A Survey on Algorithms for Big Data Analysis in Electromagnetics Scattering Problems

Christian O. Díaz-Cáez
Department of Electrical Engineering
and Computer Science
Howard University
Washington, D.C. USA
e-mail: christian.diaz@bison.howard.edu

Chunmei Liu
Department of Electrical Engineering
and Computer Science
Howard University
Washington, D.C. USA
e-mail: chuliu@howard.edu

***Abstract*— Computational Electromagnetics is a discipline that deals with the processing and modeling of multi-physics and electromagnetic problems. Thanks to the advent of computers and numerical methods, engineers today can develop algorithms and software to solve Maxwell's equations numerically. The electromagnetic scattering problem leads to a very large system of equations with millions or even billions of unknowns; traditional data analysis methods are oftentimes not efficient enough to handle the problem due to data volume. The field of Big Data has emerged from the need to process a massive amount of data and is a research area that facilitates the complex work of extremely large data sizes. Fast algorithms can be developed to efficiently manage the Big Data approach to support areas of science and engineering. In this paper, we explore an application of Big Data and algorithms in computational electromagnetics scattering problems.**

Keywords—Big Data; Computational Electromagnetics (CEM); Method of Moments (MoM); Fast Algorithms, Multilevel Fast Multipole Algorithm (MLFMA).

I. INTRODUCTION

We are currently in an era of digital information. This means that a great amount of information is generated daily. To manage, analyze and store this information, very powerful tools are needed. Big Data technology plays a very important role in this area. It allows large companies to optimize decision-making and obtain results optimally. Big Data is a term used to describe a set of data or combinations of sets of data whose size, complexity, and velocity of growth make it difficult to capture, manage, process or analyze using conventional technologies and tools, such as relational databases and conventional statistics or visualization package, within the time necessary for them to be useful [1]. Although there is no firmly defined size for determining whether a data set is Big Data, and the definition continues to change over time, professionals currently refer to Big Data to be datasets ranging from 30-50 Terabytes to several Petabytes [1].

For some problems, the data size may be so large that it does not fit in the main memory of a single machine. The need to process such a huge amount of data There is a need to process such a huge amount of data through efficient algorithms in machine learning, network traffic monitoring, scientific computing, signal processing, and other areas. Some well-known examples of such algorithms are numerical linear algebra algorithms for big matrices [2] (regression, low-rank approximation, matrix completion), dimensional reduction for reducing data dimension to conserve the geometric structure [3], compressed sensing for approximation recovery of sparse signals [4] and sparse Fourier Transform as fast algorithms for signals calculation in a frequency domain [5]. To better understand Big Data's difficulty, it is often broken down using five V's: Volume, Velocity, Value, Variety, and Veracity [8][9].

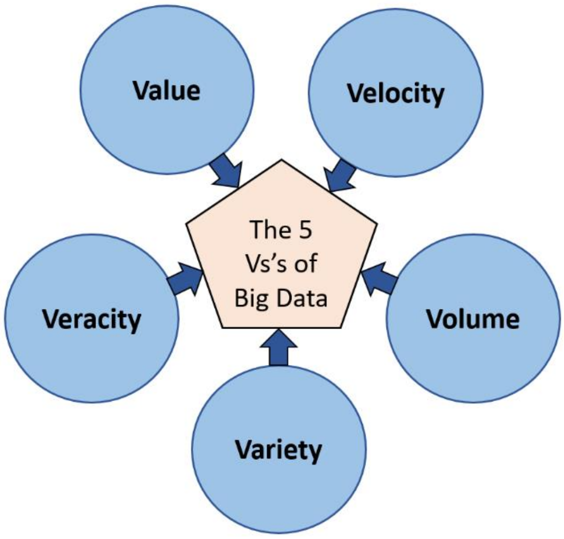

Figure 1. The 5 V's of Big Data.

The 5 V's of Big Data illustrated in Figure 1 can be defined as follows:

- Volume refers to the exponential increase in data resulting from new technologies, and the ease of generating digital data is a palpable reality. The volume means large size.
- Velocity is the rate of growth and how fast data is gathered for analysis.
- Value is indicative of substantial value, including the ability to understand the target better, accordingly, and optimize performance.
- Variety is information about the various types of data, such as structured, unstructured, semi-structured, etc.
- Veracity means the confidence established about the data to be used.

Big Data serves the purpose of converting data (information) into knowledge. Researchers have added more dimensions from 5 to 10 [6], covering terms such as validity, vulnerability, volatility, visualization, variability, and even more, which can be found in technology and data

Copyright (c) IARIA, 2021. ISBN: 978-1-61208-857-0

generation advances [7]. The rest of this paper is organized as follows: Section II describes Computational Electromagnetic (CEM) as an interdisciplinary field, Section III describes the Method of Moments (MoM) as a powerful numerical technique in CEM, Section IV addresses the algorithm techniques to exploit MoM, Section V describes the multilevel fast multipole algorithm and Section VI summarize some Big Data techniques implemented to solve different electromagnetic engineering problems. The conclusions close the article.

II. Computational Electromagnetic

Electromagnetic (EM) analysis is a discipline that solves Maxwell's equations to obtain a better understanding of complex systems. The advent of numerical methods and computers has changed the traditional ways of EM analyzing, and a field called Computational Electromagnetics has emerged [10]. It is a prominent EM research area that involves the modeling of the interaction of EM fields with physical objects, the study of electromagnetic compatibility between equipment in different environments, the design of antennas, the design of passive microwave circuits and components, the calculation of the Radar Cross Section (RCS) and Inverse Synthetic Aperture Radar (ISAR) images, the analysis of antennas embarked on complex structures, Doppler analysis, and radio propagation both indoors and outdoors.

When an EM problem is given for a practical application, we need to describe our problem mathematically based on EM physics to seek a numerical method. We can apply Partial Differential Equations (PDEs) and boundary conditions to define an equivalent boundary-value problem. Then, from our mathematical formulation, we can develop a numerical method effectively, and depending on the problem, we will need to decide to use an existing method or develop a new one addressing the problem. After a numerical method is selected or developed, it is necessary to develop an efficient computer program for implementation. Finally, after the computer program is validated, we can use it to solve the problem given by constructing a geometrical model and the specification of EM mediums (permittivity, permeability, and conductivity) [11]. All the steps previously discussed are summarized in Figure 2.

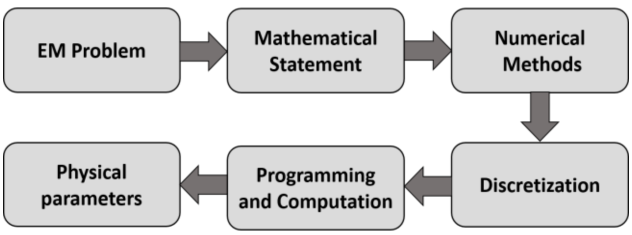

Figure 2. Numerical analysis steps for solving engineering problems.

As shown in Figure 3, CEM is a highly interdisciplinary field that combines physics, mathematics, and computer science to advance engineering applications.

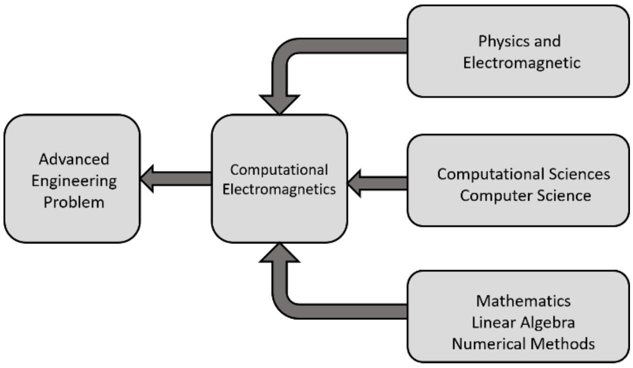

Figure 3. CEM is an interdisciplinary field for advancing engineering applications.

Today, numerical methods for EM scattering problems need to process a very large system of equations with millions or even billions of unknown variables [12]. Traditional methods are inefficient and fast algorithms in EM have been developed to solve this problem in an efficient manner [10]. As a common method, we can represent our system of unknowns as a hierarchical representation with a matrix system of N number of unknowns. Fast algorithms use $O(NlogN)$ memory and approximately $O(N)$ or even $O(logN)$ time [12]. Traditional numerical methods usually require $O(N^2)$ memory and $O(N^2)$ time so in the scenario that N becomes very large, we can identify a huge discrepancy in memory and time between traditional and fast algorithms [10].

The next section describes an efficient algorithm for electromagnetic scattering problems that can be implemented in multicore-based and cluster architectures. Electromagnetics simulations are critically important in several application areas, such as antenna design for aircraft, satellites, and medical devices. We can reduce the numerical formulation cost by assuming time-harmonic solutions and reformulating Maxwell's equations to describe EM waves in terms of surface currents. The result of this approach is a numerical problem that can be solved on the surface of the object being studied.

III. Method of Moments

The method of moments is a very powerful numerical technique developed for solving complex EM problems. Compared to the Finite Element Methods (FEM), MoM also transforms the boundary-value problem into a matrix equation that can be solved on computers [13]. Mathematical-based MoM was proposed almost one century ago, but its applications did not arise until 1960s [14]. Today, it is one of the most important methods in CEM. MoM has been well studied on open-region electromagnetic problems, such as wave scattering and antenna radiation, and it is very efficient for problems involving either impenetrable or homogeneous objects [13]. Also, the capability of MoM has been improved by the development of fast algorithms that can deal with huge MoM matrix equations [12]. MoM forces the boundary conditions to be satisfied in an average sense over the entire surface.

We can see a system of equations to compute the surface currents as an inverse problem. Applying an iterative method, the inverse problem is converted to repeated solutions of the forward problem. For example, a basic problem in EM consists of computing an EM field, given the distribution of sources/charges. The forward model is well known to compute electrostatic potential

$$\Phi = \sum_{j=1}^{N} K(x, x_j)\, q_j \tag{1}$$

where q_j is the point charge at the location represented by x_j.The interaction between the field points and the charges is represented by the kernel $K(x, x_j)$ which is logarithmic in two dimensions and proportional to the inverse distance in three dimensions.

The corresponding scattering problem computes electric and magnetic fields **E** and **H** generated by surface currents on metallic objects, such as aircraft [21]. Avoiding its mathematical derivation, a simplified form is given by

$$E(r) = \int_{\partial\Omega} G(r, r') j(r') + \frac{1}{k^2} \nabla \left(G(r, r') \nabla \cdot j(r')\, dr' \right. \tag{2}$$

where $\partial\Omega$ is the surface of the object, and for computer simulation, it is discretized, ***r* is** a point in the space and $G(r, r')$ is the *Green's function* representing a point source response [21]. Figure 4 shows a visualization of an example of a discretized unit sphere. To make this type of problem solvable by computers, we need to discretize the object in *N* number of pieces. We can represent the sources and fields of the surface current by a set of *basis functions* and corresponding coefficients to approximate the solution of the surface current [19]. After the discretization, we convert the problem to a matrix equation by intruding on another set of functions called *testing* or *weighing functions* [19].

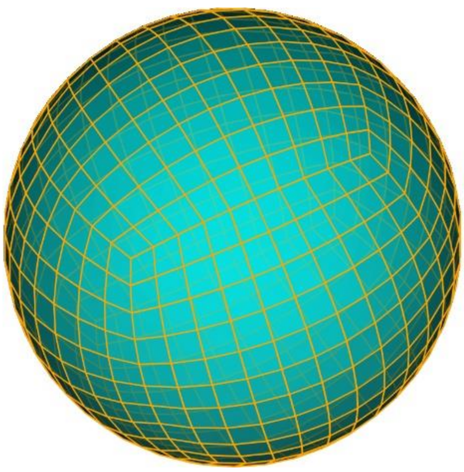

Figure 4. Discretization of a unit sphere in small patches.

It can be expressed in a compact form as

$$\sum_{j=1}^{N} Z_{ij} I_j = V_i \qquad i = 1,2,3 \ldots N \tag{3}$$

where Z_{ij} is the *N* x *N* matrix system with the unknown coefficients, I_j is the vector of unknowns, and V_i is the source vector.

IV. Fast Algorithms

Unlike FEM based on PDEs that yield to huge sparse matrix system, the method of moments, MoM, based on integral equations (IEs), produces a fully populated matrix system because of the applications of the Green's function. Now, the problem is the high complexity associated with methods for the full matrix solution. It becomes a limitation on the capability of MoM. In conventional methods for matrix solutions, such as Gaussian elimination or lower-upper (LU) decomposition, the time complexity is $O(N^3)$ and the space complexity is $O(N^2)$, where *N* is the matrix dimension. An iterative method can reduce the time complexity to $O(N^2)$, but the memory remains the same for a direct method. The total time complexity is $O(N_{iter} N^2)$ where N_{iter} is the number of iterations reaching a certain convergence. If N_{iter} is small, then an interactive process will be faster than LU decomposition just for the right-hand side of the equation, but the iterative solution must be repeated for every right-hand size [15][20], which makes MoM limited to one-, two- or three-dimensional problems.

A better understanding of the high computational complexity of traditional direct and interactive methods can be found in [15]. The complexities of $O(N^3)$and $O(N^2)$ make the time and space increase dramatically with the increase of the number *N*, and it may exceed the capabilities computers have today. A technique used to reduce time and memory complexities for iterative methods, especially for large-scale problems, is called *fast algorithms*. We can broadly define fast algorithms as algorithms that can solve both matrix and integral equations that can be discretized in a matrix equation by MoM. More details regarding fast algorithms can be found in [10][14][15]. Some examples of fast algorithms are the Conjugate Gradient–FFT (CG-FFT) method, the Adaptive Integral Method (AIM), the Fast Multipole Method (FMM), and the Adaptive Cross-Approximation (ACA) method.

For this survey, we focus on FMM because it is the base for the technique presented in the next section of this paper. FMM divides the current elements into groups by their physical locations in space. A group is then defined as a collection of current elements near each other. Figure 5 illustrates an example of an arbitrary object with basis functions divided into groups, so the computation of far fields that is calculated indirectly in multiples steps is made fast, whereas near fields are computed directly (more quickly).

FMM integrates a new concept of decomposing the MoM matrix into near-and-far-interaction components. It makes a fast matrix-vector calculation possible by multipole or plane wave expansions and eventually reduces the computational complexity to *O(NlogN)* [18].

V. MULTILEVEL FAST MULTIPOLE ALGORITHM

For a problem with N unknowns, we can divide them into *N/M* groups. For near-fields interactions, aggregations, and disaggregation, *O(NM)* operations are required, whereas the calculation of translation requires $O(N^2/M)$ operations. A small or large number of groups M will improve the complexity performance of the operation count for the calculation of near-fields interactions or translation calculation. An optimal choice of M is M proportional to $\sqrt{N}$ and the operation count in each calculation is balanced to $O(N^{3/2})$ [15]. We can apply FMM to each group; if we have small groups and each group has only a few basis functions, the calculation of near-fields interaction will be only $O(N)$, and the same will be the case for aggregation and desegregation [15]. To reduce the translation calculation, when the groups are far from each other, we aggregate the field from the center of a group to another large group and designate the received field to the groups residing in the second larger group. This process reduces the translation counts, and this idea can be extended to multiple levels until there are no far-apart groups among the highest-level group. The algorithm that results from all this procedure is called Multilevel Fast Multipole Algorithm (MLFMA) [10].

In [10] and [15], the authors introduce a comparison example of a telephone communication scenario to understand how FMM and MLFMA work. We can consider a network with N telephones. Imagine that all the telephones are directly connected. In that case, we will need N^2 telephone lines. If we divide the telephones into groups according to their proximity to each other, and then connect all the telephones in the same group to a single hub, and then connect the hubs, we can reduce the number of telephones lines to $O(N^{3/2} log\ N)$; this is basically what FMM does. Now, imagine that we can establish a second level of hubs which can further reduce the number of telephone lines. If the number of telephone lines is very large, we can reduce the number of telephone lines to *O(NlogN)* by establishing multiple levels of hubs.

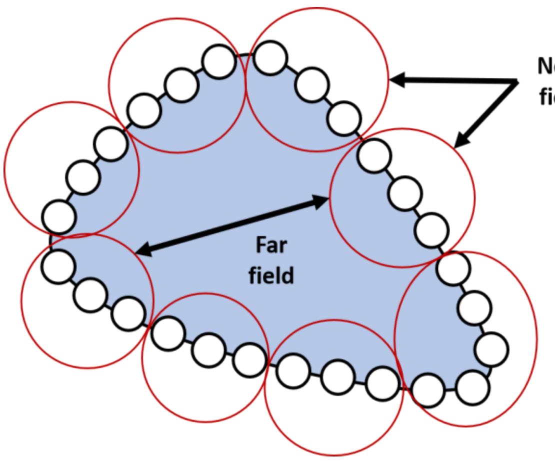

Figure 5. Basis functions are divided into groups for fast far-field computation.

Similarly, MLFMA reduces the operation counts and memory requirement of the FMM to *O(NlogN)*.

Finally, as a real application example, in [22], there is a snapshot of the surface current on a card induced by a Hertzian dipole at 1.0 GHz and a snapshot of the surface current on an airplane induced by an incident plane wave at 2.0 GHz. The discretization of the airplane surface results in nearly 1 million unknowns. Storing it in its corresponding full MoM matrix would take around 8TB of storage memory. Using MLFMA, the memory storage requirement is reduced to 2.5 GB. In [18], the same airplane is simulated with approximately 10 million unknowns at 8 GHz. Also, in [12], we can find another example of surface current on an aircraft from a boundary element with approximately 2 million unknowns.

In addition to all the topics discussed above, there are parallelization approaches of the MLFMA on distributed memory computers. The most common parallelization approach is to partition the data as tree structures over computational nodes. To make this possible, we apply Message Passing Interface (MPI) based parallelization, such as [16][17].

This paper has briefly discussed the complexity performance for MoM and the accelerated versions with FMM and MLFMA as integral method solvers for EM problems in the frequency domain. A comparison of the complexity performance for these three algorithms using iterative solvers is summarized in Table 1.

TABLE I. MOM BASED FAST ALGORITHM COMPLEXITIES

Method	**Complexity**	
	Time	***Memory***
MoM	$O(N^2)$	$O(N^2)$
FMM	$O(N^{1.5})$	$O(N^{1.5})$
MLFMA	*O(NlogN)*	*O(NlogN)*

VI. BIG DATA TECHNIQUES IN ELECTROMAGNETIC ENGINEERING PROBLEMS

The electromagnetic spectrum has shown four characteristics of Big Data, namely, Variety, Volume, Value, and Velocity [32]. One application of Big Data is reported in [32], where data mining is used to detect abnormal spectrum and abnormal positioning targets from massive EM data in real-time. Another application of Big Data in EM problems is Symbolic Regression (SR). This type of regression analysis is used to perform a search in an analytical expression that fits a large dataset [23] SR is classified as a Machine Learning technique and can be applied to derive a full-wave simulation-based analytical expression for the characteristic impedance Z_0 of microstrip lines using Big Data resulting from a 3D-EM simulation [23]. SR is considered a suitable algorithm for obtaining accurate analytical expressions where the interrelations within the data are highly complex in a very large dataset [23]. A different implementation of machine learning to manage the large size of data for design optimization in EM can be found in the literature, such as reinforcement learning for antenna configuration and design [33], deep learning for microwave filter and circuit design [34], EM inverse problems in oil and gas exploration, as well as microwave and optical imaging [30].

Big Data in EM is found in the design of tilted-beam antennas with aperiodic Partially Reflective Surfaces (PRS). To design antennas for beamforming and high gain wireless application, PRS are highly reflective metasurfaces considered well suitable for the design of antennas [25]. During the optimization process of the aperiodic PRS, a large data size is generated. An improved Hybrid Real-binary Bat Algorithm (HRBBA) is applied to optimize the aperiodic PRS [26]. Bat Algorithm (BA), inspired by the echolocation of microbats, efficiently and reliably process Big Data optimization problems [27]-[29]. In [30], a statistical approach is proposed based on the Markov Chain Monte Carlo (MCMC) for Large-Scale Georsteering inversion using directional electromagnetic logging measurements. Due to the high volume of data collection in the oil and gas industry, the proposed method in [30] addresses large-scale inverse problems.

Today, the convergence between Big Data Analytics and High-Performance Computing is considered a promising research area [35]. In CEM, training deep learning or running large-scale simulations can take a tremendous amount of time. For this reason, parallel and high-performance computing are essential to efficiently accelerate the convergence of an algorithm toward an accurate solution. An application of Big Data techniques in the EM scattering problem can be found in [31]. This work proposes a method to predict the number and location of scattering grating lobes produced by an array antenna. The method used implements the idea of decomposing the RCS of the array antenna into a multiplication of the array RCS factor and the element RCS factor.

Fast Algorithms such as MLFMA are developed to accelerate the algorithm execution. At the same time, they can reduce the complexity of the algorithm in terms of memory and time; especially, it considerably alleviates the memory requirement to store the matrix system that can store millions or billions of unknown's values. Parallel computing is implemented to reduce the computational time of the algorithm; in addition, it extends the usability of multiple threats for the mathematical operations in solving the problem. Applications of high-performance computing in EM engineering applications can be found in areas such as EM radiation, propagation and scattering, antenna analysis, RCS, analysis of Electromagnetic Compatibility (ECM) and Electromagnetic Interference (EMI), circuits modeling, microwave, analysis, nano-electronic devices among others [36]-[40].

VII. CONCLUSION

Big Data has become one of the most important fields for complex research related to engineering applications. We have seen that the term Big Data does not only mean a very large amount of data; it is also a concept considering several important factors, such as how we interpret data, how valuable it is, and even how variable the data could be, like the well-known 5 V's of Big Data. Besides, Big Data helps with the management of structured, unstructured, or misstructured data. Efficient algorithms exploit Big Data's potential by reducing its computational complexity in modern computers. High-performance computing supports efficient large-scale data-intensive processing to enable complex applications in different scientific and engineering fields.

In this survey, we have described what computational electromagnetics is and how highly multidisciplinary of a field it is. We have also described the numerical procedures of MoM and its application in EM scattering problems. MoM has been the base for fast algorithm implementations, such as FMM and MLFMA. It is important to state that MLFMA has been one of the most important advances in CEM in the last two decades. The development of numerical methods can be applied effectively across spatial, temporal, and frequency scales with the modeling and simulation of physical phenomena, such as circuits, heat transfer, and charge transport. This opens a new opportunity for computational electromagnetics research.

ACKNOWLEDGMENT

This work is supported by the US Science & Technology Center grant (CCF-0939370).

REFERENCES

[1] R. Narasimhan and T. Bhuvaneshwari, "Big Data - A Brief Study," International Journal of Scientific & Engineering Research, vol. 5, no. 9, pp. 350-353, 2014.

[2] T. Sarlos, "Improved Approximation Algorithms for Large Matrices via Random Projections," In Proceedings of the 47th Annual IEEE Symposium on Foundations of Computer Science, pp. 143-152, 2016.

[3] C. Sorzano, J. Vargas, and A. Pascual Montano, "A survey of dimensionality reduction techniques," arXiv preprint arXiv:1403.2877, 2014.

[4] D. Jin, Y. Yang, T. Ge, and D. Wu, "A Fast Sparse Recovery Algorithm for Compressed Sensing Using Approximate l_0 Norm and Modified Newton Method," Materials, vol. 12, no. 8, pp. 1227, 2019.

[5] A. Gilbert, P. Indyk, M. Iwen, and L. Schmidt, "Recent Developments in the Sparse Fourier Transform: A compressed Fourier transform for big data," IEEE Signal Processing Magazine, , vol. 31, no. 5, pp. 91-100, 2014.

[6] H. Shah, G. Badsha, A. Abbasi, and S. Salehian, "The 10 Vs, Issues and Challenges of Big Data," Proceedings of 2018 the International Conference on Big Data and Education, pp. 52-56, 2018.

[7] N. Khan et al., "The 51 V's Of Big Data: Survey, Technologies, Characteristics, Opportunities, Issues, and Challenges," Proceedings of the International Conference on Omni-Layer Intelligent Systems, pp. 19-24, 2019.

[8] R. Patgiri and A. Ahmed, "Big Data: The V's of the Game Changer Paradigm," 2016 IEEE 18th International Conference on High-Performance Computing and Communications; IEEE 14th International Conference on Smart City; IEEE 2nd International Conference on Data Science and Systems (HPCC/SmartCity/DSS), pp. 17-24, 2016.

[9] A. Shanin, "Big Data Five V's Characteristics," International Journal of Advances in Electronics and Computer Sciences, vol. 2, no. 1, 2015.

[10] W. C. Chew, "Introduction to Electromagnetic Analysis and Computational Electromagnetics," Fast and Efficient Algorithms in Computational Electromagnetics, pp. 1-17, 2001.

[11] J. M. Jin, "Concluding Remarks on Computational Electromagnetics," Theory, and Computation of Electromagnetic Fields, Ed. 2, pp. 651-671, 2015.

[12] E. Larsson et al., “Parallelization of Hierarchical Matrix Algorithms for Electromagnetic Scattering Problems, High-Performance Modelling and Simulation for Big Data Applications,” In High-Performance Modelling and Simulation for Big Data Applications, pp. 36-68, 2019.

[13] J. M. Jin, “The Method of Moments,” Theory and Computation of Electromagnetic Fields, Ed. 2, pp. 651-671, 2015.

[14] K. K. Mei and J. Van Bladel, “Scattering by perfectly conducting rectangular cylinders,” IEEE Trans. Antennas Propag., vol. 11, no. 2, pp. 185–192, 1963.

[15] J. M. Jin, “Fast Algorithms and Hybrid Techniques,” Theory and Computation of Electromagnetic Fields, Ed. 2, pp. 651-671, 2015.

[16] K. C. Donepudi, J. M. Jin, S. Velamparambil, J. M. Song, and W. C. Chew, “A higher-order parallelized multilevel fast multipole algorithm for 3D scattering,” IEEE Trans. Antennas Propag., vol. 49, pp. 1069–1078, 2001.

[17] J. Kurzak and B. Pettitt, “Massively parallel implementation of a fast multipole method for distributed memory machines,” Journal of Parallel and Distributed Computing, vol. 65, no. 7, pp. 870–881, 2005.

[18] W. C. Chew, J. M. Jin, E. Michielssen, and J. M. Song, “Fast and Efficient Algorithms in Computational Electromagnetics,” Norwood, MA: Artech House, 2001.

[19] D. B. Davidson, “A one-dimensional introduction to the method of moments: thin-wire modeling,” Computational Electromagnetics for RF and Microwave Engineering, pp. 118-145, 2005.

[20] D. B. Davidson, “The method of moments for surface modeling,” Computational Electromagnetics for RF and Microwave Engineering, pp 184-230, 2005.

[21] T. J. Cui and W. C. Chew, “Fast Forward and Inverse Methods for Buried Objects,” Fast and Efficient Algorithms in Computational Electromagnetics, pp. 347-424, 2001.

[22] J. M. Song, C. C. Lu, W. C. Chew, and S. W. Lee, “Fast Illinois solver code (FISC),” IEEE Antennas Propag. Mag., vol. 40, no. 3, pp. 27–34, 1998.

[23] P. Mahouti et al., “Symbolic regression for derivation of an accurate analytical formulation using "Big Data": An application example,” Applied Computational Electromagnetics Society Journal, vol. 32, no. 5, pp. 372-380, 2017.

[24] M. F. Korns, “Extremely accurate symbolic regression for large feature problems,” Genetic Programming Theory and Practice XII, Genetic and Evolutionary Computation, pp. 109-131, 2015.

[25] A. Hoorfar and C. Israel, “Nature-Inspired Optimization of Aperiodic Metasurfaces for Antenna Beam-shaping,” 2019 IEEE International Symposium on Antennas and Propagation and USNC-URSI Radio Science Meeting, pp. 1035-1036, 2019.

[26] Y. F. Cheng et al., “Design of Tilted-Beam Fabry-Perot Antenna with Aperiodic Partially Reflective Surface,” Applied Computational Electromagnetics Society Journal, vol 32, no. 5, 2017.

[27] X. S. Yang, “A New Metaheuristic Bat-Inspired Algorithm,” In: J. R. Gonzalez et al. (eds), Nature Inspired Cooperative Strategies for Optimization (NISCO 2010), Springer, Berlin, pp. 65-74, 2010.

[28] A. H. Gandomi, X. S. Yang, A. H. Alavi, and S. Talatahari, “Bat algorithm for constrained optimization tasks,” Neural Computing and Applications, pp. 1239-1255, 2012.

[29] X. S. Yang, “Bat algorithm for multi-objective optimization,” International Journal of Bio-Inspired Computation, pp. 267-274, 2013.

[30] Q. Shen, X. Wu, J. Chen, and Z. Han, “Distributed Markov Chain Monte Carlo Method on Big-Data Platform for Large-Scale Geosteering Inversion Using Directional Electromagnetic Well Logging Measurements,” Applied Computational Electromagnetics Society Journal, vol. 32, no. 5, 2017.

[31] S. Zhang, X. Wang, L. Guo, and J. Ma, “Multiplication Theory for Prediction of the Scattering Grating-lobe of Array Antenna,” Applied Computational Electromagnetics Society Journal, vol. 32, no. 5, 2017.

[32] L. N. Liu, R. Shi, B. Hee, and M. Chen, “Detection on Abnormal Usage of Spectrum by Electromagnetic Data Mining,” In 2019 IEEE 4th International Conference on Big Data Analytics (ICBDA), pp. 182-187, 2019.

[33] X. Chai et al., “Reinforcement Learning Based Antenna Selection in User-Centric Massive MIMO,” In 2020 IEEE 91st Vehicular Technology Conference (VTC2020-Spring), pp. 1-6, 2020.

[34] J. Jing et al., “Recent Advances in Deep Neural Network Technique for High-Dimensional Microwave Modeling,” In 2020 IEEE MTT-S International Conference on Numerical Electromagnetic and Multiphysics Modeling and Optimization (NEMO), pp. 1-3, 2020.

[35] S. Caíno-Lores, J. Carretero, B. Nicolae, O. Yildiz, and T. Peterka, “Toward High-Performance Computing and Big Data Analytics Convergence: The Case of Spark-DIY,” IEEE Access, vol. 7, pp. 156929-156955, 2019, doi: 10.1109/ACCESS.2019.2949836.

[36] M. J. Gander and S. Vandewalle, “Analysis of the parareal time-parallel time-integration method,” SIAM Journal on Scientific Computing, vol. 29, no. 2, pp. 556-578, 2007.

[37] C. Cong, X. C. Cai, and K. Gustafson, “Implicit space-time domain decomposition methods for stochastic parabolic partial differential equations,” SIAM Journal on Scientific Computing, vol. 36, no. 1, pp. C1–C24, 2014.

[38] M. Emmett, and M. Minion, “Toward an efficient parallel in time method for partial differential equations,” Communications in Applied Mathematics and Computational Science vol. 7, no. 1, pp. 105-132, 2012.

[39] A. J. Christlieb, C. B. Macdonald, and B. W. Ong, “Parallel high-order integrators,” SIAM Journal on Scientific Computing, vol. 32, no. 2, pp. 818-835, 2010.

[40] M. J. Gander, “50 years of time parallel time integration,” In Multiple shooting and time domain decomposition methods, Springer, Cham, pp. 69-113, 2015.