magnetic fields which arise due to electric charges, currents. The differential form of Maxwell's equations in the frequency domain [30, 31] in a homogeneous medium with permittivity,  $\epsilon$ , and permeability,  $\mu$ , are as follows:

$$\nabla \times \mathbf{E} = -j\omega\mu\mathbf{H} \quad (2.1)$$

$$\nabla \times \mathbf{H} = \mathbf{J} + j\omega\epsilon\mathbf{E} \quad (2.2)$$

$$\nabla \cdot \mathbf{D} = q_e \quad (2.3)$$

$$\nabla \cdot \mathbf{B} = 0. \quad (2.4)$$

The field quantities in (2.1) to (2.4) are time-varying and phasors, and their time dependence,  $e^{j\omega t}$ , is suppressed for simplicity. The quantities featuring in (2.1) to (2.4) are defined as follows:

- $\mathbf{E}$ : Electric field intensity
- $\mathbf{H}$ : Magnetic field intensity

- $\mathbf{D}$ : Electric flux density
- $\mathbf{B}$ : Magnetic flux density
- $\mathbf{J}$ : Electric current density
- $q_e$ : Electric charge density.

The relations between electric and magnetic fields and their respective flux densities are as follows:

$$\mathbf{D} = \epsilon \mathbf{E} \quad (2.5)$$

$$\mathbf{B} = \mu \mathbf{H}. \quad (2.6)$$

### 2.2.1 Boundary Conditions

Electromagnetic boundary conditions describe how the fields and flux densities behave at the interface between two media. Boundary conditions are required so as Maxwell's set of equations has a unique solution in order to construct the EFIE. In the formulation of the EFIE for the MoM derived in this chapter, the metallic surfaces of the antennas are considered to be PEC, hence only the boundary conditions between PEC and free space are relevant, as follows:

$$-\hat{n} \times \mathbf{E} = 0 \quad (2.7)$$

$$\hat{n} \times \mathbf{H} = \mathbf{J}_s \quad (2.8)$$

$$\hat{n} \cdot \mathbf{D} = q_e \quad (2.9)$$

$$\hat{n} \cdot \mathbf{B} = 0 \quad (2.10)$$

where  $\hat{n}$  is the unit normal at the interface pointing into the free space region and  $\mathbf{J}_s$  is the electric surface current density.

### 2.2.2 Relationship Between Radiating Field and Electric Current

In this section, an expression is derived that allows the computation of the radiated electric field everywhere in space due to an electric current distribution. The magnetic vector potential is used as an intermediate to solve for the radiated field. The magnetic vector potential is described by an integral of the electric current, and the electric radiated field is then evaluated directly from the potential.

The magnetic vector potential is first defined. Since, the magnetic flux is solenoidal (2.4), it can be written as the curl of an arbitrary vector,  $\mathbf{A}$ , as in (2.11).  $\mathbf{A}$  is referred to as the magnetic vector potential.

$$\nabla \cdot (\nabla \times \mathbf{A}) = 0. \quad (2.11)$$

The magnetic field intensity in Faraday's law (2.1) is substituted by  $\mathbf{H} = \frac{1}{\mu} \nabla \times \mathbf{A}$  using (2.6) and  $\mathbf{B}$  is then replaced by the curl of  $\mathbf{A}$ , to obtain

$$\nabla \times \mathbf{E} = -j\omega\mu \mathbf{H} = -j\omega \nabla \times \mathbf{A}. \quad (2.12)$$

Equation (2.12) can be re-written as

$$\nabla \times (\mathbf{E} + j\omega\mathbf{A}) = 0. \quad (2.13)$$

A curl-free vector field can be expressed as the gradient of an arbitrary scalar potential, therefore

$$\mathbf{E} = -j\omega\mathbf{A} - \nabla\phi_e \quad (2.14)$$

where  $\phi_e$  is a scalar defined as the electric scalar potential. Equation (2.14) is an expression of the radiated field in terms of the vector magnetic potential and scalar electric potential. By taking the curl on both sides of  $\mu\mathbf{H} = \nabla \times \mathbf{A}$  (from (2.6) and (2.11)) and using the identity

$$\nabla \times \nabla \times \mathbf{A} = \nabla(\nabla \cdot \mathbf{A}) - \nabla^2 \mathbf{A} \quad (2.15)$$

the following equation is obtained

$$\mu\nabla \times \mathbf{H} = \nabla(\nabla \cdot \mathbf{A}) - \nabla^2 \mathbf{A}. \quad (2.16)$$

Equation (2.16) is equated to the Maxwell-Ampere's law (2.2), leading to

$$\mu\mathbf{J} + j\omega\mu\epsilon\mathbf{E} = \nabla(\nabla \cdot \mathbf{A}) - \nabla^2 \mathbf{A}. \quad (2.17)$$

Substituting (2.14) into (2.17) leads to

$$\mu\mathbf{J} + j\omega\mu\epsilon(-j\omega\mathbf{A} - \nabla\phi_e) = \nabla(\nabla \cdot \mathbf{A}) - \nabla^2 \mathbf{A}. \quad (2.18)$$

Rearranging (2.18) yields

$$\nabla^2 \mathbf{A} + k^2 \mathbf{A} = -\mu\mathbf{J} + \nabla(\nabla \cdot \mathbf{A} + j\omega\mu\epsilon\phi_e) \quad (2.19)$$

where  $k$  is the wavenumber,  $k = \omega\sqrt{\mu\epsilon}$ . Only the curl of  $\mathbf{A}$  is defined so far. Its divergence may be freely defined as long as the definition is consistent. To simplify (2.19), the *Lorenz gauge condition* is used, that is

$$\nabla \cdot \mathbf{A} = -j\omega\epsilon\mu\phi_e, \quad (2.20)$$

thus simplifying (2.19) into an inhomogeneous vector Helmholtz equation of  $\mathbf{A}$  as

$$\nabla^2 \mathbf{A} + k^2 \mathbf{A} = -\mu\mathbf{J}. \quad (2.21)$$

The solution of the inhomogeneous vector Helmholtz equation can be written as a convolution of the current,  $\mathbf{J}$ , and the Green's function  $G(\mathbf{r}, \mathbf{r}')$ , such as

$$\mathbf{A}(\mathbf{r}) = \mu \iiint_V \mathbf{J}(\mathbf{r}')G(\mathbf{r}, \mathbf{r}')d\mathbf{r}' \quad (2.22)$$

where  $\mathbf{r}'$  and  $\mathbf{r}$  are source and observer position vectors, respectively. The Green's function is defined as

$$G(\mathbf{r}, \mathbf{r}') = \frac{e^{-jk|\mathbf{r}-\mathbf{r}'|}}{4\pi|\mathbf{r}-\mathbf{r}'|}. \quad (2.23)$$

By making  $\phi_e$  the subject of (2.20) and substituting into (2.14), an expression describing the electric field in terms of  $\mathbf{A}$  can be written as

$$\mathbf{E} = -j\omega\mathbf{A} - \nabla\phi_e = -j\omega\mathbf{A} - \frac{j}{\omega\mu\epsilon}\nabla(\nabla \cdot \mathbf{A}). \quad (2.24)$$

The electric field due to the current distribution can now be computed as a superposition integral, by substituting (2.22) into (2.24), yielding

$$\mathbf{E}(\mathbf{r}) = -j\omega\mu \iiint_V \mathbf{J}(\mathbf{r}')G(\mathbf{r}, \mathbf{r}')d\mathbf{r}' - \frac{1}{k^2}\nabla \iiint_V \nabla' \cdot \mathbf{J}(\mathbf{r}')G(\mathbf{r}, \mathbf{r}')d\mathbf{r}'. \quad (2.25)$$

## 2.3 Electric Field Integral Equation

In Section 2.2.2, a general expression for the electric field everywhere in free-space due to an electric current distribution was derived. In this section, the EFIE is derived which relates the unknown surface current,  $\mathbf{J}_s$ , to the known incident electric field,  $\mathbf{E}^i$ . The total electric field,  $\mathbf{E}$ , can be expressed as

$$\mathbf{E} = \mathbf{E}^s + \mathbf{E}^i \quad (2.26)$$

where  $\mathbf{E}^s$  is the scattered electric field and  $\mathbf{E}^i$  is the incident electric field. By restricting  $\mathbf{r}$  to the PEC surface of the antenna and using the boundary condition in (2.7), the following can be obtained:

$$-\hat{n} \times \mathbf{E} = -\hat{n} \times \mathbf{E}^s - \hat{n} \times \mathbf{E}^i = 0 \quad (2.27)$$

$$\hat{n} \times \mathbf{E}^s = -\hat{n} \times \mathbf{E}^i. \quad (2.28)$$

The scattered field is the field radiated by the induced surface current,  $\mathbf{J}_s$ . Thus, restricting the observation point in (2.25) to locations on the PEC surface, and further substituting it into (2.28), yields the final EFIE:

$$\hat{n} \times \mathbf{E}^i(\mathbf{r}) = \hat{n} \times [j\omega\mu \iint_S \mathbf{J}_s(\mathbf{r}')G(\mathbf{r}, \mathbf{r}')d\mathbf{r}' + \frac{1}{k^2} \nabla \iint_S \nabla' \cdot \mathbf{J}_s(\mathbf{r}')G(\mathbf{r}, \mathbf{r}')d\mathbf{r}']. \quad (2.29)$$

During the rest of this thesis, the aim is to solve for the unknown surface current,  $\mathbf{J}_s$ , in (2.29) for large antenna arrays, using the MoM, as described next.

## 2.4 Method of Moments

In this section, the Method of Moments (MoM) is introduced. By using basis and testing functions, the MoM allows for converting the EFIE into a matrix format which can be solved numerically using a computer.

### 2.4.1 Background

Consider the general problem:

$$\mathcal{L}f = g \quad (2.30)$$

where  $\mathcal{L}$  is a linear integro-differential operator,  $g$  is the excitation source, and  $f$  is the unknown function to be solved for. It is assumed that (2.30) cannot be solved analytically. To solve for the unknown function,  $f$ , using a numerical technique, the problem must first be discretized with finite elements. The function,  $f$ , can be approximated into a sum of weighted basis functions,  $f_n$ , such as

$$f \approx \sum_{n=1}^N \alpha_n f_n \quad (2.31)$$

where  $\alpha_n$  are unknown weighting coefficients to be solved for. Substituting (2.31) into (2.30) leads to

$$\sum_{n=1}^N \alpha_n \mathcal{L}f_n \approx g. \quad (2.32)$$

To enforce the boundary condition equality, the fields are integrated over testing functions,  $f_m$ , such as

$$\sum_{n=1}^N \alpha_n \langle f_m, \mathcal{L}f_n \rangle = \langle f_m, g \rangle. \quad (2.33)$$

The integration process is defined using an inner product,  $\langle \cdot \rangle$ , in (2.33). Enforcing boundary condition in the case of the EFIE refers to an averaged zero tangential electric field on the testing function domain. Equation (2.33) can be expressed as the MoM matrix equation:

$$ZI = V \quad (2.34)$$

where  $Z$  is the impedance matrix,  $V$  is the right-hand side excitation vector and  $I$  is the unknown coefficient vector. The matrix and vectors entries in (2.34) are expressed as

$$Z_{mn} = \langle f_m, \mathcal{L}f_n \rangle \quad (2.35)$$

$$V_m = \langle f_m, g \rangle \quad (2.36)$$

$$I_n = \alpha_n \quad (2.37)$$

where  $m, n \in \{1, 2, \dots, N\}$  and  $N$  is the total number of basis and testing functions defined over the geometry. The impedance matrix,  $Z$ , is a fully populated matrix since each basis function interacts with all testing functions (self and non-self). The fill-in time and memory cost of  $Z$  scale as  $\mathcal{O}(N^2)$ . The runtime to solve (2.34) using Gaussian elimination scales as  $\mathcal{O}(N^3)$ .

### 2.4.2 Rao-Wilton-Glisson Basis Function

In the previous section, a generalized overview of the MoM was given. In this section, the EFIE will be discretized with finite elements, and basis and testing functions will be employed to create the MoM matrix equation. The first step is to approximate the surface current,  $\mathbf{J}_s$ , as a sum of weighted basis functions as was done for the generalized unknown function,  $f$ , in (2.31). It can be approximated as

$$\mathbf{J}_s \approx \sum_{n=1}^N J_n \mathbf{f}_n \quad (2.38)$$

where  $J_n$  is the  $n$ -th unknown current coefficient and the basis function,  $\mathbf{f}_n$ , is vector-valued. The Rao-Wilton-Glisson basis function [32], commonly referred to as the RWG basis function, will be used after discretizing the entire surface with a conformal mesh of flat triangles. The RWG function is defined over two adjacent triangles with a common edge, shown in Fig. 2.1, and the  $n$ -th RWG function is formally described as

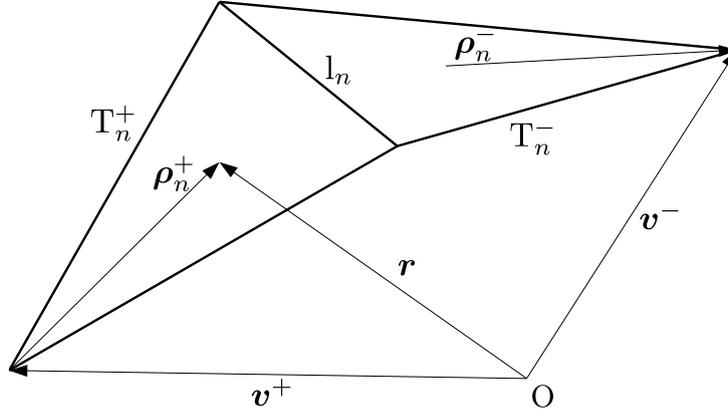
$$\mathbf{f}_n(\mathbf{r}) = \begin{cases} \frac{l_n}{2A_n^+} \boldsymbol{\rho}_n^+ & \forall \mathbf{r} \text{ in } T_n^+ \\ \frac{l_n}{2A_n^-} \boldsymbol{\rho}_n^- & \forall \mathbf{r} \text{ in } T_n^- \\ 0 & \text{otherwise} \end{cases} \quad (2.39)$$

where  $T_n^+$  and  $T_n^-$  are triangles with the shared  $n$ -th edge as shown in Fig 2.1.  $A_n^\pm$  are the areas of triangles  $T_n^\pm$  and  $l_n$  is the length of the  $n$ -th edge. The factors  $\frac{l_n}{2A_n^\pm}$  are normalizing constants in (2.39). And,  $\boldsymbol{\rho}_n^\pm$  are position vectors within the triangles  $T_n^\pm$  and are defined as

$$\boldsymbol{\rho}_n^+ = \mathbf{r} - \mathbf{v}^+ \quad \mathbf{r} \text{ in } T_n^+ \quad (2.40)$$

$$\boldsymbol{\rho}_n^- = \mathbf{v}^- - \mathbf{r} \quad \mathbf{r} \text{ in } T_n^- \quad (2.41)$$

where  $\mathbf{v}^\pm$  are the free vertices of triangles  $T_n^\pm$ .



**Figure 2.1:** Triangle pair with common edge, over which the  $n$ -th RWG function is defined.

The RWG functions have the following properties:

1. The RWG function has no normal components to the non-shared edges on both triangles.
2. The component of the RWG function normal to the shared edge is unity.
3. The surface divergence of  $\mathbf{f}_n(\mathbf{r})$  within triangles  $T_n^+$  and  $T_n^-$  are  $-\frac{l_n}{A_n^+}$  and  $\frac{l_n}{A_n^-}$ , respectively. Since the divergence of the current is proportional to the surface charge density, the total charge associated with an RWG function is zero.

### 2.4.3 Applying Galerkin Testing to the EFIE

In the previous section, the RWG basis function was employed to approximate the surface current,  $\mathbf{J}_s$ . The next step is to choose a testing function to enforce the boundary condition. There is freedom in choosing the testing function, however, this choice will have an impact on the accuracy and also the simplicity of the formulation. In the MoM, it is very common practice to use the Galerkin formulation, that is using identical basis and testing functions. Basis and testing RWG functions are employed to the EFIE. The resulting equation for the  $m$ -th testing function is

$$\begin{aligned} \iint_{T_m} \mathbf{f}_m(\mathbf{r}) \cdot \mathbf{E}^i(\mathbf{r}) ds &= j\omega\mu \sum_{n=1}^N J_n \left[ \iint_{T_m} \mathbf{f}_m(\mathbf{r}) \cdot \iint_{T_n} \mathbf{f}_n(\mathbf{r}') G(\mathbf{r}, \mathbf{r}') ds' ds \right] \\ &+ \sum_{n=1}^N J_n \left[ \frac{1}{k^2} \iint_{T_m} \mathbf{f}_m(\mathbf{r}) \cdot \left[ \nabla \iint_{T_n} \nabla' \cdot \mathbf{f}_n(\mathbf{r}') G(\mathbf{r}, \mathbf{r}') ds' \right] ds \right]. \end{aligned} \quad (2.42)$$

Using the vector identity

$$\mathbf{f}(\mathbf{r}) \cdot \nabla S(\mathbf{r}) = \nabla \cdot [\mathbf{f}(\mathbf{r}) S(\mathbf{r})] - [\nabla \cdot \mathbf{f}(\mathbf{r})] S(\mathbf{r}) \quad (2.43)$$

the expression

$$\iint_{T_m} \mathbf{f}_m(\mathbf{r}) \cdot \nabla S(\mathbf{r}) ds = \iint_{T_m} \mathbf{f}_m(\mathbf{r}) \cdot \left[ \nabla \iint_{T_n} \nabla' \cdot \mathbf{f}_n(\mathbf{r}') G(\mathbf{r}, \mathbf{r}') ds' \right] ds \quad (2.44)$$

can be re-written as

$$\iint_{T_m} \mathbf{f}_m(\mathbf{r}) \cdot \nabla S(\mathbf{r}) ds = \iint_{T_m} \nabla \cdot [\mathbf{f}_m(\mathbf{r}) S(\mathbf{r})] ds - \iint_{T_m} [\nabla \cdot \mathbf{f}_m(\mathbf{r})] S(\mathbf{r}) ds \quad (2.45)$$

The first term on the right hand side of (2.45) can be transformed using the 2D divergence theorem. The term goes to zero when the normal component of the integrand is integrated over the contour of the triangles supporting the RWG function.

$$\iint_{T_m} \nabla \cdot [\mathbf{f}_m(\mathbf{r}) S(\mathbf{r})] ds = \int_C \hat{\mathbf{n}} \cdot [\mathbf{f}_m(\mathbf{r}) S(\mathbf{r})] dl = 0 \quad (2.46)$$

Hence, (2.42) can be re-written as

$$\begin{aligned} \iint_{T_m} \mathbf{f}_m(\mathbf{r}) \cdot \mathbf{E}^i(\mathbf{r}) ds &= j\omega\mu \sum_{n=1}^N J_n \left[ \iint_{T_m} \mathbf{f}_m(\mathbf{r}) \cdot \iint_{T_n} \mathbf{f}_n(\mathbf{r}') G(\mathbf{r}, \mathbf{r}') ds' ds \right] \\ &- \sum_{n=1}^N J_n \left[ \frac{1}{k^2} \iint_{T_m} \nabla \cdot \mathbf{f}_m(\mathbf{r}) \left( \iint_{T_n} \nabla' \cdot \mathbf{f}_n(\mathbf{r}') G(\mathbf{r}, \mathbf{r}') ds' \right) ds \right]. \end{aligned} \quad (2.47)$$

This must hold for every testing function,  $m = 1, \dots, N$ . The  $N$  equations can be formatted into MoM matrix equation,  $ZI = V$ , as in (2.34). The  $Z_{mn}$  entry in the MoM impedance matrix,  $Z$ , can then be written as

$$Z_{mn} = \iint_{T_m} \iint_{T_n} \left[ \mathbf{f}_m(\mathbf{r}) \cdot \mathbf{f}_n(\mathbf{r}') - \frac{1}{k^2} \right] G(\mathbf{r}, \mathbf{r}') ds' ds \quad (2.48)$$

and, the  $m$ -th entry in the excitation vector,  $V$ ,

$$V_m = \frac{-j}{\omega\mu} \iint_{T_m} \mathbf{f}_m(\mathbf{r}) \cdot \mathbf{E}^i(\mathbf{r}) ds. \quad (2.49)$$

Finally, the  $m$ -th entry of the solution vector is

$$I_m = J_m. \quad (2.50)$$

## 2.5 Conclusion

Starting with Maxwell's set of equations and the electromagnetic boundary conditions, the EFIE is derived, which relates the unknown surface current to the incident electric field intensity. The EFIE cannot be solved analytically for arbitrary structures, hence the MoM, which is a numerical technique, is employed. After discretizing the geometry with triangular patches and using RWG basis and testing functions, the MoM matrix equation,  $ZI = V$ , is derived from the EFIE.



# Chapter 3

## Macro Basis Functions

### 3.1 Introduction

In this chapter, the MBF technique [22] is introduced. MBFs belong to a class of domain decomposition techniques that can be used to analyze large antenna arrays directly (non-iteratively). In these techniques, MBFs are defined over each antenna in the array by aggregating linear combinations of low-level basis functions (e.g. RWGs). Since the number of MBFs is fewer than the initial DoFs, the technique results in easing the computational burden by creating a reduced matrix system which can be solved directly. Various ways of constructing the physics-based MBFs have been proposed in the literature. Some of these are the Characteristic Basis Function Method (CBFM) [33], Synthetic-Function Approach (SFX) [34], Eigencurrent method [35], the sub-entire-domain basis function method [36] (the “windowed MBF” scheme proposed in [37] is very similar), and iteratively refined MBFs [38], among others.

Equation (3.1) is the partitioned MoM matrix equation resulting from an antenna array containing  $M$  elements. Each subdomain after the domain decomposition is an antenna element.

$$\begin{bmatrix} Z_{11} & Z_{12} & \dots & Z_{1M} \\ Z_{21} & Z_{22} & \dots & Z_{2M} \\ \vdots & \vdots & \ddots & \vdots \\ Z_{M1} & Z_{M2} & \dots & Z_{MM} \end{bmatrix} \begin{bmatrix} I_1 \\ I_2 \\ \vdots \\ I_M \end{bmatrix} = \begin{bmatrix} V_1 \\ V_2 \\ \vdots \\ V_M \end{bmatrix}. \quad (3.1)$$

The block-diagonal entries,  $Z_{aa}$  ( $a \in \{1, 2, \dots, M\}$ ), in the partitioned impedance matrix (3.1) represent the self-interactions of isolated subdomains. And, non block-diagonal entries,  $Z_{ab}$  ( $a, b \in \{1, 2, \dots, M\}, a \neq b$ ), are the mutual interactions between two subdomains. An MBF solver consists of 3 steps, namely:

1. generation of MBFs;
2. creating the reduced matrix;
3. solving the reduced matrix.

### 3.2 Generation of MBFs

MBFs are generated over each antenna and can be classified as primary, secondary [33] and higher-order MBFs. A primary MBF takes into account the effect of self-interaction of

an antenna, that is, the current coefficient of an isolated radiating antenna. The primary MBF,  $I_{\text{prim}}$ , is obtained as:

$$Z_{ii}I_{\text{prim}} = V_i. \quad (3.2)$$

Secondary MBFs take into account the effect of mutual coupling between neighboring antennas. The secondary current is the induced current by the radiated field produced by primary MBFs belonging to neighboring antennas within a pre-defined radius. The secondary MBF,  $I_{\text{sec}}$ , can be obtained as:

$$Z_{jj}I_{\text{sec}} = -Z_{ji}I_{\text{prim}}. \quad (3.3)$$

Higher-order MBFs (e.g. tertiary MBFs [39]) are obtained by allowing MBFs of a lower order (e.g. secondary MBF) to radiate onto neighboring antennas. The MBFs belonging to an antenna are then grouped in a matrix,  $\tilde{J}$ , such that

$$\tilde{J} = [I_{\text{prim}} \ I_{\text{sec}}^{\text{all}} \ I_{\text{HO}}^{\text{all}} \ \dots], \quad (3.4)$$

where  $I_{\text{sec}}^{\text{all}}$  is a matrix of size  $N_i \times K_{\text{sec}}$  containing all the secondary MBFs, where  $K_{\text{sec}}$  and  $N_i$  are the number of secondary MBFs and low-level basis function defined on the antenna, respectively. And,  $I_{\text{HO}}^{\text{all}}$  contains all the higher-order MBFs, with size  $N_i \times K_{\text{HO}}$  where  $K_{\text{HO}}$  is the number of higher-order MBFs. A Singular Value Decomposition (SVD) operation is then performed on  $\tilde{J}$ , such that

$$\tilde{J} = \tilde{U}\Sigma\tilde{V}^H, \quad (3.5)$$

where  $\tilde{U}$  and  $\tilde{V}$  are unitary matrices, and  $H$  in the superscript refers to the Hermitian transpose.  $\Sigma$  is a diagonal matrix containing singular values of  $\tilde{J}$  and are ordered in descending order. The  $n$ -th column of the  $\tilde{U}$  matrix corresponds to the  $n$ -th singular value in  $\Sigma$ . The singular values are normalized, and entries greater than a pre-defined threshold,  $\tau_{\text{SVD}}$ , are retained. The number of selected singular values is denoted as  $K$ . The final set of MBF,  $J$ , is the first  $K$ -th columns of  $\tilde{U}$ , such that

$$J = [\tilde{U}^1 \ \dots \ \tilde{U}^K]. \quad (3.6)$$

The SVD operation is performed to generate orthogonal basis functions to avoid ill-conditioning. Moreover, by trimming out redundant MBFs, only a minimum number of MBFs are kept which further reduce the matrix system. For arrays of identical and disjoint elements, the primary MBF is computed only once for all antennas. However, secondary and higher-order MBFs are computed individually for every antenna. In the case of regular arrays, the symmetry of the array can be exploited by computing non-primary MBFs for only a few scenarios (e.g. central and the various edge element positions).

### 3.3 Creating the Reduced Matrix

The MBFs obtained for each antenna are used to create a reduced matrix system which can be solved directly. The block entries of the reduced matrix are obtained by pre and post-multiplying impedance matrix blocks with the corresponding MBF definitions. The reduced impedance matrix equation can be written as:

$$\begin{bmatrix} \langle J_1^T, Z_{11}J_1 \rangle & \langle J_1^T, Z_{12}J_2 \rangle & \dots & \langle J_1^T, Z_{1M}J_M \rangle \\ \langle J_2^T, Z_{21}J_1 \rangle & \langle J_2^T, Z_{22}J_2 \rangle & \dots & \langle J_2^T, Z_{2M}J_M \rangle \\ \vdots & \vdots & \ddots & \vdots \\ \langle J_M^T, Z_{M1}J_1 \rangle & \langle J_M^T, Z_{M2}J_2 \rangle & \dots & \langle J_M^T, Z_{MM}J_M \rangle \end{bmatrix} \begin{bmatrix} I_1^{\text{MBF}} \\ I_2^{\text{MBF}} \\ \vdots \\ I_M^{\text{MBF}} \end{bmatrix} = \begin{bmatrix} \langle J_1^T, V_1 \rangle \\ \langle J_2^T, V_2 \rangle \\ \vdots \\ \langle J_M^T, V_M \rangle \end{bmatrix}. \quad (3.7)$$

The size of the reduced impedance matrix is  $MK \times MK$ , instead of  $MN_i \times MN_i$  for the original MoM matrix, where  $M$  is the number of antennas in the array. Since  $K \ll N_i$ , the reduced impedance matrix can be of one or two orders of magnitude smaller than the original MoM matrix [40, 22].

Creating the reduced matrix is the most time consuming step of an MBF solver. This step is not expensive memory-wise since the entire problem is divided into subdomains. Referring to (3.7), the block entry of the reduced impedance matrix for the interaction between antennas  $p$  and  $q$  can be expressed as  $Z_{pq}^{\text{Red}} = \langle J_p^T, Z_{pq} J_q \rangle$ . Thus, the memory requirement is  $\mathcal{O}(N_i^2)$ . This operation needs to be performed  $M^2$  times (every possible interaction among subdomains), leading to a computational time scaling of  $\mathcal{O}(M^2 N_i^2 K)$ . The computational cost of creating the reduced impedance matrix is not much of an improvement compared to the MoM. However, as the source and observer antennas become electrically well-separated, fewer DoFs can be used to represent the interaction, that is, sub-blocks for far interacting antennas are low-ranked. This is shown in Figure 3.1, where the normalized singular values (obtained after normalizing the entries of the diagonal matrix in an SVD operation) for different sub-blocks in the MoM impedance matrix is shown. The singular values indicate the numerical rank of a matrix. The self-interaction of a subdomain (an antenna element) is full ranked, and as the distance between the source and observer subdomains increases, the sub-block becomes low ranked.

From literature [22], both the transpose and Hermitian transpose of the MBF ( $J^T$  and  $J^H$ , respectively) have been used to compute the entries of the reduced impedance matrix and the reduced excitation vector. In this thesis, the transpose operation is preferred. This is because, the observer and source ‘cluster bases’ (to be defined in Chapter 4) of an antenna are transposed. Using the transpose of the MBF when creating ‘reduced cluster bases’ (to be defined in Chapter 5), the observer and source ‘reduced cluster bases’ of an antenna will also be transposed. Hence, only the observer ‘cluster basis’ and ‘reduced cluster basis’ of an antenna need to be computed, and the source counterpart can be obtained through a transposed operation.

### 3.3.1 Fast Computation of Reaction Terms

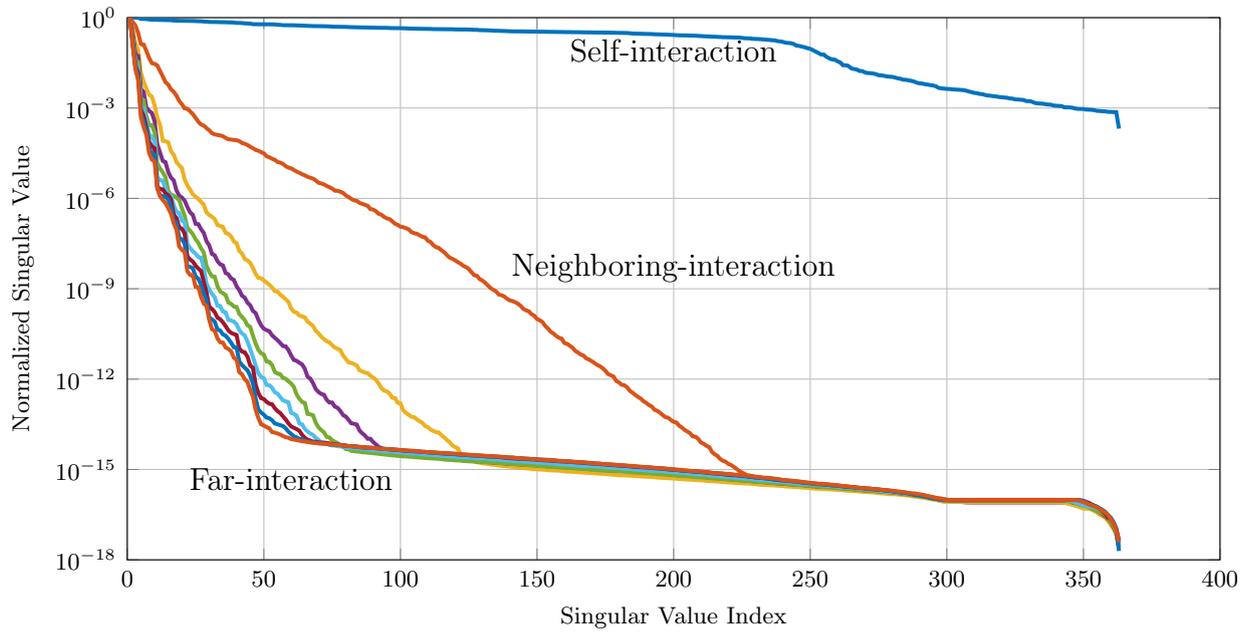
Based on the observation that fewer DoFs are sufficient to represent far-interacting subdomains, efficient and fast fill-in of the reduced matrix has been the subject of various research works. For instance, in [41], the Adaptive Cross Approximation (ACA) [26] method is used in a CBFM solver. The ACA is an algebraic technique which can be used to factorize a rank-deficient MoM sub-block,  $Z_{pq}$ , on-the-fly without prior knowledge of all the entries in the sub-block. A comprehensive description of the ACA algorithm is provided in [26]. Using the ACA,  $Z_{pq}$  can be approximated as

$$Z_{pq} \approx UV. \quad (3.8)$$

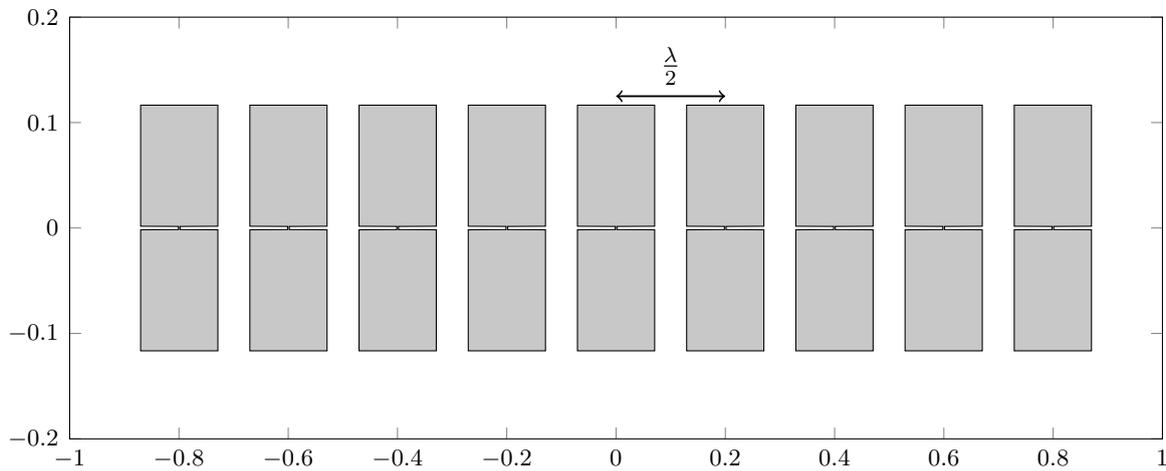
$U$  and  $V$  have sizes of  $N_i \times R$  and  $R \times N_i$ , respectively, where  $R$  is the approximated numerical rank (depending on a pre-defined threshold) and  $R \ll N_i$ . The block entry between subdomain  $p$  and  $q$  in the MBF reduced matrix can then be approximated using ACA such as

$$Z_{pq}^{\text{Red}} \approx (J_p^T U)(V J_q). \quad (3.9)$$

The computational cost to compute the reduced impedance matrix still scales as  $\mathcal{O}(M^2)$ , however the use of ACA drastically reduces the time to compute each sub-block since the computation in (3.9) scales as  $\mathcal{O}(2KN_i R + K^2 R)$ . The CBFM-ACA solver in [41] was



(a)



(b)

**Figure 3.1:** (a) Normalized singular values of MoM impedance matrix sub-blocks, generated by a linear array of 9 blade dipole antennas. Each sub-block represents the interaction between an antenna pair (self- or non-self-interaction). Self-interaction blocks are full-rank, while non-self-interaction blocks are rank deficient. (b) Blade dipole array configuration used to generate the MoM impedance matrix. The inter-element spacing is half wavelength.

employed to analyze periodic antenna arrays. By exploiting the Toeplitz structure of the MoM matrix,  $\mathcal{O}(M)$  interactions need to be computed using (3.9).

In [42], the multipole expansion was used to rapidly compute the interaction between MBFs. The computational time to construct a reduced impedance block scales with  $N_i$ . However, the approximation error was found to be too large for small inter-antenna distance [43]. For the application of irregular array analysis in [40], the interactions among MBFs were computed over a few “baseline” cases in a two-dimensional plane. After some physical transformations, namely subtraction of the far-field function, phase extraction and change of distance variable, a smooth function is fitted in a harmonic-polynomial representation. During the reduced matrix fill-in phase, the reaction term of any sets of MBFs can rapidly be computed using the harmonic-polynomial representation. Other approaches worth mentioning is MLFMA-CBFM in [44, 45] which also alleviate the need of computing the low-level coupling coefficients and AIM-SFX in [46] which improves the computational time of computing the matrix-vector product  $Z_{pq}J_q$  in the reaction term.

In Chapter 4, matrices with hierarchical structures will be introduced. And, the use of hierarchical matrices to accelerate computation of reaction terms in MBF solvers will be discussed.

### 3.4 Solving the Reduced Impedance Matrix

The MBF matrix equation is computed in (3.7), and can be written succinctly as

$$Z^{\text{Red}} I^{\text{MBF}} = V^{\text{Red}}. \quad (3.10)$$

$I^{\text{MBF}}$  is the unknown MBF coefficient vector and as previously mentioned, the size of the reduced impedance matrix,  $Z^{\text{Red}}$  is  $MK \times MK$ . For up to moderately large problems, the reduced system can be solved directly through LU decomposition, with memory requirement and runtime scaling as  $\mathcal{O}((KM)^2)$  and  $\mathcal{O}((KM)^3)$ , respectively.

For extremely large problems, where (3.9) cannot be solved directly despite the reduction in DoFs, a variety of methods have been proposed in the literature. In the MLFMA-CBFM [44, 45] approach, an iterative solver is used to solve the reduced impedance matrix. Only near field entries (block and near block diagonal entries) are stored fully. Far-field entries are not stored as full blocks and the fast multipole method is used for a fast matrix-vector product. Moreover, in [47], an MBF-GMRES iterative solver has been proposed for large antenna arrays. However, the drawback is that the iteration process needs to start anew for each excitation scheme. Preconditioners for the reduced impedance matrix have been developed to reduce the number of iterations, for instance a Sparse Approximation Inverse (SAI) preconditioner is proposed in [48] for an iterative-MBF solver. Sparse direct solutions to address this issue will be discussed in later sections.

### 3.5 Conclusion

In this chapter, the MBF approach has been introduced. It aims to reduce the DoFs through the introduction of physics-based functions defined over each antenna. The main bottlenecks of an MBF solver is the computational time to compute reaction terms of the reduced matrix; and solving the reduced matrix directly for the case of extremely large problems. Propositions based on advanced ACA-based methods, to address these drawbacks will be discussed in later chapters.

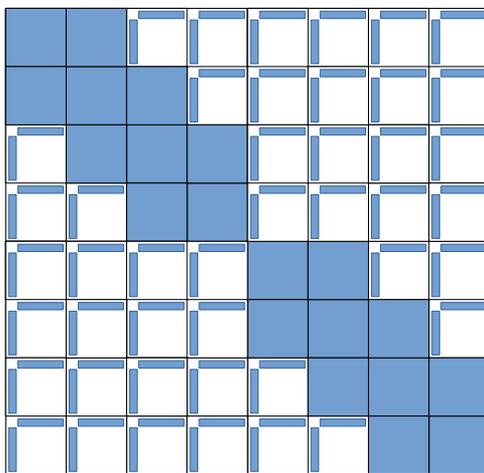


# Chapter 4

## Hierarchical Matrices

### 4.1 Introduction

A flat or non-hierarchical structure is usually employed to create the reduced impedance matrix in an MBF solver for array analysis, for instance in the CBFM-ACA solver in [41]. In this partitioning strategy, each subdomain is an antenna element, and thus the subdomains are of equal sizes. An example of the resulting partitioned matrix structure is shown in Figure 4.1. The blue blocks are full MoM entries for self- and near-interactions, whereas the far-interactions are represented in a compressed form as the factors resulting from a fast low-rank factorization technique such as the ACA. However, this flat structure requires the factorization of all non-neighboring interactions and can still be improved upon, with respect to computational cost.



**Figure 4.1:** Representation of a flat/non-hierarchical block matrix structure.

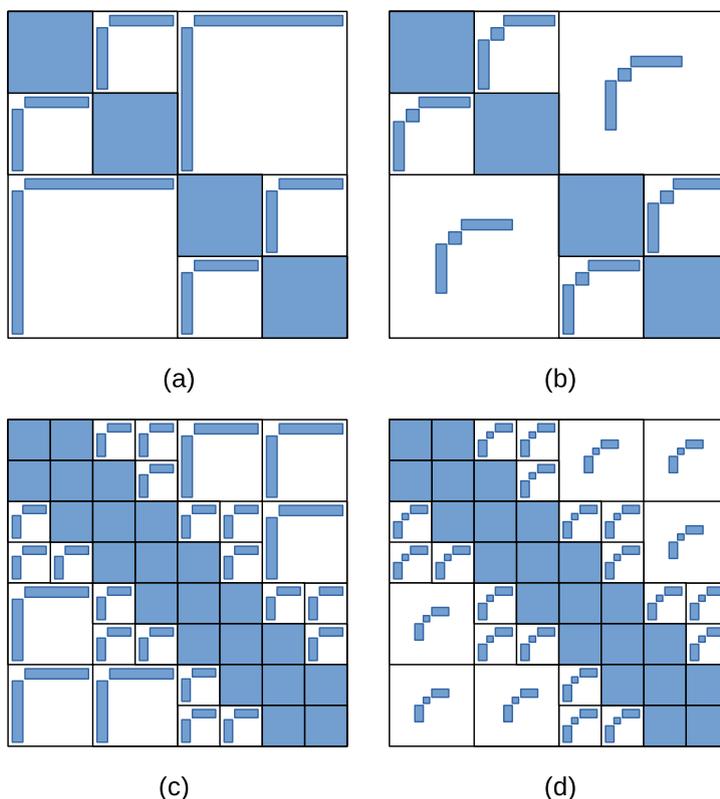
In this chapter, hierarchical matrices will be introduced as an alternative approach to create the reduced impedance matrix in MBF solvers. In hierarchical matrices, the entire geometry is partitioned in a multi-level fashion using a binary tree partitioning strategy. At the leaf level (finest level), a cluster (or subdomain) is an antenna element, and, at higher levels, a cluster contains multiple neighboring antennas. Interactions between admissible clusters at all levels are then factorized as  $Z_{pq} \approx U_p S_{pq} V_q$  or  $Z_{pq} \approx U_{pq} V_{pq}$ , depending on the technique used. The meaning and generation of these factors will be detailed during the remainder of the chapter. The motivation to use hierarchical structures is twofold, firstly to reduce the computational cost of creating the reduced

	Buffer	Refined off-diagonal	Nested
<b>HODLR</b>	No	No	No
<b>HSS</b>	No	No	Yes
<b><math>\mathcal{H}</math>-matrix</b>	Maybe	Yes	No
<b><math>\mathcal{H}^2</math>-matrix</b>	Maybe	Yes	Yes

**Table 4.1:** Description of the four hierarchical matrices.

impedance matrix. Instead of reducing the computational cost of computing the reaction terms by a constant factor using the flat partitioning structure, hierarchical matrices can achieve a log-linear scaling through the use of a binary tree partitioning. An MBF solver for antenna array analysis where computation of reaction term for the reduced matrix is accelerated through the use of factors from a hierarchical matrix technique is the subject of Chapter 5. Secondly, together with the full MoM sub-blocks for near interactions, the factors arising from the far interactions can be used in an approximate direct solver without the need of creating a fully populated matrix, which is the subject of Chapter 6. Thus, the reduced impedance matrix will not be stored entirely.

Four types of hierarchical matrices will be discussed in this section, namely, Hierarchically Off-Diagonal Low-Rank (HODLR) [49], Hierarchically Semi-Separable (HSS) [50],  $\mathcal{H}$ -matrix [51, 52], and  $\mathcal{H}^2$ -matrix [53, 52]. Figure 4.2 and Table 4.1 show the differences among the four types of hierarchical matrices.



**Figure 4.2:** Structure of various hierarchical matrices: (a) HODLR, (b) HSS, (c)  $\mathcal{H}$  matrix and (d)  $\mathcal{H}^2$  matrix.

### 4.1.1 HODLR and HSS

HODLR and HSS matrices have the same matrix partitioning strategy (shown in Figure 4.2 (a)-(b)); HSS matrices have a nested structure (shown using shorter rectangles to represent low-rank approximation in Figure 4.2(b)) while HODLR matrices do not. Nested structure means that the low-rank factors of a cluster at a parent level (e.g. level  $l$ ) can be constructed from that of its children clusters (level  $l + 1$ ), thus leading to a reduction in memory usage. The whole geometrical domain is first decomposed into two subdomains, this will lead to a partitioning of the impedance matrix into four sub-matrices. The non-self-interacting subdomains are considered to be rank-deficient and hence are compressed using low-rank approximations, while the diagonal blocks are further decomposed. This step is repeated for various levels until the diagonal blocks are small enough (e.g. by a pre-defined number of levels or finest block size). All off-diagonal blocks at every level are approximated with low-rank factors. In [49], a fast direct solver for HSS and HODLR matrices was proposed, where the impedance matrix,  $Z^{\text{MoM}}$ , is factorized as a product of block diagonal matrices such as

$$Z^{\text{MoM}} = \dot{Z}_L Z_L Z_{L-1} Z_{L-2} \cdots Z_1, \quad (4.1)$$

where  $L$  is the leaf level. Considering a single-level algorithm,

$$Z^{\text{MoM}} = \begin{bmatrix} Z_{11} & U_{12}V_{12} \\ U_{21}V_{21} & Z_{22} \end{bmatrix} = \begin{bmatrix} Z_{11} & 0 \\ 0 & Z_{22} \end{bmatrix} \begin{bmatrix} I & Z_{11}^{-1}U_{12}V_{12} \\ Z_{22}^{-1}U_{21}V_{21} & I \end{bmatrix}, \quad (4.2)$$

where  $I$  and  $0$  are identity and zero matrices, respectively.  $U_{pq}$  and  $V_{pq}$  ( $p, q \in \{1, 2\}, p \neq q$ ) are low-rank factors of non-self interactions. Following (4.2) and generalizing to  $L$  levels, the block diagonals of the matrix  $\dot{Z}_L$  are the self-interactions at leaf level, while the block diagonals of the matrices  $Z_l$  ( $l = 1, 2, \dots, L$ ) have the form of  $(I + \hat{U}\hat{V})$ . Considering a MoM matrix equation:  $Z^{\text{MoM}}I^{\text{MoM}} = V^{\text{MoM}}$ , the MoM current coefficient can be obtained using the factorized matrix in (4.1), such as

$$I^{\text{MoM}} = Z_1^{-1} Z_2^{-1} \cdots Z_L^{-1} \dot{Z}_L^{-1} V^{\text{MoM}}. \quad (4.3)$$

Because of the special structure of the block diagonals in matrices  $Z_l$  ( $l = 1, 2, \dots, L$ ) in (4.3), the solution can rapidly be obtained using the Sherman-Morrison-Woodbury (SMW) [54] identity.

#### 4.1.1.1 Application to MBF Solvers

In [55], the HODLR matrix structure was used when computing the reaction terms within a CBFM solver to analyze large scattering problems. ACA was used for low-rank approximation of the non-self interactions. Both the ACA factors at all levels and the full MoM self-interacting blocks at the leaf level were pre- and post-multiplied by the corresponding CBF definitions to create the reduced system in the HODLR format. The direct solver proposed in [49] was then used to compute the CBF coefficients.

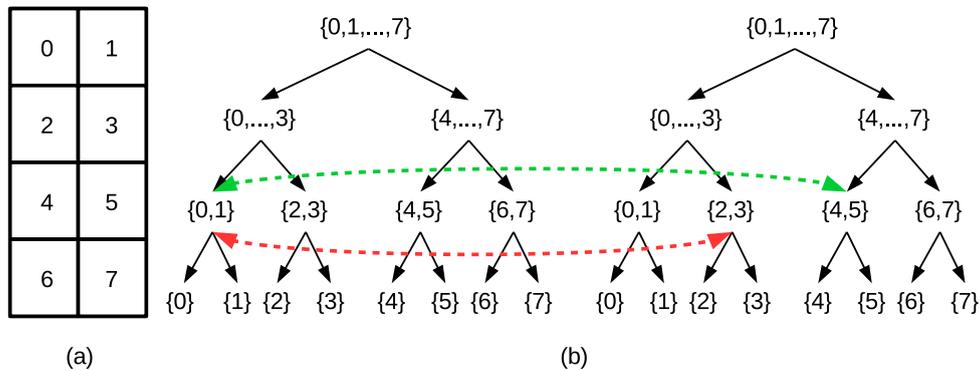
The computational cost and memory scaling of the fast HODLR solver in [49] is reported to scale as  $\mathcal{O}(N \log^2 N)$  and  $\mathcal{O}(N \log N)$  respectively. This is the case when the problem in question has a smooth kernel or is in the low frequency regime. For higher frequency problems, the ranks are expected to increase with the electrical size of the domain. In the application to electromagnetic scattering by large structures, the computational time and memory scaling tend to be poorer, especially as the electrical size

increases. To improve this situation, in [56] a multiscale ACA followed by SVD operation is proposed for the low-rank factorization of interactions between sibling clusters.

The fast HODLR/HSS matrix factorization is not an appropriate solution for the large antenna array problems, because of the assumption made that all non-diagonal blocks at every level are low-ranked. For large 2D (planar antenna arrays) and highly oscillatory kernel problems, the rank of the off-diagonal blocks might be too large (especially at levels with large subdomains), and hence harmful to the computational scaling of the solver. In the coming sections, other structures are discussed which are more suitable for the planar antenna array analysis problem.

### 4.1.2 $\mathcal{H}$ -Matrix

In  $\mathcal{H}$ -Matrix [51] partitioning, the non-self-interacting blocks can be refined as opposed to HODLR/HSS partitioning. An example of the  $\mathcal{H}$ -Matrix structure is shown in Figure 4.2(c). The original impedance matrix is partitioned using a block cluster tree as shown in Figure 4.3.



**Figure 4.3:** (a) Decomposition of a geometrical structure. (b) Block cluster tree which depicts the partitioning of source and observer clusters. Green dotted arrow represents interaction between admissible blocks (Low-ranked blocks); Red dotted arrow represents interaction between inadmissible blocks (Non-compressed blocks).

The blocks are classified as admissible (rank deficient and compressible) or inadmissible (non-compressible) blocks. The admissible blocks can be approximated using low-rank factorization such as  $Z_{pq} \approx U_{pq} V_{pq}$ . To determine which blocks are admissible, an admissibility criterion is used, which reads as follows:

$$\max(D_p, D_q) \leq \eta_{\text{low}} \text{dist}(p, q), \quad (4.4)$$

where  $D_p$  and  $D_q$  are the diameters of bounding boxes enclosing clusters  $p$  and  $q$ , respectively, and  $\text{dist}(p, q)$  is the minimum distance between the bounding boxes of cluster  $p$  and  $q$ .  $\eta_{\text{low}}$  is a constant parameter to be set. Starting from the roots of the block cluster trees (whole geometrical domain), interactions at every level are verified using the admissibility criterion. If met, the block is qualified as being admissible and hence no further partitioning is required. If the criterion is not met, the clusters are further partitioned into their children clusters. This procedure is repeated until the finest level (leaf) is reached. Narrowing down to our application, the inverse of a block [51, 52],  $Z$ , can be obtained using its children as follows:

$$Z^{-1} = \begin{bmatrix} Z_{11}^{-1} + Z_{11}^{-1} Z_{12} C^{-1} Z_{21} Z_{11}^{-1} & -Z_{11}^{-1} Z_{12} C^{-1} \\ -C^{-1} Z_{21} Z_{11}^{-1} & C^{-1} \end{bmatrix}, \quad (4.5)$$

where  $C$  is the Schur complement, denoted as  $C = Z_{22} - Z_{21}Z_{11}^{-1}Z_{12}$ . Using (4.5), inversion of the entire matrix can be obtained in a recursive manner. This concept is exploited for electromagnetic problems in a technique called Multiscale Compressed Block Decomposition (MS-CBD) [57]. ACA followed by SVD (for optimal compression) is applied to admissible blocks in MS-CBD. The memory storage and computational time for the method scales as  $\mathcal{O}(N^{1.5})$  and  $\mathcal{O}(N^2)$  respectively.

#### 4.1.2.1 Application to MBF Solvers

In [58], the CBFM is combined with the MS-CBD to reduce the DoFs associated with the problem. In other words, the reduced impedance matrix is constructed in a multiscale manner as in MS-CBD, and is solved using the recursive factorization procedure. A Fast-ACA algorithm [59] is used in [58] instead of the ACA-SVD in MS-CBD to improve the performance. CBFs are generated for leaf-level clusters. At level  $l$ , the CBF group of cluster  $i$ ,  $J_i^l$ , is a block diagonal matrix and is obtained by aggregating leaf-level CBFs, such as

$$J_i^l = \begin{bmatrix} J_{i,1} & & & \\ & J_{i,2} & & \\ & & \ddots & \\ & & & J_{i,G} \end{bmatrix}. \quad (4.6)$$

$J_{i,g}$  ( $g \in \{1, 2, \dots, G\}$ ) in (4.6) is the CBF definition for the  $g$ -th leaf-level child (or grandchild) of cluster  $i$ . This method has been applied to electromagnetic scattering problems with multiple excitations, and good performance is achieved.

#### 4.1.3 $\mathcal{H}^2$ -Matrix

$\mathcal{H}^2$ -Matrices have a similar matrix partitioning strategy as that of  $\mathcal{H}$ -Matrices, as shown in Figure 4.2(d). However, the first difference is that rank-deficient admissible blocks in  $\mathcal{H}^2$ -Matrices are factorized as

$$Z_{pq} \approx U_p S_{pq} V_q, \quad (4.7)$$

where  $U_p$  and  $V_q$  are the cluster bases and  $S_{pq}$  is the coupling matrix. The sizes of  $U_p$  and  $V_q$  are  $N_i \times R_p$  and  $R_q \times N_i$ , respectively, and the size of  $S_{pq}$  is  $R_p \times R_q$ , where  $R_p$  and  $R_q$  are the numerical ranks of clusters  $p$  and  $q$ . Unlike the factorizations in  $\mathcal{H}$ -Matrix or ACA,  $Z_{pq} \approx U_{pq} V_{pq}$ , the cluster bases in  $\mathcal{H}^2$ -Matrices,  $U_p$  and  $V_q$ , are strictly associated with the observer  $p$  and source  $q$  domains, respectively. The coupling matrix,  $S_{pq}$ , relates the clusters  $p$  and  $q$ .

Secondly,  $\mathcal{H}^2$ -Matrices have a nested property, meaning that at non-leaf levels, cluster bases can be created using their children cluster bases and transfer matrices. Transfer matrices are the interpolation and antinterpolation terms which link parent-children clusters across levels, and are smaller in size compared to cluster bases. The size of a transfer matrix is  $R_p \times R_c$  (or,  $R_c \times R_p$  depending if one is marching up or down the cluster tree) where  $R_p$  and  $R_c$  are the numerical ranks of the parent and children clusters respectively. For instance in (4.8), a cluster basis at level  $L-1$ ,  $V^{L-1}$ , is constructed using its leaf-level children clusters,  $v_1^L$  and  $v_2^L$  and source transfer matrices,  $C_1$  and  $C_2$ .

$$V^{L-1} = \begin{bmatrix} v_1^L \\ v_2^L \end{bmatrix} \begin{bmatrix} C_1 \\ C_2 \end{bmatrix} \quad (4.8)$$

The nested property in  $\mathcal{H}^2$ -Matrices aims at improving the memory requirement by permitting an efficient reuse of information across the cluster tree. Cluster bases are only stored at the leaf level, and for non-leaf levels, only smaller transfer matrices and coupling matrices are stored.

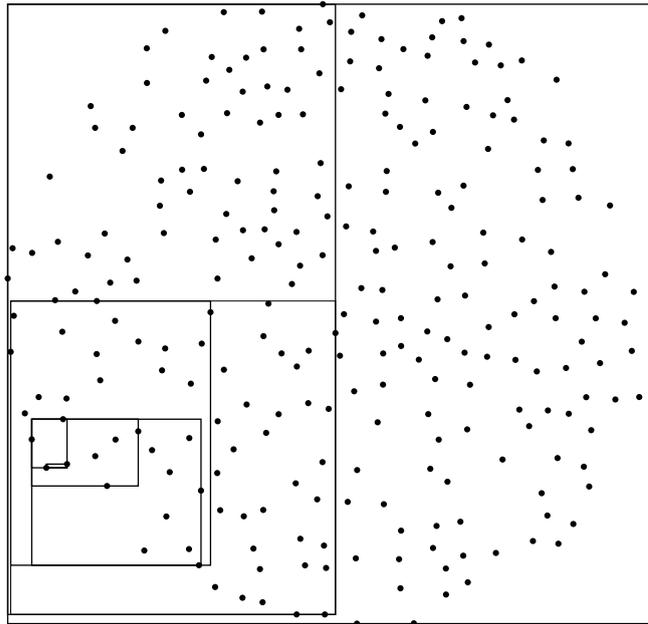
For problems with a smooth kernel in the EFIE formulation (e.g. the electro-quasistatic case), a constant rank across the levels can be expected. In these cases, since the scaling of  $\mathcal{H}^2$ -Matrices are independent of the depth of the cluster tree, the memory cost of  $\mathcal{H}^2$ -Matrices scale as  $\mathcal{O}(N)$ , whereas that of  $\mathcal{H}$ -Matrices scale as  $\mathcal{O}(N \log N)$  [52]. However, this statement is not true for an oscillatory kernel since the numerical rank increases with the electrical size of the structure. To deal with this, directional  $\mathcal{H}^2$ -Matrices [60] will be discussed in later sections. For the rest of the thesis,  $\mathcal{H}^2$ -Matrices and directional  $\mathcal{H}^2$ -Matrices will be employed for the fast computation of reaction terms in MBF solvers for antenna array analysis.

## 4.2 Nested Cross Approximation

The Nested Cross Approximation [27] (NCA) is an extension to the ACA algorithm and has an  $\mathcal{H}^2$ -Matrix structure. As already mentioned in Section 4.1.3, a compressible block is factorized as  $Z_{pq} \approx U_p S_{pq} V_q$ .

### 4.2.1 Partitioning and Admissibility Condition

The antenna array is partitioned through the use of a binary tree and clusters are created at every level. The number of antennas in a cluster is halved at every level. Level 0 is the entire array and the leaf level clusters (finest level) contain only one antenna element. An example of the binary partitioning of an irregular antenna array is shown in Figure 4.4.



**Figure 4.4:** Binary partitioning of an irregular antenna array containing 256 elements. Each dot represents the position of an antenna element. The radius of the circular array is  $15\lambda$ .

At each level, an admissibility criterion test for the NCA, as described in (4.9), is performed to verify if the interactions between observer-source cluster pairs are compressible. The admissibility criterion is similar to the one described for  $\mathcal{H}$ -matrices, and is repeated here for convenience.

$$\max(D_p, D_q) \leq \eta_{\text{low}} \text{dist}(p, q) \quad (4.9)$$

If an observer-source cluster pair at a given level satisfies the criterion, the interaction is compressed, and the children interactions are not considered. However, if a pair does not satisfy the criterion, the observer and source clusters are partitioned, and the children interactions are tested. This process continues till the leaf level and a list of admissible pairs at every level is stored. Non-admissible pairs at the leaf level are self and neighboring interaction which are not compressed, rather the full MoM sub-blocks of these interactions are stored.

### 4.2.2 Generation of NCA Factors

At the leaf level, the observer and source cluster bases are obtained through an inverse source process and is defined as

$$U = Z_{t\sigma_t} (Z_{\tau_t\sigma_t})^{-1} \quad (4.10)$$

$$V = (Z_{\tau_s\sigma_s})^{-1} Z_{\tau_s s}. \quad (4.11)$$

The symbols  $t$ ,  $s$ ,  $\tau_t$ ,  $\sigma_t$ ,  $\tau_s$ , and  $\sigma_s$  denote sets of local and far-field RWG indices for the observer and source clusters. The descriptions of these indices are given in Table 4.2 and shown in Figure 4.5. The term ‘local domain’ in Table 4.2 and Figure 4.5 simply refers to the cluster (observer or source) over which the  $U$  and  $V$  cluster bases are defined. And, the ‘far-field domain’ is the region within which clusters satisfy (4.9) with respect to the local domain.

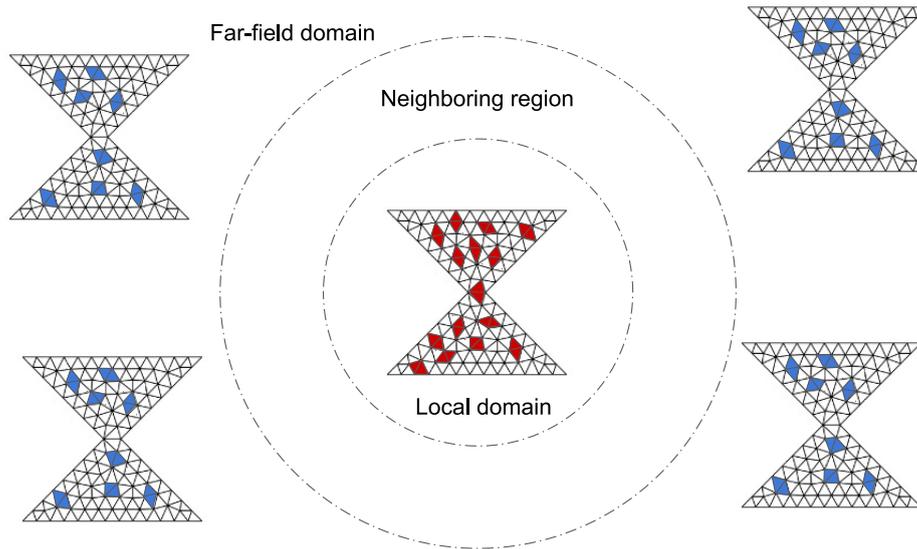
Index	Description
$t$	All the testing functions on the observer’s local domain
$s$	All the basis functions on the source’s local domain
$\tau_t$	Subset of testing functions on the observer’s local domain
$\sigma_t$	Subset of basis functions on the antennas within the observer’s far-field domain
$\tau_s$	Subset of testing functions on the antennas within the source’s far-field domain
$\sigma_s$	Subset of basis functions on the source’s local domain

**Table 4.2:** Description of indices during the construction of cluster bases.

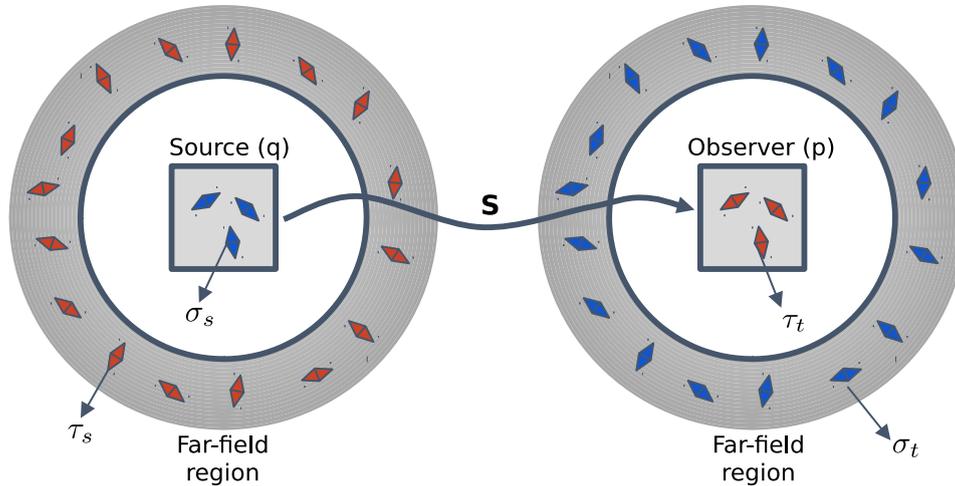
The details on how to obtain the sets of indices described in Table 4.2 will be explained in Section 4.2.2.1. The coupling matrix,  $S$ , is then denoted as the interaction between the  $\sigma_s$  basis functions on the source domain and the  $\tau_t$  testing functions on the observer domain as

$$S = Z_{\tau_t\sigma_s}. \quad (4.12)$$

Figure 4.6 is a depiction of the local and far-field domains of both the source and observer clusters. The  $S$  interaction between the  $\sigma_s$  basis functions on the source cluster and  $\tau_t$  testing functions on the observer cluster is also shown.



**Figure 4.5:** Depiction of RWG indices described in Table 4.2 at leaf level. Red RWG functions: Subset of RWGs within the local domain, the set of indices is denoted as  $\tau_t$  (observer) or  $\sigma_s$  (source). Blue RWG functions: Subset of RWGs within the far-field domain, the set of indices is denoted as  $\sigma_t$  (observer) or  $\tau_s$  (source). The set of indices on all the RWGs on the local domain is denoted as  $t$  (observer) or  $s$  (source). The antennas in the neighboring region are omitted for simplicity.



**Figure 4.6:** Local and far-field domains for the source and observer clusters.

For non-leaf levels, transfer matrices (defined  $B$  and  $C$  for observer and source clusters, respectively) are used to interpolate or antepolate from their children clusters. The  $C$  transfer matrix is obtained as

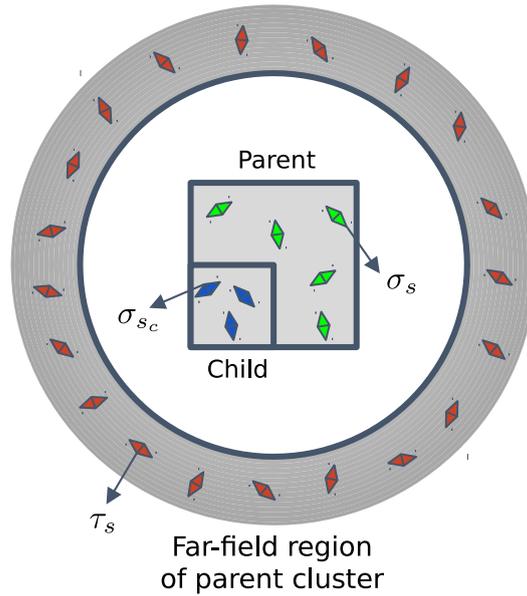
$$C = (Z_{\tau_s \sigma_s})^{-1} Z_{\tau_s \sigma_{s_c}}. \quad (4.13)$$

The symbols  $\sigma_s$  and  $\tau_s$  are similar to the descriptions in Table 4.2 and belongs to the parent cluster. The symbol  $\sigma_{s_c}$  is the  $\sigma_s$  indices of the child cluster, where the subscript ‘ $c$ ’ refers to child. The transfer matrix described in (4.13) is obtained through an inverse source process between the child’s cluster and the far-field of the parent. For instance, current

coefficients defined on the  $\sigma_{s_c}$  basis functions (RWGs indices on the child's local domain) is radiated onto the  $\tau_s$  testing functions in the parent's far-field region. The resulting field on the testing functions will in turn induce a current on the  $\sigma_s$  basis functions on the parent cluster's local domain. The transfer matrix on the observer side is also obtained through an inverse source process, and is formally described as

$$B = Z_{\tau_t \sigma_t} (Z_{\tau_t \sigma_t})^{-1}. \quad (4.14)$$

Similarly, the  $\tau_t$  and  $\sigma_t$  are local and far-field indices of the parent cluster, and  $\tau_{t_c}$  is a set of local indices on the child cluster. Figure 4.7 illustrates a child and parent clusters, and the far-field of the parent cluster to form the transfer matrix during the inverse source process. After obtaining the transfer matrices, the interaction between two antennas ( $p_c$



**Figure 4.7:** Local domain of the child and parent clusters, and the far-field domain of the parent cluster.

and  $q_c$ ) using a 2-level NCA can be written as:

$$Z_{p_c q_c} \approx U_{p_c} B_{p_c, p} S_{pq} C_{q, q_c} V_{q_c}, \quad (4.15)$$

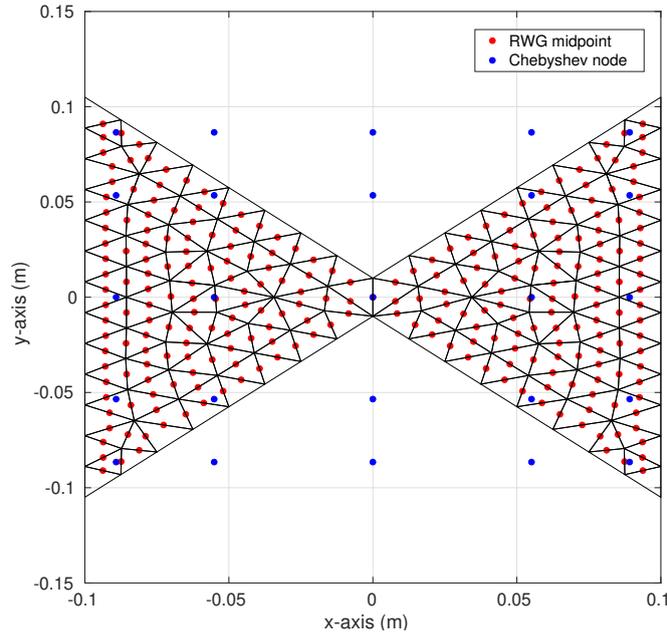
where  $p_c$  and  $q_c$  are the children clusters of  $p$  and  $q$ , respectively.

#### 4.2.2.1 Selecting Basis and Testing Functions for NCA Factors

As a consequence of the EFIE MoM matrix symmetry, the cluster basis,  $V$ , and transfer matrix,  $C$ , for a given cluster are the transposes of  $U$  and  $B$ , respectively. Hence, only the  $U$  and  $B$  factors are computed. As described in (4.10) and (4.13), the local domain indices,  $\tau_t$ , and far-field domain indices,  $\sigma_t$ , must be available to compute  $U$  and  $B$ . To obtain these indices, the matrix  $Z_{\tilde{\tau}_t \tilde{\sigma}_t}$  is first constructed which represents an incomplete interaction between the antenna and its far-field, where  $\tilde{\tau}_t$  and  $\tilde{\sigma}_t$  are sampled indices of the local and far-field RWGs such that  $t \supseteq \tilde{\tau}_t \supset \tau_t$  and  $N_f \supset \tilde{\sigma}_t \supset \sigma_t$ , and  $N_f$  is a set of indices of all RWGs belonging to antennas in the far-field region. The ACA algorithm with

thresholding is then performed on the matrix  $Z_{\tilde{\tau}_t \tilde{\sigma}_t}$ , and the selected rows and columns indices through the ACA process are denoted as  $\tau_t$  and  $\sigma_t$  respectively. Ideally, ACA should have been performed on the matrix  $Z_{tN_f}$ , however this operation is impractical from a computational point of view since  $N_f$  is too large and will also lead the runtime to be proportional to  $M$ .

The sample  $\tilde{\sigma}_t$  is obtained by placing a Chebyshev grid on top of bounding boxes of all antennas belonging to clusters in the far-field region. Figure 4.8 shows an example of Chebyshev nodes on top of a planar bow-tie antenna. RWGs with midpoints closest to a pre-defined number of Chebyshev nodes are then selected for every far-field antenna and the union of all sampled RWGs is denoted as  $\tilde{\sigma}_t$ . The number of Chebyshev nodes per antenna,  $C_i = \text{ceil}\left(c \frac{N_i}{M_f}\right)$ , where  $M_f$  is the number of far-field antennas and  $c$  is a constant set to 1 in this implementation. However,  $C_i \rightarrow 1$  as  $M_f \rightarrow \infty$ , hence a minimum constraint of  $C_i$  is included so as  $C_i \geq 2^{\nabla_d}$ , where  $\nabla_d$  is the number of dimensions of the bounding box (e.g. 2D or 3D). Moreover, the number of Chebyshev nodes on each axis is set to be proportional to the lengths of the bounding box axes. In the special case where the cluster's far-field is empty,  $\tilde{\sigma}_t$  is obtained by sampling the RWGs in the parent's far-field region, through the same procedure. At leaf level,  $\tilde{\tau}_t = t$  since  $c = 1$ . Otherwise if  $t$  is too large,  $\tilde{\tau}_t$  can be obtained by placing a Chebyshev grid on the local domain, and the  $\text{ceil}(cN_i)$  closest RWGs to the nodes are selected, where  $c < 1$ . To avoid uncontrolled errors,  $c < 1$  can only be set if  $N_i$  is much larger than the anticipated rank, and also diversity in the orientations of the selected RWGs must be ensured. At non-leaf levels,  $\tilde{\tau}_t$  is obtained as a union of the children clusters'  $\tau_t$  indices. Thus,  $\tau_t$  at the child level must already be available in order to obtain  $\tilde{\tau}_t$  for a non-leaf cluster.



**Figure 4.8:** Meshed bow-tie antenna. Red dots represent the midpoints of RWG functions; Blue dots represent the Chebyshev nodes.

### 4.3 Directional Cross Approximation

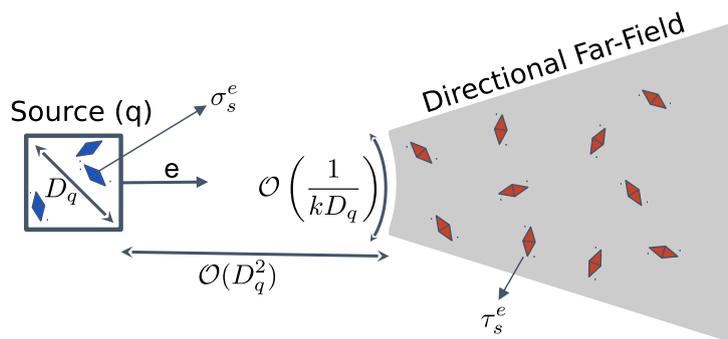
The NCA scales as  $\mathcal{O}(N)$  for problems with smooth kernels such as in electrostatic or magnetostatic cases, with the assumption that the rank is constant across levels. However, for the analysis of antenna arrays, the EFIE contains an oscillatory Green's function as

$$G(\mathbf{r}, \mathbf{r}') = \frac{e^{-jk|\mathbf{r}-\mathbf{r}'|}}{4\pi|\mathbf{r}-\mathbf{r}'|}. \quad (4.16)$$

In this case and when operating in the high frequency regime, the DoFs required to represent an interaction between two clusters, for a pre-defined accuracy, can no longer be assumed to be constant irrespective of the size of the clusters. In fact, the numerical rank increases with the electrical size of the domains. A high frequency regime is defined as in (4.17), where  $k$  is the wavenumber.

$$k \min(D_q, D_p) > 1. \quad (4.17)$$

For high-frequency problems, the rank can be kept constant by using a directional  $\mathcal{H}^2$ -Matrix structure. In this formulation, the far-field of a cluster when computing the factors is not omnidirectional, rather it is limited by an angular field of view as shown in Figure 4.9. Some examples of the directional  $\mathcal{H}^2$ -Matrix structure include the directional  $\mathcal{H}^2$ -Matrix with interpolation [60], Directional Cross Approximation (DCA) [28] and the Wideband Nested Equivalent Source Approximation (WNESEA) [61]. The WNESEA is inspired by the DCA algorithm, and has been applied to the MoM for high-frequency problems in [61]. However, the major difference is that the low-rank factors are created through the use of equivalent sources in WNESEA, while the ACA is employed on selected basis and testing functions in the DCA. In this thesis, the focus will be on DCA which is an extension of the NCA discussed in Section 4.2.



**Figure 4.9:** Inverse source process to obtain directional cluster bases.

In DCA, a low-rank sub-block,  $Z_{pq}$ , can be factorized as

$$Z_{pq} \approx U_p^e S_{pq} V_q^{-e}, \quad (4.18)$$

where  $U_p^e$  and  $V_q^{-e}$  are the directional cluster bases and the superscript ‘ $e$ ’ and ‘ $-e$ ’ refers to the direction of the cluster bases. Each cluster will have multiple  $U^e$  and  $V^e$  factors, one for each direction, as opposed the NCA, where a cluster has only one  $U$  and one  $V$  cluster basis. A similar partitioning of the array through the use of a binary tree is performed. However, the admissibility criterion is changed and contains both a distance and angular

condition, to ensure directional asymptotic smoothness for the oscillatory Green's function [28]. The admissibility criterion for the DCA algorithm is expressed as

$$\eta_{\text{high}} \text{dist}(p, q) \geq k \max(D_p^2, D_q^2) \quad (4.19)$$

$$\beta \propto \left( \frac{1}{kD_q} \right). \quad (4.20)$$

Equation (4.19) is the distance criterion, where the square of the diameter of the cluster ( $D_p^2$  or  $D_q^2$ ) is used instead of the diameter itself. Moreover, the distance criterion takes into account the frequency by including the wavenumber,  $k$ , in the equation. The term  $\eta_{\text{high}}$  is a constant to be set, which can be a different value from that of  $\eta_{\text{low}}$  in the NCA. Equation (4.20) is the angular criterion, where the angle of the sector is inversely proportional to the wavenumber,  $k$ , and the diameter of the cluster,  $D_q$ . The far-field region is segmented into  $\frac{2\pi}{\beta}$  angular regions, and each cluster satisfying the distance criterion will fall in one of the angular regions based on the position of the centroid of its bounding box. Clusters at the leaf level have the widest angle since the clusters are smallest in size (only one antenna), thus the leaf level has the least number of directional far-field regions. Moving up the binary tree, at every level, the angle is halved and the number of directional far-field sectors is doubled. Moreover, the directional far-field of a parent cluster is a subset of the children's directional far-field. In the implementation, an angular far-field of  $\frac{\pi}{2}$  rad has been found suitable at the leaf level (as discussed in [1]), meaning the far-field is segmented into four regions.

The directional cluster bases are obtained through a similar inverse source process as in the NCA, however the far-field is limited as shown in Figure 4.9. The  $U^e$  and  $V^e$  directional cluster bases are defined as follows:

$$U^e = Z_{t\sigma_t^e} (Z_{\tau_t^e \sigma_t^e})^{-1} \quad (4.21)$$

$$V^e = (Z_{\tau_s^e \sigma_s^e})^{-1} Z_{\tau_s^e s}. \quad (4.22)$$

The symbols  $\tau_t^e$ ,  $\sigma_t^e$ ,  $\tau_s^{-e}$ , and  $\sigma_s^{-e}$  from (4.21) and (4.22) are defined in Table 4.3.

Index	Description
$\tau_t^e$	Subset of testing functions on the observer's local domain in regards to the directional far-field, $e$
$\sigma_t^e$	Subset of basis functions on the antennas within the observer's directional far-field domain, $e$
$\tau_s^{-e}$	Subset of testing functions on the antennas within the source's directional far-field domain, $-e$
$\sigma_s^{-e}$	Subset of basis functions on the source's local domain in regards to the directional far-field, $-e$

**Table 4.3:** Description of indices during the construction of directional cluster bases.

The coupling matrix,  $S$ , in the DCA is the interaction between  $\sigma_s^{-e}$  basis functions on the source cluster and the  $\tau_t^e$  testing function on the observer cluster, and is written as:

$$S = Z_{\tau_t^e \sigma_s^{-e}}. \quad (4.23)$$

And finally, the transfer matrices are expressed as in (4.24) and (4.25), respectively. The suffix ‘ $e_c$ ’ in (4.24) and (4.25) refers to the direction of the child cluster.

$$B^e = Z_{\tau_t^{e_c} \sigma_t^e} (Z_{\tau_t^e \sigma_t^e})^{-1} \quad (4.24)$$

$$C^e = (Z_{\tau_s^e \sigma_s^e})^{-1} Z_{\tau_s^e \sigma_s^{e_c}} \quad (4.25)$$

## 4.4 Conclusion

In this chapter, four variants of hierarchical matrices from the literature are explored for the fast computation of reaction terms in MBF solvers for array analysis. Among the methods, HODLR and HSS are found not be efficient for the application of array analysis since all non-self interactions are considered to be low-ranked and compressible.  $\mathcal{H}$ - and  $\mathcal{H}^2$ -Matrices are more suitable for electromagnetic analysis because of the partitioning strategy and use of admissibility conditions.  $\mathcal{H}^2$ -Matrices have a higher efficiency due to their nested property. To ensure constant ranks for the oscillatory kernel of the EFIE, irrespective of the cluster size across levels in the  $\mathcal{H}^2$ -Matrix, the far-field is partitioned into directional sectors when computing the factors. As such, the DCA technique, which has a directional  $\mathcal{H}^2$ -Matrix structure, has been found to be appropriate for array analysis.



## Chapter 5

# Fast Computation of MBF Reaction Terms with DCA Factors

### 5.1 Introduction

MBFs were introduced in Chapter 3, and one of the bottlenecks in the MBF solver is the expensive computational cost associated with calculation of reaction terms to create the reduced impedance matrix. In Section 3.3.1, various ways proposed in literature to accelerate the reduced matrix setup were mentioned. In particular, the single-level version of the original ACA algorithm was used to rapidly compute the reaction terms. However, the computational cost still scales as  $\mathcal{O}(M^2)$ , hence this approach might not be suitable for the analysis for large irregular arrays. In this chapter, the DCA, which is the most recent extension of the ACA specifically for oscillatory kernels, in multilevel format and with nested factorizations, as mentioned in Chapter 4, is employed with the aim of yielding better scaling. It is the first application of the algorithm to MoM matrices and especially to accelerate MBF reaction terms, to the author's knowledge.

### 5.2 Memory requirement

In the DCA algorithm, the following factors must be stored for non-near interactions:

- $\mathcal{O}(M)$  directional cluster bases at leaf level. Only  $U^e$  cluster bases are computed and stored since  $V^e = (U^e)^t$ .
- $\mathcal{O}(M^{l+1})$  directional transfer matrices at every non-leaf level, where  $M^l$  is the number of clusters on level  $l$ . Only  $B^e$  transfer matrices are computed since  $B^e = (C^e)^t$ .
- $\mathcal{O}(M_f^{l2})$  coupling matrices at every level, where  $M_f^l$  is the number of far-field (or admissible) clusters per level.

The above, together with the leaf-level near and self interactions which are stored directly, make up the total compressed MoM matrix memory requirement for the solver. The leaf level groups are set as the individual antennas, to facilitate the use of MBFs with elemental supports, as described next.

### 5.3 Using DCA for Fast Computation of MBF Reaction Terms

As discussed in Chapter 3, a block representing the interaction between antenna  $p$  and  $q$  in the MBF reduced impedance matrix can be written as

$$Z_{pq}^{\text{Red}} = \langle J_p^T, Z_{pq} J_q \rangle. \quad (5.1)$$

With the sub-block  $Z_{pq}$  being represented in single-level DCA compressed format, (5.1) can be written as

$$Z_{pq}^{\text{Red}} \approx \langle (J_p^T U_p^e), S_{pq} (V_q^{-e} J_q) \rangle. \quad (5.2)$$

In multilevel DCA, (5.2) is exactly the format encountered at leaf level. The product of the MBF definition and directional cluster basis,  $J^T U^e$ , is computed on the fly while generating the cluster bases, hence there is no need to store  $U^e$ . The term  $J^T U^e$  will be referred to as ‘reduced cluster basis’ in the rest of the text. The reduced cluster bases together with the rest of the original DCA factors at non-leaf levels gives a full representation of the reduced impedance matrix in compressed format, for non-neighboring interactions. Self and near interactions in the reduced matrix are computed using (5.1).

At level  $L - 1$  (where  $L$  is the leaf level), the reduced interaction between the level  $L - 1$  clusters,  $p$  and  $q$ , is expressed as

$$Z_{pq}^{\text{Red}} \approx \left\langle \left[ \begin{array}{c} \left( J_{p_{c1}}^T \ U_{p_{c1}}^{e_{c1}} \right) B_{p_{c1},p}^e \\ \left( J_{p_{c2}}^T \ U_{p_{c2}}^{e_{c2}} \right) B_{p_{c2},p}^e \end{array} \right], S_{pq} \left[ \begin{array}{c} C_{q,q_{c1}}^{-e} \left( V_{q_{c1}}^{-e_{c1}} J_{q_{c1}} \right) \\ C_{q,q_{c2}}^{-e} \left( V_{q_{c2}}^{-e_{c2}} J_{q_{c2}} \right) \end{array} \right] \right\rangle. \quad (5.3)$$

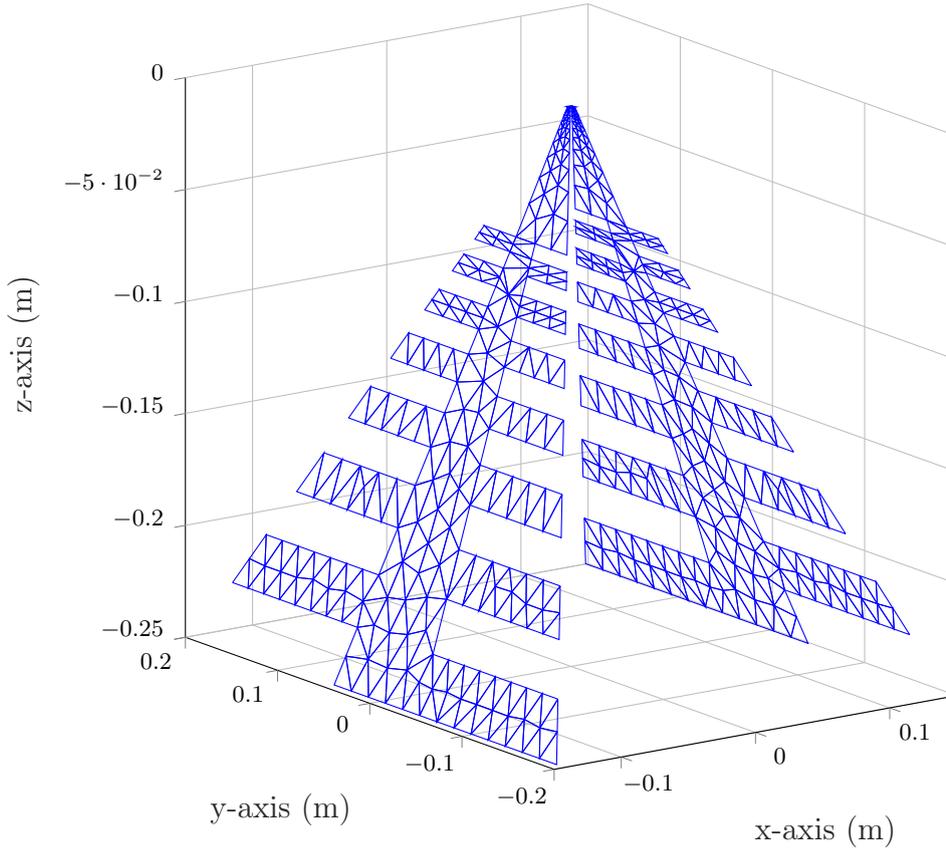
Since  $p$  and  $q$  in (5.3) are level  $L - 1$  clusters, they each support  $2N_i$  DoFs. The children of cluster  $p$  are referred as  $p_{c1}$  and  $p_{c2}$  in (5.1), and  $e_c$  is the direction of the child cluster.  $B_{p_{c1},p}^e$  is the transfer matrix which represents the interaction between the parent cluster  $p$  to its child cluster  $p_{c1}$ . These definitions extends for cluster  $q$  and transfer matrix  $C$ . In (5.3), the product of reduced cluster bases and their respective transfer matrices for all children clusters at leaf level are computed and concatenated. These concatenated matrices for the observer and source clusters are then used together with the coupling matrix,  $S_{pq}$ , to compute the  $Z_{pq}^{\text{Red}}$ .

### 5.4 Numerical Experiment Setup

A bent LPDA antenna shown in Fig. 5.1, designed using the Antenna Magus software [62], is used for numerical experiments. The antenna operates over the band of 500 MHz to 1350 MHz and meshing of the antenna in Fig. 5.1 is performed at 750 MHz with average mesh size of 0.05 wavelengths, resulting in  $N_i = 848$ . The antenna is used as the element in irregular antenna arrays with minimum inter-element spacing of 0.9m, consisting of 32, 64, 128, 256, and 512 elements. The arrays are used in the numerical experimentations. An example of the antenna positions in the 256-elements irregular array is shown in Fig. 5.2.

For efficient performance of the algorithm, parameters in the DCA and MBF need to be set. The following DCA parameters are set:

- The distance parameter,  $\eta_{\text{high}}$ , in (4.19) is set as  $\eta_{\text{high}} = 5.5$ . Setting the value of  $\eta_{\text{high}}$  is a trade-off between the memory requirement to store near MoM interactions, and the numerical ranks of the DCA factors. A large value of  $\eta_{\text{high}}$  implies that the



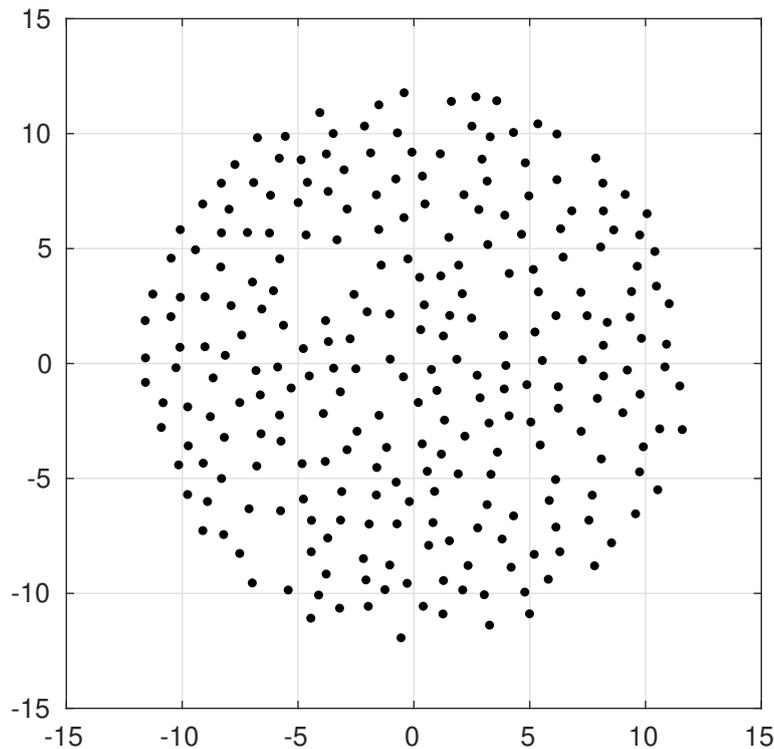
**Figure 5.1:** Mesh of bent LPDA antenna at the frequency of 750 MHz, resulting in 848 Dofs.

directional far-field is nearer, and thus less near interaction blocks need to be stored, but far field ranks are higher. In general, the aim in the multilevel algorithm is to have  $\eta_{\text{high}}$  as high as possible, since the benefit of more nested levels outweigh the drawback of higher far-field ranks.

- The angle of the directional far-field sectors,  $\beta$ , in (4.20) is set as  $\beta = \frac{\pi}{2}$  at leaf level, leading to four directional far-field regions for every cluster at leaf level. For non-leaf levels, the number of directional far-field regions is doubled at every level.
- An ACA threshold of  $10^{-3}$  is used during the ACA process on the matrix  $Z_{\tau_t^e \sigma_t^e}$  to obtain the sets of indices  $\tau_t^e$  and  $\sigma_t^e$ . This ACA threshold controls the numerical rank of the factor and the accuracy of the DCA compression.

The following parameters are set in the MBF algorithm:

- Up to tertiary MBFs are included, to ensure good accuracy.
- A radius of influence of 2m is used, within which the influence of secondary and tertiary MBFs are included. This radius of influence is specific to the array configuration. The radius of influence for the MBFs is larger than the near-field region of the DCA (determined from (4.19)).
- An SVD threshold value of  $10^{-4}$  is used during the construction of the final set of MBFs.



**Figure 5.2:** Antenna array configuration containing 256 elements with minimum inter-element distance of 0.9m.

Stringent MBF parameters have been set to ensure the MBF algorithm yields accurate result, and error norm can be attributed to the DCA.

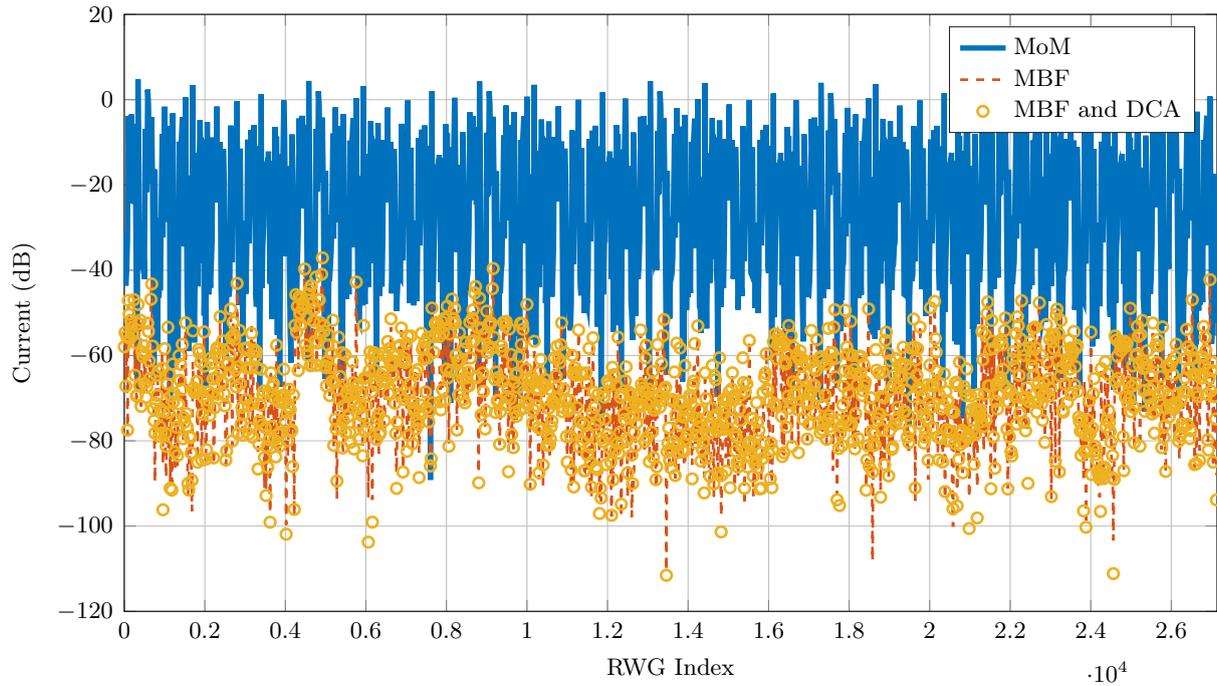
## 5.5 Numerical results

The current coefficients of an antenna array of 32 elements, solved by the MoM is shown in Fig. 5.3. The current coefficient errors relative to the MoM when using an MBF only solver and the MBF-DCA solver are also plotted in Fig. 5.3, and a good accuracy can be observed. The traces in Fig. 5.3 are plotted for every 20-th point to avoid the graph being overcrowded. The relative error in the global current solution is defined as

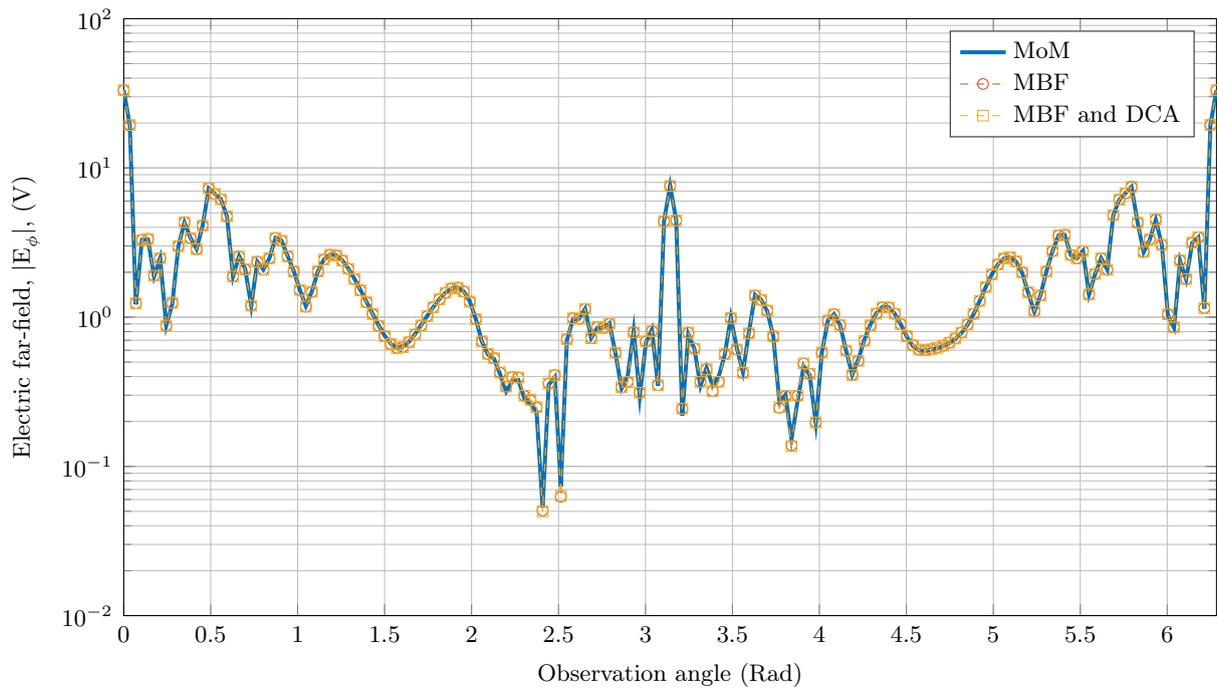
$$\text{Current Error Norm} = \frac{\|I_{\text{MoM}} - I_x\|_2}{\|I_{\text{MoM}}\|_2}, \quad (5.4)$$

where  $I_x$  refers to current coefficients being compared to the MoM solution. The current error norms for the MBF and MBF-DCA solvers are  $4.5 \times 10^{-3}$  and  $4.6 \times 10^{-3}$ , respectively. The electric far-fields obtained by the MoM, and the MBF only and MBF-DCA solvers are plotted in Fig. 5.4, and an excellent agreement among the three traces can be observed. The results in Fig. 5.3 and Fig. 5.4 serve as validation for the accuracy of the MBF-DCA solver, before moving to large arrays.

The MBF-DCA solver is used to analyze irregular arrays with 32, 64, 128, 256 and 512 elements. In Table 5.1 the memory requirement to store the DCA factors and near interactions, for varying array sizes are shown. Moreover, the memory requirement after the directional cluster bases and near interaction blocks are reduced with MBF definitions are also shown. Taking advantage of the EFIE MoM matrix symmetry, only the  $U^e$ ,  $B^e$



**Figure 5.3:** MoM current coefficient for antenna array of 32 elements, and the current errors using MBF only and MBF-DCA solvers.



**Figure 5.4:** Electric far-field of antenna array of 32 elements using the MoM, and MBF only and MBF-DCA solvers.

are stored, and only half of the near interactions and coupling matrices are stored. A very good improvement in the total required memory due to the MBF reduction can be noted. In Fig. 5.5, the average number of neighbors and average ranks at level  $L$ ,  $L - 1$ , and  $L - 2$  are shown, where  $L$  is the leaf level. A fairly constant average number of neighbors can be noted. As for the rank, it can be observed that the ranks at level  $L$  converge as the

array size increases. The rank at level  $L - 1$  is not converged yet, since for smaller arrays only a fraction of the  $L - 1$  clusters have admissible interaction. Only the 512-element array has level  $L - 2$  interactions, as seen in the graph. Moreover, the total memory associated with the DCA before and after the MBF reduction for varying array sizes are plotted in Fig. 5.6. The total memory requirement of the MBF reduced matrix is also shown in Fig. 5.6. The memory scaling for the DCA before MBF reduction conforms to the expected log-linear scaling, which is due to the directional and nested properties of the DCA algorithm.

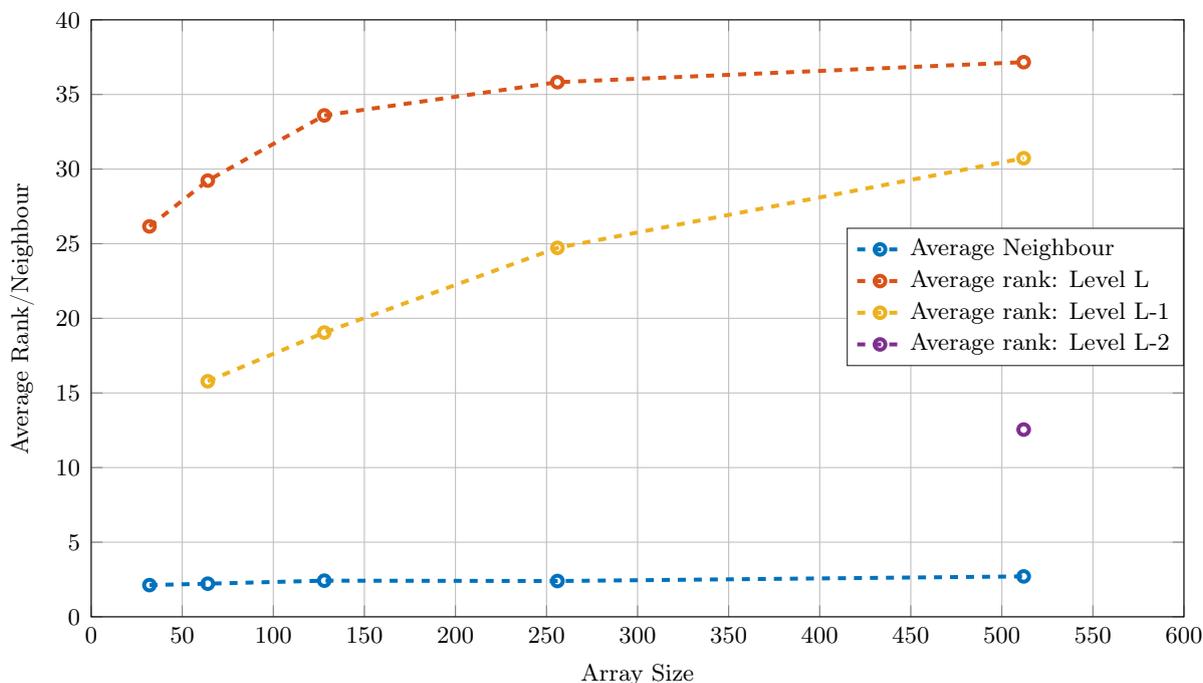
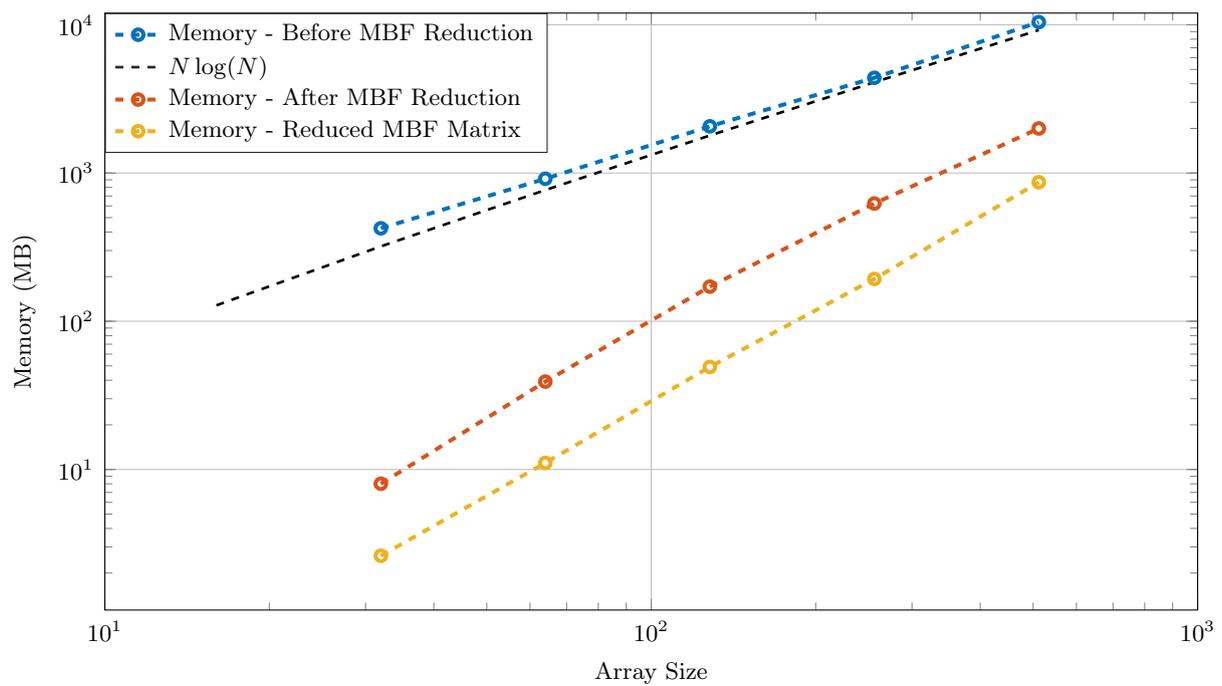


Figure 5.5: Average number of neighbors and ranks for varying array sizes.

## 5.6 Conclusion

In this chapter, the DCA algorithm has been used to efficiently represent and compute the reaction terms in MBF solvers. Firstly, the accuracy of using the MBF-DCA solver has been validated by comparing its solution to that of the MoM, for analysis of a 32-element array. Secondly, very favorable, log-linear memory scaling when using the DCA to compute MBF reaction terms has been shown for large-scale antenna array analysis.

If the objective was to only compress the original MoM matrix as efficiently as possible with the multilevel nested DCA, then the antenna elements would be subdivided to form further levels. However, for representing the reduced matrix in compressed format, these results do represent optimal values.



**Figure 5.6:** Memory requirement for the DCA, before and after combining with MBFs. Memory requirement for the reduced MBF matrix is also shown.

	Before MBF Reduction (MB)	After MBF Reduction (MB)
Array size: 32 elements; Number of DoFs	27136 (DoFs)	414 (DoFs)
Near Interaction	378.56	0.09
Directional cluster Bases ( $U^e$ )	37.92	0.60
Directional Transfer Matrices ( $B^e$ )	-	-
Coupling Matrices ( $S$ )	7.32	7.32
<b>Total</b>	<b>423.80</b>	<b>8.01</b>
Array size: 64 elements; Number of DoFs	54272 (DoFs)	852 (DoFs)
Near Interaction	790.03	0.19
Directional cluster Bases ( $U^e$ )	87.39	1.37
Directional Transfer Matrices ( $B^e$ )	0.51	0.51
Coupling Matrices ( $S$ )	37.08	37.08
<b>Total</b>	<b>915.01</b>	<b>39.15</b>
Array size: 128 elements; Number of DoFs	108544 (DoFs)	1795 (DoFs)
Near Interaction	1711.73	0.47
Directional cluster Bases ( $U^e$ )	186.98	3.09
Directional Transfer Matrices ( $B^e$ )	4.02	4.02
Coupling Matrices ( $S$ )	163.68	163.68
<b>Total</b>	<b>2066.41</b>	<b>171.26</b>
Array size: 256 elements; Number of DoFs	217088 (DoFs)	3556 (DoFs)
Near Interaction	3379.57	0.91
Directional cluster Bases ( $U^e$ )	394.55	6.46
Directional Transfer Matrices ( $B^e$ )	16.01	16.01
Coupling Matrices ( $S$ )	599.39	599.39
<b>Total</b>	<b>4389.52</b>	<b>622.77</b>
Array size: 512 elements; Number of DoFs	434176 (DoFs)	7536 (DoFs)
Near Interaction	7614.99	2.29
Directional cluster Bases ( $U^e$ )	831.65	14.43
Directional Transfer Matrices ( $B^e$ )	45.34	45.34
Coupling Matrices ( $S$ )	1936.69	1936.69
<b>Total</b>	<b>10428.67</b>	<b>1998.75</b>

**Table 5.1:** Memory requirement for various antenna array sizes.

## Chapter 6

# Inverse Fast Multipole Method Based on DCA Factors

### 6.1 Introduction

In this chapter, the single-level version of the DCA is formulated together with a sparse direct solver scheme to obtain antenna array MoM solutions directly. The direct solver is an extension of the single-level Inverse Fast Multipole Method (IFMM) [23, 24] for the directional case. This represents the first combination of the IFMM scheme with *directional* low-rank factorization, as well as its application to high-frequency electromagnetic analysis, to the author's knowledge.

Related to the above work and also reported in this chapter, is a single-level compression scheme which is devised for a domain Green's function method (DFGM) based solver.

### 6.2 IFMM

The IFMM is an error-controllable direct solver based on the  $\mathcal{H}^2$ -matrix formulation. In the IFMM, an extended sparse matrix is created by introducing auxiliary variables. Despite the extended sparse matrix being larger in size compared to the MoM impedance matrix, the idea is to exploit the sparsity of the matrix to obtain a favorable complexity. However, performing Gaussian elimination on the sparse matrix will lead to undesirable fill-ins which are harmful to the complexity. In the IFMM algorithm, some of the fill-ins will be discarded and redirected by updating existing interactions. Extending the system allows for such redirection. To formulate the extended sparse matrix based on the single-level DCA, the MoM impedance matrix,  $Z$ , is written as

$$Z = Z^{\text{Near}} + U^{e_1, \dots, e_d} S V^{e_1, \dots, e_d}. \quad (6.1)$$

$Z^{\text{Near}}$  is a sparse matrix, where the block entries are the self and near MoM interactions.  $U^{e_1, \dots, e_d}$  is a block-diagonal sparse matrix, where each diagonal block is the concatenated directional cluster bases of an antenna. The superscript ' $e_d$ ' is the number of directional far-field regions. The matrix  $V^{e_1, \dots, e_d}$  is the transpose of  $U^{e_1, \dots, e_d}$ .  $S$  is matrix whose block entries are the coupling matrices. The next step is to define the  $x$ ,  $y$  and  $z$  auxiliary

variables, such that

$$x = I_{\text{MoM}} \quad (6.2)$$

$$y = V^{e_1, \dots, e_d} x \quad (6.3)$$

$$z = Sy. \quad (6.4)$$

By substituting equations (6.2) to (6.4) into (6.1), the MoM matrix equation,  $ZI_{\text{MoM}} = V$ , can be re-written as

$$Z^{\text{Near}} x + U^{e_1, \dots, e_d} z = V. \quad (6.5)$$

Using equations (6.3) to (6.5), the extended sparse matrix equation,  $EJ = B$ , is expressed as follows, where the  $-I$  are negative identity matrices:

$$\begin{bmatrix} Z_{\text{near}} & U^{e_1, \dots, e_d} & \\ V^{e_1, \dots, e_d} & & -I \\ & -I & S \end{bmatrix} \begin{bmatrix} x \\ z \\ y \end{bmatrix} = \begin{bmatrix} V \\ 0 \\ 0 \end{bmatrix}. \quad (6.6)$$

However, unlike in the non-directional  $\mathcal{H}^2$ -Matrix IFMM in [23] and [24], it is found that the resulting block diagonals are not well-conditioned during the Gaussian elimination process for the directional case, thus leading to inaccurate results. For this reason, a row permutation is performed on (6.6) so that the  $-I$  blocks are on the diagonal, ensuring well-conditioned diagonal blocks throughout the Gaussian elimination process:

$$\begin{matrix} x : \\ z : \\ y : \end{matrix} \begin{bmatrix} Z_{\text{near}} & U^{e_1, \dots, e_d} & \\ & -I & S \\ V^{e_1, \dots, e_d} & & -I \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} V \\ 0 \\ 0 \end{bmatrix}. \quad (6.7)$$

The implication of performing the row permutation on the extended sparse matrix is that a new set of rules for the elimination and re-direction of fill-ins must be devised. These are explained in [1].

### 6.2.1 Application of Non-Directional IFMM to the MoM in Combination with MBFs

Intermediate works regarding the application of IFMM to the MoM for array analysis is presented in [5, 6]. In [5], the NCA factors from Section 4.2 serve as an input to the single-level non-directional IFMM, that is (6.6), to obtain the MoM solution for antenna arrays. As mentioned, performing Gaussian elimination will lead to fill-ins. The procedures described in [24] is used for the elimination and redirection of the fill-ins. In [6], the IFMM is used to solve the reduced MBF system, rather than the original MoM system. The NCA cluster bases are pre-multiplied by the MBF definitions similarly as with the DCA in Chapter 5.

### 6.2.2 Application of Directional IFMM to the MoM

As previously explained, the NCA is not suitable for electromagnetic simulation in the high-frequency regime, since the rank will increase with the increasing size of clusters if the far-field region is not partitioned while computing the low-rank factors. As such, the DCA is combined with the directional IFMM in [1]. As mentioned, using the row permuted extended sparse matrix in (6.7) leads to a new set of rules regarding the elimination

Scenario	Runtime (s)	Memory (GB)
Array size: 32 elements; Original $E$ memory: 4.21 GB; Full MoM memory: 10.97 GB		
No elimination and redirection of fill-ins	398.94	10.16
Elimination and redirection of all compressible fill-ins	153.68	5.80
No elimination and redirection of $xy$ fill-ins	145.02	6.06
Array size: 64 elements; Original $E$ memory: 8.27 GB; Full MoM memory: 43.89 GB		
No elimination and redirection of fill-ins	–	–
Elimination and redirection of all compressible fill-ins	674.88	14.21
No elimination and redirection of $xy$ fill-ins	585.12	15.87

**Table 6.1:** Runtime and memory requirements for Gaussian elimination of extended sparse matrices.

and redirection of fill-ins. Three types of re-directable fill-ins will occur, namely the  $xx$ -interaction fill-ins,  $xy$ -interaction fill-ins and  $zy$ -interaction fill-ins. The details of these re-directions are explained in [1].

The runtime and memory requirements to perform Gaussian elimination on (6.7) for irregular antenna arrays of 32 and 64 elements, using the LPDA antenna element shown in Fig. 5.1 is presented in Table 6.1. The results are shown for three scenarios, namely, ‘No elimination and redirection of fill-ins’, ‘Elimination and redirection of all compressible fill-ins’, and ‘No elimination and redirection of  $xy$  fill-ins’. The case of ‘No elimination and redirection of fill-ins’ means performing Gaussian elimination without removal of any fill-ins. This case has the highest runtime and memory requirement. In fact no results is presented for the 64 element array since the simulation exceeded the available memory. Elimination of  $xy$ -interaction fill-ins are expensive since a large amount of these fill-ins occur which needs to be re-directed individually. As seen in Table 6.1, the case of ‘No elimination and redirection of  $xy$  fill-ins’ is actually faster than the ‘No elimination and redirection of fill-ins’. In both solutions, the direct solver requires less than twice the storage of the original compressed matrix itself, which is considered a very good result. In future work, the possibility of redirecting the  $xy$ -interactions in a faster way could be investigated.

### 6.3 Hybrid Single-Level ACA Compression for the IRB-DGFM

Another application of single-level ACA technology for efficient array analysis came in the form of a hybrid compression scheme developed by the present author for the iterative radius-based domain Green’s function method (IRB-DGFM) developed by M. Chose [2, 6]. The IRB-DGFM is a substantial extension of the standard DGFM formulation [63, 64, 65]. It allows for convergence to the true MoM solution via iterations involving only solving

Array elem.	Tot. DoFs	ACA IRB-DGFM	MLFMM (FEKO)
625	442,500	8.22 GB	6.67 GB
961	680,388	13.71 GB	10.78 GB
1225	867,300	17.43 GB	14.91 GB

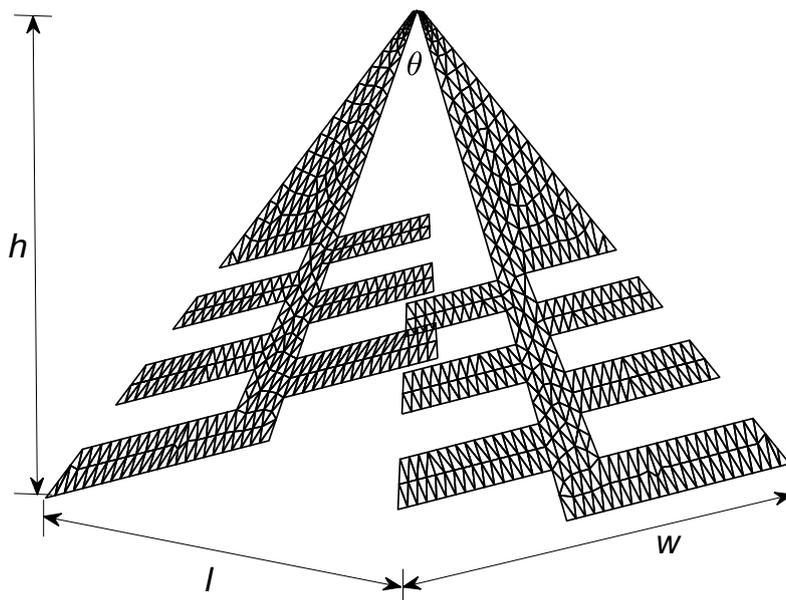
**Table 6.2:** Memory requirements for solving bent log-periodic dipole antenna arrays of various sizes, at 300 MHz. For the ACA IRB-DGFM solutions the optimal value of  $R_\tau = 8\lambda$  is used.

local problems associated with each array element. A single-level ACA approach with each element constituting a group of DoFs, is well suited to efficiently and repeatedly set up the many required local matrices (requiring the reconstitution of inter-antenna coupling blocks). It is a hybrid scheme, meaning that standard ACA (see Section 3.3.1) is used for non-self interactions up to an inter-element distance denoted  $R_\tau$ , beyond which the ranks are stabilized at close to their lowest value. For all these low-rank interactions beyond inter-element distance  $R_\tau$ , single-level NCA is employed [5]. In [2] this latter scheme is referred to as translation-based ACA (TACA). Given the requirement that any inter-antenna coupling block should be easy to reconstitute, the hybrid single-level scheme is very efficient.

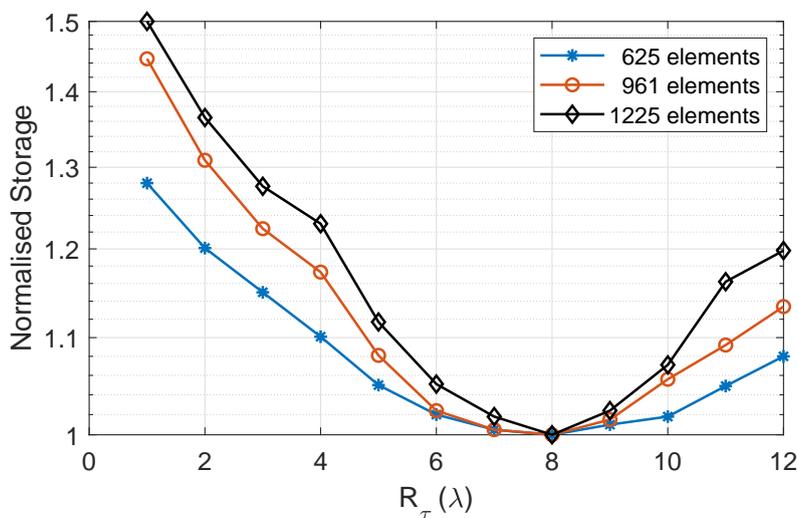
Consider the bent log-periodic dipole antenna shown in Fig. 6.1. The antenna is designed for a  $-10$  dB bandwidth of 100 MHz to 1.5 GHz [62]. The average mesh size is set at  $\lambda/20$ . The mesh shown in the figure is for the analysis frequency of 300 MHz, yielding 708 DoFs per antenna. Three arrays with antennas placed regularly on a rectangular grid are considered, with  $\lambda/4$  inter-element spacing in both directions. The configurations are square with  $25 \times 25 = 625$ ,  $31 \times 31 = 961$  and  $35 \times 35 = 1225$  elements, respectively. For each of these arrays, the value of  $R_\tau$  is varied and the total storage recorded. Fig. 6.2 shows the results, which affirms that for sufficiently large arrays  $R_\tau$  will have an optimal value independent of the exact array layout and the total array diameter. Furthermore, the nature of the memory dependence upon  $R_\tau$  is as expected. For small values of  $R_\tau$  the TACA ranks will be large which result in the translation factors requiring more storage. For large values of  $R_\tau$  the TACA ranks have already converged and increasingly, matrix blocks are unnecessarily being compressed with more expensive standard ACA factorization. Table 6.2 lists the memory requirements at the optimal value of  $R_\tau = 8\lambda$ . For comparison, the memory requirements for the MLFMM solver in FEKO [66], are also listed in the table. It can be seen that the hybrid ACA scheme is quite competitive with the MLFMM for these large arrays, though for sufficiently large arrays the MLFMM scaling is well known to be superior. However, the MLFMM is not suitable for IRB-DGFM acceleration due to its nested multilevel structure. A comprehensive description of the compression scheme, further performance assessments and further discussion of the IRB-DGFM scheme's merits are provided in [2].

## 6.4 Conclusion

In this chapter, the factors from the single level versions of the NCA and DCA algorithms described in Chapter 4 serve as inputs to a sparse direct solver, namely a modified version of the IFMM, for array analysis. Additionally, a hybrid single-level ACA compression



**Figure 6.1:** Mesh at 300 MHz, of a bent log-periodic dipole antenna, with dimensions  $l = 468.05$  mm,  $w = 926.10$  mm,  $h = 270.23$  mm and  $\theta = 120^\circ$ .



**Figure 6.2:** Storage requirements for the ACA IRB-DGFM solution of arrays with varying numbers of elements, of the bent log-periodic dipole antenna shown in Fig. 6.1, at 300 MHz. Each trace is obtained by varying  $R_\tau$  and recording the total memory required for the ACA factors and then normalizing the data for each array with respect to the minimum storage achieved for that array. The actual minimum values are listed in Table 6.2.

scheme is devised for the acceleration of the IRB-DGFM solver. The acceleration algorithm combines standard ACA and single-level NCA compression. Both fast schemes are shown to be very efficient.

The possibility and challenges of combining DCA-based IFMM with MBFs is discussed in [1].



# Chapter 7

## Conclusion

This work is focused on developing efficient MoM-based algorithms for the analysis of large antenna arrays, such as being considered as part of the international SKA radio astronomy project currently under development. A variety of techniques based on cross approximation is explored and implemented. In this context, two DCA-based solvers are devised for antenna array analysis. The DCA is a nested multilevel algorithm, which efficiently compresses MoM sub-blocks due to far-interactions as a product of low-rank factors. During the computation of these factors, the far-field is segmented in angular sectors to ensure the numerical rank is limited irrespective of the cluster size.

Firstly, the DCA is combined with the MBF technique. One of the MBF solvers' bottlenecks is the high computational cost associated with the fill-in of the reduced impedance matrix due to matrix-vector products. As such, the DCA algorithm is used to efficiently represent and compute the reaction terms in MBF solvers. The accuracy of using the MBF-DCA solver is validated, and a favorable memory scaling is obtained.

As a second contribution, the single-level version of the DCA is formulated together with a sparse direct solver scheme, based on the IFMM, to obtain antenna array MoM solutions directly. The original IFMM formulation is extended for the directional case, and a new procedure to eliminate and redirect compressible fill-ins during the Gaussian elimination of the sparse matrix is devised. This is the first use of a directional  $\mathcal{H}^2$ -matrix based IFMM for high-frequency electromagnetic analysis, to the author's knowledge.

Lastly, a hybrid single-level ACA compression scheme is devised to accelerate the IRB-DGFM solver. The compression algorithm combines the standard ACA to compress intermediate interactions, and the single-level NCA to represent far interactions efficiently.

This work has shown the potential benefits of the recently proposed DCA scheme (both in multilevel and single-level), as well as the NCA, to improve the efficiency of MoM-based electromagnetic simulation of antenna arrays, with prospective applications well beyond this niche application. The DCA factorization scheme is fairly complex, but never the less it can be implemented on top of any existing RWG MoM code without having to implement alternative representations of the Green's function. In this respect, it has a distinct advantage over the multilevel-FMM. It is expected that beneficial applications of the DCA within full-wave electromagnetic analysis will continue to be explored in the coming years.

An interesting objective for future research would be to extend the DCA-IFMM solver to a multilevel version, combined with the MBFs for further improved efficiency for sparse direct solutions of large antenna arrays.



## Appendix A

### Journal paper — DCA-based IFMM solver [1]

K. Sewraj and M. M. Botha, “A Sparse Direct Solver Based on Directional Cross Approximation for Antenna Array Analysis,” in preparation.

# A Sparse Direct Solver Based on Directional Cross Approximation for Antenna Array Analysis

Keshav Sewraj, and Matthys M. Botha, *Member, IEEE*

**Abstract**—The well-established method of moments (MoM) is highly suitable for analyzing antenna arrays; however, limited by the electrical size due to the high computational cost associated with the MoM. An MoM-based direct solver is presented in this paper to analyze large antenna arrays. A rank-revealing method, namely the single-level version of the directional cross approximation (DCA) technique, is used to approximate and compress interactions of far antennas. The DCA factors and near interaction MoM sub-blocks are then used to construct an extended sparse matrix. The matrix equation is then solved using an approximate direct solver called the inverse fast multipole method (IFMM). In this technique, the compressible fill-ins which occur during the Gaussian elimination of the sparse matrix are discarded and redirected, to preserve the sparsity of the matrix. This is the first extended sparse direct solver based on the DCA, as well as its application to high-frequency electromagnetic analysis, to the authors' knowledge. Finally, the idea of incorporating macro basis functions (MBF) in the DCA-IFMM solver for antenna array analysis will be discussed.

**Index Terms**—Antenna array, directional cross approximation (DCA), inverse fast multipole method (IFMM), electric field integral equation (EFIE), method of moments (MoM).

## I. INTRODUCTION

LARGE antenna arrays have a growing importance in many areas such as communication, radar systems, remote sensing, and radio-astronomy, to name a few. One such radio-astronomy project is the Square Kilometre Array (SKA) [1], which intends on building large antenna arrays as interferometers within the Mid-Frequency Aperture Array (MFAA). The antenna design to be used and the array structure (regular or irregular, and dense or sparse) are still in the research phase, however, a few thousands of antennas can be expected due to the high sensitivity requirement.

During the design phase of such arrays, numerical electromagnetic analysis is an essential tool. The well-established Method of Moments (MoM) is often the technique of choice for antenna analysis due to the nature of the integral-equation, which makes it appropriate for radiation problems. However, the memory requirement and computational time to solve the MoM matrix equation scale as  $\mathcal{O}(N^2)$  and  $\mathcal{O}(N^3)$  respectively, where  $N$  is the number of basis functions. Thus, electromagnetic analysis with the MoM is restricted by the electrical size of the problem. For large structures, a variety of fast numerical techniques have been developed on top of the

MoM over decades, and these fast techniques can be categorized as either being an iterative or direct solution. The use of techniques such as the Fast Multipole Method (FMM) [2] and its multi-level version, Adaptive Integral Method [3], among others, have been very successful in drastically accelerating matrix-vector products in Krylov subspace iterative solvers. However, for antenna array analysis, the iterations need to start anew for each excitation scheme, e.g. for Embedded Element Pattern (EEP) analysis of multiple elements, which is costly. Thus, direct solvers are important for antenna array analysis.

One such direct solution is the Macro Basis Function (MBF) method [4], with variants and related techniques such as the Characteristic Basis Function Method (CBFM) [5], Synthetic-Function Approach (SFX) [6], Eigencurrent method [7], among others. In the MBF method, physics-based functions called MBFs are defined for each antenna, through linear combinations of the low-level basis functions defined on that domain. Since the number of MBFs defined is fewer than the initial degrees of freedom (DoFs) by a few orders of magnitude, the MBFs are used to create a reduced matrix system, which can then be solved directly. The computation of reaction terms during the fill-in of the reduced matrix can be very expensive, and fast techniques are used to improve the computation time. The concept of using macro basis functions for array analysis has proven to be very effective; for instance in [8], the CBFM was used to analyze large regular arrays and the Adaptive Cross Approximation (ACA) [9] was employed to rapidly compute the reaction terms during the fill-in process of the reduced matrix. The ACA is an algebraic algorithm which factorizes a low-rank MoM sub-block,  $Z_{pq}$ , without the knowledge of all the entries in the block, such that  $Z_{pq} \approx U_{pq}V_{pq}$ . The size of  $U_{pq}$  and  $V_{pq}$  are  $N_i \times R$  and  $R \times N_i$ , where  $N_i$  is the number of basis function per antenna and  $R$  is the numerical rank of the interaction according to a pre-defined threshold. For large irregular arrays, the ACA might not be very effective since the low-rank factorization algorithm needs to be performed  $\mathcal{O}(M^2)$  times during the reduced matrix fill-in phase, where  $M$  is the number of antennas in the array. There is also a more recently proposed version of the ACA, namely the Directional Cross Approximation (DCA) [10], which strictly retains the efficiency of the ACA for oscillatory kernels, as the standard ACA ranks will grow for blocks of increasing electrical size. The DCA in multilevel form has very recently been applied to electromagnetic field analysis for the first time, to construct the reduced system efficiently [11], [12]. For the analysis of large irregular arrays in [13], the interactions between pairs of MBFs over a “baseline” were computed beforehand. After some physical transformations,

This work was supported in part by the South African Radio Astronomy Observatory (SARAO) and in part by the National Research Foundation under Grant 75322 and Grant 106033. (Corresponding author: Matthys M. Botha.)

The authors are with the Department of Electrical and Electronic Engineering, Stellenbosch University, Stellenbosch 7600, South Africa (e-mail: 20635958@sun.ac.za, mmbotha@sun.ac.za).

a smoothed function was fitted in a harmonic-polynomial representation. During the fill-in phase of the MBF reduced matrix, the reaction terms were rapidly computed using the harmonic-polynomial representation. Moreover, in [14], the multipole expansion was used to rapidly compute interaction between MBFs.

Another advancement for fast integral-equation based solvers is the  $\mathcal{H}^2$ -matrix structure.  $\mathcal{H}^2$ -matrices can exploit the data-sparse structure of the MoM impedance matrix. The matrix sub-blocks between far interacting antennas are low-ranked as discussed and can be approximated as  $Z_{pq} \approx U_p S_{pq} V_q$ . Multilevel versions of the  $\mathcal{H}^2$ -matrix have a nested property, that is, factors at a parent level can be constructed from that of its children. The  $\mathcal{H}^2$ -matrix formulation can then be solved iteratively or using a direct solver. One such direct solver for  $\mathcal{H}^2$ -matrix is the Inverse Fast Multipole Method (IFMM) introduced in [15]. The IFMM creates an extended sparse matrix equation which satisfies the MoM  $\mathcal{H}^2$ -matrix formulation. The extended sparse matrix consists of the  $U_p$ ,  $S_{pq}$  and  $V_q$  factors, and also near and self MoM sub-block interactions. Solving the sparse matrix through Gaussian elimination will lead to fill-ins. Compressible fill-ins are discarded and re-routed during the elimination process in order to preserve the sparsity of the matrix and the favourable complexity of the algorithm.

In this paper, a sparse direct solver based on the DCA factorization combined with an IFMM approach modified to the DCA, is presented for array analysis. The formulation is for the full MoM system. The possibility and challenges of combining this new solver in future with MBFs for sparse-direct solution of the reduced system, is discussed. In Section II, the integral equation formulation which forms the basis of the MoM will be introduced. In Section III, the single-level version of the DCA is employed to efficiently compress far interactions. The DCA has a directional  $\mathcal{H}^2$ -matrix structure, and the term ‘directional’ will be explained in Section III; however, it is used to limit the numerical rank of the DCA factors. The new sparse-direct DCA-IFMM solver is presented in Section IV. Finally, the noted discussion on possible future extensions of this work follows in Section V, with the paper concluded in Section VI.

## II. MOM FORMULATION

The Electric Field Integral Equation (EFIE) described in (1) is used in the paper and is defined over the surface of the antennas in the array, assumed to be Perfectly Electrical Conductor (PEC).

$$\hat{n} \times \mathbf{E}^{\text{inc}}(\mathbf{r}) = \hat{n} \times \int_S \left[ jk\eta \mathbf{J}(\mathbf{r}') G(\mathbf{r}, \mathbf{r}') + \frac{\eta}{jk} (\nabla' \cdot \mathbf{J}(\mathbf{r}')) \nabla' G(\mathbf{r}, \mathbf{r}') \right] dS', \quad (1)$$

where  $\hat{n} \times (\cdot)$  is the tangential component,  $k$  is the wave number,  $\eta$  is the intrinsic impedance, and  $G(\mathbf{r}, \mathbf{r}')$  is the Green’s function, with  $\mathbf{r}'$  and  $\mathbf{r}$  being position vectors on the source and observer domains, respectively.  $\mathbf{E}^{\text{inc}}(\mathbf{r})$  is the incident electric field and  $\mathbf{J}(\mathbf{r}')$  is the unknown surface current

that needs to be solved for. The Green’s function in (1) is given as in

$$G(\mathbf{r}, \mathbf{r}') = \frac{e^{-jk|\mathbf{r}-\mathbf{r}'|}}{4\pi|\mathbf{r}-\mathbf{r}'|}. \quad (2)$$

After discretizing the EFIE with Rao-Wilton-Glisson (RWG) [16] basis and testing function, the MoM matrix equation is obtained as

$$Z I_{\text{MoM}} = V_{\text{MoM}}, \quad (3)$$

where  $Z$  ( $N \times N$ ) is the impedance matrix containing the interactions between basis and testing functions,  $V_{\text{MoM}}$  is the excitation vector, and  $I_{\text{MoM}}$  is the unknown current coefficients to solve for.  $N$  is the total number of RWG functions defined over the  $M$  antennas. As mentioned in Section I,  $N_i$  refers to the number of RWG functions per antenna.

## III. DIRECTIONAL CROSS APPROXIMATION

In this section, the background of the DCA [10] is reviewed, which will be used to compress the far interactions in the MoM impedance matrix. In the DCA, a rank-deficient sub-block,  $Z_{pq}$ , is factorized as follows:

$$Z_{pq} \approx U_p^e S_{pq} V_q^{-e}, \quad (4)$$

where  $U_p^e$  ( $N_i \times R_p$ ) and  $V_q^{-e}$  ( $R_q \times N_i$ ) are called directional cluster bases belonging to antennas  $p$  and  $q$ , respectively.  $R_p$  and  $R_q$  are the numerical ranks of antennas  $p$  and  $q$  respectively, where  $\max(R_p, R_q) \ll N_i$ . The directional cluster bases describe the interaction between an antenna and its directional far-fields. For a given antenna in the array, the far-field (to be defined in (5)) is subdivided into a pre-defined number of sectors, with the angle being dependent on the size of the antenna. The superscript ‘ $e$ ’ and ‘ $-e$ ’ in (4) refers to the direction of the cluster bases.  $S_{pq}$  ( $R_p \times R_q$ ) is the coupling matrix representing the interaction between the source and observer domains, analogous to the translation operator in the FMM.

If an omnidirectional far-field is used, the numerical rank of the cluster basis increases with the size of the domain when the integral equation contains an oscillatory Green’s function (described in (1) and (2)), and operating in the high-frequency realm, that is,  $kD > 1$ , where  $k$  is wavenumber and  $D$  is the diameter of the bounding box enclosing an antenna. For high-frequency problems such as the analysis of large arrays of wideband antennas, a directional approach can be used instead of a non-directional  $\mathcal{H}^2$ -matrix structure so as to ensure that the rank is bounded irrespective of the domain size by limiting the angular far-field.

The distance and angular admissibility criteria to compress an observer-source interaction with the DCA are respectively described as:

$$\eta_{\text{high}} \text{dist}(p, q) \geq kD^2 \quad (5)$$

$$\beta \propto \left( \frac{1}{kD} \right). \quad (6)$$

In (5),  $\text{dist}(p, q)$  refers to the minimum distance between the bounding boxes enclosing the antennas  $p$  and  $q$ ,  $\eta_{\text{high}}$  is a parameter to be set,  $D^2$  refers to the square of the diameter

of the bounding box enclosing an antenna element, and  $\beta$  is the angle of the sector representing a directional far-field.

In the single-level DCA,  $\mathcal{O}(M)$  directional cluster bases with sizes  $N_i \times R$  and  $\mathcal{O}(M^2)$  coupling matrices with sizes  $R \times R$  need to be computed and stored, as opposed to the ACA, where  $\mathcal{O}(M^2)$   $U_{pq}$  and  $V_{pq}$  factors need to be stored. Since,  $R \ll N_i$ , the use of the DCA is more efficient than the ACA for the analysis of large irregular arrays.

### A. Computing DCA Factors

The  $U^e$  and  $V^{-e}$  cluster bases are obtained through an inverse source process and are defined as in (7) and (8).

$$U^e = Z_t \sigma_t^e (Z_{\tau_t^e} \sigma_t^e)^{-1} \quad (7)$$

$$V^{-e} = (Z_{\tau_s^{-e}} \sigma_s^{-e})^{-1} Z_{\tau_s^{-e}} \quad (8)$$

In (7) and (8),  $t$ ,  $s$ ,  $\tau_t^e$ ,  $\sigma_t^e$ ,  $\tau_s^{-e}$ , and  $\sigma_s^{-e}$  are sets of indices on the source and observer antennas, and their directional far-fields. The descriptions of the indices are given in Table I, and the details on how to obtain these indices is explained later in this section.

TABLE I  
DESCRIPTION OF INDICES DURING THE CONSTRUCTION OF DIRECTIONAL CLUSTER BASES.

Index	Description
$t$	All the testing functions on the observer antenna
$s$	All the basis functions on the source antenna
$\tau_t^e$	Subset of testing functions on the observer antenna in regards to the directional far-field, $e$
$\sigma_t^e$	Subset of basis functions on the antennas within the observer's directional far-field, $e$
$\tau_s^{-e}$	Subset of testing functions on the antennas within the source's directional far-field, $-e$
$\sigma_s^{-e}$	Subset of basis functions on the source antenna in regards to the directional far-field, $-e$

The coupling matrix is then denoted as the interaction between the  $\sigma_s^{-e}$  basis functions on the source domain and the  $\tau_t^e$  testing functions on the observer domain as:

$$S = Z_{\tau_t^e} \sigma_s^{-e}. \quad (9)$$

As a consequence of the EFIE MoM matrix symmetry, the cluster basis  $V^e$  of an antenna with directional far-field ' $e$ ' is the transpose of  $U^e$ , hence only  $U^e$  bases are computed. As described in (7), the local domain indices,  $\tau_t^e$ , and far-field domain indices,  $\sigma_t^e$ , must be available to compute  $U^e$ . To obtain these indices, the matrix  $Z_{\tau_t^e} \sigma_t^e$  is first constructed which represents an incomplete interaction between the antenna and its directional far-field, where  $\tau_t^e$  and  $\sigma_t^e$  are sampled indices of the local and far-field RWGs such that  $t \supseteq \tau_t^e \supset \sigma_t^e$  and  $N_f \supset \sigma_t^e \supset \sigma_s^{-e}$ , and  $N_f$  is a set of indices of all RWGs belonging to antennas in  $e$ . The ACA algorithm with thresholding is then performed on the matrix  $Z_{\tau_t^e} \sigma_t^e$ , and the rows and columns indices selected through the ACA process are denoted as  $\tau_t^e$  and  $\sigma_t^e$  respectively. Ideally, ACA should have been performed on the matrix  $Z_{tN_f}$ , however this operation is impractical from a computational point of view since  $N_f$  is too large and will also lead to the runtime being proportional to  $M$ . The sample  $\sigma_t^e$  is obtained

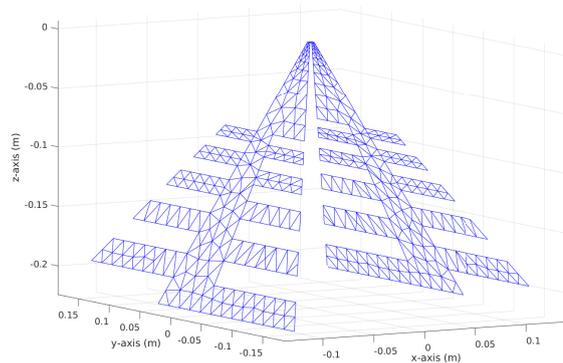


Fig. 1. Mesh of bent LPDA antenna, with 5 teeth, used in this paper. Number of DoFs is 790.

by placing a Chebyshev grid on top of bounding boxes of each antenna in the directional far-field. RWGs closest to a pre-defined number of Chebyshev nodes are then selected for every directional far-field antenna and the union of all sampled RWGs is denoted as  $\tilde{\sigma}_t^e$ . The number of Chebyshev nodes per antenna,  $C_i = \text{ceil}\left(c \frac{N_i}{M_f}\right)$ , where  $M_f$  is the number of far-field antennas and  $c$  is a constant set to 1 in this implementation. However,  $C_i \rightarrow 1$  as  $M_f \rightarrow \infty$ , hence a minimum constraint of  $C_i$  is included so as  $C_i \geq 2^\nabla$ , where  $\nabla$  is number of dimensions of the bounding box (e.g. 2D or 3D). Moreover, the number of Chebyshev nodes on each axis is set to be proportional to the lengths of the bounding box axes. In this implementation,  $\tilde{\tau}_t^e = t$  since  $c = 1$ . Otherwise if  $t$  is too large,  $\tilde{\tau}_t^e$  can be obtained by placing a Chebyshev grid on the local domain, and the  $\text{ceil}(cN_i)$  closest RWGs to the nodes are selected, where  $c < 1$ . To avoid uncontrolled accuracy,  $c < 1$  can only be used if  $N_i$  is much larger than the anticipated rank, and diversity in RWG orientation is ensured.

Once  $\tau_t^e$  and  $\sigma_t^e$  are available for all the antennas, cluster bases and coupling matrices are computed according to (7) and (9). The IFMM algorithm to be described in Section IV takes these DCA factors ( $U^e$ ,  $V^e$ ,  $S$ ) as input.

### B. Selecting DCA Parameters

In this section, numerical experiments are performed to determine the distance parameter,  $\eta_{\text{high}}$ , in (5) and angle of directional far-field sectors,  $\beta$  in (6) for efficient compression performance. A bent LPDA antenna as shown in Fig. 1 is used, operating over the band of 500 MHz to 900 MHz, and the antenna is designed using the Antenna Magus [17] software. The meshing of the antenna is performed at the highest frequency with average mesh size of 0.05 wavelengths, resulting in  $N_i = 790$ . An irregular antenna array consisting of 64 antennas with a minimum inter-element distance of 0.6 m, as shown in Fig. 2, is analyzed.

In Fig. 3, the average number  $\tau_t^e$  indices (similar to the average number of  $\sigma_t^e$  and the numerical average ranks of the DCA factors) of all antennas and directions are plotted by varying the parameters  $\eta_{\text{high}} = \{2, 3, 4\}$ ,  $\beta = \{2\pi, \pi, \frac{\pi}{2}\}$

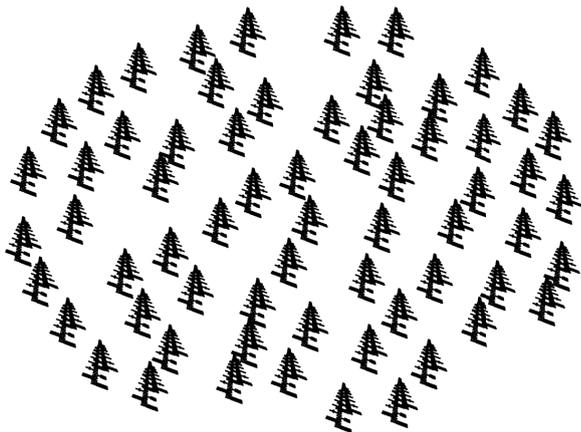


Fig. 2. 64-element irregular array of bent LPDA antennas. The minimum inter-element distance is 0.6m, and the size of the array is 3.5m by 3.5m. Each dot represents the midpoint of an RWG function.

rad, and frequency = {500, 700, 900} MHz. Moreover, an ACA threshold value of  $1e-4$  is used when performing the ACA algorithm on the matrix  $Z_{\tau_t^e \sigma_t^e}$ . The angle  $\beta = 2\pi$  refers to the omnidirectional version of the DCA. As discussed in Section III and from the traces of  $\beta = 2\pi$  in Fig. 3, the numerical rank for the omnidirectional version of DCA is high and tend to increase strongly with frequency. For the cases of  $\beta = \pi$  and  $\beta = \frac{\pi}{2}$  (referring to 2 and 4 directional sectors, respectively) in Fig. 3, the average numerical rank decreases as more directional far-field sectors are included. A high rank is not desired for the performance of the algorithm, but more importantly, a constant rank is required to preserve the asymptotic complexity of the algorithm. Based on observations from Fig. 3, a segmented far-field is needed when computing the indices  $\tau_t^e$  and  $\sigma_t^e$  for the efficient analysis of large arrays. Additionally in Fig. 3, the distance constant,  $\eta_{\text{high}}$  is varied, where  $\eta_{\text{high}} = 2$  refers to far-field being furthest from the local antenna and  $\eta_{\text{high}} = 4$  being the closest. As expected, the average numerical rank decreases with decreasing  $\eta_{\text{high}}$ . The notable increase in numerical rank with frequency for the trace of  $\eta_{\text{high}} = 4$  and  $\beta = \frac{\pi}{2}$  rad suggest that the far-field is too close for the case of  $\eta_{\text{high}} = 4$ . The desired constant rank is effectively achieved from Fig. 3, since the number of directions should be increased with frequency according to (6), for a given value of  $\eta_{\text{high}}$ .

The average number of neighboring antennas for  $\eta_{\text{high}} = 2, 3, \text{ and } 4$  are shown in Fig 4. The number of neighbors is larger for smaller values of  $\eta$  since the far-field is further away. The number of neighbors for the case of  $\eta = 2$  is considered to be too large especially at 900 MHz since these interactions need to be stored in full MoM sub-blocks, hence increasing the computation cost and memory.

The graphs of computational time to generate  $U^e$  and  $S$  for different values of  $\eta_{\text{high}}$ ,  $\beta$ , and frequency are shown in Fig 5 and Fig 6 respectively. It should be noted that the MoM code is not optimized and the importance of the runtime

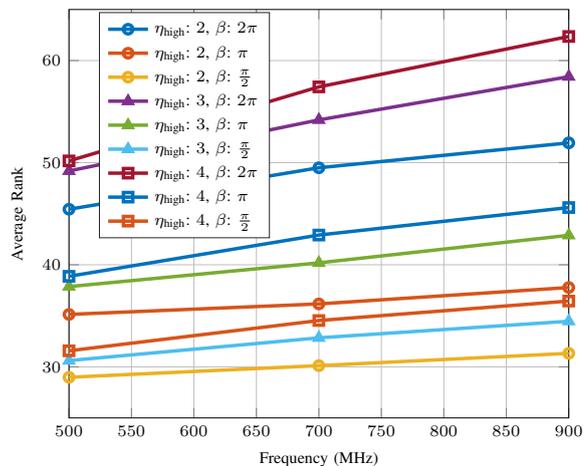


Fig. 3. Average numerical rank of DCA factors when varying  $\eta_{\text{high}}$  and  $\beta$ .

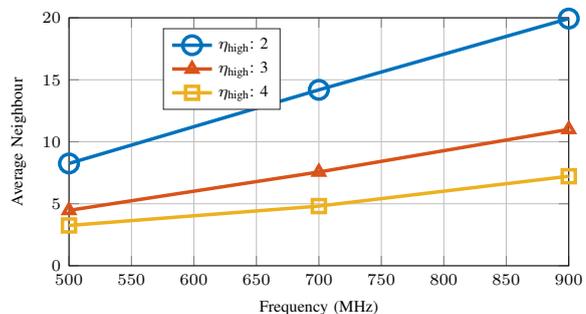


Fig. 4. Average number of neighbors for  $\eta = 2, 3, \text{ and } 4$ .

results is to show the effect of the parameters, rather than the absolute values. For a given value of  $\eta_{\text{high}}$ , it can be observed that the computational time to generate  $U^e$  increases with the number of directional far-field, since for each antenna multiple cluster bases need to be computed. Secondly, for a given value of  $\beta$ , the computational time to generate  $U^e$  decreases with decreasing  $\eta_{\text{high}}$  due to the lower rank. For coupling matrices, the computational time is dependent on the numerical rank and number of far-field antennas. Thus, the case of  $\eta_{\text{high}} = 2$  and  $\beta = \frac{\pi}{2}$  rad leads to the lowest computational cost.

Based on a trade off between average rank and neighbors, and computational time, the values of  $\eta = 3$  and  $\beta = \frac{\pi}{2}$  rad are chosen for this paper.

### C. Performance of DCA Algorithm

In this section, the performance of the DCA algorithm will be demonstrated for irregular antenna arrays consisting of 64, 128, 256 and 512 bent LPDA antennas, with minimum inter-element distance of 0.6 m. The values of  $\eta = 3$  and  $\beta = \frac{\pi}{2}$  rad are chosen as in Section III-B. The average numerical ranks of DCA factors, together with the average neighboring antennas and average far-field antennas, for the different array sizes are shown in Table II. The numerical

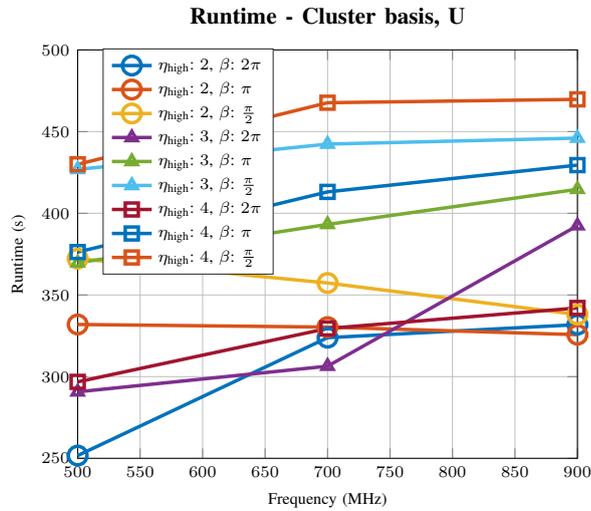


Fig. 5. Computational time to generate directional cluster bases,  $U^e$  for various values of  $\eta$  and  $\beta$ .

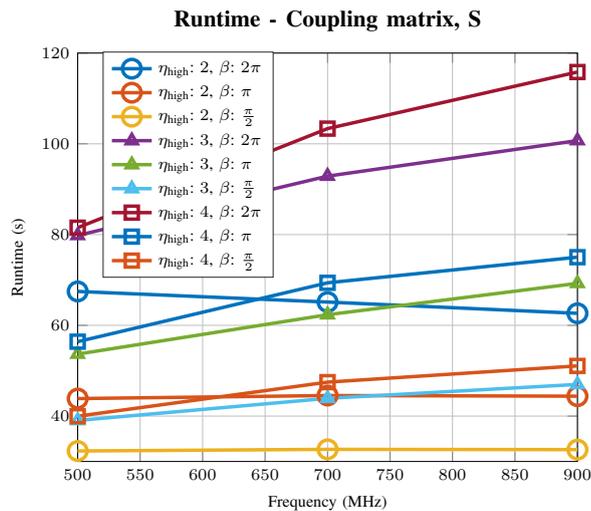


Fig. 6. Computational time to generate directional coupling matrices,  $S$  for various values of  $\eta$  and  $\beta$ .

ranks in Table II are converging with the increasing size of the array. The lower ranks associated with the smaller arrays can be explained by the fact that the ratio of edge elements to middle elements is larger, and edge antennas are expected to have a lower rank.

Fig. 7 shows the computation time to generate the directional cluster bases,  $U^e$ , for varying array sizes. The computation time consist of two parts. Firstly, the time to perform the ACA algorithm on the matrix  $Z_{\tau_t^e \sigma_t^e}$  to obtain the local and far-field selected RWG indices,  $\tau_t^e$  and  $\sigma_t^e$  respectively. Secondly, the fill-in time of the matrix  $Z_{t\sigma_t^e}$  and computation time of creating  $U^e$  cluster bases according to (7).

Two traces for the time to generate  $U^e$  are plotted in Fig. 7,

TABLE II  
AVERAGE NUMERICAL RANK FOR VARYING ARRAY SIZES

Array size	Average Neighboring Antennas	Average Far-Field Antennas	Average Numerical Rank
64	7.56	60.63	32.84
128	8.38	124.26	34.64
256	10.59	251.16	36.56
512	9.85	507.55	37.56

Average rank for array of 512 is 38.06 for special case of  $C_i \geq 2^\nabla$ .

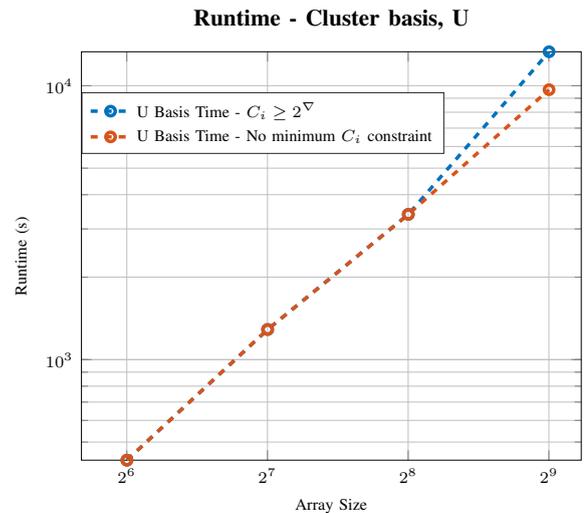


Fig. 7. Computational time to generate directional cluster bases,  $U^e$  for varying array size.  $M$  refers to the number of antennas in the array.

namely with and without the minimum criteria of Chebyshev nodes on far-field antennas,  $C_i \geq 2^\nabla$ . As observed in Fig. 7, a higher time is required for the case of 512 antennas when the minimum criteria is used. This is because for large arrays,  $\text{ceil}\left(c \frac{N_i}{M_f}\right) < 2^\nabla$  and  $C_i$  is set to  $2^\nabla$  at many instances, thus the number of columns in the matrix  $Z_{\tau_t^e \sigma_t^e}$  is no longer independent of the array size. One possible solution for large arrays is to include a radius,  $R_f \gg \frac{kD^2}{\eta_{\text{high}}}$ , which is much larger than the admissibility distance criteria defined in (5). A Chebyshev grid will be placed on only a subset of antennas belonging to the region further than  $R_f$ . The subset could be based on selecting antenna belonging to angular regions which have not been represented yet. Another solution might be to not include a minimum criteria for  $C_i$ , and different sets of RWGs are selected from the far-field antennas to ensure variation of RWGs orientations throughout the directional far-field region. The memory scaling for  $U^e$  is plotted in Fig. 8 and scales as  $\mathcal{O}\left(\left(\sum_{i=1}^M \sum_{j=1}^d e_{i,j}\right) \bar{R}\right)$ , where  $\sum_{i=1}^M \sum_{j=1}^d e_{i,j}$  is the sum of all valid directions for all antennas, and  $\bar{R}$  is the average rank.

The fill-in time to generate coupling matrices and the memory requirement for different array sizes is shown in Fig.

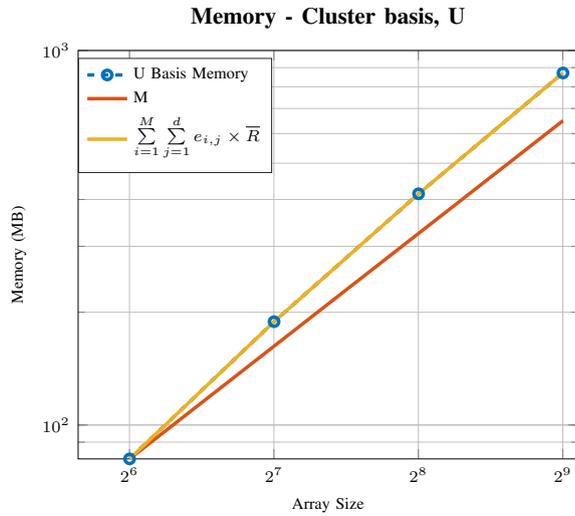


Fig. 8. Memory requirement for directional cluster bases,  $U^e$  for varying array size.  $M$  refers to the number of antennas in the array.

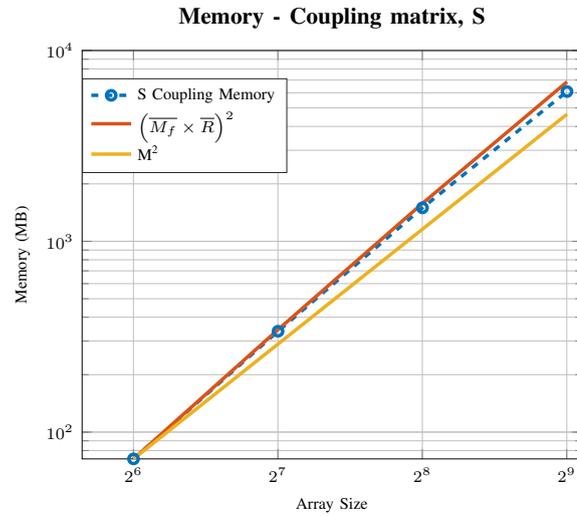


Fig. 10. Memory requirement for coupling matrices,  $S$  for varying array size.  $M$  refers to the number of antennas in the array.

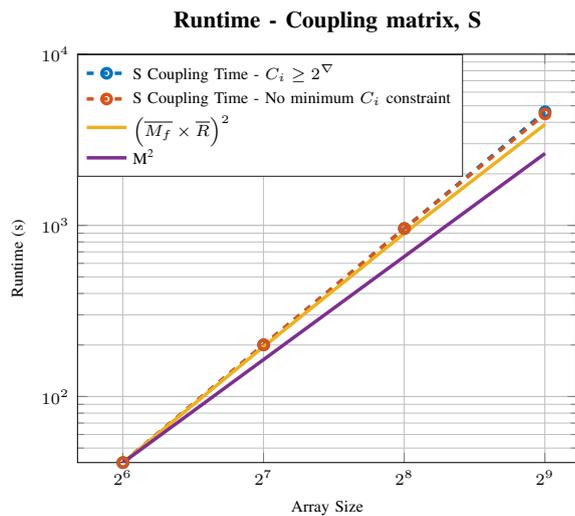


Fig. 9. Computational time to generate coupling matrices,  $S$  for varying array size.  $M$  refers to the number of antennas in the array.

9 and Fig. 10. Both the computational time and memory scale as  $\mathcal{O}(\overline{M}_f \overline{R})^2$ , which is higher than the expected scaling of  $\mathcal{O}(M^2)$  since the rank is still converging.

#### IV. INVERSE FAST MULTIPOLE METHOD

The Inverse Fast Multipole Method (IFMM) is an error-controllable approximate direct solver based on the  $\mathcal{H}^2$ -Matrix introduced in [15] and used as a preconditioner in [18]. A directional version of the single-level IFMM solver tailored to antenna array analysis, is introduced in this section. In this method, an extended sparse matrix is created using the DCA factors and the near and self MoM sub-blocks to be

solved directly. Even if the extended sparse matrix is larger than the original fully-populated impedance matrix, better computational performance can be expected by exploiting the sparse structure of the matrix. To create the extended sparse matrix system, start with the MoM impedance matrix,  $Z$ , expressed as

$$Z = Z^{\text{Near}} + U^{e_1, \dots, e_d} S V^{e_1, \dots, e_d}. \quad (10)$$

$Z^{\text{Near}}$  is a sparse block containing the near and self MoM interactions.  $U^{e_1, \dots, e_d}$  and  $V^{e_1, \dots, e_d}$  are sparse block diagonal matrices, where each diagonal block is the concatenated directional cluster bases of an antenna. The superscript ' $e_d$ ' refers to the total number of directional sectors.  $S$  is a sparse matrix containing the coupling matrices for all admissible interactions in the array. A pictorial representation of the matrix equation in (10) is shown in Figure 11.

Next, the variable  $x$  is defined as the unknown current coefficient, and two auxiliary variables, namely  $y$  and  $z$  are defined as

$$x = I_{\text{MoM}} \quad (11)$$

$$y = V^{e_1, \dots, e_d} x \quad (12)$$

$$z = S y. \quad (13)$$

By substituting equations (12) and (13) into (10), the MoM matrix equation,  $Z I_{\text{MoM}} = V_{\text{MoM}}$ , can be re-written as

$$Z^{\text{Near}} x + U^{e_1, \dots, e_d} z = V_{\text{MoM}}. \quad (14)$$

Using equations (12) to (14), the extended sparse matrix equation,  $E J = B$ , is expressed as follows, where the  $-I$  are negative identity matrices:

$$\begin{bmatrix} Z^{\text{near}} & U^{e_1, \dots, e_d} \\ V^{e_1, \dots, e_d} & -I \end{bmatrix} \begin{bmatrix} x \\ z \\ y \end{bmatrix} = \begin{bmatrix} V_{\text{MoM}} \\ 0 \\ 0 \end{bmatrix}. \quad (15)$$

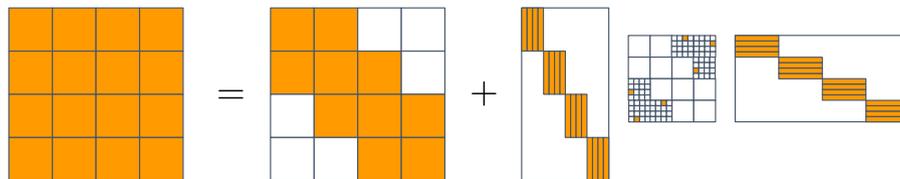


Fig. 11. Pictorial representation of the single-level DCA factorization of the MoM impedance matrix,  $Z$ , as described in (10).

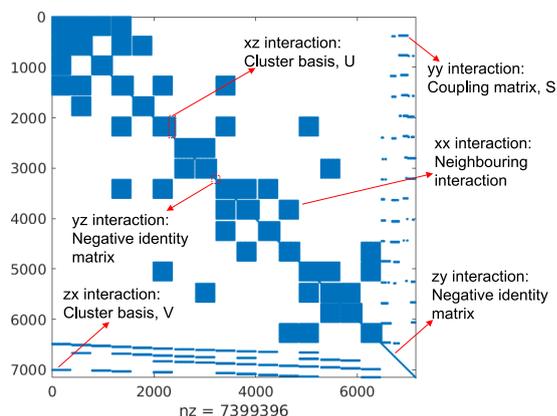


Fig. 12. Sparsity pattern of extended sparse matrix,  $E$ , before Gaussian elimination;  $nz$  refers to the number of matrix non-zero entries.

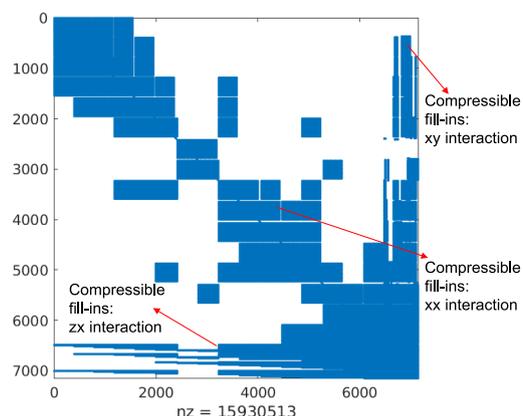


Fig. 13. Sparsity pattern of extended sparse matrix,  $E$ , after Gaussian elimination with fill-ins.

However, unlike in the non-directional  $\mathcal{H}^2$ -Matrix IFMM in [15] and [18], it has been observed that the resulting block diagonals are not well-conditioned during the Gaussian elimination process for the directional case, thus leading to inaccurate results. For this reason, a row permutation is performed so that the  $-I$  blocks are on the diagonal, ensuring well-conditioned diagonal blocks throughout the Gaussian elimination process. Performing the row permutation on (15) leads to

$$\begin{matrix} x: \\ z: \\ y: \end{matrix} \begin{bmatrix} Z_{\text{near}} & U^{e_1, \dots, e_d} \\ & -I & S \\ V^{e_1, \dots, e_d} & & -I \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} V_{\text{MoM}} \\ 0 \\ 0 \end{bmatrix}. \quad (16)$$

However, the downside of the row permutation is that  $E$  is no longer symmetric. The implication of the row permutation is a new set of rules when redirecting fill-ins.

The sparsity pattern of matrix  $E$  resulting from an antenna array of 16 elements is shown in Fig. 12 with the various matrix sub-blocks described in (16). Once (16) is constructed, the next step is to perform Gaussian elimination to solve for  $J$ . However, performing elimination on the sparse matrix  $E$  will lead to undesirable fill-ins as shown in Fig. 13. The fill-ins which occurs during the elimination affects the sparsity of the matrix, hence the complexity of the solver. However, some of the fill-ins which represent interaction between distant antennas can be compressed, removed and re-directed through existing interaction path during the Gaussian elimination process, thus preserving the sparsity to a large extent.

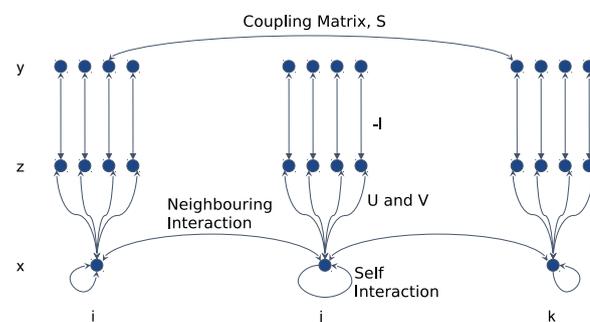


Fig. 14. Partial graph of extended sparse matrix. The blue dots are called 'nodes' and the arrows connecting the nodes are called 'edges'. Antennas  $i$ ,  $j$  and  $k$  each have four  $z$  and  $y$  nodes, implying each antenna in the graph has four directional far-fields.

To perform the elimination and re-direction of compressed fill-ins, the graph of the extended sparse matrix must first be introduced, as shown in Fig. 14. The blue dots are called 'nodes', and each node represent a variable of an antenna. And, the arrows connecting the nodes are called 'edges'. Along the vertical axis in the graph are the  $x$ ,  $y$  and  $z$  variables, and along the horizontal axis are three antennas in the array, denoted as  $i$ ,  $j$  and  $k$ . Antennas  $i$  and  $j$ , and antennas  $j$  and  $k$  are neighbors. And, antennas  $i$  and  $k$  are far-interactions (admissible interaction). The edges in the graph show the interactions between pairs of the  $x$ ,  $y$  and  $z$  nodes. The  $xx$  edges

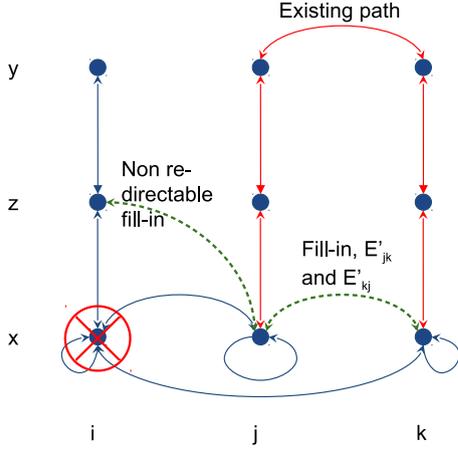


Fig. 15. Graph of fill-ins after eliminating  $x$ -nodes. Only one direction for every antenna is shown here to simplify the diagram.

are self and neighboring interactions. The intra-antenna  $zx$  and  $xz$  edges represents the  $U^{e_1, \dots, e_d}$  and  $V^{e_1, \dots, e_d}$  clusters. The intra-antenna  $zy$  and  $yz$  edges represent the  $-I$  interactions. Finally, inter-antenna  $yy$  edges represent directional coupling between admissible antennas.

#### A. Elimination and Redirection of Fill-ins

1)  $xx$ -interactions: During the Gaussian elimination of the matrix  $E$ , eliminating the  $x$ -node of antenna  $i$  as shown in Fig. 15 will lead to updates of existing non-zero entries in the matrix  $E$  or fill-ins if the block matrix entry was previously empty. Eliminating an  $x$ -node in Fig. 15 means eliminating the self-interaction MoM sub-block of an antenna and block-diagonal entry in Fig. 12. Eliminating the  $x$ -node of antenna  $i$  will lead to the following fill-ins (as depicted in Fig. 15):

- $xx$  compressible interactions between antennas  $j$  and  $k$ , where  $j, k \in N \setminus \{i\}$ ,  $j \notin N \setminus \{k\}$  and  $j \neq k$ .  $N \setminus \{i\}$  means the neighbor antennas of antenna  $i$ . The  $xx$  compressible fill-ins are denoted as  $E'_{jk}$  and  $E'_{kj}$ .
- $xz$  and  $zx$  non-compressible interactions between antennas  $i$  and  $j$ , where  $j \in N \setminus \{i\}$ .

The interactions represented in the fill-ins  $E'_{jk}$  and  $E'_{kj}$  can be re-directed through an existing path in the graph, as shown in red in Fig. 15. Re-direction means updating the matrix blocks along the existing path, such as updating the cluster bases,  $U_j^e$ ,  $V_j^e$ ,  $U_k^{-e}$ , and  $V_k^{-e}$ , and coupling matrices  $S_{jk}$  and  $S_{kj}$ , among other matrices associated with the path. For the redirection of the fill-in  $E'_{jk}$ , the matrix must first be factorized using the Singular Value Decomposition (SVD), such as

$$E'_{jk} \approx U'_{jj} S'_{jk} V'_{kk}, \quad (17)$$

where  $U'_{jj}$  and  $V'_{kk}$  are unitary matrices and  $S'_{jk}$  is a diagonal matrix containing the singular values. Since the SVD operation is expensive (scales as  $\mathcal{O}(N_i^3)$ ) and the rank of  $E'_{jk}$  is expected to be very low, the randomized SVD algorithm [19] is used

in which a partial SVD factorization is computed with a predefined numerical rank,  $\tilde{R}_{\text{ext}}$ . Thus, the sizes of matrices  $U'_{jj}$  and  $V'_{kk}$  are  $N_i \times \tilde{R}_{\text{ext}}$  and  $\tilde{R}_{\text{ext}} \times N_i$ , respectively. And the size of matrix  $S'_{jk}$  is  $\tilde{R}_{\text{ext}} \times \tilde{R}_{\text{ext}}$ . In this implementation,  $\tilde{R}_{\text{ext}}$  is set to 3.

Before proceeding, the matrix  $\Sigma_{u_j^e}$  is defined as the weight of the cluster basis,  $U_j^e$ .  $\Sigma_{u_j^e}$  is obtained as a by-product during the creation of cluster bases in the DCA algorithm by performing ACA-SVD on the matrix  $Z_{\tilde{r}_i^e} \tilde{\sigma}_i^e$ .

The next step is to concatenate the original weighted cluster basis,  $U_j^e \Sigma_{u_j^e}$  with  $U'_{jj} S'_{jk}$ , and perform SVD on the concatenated matrix as

$$[U_j^e \Sigma_{u_j^e} \mid U'_{jj} S'_{jk}] \approx \hat{U}_j^e \hat{\Sigma}_{u_j^e} \Phi_j. \quad (18)$$

It should be noted that randomized SVD cannot be used here since the rank of the matrix is at least the rank of the cluster basis,  $U_j^e$ . To avoid the matrix  $E$  to grow excessively after updates, only singular values of  $\hat{\Sigma}_{u_j^e}$  larger than  $0.1 \times \Sigma_{u_j^e}^{\text{Ori}}(R, R)$  are kept, where  $\Sigma_{u_j^e}^{\text{Ori}}(R, R)$  refers to the smallest weight from the original weight matrix of  $U_j^e$  (that is, before any update is performed). The rank of the updated cluster basis is  $R_{\text{new}} = R_U + R_{\text{ext}}$ , where  $R_U$  is the rank of  $U_j^e$  and  $R_{\text{ext}} \leq \tilde{R}_{\text{ext}}$ . The matrix  $\hat{U}_j^e$  in (18) is the updated cluster basis and  $\hat{\Sigma}_{u_j^e}$  is the updated weight of the cluster basis.  $\hat{U}_j^e$  and  $\hat{\Sigma}_{u_j^e}$  are stored to be used for later fill-ins' eliminations and updates.

$\Phi_j$  in (18) is a unitary matrix and is defined as the concatenation of two smaller matrices  $\phi_j$  and  $\phi'_j$ , such as

$$\Phi_j = [\phi_j \mid \phi'_j]. \quad (19)$$

The number of columns of  $\phi_j$  and  $\phi'_j$  are  $R_U$  and  $R_{\text{ext}}$ , respectively. Using the terms defined in (17), (18) and (19),  $r_j$  and  $r'_j$  are then defined as

$$r_j = \hat{\Sigma}_{u_j^e} \phi_j \Sigma_{u_j^e}^{-1} \quad (20)$$

$$r'_j = \hat{\Sigma}_{u_j^e} \phi'_j S'_{jk}{}^{-1}. \quad (21)$$

Together with  $\hat{U}_j^e$ , terms  $r_j$  and  $r'_j$  are used to define  $U_j^e$  and  $U'_{jj}$  respectively as

$$U_j^e = \hat{U}_j^e r_j \quad (22)$$

$$U'_{jj} = \hat{U}_j^e r'_j. \quad (23)$$

Similarly, the update of  $V_k^{-e}$  is computed through the procedures described from (18) to (23), however using the  $V_k^{-e}$  cluster basis and its weight, and  $V'_{kk}$  in the concatenated matrix in (18) instead.  $V_k^{-e}$  and  $V'_{kk}$  can then be described as in (24) and (25). The terms  $t_k$  and  $t'_k$  can be considered as the equivalents of  $r_j$  and  $r'_j$  for  $V_k^{-e}$ .

$$V_k^{-e} = t_k \hat{V}_k^{-e} \quad (24)$$

$$V'_{kk} = t'_k \hat{V}_k^{-e} \quad (25)$$

The updated coupling matrix,  $\hat{S}_{jk}$ , can then be expressed as

$$\hat{S}_{jk} = r_j S_{jk} t_k + r'_j S'_{jk} t'_k. \quad (26)$$

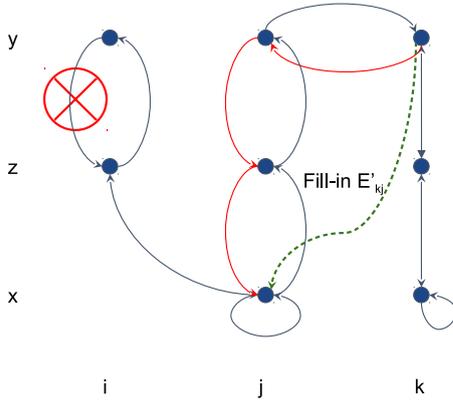


Fig. 16. Graph of fill-ins after eliminating  $-I$  blocks ( $zy$ -interactions). Only one direction for every antenna is shown here to simplify the diagram

Hence, the interaction between antenna  $j$  and  $k$  with updated edges can be written as

$$\begin{aligned}\hat{E}_{jk} &= \hat{U}_{jj}^e \hat{S}_{jk} \hat{V}_{kk}^{-e} \\ &= \hat{U}_{jj}^e \left( r_j S_{jk} t_k + r'_j S'_{jk} t'_k \right) \hat{V}_{kk}^{-e} \\ &= U_j^e S_{jk} V_k^{-e} + U'_{jj} S'_{jk} V'_{kk}.\end{aligned}\quad (27)$$

This step is repeated for the fill-in  $E'_{kj}$  in order to update  $U_k^{-e}$ ,  $V_j^e$  and  $S_{kj}$ . Other  $yy$ -edges connected to antenna  $j$  and  $k$  also need to be updated by multiplying by either  $r_j$ ,  $r_k$ ,  $t_j$  or  $t_k$  as appropriate.

2)  $xy$ -interactions: Eliminating  $-I$  blocks ( $zy$ -interactions) will lead to compressible fill-ins which need to be eliminated and re-directed. In Fig. 16, antenna  $i$  and  $j$  are neighbors. And interactions between antenna  $i$  and  $k$ , and antennas  $j$  and  $k$  are admissible. As seen in Fig. 16, Gaussian elimination of the  $zy$  interaction of antenna  $i$  (diagonal  $-I$  block) will lead to a fill-in between the node  $y$  of antenna  $k$  to the node  $x$  of antenna  $j$ , where  $j \in N\{i\}$ ,  $k \in \text{Adm}\{i, e\}$ .  $\text{Adm}\{i, e\}$  is a set of antennas which belong to the directional admissible region of antenna  $i$  and in the direction  $e$ . If  $k \in \text{Adm}\{j, e\}$ , the fill-in  $E'_{kj}$  is eliminated and re-directed through the red path in Fig. 16 using the procedure described in the previous section.

3)  $zx$ -interactions: In Fig. 17, Antennas  $i$  and  $j$ , and antennas  $j$  and  $k$  are neighbors. Antennas  $i$  and  $k$  are admissible. Node  $x$  and edge  $zy$  have already been eliminated for antenna  $i$ . Eliminating the  $x$  node of antenna  $j$  will lead to a  $zx$  fill-in between antenna  $i$  and  $k$ , denoted as  $E'_{ik}$ . The fill-in can be eliminated and redirected through the red path shown in Fig. 17.

### B. Computational Time and Memory Requirement of Directional IFMM

In this section, the computational time and memory requirement to perform Gaussian elimination on the extended sparse matrix,  $E$ , will be discussed. A bent LPDA antenna similar to Fig. 1, however with 7 teeth instead and bandwidth ranging

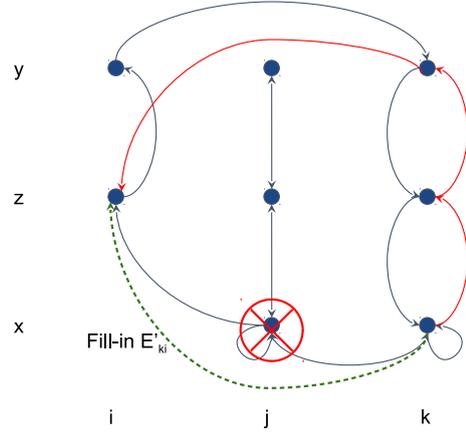


Fig. 17. Graph of fill-ins after eliminating  $zx$ -interactions. Only one direction for every antenna is shown here to simplify the diagram

from 500MHz to 1350MHz, is used. The antenna is meshed for an excitation frequency of 700MHz with 848 basis functions. Table III shows the runtime and memory requirement to perform Gaussian elimination on the extended sparse matrix,  $E$ , for irregular antenna arrays with inter-element spacing of 0.6m and consisting of 32 and 64 antennas. The DCA parameters are selected as discussed in Section III-B, that is,  $\eta_{\text{high}} = 3$  and  $\beta = \frac{\pi}{2}$  rad, and the average number of neighbors for the 32 and 64 element arrays are 11.06 and 10.72, respectively. Three cases of the Gaussian elimination will be discussed, namely ‘no elimination and redirection of fill-ins’, ‘elimination and redirection of all compressible fill-ins’, and ‘elimination and redirection of compressible fill-ins except  $xy$ -edge fill-ins’. Block 1 in Table III refers to the square sub-block of matrix,  $E$ , containing  $xx$  and  $yz$  interactions on the block diagonals. Block 2 in Table III refers to the smaller sub-block of matrix  $E$  containing  $zy$  interactions on the diagonal. The Block 2 initially only consists of  $-I$  blocks on the diagonal, but throughout the Gaussian elimination process, this block will be filled as seen from Fig 12 and Fig 13.

From Table III for the 32 antenna array, the case of ‘no elimination and re-direction of fill-ins’ has the largest memory requirement and longest runtime for Block 1, as expected due to the  $xx$  fill-in interactions, each with the size of  $N_i \times N_i$ . Block 2 has a shorter runtime compared to the cases where fill-ins are re-directed. This is because, for every fill-in redirected, the associate edges are updated as discussed, meaning that the size of the blocks associated with the updated edges are increased. This leads to a larger size of Block 2 for the cases including elimination and re-direction of fill-ins. Irrespective of the case, the majority of entries of Block 2 are filled by the end of the Gaussian elimination process. For case of ‘no elimination and re-direction of fill-ins’ for the antenna array of 64, the workstation ran out of memory before completion of the Gaussian elimination.

The elimination and redirection of  $xy$  fill-ins are expensive in the current implementation, as reported in Table III. During the Gaussian elimination of antenna  $i$ ,  $xy$  fill-ins occurred for

TABLE III  
TIME AND MEMORY REQUIREMENT FOR GAUSSIAN ELIMINATION OF EXTENDED SPARSE MATRIX

Scenario	Block 1 - time (s)	Block 2 - time (s)	Memory (GB)
Array size: 32 elements; Original $E$ memory: 4.21 GB			
No elimination and redirection of fill-ins	395.90	3.04	10.16
Elimination and redirection of all compressible fill-ins	145.15	8.53	5.80
No elimination and redirection of $xy$ fill-ins	138.10	6.92	6.06
Array size: 64 elements; Original $E$ memory: 8.27 GB			
No elimination and redirection of fill-ins	—	—	—
Elimination and redirection of all compressible fill-ins	566.61	108.27	14.21
No elimination and redirection of $xy$ fill-ins	477.65	107.47	15.87

all common directional far-field antennas of antenna  $i$  and  $N\{i\}$ . This results in a large number of fill-ins which need to be redirected individually and hence is expensive. Excluding the redirection of  $xy$  fill-ins from the solver is cheaper in the current version of the solver, however adds a strain of the required memory as shown in Table III.

### C. Accuracy of the DCA-IFMM Solver

The current coefficients of the 32 elements array used in Section IV-B and Table III, is solved using the MoM as shown in Fig. 18. The current coefficient error relative to the MoM when using the DCA-IFMM solver, described in Section IV-B, is also plotted, and a very good accuracy can be observed. The DCA-IFMM solver consists of the elimination and redirection of all compressible fill-ins. The traces in Fig. 18 are plotted for every 20-th point to avoid the graph being overcrowded. The normalized current error in the global current solution is defined as

$$\text{Normalized Current Error} = \frac{\|I_{\text{MoM}} - I_{\text{DCA-IFMM}}\|_2}{\|I_{\text{MoM}}\|_2}, \quad (28)$$

where  $I_{\text{DCA-IFMM}}$  refers to current coefficients obtained using the DCA-IFMM solver. The normalized current error for the DCA-IFMM solver is  $2.18 \times 10^{-4}$ .

## V. INCORPORATING MBFs IN THE DCA-IFMM ALGORITHM

The background of MBF is first reviewed in Section V-A, and then the use of MBFs in the DCA-IFMM algorithm will be discussed.

### A. Macro Basis Function

As mentioned in Section I, MBFs are created by aggregating linear combinations of RWGs on each antenna to create a reduced matrix equation that can be solved directly, such that  $K_i \ll N_i$  where  $K_i$  is the number of MBFs on antenna  $i$ . An MBF solver can be categorized into 3 main steps:

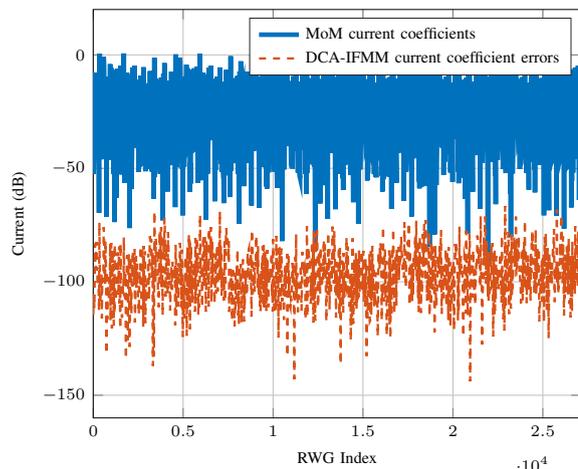


Fig. 18. MoM current coefficients for antenna array of 32 elements, and the current coefficient errors using the DCA-IFMM solver. The current coefficient errors being shown, are defined as  $I_{\text{MoM}} - I_{\text{DCA-IFMM}}$ .

1) *Generation of MBFs Definitions*: Primary and Secondary MBFs [5] can be generated to represent self and mutual interactions, respectively. A primary MBF,  $I_{\text{prim}}$ , is the current coefficient obtained by an excited antenna,  $i$ , in isolation as

$$Z_{ii}I_{\text{prim}} = V_i^{\text{MoM}}. \quad (29)$$

The primary MBF from antenna  $i$  is then allowed to radiate onto antennas  $j$ , where  $j$  are the antennas within a pre-defined radius around  $i$ . The secondary MBFs,  $I_{\text{sec}}$ , are the induced current coefficients on antennas  $j$  due to  $I_{\text{prim}}$ , such as

$$Z_{ji}I_{\text{sec}} = -Z_{ii}I_{\text{prim}} \quad (30)$$

If more accuracy is required, higher-order MBFs (e.g. tertiary MBFs [20]) can be generated by allowing lower-order neighboring MBFs to radiate onto the antenna. An SVD operation is then performed on the group of MBFs belonging to an antenna to obtain the final set of MBFs. This step is important to ensure that a set of orthogonal MBFs is obtained, but also to trim out MBFs with insignificant contribution, that is, with singular values below a pre-defined threshold. For irregular arrays, the primary MBF is the same for all antennas and can be computed only once. However, secondary and higher-order MBFs need to be computed for each antenna.

2) *Computation of Reaction Terms*: Once MBFs definitions are obtained for all the antennas, the entries of the reduced impedance matrix can be computed. The reaction terms are obtained by pre and post-multiplying MoM sub-blocks with the corresponding MBF groups. For instance, the reduced block entry for the interaction between antenna  $i$  and  $j$ ,  $Z_{ij}^{\text{Red}}$ , is obtained as

$$Z_{ij}^{\text{Red}} = \langle \text{MBF}_i^H, Z_{ij} \text{MBF}_j \rangle, \quad (31)$$

where  $Z_{ij}$  is the MoM sub-block for the interaction between antenna  $i$  and  $j$ .  $\text{MBF}_i$  and  $\text{MBF}_j$  are the MBF definitions for antenna  $i$  and  $j$  respectively. On the right hand side on the reduced matrix equation, the reduced excitation vector,  $V^{\text{Red}}$

is obtained by pre-multiplying the excitation vector with the MBF definitions, for instance,  $V_i^{\text{Red}} = \langle \text{MBF}_i^H, V_i^{\text{MoM}} \rangle$ .

3) *Solving the Reduced Matrix Equation*: Finally, the reduced matrix system is obtained as  $Z^{\text{Red}} I_{\text{MBF}} = V^{\text{Red}}$ . In the final step, the MBF coefficients  $I_{\text{MBF}}$  is solved for. The MoM current coefficients can then easily be obtained by pre-multiplying the MBF coefficients with the MBF definitions.

### B. Combining MBF with the DCA-IFMM Algorithm

The size of Block 1 from Table III is  $(N + (2 \times R_{\text{tot}})) \times (N + (2 \times R_{\text{tot}}))$  and that of Block 2 is  $(R_{\text{tot}} \times R_{\text{tot}})$  where  $R_{\text{tot}}$  is the sum of ranks for all antennas and directions, and is defined as  $R_{\text{tot}} = \sum_{i=1}^M \sum_{j=1}^d R_{ie_j}$ . Since the size of Block 1 is much larger than that of Block 2, the former can be a limiting factor in terms of the memory requirement to solve for large arrays. The size of Block 1 can drastically be reduced by employing MBFs to produce reduced MoM sub-blocks for near interactions,  $Z_R^{\text{Near}}$ , using (31) and reduced directional cluster bases,  $U_R^{e_1, \dots, e_d}$  and  $V_R^{e_1, \dots, e_d}$ , such as

$$U_i^{\text{Red}} = \text{MBF}_i^H U_i^e \quad (32)$$

$$V_i^{-e \text{Red}} = V_i^{-e} \text{MBF}_i. \quad (33)$$

The size of the reduced Block 1 is now  $(K_{\text{tot}} + (2 \times R_{\text{tot}})) \times (K_{\text{tot}} + (2 \times R_{\text{tot}}))$  where  $K_{\text{tot}}$  is the total number of MBFs for all antennas, and defined as  $K_{\text{tot}} = \sum_{i=1}^M K_i \ll N$ . The reduced extended sparse matrix,  $E^{\text{Red}}$ , can now be solved to obtain the MBF coefficients,  $I_{\text{MBF}}$ . Solving for  $E^{\text{Red}}$  instead of the original extended sparse matrix allows the possibility of solving for larger arrays. However, the Block 2 in  $E^{\text{Red}}$  now has a comparable size to the reduced Block 1. And, since Block 2 is mostly filled at the end of the Gaussian elimination, it might be more efficient to instead create and solve the fully populated reduced matrix,  $Z^{\text{Red}}$ , as described in Section V-A. Incorporating MBFs in DCA-IFMM can potentially be an attractive approach if a multi-level algorithm for DCA and IFMM is used instead. In this case, Block  $M+1$  (last diagonal block) will be the size of the sum of ranks of clusters at the highest level, which can be small compared to the reduced Block 1. The extension to a non-directional multilevel IFMM scheme is presented in [15], [18]. However, reformulation for the DCA remains a challenging, open question.

## VI. CONCLUSION

In this paper, a direct solver is implemented to analyze large antenna arrays. Firstly, the DCA algorithm is used to compress the interactions of far antennas. It has been shown that using directional far-field sectors instead of an omnidirectional far-field when computing DCA factors is more efficient, since the numerical rank is limited. Secondly, the DCA factors and MoM near-interaction sub-blocks are used to construct an extended sparse matrix to be solved using the IFMM algorithm. In the IFMM, compressible fill-ins are discarded and redirected during the Gaussian elimination process. Finally, the idea of using MBFs in a DCA-IFMM solver is discussed. Using the single-level version of IFMM to solve for the MBF coefficients

will be less efficient than using the fully populated reduced MBF matrix. The multilevel version of DCA-IFMM with MBFs must be explored.

## REFERENCES

- [1] *SKA website*. [Online]. Available: <https://www.skatelescope.org/>
- [2] R. Coifman, V. Rokhlin, and S. Wandzura, "The fast multipole method for the wave equation: A pedestrian prescription," *IEEE Antennas and Propagation Magazine*, vol. 35, no. 3, pp. 7–12, 1993.
- [3] E. Bleszynski, M. Bleszynski, and T. Jaroszewicz, "Aim: Adaptive integral method for solving large-scale electromagnetic scattering and radiation problems," *Radio Science*, vol. 31, no. 5, pp. 1225–1251, 1996.
- [4] C. Craeye, J. Laviada, R. Maaskant, and R. Mittra, "Macro basis function framework for solving Maxwell's equations in surface integral equation form," *The FERMAT Journal*, vol. 3, pp. 1–16, 2014.
- [5] V. Prakash and R. Mittra, "Characteristic basis function method: A new technique for efficient solution of method of moments matrix equations," *Microwave and Optical Technology Letters*, vol. 36, no. 2, pp. 95–100, 2003.
- [6] L. Matekovits, V. A. Laza, and G. Vecchi, "Analysis of large complex structures with the synthetic-functions approach," *IEEE Transactions on Antennas and Propagation*, vol. 55, no. 9, pp. 2509–2521, 2007.
- [7] D. J. Bakers, S. J. van Eijndhoven, and A. G. Tijhuis, "An eigencurrent approach for the analysis of finite antenna arrays," *IEEE Transactions on Antennas and Propagation*, vol. 57, no. 12, pp. 3772–3782, 2009.
- [8] R. Maaskant, R. Mittra, and A. Tijhuis, "Fast analysis of large antenna arrays using the characteristic basis function method and the adaptive cross approximation algorithm," *IEEE Transactions on Antennas and Propagation*, vol. 56, no. 11, pp. 3440–3451, 2008.
- [9] K. Zhao, M. N. Vouvakis, and J.-F. Lee, "The adaptive cross approximation algorithm for accelerated method of moments computations of emc problems," *IEEE transactions on electromagnetic compatibility*, vol. 47, no. 4, pp. 763–773, 2005.
- [10] M. Bebandorf, C. Kuske, and R. Venn, "Wideband nested cross approximation for Helmholtz problems," *Numerische Mathematik*, vol. 130, no. 1, pp. 1–34, 2015.
- [11] K. Sewraj and M. M. Botha, "Computation of MBF reaction matrices for antenna array analysis, with a directional method," in *Proceedings of the International Conference on Electromagnetics in Advanced Applications (ICEAA '19)*, Sept 2019, pp. 385–389, Granada, Spain.
- [12] —, "Directional method to compute reduced matrix system in mbf solvers," in *2020 14th European Conference on Antennas and Propagation (EuCAP)*. IEEE, 2020, pp. 1–5.
- [13] H. Bui-Van, J. Abraham, M. Arts, Q. Gueuning, C. Raucy, D. Gonzalez-Ovejero, E. de Lera Acedo, and C. Craeye, "Fast and accurate simulation technique for large irregular arrays," *IEEE Transactions on Antennas and Propagation*, vol. 66, no. 4, pp. 1805–1817, 2018.
- [14] C. Craeye, "A fast impedance and pattern computation scheme for finite antenna arrays," *IEEE transactions on antennas and propagation*, vol. 54, no. 10, pp. 3030–3034, 2006.
- [15] S. Ambikasaran and E. Darve, "The inverse fast multipole method," *arXiv preprint arXiv:1407.1572*, 2014.
- [16] S. Rao, D. Wilton, and A. Glisson, "Electromagnetic scattering by surfaces of arbitrary shape," *IEEE Transactions on antennas and propagation*, vol. 30, no. 3, pp. 409–418, 1982.
- [17] *Antenna Magus website*. [Online]. Available: <http://www.antennamagus.com>
- [18] P. Coulier, H. Pouransari, and E. Darve, "The inverse fast multipole method: using a fast approximate direct solver as a preconditioner for dense linear systems," *SIAM Journal on Scientific Computing*, vol. 39, no. 3, pp. A761–A796, 2017.
- [19] N. Halko, P.-G. Martinsson, and J. A. Tropp, "Finding structure with randomness: Probabilistic algorithms for constructing approximate matrix decompositions," *SIAM review*, vol. 53, no. 2, pp. 217–288, 2011.
- [20] S. G. Hay, J. D. O'Sullivan, and R. Mittra, "Connected patch array analysis using the characteristic basis function method," *IEEE Transactions on Antennas and Propagation*, vol. 59, no. 6, pp. 1828–1837, 2011.

## Appendix B

### Journal paper — *ACA* acceleration for the IRB-DGFM [2]

M. Chose, K. Sewraj and M. M. Botha, “Iterative Radius-Based Domain Green’s Function Method with *ACA* for Antenna Array Analysis,” 2021, submitted for publication.

# Iterative Radius-Based Domain Green's Function Method with ACA for Antenna Array Analysis

Matthews Chose, Keshav Sewraj and Matthys M. Botha, *Member, IEEE*

**Abstract**—The method of moments (MoM) is well suited to the electromagnetic analysis of antenna arrays, but the solution cost grows rapidly with array size. This paper is concerned with efficient and reliable MoM analysis of large antenna arrays consisting of identical disjoint elements in arbitrary layouts. The domain Green's function method (DGFM) is a form of domain decomposition method for array analysis requiring only the solution of local problems associated with every array element, to obtain an approximate global solution. Its main drawbacks are uncontrolled errors and poor runtime scaling. An extended version is presented, denoted the iterative radius-based DGFM (IRB-DGFM). Accuracy is improved (by introducing local MoM domains) and solution error control is introduced (by expanding it into an iterative scheme), such that the true MoM solution may be achieved to user-specified tolerance. A hybrid single-level adaptive cross approximation (ACA) compression scheme is devised to address computational efficiency. Results demonstrate very good memory efficiency and rapid convergence of the ACA IRB-DGFM. The convergence rate is dependent on the local domain size and not the global array size. The method is ideally suited to parallelization.

**Index Terms**—Domain decomposition, electric field integral equation (EFIE), fast solver, low-rank factorization.

## I. INTRODUCTION

The method of moments is well suited to the full-wave electromagnetic analysis of antenna arrays. For perfect electrically conducting (PEC) antennas an electric field integral equation (EFIE) formulation with Rao-Wilton-Glisson (RWG) basis functions is the typical approach [1]. This does yield fully populated matrices with the solution cost growing rapidly with array size. This paper is concerned with efficient and reliable EFIE RWG MoM analysis of large antenna arrays consisting of identical disjoint PEC elements.

Existing solutions fall into a few categories. Fast integral equation (IE) methods based on compression of the global MoM matrix in combination with Krylov subspace iterative methods, such as the conjugate gradient-fast Fourier transform (CG-FFT) method [2], and low-rank factorization methods such as the multilevel fast multipole method (MLFMM) [3], [4] and adaptive cross approximation (ACA) [5]–[8]. These can yield optimal cost complexity of  $\mathcal{O}(N_{\text{tot}} \log N_{\text{tot}})$ , where  $N_{\text{tot}}$  is the total degrees of freedom (DoFs). The drawbacks are that sophisticated preconditioners are required and that the number of iterations can be large and unpredictable.

This work was supported in part by the South African Radio Astronomy Observatory (SARAO) and in part by the National Research Foundation under Grant 75322 and Grant 106033. (Corresponding author: Matthys M. Botha.)

The authors are with the Department of Electrical and Electronic Engineering, Stellenbosch University, Stellenbosch 7600, South Africa (e-mail: 22468994@sun.ac.za, 20635958@sun.ac.za, mmbbotha@sun.ac.za).

Domain decomposition methods (DDMs) is another important category. The full array domain is divided into sub-problems typically consisting of individual elements or groups of elements. The problem is then solved on two levels: the local solutions and the global coupling solution. Finite element method DDMs [9], [10] and MoM DDMs [11] still involve solutions which could require many iterations. Furthermore, the global solutions' cost scaling is still a function of the total number of degrees of freedom, though with a more favourable scaling constant. Another category is macro basis function (MBF) methods. Physics-based linear combinations of low-level basis functions (typically with supports constituting individual array elements) are formed and these are used as a reduced basis set, yielding a significantly smaller global system [12]–[16]. Drawbacks are that accuracy depends upon the validity of the MBFs and that very efficient construction of the reduced system is required to realize the full potential of this approach [17]–[20].

Finally, consider the domain Green's function method (DGFM) [21]–[23] which is specifically formulated for array antennas with identical disjoint elements. As a DDM it is distinctive in that the global solution is approximated by only ever solving local problems associated with every array element. This is very attractive, as the local problems are much smaller than the global problem. However, runtime scales as  $\mathcal{O}(N_{\text{tot}}^2)$  and the solution error is problem dependent and uncontrolled. Section III will elaborate further on the shortcomings of the standard DGFM. However, its distinctive benefit prompted the present work, which extends the standard DGFM to address its shortcomings while retaining its aspect of obtaining the global solution through solving only local problems. Developmental aspects of this work presented at conferences [24], [25] has lead up to the formulation and results presented here.

Section II presents the baseline MoM formulation. Section III presents the standard DGFM formulation. Section IV then presents extensions to the standard DGFM to improve its accuracy (by introducing local MoM domains) and to allow for control of its approximation error (by expanding it into an iterative scheme). This improved formulation is called the iterative radius-based DGFM (IRB-DGFM). The computational cost of a direct implementation of the IRB-DGFM still scales the same as the standard DGFM. Therefore in Section V a hybrid, single-level ACA compression scheme which is tailor-made to the IRB-DGFM is presented, to vastly improve the scaling. Finally, Sections VI and VII present verification results and the conclusion.

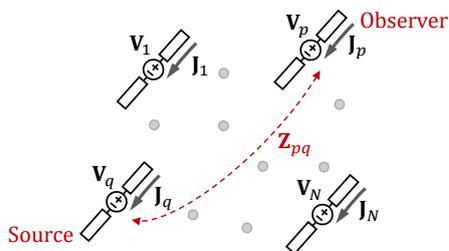


Fig. 1. Antenna array geometry of interest, with  $N$  identical disjoint elements. Each element has a unique excitation and current distribution.

## II. MOM FORMULATION AND MATRIX PARTITIONING

Formulating the standard EFIE-based MoM to solve for the electric surface current density on PEC surface structures in free space, using RWG basis functions on conforming triangle meshes and Galerkin testing, results in the following system of linear equations [1], [26]:

$$\mathbf{Z}\mathbf{J} = \mathbf{V}. \quad (1)$$

In (1)  $\mathbf{Z}$  denotes the impedance matrix,  $\mathbf{J}$  is the column vector of basis function coefficients (i.e. DoFs: degrees of freedom), and  $\mathbf{V}$  is the excitation vector which results from testing the known incident electric field.

Fig. 1 illustrates the problem of interest, namely a structure representing an antenna array consisting of  $N$  identical, disjoint antenna elements with arbitrary layout. Each element is excited with a unique voltage source value at an elemental port location (implemented as a voltage gap source [26], as is standard practice). The elemental port locations are identical across the whole array. The elemental meshes and their associated sets of basis functions are required to be identical copies, with  $M$  denoting the number of DoFs per element. The DoFs are required to be consecutively numbered per element, such that each elemental set of DoFs forms a segment of  $\mathbf{J}$ . In this case (1) can be expressed in partitioned format as

$$\begin{bmatrix} \mathbf{Z}_{11} & \cdots & \mathbf{Z}_{1q} & \cdots & \mathbf{Z}_{1N} \\ \vdots & & \vdots & & \vdots \\ \mathbf{Z}_{p1} & \cdots & \mathbf{Z}_{pq} & \cdots & \mathbf{Z}_{pN} \\ \vdots & & \vdots & & \vdots \\ \mathbf{Z}_{N1} & \cdots & \mathbf{Z}_{Nq} & \cdots & \mathbf{Z}_{NN} \end{bmatrix} \begin{bmatrix} \mathbf{J}_1 \\ \vdots \\ \mathbf{J}_p \\ \vdots \\ \mathbf{J}_N \end{bmatrix} = \begin{bmatrix} \mathbf{V}_1 \\ \vdots \\ \mathbf{V}_p \\ \vdots \\ \mathbf{V}_N \end{bmatrix} \quad (2)$$

with  $\mathbf{Z}_{pq}$ ,  $p, q \in \{1, \dots, N\}$ , denoting the  $M \times M$  sub-matrix containing the coupling coefficients resulting from testing the effect of element  $q$ 's source basis functions on element  $p$ . The  $M$ -dimensional sub-vectors  $\mathbf{J}_p$  and  $\mathbf{V}_p$  are the DoFs and excitation coefficients for element  $p$ , respectively.

With increasing  $N$  the computational cost of solving the system (2) becomes very expensive. The rest of the paper is dedicated to the fast solution of (2).

## III. STANDARD DGFM FORMULATION

The standard DGFM provides an approximate but memory-inexpensive solution of (2). Each  $\mathbf{J}_p$  is independently solved by

making the assumption that from its perspective, any current  $\mathbf{J}_q$ ,  $q \in \{1, \dots, N, q \neq p\}$ , can be approximated by a scaled version of  $\mathbf{J}_p$ , as

$$\mathbf{J}_q \approx \alpha_{qp} \mathbf{J}_p \quad \text{with} \quad \alpha_{qp} = \frac{c_q}{c_p} \text{diag}(1), \quad (3)$$

where  $c_p$  and  $c_q$  denote the excitation voltages of elements  $p$  and  $q$ , respectively. The elemental current  $\mathbf{J}_p$  is solved by substituting (3) into the  $p$ -th row of (2), as

$$\mathbf{J}_p = \left( \mathbf{z}_{pp} + \sum_{q=1, q \neq p}^N \mathbf{z}_{pq} \alpha_{qp} \right)^{-1} \mathbf{V}_p, \quad (4)$$

for  $p = 1, \dots, N$ , to yield the full array solution.

The assumption (3) is exact in case of infinite periodic arrays with sufficiently regular excitation, while it becomes increasingly accurate for finite arrays as the inter-element spacings are increased (i.e. sparse arrays). Generally, the smoother the variation of elemental currents across the array, the better. Thus it will not account well for array edge effects, especially in the case of dense arrays; also not for the effect of irregular inter-element spacings, when elemental current distributions can vary strongly between neighbors due to variations in local neighborhoods. Furthermore, the solution error in (4) is problem dependent and thus uncontrolled. The new formulation presented next, which builds on the standard DGFM of (4), aims to address these shortcomings.

## IV. IRB-DGFM FORMULATION

This section describes the two extensions that the IRB-DGFM adds to the standard DGFM formulation. The first is to incorporate local coupling via full MoM analysis. The second is to expand the formulation into an iterative scheme, such that a given solution is used to improve the  $\alpha$  scaling coefficients (3). The latter extension necessitates a generalisation to DoF-based scaling coefficients.

### A. Radius-Based Formulation

The influence of one element on another diminishes as the separation distance increases, as quantified by the Green's function. To incorporate the strong local coupling effects more accurately, clusters of strongly coupled elements are formed, one associated with each array element as shown in Fig. 2. The clusters are defined by a user-specified radius  $\ell$ , which is a global parameter. The cluster for element  $p$  is  $p$  itself together with all neighboring elements with centroids located at a distance of  $\leq \ell$  from its centroid. It can be expressed in terms of a normalized quantity  $r$ :

$$r = \frac{\ell}{\min|\mathbf{r}_i - \mathbf{r}_j|} \quad i, j \in \{1, \dots, N\} \text{ and } i \neq j, \quad (5)$$

where  $\mathbf{r}_i$  and  $\mathbf{r}_j$  denote elemental centroid position vectors.

As with the standard DGFM a local MoM problem is solved per element, except that it is now solved per elemental cluster. Denote the  $p$ -th elemental cluster domain by  $S_p$ , which is the union of the set of elements with indices  $\{p, q_1, \dots, q_Q\}$  ( $Q$  is the number of neighbors). The MoM is rigorously formulated

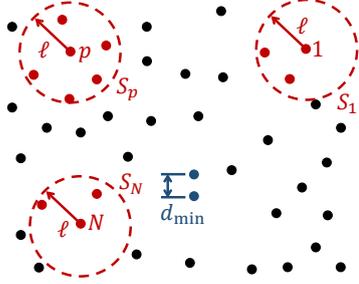


Fig. 2. Illustration of local domains for the radius-based solution scheme. Each dot represents an antenna element. Local domains 1,  $p$  and  $N$  are highlighted. The minimum distance between two elemental centroids is also highlighted as  $d_{\min} = \min|\mathbf{r}_i - \mathbf{r}_j|$  (see (5) for reference).

on  $S_p$ , with the effect of external elements included via the  $\alpha$  scaling coefficients of the standard DGFM, defined in (3). This yields the radius-based DGFM solution on local domain  $S_p$ :

$$\begin{Bmatrix} \mathbf{J}_p \\ \mathbf{J}_{q_1} \\ \vdots \\ \mathbf{J}_{q_Q} \end{Bmatrix} = \begin{bmatrix} \mathbf{Z}_{pp}^{\text{act}} & \mathbf{Z}_{pq_1} & \cdots & \mathbf{Z}_{pq_Q} \\ \mathbf{Z}_{q_1p}^{\text{act}} & \mathbf{Z}_{q_1q_1} & \cdots & \mathbf{Z}_{q_1q_Q} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{Z}_{q_Qp}^{\text{act}} & \mathbf{Z}_{q_Qq_1} & \cdots & \mathbf{Z}_{q_Qq_Q} \end{bmatrix}^{-1} \begin{Bmatrix} \mathbf{V}_p \\ \mathbf{V}_{q_1} \\ \vdots \\ \mathbf{V}_{q_Q} \end{Bmatrix} \quad (6)$$

where the active impedance matrices are defined as

$$\mathbf{Z}_{mp}^{\text{act}} = \mathbf{Z}_{mp} + \sum_{i \in P_{\text{ext}}} \mathbf{Z}_{mi} \alpha_{ip} \quad \text{with } m \in \{p, q_1, \dots, q_Q\}. \quad (7)$$

In (7) the set of indices  $p_{\text{ext}} = \{t_1, \dots, t_T\}$  denote the external elements to cluster  $p$ , such that  $N = 1 + Q + T$ . The fact that the active impedance matrices are all located in the first column of the matrix in (6) is due to all external elemental currents being represented as scaled versions of  $\mathbf{J}_p$ , which is regarded as the most accurately solved elemental current in the local problem.

The radius-based DGFM solution for the whole array is obtained by solving (6) for all  $p = 1, \dots, N$ , retaining only  $\mathbf{J}_p$  from each local solution. With  $\ell = 0$  this formulation reverts back to the standard DGFM. As  $\ell$  is increased the accuracy will improve, up until the limiting case when every local problem encompasses the whole array at which point the full MoM is retrieved. Clearly computational cost will increase as  $\ell$  is increased. This provides motivation for the iterative scheme presented next, which allows for retrieving the MoM solution to specified error tolerance, while using an  $\ell$  value which is small and not proportional to the global array diameter.

### B. Iteration Scheme with DoF-Based Scaling

For  $r$  values of  $\mathcal{O}(1)$  in (5), the radius-based DGFM solution will still exhibit potentially significant errors, which are moreover still uncontrolled. To address this, an iterative updating scheme for the  $\alpha$  scaling coefficients is introduced. As will be shown in the results section, this iterative scheme is capable of converging to the true MoM solution. It has an error

tolerance termination condition exactly as used with standard iterative matrix equation solvers.

The initial solution with iteration index  $k = 0$ , is obtained according to (3) which uses the standard DGFM's excitation voltage based  $\alpha$  coefficients. For iteration  $k > 0$  DoF-based scaling coefficients as developed in [23], [27] are used. These are defined as follows:

$$\alpha_{qp}^{(k)} = \text{diag} \left( \frac{\mathbf{J}_q^{(k-1)}(1)}{\mathbf{J}_p^{(k-1)}(1)}, \frac{\mathbf{J}_q^{(k-1)}(2)}{\mathbf{J}_p^{(k-1)}(2)}, \dots, \frac{\mathbf{J}_q^{(k-1)}(M)}{\mathbf{J}_p^{(k-1)}(M)} \right), \quad (8)$$

where the entries from the previous iteration's solution vectors for elements  $p$  and  $q$ , denoted by  $\mathbf{J}_p^{(k-1)}$  and  $\mathbf{J}_q^{(k-1)}$  are employed on the right hand side. Thus, these  $\alpha$  coefficients can account for variations in both the amplitude and shape of elemental current distributions across the array.

The normalized solution residual error at iteration  $k$  is determined in the standard way as

$$\epsilon^{(k)} = \frac{\|\mathbf{V} - \mathbf{Z}\mathbf{J}^{(k)}\|_F}{\|\mathbf{V}\|_F}, \quad (9)$$

where  $\|\cdot\|_F$  denotes the Frobenius norm. The complete IRB-DGFM formulation is as follows:

- 1) Choose the cluster radius  $\ell$ .
- 2) From  $\ell$  determine the cluster domains  $S_p$ ,  $p = 1, \dots, N$ .
- 3) For the initial solution  $k = 0$ , solve (6) using the  $\alpha$  coefficients defined in (3), for  $p = 1, \dots, N$ . From each local solution only retain  $\mathbf{J}_p$ , to form the overall array solution  $\mathbf{J}^{(0)}$ .
- 4) Calculate the normalized solution error  $\epsilon^{(k)}$  according to (9).
- 5) Test for convergence, defined as  $\epsilon^{(k)} \leq \mu$  where  $\mu$  is a user-specified error tolerance threshold. If convergence has been achieved then terminate with  $\mathbf{J}^{(k)}$  as the final solution; otherwise continue.
- 6) Set  $k := k + 1$ .
- 7) Solve (6) using the  $\alpha$  coefficients defined in (8). From each local solution only retain  $\mathbf{J}_p$ , to form the overall array solution  $\mathbf{J}^{(k)}$ .
- 8) Go to Step 4.

As noted already, a larger cluster radius should yield a more accurate solution and thus a greater reduction in the solution error from one iteration to the next. Therefore, it is expected that increasing  $\ell$  will increase the convergence rate of the iterative solution; this would of course be at the price of a higher computational cost per iteration. The next section deals with the acceleration of the solver, through low-rank factorization tailored to the IRB-DGFM formulation.

## V. HYBRIDIZED SINGLE-LEVEL ACA ACCELERATION OF THE IRB-DGFM

An efficient, single-level ACA scheme tailored to the IRB-DGFM is presented. The ACA scheme hybridizes the standard version with a translation-based version. The latter allows for vastly reduced memory requirements and reductions in arithmetic operation counts, provided it is exclusively applied to far interactions. Computational cost is discussed at the end of this section.

### A. Description of the Acceleration Scheme

Given the nature of the Green's function, it is well known that the impedance matrix blocks in (2) can be categorized according to rank. With reference to Fig. 3, these categories are as follows:

- 1) Full-rank blocks (self interactions);
- 2) Intermediate-rank blocks (near interactions where the elemental bounding boxes are  $\leq R_\tau$  apart);
- 3) Low-rank blocks (far interactions where the elemental bounding boxes are  $> R_\tau$  apart).

The blocks from each category are stored in a specific manner, aimed at minimizing storage requirements together with minimizing the runtime cost of constructing the clusters' system matrices of (6).

The full-rank (diagonal) blocks are stored directly and since they are all identical, only  $\mathbf{Z}_{11}$  is stored, at negligible cost.

The intermediate-rank blocks are compressed with standard ACA:

$$\mathbf{Z}_{pq}^{M \times M} \approx \mathbf{U}_{pq}^{M \times R_{pq}} \mathbf{V}_{pq}^{R_{pq} \times M} \quad (10)$$

where  $\mathbf{U}_{pq}$  and  $\mathbf{V}_{pq}$  are the standard ACA factors determined according to the algorithm as presented in [7]. These are uniquely determined and stored for each block  $\mathbf{Z}_{pq}$ . The compression rank  $R_{pq}$  is determined by the factorization algorithm and is dependent upon  $\mathbf{Z}_{pq}$  and the user-specified relative compression error tolerance  $\varepsilon_{ACA}$ . As long as the block is not full rank, the result is  $R_{pq} < M$  and compression is achieved.

Now consider the low-rank blocks, which are the overwhelming majority for large arrays. Should standard ACA be used for these, then  $\mathbf{U}$  and  $\mathbf{V}$  factors will have to be stored for each block and furthermore, when forming the active impedance matrices in (7), each term in the summation will have to be fully reconstituted before being added to the total. The latter would mean that the runtime scaling of setting up the local problems will be the same as for setting up the conventional MoM matrix, namely  $\mathcal{O}((NM)^2)$ . Both storage and runtime can be reduced, by rather using a translation-based version of the ACA (denoted TACA henceforth) for all the low-rank blocks.

The TACA scheme is a single-level version of the nested cross approximation scheme presented in [28]. The low-rank blocks are compressed with TACA as follows:

$$\mathbf{Z}_{pq}^{M \times M} \approx \mathbf{D}_p^{M \times R_p} \mathbf{T}_{pq}^{R_p \times R_q} \mathbf{A}_q^{R_q \times M}, \quad (11)$$

where  $\mathbf{D}_p$  and  $\mathbf{A}_q$  are disaggregation and aggregation cluster basis factors which are solely related to the testing and source basis functions, respectively;  $\mathbf{T}_{pq}$  is the translation factor, representing the interaction between two cluster bases. The cluster ranks  $R_p$  and  $R_q$  follow from the TACA factorization procedure, of which the details are presented in the next section. In the case of the Galerkin EFIE MoM formulation used here it further holds that

$$\mathbf{D}_p = \mathbf{A}_p^T \quad (12)$$

$$\mathbf{T}_{pq} = \mathbf{T}_{qp}^T. \quad (13)$$

Therefore only  $N$  disaggregation factors have to be stored together with  $\sim 0.5N^2$  translation factors. In contrast, standard ACA would require storage of  $\sim 0.5N^2$  instances of  $\mathbf{U}$  and  $\mathbf{V}$  factors. If TACA is only applied to far interaction blocks then it holds that  $R_p \approx R_q \approx R_{pq}$ , thus the storage reduction with TACA instead of standard ACA is quite dramatic. However, inclusion of intermediate-rank blocks for TACA compression would not be beneficial, as it will force the rank of the aggregation and disaggregation cluster basis factors to be higher, to cater for the intermediate-rank interactions. Since the TACA aggregation and disaggregation factors for particular elements are used for all interactions involving those elements, the size of all the translation factors representing coupling blocks involving elements with some intermediate-rank interactions, will have to be increased. For this reason standard ACA is used for intermediate-rank interactions and TACA is only used for the far (low-rank) interactions, where ranks should ideally be asymptotically low.

The above discussion speaks to the importance of the user-defined parameter  $R_\tau$ , to ensuring compression efficiency. Let  $b_p = b_w = b$  denote the bounding boxes of individual elements. The minimum requirement on the distance between elemental bounding boxes is set by the so-called  $\eta$ -admissibility condition [5], [28], which states that

$$R_\tau \geq \frac{1}{\eta} \max \{ \text{diam}(b_p), \text{diam}(b_w) \} \geq \text{diam}(b), \quad (14)$$

with  $\eta = 1$  yielding the final result, which is a commonly accepted value. This guarantees that the TACA factorization is valid. However, with reference to the above discussion on the efficiency of the hybridized single-level ACA acceleration scheme, substantially larger values for  $R_\tau$  should be chosen. A multiple of the standard antenna far-field distance [29], i.e.

$$\text{Antenna far-field distance} \equiv \frac{2[\text{diam}(b)]^2}{\lambda}, \quad (15)$$

is a good guideline. This is experimentally verified in Section VI-A.

Finally, these factorizations are used to construct the active impedance matrices by substituting (10) and (11) into (7), yielding

$$\mathbf{Z}_{mp}^{\text{act}} = \mathbf{Z}_{mp} + \underbrace{\sum_{i \in B} \mathbf{U}_{mi} \mathbf{V}_{mi} \alpha_{ip}}_{\text{Standard ACA terms}} + \underbrace{\mathbf{D}_m \left( \sum_{j \in C} \mathbf{T}_{mj} \mathbf{D}_j^T \alpha_{jp} \right)}_{\text{Translation-based ACA terms}} \quad (16)$$

with

$$B = p_{\text{ext}} \cap m_{\text{near}} \quad (17)$$

$$C = p_{\text{ext}} \cap m_{\text{far}}, \quad (18)$$

where  $m_{\text{near}}$  and  $m_{\text{far}}$  denote the sets of elemental indices corresponding to intermediate-rank and low-rank interactions with element  $m$ , respectively. In case  $m \neq p$  the first term

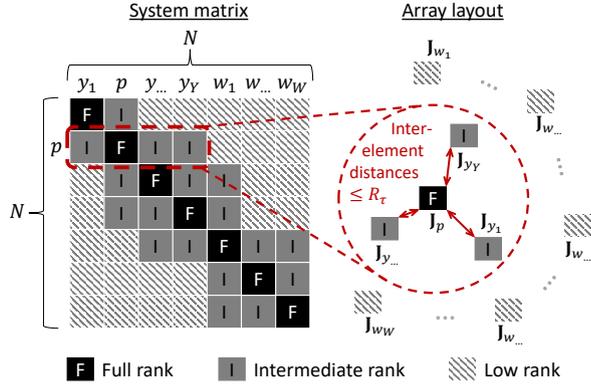


Fig. 3. Categorization of impedance matrix blocks according to rank. Intermediate-rank blocks are defined as non-self interactions between elements where the elemental bounding boxes are  $\leq R_r$  apart. The case of observer element  $p$  is highlighted, showing how the parameter  $R_r$  defines its set of  $Y$  intermediate-rank source interactions and its set of  $W$  low-rank source interactions, such that  $N = 1 + Y + W$ . The former set is compressed with standard ACA, while the latter set is compressed with TACA.

on the right hand side of (16) should also be reconstituted using either the standard ACA or TACA factors, whichever is applicable to it; similar reconstitution applies for the remaining blocks of the local solution matrix in (6).

### B. Construction of the TACA Factors

Consider observer element  $p$  in Fig. 3, with its set of far interacting source elements denoted by  $W = p_{\text{far}}$ , where  $W$  will denote the cardinality of the set when used in a mathematical expression. The cluster basis factors  $\mathbf{D}_p$  are computed through an inverse source process which starts with standard ACA factorization of a reduced version of the coupling block  $\mathbf{Z}_{pW}$ , relating  $p$  to its far field set  $W$ . The reduced version is used to avoid the expense of working directly with  $\mathbf{Z}_{pW}$ , while yet being sufficiently representative of  $\mathbf{Z}_{pW}$  such that identified row and column indices (crosses in the ACA factorization) are appropriate to the factorization of  $\mathbf{Z}_{pW}$  itself. Regarding notation, let  $M_p$  denote the set of indices of the degrees of freedom on element  $p$ , such that  $\mathbf{Z}_{pW} = \mathbf{Z}_{M_p W}$ .

The reduced version of  $\mathbf{Z}_{pW}$  is denoted  $\tilde{\mathbf{Z}}_{pW}^{M \times h}$ . The set  $h$  of columns of  $\tilde{\mathbf{Z}}_{pW}$  is a selected (sampling) subset of  $\mathbf{Z}_{pW}$ 's columns. These source sampling indices are obtained by selecting the same set of  $\beta$  sampling basis function from each element in  $W$ . On each elemental bounding box (defined in local Cartesian frames such that all boxes are identical), a number of Chebyshev nodes are established in each coordinate direction to yield coordinate points distributed throughout the volume of the box. The Chebyshev node numbers in the respective coordinate directions ( $C_x$ ,  $C_y$  and  $C_z$ ) are established to be approximately proportional to the box dimensions, while also tightly adhering to the requirement that

$$\beta = C_x C_y C_z \geq \max \left\{ 2^\Delta, \left\lceil \frac{M}{W} \right\rceil \right\}, \quad (19)$$

where  $\Delta \in \{1, 2, 3\}$  is the dimensionality of the bounding box. The condition (19) ensures a low but sufficient total number of samples ( $M \leq h = W\beta \ll WM$ ) as well as sufficient sampling diversity per source element. The actual set of  $\beta$  sampling basis functions per element are those with midpoints closest to the Chebyshev nodes. In practice this is determined on one antenna and translated to all others in  $W$ .

By performing standard ACA factorization on  $\tilde{\mathbf{Z}}_{pW}$  to the already-specified relative error threshold  $\varepsilon_{\text{ACA}}$ , the resulting sets of ACA factorization row and column indices (corresponding to the crosses selected by the ACA algorithm) are obtained:

$$\tau_p \subset M_p \quad \{\text{Row indices}\} \quad (20)$$

$$\sigma_p \subset h \quad \{\text{Column indices}\}. \quad (21)$$

The cluster basis disaggregation factors  $\mathbf{D}_p$  can now be constructed for  $p = 1, \dots, N$ , as

$$\mathbf{D}_p = \mathbf{Z}_{M_p \sigma_p} (\mathbf{Z}_{\tau_p \sigma_p})^{-1}. \quad (22)$$

Note that since the ACA with thresholding is an incomplete LU factorization, the matrix  $(\mathbf{Z}_{\tau_p \sigma_p})^{-1}$  in (22) can be obtained as a by-product during ACA factorization of  $\tilde{\mathbf{Z}}_{pW}$ ; it may also be directly obtained at slightly higher cost. The cluster basis aggregation factors  $\mathbf{A}_w$  for  $w = 1, \dots, N$  are not calculated since they are implicitly known via (13), as

$$\mathbf{A}_w = \mathbf{D}_w^T = \left[ \mathbf{Z}_{M_w \sigma_w} (\mathbf{Z}_{\tau_w \sigma_w})^{-1} \right]^T. \quad (23)$$

Finally, using the respectively selected indices  $\tau_p$  and  $\tau_w$  on the local domains of antennas  $p$  and  $w$  ( $p, w \in \{1, \dots, N\}$ ), the translation matrices are directly computed as

$$\mathbf{T}_{pw} = \mathbf{Z}_{\tau_p \tau_w}. \quad (24)$$

### C. Computational Cost

For the purposes of this estimation assume that  $R_r$  is much smaller than the array diameter and not proportional to it, i.e.  $\approx N^2$  blocks are compressed with TACA, while  $\mathcal{O}(N)$  blocks are compressed with standard ACA. Let  $\kappa \ll M$  and  $\rho < \kappa$  denote the average ranks of the standard ACA and TACA blocks, respectively. It follows that the runtime and storage costs of forming the standard ACA factors are  $\mathcal{O}(NM\kappa)$ . For the TACA disaggregation factors the runtime scales as  $\mathcal{O}(N^2\rho + NM\rho^2)$  and the storage as  $\mathcal{O}(NM\rho)$ . The runtime and storage costs of forming the TACA translation factors are  $\mathcal{O}((N\rho)^2)$ . For a sufficiently large array the total cost is dominated by that of the translation factors, yielding  $\mathcal{O}((N\rho)^2)$ .

The runtime cost of setting up all local problems via (16) is  $\mathcal{O}(\nu NM[M\kappa + N\rho^2])$  and that of solving them is  $\mathcal{O}(\nu NM^3)$ , where  $\nu$  is the number of iterations. The cost of the convergence tests (9) is  $\mathcal{O}(\nu N[M\kappa + N\rho^2 + M\rho])$ . The dominant component of the total runtime depends on the relative values of  $N$  and  $M$ . In the limit of very large arrays with  $N > M$ , the total solution runtime scales as  $\mathcal{O}(\nu M(N\rho)^2)$ . It will be demonstrated that  $\nu$  is very small in practice.

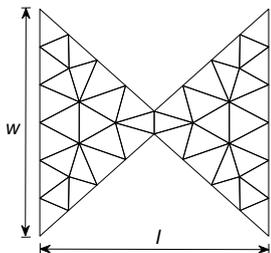


Fig. 4. Planar bow-tie antenna geometry with  $l = w = 499.7$  mm, which is  $\lambda/2$  at the simulation frequency of 300 MHz. It is meshed with 46 elements resulting in 55 RWG DoFs. The antenna is excited with a standard edge voltage source at its centre.

## VI. RESULTS

This Section presents numerical results to assess the performance of the IRB-DGFM formulation itself, as well as the hybridized single-level ACA acceleration scheme. Section VI-A assesses the impact of the far-interaction radius  $R_\tau$  and the ACA error tolerance  $\varepsilon_{ACA}$ , upon the TACA ranks. Section VI-B assesses the convergence performance of the IRB-DGFM for various array layouts, as well as to verify that the ACA IRB-DGFM is a true representation of the uncompressed version. Finally, Section VI-C presents further results for a more complex antenna array element, in support of the existing findings; as well as comparing memory requirements with the MLFMM. All arrays are uniformly excited.

### A. TACA Compression Assessment

The performance of the standard ACA for the MoM is well documented, see e.g. [7]. Here a test is conducted to verify the expected compression rank behaviour of the TACA scheme. Consider the simple planar bow-tie antenna array element shown in Fig. 4. Its overall dimensions are  $\lambda/2 \times \lambda/2$ , with  $\lambda/12$  meshing resulting in 55 DoFs. The compressed matrix is constructed for a regular array of  $31 \times 31 = 961$  elements with  $1\lambda$  distance between excitation ports (i.e.  $\lambda/2$  inter-element spacing), with varying  $R_\tau$  and  $\varepsilon_{ACA}$  values. Fig. 5 shows the resulting average TACA disaggregation ranks (i.e.  $\text{avg}(R_p)$ ),  $p = 1, \dots, N$ ; see (11). These results confirm that the ranks increase as the ACA compression error tolerance is tightened. The results (together with others not shown) also confirm that the TACA ranks reduce as  $R_\tau$  is increased, and converges to a certain minimum value when  $R_\tau$  is a sufficiently large multiple of the antenna element's far field condition (15). Thus it is important that for a large array, the value of  $R_\tau$  be sufficiently large, though not unnecessarily so. Further results in this respect are presented in Section VI-C.

### B. ACA IRB-DGFM Convergence Assessment

With the ACA compression threshold henceforth fixed at  $\varepsilon_{ACA} = 10^{-3}$ , consider next the convergence behaviour of the IRB-DGFM scheme by plotting the iteration residual (9) as a function of iteration number, for various normalized local domain radii  $r$  (5). Three arrays constructed from the antenna element in Fig. 4 are considered: a tightly-spaced regular array

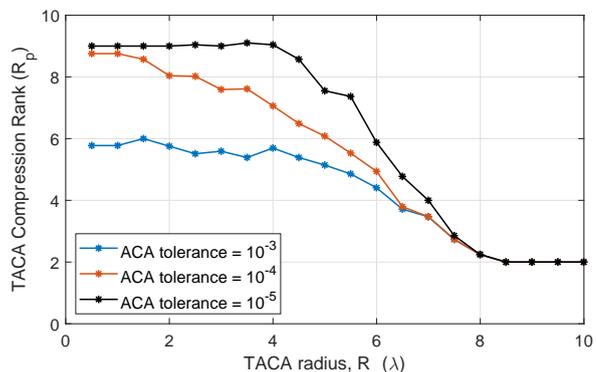


Fig. 5. Average TACA compression rank as a function of the far-interaction classification radius  $R_\tau$  beyond which TACA compression is used, for fixed ACA compression error threshold values of  $\varepsilon_{ACA} \in \{10^{-3}, 10^{-4}, 10^{-5}\}$ . These results are for a 961-element array of the bow-tie element in Fig. 4.

with inter-element spacing of  $\lambda/8$ , one with moderate spacing of  $\lambda/2$  and one with irregular spacing, as shown in Figs 6, 7 and 8, respectively. Three residual plots are shown for each array, corresponding to the extent to which ACA compression is used, as described in Table I.

Comparing sub-figures (b) and (c) in each instance, it is seen that the convergence histories are identical until the residual drops below  $10^{-3}$  (the ACA tolerance). In the full MoM case the error continues to decrease with further iterations, while it saturates in the case of ACA compression with full MoM testing. This saturation is due to the compression errors, converging to a non-zero residual value of

$$\epsilon^{(\infty)} = \frac{\|\mathbf{V} - \mathbf{Z}_{ACA}^{-1} \mathbf{V}\|_F}{\|\mathbf{V}\|_F}, \quad (25)$$

where  $\mathbf{Z}_{ACA}$  is the ACA-based approximation of the system matrix. Further comparing sub-figures (b) and (d) in each instance, it is seen that the reference convergence history (full MoM) is effectively identical to the full ACA case (which is the final formulation for practical use). Particularly so above the ACA compression error threshold, where the ACA results can be regarded as the true results. Clearly the hybridized single-level ACA compression is reliable, meaning that the errors introduced are below the set compression threshold. It is recommended to set the iterative scheme's termination threshold to

$$\mu = \varepsilon_{ACA} \quad \{\text{see Section IV-B}\}. \quad (26)$$

Lastly, it is worth noting that in spite of the significantly differing layouts, a fixed  $R_\tau$  value (noted in the figure captions) resulted in practically the same ACA accuracy characteristics (see sub-figures (c)). This affirms that  $R_\tau$  may be freely specified to maximize computational efficiency.

Now consider the actual performance of the IRB-DGFM in sub-figures (b) (or equivalently (d)). It is seen that the local domain radius has a very significant influence on convergence rate. For the tightly-spaced array  $r \geq 2$  is required for convergence, while  $r \geq 1$  suffices for the more loosely spaced

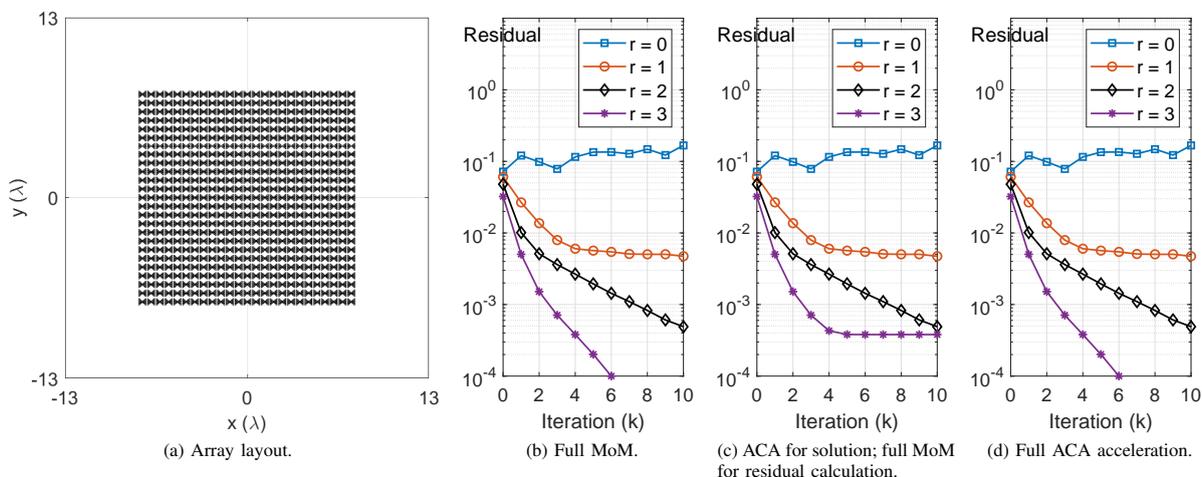


Fig. 6. IRB-DGFM residual vs. iteration count for a regular array with  $25 \times 25 = 625$  bow-tie antenna elements from Fig. 4, with tight inter-element spacing of  $\lambda/8$  in both directions. The hybridized ACA is performed with  $R_\tau = 5\lambda$ .

TABLE I  
SOLUTION CONFIGURATIONS FOR TESTING THE ERROR CONVERGENCE OF THE ACA-ACCELERATED IRB-DGFM SCHEME.

Sub-figure label in Figs. 6, 7 and 8 corresponding to this test setup	Matrix representation used for constructing local solutions (6)	Matrix representation used for evaluating the residual (9)
(b)	Full MoM	Full MoM
(c)	ACA factors	Full MoM
(d)	ACA factors	ACA factors

arrays. In all converging cases, the convergence is extremely rapid, requiring 3 ( $r = 3$ ) or 8 ( $r = 2$ ) iterations for the tightly-spaced array and 2 ( $r = 3$ ), 3 ( $r = 2$ ) or 7 ( $r = 1$ ) iterations for the other two. When considering the two more loosely spaced arrays, it is seen that the (ir)regularity of the layout does not influence the convergence rate. Given how low all of these iteration counts are and given the increase in size of the local problems (6) when  $r$  is increased, it is recommended to use the minimum  $r$  value which yields a convergent solution. Finally, for all three arrays the solution diverges at  $r = 0$ . This is interesting, since it corresponds to the standard DGFM being used in an iterative manner. It is concluded that the standard DGFM should be used with due caution.

### C. Results for a Log-Periodic Dipole Array

In this final section, results are presented for arrays with a more complex element, namely the bent log-periodic dipole antenna shown in Fig. 9. The antenna is designed for a  $-10$ dB bandwidth of 100MHz to 1.5GHz [30]. The average mesh size is set at  $\lambda/20$ . The mesh shown in the figure is for 300 MHz, yielding 708 DoFs per antenna.

Three arrays with antennas placed regularly on a rectangular grid are considered at 300 MHz, with  $\lambda/4$  inter-element spacing in both directions. The configurations are square with

$25 \times 25 = 625$ ,  $31 \times 31 = 961$  and  $35 \times 35 = 1225$  elements, respectively. For each of these arrays, the value of  $R_\tau$  is varied and the total storage recorded. Fig. 10 shows the results, which affirms that for sufficiently large arrays of a given element at a given frequency,  $R_\tau$  will have an optimal value independent of the exact array layout and the total array diameter. Furthermore, the nature of the memory dependence upon  $R_\tau$  is as expected. For small values of  $R_\tau$  the TACA ranks will be large which results in the translation factors requiring more storage. For large values of  $R_\tau$  the TACA ranks have already converged and increasingly, matrix blocks are unnecessarily being compressed with more expensive standard ACA factorization. Table II lists the memory requirements at the optimal value of  $R_\tau = 8\lambda$ .

For the 961-element array, Fig. 11 shows the residual convergence histories for various  $r$  values. For  $r \geq 1$  rapid convergence is again observed. Consider now the memory requirements of the present solver, in comparison with those from the MLFMM solver in FEKO [31], also listed in Table II. It can be seen that the hybrid ACA scheme proposed here is very much competitive with the MLFMM, though of course for sufficiently large arrays the MLFMM scaling is well known to be superior. Notwithstanding the latter fact, the ACA-accelerated IRB-DGFM solver offers a compelling set of performance characteristics: storage is very low, though not as low as the MLFMM, but the iterative convergence is extremely rapid and very reliable, which is in contrast to a preconditioned Krylov-subspace solver for the EFIE MoM where the convergence can be unpredictable and very often requires very many more iterations than the  $2 \sim 8$  typically found with the IRB-DGFM solver. Furthermore, the ACA IRB-DGFM is ideally suited to parallelization.

Finally, consider an array with a fixed square grid layout, consisting of  $21 \times 21 = 441$  elements, with 249.83 mm inter-element spacings in both directions (equal to  $\lambda/4$  at 300 MHz). Table III presents the storage requirements and iteration counts for three different excitation frequencies and

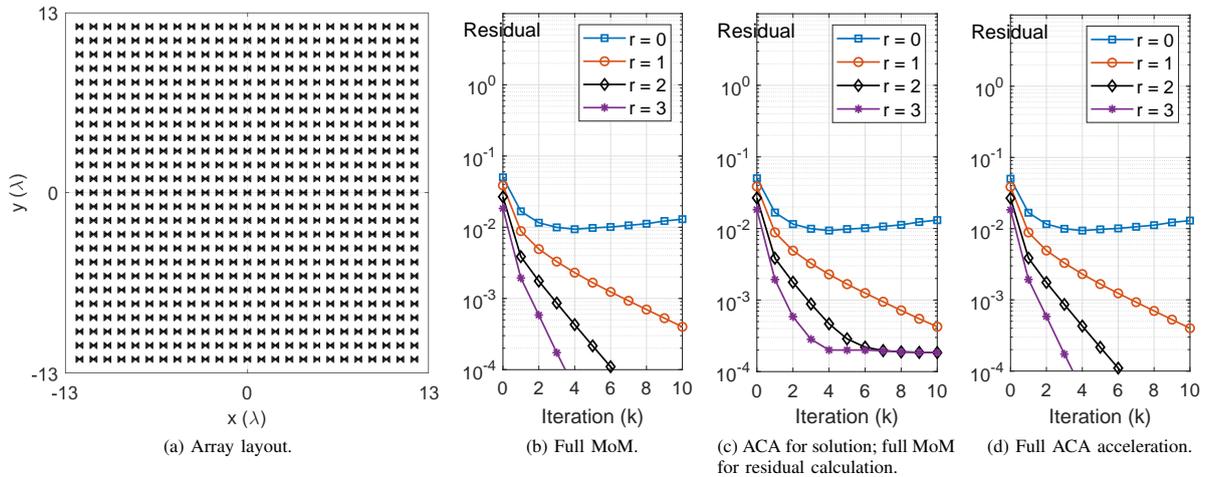


Fig. 7. IRB-DGFM residual vs. iteration count for a regular array with  $25 \times 25 = 625$  bow-tie antenna elements from Fig. 4, with moderate inter-element spacing of  $\lambda/2$  in both directions. The hybridized ACA is performed with  $R_\tau = 5\lambda$ .

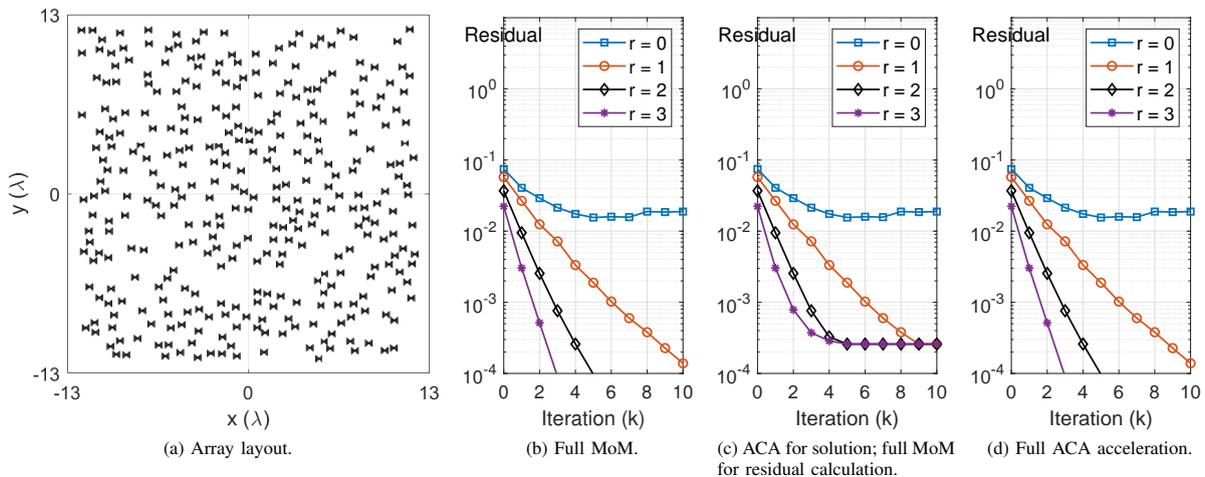


Fig. 8. IRB-DGFM residual vs. iteration count for an irregular array with 521 bow-tie antenna elements from Fig. 4. Spacing vary across the array with no two elements touching. The hybridized ACA is performed with  $R_\tau = 5\lambda$ .

three different  $R_\tau$  specifications. Convergence is determined by  $\mu = 10^{-3}$  (see (26)), with  $r = 2$  set in the IRB-DGFM. It is seen that the iteration counts are independent of frequency, remaining constant at the low value of 4. Considering the memory requirements (which are broken down into standard ACA and TACA contributions), it is seen that at 300 MHz the required memory increases for  $R_\tau > 8\lambda$  due to excessive use of standard ACA; this is consistent with the results in Fig. 10. At the higher frequencies it is seen that the total required memory continues to decrease as  $R_\tau$  is increased, over the  $R_\tau$ -ranges considered. This demonstrates the point that  $R_\tau$  should be set to a multiple of the far field distance (15), which increases linearly with frequency for a fixed antenna geometry. However, the storage requirements at the higher frequencies are very weakly dependent upon  $R_\tau$  over the considered ranges. Also, the storage requirements scale almost linearly with DoFs, which is very good. These observations

TABLE II  
MEMORY REQUIREMENTS FOR SOLVING BENT LOG-PERIODIC DIPOLE ARRAYS OF VARIOUS SIZES, AT 300 MHz. FOR THE ACA IRB-DGFM SOLUTIONS THE OPTIMAL VALUE OF  $R_\tau = 8\lambda$  IS USED.

Array elem.	Tot. DoFs	ACA IRB-DGFM	MLFMM (FEKO)
625	442,500	8.22 GB	6.67 GB
961	680,388	13.71 GB	10.78 GB
1225	867,300	17.43 GB	14.91 GB

lead to the conclusion that a broad range of  $R_\tau$  values around its exact optimal value, yields very efficient performance.

## VII. CONCLUSION

An accelerated iterative method for efficient and reliable solution of arrays consisting of identical disjoint elements, is presented. The most important benefits of the method are

TABLE III  
STORAGE REQUIREMENTS AND ITERATION COUNTS (WITH  $r = 2$  TO ACHIEVE CONVERGENCE WITH  $\mu = 10^{-3}$ ) FOR SOLVING A 441-ELEMENT BENT LOG-PERIODIC DIPOLE ARRAY AT THREE DIFFERENT FREQUENCIES, WITH THREE DIFFERENT  $R_\tau$  VALUES.

Freq. (MHz)	Tot. DoFs	Iterations	$R_\tau$	D-Factors (GB)	T-Matrices (GB)	Std. ACA-Factors (GB)	Tot. Storage (GB)
300	312, 228	4	$8\lambda$	0.0233	0.0725	4.653	4.748
			$12\lambda$	0.0224	0.0716	4.845	4.939
			$16\lambda$	0.0223	0.0715	5.041	5.135
600	596, 232	4	$8\lambda$	0.0381	0.1501	8.675	8.863
			$12\lambda$	0.0299	0.1063	8.721	8.857
			$16\lambda$	0.0267	0.0975	8.727	8.852
900	1, 133, 370	4	$8\lambda$	0.0714	0.3207	18.741	19.079
			$12\lambda$	0.0467	0.1621	18.791	19.000
			$16\lambda$	0.0302	0.0953	18.833	18.958

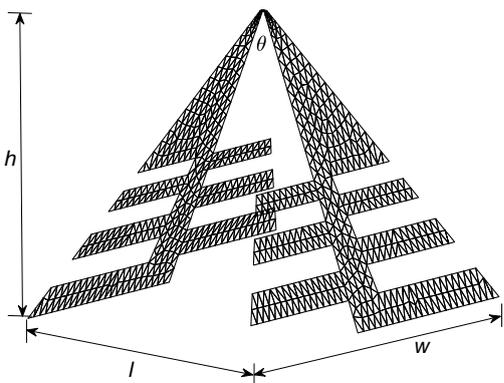


Fig. 9. Mesh at 300 MHz, of a bent log-periodic dipole antenna, with dimensions  $l = 468.05$  mm,  $w = 926.10$  mm,  $h = 270.23$  mm and  $\theta = 120^\circ$ .

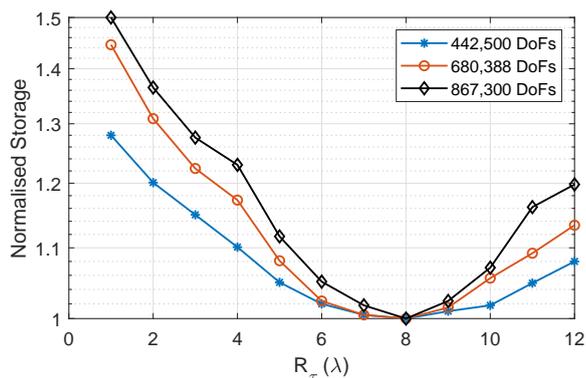


Fig. 10. Storage requirements for the ACA IRB-DGFM solution of arrays with varying numbers of elements, of the bent log-periodic dipole antenna shown in Fig. 9, at 300 MHz. Each trace is obtained by varying  $R_\tau$  and recording the total memory required for the ACA factors and then normalizing the data for each array with respect to the minimum storage achieved for that array. The actual minimum values are listed in Table II.

that it converges rapidly and that its memory requirement is comparable to that of the MLFMM for arrays with in excess of a thousand closely-spaced elements. Although the memory scaling is not as favorable as with the MLFMM, the new

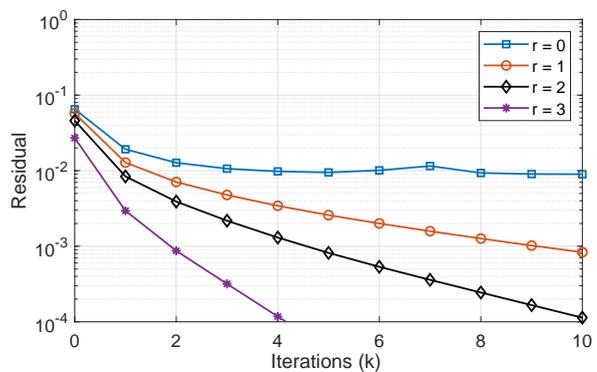


Fig. 11. ACA IRB-DGFM (with  $R_\tau = 8\lambda$ ), residual convergence histories for various local domain radii ( $r$ ), solving the 961-element bent log-periodic dipole array at 300 MHz.

method converges in a very small number of iterations, with the rate being dependent on the local domain size and not the global array size; and no preconditioner is required. This, together with relative implementation simplicity (in particular the hybridized single-level ACA compression scheme) and the potential for very efficient parallelization (since only local problems are solved) makes for an attractive option for analysis of very large problems falling within the scope of applicability.

Comparing the ACA IRB-DGFM with a leading MBF-based solver for the same class of problems [19], the runtimes scale respectively as  $\mathcal{O}(\nu M(N\rho)^2)$  and  $\mathcal{O}((NM_{\text{MBF}})^3)$ , where  $M_{\text{MBF}}$  refers to the number of MBFs per array element (typically,  $M_{\text{MBF}} \ll M$ ). Should MBFs be introduced into the ACA IRB-DGFM, then its runtime scaling reduces further to  $\mathcal{O}(\nu M_{\text{MBF}}(N\rho)^2)$ , although the property of convergence to the true MoM solution is then replaced with convergence to the true MBF-MoM solution. In either case, the ACA IRB-DGFM would be more efficient for sufficiently large problems  $N \rightarrow \infty$ . For smaller problems, an MBF-based solver with fast construction of the reduced system (such as in [19]) will likely be faster. In addition to the runtime scaling analysis in Section V-C, measured runtimes are not presented as the research code is not optimized for speed, but the presented memory results do

confirm the effective performance of the acceleration scheme. Note that the sparsification and tessellation concepts presented in [19] can similarly be used to reduce the ACA IRB-DGFM solution cost for extremely large arrays.

All results have focussed on active antenna arrays. The IRB-DGFM is also applicable to the case of a single excited element, e.g. to calculate  $Z$ -parameters or embedded element patterns. However, its convergence rate slows down due to initial lack of excitation on some local domains. See [24], [32] for preliminary work on addressing this issue. Another useful extension would be to allow for array elements which are not all identical.

## REFERENCES

- [1] S. M. Rao, D. R. Wilton, and A. W. Glisson, "Electromagnetic scattering by surfaces of arbitrary shape," *IEEE Transactions on Antennas and Propagation*, vol. 30, no. 3, pp. 409–418, May 1982.
- [2] Y. Zhuang, K.-L. Wu, C. W. Wu, and J. Litva, "A combined full-wave CG-FFT method for rigorous analysis of large microstrip antenna arrays," *IEEE Transactions on Antennas and Propagation*, vol. 44, no. 1, pp. 102–109, 1996.
- [3] W. C. Chew, E. Michielssen, J. M. Song, and J. M. Jin, *Fast and efficient algorithms in computational electromagnetics*. Artech House, Inc., 2001.
- [4] T. F. Eibert, "A diagonalized multilevel fast multipole method with spherical harmonics expansion of the  $k$ -space integrals," *IEEE Transactions on Antennas and Propagation*, vol. 53, no. 2, pp. 814–817, 2005.
- [5] M. Bebendorf, "Approximation of boundary element matrices," *Numerische Mathematik*, vol. 86, no. 4, pp. 565–589, 2000.
- [6] S. Kurz, O. Rain, and S. Rjasanow, "The adaptive cross-approximation technique for the 3D boundary-element method," *IEEE Transactions on Magnetics*, vol. 38, no. 2, pp. 421–424, 2002.
- [7] K. Zhao, M. N. Vouvakis, and J.-F. Lee, "The adaptive cross approximation algorithm for accelerated method of moments computations of EMC problems," *IEEE Transactions on Electromagnetic Compatibility*, vol. 47, no. 4, pp. 763–773, 2005.
- [8] A. Schröder, H.-D. Brüns, and C. Schuster, "Fast evaluation of electromagnetic fields using a parallelized adaptive cross approximation," *IEEE Transactions on Antennas and Propagation*, vol. 62, no. 5, pp. 2818–2822, 2014.
- [9] S.-C. Lee, M. N. Vouvakis, and J.-F. Lee, "A non-overlapping domain decomposition method with non-matching grids for modeling large finite antenna arrays," *Journal of Computational Physics*, vol. 203, no. 1, pp. 1–21, 2005.
- [10] Y.-J. Li and J.-M. Jin, "A new dual-primal domain decomposition approach for finite element simulation of 3-D large-scale electromagnetic problems," *IEEE Transactions on Antennas and Propagation*, vol. 55, no. 10, pp. 2803–2810, 2007.
- [11] Z. Peng, X.-C. Wang, and J.-F. Lee, "Integral equation based domain decomposition method for solving electromagnetic wave scattering from non-penetrable objects," *IEEE Transactions on Antennas and Propagation*, vol. 59, no. 9, pp. 3328–3338, 2011.
- [12] W. B. Lu, T. J. Cui, Z. G. Qian, X. X. Yin, and W. Hong, "Accurate analysis of large-scale periodic structures using an efficient sub-entire-domain basis function method," *IEEE Transactions on Antennas and Propagation*, vol. 52, no. 11, pp. 3078–3085, 2004.
- [13] L. Matekovits, V. A. Laza, and G. Vecchi, "Analysis of large complex structures with the synthetic-functions approach," *IEEE Transactions on Antennas and Propagation*, vol. 55, no. 9, pp. 2509–2521, 2007.
- [14] R. Maaskant, R. Mittra, and A. Tjihuis, "Fast analysis of large antenna arrays using the characteristic basis function method and the adaptive cross approximation algorithm," *IEEE Transactions on Antennas and Propagation*, vol. 56, no. 11, pp. 3440–3451, 2008.
- [15] C. Craeye, J. Laviada, R. Maaskant, and R. Mittra, "Macro basis function framework for solving Maxwell's equations in surface integral equation form," *The FERMAT Journal*, vol. 3, pp. 1–16, 2014.
- [16] D. J. Ludick, M. M. Botha, R. Maaskant, and D. B. Davidson, "The CBFM-enhanced Jacobi method for efficient finite antenna array analysis," *IEEE Antennas and Wireless Propagation Letters*, vol. 16, pp. 2700–2703, 2017.
- [17] C. Craeye, "A fast impedance and pattern computation scheme for finite antenna arrays," *IEEE Transactions on Antennas and Propagation*, vol. 54, no. 10, pp. 3030–3034, October 2006.
- [18] A. Freni, P. De Vita, P. Pirinoli, L. Matekovits, and G. Vecchi, "Fast-factorization acceleration of MoM compressive domain-decomposition," *IEEE Transactions on Antennas and Propagation*, vol. 59, no. 12, pp. 4588–4599, 2011.
- [19] H. Bui-Van, J. Abraham, M. Arts, Q. Gueuning, C. Raucy, D. González-Ovejero, E. de Lera Acedo, and C. Craeye, "Fast and accurate simulation technique for large irregular arrays," *IEEE Transactions on Antennas and Propagation*, vol. 66, no. 4, pp. 1805–1817, 2018.
- [20] K. Sewraj and M. M. Botha, "Directional method to compute reduced matrix system in MBF solvers," in *Proceedings of the European Conference on Antennas and Propagation (EuCAP)*, March 2020, pp. 1–5, Copenhagen, Denmark.
- [21] A. K. Skrivervik and J. R. Mosig, "Analysis of finite phase arrays of microstrip patches," *IEEE Transactions on Antennas and Propagation*, vol. 41, no. 8, pp. 1105–1114, August 1993.
- [22] —, "Analysis of printed array antennas," *IEEE Transactions on Antennas and Propagation*, vol. 45, no. 9, pp. 1411–1418, September 1997.
- [23] D. J. Ludick, R. Maaskant, D. B. Davidson, U. Jakobus, R. Mittra, and D. I. L. de Villiers, "Efficient analysis of large aperiodic antenna arrays using the domain Green's function method," *IEEE Transactions on Antennas and Propagation*, vol. 62, no. 4, pp. 1579–1588, April 2014.
- [24] M. Chose and M. M. Botha, "Improvements to the domain Green's function method for antenna array analysis," in *Proceedings of the International Conference on Electromagnetics in Advanced Applications (ICEAA '19)*, Sept 2019, pp. 392–395, Granada, Spain.
- [25] —, "An iterative method for the analysis of large disjoint antenna arrays," in *2020 14th European Conference on Antennas and Propagation (EuCAP)*. IEEE, 2020, pp. 1–4, Copenhagen, Denmark.
- [26] W. C. Gibson, *The method of moments in electromagnetics*. CRC press, 2014.
- [27] D. Ludick, M. Botha, D. Davidson, and U. Jakobus, "Introducing the iterative domain green's function method for finite array analysis," in *2016 IEEE/ACES International Conference on Wireless Information Technology and Systems (ICWITS) and Applied Computational Electromagnetics (ACES)*. IEEE, 2016, pp. 1–2.
- [28] M. Bebendorf and R. Venn, "Constructing nested bases approximations from the entries of non-local operators," *Numerische Mathematik*, vol. 121, no. 4, pp. 609–635, 2012.
- [29] W. L. Stutzman and G. A. Thiele, *Antenna Theory and Design*, 2nd ed. New York: John Wiley and Sons, 1998.
- [30] Simulia, *Antenna Magus*, 2020. [Online]. Available: <https://www.3ds.com/products-services/simulia/products/antenna-magus/>
- [31] Altair, *FEKO*. 2020. [Online]. Available: <https://www.altair.com/feko/>
- [32] M. Chose and M. M. Botha, "Efficient iterative method for computation of antenna array embedded element patterns," in *Proceedings of the IEEE International Antennas and Propagation Symposium*, July 2020, pp. 1051–1052, Montréal, Canada.



## Appendix C

### Conference paper — Reaction Matrix Computation with NCA and DCA [3]

K. Sewraj and M. M. Botha, “Computation of MBF Reaction Matrices for Antenna Array Analysis, with a Directional Method,” in *Proceedings of the International Conference on Electromagnetics in Advanced Applications (ICEAA)*, Granada, Spain, September 2019, pp. 385–389.

# Computation of MBF Reaction Matrices for Antenna Array Analysis, with a Directional Method

Keshav Sewraj

Dept. of Electrical and Electronic Engineering  
Stellenbosch University  
Stellenbosch, South Africa  
20635958@sun.ac.za

Matthys M. Botha

Dept. of Electrical and Electronic Engineering  
Stellenbosch University  
Stellenbosch, South Africa  
mmbbotha@sun.ac.za

**Abstract**—Computing the reduced matrix reaction terms in macro basis function (MBF) solvers for large antenna array analysis, is computationally expensive. A hierarchical structure together with a fast low-rank factorization technique can be used in order to improve the computational complexity. However, the rank is expected to increase with the subdomain size for electrically large problems, hence degrading the performance of the algorithm. Directional methods are used to ensure a constant rank for oscillatory kernel applications by subdividing the interacting region into pyramids. This paper report on preliminary investigations for using the directional cross approximation method to compute reaction terms.

**Index Terms**—Adaptive cross approximation (ACA), directional cross approximation, electric field integral equation (EFIE), method of moments (MoM).

## I. INTRODUCTION

A numerical electromagnetic field solver is a fundamental tool during the design process of antenna arrays. Extremely large arrays, namely the Mid-Frequency Aperture Array (MFAA) [1], will be a central part of the Square Kilometre Array (SKA) radio-astronomy project [2]. The MFAA, consisting of several stations (antenna arrays), will have stringent design requirements such as high sensitivity, large field of view, and wide bandwidth. The design of the MFAA is still in the research phase. The array configuration will either be dense or sparse, and both periodic and irregular element arrangements are under consideration. The number of antennas per station will depend on the antenna used and array configuration, however, due to the high sensitivity required, thousands of antennas can be expected. It is a serious challenge to accurately analyze antenna arrays of this size for multiple excitation schemes. During the last decade, the Macro Basis Function (MBF) [3] methods have successfully been applied to antenna array analysis problems for SKA applications such as the Characteristic Basis Function Method (CBFM) [4] for periodic structures. An MBF approach together with a fast interpolatory technique [5] is used for the irregular array configuration in the SKA Low-Frequency Aperture Array

(LFAA) [6]. The Domain Green's Function Method (DGFM) [7] has been applied to analyze sparse antenna arrays.

In macro basis function techniques, physics-based functions called MBFs are used to reduce the size of the Method of Moments' (MoM) impedance matrix, which can then be solved directly. A direct MBF solver for array analysis consists of 3 steps, namely

- 1) Generate MBFs (primary, secondary/higher-order MBFs) [8] on individual antenna elements.
- 2) Create the reduced impedance matrix. The reduced impedance matrix block for the interaction between subdomains  $p$  and  $q$  can be expressed as in (1), where  $J$  is an MBF group and  $H$  is the hermitian transpose.

$$Z_{pq}^{\text{Red}} = \langle J_p^H, Z_{pq} J_q \rangle. \quad (1)$$

- 3) Solve the reduced impedance matrix system to obtain MBF coefficients.

The first bottleneck of MBF solvers when up-scaling the size of the array is the quadratic scaling of the computational time to obtain the reaction terms of the reduced impedance matrix (step 2) since (1) needs to be repeated for all antenna interactions ( $\mathcal{O}(M^2)$  interactions, where  $M$  is the number of antennas in the array). Secondly, the size of the reduced impedance matrix becomes large (tens of thousands of MBFs for extremely large arrays) and thus expensive to solve directly (step 3), despite the massive reduction from the original MoM impedance matrix.

Improving the computational time to create the reduced matrix has been the subject of various research endeavours. For periodic array analysis in [4], Adaptive Cross Approximation (ACA) is applied to factorize low-rank sub-blocks as  $Z_{pq} \approx UV^T$ . Matrices  $U$  and  $V$  have the size of  $N_i \times R$ , where  $R$  is the numerical rank (depending on a pre-defined threshold),  $N_i$  is the number of elements per subdomain and  $R \ll N_i$ . The reduced impedance sub-block between subdomain  $p$  and  $q$  can then be written as  $Z_{pq}^{\text{Red}} \approx \langle J_p^H, UV^T J_q \rangle$ , which only consists of vector-vector products instead of the more expensive matrix-vector product. The Toeplitz nature of the impedance matrix due to the translational symmetry in the array was then exploited to compute only  $\mathcal{O}(M)$  interactions instead of  $\mathcal{O}(M^2)$ . For large irregular arrays, applying ACA to

Funding by the South African SKA Project (SKA SA) towards this research is hereby acknowledged (www.ska.ac.za). The National Research Foundation of South Africa also supported this work (Grant Numbers: 75322, 96222 and 106033).

$\mathcal{O}(M^2)$  interaction is computationally too expensive. In [5], a harmonic-polynomial (HARP) model was created beforehand based on a limited number of MBF interactions, positioned on a grid. During the process of creating the reduced impedance matrix, the reaction terms can then be computed rapidly using the harmonic-polynomial expression.

Alternatively, one can attempt to improve the computational complexity of creating the reduced system by partitioning the entire domain using a hierarchical tree, and use fast factorization techniques (for instance, ACA) to factorize matrix blocks at every level. Two different hierarchical partitioning strategies to rapidly compute reaction terms at different levels in CBFM solvers have been used in [9]–[11] for electromagnetic scattering applications, such that the reduced impedance matrix is only set up in factorized form, saving computational resources. However, applying ACA to compute block interactions at every level might not be an optimal solution for large array analysis, since for oscillatory kernels the ACA algorithm can scale as poorly as  $\mathcal{O}(N_i^3)$  when the size of the subdomains increase ( $N_i \rightarrow \infty$ ) [12]. Instead, directional methods are more appropriate in such cases. This paper presents an exploration of this idea with preliminary results.

## II. DIRECTIONAL METHOD

### A. Nested Cross Approximation

Before going to the directional methods, the nested cross approximation [13] is first reviewed. Instead of factorizing matrix blocks as  $Z_{pq} \approx UV^T$  as in ACA, it is desired to factorize the matrix as in an  $\mathcal{H}^2$ -matrix [14] format, such as

$$Z_{pq} \approx U_p S_{pq} V_q^T. \quad (2)$$

Matrices  $U$  and  $V$  are called cluster bases which depend only on the observer ( $p$ ) and source ( $q$ ) domains respectively.  $S$  is the coupling matrix having the size of  $R_p \times R_q$  (numerical rank of observer and source domains). Secondly, the factorization is nested. That is,  $U$  and  $V$  are created only at the leaf level (deepest level of the hierarchical tree). At non-leaf levels, cluster basis of a parent (e.g. level  $l$ ) can be built from that of its child cluster (level  $l+1$ ) through a transfer matrix, such as

$$U_p^l = U_{p^c}^{l+1} B^{(l+1,l)} \quad (3)$$

$$V_q^l = C^{(l,l+1)} V_{q^c}^{l+1}. \quad (4)$$

Matrices  $B$  and  $C$  are the transfer matrices and  $c$  refers to the child cluster of  $p$  or  $q$ . This structure is more memory efficient than the ACA for large structures, since only the transfer matrices need to be stored for each cluster at non-leaf levels, and coupling matrices for every interaction. These matrices are much smaller in size compared to the factorized matrix in ACA when the subdomain size is large.

In nested cross approximation, the following procedure is used to create the cluster basis,  $U$ , of a given leaf-level subdomain.

- All the elements in the subdomain are selected and denoted as  $\tilde{\tau}_t$ .

- A pre-defined number of source elements is selected from the far-field of the subdomain by choosing the closest elements to a grid of Chebyshev nodes. The selected elements are denoted as  $\tilde{\sigma}_t$ . This step is performed hierarchically starting from the top of the tree to ensure the asymptotic scaling is maintained.
- ACA, with a pre-defined threshold value, is then performed on the matrix block  $Z_{\tilde{\tau}_t, \tilde{\sigma}_t}$ , which is a small subset of the full MoM impedance matrix,  $Z$ . The selected rows and columns (or, the pivots) are denoted as  $\tau_t$  and  $\sigma_t$ .

The  $U$  matrix can then be constructed as

$$U = Z_{t\sigma_t} (Z_{\tau_t\sigma_t})^{-1}, \quad (5)$$

where  $t$  refers all the low-level elements in the subdomain. The inverse in (5) can be obtained as a by-product of the ACA while obtaining the pivots. The  $\tilde{\tau}_t$  elements at non-leaf levels are obtained by unioning all the  $\tau_t$  elements of the children clusters. The transfer matrix is obtained as

$$B = Z_{\tau_{t_c}\sigma_t} (Z_{\tau_t\sigma_t})^{-1}, \quad (6)$$

where  $\tau_{t_c}$  is the  $\tau_t$  elements of the child's cluster. If Galerkin testing is used, matrices  $V$  and  $C$  can be obtained as the transpose of  $U$  and  $B$  respectively.

### B. Directional Cross Approximation

The nested cross approximation method has a linear complexity for applications with a smooth Green's function or low-frequency. However, this is not the case for oscillatory kernels as in (7) since the numerical rank increases as the domain size increases.

$$G(\mathbf{r}, \mathbf{r}') = \frac{e^{-jk|\mathbf{r}-\mathbf{r}'|}}{4\pi|\mathbf{r}-\mathbf{r}'|}. \quad (7)$$

Directional methods [15]–[19] can be used to obtain a fixed rank compression for oscillatory kernel applications, as in the case of antenna array analysis. The far-field region (subset of the two-dimensional plane containing the antenna array) is subdivided into triangles (pyramids for general volumetric structures) with angle depending on the subdomain size as  $\mathcal{O}(\frac{1}{kD})$ , where  $k$  is the wavenumber and  $D$  is the diameter of the domain. Hence, subdomains at the leaf level covers the widest angle. The directional subdivision is done hierarchically, starting from the leaf level. At every level up the tree, the triangles are divided into 2 (hence halving the angle). This leads to the directional far-field of a parent cluster to be a subset of the directional far-field of its child clusters.

The distance criterion for the interaction between two subdomains ( $p$  and  $q$ ) to be considered admissible is given as

$$\eta_{\text{high}} \text{dist}(p, q) \geq k \max(D_p^2, D_q^2), \quad (8)$$

where  $\eta_{\text{high}}$  is an admissibility constant, and  $D_p$  and  $D_q$  are the diameters of subdomains  $p$  and  $q$ , respectively. Whereas, the admissibility criterion for nested-cross approximation (for low-frequency) is given as

$$\eta_{\text{low}} \text{dist}(p, q) \geq \max(D_p, D_q), \quad (9)$$

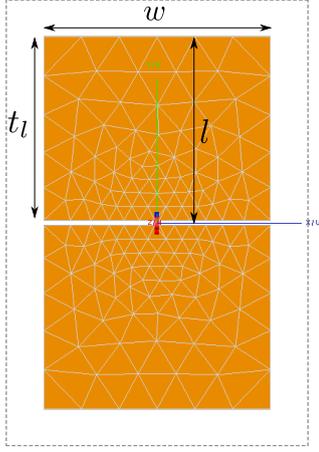


Fig. 1. Blade dipole antenna with meshing for 750 MHz. Dimensions:  $l = 11.65$  cm,  $w = 14.13$  cm,  $t_l = 11.49$  cm.

where  $\eta_{\text{low}}$  is the admissibility constant for low-frequency. The  $U$  matrix in the direction  $e$  can then be constructed as

$$U^e = Z_{t\sigma_t^e} (Z_{\tau_t^e \sigma_t^e})^{-1}, \quad (10)$$

where  $\sigma_t^e$  is the selected elements in the directional far-field  $e$  of the subdomain and  $\tau_t^e$  is the selected elements in the subdomain due to sources in the directional far-field  $e$ . The transfer matrix in the direction  $e$  is then obtained as

$$B^e = Z_{\tau_{t_{e_c}} \sigma_t^e} (Z_{\tau_t^e \sigma_t^e})^{-1}, \quad (11)$$

where  $\tau_{t_{e_c}}$  is the  $\tau_t$  elements of the child cluster in the direction of  $e_c$ . For a more detailed explanation of the methods reviewed in Sections II-A and II-B, see [13], [17].

### III. EARLY STAGE EXPERIMENTATIONS

The antenna array element used in this paper is a blade dipole antenna as shown in Figure 1, and a random array configuration of 1024 antennas is considered, as shown in Figure 2. Two excitation frequencies are considered, namely 750 MHz and 950 MHz, resulting in 363 and 511 low-level basis functions per antenna element, respectively.

#### A. Comparison of Rank for Directional and Non-Directional Methods

The impedance matrix resulting from the antenna array shown in Figure 2 is partitioned hierarchically, where an antenna element is the subdomain at the leaf level (level 7 in Figure 3). The number of antennas per subdomain is doubled at every level, and level 1 (in Figure 3) contains the largest subdomains. The total number of basis functions are 371, 712 and 523, 264 for excitation frequencies 750MHz and 950MHz, respectively. Figure 3 shows the average rank at each level. The directional method results in fewer levels due to the distance admissibility criterion in (8), which depends on the diameter-squared of the subdomains, rather than the diameter as in

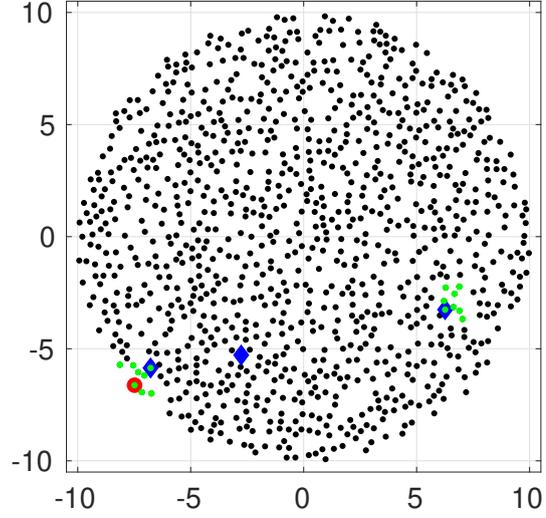


Fig. 2. Antenna array configuration containing 1024 elements with minimum inter-element distance of 0.3 m. Red circle and blue diamonds refer to the observer antenna (Ant. 1) and source antennas (Ant. 8, 197 and 713) respectively, in Table III. The green dots represent the interacting subdomains at level 4 in Row 3 of Table III.

the non-directional case (9). For non-directional methods, the average rank per level drastically increases with the subdomain size. The decrease in average rank for levels 1 and 2 is because these levels contain fewer interacting subdomains.

For the directional method, the higher rank at level 7 is because the interaction region has been divided with only 2 directions at the leaf level in this experimental implementation, which is not sufficient. Should more directions be used at the leaf level, a nearly constant rank is expected for the first 3 levels, until the rank drops due to fewer subdomain interactions at the last two levels.

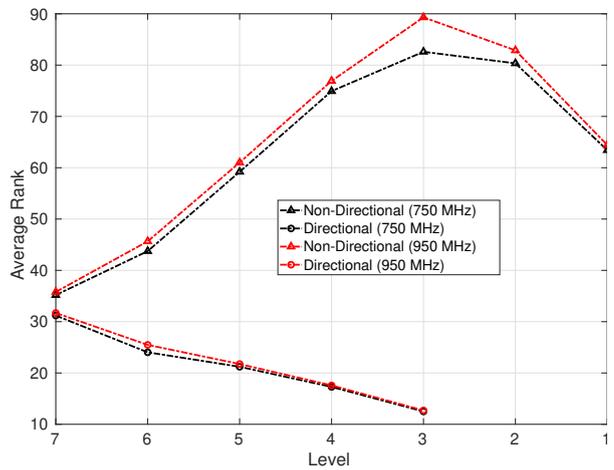


Fig. 3. Comparison of average rank per level for directional and non-directional methods with ACA thresholding of  $1e-4$ .

### B. Memory requirement

The memory requirement to store the  $U$ ,  $B$  and  $S$  matrices for 750 MHz and 950 MHz are shown in Tables I and II. Only half of the interactions for the coupling matrices have been stored due to the symmetry of the impedance matrix (for Galerkin testing). The poor performance of the directional method to store the  $U$  matrices in the experimental implementation can be seen in Tables I and II. This is because antenna elements were used as subdomains at the leaf level, which are actually too large and inefficient. Consequently, even though the directional method results in lower ranks, large  $U$  matrices ( $N_i \times R$ ) are stored for multiple directions, which leads to higher memory consumption than with the non-directional method. This is a strong indication that the antennas should be further partitioned and compressed. Then at the antenna levels, only transfer matrices will be used.

On the other hand, the directional method uses less memory to store the coupling matrices. This provides motivation to use the directional method as the array size or frequency increases.

TABLE I  
MEMORY REQUIREMENT AT 750 MHz

	Directional (MB)	Non-Directional (MB)
Cluster Bases ( $U$ )	867.0	490.6
Transfer Matrices ( $B$ )	192.2	223.0
Coupling Matrices ( $S$ )	462.3	615.4
<b>Total</b>	<b>1521.5</b>	<b>1329.0</b>

TABLE II  
MEMORY REQUIREMENT AT 950 MHz

	Directional (MB)	Non-Directional (MB)
Cluster Bases ( $U$ )	1187.1	702.6
Transfer Matrices ( $B$ )	206.7	239.9
Coupling Matrices ( $S$ )	1253.4	1519.0
<b>Total</b>	<b>2647.2</b>	<b>2461.5</b>

### C. Accuracy

The accuracy of a few approximated, reduced impedance matrix blocks at antenna level are shown in this section. An approximated, reduced impedance matrix block representing the coupling between subdomains  $p$  and  $q$ , is given as

$$\tilde{Z}_{pq}^{\text{Red}} = \left\langle J_p^H, \tilde{Z}_{pq} J_q \right\rangle, \quad (12)$$

where  $\tilde{Z}_{pq}$  is the reconstructed matrix block using the factorized terms from the directional cross approximation method. Primary and secondary MBFs [8] are used, using a radius of influence of 1.5 m around the antenna in question. A Singular Value Decomposition (SVD) operation is performed on the MBF groups as in [4]. Both the SVD and ACA thresholds are set to  $1e-4$ . The error norm is given as

$$\text{Error Norm} = \frac{\|Z_{pq}^{\text{Red}} - \tilde{Z}_{pq}^{\text{Red}}\|_2}{\|Z_{pq}^{\text{Red}}\|_2} \quad (13)$$

In Table III, Antenna 1 is set as the observer subdomain, and various source antennas are used. Depending on the distance between the source and observer antenna, the interaction happens at different levels. The ‘‘Highest Level’’ in Table III corresponds to the levels (on the abscissa) in Figure 3. Level 7 is the leaf level, containing a single antenna per subdomain, and levels 5 and 4 contain 4 and 8 antennas per subdomain, respectively. It can be shown that the accuracy requirements as set by the thresholding value are met.

TABLE III  
ERROR NORM OF APPROXIMATED, REDUCED IMPEDANCE MATRIX SUB-BLOCKS

Source Antenna	Highest Level	Error Norm
8	7	$1.750e-4$
197	5	$6.747e-4$
713	4	$4.993e-4$

Observer subdomain: Antenna 1.

## IV. DISCUSSION

Hierarchical partitioning is more efficient than single-level partitioning, for compression of large MoM matrices. The preliminary results presented in this paper show that numerical ranks with the directional method, do not increase when moving up the hierarchical tree, for analyzing electrically large structures. The directional method shows promise for the fast computation of a factorized version of the reduced impedance matrix, for analysis of very large antenna arrays. This is work in progress.

## REFERENCES

- [1] S. Torchinsky, J. Broderick, A. Gunst, A. Faulkner, and W. van Cappellen, ‘‘SKA aperture array mid frequency science requirements,’’ *arXiv preprint arXiv:1610.00683*, 2016.
- [2] SKA website. [Online]. Available: <https://www.skatelescope.org/>
- [3] E. Suter and J. R. Mosig, ‘‘A subdomain multilevel approach for the efficient MoM analysis of large planar antennas,’’ *Microwave and Optical Technology Letters*, vol. 26, no. 4, pp. 270–277, 2000.
- [4] R. Maaskant, R. Mittra, and A. Tijhuis, ‘‘Fast analysis of large antenna arrays using the characteristic basis function method and the adaptive cross approximation algorithm,’’ *IEEE Transactions on Antennas and Propagation*, vol. 56, no. 11, pp. 3440–3451, 2008.
- [5] H. Bui-Van, J. Abraham, M. Arts, Q. Gueuning, C. Raucy, D. Gonzalez-Ovejero, E. de Lera Acedo, and C. Craeye, ‘‘Fast and accurate simulation technique for large irregular arrays,’’ *IEEE Transactions on Antennas and Propagation*, vol. 66, no. 4, pp. 1805–1817, 2018.
- [6] E. de Lera Acedo, N. Razavi-Ghods, N. Troop, N. Drought, and A. Faulkner, ‘‘SKALA, a log-periodic array antenna for the SKA-low instrument: design, simulations, tests and system considerations,’’ *Experimental Astronomy*, vol. 39, no. 3, pp. 567–594, 2015.
- [7] D. J. Ludick, R. Maaskant, D. B. Davidson, U. Jakobus, R. Mittra, and D. de Villiers, ‘‘Efficient analysis of large aperiodic antenna arrays using the domain Green’s function method,’’ *IEEE Transactions on Antennas and Propagation*, vol. 62, no. 4, pp. 1579–1588, 2014.
- [8] V. Prakash and R. Mittra, ‘‘Characteristic basis function method: A new technique for efficient solution of method of moments matrix equations,’’ *Microwave and Optical Technology Letters*, vol. 36, no. 2, pp. 95–100, 2003.
- [9] X. Chen, C. Gu, Z. Li, and Z. Niu, ‘‘Accelerated direct solution of electromagnetic scattering via characteristic basis function method with Sherman-Morrison-Woodbury formula-based algorithm,’’ *IEEE Trans. Antennas Propag.*, vol. 64, no. 10, pp. 4482–4486, 2016.

- [10] X. Fang, Q. Cao, Y. Zhou, and Y. Wang, "Multiscale compressed and spliced Sherman–Morrison–Woodbury algorithm with characteristic basis function method," *IEEE Transactions on Electromagnetic Compatibility*, vol. 60, no. 3, pp. 716–724, 2018.
- [11] —, "Multiscale compressed block decomposition method with characteristic basis function method and fast adaptive cross approximation," *IEEE Transactions on Electromagnetic Compatibility*, vol. 61, no. 1, pp. 191–199, 2018.
- [12] K. Zhao, M. N. Vouvakis, and J.-F. Lee, "The adaptive cross approximation algorithm for accelerated method of moments computations of emc problems," *IEEE transactions on electromagnetic compatibility*, vol. 47, no. 4, pp. 763–773, 2005.
- [13] M. Bebendorf and R. Venn, "Constructing nested bases approximations from the entries of non-local operators," *Numerische Mathematik*, vol. 121, no. 4, pp. 609–635, 2012.
- [14] W. Hackbusch and S. Börm, " $\mathcal{H}^2$ -matrix approximation of integral operators by interpolation," *Applied numerical mathematics*, vol. 43, no. 1-2, pp. 129–143, 2002.
- [15] B. Engquist and L. Ying, "Fast directional multilevel algorithms for oscillatory kernels," *SIAM Journal on Scientific Computing*, vol. 29, no. 4, pp. 1710–1737, 2007.
- [16] M. Messner, M. Schanz, and E. Darve, "Fast directional multilevel summation for oscillatory kernels based on chebyshev interpolation," *Journal of Computational Physics*, vol. 231, no. 4, pp. 1175–1196, 2012.
- [17] M. Bebendorf, C. Kuske, and R. Venn, "Wideband nested cross approximation for Helmholtz problems," *Numerische Mathematik*, vol. 130, no. 1, pp. 1–34, 2015.
- [18] 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 Transactions on Antennas and Propagation*, vol. 63, no. 5, pp. 2122–2134, 2015.
- [19] S. Börm, "Directional  $\mathcal{H}^2$ -matrix compression for high-frequency problems," *arXiv preprint arXiv:1510.07087*, 2015.

## Appendix D

### Conference paper — Reaction Matrix Computation with DCA [4]

K. Sewraj and M. M. Botha, “Directional Method to Compute Reduced Matrix System in MBF Solvers,” in *Proceedings of the European Conference on Antennas and Propagation (EuCAP)*, Copenhagen, Denmark, March 2020, pp. 1–5.

# Directional Method to Compute Reduced Matrix System in MBF Solvers

Keshav Sewraj\*, Matthys M. Botha\*

\*Department of Electrical and Electronic Engineering, Stellenbosch University, Stellenbosch, South Africa, 20635958@sun.ac.za, mmbbotha@sun.ac.za

**Abstract**—Computation of reaction terms in macro basis function (MBF) solvers is computationally very expensive. A directional cross approximation technique is used in this paper to compute the reaction terms, which is a multilevel low-rank compression technique. The motivation to use a directional method is to keep the compression rank bounded irrespective of subdomain size, so as a fast algorithm can be ensured. Apart from the computational efficiency, a controlled and very good accuracy can be obtained with the directional method. The application focus of this work is large antenna arrays.

**Index Terms**—Antenna array, directional cross approximation, electric field integral equation (EFIE), method of moments (MoM).

## I. INTRODUCTION

Integral-equation based methods such as the Method of Moments (MoM) are very well suited for the analysis of radiation problems, including antenna arrays. However, the memory and computational cost scale as  $\mathcal{O}(N^2)$  and  $\mathcal{O}(N^3)$  respectively, where  $N$  is the number of basis functions (typically RWG [1] basis functions for 3D electromagnetic problems). Hence, the analysis of large antenna arrays using the MoM is computationally prohibitive.

Several acceleration techniques have been designed to be used on top of the MoM, and most of these techniques are in the realm of iterative solvers. For instance, among the most successful techniques is the Multilevel Fast Multipole Method (MLFMM) [2] to perform fast matrix-vector products in a Krylov subspace solver. However, for antenna arrays, direct methods are often preferred since analysis needs to be performed for multiple excitation schemes, such as when computing Embedded Element Patterns (EEPs).

Macro Basis Function (MBF) [3] techniques have been proven to be very effective for antenna array analysis. In these techniques physics-based functions (macro basis functions) are used to create a reduced impedance matrix system, which can be solved directly. However, the process of creating the reduced system is still very costly since every MoM impedance matrix sub-block (representing interactions between 2 antennas) needs to be pre- and post-multiplied by MBF groups. The acceleration of this step in MBF solvers has been an active area of research. Various acceleration techniques have been proposed such as using the Adaptive Cross Approximation (ACA) [4] in a Characteristic Basis Function Method (CBFM) [5] solver for a regular antenna array [6]. For irregular arrays,

in [7], a Harmonic-polynomial (HARP) model was created and used to accelerate the reduced impedance matrix fill-ins.

This paper is an extension of [8] with results regarding the accuracy of solving the reduced matrix system. In [8] and this paper, the directional cross approximation technique [9] is used to rapidly compute the reaction terms in MBF solvers for large antenna arrays. The directional cross approximation is a fast multilevel algorithm for compression, and ensures constant rank irrespective of the size of the subdomain.

The rest of the paper is structured as follows. Section II reviews the directional cross approximation method. In Section III, the MBF method is reviewed and the use of directional cross approximation in an MBF solver is discussed. Numerical results are presented in Section IV, and Section V concludes the paper. The implementation in this paper is in the context of the electric field integral equation (EFIE), RWG MoM.

## II. DIRECTIONAL CROSS APPROXIMATION

The directional cross approximation compresses low-ranked matrix blocks (e.g.  $Z_{pq}$  in (1)) in the MoM impedance matrix as:

$$Z_{pq} \approx U_p^e S_{pq} V_q^{-e}. \quad (1)$$

$Z_{pq}$  is an  $N_i \times N_i$  matrix, where  $N_i$  is the number of basis and testing functions in cluster  $p$  and  $q$  respectively.  $U_p^e$  and  $V_q^{-e}$  are the directional cluster bases with sizes  $N_i \times R$  and  $R \times N_i$  respectively, where  $R$  is the number of columns equivalent to the numerical rank according to a pre-determined error threshold and  $R \ll N_i$ . For simplicity, the rank of the two clusters are assumed identical. The superscript ‘ $e$ ’ in the cluster bases ( $U$  and  $V$ ) refers to the direction ‘ $e$ ’.  $S_{pq}$  is the coupling matrix relating the clusters  $p$  and  $q$  with size  $R \times R$ .

The directional cross approximation has a directional  $\mathcal{H}^2$ -matrix structure, meaning:

- The antenna array is partitioned hierarchically into clusters at different levels, as shown in Figure 1. Level 0 is the entire array and leaf level clusters (finest level) contain only one antenna element. At each level an admissibility criterion test is performed to verify if the interaction between pairs of source and observer clusters can be compressed.
- Nested structure, where compressed matrix factors at parent levels (larger clusters) are created using information

gathered from children clusters (smaller clusters), thus reducing the memory required.

- Directional partitioning of the far-field while computing the factors of the compressed matrix, to make sure the rank is constant irrespective of the cluster size at different levels. At the leaf level, the far-field is partitioned into 4 directions in this implementation.

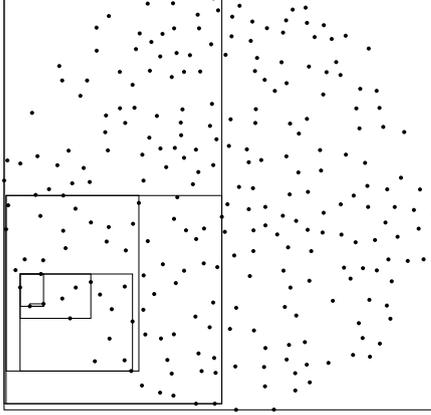


Fig. 1. Binary partitioning of an irregular antenna array containing 256 elements. Each dot represents the position of an antenna element. The radius of the circular array is  $15\lambda$ .

#### A. Admissibility Criterion

A distance and angular admissibility criterion need to be satisfied for a sub-block matrix to be admissible. For each source and observer cluster pair at every level, the distance-based criterion test in (2) is performed:

$$\eta \text{ dist}(p, q) \geq k \max(D_p^2, D_q^2), \quad (2)$$

where  $\eta$  is an admissibility constant,  $k$  is the wavenumber and  $D_p$  and  $D_q$  are the diameters of the observer and source subdomains  $p$  and  $q$  respectively. Further to the distance criterion, a field-of-view angular criterion is also given as  $\mathcal{O}\left(\frac{1}{kD}\right)$  (shown in Figure 2) where  $D$  is the diameter of the cluster. Thus, as the size of the cluster and the frequency increases, the directional far-field decreases.

#### B. Constructing Cluster Bases and Coupling Matrices

At the leaf level, directional cluster bases  $U^e$  are created for every observer cluster and direction, through an inverse source process as follows:

$$U^e = Z_{t\sigma_t^e} (Z_{\tau_t^e \sigma_t^e})^{-1}, \quad (3)$$

where  $t$  refers to the set of RWG functions on the cluster,  $\tau_t$  and  $\sigma_t$  are subsets of RWG functions on the cluster and directional far-field respectively. Similarly, for every source cluster and direction,  $V^e$  cluster bases can be defined as:

$$V^e = (Z_{\tau_s^e \sigma_s^e})^{-1} Z_{\tau_s^e \sigma_s^e}, \quad (4)$$

where  $s$  refers to the set of RWG functions on the source cluster,  $\sigma_s$  and  $\tau_s$  are subsets of RWG functions on the cluster and directional far-field respectively. The process of creating directional cluster bases is depicted in Figure 2. The subsets of RWG functions on the cluster and directional far-field ( $\tau_t, \sigma_t, \tau_s$  and  $\sigma_s$ ) are selected through a process which includes a pre-selection sampling and ACA. The details of the selection are described in [10]. The coupling matrix,  $S$  in (1), is expressed as

$$S = Z_{\tau_t \sigma_s}. \quad (5)$$

For non-leaf levels, transfer matrices ( $B$  and  $C$  for observer and source clusters respectively) are used to interpolate or antepolate from their children clusters. The  $B$  matrix can be obtained as

$$B^e = Z_{\tau_t^e \sigma_t^e} (Z_{\tau_t^e \sigma_t^e})^{-1}, \quad (6)$$

where  $c$  means child cluster in (6). Similarly, the transfer matrix,  $C$ , on the observer side can be constructed as

$$C^e = (Z_{\tau_s^e \sigma_s^e})^{-1} Z_{\tau_s^e \sigma_s^e}. \quad (7)$$

The interaction between two antennas ( $p_c$  and  $q_c$ ) using a 2-level directional cross approximation can thus be written as:

$$Z_{p_c q_c} \approx U_{p_c}^{e_c} B_{p_c, p}^e S_{pq} C_{q, q_c}^{-e_c} V_{q_c}^{-e_c}, \quad (8)$$

where  $p_c$  and  $q_c$  are the children clusters of the  $p$  and  $q$  respectively. And,  $e_c$  refers to the direction of the child cluster.

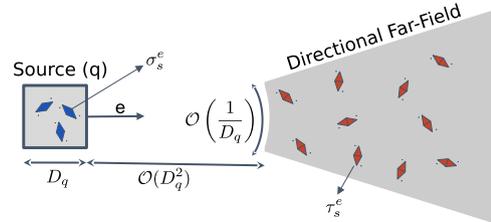


Fig. 2. Inverse source process to obtain directional cluster bases.

### III. MBF SOLVER WITH DIRECTIONAL CROSS APPROXIMATION

As mentioned in Section I, the first step of an MBF solver is to generate MBFs. Similar to CBFM, up to tertiary MBFs [11] are generated in this implementation since accurate results were not obtained using only up to secondary MBFs when the antennas are closely situated in the array. Primary MBF,  $I_{\text{Prim}}$ , is obtained as

$$I_{\text{Prim}} = Z_{qq}^{-1} V_q, \quad (9)$$

$Z_{qq}$  is the self-interacting MoM sub-block for antenna (cluster at leaf level)  $q$ , and  $V_q$  is the excitation vector for cluster  $q$ . Secondary MBFs are obtained by allowing primary MBFs to

radiate and induce a surface current on neighbouring antennas such as

$$I_{\text{Sec}} = -Z_{pp}^{-1} Z_{pq} I_{\text{Prim}}. \quad (10)$$

Tertiary MBFs are obtained similarly by allowing secondary MBFs to radiate onto neighbouring antennas. The MBFs generated for each antenna are concatenated in a column augmented vector, after which a Singular Value Decomposition (SVD) operation is performed. Only MBFs with singular values greater than a pre-defined threshold are kept. In this implementation, an SVD threshold of  $10^{-5}$  has been used. The new set of MBFs for cluster  $p$  is denoted as  $J_p$ .

A sub-block of the reduced impedance matrix,  $Z_{pq}^{\text{Red}}$ , can then be obtained as in (11).

$$Z_{pq}^{\text{Red}} \approx \langle J_p^H, Z_{pq} J_q \rangle. \quad (11)$$

Compressing the sub-block  $Z_{pq}$  using single level directional cross approximation, (11) can be re-written as:

$$Z_{pq}^{\text{Red}} \approx \langle J_p^H U_p^e, S_{pq} V_q^{-e} J_q \rangle. \quad (12)$$

The product of the MBF definition and cluster bases can be computed on the fly while generating the cluster bases. For non-leaf levels, the product of the MBF definition, cluster basis and transfer matrix for each child at leaf level is computed and concatenated. Sub-blocks of the reduced impedance matrix is computed for every admissible interaction pair at every level. Equation (11) is used to compute the reduced impedance sub-blocks for non-admissible interactions, that is, self and neighbouring interactions.

#### IV. NUMERICAL RESULTS

An irregular antenna array configuration such as in Figure 1 will be used in this paper, however, the number of antennas in the array will be varied. The mesh of the antenna used in this paper is shown in Figure 3, with the dimensions stated in the caption. The antenna is discretized with 363 RWG functions at 750 MHz.

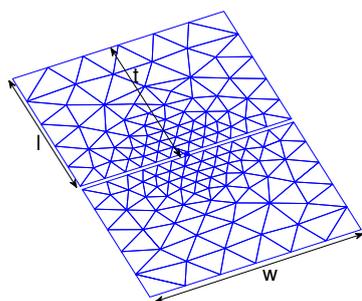


Fig. 3. Mesh of blade antenna at 750 MHz, discretized with 363 RWG functions. Dimensions:  $t = 11.65$  cm,  $w = 14.13$  cm and  $l = 11.49$  cm.

#### A. Accuracy

The current coefficients of an antenna array of 16 elements, solved by a MoM solver, is shown in Figure 4 (blue plot). The current coefficient errors (relative to the MoM solution) obtained by an (i) MBF solver and (ii) MBF solver with directional cross approximation, are also plotted in Figure 4. A very accurate MBF solver is used with an average of 17 MBFs per antenna (up to tertiary MBFs and SVD threshold:  $10^{-5}$ ), since the emphasis is on the additional error due to the directional cross approximation. The error norm given by (13) is  $4.459 \times 10^{-4}$ , when the average rank for the directional cross approximation (number of columns in cluster bases) is 16. The far-field electric fields obtained using (i) MoM, (ii) MBF and (iii) MBF with directional cross approximation are plotted in Figure 5, and excellent agreement among the three traces can be observed. The far-field electric field of an antenna array of 64 elements is also plotted in Figure 6, excluding the MoM solution since the problem is too large (23, 232 RWGs) to be solved directly. A 2-level directional cross approximation is used in this setup, and an excellent agreement with the MBF solver solution can be seen in Figure 6.

$$\text{Current Error Norm} = \frac{\|I_{\text{MBF}} - I_{\text{MBF-Dir. Cross. Approx}}\|_2}{\|I_{\text{MBF}}\|_2}. \quad (13)$$

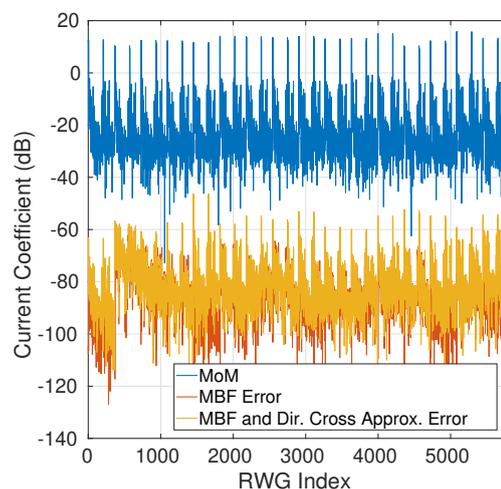


Fig. 4. Current coefficients of 16 element antenna array using the MoM. Current coefficient error (relative to the MoM) using (i) MBF solver and (ii) MBF with directional cross approximation solver.

#### B. Rank and Memory Requirement for a Large Array

In this section, the rank and memory requirement of an irregular array consisting of 256 antennas (92, 928 RWG functions) is presented. One of the motivations to use the directional cross approximation technique is to ensure the rank to remain bounded irrespective of the cluster size, provided the admissibility criteria are met. In Figure 7, the average

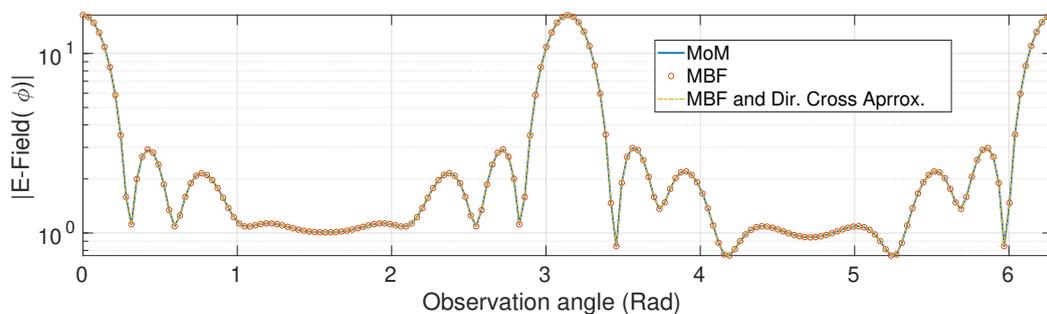


Fig. 5. Far-field electric field of an irregular antenna array consisting of 16 elements. All antennas are excited with uniform amplitude and in phase.

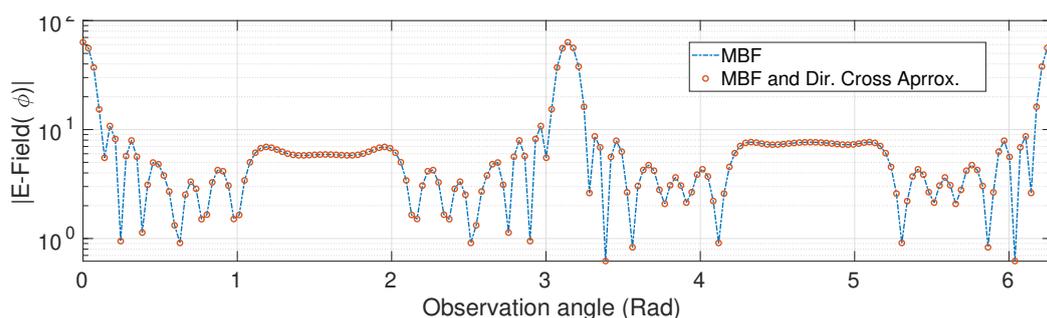


Fig. 6. Far-field electric field of an irregular antenna array consisting of 64 elements. All antennas are excited with uniform amplitude and in phase.

rank of cluster bases remains bounded as the size of the cluster increases. Level 3 is the leaf level, where a cluster is an antenna element and at Level 1, a cluster consists of 4 antenna elements. The ACA threshold of  $10^{-3}$  is used during the process of creating cluster bases (as mentioned in Section II) to control the accuracy and rank.

An alternative to the directional cross approximation, nested cross approximation [10] can be used for fast computation of reaction terms in MBF solvers. The nested cross approximation is omnidirectional and does not partition the far-field while creating the cluster bases. In [8], a comparison between nested and directional cross approximation was performed, and it was shown that the rank is not bounded when using the nested cross approximation.

The memory required to store directional cross approximation factors for an array of 256 antennas is shown in the second column of Table I. Taking advantage of reciprocity, only the observer cluster bases ( $U$ ) and transfer matrices ( $B$ ) are stored, and only half of the near interactions and coupling matrices are stored. Column 3 in Table I shows the memory after (i) pre- and post-multiplying the near interactions with MBF definitions as in (11); and (ii) product of MBF definitions and cluster bases as in (12).

## V. CONCLUSION

In this paper, a directional cross approximation method is used to compute the reaction terms in MBF solvers. Using the compressed factors, a reduced impedance matrix system

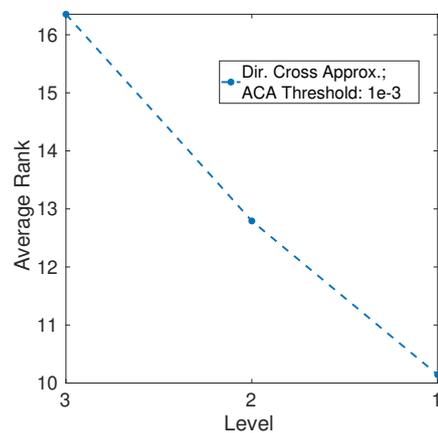


Fig. 7. Average rank of clusters. Level 3 refers to the leaf level where a cluster is an antenna. Level 1 refers to the level where a cluster contains 4 antennas

is created. Very good accuracy is observed for the current coefficients and far-field electric field using the directional cross approximation. The rank of compressed factors is bounded irrespective of the cluster size. The memory requirement to store the directional cross approximation factors for an array of 256 antennas (92,928 RWG functions) is presented.

TABLE I  
MEMORY REQUIREMENT FOR ANTENNA ARRAY OF 256 ELEMENTS.

	Before MBF Compression (MB)	After MBF Compression (MB)
<b>Near Interaction</b>	512.8	0.7
<b>Cluster Bases (U)</b>	87.7	3.4
<b>Transfer Matrices (B)</b>	5.5	5.5
<b>Coupling Matrices</b>	99.6	99.6
<b>Total</b>	705.6	109.2

#### ACKNOWLEDGMENT

This research was supported by the South African Radio Astronomy Observatory, which is a facility of the National Research Foundation, an agency of the Department of Science and Technology. This work was also supported in part, by the National Research Foundation of South Africa under Grants 75322 and 106033.

#### REFERENCES

- [1] S. Rao, D. Wilton, and A. Glisson, "Electromagnetic scattering by surfaces of arbitrary shape," *IEEE Transactions on Antennas and Propagation*, vol. 30, no. 3, pp. 409–418, 1982.
- [2] J. Song, C.-C. Lu, and W. C. Chew, "Multilevel fast multipole algorithm for electromagnetic scattering by large complex objects," *IEEE Transactions on Antennas and Propagation*, vol. 45, no. 10, pp. 1488–1493, 1997.
- [3] E. Suter and J. R. Mosig, "A subdomain multilevel approach for the efficient MoM analysis of large planar antennas," *Microwave and Optical Technology Letters*, vol. 26, no. 4, pp. 270–277, 2000.
- [4] K. Zhao, M. N. Vouvakis, and J.-F. Lee, "The adaptive cross approximation algorithm for accelerated method of moments computations of emc problems," *IEEE Transactions on Electromagnetic Compatibility*, vol. 47, no. 4, pp. 763–773, 2005.
- [5] V. Prakash and R. Mittra, "Characteristic basis function method: A new technique for efficient solution of method of moments matrix equations," *Microwave and Optical Technology Letters*, vol. 36, no. 2, pp. 95–100, 2003.
- [6] R. Maaskant, R. Mittra, and A. Tjihuis, "Fast analysis of large antenna arrays using the characteristic basis function method and the adaptive cross approximation algorithm," *IEEE Transactions on Antennas and Propagation*, vol. 56, no. 11, pp. 3440–3451, 2008.
- [7] H. Bui-Van, J. Abraham, M. Arts, Q. Gueuning, C. Raucy, D. Gonzalez-Ovejero, E. de Lera Acedo, and C. Craeye, "Fast and accurate simulation technique for large irregular arrays," *IEEE Transactions on Antennas and Propagation*, vol. 66, no. 4, pp. 1805–1817, 2018.
- [8] K. Sewraj and M. M. Botha, "Computation of MBF reaction matrices for antenna array analysis, with a directional method," in *Proceedings of the International Conference on Electromagnetics in Advanced Applications (ICEAA '19)*, Sept 2019, pp. 1–5, Granada, Spain.
- [9] M. Bebendorf, C. Kuske, and R. Venn, "Wideband nested cross approximation for Helmholtz problems," *Numerische Mathematik*, vol. 130, no. 1, pp. 1–34, 2015.
- [10] M. Bebendorf and R. Venn, "Constructing nested bases approximations from the entries of non-local operators," *Numerische Mathematik*, vol. 121, no. 4, pp. 609–635, 2012.
- [11] S. Hay, J. O'Sullivan, and R. Mittra, "Analysis of connected patch arrays using the characteristic basis function method," in *IEEE Antennas and Propagation Society International Symposium*, 2008, pp. 1–4.

## Appendix E

### Conference paper — Non-Directional IFMM with MBFs [5]

K. Sewraj and M. M. Botha, “Application of the Inverse Fast Multipole Method to Antenna Array Analysis,” in *Proceedings of IEEE International Symposium on Antennas and Propagation (IEEE AP-S)*, Montréal, Canada, July 2020, pp. 1993–1994.

# Application of the Inverse Fast Multipole Method to Antenna Array Analysis

Keshav Sewraj and Matthys M. Botha  
 Department of Electrical and Electronic Engineering  
 Stellenbosch University  
 Stellenbosch, South Africa  
 20635958@sun.ac.za, mmbbotha@sun.ac.za

**Abstract**—The analysis of antenna arrays using the Method of Moments (MoM) is limited by the electrical size due to the high computational cost associated with the MoM. Fast numerical techniques such as  $\mathcal{H}^2$ -matrices with iterative solvers can be used to solve for larger arrays efficiently. However, direct solvers are preferred for array analysis due to the need to solve for many different excitation schemes. The Inverse Fast Multipole Method (IFMM), which provides a direct solution for an  $\mathcal{H}^2$ -matrix formulation, is proposed for antenna array analysis.

## I. INTRODUCTION

The analysis of large antenna arrays with the Method of Moments (MoM) is computationally expensive. Thus, fast numerical techniques based on the MoM need to be used. One such technique is  $\mathcal{H}^2$ -matrices [1], which can be thought of as an algebraic version of the fast multipole method. In this technique the entire structure is hierarchically partitioned and interactions by far subdomains at every level are factored into low-rank representations. Furthermore, the  $\mathcal{H}^2$ -matrix has a nested structure meaning that factors of a parent subdomain (at a higher-level) can be built from that of its children subdomains (at a lower-level), hence leading to a more efficient memory storage scaling. The product of an  $\mathcal{H}^2$ -matrix with a vector can be performed in linear time complexity for smooth kernels, hence the  $\mathcal{H}^2$ -matrix technique finds its application in fast matrix-vector products in iterative solvers. The Inverse Fast Multipole Method (IFMM) was proposed in [2] as a direct solver based on the  $\mathcal{H}^2$ -matrix formulation, and used in [3] to construct preconditioners in iterative solvers. The IFMM makes use of an extended sparse matrix formulation, followed by compression and redirecting compressible fill-ins during the Gaussian elimination process. In this paper a single-level cross approximation method [4] is used to generate the  $\mathcal{H}^2$ -matrix factors for antenna array problems and the IFMM will be used to solve the problem once the cross approximation factors are available.

## II. SINGLE-LEVEL CROSS APPROXIMATION

For subdomain interactions to be compressible, the admissibility condition in (1) needs to be met:

$$\eta \text{ dist}(i, j) \geq \max(D_i, D_j), \quad (1)$$

where  $\text{dist}(i, j)$  is the minimum distance between subdomain  $i$  and  $j$ .  $D_i$  and  $D_j$  are the size of subdomains  $i$  and  $j$ ,

respectively and  $\eta$  is a parameter set to a value of 1 in this paper. For each antenna, observer and source cluster bases,  $U$  and  $V$  respectively, are defined as in (2) and (3).

$$U = Z_{t\sigma_t}(Z_{\tau_t\sigma_t})^{-1} \quad (2)$$

$$V = (Z_{\tau_s\sigma_s})^{-1}Z_{\tau_s s}. \quad (3)$$

In (2) and (3),  $t$  and  $s$  refer to all the basis functions (Rao-Wilson-Glisson (RWG) [5], solving the electric field integral equation) on the observer and source domains, respectively. Indices  $\tau_t$  and  $\sigma_t$  refer to a few selected RWGs on the observer's local and far-field domains, respectively. Similarly,  $\sigma_s$  and  $\tau_s$  refer to a few selected RWGs on the source's local and far-field domains, respectively. The details of how to obtain the sampled RWGs in the local and far-field are found in [4]. For every compressible interaction, coupling matrix  $S$  is defined such as

$$S = Z_{\tau_t\sigma_s}. \quad (4)$$

The interaction between two admissible subdomains is then given as

$$Z_{ij} = U_i S_{ij} V_j. \quad (5)$$

## III. SOLVING THE MATRIX SYSTEM

After obtaining the factors from the cross approximation stage, an extended sparse matrix (described in (6)) is constructed, upon which Gaussian elimination is performed to obtain the solution.  $N$  in (6) is the sub-matrix containing the self- and near-interactions. The  $N$  sub-blocks are MoM solutions for non-compressible interactions according to (1).  $U$  and  $V$  are the cluster bases as given in (2) and (3).  $S$  contains the coupling matrices given in (4). The block  $-I$  contains negative identity matrices. In (6),  $x$  is the unknown current coefficient that needs to be solved for;  $z$  and  $y$  are auxiliary variables. On the right-hand side,  $v$  is the excitation vector.

$$\begin{bmatrix} N & U & \\ V & & -I \\ & -I & S \end{bmatrix} \begin{bmatrix} x \\ z \\ y \end{bmatrix} = \begin{bmatrix} v \\ 0 \\ 0 \end{bmatrix} \quad (6)$$

After a block row and column re-ordering in (6), the extended sparse matrix structure resulting from a planar irregular antenna array of 16 elements is shown in Figure 1 (left).

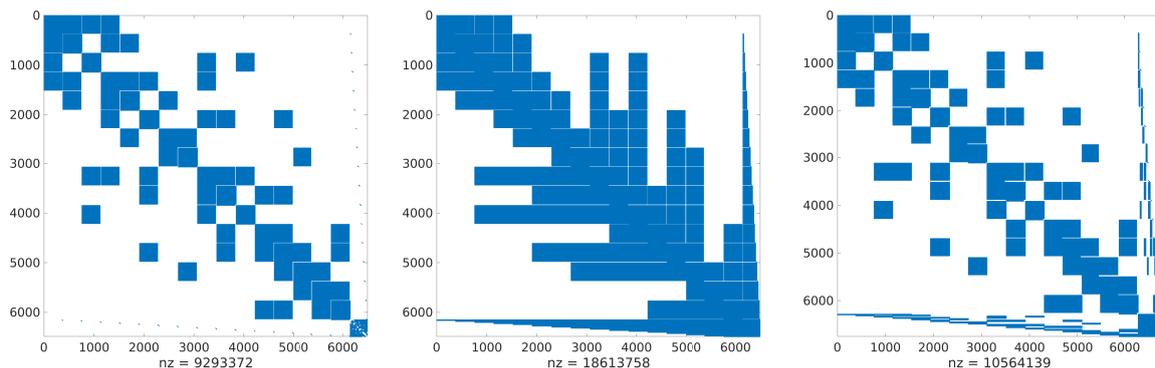


Fig. 1: Extended sparse matrix. Left: Before Gaussian elimination. Middle: After Gaussian elimination, with fill-ins. Right: After Gaussian elimination, fill-ins are eliminated and redirected. The number of non-zero entries in the sparse matrix is denoted as  $nz$ .

The row and column re-ordering is performed since the  $x$  and  $z$  variables for a subdomain are first eliminated during the Gaussian elimination process before moving to the next subdomain. Performing Gaussian elimination on the extended sparse matrix in Figure 1 (left) will lead to undesirable fill-ins as shown in Figure 1 (middle). Instead, as described in [2] and [3], during the elimination of each variable, compressible fill-ins are eliminated, compressed and re-directed towards existing compressible paths. The extended sparse matrix after the IFMM process is shown in Figure 1 (right). The elimination of the fill-ins is important to preserve the sparsity of the matrix and scalability of solving the problem.

#### A. Accuracy of IFMM

The electric far-field of the 16-element array with uniform excitation is computed using both the standard MoM and IFMM solver, as shown in Figure 2. Very good agreement between the two methods can be observed, even after the elimination and redirection of compressible fill-ins.

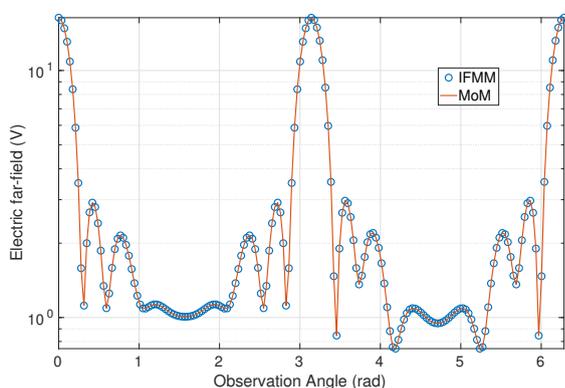


Fig. 2: Electric far-field: Comparison between a standard MoM and IFMM solver.

#### IV. CONCLUSION

In this paper, the single-level cross approximation method (single-level  $\mathcal{H}^2$ -matrix formulation) is used to describe antenna array problems. Instead of using the cross approximation in an iterative solver, the IFMM is proposed to be used for a direct solution. The IFMM provides accurate results when compared to the MoM. A single-level cross approximation method is used in this paper since an oscillatory Green's function is used in the formulation of the problem. Using a multi-level  $\mathcal{H}^2$ -matrix formulation, the rank of the subdomains increases with the size of the subdomain as discussed in [6]. A non-bounded rank is harmful for the complexity of the solver. Further work is required in this area.

#### ACKNOWLEDGEMENT

This work was supported by the South African Radio Astronomy Observatory (SARAO, [www.ska.ac.za](http://www.ska.ac.za)) and the National Research Foundation under Grants 75322 and 106033.

#### REFERENCES

- [1] W. Hackbusch and S. Börm, " $\mathcal{H}^2$ -matrix approximation of integral operators by interpolation," *Applied Numerical Mathematics*, vol. 43, no. 1-2, pp. 129–143, 2002.
- [2] S. Ambikasaran and E. Darve, "The inverse fast multipole method," *arXiv preprint arXiv:1407.1572*, 2014.
- [3] P. Coulier, H. Pouransari, and E. Darve, "The inverse fast multipole method: using a fast approximate direct solver as a preconditioner for dense linear systems," *SIAM Journal on Scientific Computing*, vol. 39, no. 3, pp. A761–A796, 2017.
- [4] M. Bebendorf and R. Venn, "Constructing nested bases approximations from the entries of non-local operators," *Numerische Mathematik*, vol. 121, no. 4, pp. 609–635, 2012.
- [5] S. Rao, D. Wilton, and A. Glisson, "Electromagnetic scattering by surfaces of arbitrary shape," *IEEE Transactions on antennas and propagation*, vol. 30, no. 3, pp. 409–418, 1982.
- [6] K. Sewraj and M. M. Botha, "Computation of MBF reaction matrices for antenna array analysis, with a directional method," in *Proceedings of the International Conference on Electromagnetics in Advanced Applications (ICEAA '19)*, Sept 2019, pp. 385–389, Granada, Spain.



## Appendix F

### Conference paper — Overview of Group's Results [6]

K. Sewraj, M. Chose, and M. M. Botha, “Developments in the Analysis of Large Antenna Arrays with Disjoint Elements,” *Proceedings of the European Conference on Antennas and Propagation (EuCAP)*, Dusseldorf, Germany, March 2021, submitted for publication.

# Developments in the Analysis of Large Antenna Arrays with Disjoint Elements

Keshav Sewraj\*, Matthews Chose\*, Matthys M. Botha\*

\*Department of Electrical & Electronic Engineering, Stellenbosch University, Stellenbosch, South Africa, [mbotha@sun.ac.za](mailto:mbotha@sun.ac.za)

**Abstract**—Efficient electromagnetic analysis of very large antenna arrays is important for various applications, including e.g., radio astronomy. The method of moments (MoM) is generally the most suitable computational electromagnetics method to deal with such open-region problems. However, its computational cost grows rapidly with problem size. This paper presents a brief overview of past work as well as the latest developments by the authors, on devising fast MoM-based methods for arrays with identical disjoint elements and arbitrary layouts. Two different lines of development are reported on, namely methods aimed at efficiently solving the full MoM system through the use of macro basis functions and adaptive cross approximation (ACA) acceleration, and a fast-converging iterative method involving localised solutions.

**Index Terms**—Adaptive cross approximation (ACA), domain decomposition, macro basis function (MBF), sparse direct solver.

## I. INTRODUCTION

Efficient electromagnetic analysis of very large antenna arrays is important for various applications, including radio astronomy. For example, the Square Kilometre Array Mid-Frequency Aperture Array (MFAA) is envisaged to consist of thousands of elements with dense and/or sparse layout [1]. During the design of such arrays, many full-wave electromagnetic analyses of candidate designs are required. The method of moments (MoM) is generally the most suitable computational electromagnetics method to deal with such open-region problems. However, its computational cost grows rapidly with problem size. Therefore acceleration schemes are required to increase its efficiency to meet the challenging design requirements. This paper presents a brief overview of past work as well as the latest developments by the authors, on devising such methods for arrays with identical disjoint elements and arbitrary layouts. Two different lines of development are reported on, namely methods aimed at efficiently solving the full MoM system and a method of decomposing the full MoM problem into a set of smaller problems. Further progress will be reported at the symposium.

## II. FAST SOLUTIONS BY COMPRESSING AND REDUCING THE GLOBAL SYSTEM

The storage requirement for the full MoM matrix can be dramatically reduced by using low-rank compression schemes,

The financial assistance of the South African Radio Astronomy Observatory (SARAO) towards this research is hereby acknowledged ([www.ska.ac.za](http://www.ska.ac.za)). This work was also supported in part, by the National Research Foundation of South Africa under Grant 75322.

such as the MLFMM [2] and adaptive cross approximation (ACA) [3], [4]. The dimension of the system matrix can also be reduced by making use of macro basis functions (MBFs), which are well-suited to array analysis [5]. MBFs are fixed linear combinations of the low-level basis functions. The combination of ACA-based compression and MBF-based reduction is discussed in this section.

### A. Macro Basis Functions

Various schemes exist for generating MBFs for antenna arrays, such as the characteristic basis function method (CBFM) [6], the sub-entire-domain basis function method [7] (the “windowed MBF” scheme proposed in [8] is very similar), and iteratively refined MBFs [9]. In the results to follow, the CBFM is used.

### B. Fast Construction of the MBF-Reduced System Matrix

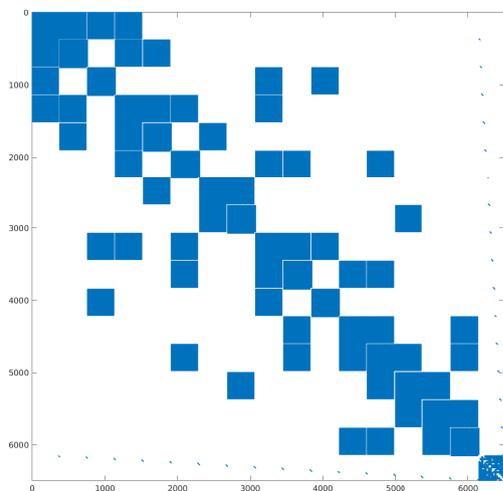
When using MBFs, the reduced matrix is often small enough to be fully stored and solved directly. To be efficient, this approach requires a low-rank factorisation of the original MoM matrix which is amenable to being multiplied out to reconstitute the reduced matrix. This is done by using the nested, directional, multilevel ACA [10], [11], with preliminary results having been presented in [12].

### C. Sparse Direct Solver for the Factorised MBF-Reduced System Matrix

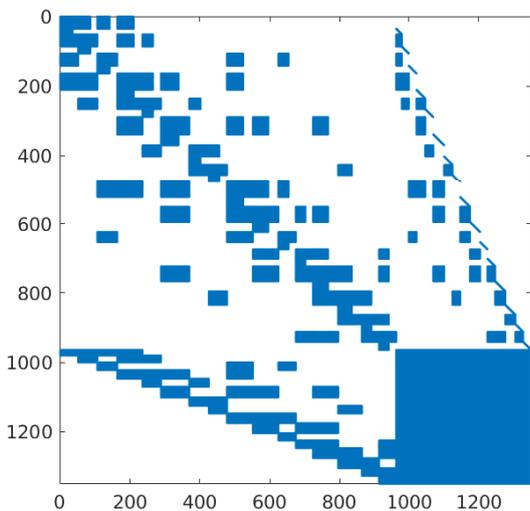
For sufficiently large arrays, the MBF-reduced matrix becomes too large for full storage and a direct solution. A new sparse direct solver based on the IFMM [13] was developed for the ACA-factorised MoM matrix [14]. It is based on redirecting fill-ins during the LU decomposition process. This solver is currently being extended to the MBF case, and a preliminary result is shown next.

### D. Results

The same small test problem as in [14] is considered, namely an array of 16 blade dipoles. Each element has 363 Rao-Wilton-Glisson basis functions for a global total of 4,356 degrees of freedom. The extended sparse matrix representation of the MoM matrix in terms of the ACA factors is shown in Figure 1a. The sparse direct solver in combination with the use of the MBFs, yields as end result the LU decomposition of the extended sparse MBF-MoM matrix shown in Figure 1b. Note that the full MoM matrix has 18,974,736 non-zero



(a) Extended sparse matrix representation of the MoM matrix in terms of the ACA factors: NNZ = 9, 293, 372.



(b) Corresponding sparse LU-decomposed MBF-MoM matrix: NNZ = 387, 306.

Fig. 1. Non-zero matrix entries for the ACA-based sparse direct MBF-MoM solver applied to a small test array.

entries, while its extended sparse version has 9, 293, 372 non-zero entries, with the final sparse LU-decomposed MBF-MoM matrix only having 387, 306 non-zero entries. In this small test, these differences are not very dramatic, but they should become much greater as the problem size is increased.

### III. FAST SOLUTIONS BY DECOMPOSING THE GLOBAL SYSTEM

Another approach to solving large problems is to break them up into a collection of smaller problems. One such method is the domain Green's function method (DGFM), which is applicable to arrays with disjoint, identical array elements.

The standard DGFM is taken as a starting point and further developed into an iterative scheme, such that the solution error is controllable. To enable the scheme's practical use for very large arrays, a tailor-made ACA acceleration scheme is added.

#### A. Standard DGFM

The standard DGFM requires solving a small MoM problem for the current distribution on each individual antenna [15], [16]. The effects of the other elements in the array are incorporated into the local solution via the DGFM approximation. It assumes that the current on each of the other elements can be expressed as a scaled version of that being solved for, with the scaling constant being the ratio of the excitation voltages.

#### B. Iterative Radius-Based DGFM (IRB-DGFM)

The drawback of the standard DGFM is that it can be quite inaccurate, due to rapid variations in the shape of elemental current distributions, e.g., at array edges. Furthermore, the error relative to the true global MoM solution cannot be controlled. To remedy these issues the IRB-DGFM is being developed, with progress reported in [17], [18].

When solving the current on a specific element, a local radius is introduced around it, with the effect of those elements being taken into account via rigorous MoM analysis. Only the effects of the elements external to this local neighbourhood are accounted for via the DGFM assumption. This radius-based formulation naturally increases the accuracy of the DGFM, as in the zero radius limit it yields the standard DGFM, while in the infinite radius limit it yields the rigorous MoM.

Incorporating these finite local domains does however not solve the problem of uncontrolled errors in the results. This issue is addressed by expanding the method into an iterative scheme, where the previous solution is used to improve the scaling coefficients used in the DGFM parts of the radius-based method. This results in the IRB-DGFM, which can converge to a solution within any specified error tolerance to the true MoM solution.

#### C. Acceleration of the IRB-DGFM

An ACA-based factorisation scheme tailored to the requirements of the IRB-DGFM is being developed. The full details are beyond the scope of the present paper, but it allows for fast reconstitution of the local system matrices from the stored ACA factors.

#### D. Results

Figure 2 shows a mesh representing  $49 \times 49 = 2401$  bow-tie antenna elements, each element has 171 Rao-Wilton-Glisson basis functions for a global total of 410, 571 degrees of freedom. The accelerated IRB-DGFM reduces the storage requirement to 0.7% of the full MoM requirement. This is a preliminary figure which might be further reduced with fine-tuning of the compression scheme's parameters. Figure 3 shows the normalised solution residual ( $\|b - Ax\|/\|b\|$ ) convergence history for various local radii. The array is uniformly excited. It can be seen that the iterative scheme

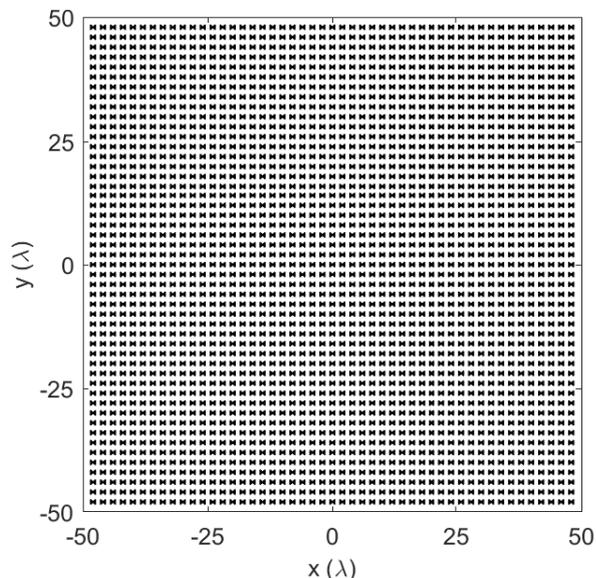


Fig. 2. Antenna array with 2401 bow-tie elements, spaced one wavelength ( $\lambda$ ) apart.

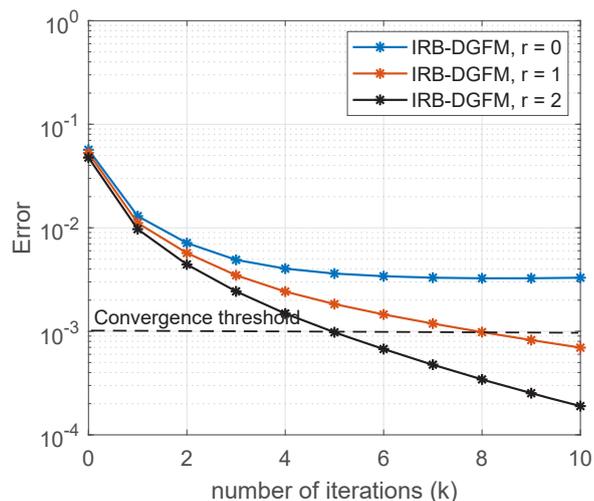


Fig. 3. Accelerated IRB-DGFM residual error convergence history for local radii of  $0$ ,  $\lambda$  and  $2\lambda$ .

converges rapidly for a sufficiently large local radius. The indicated convergence threshold is the ACA compression tolerance ( $10^{-3}$ ).

#### IV. CONCLUSION

Developments are presented on two types of specialised methods for efficient MoM analysis of large arrays consisting of disjoint elements. The sparse direct MBF-MoM solver must still be tested on larger problems and with different MBF schemes. It could potentially be very useful for analysing a

given array with various excitation configurations. The full potential of the accelerated IRB-DGFM method must still be realised through fine-tuning of its user-specified parameters. Its fast convergence is a significant benefit over conventional iterative solution methods.

#### REFERENCES

- [1] S. Torchinsky, J. Broderick, A. Gunst, A. Faulkner, and W. van Cappellen, "SKA aperture array mid frequency science requirements," *arXiv preprint arXiv:1610.00683*, 2016.
- [2] W. C. Chew, E. Michielssen, J. M. Song, and J. M. Jin, *Fast and efficient algorithms in computational electromagnetics*. Artech House, Inc., 2001.
- [3] S. Kurz, O. Rain, and S. Rjasanow, "The adaptive cross-approximation technique for the 3D boundary-element method," *IEEE Transactions on Magnetics*, vol. 38, no. 2, pp. 421–424, 2002.
- [4] K. Zhao, M. N. Vouvakis, and J.-F. Lee, "The adaptive cross approximation algorithm for accelerated method of moments computations of EMC problems," *IEEE Transactions on Electromagnetic Compatibility*, vol. 47, no. 4, pp. 763–773, 2005.
- [5] C. Craeye, J. Laviada, R. Maaskant, and R. Mittra, "Macro basis function framework for solving Maxwell's equations in surface integral equation form," *The FERMAT Journal*, vol. 3, pp. 1–16, 2014.
- [6] R. Maaskant, R. Mittra, and A. Tjihuis, "Fast analysis of large antenna arrays using the characteristic basis function method and the adaptive cross approximation algorithm," *IEEE Transactions on Antennas and Propagation*, vol. 56, no. 11, pp. 3440–3451, 2008.
- [7] W. B. Lu, T. J. Cui, Z. G. Qian, X. X. Yin, and W. Hong, "Accurate analysis of large-scale periodic structures using an efficient sub-entire-domain basis function method," *IEEE Transactions on Antennas and Propagation*, vol. 52, no. 11, pp. 3078–3085, 2004.
- [8] K. Sewraj, R. C. Beswick, and M. M. Botha, "Efficiency of various macro basis function schemes for antenna array analysis," in *Proceedings of the International Conference on Electromagnetics in Advanced Applications (ICEAA)*, 2018, pp. 254–257, Cartagena, Colombia.
- [9] D. J. Ludick, M. M. Botha, R. Maaskant, and D. B. Davidson, "The CBFM-enhanced Jacobi method for efficient finite antenna array analysis," *IEEE Antennas and Wireless Propagation Letters*, vol. 16, pp. 2700–2703, 2017.
- [10] M. Bebendorf and R. Venn, "Constructing nested bases approximations from the entries of non-local operators," *Numerische Mathematik*, vol. 121, no. 4, pp. 609–635, 2012.
- [11] M. Bebendorf, C. Kuske, and R. Venn, "Wideband nested cross approximation for Helmholtz problems," *Numerische Mathematik*, vol. 130, no. 1, pp. 1–34, 2015.
- [12] K. Sewraj and M. M. Botha, "Directional method to compute reduced matrix system in MBF solvers," in *Proceedings of the European Conference on Antennas and Propagation (EuCAP)*, March 2020, pp. 1–5, Copenhagen, Denmark.
- [13] S. Ambikasaran and E. Darve, "The inverse fast multipole method," *arXiv preprint arXiv:1407.1572*, 2014.
- [14] K. Sewraj and M. M. Botha, "Application of the inverse fast multipole method to antenna array analysis," in *Proceedings of the IEEE International Antennas and Propagation Symposium*, July 2020, pp. 1–2, Montréal, Canada.
- [15] A. K. Skrivervik and J. R. Mosig, "Analysis of printed array antennas," *IEEE Transactions on Antennas and Propagation*, vol. 45, no. 9, pp. 1411–1418, September 1997.
- [16] D. J. Ludick, R. Maaskant, D. B. Davidson, U. Jakobus, R. Mittra, and D. I. L. de Villiers, "Efficient analysis of large aperiodic antenna arrays using the domain Green's function method," *IEEE Transactions on Antennas and Propagation*, vol. 62, no. 4, pp. 1579–1588, April 2014.
- [17] M. Chose and M. M. Botha, "Improvements to the domain Green's function method for antenna array analysis," in *Proceedings of the International Conference on Electromagnetics in Advanced Applications (ICEAA)*, 2019, pp. 392–395, Granada, Spain.
- [18] —, "An iterative method for the analysis of large disjoint antenna arrays," in *Proceedings of the European Conference on Antennas and Propagation (EuCAP)*, March 2020, pp. 1–4, Copenhagen, Denmark.

# List of References

- [1] K. Sewraj and M. M. Botha, “A sparse direct solver based on directional cross approximation for antenna array analysis,” in preparation.
- [2] M. Chose, K. Sewraj, and M. M. Botha, “Iterative radius-based domain Green’s function method with ACA for antenna array analysis,” 2021, submitted for publication.
- [3] K. Sewraj and M. M. Botha, “Computation of MBF reaction matrices for antenna array analysis, with a directional method,” in *Proceedings of the International Conference on Electromagnetics in Advanced Applications (ICEAA '19)*, Sept 2019, pp. 385–389, Granada, Spain.
- [4] —, “Directional method to compute reduced matrix system in MBF solvers,” in *2020 14th European Conference on Antennas and Propagation (EuCAP)*. IEEE, 2020, pp. 1–5.
- [5] —, “Application of the inverse fast multipole method to antenna array analysis,” in *Proceedings of the IEEE International Antennas and Propagation Symposium*, July 2020, pp. 1–2, Montreal, Canada.
- [6] K. Sewraj, M. Chose, and M. M. Botha, “Developments in the analysis of large antenna arrays with disjoint elements,” in *Proceedings of the European Conference on Antennas and Propagation (EuCAP)*, March 2021, pp. 1–3, submitted for publication, Dusseldorf, Germany.
- [7] *SKA website*. [Online]. Available: <https://www.skatelescope.org/>
- [8] *African VLBI Network*. [Online]. Available: <https://www.sarao.ac.za/science/avn/>
- [9] A. Faulkner and J. G. B. de Vaate, “SKA low frequency aperture array design,” in *2013 IEEE International Symposium on Phased Array Systems and Technology*. IEEE, 2013, pp. 768–775.
- [10] S. Torchinsky, J. Broderick, A. Gunst, A. Faulkner, and W. van Cappellen, “SKA aperture array mid frequency science requirements,” *arXiv preprint arXiv:1610.00683*, 2016.
- [11] E. de Lera Acedo, N. Razavi-Ghods, N. Troop, N. Drought, and A. Faulkner, “SKALA, a log-periodic array antenna for the SKA-low instrument: design, simulations, tests and system considerations,” *Experimental Astronomy*, vol. 39, no. 3, pp. 567–594, 2015.
- [12] M. Ruiter and E. van der Wal, “EMBRACE, a 10000 element next generation aperture array telescope,” in *Microwave Conference, 2009. EuMC 2009. European*. IEEE, 2009, pp. 326–329.
- [13] Y. Zhang and A. K. Brown, “Octagonal ring antenna for a compact dual-polarized aperture array,” *IEEE Transactions on Antennas and Propagation*, vol. 59, no. 10, pp. 3927–3932, 2011.

- [14] J. Gilmore, D. Davidson, and J. Noordam, "A dense dipole array for mid-frequency aperture arrays," in *Antennas and Propagation & USNC/URSI National Radio Science Meeting, 2015 IEEE International Symposium on*. IEEE, 2015, pp. 1524–1525.
- [15] J. Abraham, E. Colin-Beltran, E. de Lera Acedo, and A. Faulkner, "A 16-element LPDA random sparse prototype array for the SKA AA-mid instrument," in *Antennas and Propagation (EuCAP), 2016 10th European Conference on*. IEEE, 2016, pp. 1–4.
- [16] H. Bui-Van, C. Craeye, and E. d. L. Acedo, "Main beam modeling for large irregular arrays: The ska1-low telescope case," *arXiv preprint arXiv:1705.10084*, 2017.
- [17] R. F. Harrington and J. L. Harrington, *Field computation by moment methods*. Oxford University Press, 1996.
- [18] J.-M. Jin, *The finite element method in electromagnetics*. John Wiley & Sons, 2015.
- [19] A. Taflove and S. C. Hagness, *Computational electrodynamics: the finite-difference time-domain method*. Artech house, 2005.
- [20] R. Coifman, V. Rokhlin, and S. Wandzura, "The fast multipole method for the wave equation: A pedestrian prescription," *IEEE Antennas and Propagation Magazine*, vol. 35, no. 3, pp. 7–12, 1993.
- [21] W. C. Chew, E. Michielssen, J. Song, and J.-M. Jin, *Fast and efficient algorithms in computational electromagnetics*. Artech House, Inc., 2001.
- [22] C. Craeye, J. Laviada, R. Maaskant, and R. Mittra, "Macro basis function framework for solving Maxwell's equations in surface integral equation form," *The FERMAT Journal*, vol. 3, pp. 1–16, 2014.
- [23] S. Ambikasaran and E. Darve, "The inverse fast multipole method," *arXiv preprint arXiv:1407.1572*, 2014.
- [24] P. Coulier, H. Pouransari, and E. Darve, "The inverse fast multipole method: using a fast approximate direct solver as a preconditioner for dense linear systems," *SIAM Journal on Scientific Computing*, vol. 39, no. 3, pp. A761–A796, 2017.
- [25] S. Kurz, O. Rain, and S. Rjasanow, "The adaptive cross-approximation technique for the 3d boundary-element method," *IEEE transactions on Magnetics*, vol. 38, no. 2, pp. 421–424, 2002.
- [26] K. Zhao, M. N. Vouvakis, and J.-F. Lee, "The adaptive cross approximation algorithm for accelerated method of moments computations of EMC problems," *IEEE Transactions on Electromagnetic Compatibility*, vol. 47, no. 4, pp. 763–773, 2005.
- [27] M. Bebendorf and R. Venn, "Constructing nested bases approximations from the entries of non-local operators," *Numerische Mathematik*, vol. 121, no. 4, pp. 609–635, 2012.
- [28] M. Bebendorf, C. Kuske, and R. Venn, "Wideband nested cross approximation for Helmholtz problems," *Numerische Mathematik*, vol. 130, no. 1, pp. 1–34, 2015.
- [29] R. C. Beswick, "The development of matlab based tools for antenna array analysis," Ph.D. dissertation, 2020.
- [30] C. A. Balanis, *Advanced engineering electromagnetics*. John Wiley & Sons, 1999.
- [31] W. C. Gibson, *The method of moments in electromagnetics*. CRC press, 2014.

- [32] S. Rao, D. Wilton, and A. Glisson, "Electromagnetic scattering by surfaces of arbitrary shape," *IEEE Transactions on antennas and propagation*, vol. 30, no. 3, pp. 409–418, 1982.
- [33] V. Prakash and R. Mittra, "Characteristic basis function method: A new technique for efficient solution of method of moments matrix equations," *Microwave and Optical Technology Letters*, vol. 36, no. 2, pp. 95–100, 2003.
- [34] L. Matekovits, V. A. Laza, and G. Vecchi, "Analysis of large complex structures with the synthetic-functions approach," *IEEE Transactions on Antennas and Propagation*, vol. 55, no. 9, pp. 2509–2521, 2007.
- [35] D. J. Bekers, S. J. van Eijndhoven, and A. G. Tijhuis, "An eigencurrent approach for the analysis of finite antenna arrays," *IEEE Transactions on Antennas and Propagation*, vol. 57, no. 12, pp. 3772–3782, 2009.
- [36] W. B. Lu, T. J. Cui, Z. G. Qian, X. X. Yin, and W. Hong, "Accurate analysis of large-scale periodic structures using an efficient sub-entire-domain basis function method," *IEEE transactions on antennas and propagation*, vol. 52, no. 11, pp. 3078–3085, 2004.
- [37] K. Sewraj, R. C. Beswick, and M. M. Botha, "Efficiency of various macro basis function schemes for antenna array analysis," in *2018 International Conference on Electromagnetics in Advanced Applications (ICEAA)*. IEEE, 2018, pp. 254–257.
- [38] D. J. Ludick, M. M. Botha, R. Maaskant, and D. B. Davidson, "The cbfm-enhanced jacobi method for efficient finite antenna array analysis," *IEEE Antennas and Wireless Propagation Letters*, vol. 16, pp. 2700–2703, 2017.
- [39] S. G. Hay, J. D. O'Sullivan, and R. Mittra, "Connected patch array analysis using the characteristic basis function method," *IEEE Transactions on Antennas and Propagation*, vol. 59, no. 6, pp. 1828–1837, 2011.
- [40] H. Bui-Van, J. Abraham, M. Arts, Q. Gueuning, C. Raucy, D. Gonzalez-Ovejero, E. de Lera Acedo, and C. Craeye, "Fast and accurate simulation technique for large irregular arrays," *IEEE Transactions on Antennas and Propagation*, vol. 66, no. 4, pp. 1805–1817, 2018.
- [41] R. Maaskant, R. Mittra, and A. Tijhuis, "Fast analysis of large antenna arrays using the characteristic basis function method and the adaptive cross approximation algorithm," *IEEE Transactions on Antennas and Propagation*, vol. 56, no. 11, pp. 3440–3451, 2008.
- [42] C. Craeye, "A fast impedance and pattern computation scheme for finite antenna arrays," *IEEE Transactions on Antennas and Propagation*, vol. 54, no. 10, pp. 3030–3034, 2006.
- [43] Q. Gueuning, E. de Lera Acedo, E. Colin-Beltran, and C. Craeye, "Mutual coupling analysis of large irregular arrays: From multipole to interpolatory methods," in *Antennas and Propagation (EuCAP), 2015 9th European Conference on*. IEEE, 2015, pp. 1–4.
- [44] E. Garcia, C. Delgado, F. S. de Adana, F. Catedra, and R. Mittra, "Incorporating the multilevel fast multipole method into the characteristic basis function method to solve large scattering and radiation problems," in *Antennas and Propagation Society International Symposium, 2007 IEEE*. IEEE, 2007, pp. 1285–1288.
- [45] E. Garcia, C. Delgado, I. G. Diego, and M. F. Catedra, "An iterative solution for electrically large problems combining the characteristic basis function method and the multilevel fast multipole algorithm," *IEEE Transactions on Antennas and Propagation*, vol. 56, no. 8, pp. 2363–2371, 2008.

- [46] P. De Vita, A. Freni, L. Matekovits, P. Pirinoli, and G. Vecchi, “A combined AIM-SFX approach for large complex arrays,” in *Antennas and Propagation Society International Symposium, 2007 IEEE*. IEEE, 2007, pp. 3452–3455.
- [47] H. Bui-Van, T. Pairon, and C. Craeye, “Fast iterative techniques for the simulations of large antenna arrays,” in *2018 International Conference on Electromagnetics in Advanced Applications (ICEAA)*. IEEE, 2018, pp. 247–249.
- [48] C. Delgado and M. F. C atedra, “Sparse approximate inverse preconditioner with parametric sparsity pattern applied to the macro basis function methods,” *IEEE Antennas and Wireless Propagation Letters*, vol. 17, no. 5, pp. 849–852, 2018.
- [49] S. Ambikasaran and E. Darve, “An  $\mathcal{O}(N \log N)$  fast direct solver for partial hierarchically semi-separable matrices,” *Journal of Scientific Computing*, vol. 57, no. 3, pp. 477–501, 2013.
- [50] S. Chandrasekaran, M. Gu, and T. Pals, “A fast ULV decomposition solver for hierarchically semiseparable representations,” *SIAM Journal on Matrix Analysis and Applications*, vol. 28, no. 3, pp. 603–622, 2006.
- [51] W. Hackbusch, “A sparse matrix arithmetic based on  $\mathcal{H}$ -matrices. part i: Introduction to  $\mathcal{H}$ -matrices,” *Computing*, vol. 62, no. 2, pp. 89–108, 1999.
- [52] M. Bebendorf, *Hierarchical matrices*. Springer, 2008.
- [53] W. Hackbusch and S. B orm, “ $\mathcal{H}^2$ -matrix approximation of integral operators by interpolation,” *Applied Numerical Mathematics*, vol. 43, no. 1-2, pp. 129–143, 2002.
- [54] G. H. Golub and C. F. Van Loan, *Matrix computations*, 4th ed. JHU press, 2013, p. 65.
- [55] X. Chen, C. Gu, Z. Li, and Z. Niu, “Accelerated direct solution of electromagnetic scattering via characteristic basis function method with sherman-morrison-woodbury formula-based algorithm,” *IEEE Transactions on Antennas and Propagation*, vol. 64, no. 10, pp. 4482–4486, 2016.
- [56] X. Fang, Q. Cao, Y. Zhou, and Y. Wang, “Multiscale compressed block decomposition method with characteristic basis function method and fast adaptive cross approximation,” *IEEE Transactions on Electromagnetic Compatibility*, vol. 61, no. 1, pp. 191–199, 2018.
- [57] A. Heldring, J. M. Rius, J. M. Tamayo, J. Parr on, and E. Ubeda, “Multiscale compressed block decomposition for fast direct solution of method of moments linear system,” *IEEE Transactions on Antennas and Propagation*, vol. 59, no. 2, pp. 526–536, 2011.
- [58] X. Fang, Q. Cao, Y. Zhou, and Y. Wang, “Multiscale compressed block decomposition method with characteristic basis function method and fast adaptive cross approximation,” *IEEE Transactions on Electromagnetic Compatibility*, vol. 61, no. 1, pp. 191–199, 2018.
- [59] X. Chen, C. Gu, J. Ding, Z. Li, and Z. Niu, “Multilevel fast adaptive cross-approximation algorithm with characteristic basis functions,” *IEEE Transactions on Antennas and Propagation*, vol. 63, no. 9, pp. 3994–4002, 2015.
- [60] S. B orm, “Directional  $\mathcal{H}^2$ -matrix compression for high-frequency problems,” *arXiv preprint arXiv:1510.07087*, 2015.
- [61] 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 Transactions on Antennas and Propagation*, vol. 63, no. 5, pp. 2122–2134, 2015.

- [62] Simulia, *Antenna Magus*, 2020. [Online]. Available: <https://www.3ds.com/products-services/simulia/products/antenna-magus/>
- [63] A. K. Skrivervik and J. R. Mosig, “Analysis of finite phase arrays of microstrip patches,” *IEEE Transactions on Antennas and Propagation*, vol. 41, no. 15, pp. 1105–1114, August 1993.
- [64] —, “Analysis of printed array antennas,” *IEEE Transactions on Antennas and Propagation*, vol. 45, no. 9, pp. 1411–1418, September 1997.
- [65] D. J. Ludick, R. Maaskant, D. B. Davidson, U. Jakobus, R. Mittra, and D. de Villiers, “Efficient analysis of large aperiodic antenna arrays using the domain Green’s function method,” *IEEE Transactions on Antennas and Propagation*, vol. 62, no. 4, pp. 1579–1588, 2014.
- [66] *Feko website*. [Online]. Available: [www.altair.com/feko/](http://www.altair.com/feko/)