International Journal of Mathematics and Computer Research

ISSN: 2320-7167

Volume 09 Issue 09 September 2021, Page no. – 2376-2380

Index Copernicus ICV: 57.55, Impact Factor: 7.184

DOI: 10.47191/ijmcr/v9i9.01

The New Modification of the Approximating Functions Method for Cloud Computing

Denis Zolotariov

Dr. Math., Kharkov, Ukraine, https://orcid.org/0000-0003-4907-7810

ARTICLE INFO

Published Online:
16 September 2021

Corresponding Author:
Denis Zolotariov

ABSTRACT

The approach for building cloud-ready fault-tolerant calculations by approximating functions method, which is an analytical-numerical part of Volterra integral equation method for solving 1D+T nonlinear electromagnetic problems, is presented. The solving process of the original algorithm of the method is modified: it is broken down into the sequential chain of stages with a fixed number of sequential or parallel steps, each of which is built in a fault-tolerant manner and saves execution results in fault-tolerant storage for high availability. This economizes RAM and other computer resources and does not damage the calculated results in the case of a failure, and allows stopping and starting the calculations easily after manual or accidental shutdown. Also, the proposed algorithm has self-healing and data deduplication for cases of corrupted saved results. The presented approach is universal and does not depend on the type of medium or the initial signal. Also, it does not violate the natural description of non-stationary and nonlinear features, the unified definition of the inner and outer problems, as well as the inclusion of the initial and boundary conditions in the same equation as the original approximating functions method. The developed approach stress-tested on the known problems, stability checked and errors compared.

KEYWORDS: Cloud computing, Fault-tolerant computing, High-availability, Integral equation, Approximating functions method, Nonlinear dielectric layer, Volterra integral equation method

I. INTRODUCTION

The interaction of electromagnetic fields with nonlinear media has fundamental importance for developing laser technologies, technologies of signal processing, optical communications, nanocomputers and others, where electromagnetic wave interacts with nonlinear media in time in bounded spatial regions. Solving process of such problems requires the development of adequate mathematical models involving new methods for solving initial-boundary problems of electrodynamics. The most practical problems are now solved using only numerical or analytical-numerical methods.

Currently, methods based on direct digitization of Maxwell equations in differential form for constructing numerical algorithms for solving electrodynamics problems are very popular: FDTD [1], TLM method [2], FVTD [1, 3, 4]. The method of finite elements remains important in the time as well as in the frequency domains [5, 6]. These methods have high universality but often do not take into account particular features of the problem that deteriorates the solution quality. They also often have a number of essential restrictions when solving problems with non-stationary and nonlinear media such as the lack of difference schemes, the description of the boundary conditions, general stability, the solution accuracy, and high demand for computer resources. Exclusively numerical nature of these methods also hampers the qualitative analysis of the phenomenon under study.

An alternative approach to the differential formulation of the problem is based on integral equations equivalent to Maxwell equations [7, 8]. Its key features are natural description of non-stationary and nonlinear features, unified definition of the inner and outer problems, and inclusion of initial and boundary conditions in the same equations – the equations form is the same for various media and does not depend on the initial signal equation. This significantly simplifies the problem statement and its solution process, based on the universal modeling algorithms for a wide range of electrodynamics problems [9, 10].

The one of the most promising analytical-numerical approach for obtaining solution in this case is the approximating functions method [11, 12]. It is a particular case of the finite elements method that is based on a partitioning the definition region by cells and approximation of the desired function by orthogonal polynomials in each of them. The method reduces the calculation process to the problem of solving a system of nonlinear algebraic equations, for which there are many well-known methods, for example Newton's method.

This method in the classical formulation [12] involves constructing and solving a problem at once for the entire space and time domain. Failure at any stage of the solution process: during construction of a system of equations, during its solving, during calculation of the fields outside the inhomogeneity, etc. – leads to the need to restart the entire solution process from the beginning. Also, an increase in the size of the domain or a decrease in the simulation step leads to an increase in the size of the system of nonlinear equations and in the dimensions of the Jacobi matrix in Newton method in quadratic proportion. Because of this, the solving process may become too demanding on computational resources or simply impossible due to their limitations. So, all this shows that the original solution scheme is not fault-tolerant with respect to either software or hardware failures and so is not could-ready.

The construction of a cloud-ready fault-tolerant calculation scheme for approximating functions method for solving nonlinear integral equations that represents electromagnetic problems by Volterra integral equation method is demonstrated in this paper.

II. THE APPROXIMATING FUNCTIONS METHOD

The subject for consideration is the 2D Volterra integral equation of the second kind that describes electromagnetic processes in the space-time domain (1D+T) [9], which in the dimensionless variables [12] can be presented as follows

$$E(\tau,\xi)=E_0(\tau,\xi)-$$
$$-\frac{1}{2}\frac{\partial}{\partial\tau}\left[\int\limits_{\tau^L_{\min}}^{\tau}\left(\begin{array}{l}\frac{1}{\varepsilon\varepsilon_0}P(\tau',\xi-\tau+\tau')-\\ -\frac{\varepsilon-1}{\varepsilon}E(\tau',\xi-\tau+\tau')\end{array}\right)d\tau'+\right. \qquad (1)$$
$$\left.+\int\limits_{\tau^H_{\min}}^{\tau}\left(\begin{array}{l}\frac{1}{\varepsilon\varepsilon_0}P(\tau',\xi+\tau-\tau')-\\ -\frac{\varepsilon-1}{\varepsilon}E(\tau',\xi+\tau-\tau')\end{array}\right)d\tau'\right].$$

Here $E(\tau,\xi)$ is the electric field inside or outside an inhomogeneity located in the spatial interval $\xi\in[0,1]$, $E_0(\tau,\xi)$ is an undisturbed electric field without the inhomogeneity, $P(\tau,\xi)$ is the polarization of the medium inside the inhomogeneity which has different electromagnetic characteristics than the environment outside of interval $\xi\in[0,1]$, ε is the permittivity of the medium in the environment, ε_0 is the permittivity of vacuum, and lower limits are

$$\begin{aligned}\tau^L_{\min}&=\max(0,\tau-\xi),\\ \tau^H_{\min}&=\max(0,\tau+\xi-1).\end{aligned} \qquad (2)$$

Equation contains the initial and boundary conditions and describes the electromagnetic field evolution in the time interval $\tau\in[0,T]$. For points $\xi\in[0,1]$ this integral equation formulates the problem for the inner field, the inner problem. Otherwise, it is the formula for calculation of the outer field through the inner one, the outer problem.

Nonlinearity is introduced to Equation (1) by the features of the material inside the inhomogeneity which are described by the polarization

$$P(\tau,\xi)=\sum_{i=1}^{n}\gamma_i E^i(\tau,\xi), \qquad (3)$$

where $\gamma_1=\varepsilon_0(\varepsilon_1-1)$. The linear feature of the medium inside the inhomogeneity is described by the permittivity ε_1, the nonlinear features of the i-th order ($i\geq 2$) by nonlinear susceptibilities γ_i.

For solving Equation (1) by approximating functions method a mesh of semi-closed squares in the rectangle $D=[0,T]\times[0,1]$ is constructed

$$\begin{aligned}&D=\bigcup D_{ij},\\ &D_{ij}=\{ih\leq\tau<(i+1)h,\ \ jh\leq\xi<(j+1)h\},\\ &i=\overline{0,\text{n-1}},\ \ j=\overline{0,\text{m-1}},\end{aligned} \qquad (4)$$

where h is a square side, a spacing of the mesh. The solution to Equation (1) is constructed approximately as a sum of piecewise-smooth functions $\hat{E}_{i,j}(\tau,\xi)$ each of which is determined in the corresponding grid cell D_{ij}:

$$E(\tau,\xi)\approx\hat{E}(\tau,\xi)=\sum_{i=0}^{n-1}\sum_{j=0}^{m-1}\hat{E}_{i,j}(\tau,\xi). \qquad (5)$$

These functions are constructed from the four approximating polynomials of the special form [12] that can be written in general as follows:

$$T^{d_1,d_2}_{i,j}(\tau,\xi)=T^{d_1}_i(\tau)\cdot T^{d_2}_j(\xi), \qquad (6)$$

$$T^d_i(s)=(1-d)+(-1)^{d+1}\frac{s-ih}{h},d=\overline{0,1}, \qquad (7)$$

with corresponding weighting coefficients $c_{i,j}$ that in fact are approximated values for the function $\hat{E}$ from Equation (5) at the vertexes (i, j) of the cell: $c_{i,j}=\hat{E}(ih,jh)$.

Substitution of Equation (5) in (1) and calculation of the whole Equations (1) at the points

$$(\tau_i=ih,\xi_j=jh),i=\overline{0,\text{n}},\ j=\overline{0,\text{m}} \qquad (8)$$

give the system of algebraic equations for the weighting coefficients $c_{i,j}$, that is solved by classical Newton method.

For the obtaining solution for an outer problem the fields of the reflected wave as well as of the transmitted one are represented in similar forms

$$\hat{E}_{refl}\left(\tau,\xi\right)=\sum_{i=0}^{n-1}\sum_{j=m_{ref}}^{-1}\hat{E}_{i,j}^{refl}\left(\tau,\xi\right),$$
$$\hat{E}_{trans}\left(\tau,\xi\right)=\sum_{i=0}^{n-1}\sum_{j=m}^{m_{trans}-1}\hat{E}_{i,j}^{trans}\left(\tau,\xi\right), \tag{9}$$

where m_{ref} and m_{trans} are the minimal and maximal magnitudes of the spatial variable in the index units.

The accuracy of the calculations is ensured by checking the energy balance, which is calculated after obtaining the solution to the inner problem. The energy flow on the inhomogeneity boundaries $\xi_b\in\left\{0,1\right\}$ (0 for the left side boundary and 1 for the right one) is calculated according to [12] after which energy balance is calculated.

III. FORMULATION OF THE APPROACH

To build a cloud-ready scheme of solving process of the problem described above, the following approach is proposed.

The actions of the described method of solving the problem must be divided into stages, each of which has only backward dependency. Each stage should be broken down into steps with no dependencies on each other or only with backward dependency on the results of previous steps. Steps can be executed in sequence or, if possible, in parallel mode. The results of execution of each stage step should be stored in fault-tolerant storage systems.

A computational process built on such principles will save its state up to the current step of the current stage, which means that by fact it will be journaling for high-availability, the basic principles of which are described, for example, by the authors of Apache Kafka [13]. This means that if calculations are interrupted and restarted at an arbitrary time, they will automatically resume exactly from the last correctly executed and completely saved stage’s step before the moment of interruption. This will happen due to the automatic full restoration of the state of the computational process after a reload of all previously correctly executed stages and their steps from the storage of the results. In case of missing or corrupted saved results, only the results up to the first correct one should be loaded, and the rest should be discarded. Execution will continue as if the discarded results have not yet been obtained.

It is also important to keep in mind that in cases of short-term interruption of the network connection or other reasons, duplicate data may appear in storage due to their repeated sending. To solve this problem, it is necessary to load the calculation results in a way that relies on the unambiguous difference between the results of two arbitrary stages and stage steps, which will provide automatic data deduplication on the client side of the storage system.

IV. Application of the Approach to the Problem

To apply the approach discussed in the previous section for the approximation functions method let’s break down its algorithm into the following stages:

1. Calculating the initial field values at all mesh points necessary for the solving inner and outer problems;
2. Solving inner problem;
3. Solving outer problem;
4. Calculating the energy flow and balance and checking the accuracy of the obtained solution.

Stage 1 is trivial. The calculation of the initial field value at one mesh point is a stage step, which, obviously, can be performed in parallel. Also the storage of stage results must allow random or at least sequential access for using the results in the stages 2 and 3.

To solve the inner problem at stage 2 it is necessary to split the whole domain D from Equation (4) into subdomains (windows) $\widehat{D}_w$

$$D=\bigcup\widehat{D}_w,$$
$$w=\overline{0,\lceil \mathrm{n/K}\rceil-1}, \tag{10}$$

where K is the window size in index units, a multiple of the step h, w is a window number.

The size in time domain of each window must be a multiple of the mesh step h, and in spatial domain it must fill the entire area $\xi\in\left[0,1\right]$:

$$\widehat{D}_w=\bigcup D_{ij}=\left\{ih\le\tau<\left(i+1\right)h,0\le\xi<1\right\},$$
$$i=\overline{wK,\,\min\left[n-1,\left(w+1\right)K-1\right]}. \tag{11}$$

The solving of whole inner problem then is divided into its sequential solving in windows $\widehat{D}_w$. The system of nonlinear equations for the coefficients $c_{i,j}$ now is constructed for a window and solved in the same way as it was done for the entire region D, because all the values of the coefficients $c_{i,j}$ in the previous windows are already known.

The solution of Equation (1) for one window $\widehat{D}_w$ is the stage’s step.

It is recommended to choose the size of the window K guided by the principle: the computer process of problem solving should use the most of the available computer resources without overfilling them.

Let's designate $\tau_w=wh$ as the beginning of the current window $\widehat{D}_w$. Due to the peculiarities of integration paths in Equation (1), the minimum time coordinate of the known coefficients $c_{i,j}$ until the moment τ_w, which are used to solve the problem in the current window $\widehat{D}_w$, will be calculated by the formula

$$\tau_w^{use}=\tau_w-1 \tag{12}$$

or the same in the index units

$$i_w^{use}=w-m\,. \tag{13}$$

So, at each step, only the results of the n_w previous steps need to be loaded into the RAM, where n_w is calculated by the formula

$$n_w = \lceil m/K \rceil. \quad (14)$$

The same applies to the case of restarting interrupted computations of this stage.

Stages 3 and 4 are just calculations based on Equation (1) with all known coefficients $c_{i,j}$. Since the calculation of the external fields can be performed sequentially for each mesh point outside the inhomogeneity, starting from its boundaries ξ_b, for all $\tau_i \in [0,T]$, such a calculation will be a stage’s step.

Stage 4 has only one step because its result is a only number – the problem calculation error.

V. VERIFICATION

For verification of the developed approach for the cloud-ready fault-tolerant approximation functions method execution and for the error comparison purposes the original paper’s [12] problems of interaction of electromagnetic waves with a dielectric layer was used: the problem of reflection and transmission of a plane electromagnetic wave from a dielectric layer and the problem of passing the Gaussian pulse through the layer with the quadratic nonlinear medium. The simulation parameters were chosen the same as the original work.

Computational resources were specially limited to create conditions when the problem cannot be solved with saving the results of all stages in RAM and in one window on stage 2. Apache Kafka content delivery platform, formed from three nodes cluster, was used as a storage system that is fault-tolerant to a one node failure. The name of saved result of each step of each stage contained the name of the entity it stores and the coordinates of the mesh point. For example: "E0[i, j]" is the name for a value of initial wave in mesh point (i, j). So, data deduplication was done automatically in the client-side application due of these unique names. The correctness of the loaded results was checked inside the application when loading the results of steps.

For the problems under consideration, the following formula was obtained empirically for choosing the optimal window size:

$$K = \lfloor h \cdot M \cdot 5^4 \rfloor, \quad (15)$$

where M is the amount of RAM available for the process in GB.

The developed approach was tested according to the following scheme. After the calculation start, at an arbitrary moment of execution of a random step of each stage: at the beginning, in the middle, or during the saving of the result to the storage, – an error was artificially introduced that interrupted the execution of the program. After that, damages were added also to the saved results of the random step of the random stage and evaluation process was rerun from the start with reload the state from the storage as described above.

The system built using the developed approach showed good stability in loading data and correct continuation of calculations. The results are the same as in the original paper [12], but the margin of error is different. It increased by 0.07% for a plane electromagnetic wave and by 0.73% for a Gaussian pulse. This happened due to the fact that the first problem is for a layer with a linear medium, and the second one with a quadratic nonlinear medium, and the error in Newton method accumulates in different ways.

VI. CONCLUSION

The approach for building cloud-ready fault-tolerant calculations by approximating functions method, as an analytical-numerical component of the Volterra integral equation method, for solving 1D+T nonlinear electromagnetic problems is developed.

The solving process is broken down into the stages, with backward dependency only, with fixed number of steps, each of which is built in fault-tolerant manner and saves execution results in fault-tolerant storage. This allows stopping and starting the algorithm easily after manual or accidental shutdown.

The domain for solving the inner problem is divided into disjoint adjacent subdomains, windows. The solving process of the whole problem then is divided into its sequential solving of sub-problems in these windows. The formula for calculating the window size depending on the modeling step and the amount of computer RAM is shown. Due to the fact that the developed algorithm inside inhomogeneity is always working in a window a certain number of values calculated beforehand are used. The rest is stored on outside the RAM that allows making the calculation step smaller. The formula for determining the exact number of windows that must be loaded from storage to calculating the field inside the inhomogeneity in the current window is given.

Also the reflected and transmitted fields, as well as energy flow and balance, are calculated after the field inside the structure is calculated and its results are stored. This significantly economizes RAM and other computer resources and does not damage the calculated results in the event of a failure.

The original software for computer modeling of the algorithm is developed and the reliability and fault-tolerance of the proposed approach are verified through its stress testing.

Comparison of errors for problems of the transformation of a plane wave and a Gaussian pulse by a layer with nonlinear features is shown. Due to the accumulation of error by Newton method, the presented approach is less accurate than the original one, but shows good results.

VII. MATERIALS AND METHODS

To implement the calculation experiments Wolfram Mathematica was used. The Mathematica document had its own kernel of the mathematical processor, which was allocated its own CPU core for guaranteed independent processing. The connection to the Apache Kafka cluster was performed as described in [14-16]. Cloud technologies of DigitalOcean were chosen as a platform for deployment, where servers based on Ubuntu 20.04 LTS x64 OS were located.

All tests were run at least 10 times for each electromagnetic problem to check the stability of the errors. The median was chosen from all error values.

CONFLICTS OF INTEREST

The author declare that there is no conflict of interest regarding the publication of this article.

REFERENCES

1. A. Taflove, Computational Electrodynamics: The Finite-Difference Time-Domain Method (Boston: Artech House, 1995).
2. W. J. R. Hoefer, "The transmission line method-theory and applications," IEEE Trans. Microwave Theory Tech., vol. 33, pp. 882, 1985.
3. K. S. Yee and J. S. Chen, "The Finite-Difference Time-Domain (FDTD) and the Finite-Volume Time-Domain (FVTD) Methods in Solving Maxwell’s Equations," IEEE Trans. Antennas Propagat., vol. 45, pp. 354–363, 1997.
4. J. N. Reddy, An Introduction to the Finite Element Method (Third ed. (McGraw-Hill, 2005).
5. J. Jin, The Finite Element Method in Electromagnetics (NY: Wiley 1993).
6. J. F. Lee, R. Lee, A. Cangellaris, "Time-Domain Finite-Element Methods," IEEE Trans. Antennas Propagat, vol. 45, pp. 430–442, 1997.
7. Y. Shifman, Y. Leviatan, "On the use of spatio-temporal multiresolution analysis in method of moments solutions of transient electromagnetic scattering," IEEE Trans. on Antennas and Propagation, vol.49, pp. 1123, 2001.
8. M. R. Gomez, A. Salinas, A. R. Bretones, "Time-Domain Integral Equation Methods For Transient Analysis," IEEE Antennas and Propagation Magazine, vol. 34, pp. 15-24, 1992.
9. A. G. Nerukh, N. K. Sakhnenko, T. Benson, P. Sewell, Non-stationary electromagnetics (Singapore: Pan Stanford Publishing Ltd., 2013).
10. A. Nerukh, and T. Benson, Non-Stationary Electromagnetics: An Integral Equations Approach (2nd ed.) (Jenny Stanford Publishing, 2018), https://doi.org/10.1201/9780429027734.
11. D. Zolotariov, A. Nerukh, "Extension of the approximation functions method for 2D nonlinear Volterra integral equations," Applied Radioelectronics, vol. 10, no. 1, pp. 39- 44, 2011.
12. A. Nerukh, D. Zolotariov and T. Benson, "The approximating functions method for nonlinear Volterra integral equations," Opt Quant Electron, vol. 47, pp. 2565–2575, 2015, https://doi.org/10.1007/s11082-015-0141-2.
13. "Apache Kafka," Apache, accessed June 17, 2021), https://kafka.apache.org/documentation/
14. D. Zolotariov, "The mechanism for creation of event-driven applications based on Wolfram Mathematica and Apache Kafka," Innovative Technologies and Scientific Solutions for Industries, vol. 15, no. 1, pp. 53–58, 2021, https://doi.org/10.30837/ITSSI.2021.15.053.
15. D. Zolotariov, "The mechanism for interaction with Apache Kafka" Wolfram Community Forum, 2021, https://community.wolfram.com/groups/-/m/t/2255009.
16. D. Zolotariov, "The platform for creation of event-driven applications based on Wolfram Mathematica and Apache Kafka," Innovative Technologies and Scientific Solutions for Industries, vol. 16, no. 2, pp. 12–18, 2021, https://doi.org/10.30837/ITSSI.2021.16.012.