Distributed-Memory Parallelization of an Explicit Time-Domain Volume Integral Equation Solver on Blue Gene/P

Ahmed Al-Jarro[1], Mark Cheeseman[2], and Hakan Bağcı[1]

[1] Division of Physical Sciences and Engineering
King Abdullah University of Science and Technology, Thuwal, 23955-6900, Saudi Arabia
ahmed.aljarro@kaust.edu.sa, hakan.bagci@kaust.edu.sa

[2] KAUST Supercomputing Laboratory
King Abdullah University of Science and Technology, Thuwal, 23955-6900, Saudi Arabia
mark.cheeseman@kaust.edu.sa

Abstract **─** Two distributed-memory schemes for efficiently parallelizing the explicit marching-on-in-time based solution of the time domain volume integral equation on the IBM Blue Gene/P platform are presented. In the first scheme, each processor stores the time history of all source fields and only the computationally dominant step of the tested field computations is distributed among processors. This scheme requires all-to-all global communications to update the time history of the source fields from the tested fields. In the second scheme, the source fields as well as all steps of the tested field computations are distributed among processors. This scheme requires sequential global communications to update the time history of the distributed source fields from the tested fields. Numerical results demonstrate that both schemes scale well on the IBM Blue Gene/P platform and the memory-efficient second scheme allows for the characterization of transient wave interactions on composite structures discretized using three million spatial elements without an acceleration algorithm.

Index Terms **─** Distributed-memory parallelization, explicit solvers, IBM Blue Gene/P, marching-on-in-time, time domain volume integral equation.

I. INTRODUCTION

Time domain volume integral equation (TDVIE) solvers [1- 6] are becoming an attractive alternative to time domain finite element [7, 8] and finite-difference time domain [8, 9] methods for analyzing transient electromagnetic wave interactions on inhomogeneous dielectric structures. TDVIEs, which relate total electric fields to equivalent polarization source fields/currents induced in the dielectric volume, are constructed using the volume equivalence principle. Often times, marching-on-in-time (MOT) schemes are the method of choice for discretizing the TDVIEs and solving the resulting system of equations. The MOT scheme requires, at each time step, tested (retarded scattered) fields to be computed from the discretized spatial and temporal convolutions of the source fields/currents' time history with the free-space time-domain Green function. Unlike the differential equation based finite element and finite difference time domain methods, the Green function approach requires discretization of only the dielectric volume, avoids the need for absorbing boundary conditions, and virtually eliminates numerical phase dispersion. On the other hand, it renders MOT-TDVIE solvers susceptible to late-time instabilities and significantly increases their computational complexity. The computational cost of the MOT-TDIE solvers has been reduced with the development of the plane wave time domain

Submitted On: Oct. 15, 2011
Accepted On: Jan. 21, 2012

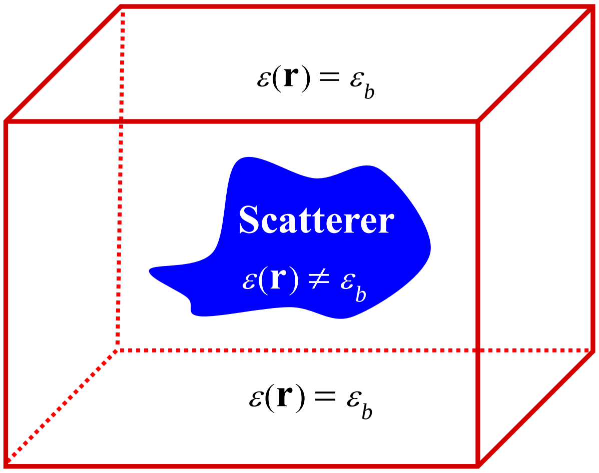

$\varepsilon(\mathbf{r}) = \varepsilon_b$
Scatterer
$\varepsilon(\mathbf{r}) \neq \varepsilon_b$
$\varepsilon(\mathbf{r}) = \varepsilon_b$

Fig. 1. Pictorial description of the volumetric scatterer in a background medium with relative permittivity ε_b.

(PWTD) algorithm [2, 3] and the (blocked) fast Fourier transform (FFT) based schemes [10-13] aimed at accelerating the computation of the discretized spatial and temporal convolutions. The problem of late-time instability has been practically alleviated with the development of implicit MOT schemes [1-3], which make use of accurate temporal interpolation rules [14, 15] and highly accurate (semi-) analytic integration techniques [16, 17].

When compared to the implicit techniques, explicit MOT schemes are less stable but more efficient: they do not require a matrix inversion at every time step and also, as a consequence, they do not suffer from possible ill-conditioning problems [18-20]. Until recently, instability of the explicit MOT-TDVIE solvers has been alleviated using effective but computationally expensive temporal filtering techniques [4]. To eliminate the use of these filtering techniques while maintaining the stability and explicitness of the solver, a predictor-corrector scheme is proposed in [5, 6].

In this work, to allow for the application of this predictor-corrector based TDVIE solver to the analysis of transient electromagnetic wave interactions with electrically large dielectric structures, two distributed-memory schemes are proposed. It should be noted here that the research efforts on the development of parallelization strategies for MOT-TDIE solvers are as recent as the work on the development of PWTD and FFT-based acceleration engines [11, 12, 21]. Especially, efficient distributed-memory parallelization schemes have been as indispensable as the acceleration engines in enabling the use of MOT-TDIE solvers in the analysis of transient electromagnetic wave interactions on electrically large structures.

Before reading the description of the parallelization schemes, one should be reminded here that the predictor-corrector based TDVIE solver parallelized in this work does not pre-compute or store any interaction matrices, which represent discretized retarded field interactions [5, 6]. Since it utilizes a nodal discretization scheme, the computation of the interactions is rather fast and it can be repeated without increasing the MOT time drastically. This approach makes the predictor-corrector based TDVIE solver memory efficient since only the time history of the source fields are stored. It should also be added here that the predictor-corrector based TDVIE solver approximates space and time derivatives, which operate on the scattered field by finite differences. It is well known that in parallel implementations of finite-difference time-domain method, computation of spatial finite differences require “halo” type localized communications between processors [22, 23]. Similar types of communications are needed in parallelized computation of the spatial finite differences in implementations of the TDVIE solver. The cost of these localized communications is much smaller than the global communications needed for parallel computation of the discretized integral present in the retarded field interactions. Therefore, the focus of this work is on comparing the performance of two different distributed-memory parallelization schemes for computing this integral discretized in space and time.

The first parallelization scheme implemented in this work is rather straightforward; each processor stores the time history of all source fields and only the computationally dominant step of the test field computations is distributed among processors. In this scheme, all processors compute the part of the tested fields that is assigned to them from the time history of all the source fields they store. “All-to-all” global communications are needed to update, from the tested fields, the time history of the source fields, which will be used in the computation of the next time step’s tested fields. The second parallelization scheme is slightly more difficult to implement; the source fields as well as all steps of the tested field computations are distributed among the processors. In this scheme, all processors compute in parallel the tested fields of a given processor only from the history of the source fields they store. After this step, tested fields are communicated to the given processor via an ‘all reduce sum’ operation to update the history of the source fields stored on that processor. These steps are repeated in a sequential manner for all processors storing different parts of the tested and source fields.

The two distributed-memory parallelization schemes described above run efficiently on Shaheen, an IBM Blue Gene/P platform, located at the Supercomputing Laboratory of the King Abdullah University of Science and Technology. The IBM Blue Gene/P platform possesses software and hardware optimizations that significantly improve MPI global communication operations, which form the essential core of both parallelization schemes. The hardware support comes in the form of a low latency tree network specifically dedicated to MPI global communication operations. This tree network is a unique defining feature of the IBM Blue Gene/P platform. Additionally, IBM has included its own MPI implementation, which is called DCMF (deep computing message framework) and provides optimized global MPI operations such as “MPI All-to-All” and “MPI Reduce”. All of these optimizations result in substantial performance benefits for the proposed parallelization schemes as shown in Section IV.

Numerical results demonstrate that both schemes scale well on the IBM Blue Gene/P platform and the memory-efficient second scheme allows for characterization of transient electromagnetic wave interactions on dielectric structures discretized using three million spatial elements without any acceleration engine. Additionally, the effectiveness of the parallelized predictor-corrector based TDVIE solver is demonstrated via its application to the characterization of scattering of light from a red blood cell [24-26].

II. FORMULATION

In this section, the formulation and the space-time discretization scheme underlying the predictor-corrector based TDVIE solver is reviewed. For more details on the formulation and the discretization scheme, the reader is referred to [6].

Consider a scatterer compromising potentially inhomogeneous dielectric volumes represented by V with relative permittivity and permeability, $\varepsilon(\mathbf{r})$ and μ (Fig. 1). The scatterer resides in an infinite homogeneous (background) medium with relative permittivity and permeability, ε_b and μ_b. It is assumed that $\varepsilon(\mathbf{r})$, ε_b, μ, and μ_b, are frequency independent and $\mu = \mu_b = 1$. The wave speed in the background medium is given by $c_b = c_0 / \sqrt{\varepsilon_b \mu_b}$, where c_0 is the wave speed in free space. Let $\mathbf{E}_0(\mathbf{r},t)$ represent an incident electric field that is vanishingly small for $\mathbf{r} \in V$ and $t \leq 0$. $\mathbf{E}_0(\mathbf{r},t)$ excites the scatterer; in return the equivalent currents induced in V generate the scattered electric field $\mathbf{E}^{\mathrm{sca}}(\mathbf{r},t)$. Expressing $\mathbf{E}^{\mathrm{sca}}(\mathbf{r},t)$ in terms of equivalent currents, currents in terms of the total electric field $\mathbf{E}(\mathbf{r},t)$, and enforcing the fundamental field electric relation $\mathbf{E}(\mathbf{r},t) = \mathbf{E}_0(\mathbf{r},t) + \mathbf{E}^{\mathrm{sca}}(\mathbf{r},t)$ for $\mathbf{r} \in V$ yields the TDVIE in the unknown $\mathbf{E}(\mathbf{r},t)$, $\mathbf{r} \in V$ [6]:

$$\mathbf{E}(\mathbf{r},t) = \mathbf{E}_0(\mathbf{r},t) + \left[\nabla\nabla \cdot - \frac{\partial_t^2}{c_b^2} \right] \int_V d\mathbf{r}' \mathbf{E}(\mathbf{r}',t') \frac{\varepsilon(\mathbf{r}') - \varepsilon_b}{4\pi\varepsilon_b R}, \ \mathbf{r} \in V. \tag{1}$$

Here, $R = |\mathbf{r} - \mathbf{r}'|$ is the distance between the observation and source points, $\mathbf{r} = \hat{\mathbf{x}}x + \hat{\mathbf{y}}y + \hat{\mathbf{z}}z$ and $\mathbf{r} = \hat{\mathbf{x}}x' + \hat{\mathbf{y}}y' + \hat{\mathbf{z}}z'$ located in V, $t' = t - R/c_b$ is the retarded time, and ∂_t^2 represents the second order partial derivative with respect to time.

Equation (1) is discretized using a nodal discretization scheme to approximate the volume integral and finite differences to approximate the second order derivatives in space and time. Consider a spatial discretization, where cubic elements of dimension Δd are used to divide V into N_e number of elements and a uniform time discretization, where Δt and N_t represent the time step size and the number of total time steps. Electric field is sampled at the centers of the cubic elements at $\mathbf{r} = \mathbf{r}_i$, $i = 1,..,N_e$ and at times $t = t_n = n\Delta t$, $n = 1,..,N_t$. Following the detailed derivation in [6], the final form of the predictor-corrector algorithm is provided below in pseudo-code format:

```
Assume zero initial conditions for all variables
for $n = 1 : N_t$
    $\mathbf{F}(\mathbf{r}_i,t_n) = \sum_{i \neq j}^{N_e} \Delta d^3 \frac{\varepsilon(\mathbf{r}_j)-\varepsilon_b}{4\pi\varepsilon_b R_{ij}} \mathbf{E}(\mathbf{r}_j, t_n - R_{ij}/c_b),\ i=1,...,N_e,$    (2)
    $\mathbf{F}^s(\mathbf{r}_i,t_n) = \mathbf{F}(\mathbf{r}_i,t_n) + \frac{\varepsilon(\mathbf{r}_i)-\varepsilon_b}{4\pi\varepsilon_b} \mathbf{E}(\mathbf{r}_i,t_n) S(\mathbf{r}_i),\ i=1,...,N_e,$    (3)
    $\mathbf{f}(\mathbf{r}_i,t_n) = \tilde{\nabla}\tilde{\nabla}\cdot\mathbf{F}^s(\mathbf{r}_i,t_n) - \frac{\tilde{\partial}_t^2}{c_b^2}\mathbf{F}(\mathbf{r}_i,t_n), i=1,...,N_e,$    (4)
    $\mathbf{E}^p(\mathbf{r}_i,t_n) = \frac{1}{(1+M\Delta t^2)} \big[ 2\mathbf{E}(\mathbf{r}_i,t_{n-1}) - \mathbf{E}(\mathbf{r}_i,t_{n-2}) + M\Delta t^2 \{\mathbf{E}_0(\mathbf{r}_i,t_n) + \mathbf{f}(\mathbf{r}_i,t_n)\} \big],\ i=1,...,N_e,$    (5)
    $\mathbf{E}(\mathbf{r}_i,t_n) = \mathbf{E}^p(\mathbf{r}_i,t_n),\ i=1,...,N_e,$    (6)
    Update $\mathbf{f}(\mathbf{r}_i,t_n)$ for $\forall(\mathbf{r}_i,\mathbf{r}_j) \mid |\mathbf{r}_i-\mathbf{r}_j| < 2c_b\Delta t,$    (7)
    $\mathbf{E}^c(\mathbf{r}_i,t_n) = \frac{1}{(1+M\Delta t^2)} \big[ 2\mathbf{E}(\mathbf{r}_i,t_{n-1}) - \mathbf{E}(\mathbf{r}_i,t_{n-2}) + M\Delta t^2 \{\mathbf{E}_0(\mathbf{r}_i,t_n) + 0.5\mathbf{f}(\mathbf{r}_i,t_n) + 0.5\mathbf{f}(\mathbf{r}_i,t_{n-1})\} \big],\ i=1,...,N_e,$    (8)
    $\mathbf{E}(\mathbf{r}_i,t_n) = \mathbf{E}^c(\mathbf{r}_i,t_n),\ i=1,...,N_e,$    (9)
    Update $\mathbf{f}(\mathbf{r}_i,t_n)$ for $\forall(\mathbf{r}_i,\mathbf{r}_j) \mid |\mathbf{r}_i-\mathbf{r}_j| < 2c_b\Delta t,$    (10)
end for
```

In the algorithm provided above

$$M = \frac{c_b^2 4\pi\varepsilon_b}{[\varepsilon(\mathbf{r}_i)-\varepsilon_b] S(\mathbf{r}_i)}, \tag{11}$$

$R_{ij} = |\mathbf{r}_i - \mathbf{r}_j|$ is the distance between the test and the source points, $\mathbf{r}_i$ and $\mathbf{r}_j$, the singular integral

$$S(\mathbf{r}_i) = \int_{V_i} \frac{d\mathbf{r}'}{|\mathbf{r}_i - \mathbf{r}'|}, \tag{12}$$

where V_i is the support of the i^{th} cubic element, is evaluated analytically as described in [27], and the operators "$\tilde{\nabla}\tilde{\nabla}\cdot$" and "$\tilde{\partial}_t^2$" are finite difference approximations of the continuous operators "$\nabla\nabla\cdot$" and "∂_t^2". Several comments about the above algorithm are in order: (i) Steps (5) and (8) are the predictor and the corrector steps, respectively, and the samples $\mathbf{E}^p(\mathbf{r}_i,t_n)$ and $\mathbf{E}^c(\mathbf{r}_i,t_n)$ are collectively termed "tested fields" while the samples $\mathbf{E}(\mathbf{r}_i,t_n)$ are termed "source fields". (ii) At step (4), $\tilde{\partial}_t^2$ is evaluated using a backward difference formula for test-source point pairs $(\mathbf{r}_i,\mathbf{r}_j)$, which satisfy the condition $|\mathbf{r}_i - \mathbf{r}_j| < 2c_b\Delta t$ and using a central difference formula for all other pairs. At step (7), $\tilde{\partial}_t^2$ is re-evaluated using a central difference formula for test-source point pairs $(\mathbf{r}_i,\mathbf{r}_j)$, which satisfy the condition $|\mathbf{r}_i - \mathbf{r}_j| < 2c_b\Delta t$. The use of central difference is now possible since $\mathbf{E}(\mathbf{r}_i,t_n)$ that were not known at step (4) (due to causality) are replaced by $\mathbf{E}^p(\mathbf{r}_i,t_n)$ obtained at the predictor step. Also note that, at step (10), $\mathbf{f}(\mathbf{r}_i,t_n)$ is "corrected" only for the same test-source point pairs since now $\mathbf{E}(\mathbf{r}_i,t_n)$ are replaced by $\mathbf{E}^c(\mathbf{r}_i,t_n)$ computed at the corrector step. The corrected $\mathbf{f}(\mathbf{r}_i,t_n)$ is used at step (8) of the next time step. This approach increases the accuracy of the finite difference approximations while maintaining the explicitness of the MOT scheme. (iii) When $t_n - R_{ij}/c_b$ is not an integer multiple of Δt, $\mathbf{E}(\mathbf{r}_j, t_n - R_{ij}/c_b)$ is approximated using a linear interpolation between $\mathbf{E}(\mathbf{r}_j,t_n)$ and $\mathbf{E}(\mathbf{r}_j,t_{n-1})$. (v) Note that, in (2), $\mathbf{E}(\mathbf{r}_j, t_n - R_{ij}/c_b)$, which satisfy the condition $t_n - R_{ij}/c_b < 0$ do not contribute to $\mathbf{F}(\mathbf{r}_i,t_n)$ since the fields radiated from the source point $\mathbf{r}_j$ have not yet reached the test point $\mathbf{r}_j$ at time $t = t_n$. (iv) The length of the temporal history of the source fields stored, in terms of time steps, is $\min(n-1, N_g)$; i.e., only $\mathbf{E}(\mathbf{r}_j,t_{n-m})$, $m = 1,...,\min(n-1,N_g)$ are stored. Here, $N_g = \lfloor D_{\max}/c_b\Delta t \rfloor + 2$, where $D_{\max}$ is the maximum distance between any test-source point pair on V. (vi) Unlike the classical MOT schemes, the scheme described above does not pre-compute or store any interaction matrices; the full computation of $\mathbf{f}(\mathbf{r}_i,t_n)$ at step (4) and its correction at steps (7) and (10), in a sense, replaces the multiplication of these interaction matrices with the samples of the source fields' temporal history. (vii) Numerical results presented

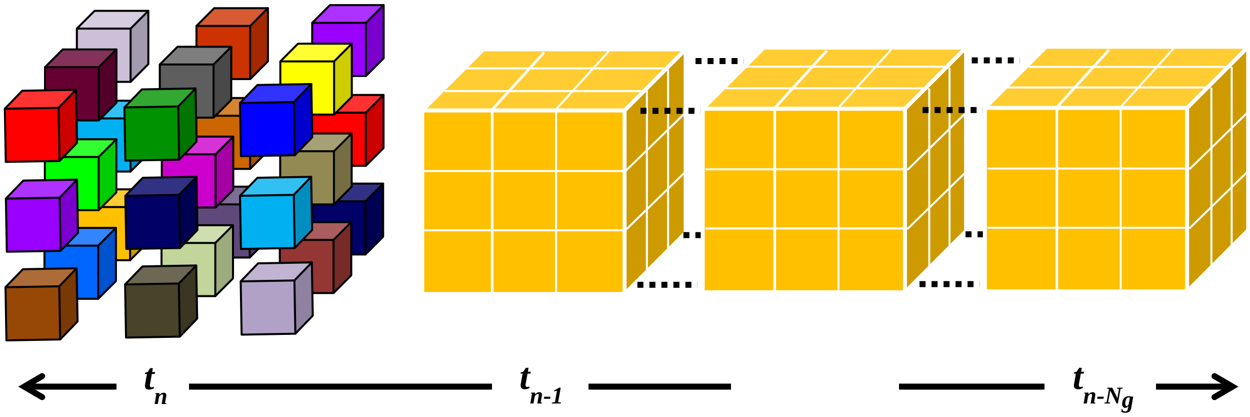


Fig. 2. Pictorial description of scheme 1. Each box of sampling points (represented with a different color) is assigned to a processor. Only computation of $\mathbf{F}(\mathbf{r}_i,t_n)$, $i=1,...,N_e$, at time step t_n is parallelized; every processor computes $\mathbf{F}(\mathbf{r}_i,t_n)$ for all $\mathbf{r}_i$ that reside in its box using the time history of the all source fields, $\mathbf{E}(\mathbf{r}_j,t_{n-m})$, $j=1,...,N_e$, $m=1,...,\min(n-1,N_g)$, which all processors store (represented with solid yellow blocks).

here and in [5, 6] show that the MOT scheme described above is stable when $\varepsilon(\mathbf{r})-\varepsilon_b$ is not large.

III. PARALLELIZATION

The per-time-step computational cost of the predictor-corrector based MOT scheme described in Section II is dominated by the computational cost of evaluating the discrete convolutions $\mathbf{F}(\mathbf{r}_i,t_n)$, $i=1,...,N_e$ at step (2). At the N_g^{th} time step, the fields radiated from all source points $\mathbf{r}_j$, $j=1,...,N_e$ reach all test points $\mathbf{r}_i$, $i=1,...,N_e$. From this time step onwards, the cost of evaluating $\mathbf{F}(\mathbf{r}_i,t_n)$, $i=1,...,N_e$ at time step t_n for $n>N_g$ is $O(N_e^2)$, since all source points interact with all test points. All other operations that are carried out at steps (2)-(10) of the MOT scheme are localized in space and time. Their contribution to the computational cost is very limited especially for large N_e. As a result, the total computational cost of the predictor-corrector based MOT scheme scales as $O(N_t N_e^2)$ under the assumption that $N_t \gg N_g$. This high computational cost could be reduced by integrating PWTD- [2, 3] or blocked FFT-based [10-13] schemes into the predictor-corrector based MOT scheme. Another way of rendering the MOT scheme applicable to the analysis of transient electromagnetic wave interactions on electrically large dielectric structures is through parallelization; which allows for executing the scheme on distributed memory clusters with thousands of cores.

In this section, two distributed-memory parallelization schemes are proposed for accelerating the predictor-corrector based MOT-TDVIE solver. Scheme 1 is a straightforward MPI/OpenMP hybrid parallelization scheme that involves global all-to-all operations; and scheme 2 is a slightly more complex algorithm that involves sequential global reductions. Both schemes are fundamentally aimed at accelerating the computation of $\mathbf{F}(\mathbf{r}_i,t_n)$, $i=1,...,N_e$ at step (2). Detailed descriptions of the schemes are presented next.

A. Scheme 1

The partitioning of the geometry has no effect on the parallelization efficiency of scheme 1. The space sampling points, $\mathbf{r}_i$, $i=1,...,N_e$ can be randomly assigned to processors; as long as they are equally distributed, the scheme will provide the highest efficiency. For the sake of simplicity in the visualization, one can assume that the space sampling points on a rectangular scatterer are assigned to processors as shown in Fig. 2. In this figure, each box of sampling points (shown with a different color on the left) is assigned to a processor.

In this scheme, each processor computes and stores all tested fields, $\mathbf{E}^p(\mathbf{r}_i,t_n)$ and $\mathbf{E}^c(\mathbf{r}_i,t_n)$, $i=1,...,N_e$ at a given time step t_n, and updates and stores the time history of all source fields, $\mathbf{E}(\mathbf{r}_j,t_{n-m})$, $j=1,...,N_e$, $m=1,...,\min(n-1,N_g)$. Only the computation of $\mathbf{F}(\mathbf{r}_i,t_n)$, $i=1,...,N_e$ at step (2) is parallelized. At time step t_n, each

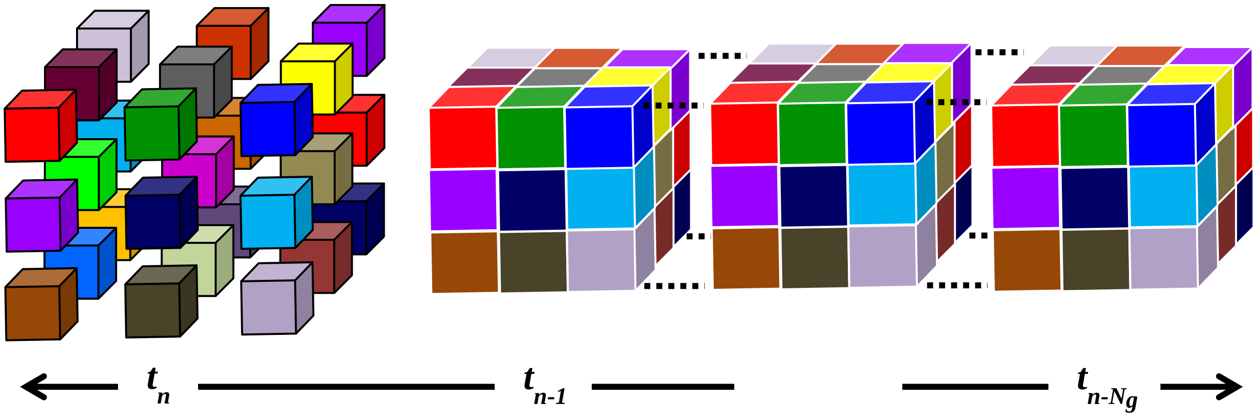


Fig. 3. Pictorial description of scheme 2. Each box of sampling points (represented with a different color) is assigned to a processor. All steps in the computation of tested fields $\mathbf{E}^p(\mathbf{r}_i,t_n)$ and $\mathbf{E}^c(\mathbf{r}_i,t_n)$, $i=1,...,N_e$, at time step t_n is parallelized; every processor stores the tested fields $\mathbf{E}^p(\mathbf{r}_i,t_n)$, $\mathbf{E}^c(\mathbf{r}_i,t_n)$, and the time history of the source fields, $\mathbf{E}(\mathbf{r}_j,t_{n-m})$, $j=1,...,N_e$, $m=1,...,\min(n-1,N_g)$, for all $\mathbf{r}_i$ that reside in its box.

processor independently computes $\mathbf{F}(\mathbf{r}_i,t_n)$ for all $\mathbf{r}_i$ that reside in its box. The parts of $\mathbf{F}(\mathbf{r}_i,t_n)$, $i=1,...,N_e$, which are computed and stored on different processors are communicated to every processor via a global all-to-all operation. At this point, each processor stores $\mathbf{F}(\mathbf{r}_i,t_n)$, for all $i=1,...,N_e$. Thereafter, computations at steps (3), (4), and (5) are replicated on all processors. $\mathbf{E}^p(\mathbf{r}_i,t_n)$, $i=1,...,N_e$ is computed and stored on all processors. At step (6), $\mathbf{E}(\mathbf{r}_i,t_n)$, $i=1,...,N_e$ is updated and stored on all processors. At step (7), each processor updates $\mathbf{f}(\mathbf{r}_i,t_n)$ for every test-source pair $(\mathbf{r}_i,\mathbf{r}_j)$ that satisfies $\left|\mathbf{r}_i-\mathbf{r}_j\right|<2c_b\Delta t$. At step (8), $\mathbf{E}^p(\mathbf{r}_i,t_n)$, $i=1,...,N_e$ is computed and stored on all processors. At step (9), $\mathbf{E}(\mathbf{r}_i,t_n)$, $i=1,...,N_e$ is updated and stored on all processors. Step (10) is the same as step (7).

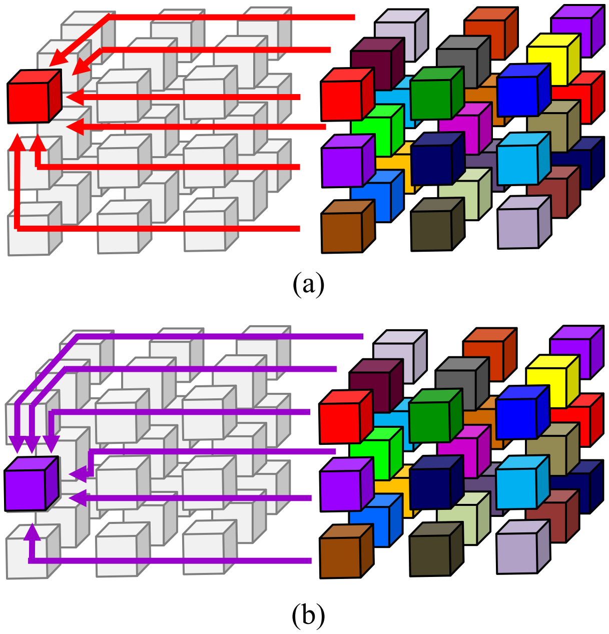
Several remarks about scheme 1 are in order: (i) The scheme is straightforward to implement. The parallelization can be carried out on the serial version of the predictor-corrector based MOT scheme by simply incorporating calls to "MPI_Allgatherv" subroutine after step (2). (ii) Since $\mathbf{E}(\mathbf{r}_j,t_{n-m})$, $j=1,...,N_e$, $m=1,...,\min(n-1,N_g)$ is stored on every processor, the largest spatial discretization that can be handled by this approach is limited by the memory available on a given processor. (iii) Storing $\mathbf{E}^p(\mathbf{r}_i,t_n)$, $\mathbf{E}^c(\mathbf{r}_i,t_n)$, and $\mathbf{F}(\mathbf{r}_i,t_n)$, $i=1,...,N_e$ as well as $\mathbf{E}(\mathbf{r}_j,t_{n-m})$, $j=1,...,N_e$, $m=1,...,\min(n-1,N_g)$ on all processors avoids the local communications needed to compute the finite differences in steps (4), (7), and (10). This simplifies the implementation considerably and provides the flexibility in geometry partitioning mentioned at the beginning of Section III-A. (iv) The extension of the scheme to include shared memory hybridization is straightforward: OpenMP is used throughout, that is in steps (2)-(10) to parallelize the computations when they are distributed over multi-core processors.

B. Scheme 2

Unlike scheme 1, the partitioning of the geometry has an effect on the parallelization efficiency, even though it is small, as explained in the text below. Similar to the description of scheme 1, for the sake of simplicity in the visualization, one can assume that the space sampling points on a rectangular scatterer are assigned to processors as shown in Fig. 3. In this figure, each box of sampling points is assigned to a processor.

In this scheme, each processor stores, for all $\mathbf{r}_i$ that reside in its box, the tested fields, $\mathbf{E}^p(\mathbf{r}_i,t_n)$ and $\mathbf{E}^c(\mathbf{r}_i,t_n)$ at a given time step t_n and the corresponding time history of source fields, $\mathbf{E}(\mathbf{r}_i,t_{n-m})$, $m=1,...,\min(n-1,N_g)$. Note that unlike scheme 1, all steps of the MOT scheme are parallelized. The computation of $\mathbf{F}(\mathbf{r}_i,t_n)$, $i=1,...,N_e$ at step (2) is parallelized in a sequential manner (over processors) that makes use of global reductions. Assume that, at time step t_n, $\mathbf{F}(\mathbf{r}_i,t_n)$ is being computed for all $\mathbf{r}_i\subset RB$, i.e., all sampling points reside in red box (Fig. 4(a) left side); and let P_R represent the processor assigned to RB. Note that each $\mathbf{F}(\mathbf{r}_i,t_n)$ with $\mathbf{r}_i \subset RB$ is a summation of sampled field contributions radiated from all source points, $\mathbf{r}_j$, $j=1,...,N_e$, which are distributed among the processors. Each processor independently computes its contributions to $\mathbf{F}(\mathbf{r}_i,t_n)$, for all $\mathbf{r}_i \subset RB$, from only $\mathbf{E}(\mathbf{r}_j,t_{n-m})$, $m=1,...,\min(n-1,N_g)$ that it stores. Then, these contributions are communicated to P_R via a global reduction call, which is executed with the "sum" flag. At this point, $\mathbf{F}(\mathbf{r}_i,t_n)$ for all $\mathbf{r}_i \subset RB$ is fully computed and stored on P_R. Then, the scheme moves, for example, to the purple box, represented with PB (Fig. 4(b), left side), and repeats the parallel computation of $\mathbf{F}(\mathbf{r}_i,t_n)$, for all $\mathbf{r}_i \subset PB$. This step is repeated processor by processor until $\mathbf{F}(\mathbf{r}_i,t_n)$, for all $\mathbf{r}_i = 1,...,N_e$ is computed and stored part-by-part on all processors. Step (3) is executed independently on every processor without any communications. At step (4), if the computation of $\tilde{\nabla}\tilde{\nabla}\cdot\mathbf{F}(\mathbf{r}_i,t_n)$ requires a (local) finite difference grid that strides across the boundary of two partitioning boxes, then a local communication must occur between the two processors that are assigned to those boxes. At step (5), each processor computes and stores $\mathbf{E}^p(\mathbf{r}_i,t_n)$ for all $\mathbf{r}_i$ that reside in its box. At step (6), each processor updates $\mathbf{E}(\mathbf{r}_i,t_n)$ from $\mathbf{E}^p(\mathbf{r}_i,t_n)$ for all $\mathbf{r}_i$ that reside in its box. At step (7), each processor updates $\mathbf{f}(\mathbf{r}_i,t_n)$ for every test-source pair $(\mathbf{r}_i,\mathbf{r}_j)$ that satisfies $\left|\mathbf{r}_i-\mathbf{r}_j\right| < 2c_b\Delta t$. Similar to step (4), local communications are required. At step (8), each processor computes and stores $\mathbf{E}^c(\mathbf{r}_i,t_n)$ for all $\mathbf{r}_i$ that reside in its box. At step (9), each processor updates $\mathbf{E}(\mathbf{r}_i,t_n)$ from $\mathbf{E}^c(\mathbf{r}_i,t_n)$ for all $\mathbf{r}_i$ that reside in its box. Step (10) is the same as step (7).

Several remarks about scheme 2 are in order: (i) Scheme 2 is slightly more complicated to implement than scheme 1. The global reduction call at step (2) is implemented by incorporating calls to "MPI_Reduce" subroutine with a "sum" flag. (ii) Since, $\mathbf{E}(\mathbf{r}_j,t_{n-m})$, $j=1,...,N_e$, $m=1,...,$ $\min(n-1,N_g)$ are distributed among processors in scheme 2, it is more memory efficient when compared to scheme 1. (iii) The halo type local communications needed at steps (4), (7), and (10) which are also used in many other parallelization schemes [22, 23] have little effect on the scheme's overall parallelization performance. Note that to minimize halo type communications; one may need to find an optimal strategy to partition the geometry.


(a)

(b)

Fig. 4. Pictorial description of the parallelization of the computation of $\mathbf{F}(\mathbf{r}_i,t_n)$, at time step t_n in scheme 2. (a) $\mathbf{F}(\mathbf{r}_i,t_n)$ for all $\mathbf{r}_i$ that reside in the red box (on the left) is computed in parallel. (b) $\mathbf{F}(\mathbf{r}_i,t_n)$ for all $\mathbf{r}_i$ that reside in the purple box is computed in parallel.

IV. NUMERICAL EXPERIMENTS

Scalability tests of the two parallelization schemes proposed in this paper for accelerating the predictor-corrector based MOT TDVIE solver are performed on the IBM Blue Gene/P platform located at Supercomputing Laboratory of the King Abdullah University of Science and Technology. The IBM Blue Gene/P platform, named Shaheen, possesses an IBM design, which was awarded the National Medal of Technology and Innovation in U.S. in 2009. Shaheen has 16384 compute nodes, each of which contains four processing cores and 4 GB shared physical memory. Each processing core runs at a modest clock rate of 850 MHz. However, the addition of a double floating-point unit, an 8MB high speed cache memory, and a fast main memory bandwidth of 13.6 GB/sec raises the peak processing limit of each core to a respectable 13.6 GF/sec. Superior connectivity between the IBM Blue Gene/P platform's compute nodes is

Table 1: Tabulated scaling results for the problem with $N_e = 531441$

$4N_p$	$S^1_{N_p}$	$S^2_{N_p}$
512	0	0
1024	0.9781	0.9865
2048	1.9561	1.9931
4096	2.8087	2.8004
8192	3.6827	3.6994
16384	4.2971	4.8783
32768	4.7659	5.1977

Table 2: Tabulated scaling results for the problem with $N_e = 3048625$

$4N_p$	$S^1_{N_p}$	$S^2_{N_p}$
512	0	0
1024	0.9884	0.9257
2048	1.9783	1.7916
4096	2.9376	2.5422
8192	3.8745	3.4167
16384	4.7459	4.4652
32768	5.5259	5.2762

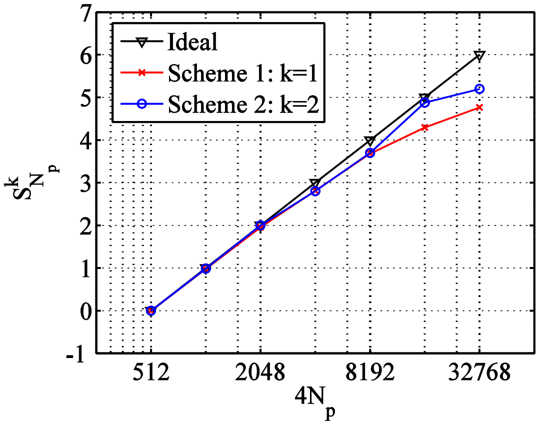


Fig. 5. Scalability of schemes 1 and 2 for a moderate-size problem with $N_e = 531441$.

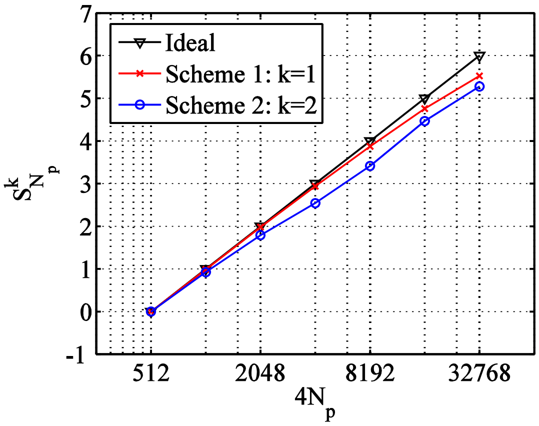


Fig. 6. Scalability of schemes 1 and 2 for a large problem with $N_e = 3048625$.

provided via two dedicated communication networks. A 3D torus network is available for fast point-to-point MPI communication between compute nodes. This network has a peak bandwidth of 5.1 GB/sec and a low latency of only 3.5 microseconds. The second network is dedicated to MPI global communication operations. Every compute node possesses three connections to this tree-based network providing a low latency of 2.5 microseconds per MPI message. Lastly, installed on the IBM Blue Gene/P is a software stack that includes the MPI library called DCMF (deep computing message framework). DCMF library provides optimized versions of normal MPI operations including ‘MPI All-to-All’ and ‘MPI Reduce’, which are heavily optimized to run efficiently on the Blue Gene/P platform including the two dedicated networks interconnecting the compute nodes. It should be emphasized here that the parallelization schemes proposed in this work benefit from the superiority of the second network dedicated to global communications and the optimized DCMF library, since the schemes extensively utilize global MPI communications.

In the remainder of this section, first the weak scalability of the two parallelization schemes are compared, then the effectiveness of the parallelized predictor-corrector based MOT TDVIE solver is demonstrated via its application to the characterization of scattering of light from a red blood cell [24-26].

A. Scalability

Weak scalability for the proposed schemes is investigated here for two scenarios: a moderately sized problem with $N_e = 531441$ and a much larger problem with $N_e = 3048625$. For both examples, scalability results are presented in Tables 1 and 2 and plotted in Figs. 5 and 6. Here, N_p represents the number of compute nodes of the IBM Blue Gene/P platform used in the simulations. This means that for scheme 1, which is hybridized with OpenMP that uses four threads per node, the number of parallel tasks is $4N_p$. Similarly, scheme 2, which is a pure distributed memory implementation, is executed on $4N_p$ cores since each compute node of the IBM Blue Gene/P platform has four processing cores. The weak scaling is defined as $S^k_{N_p} = \log_2(T^k_{N_p} / T^k_{ref})$. Here, the subscript "$N_p$" refers to the simulation carried on N_p nodes, and the superscript "k" refers to the parallelization scheme used. The recorded total times, $T^k_{N_p}$, include both communication and computation times. T^k_{ref} is the reference total time recorded for the simulation that is executed with the lowest N_p. It is clear from Fig. 5 that, for the smaller size problem, the scalability of scheme 1 is hindered by the communication costs when $4N_p$ is larger than 8192. On the other hand, Fig. 5 shows that, for the larger problem, the scalability of scheme 1 carries over even for large values of $4N_p$ around 32768. Figs. 5 and 6 also demonstrate that there is no distinct difference in scheme 2's scalability behaviour for the two problems up to $4N_p = 32768$. Even if the scalability behaviour of the two schemes is different, as clearly demonstrated by the results presented here, they both scale very well on the IBM Blue Gene/P platform.

It should be noted here that, to be able to run the large problem with parallelization scheme 1, N_g was artificially set to a small number. If the actual N_g was used in the simulation, 4GB memory of a single node of the IBM Blue Gene/P platform would not be large enough to store the time history of the source fields. Note that this problem does not exist for the parallelization scheme 2; its memory efficient implementation allows for storing the time history of the source fields using the actual value of N_g.

B. Light scattering from red blood cells

Over the last two decades many biomedical devices utilizing lasers for disease diagnosis have been developed. Consequently, there is an increasing interest in understanding how electromagnetic waves interact with biological cells and tissue. In particular, the analysis of light-scattering from red blood cells (RBCs) have attracted the interest of many researchers [24-26]

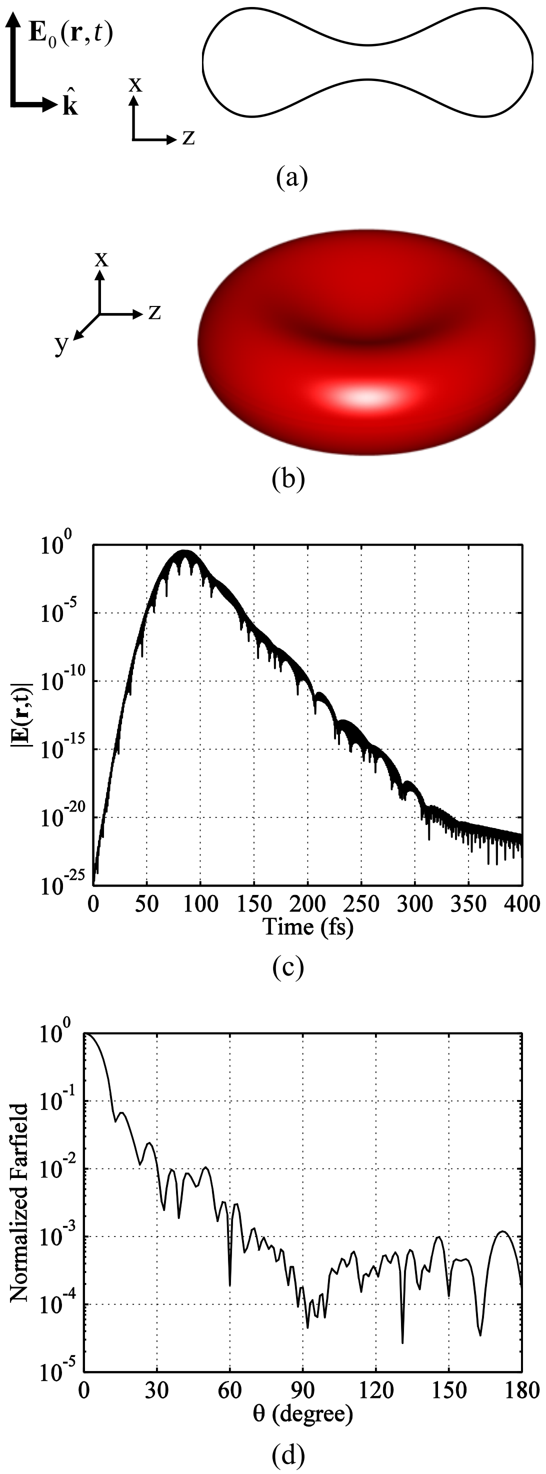


Fig. 7. Analysis of light scattering from an RBC. (a) Cross-section of the RBC model along the xz - plane and the plane wave excitation. (b) Three dimension view of the model. (c) The amplitude of the transient electric field induced at the center of the RBC. (d) Normalized amplitude of the electric far-field on the xz - plane.

since this type of analysis may provide essential information for the diagnosis of blood related diseases [26].

For this example, the parallelized predictor-corrector MOT-TDVIE solver is used to analyze light scattering from an RBC residing in plasma. The membrane of RBCs has a negligible effect on the scattered field [24], and hence the RBC model does not include the membrane or any other internal structure and is formulated as a biconcave volume as described in [24]. Figures 7 (a) and (b) depict the xz-cross section and three-dimensional view of this model, respectively. The diameter of the cell is $7.82\ \mu\text{m}$, its largest and smallest thickness values on xz-plane are $2.565\ \mu\text{m}$ and $0.81\ \mu\text{m}$, respectively, producing a volume of $94\ \mu\text{m}^3$. The relative permittivities of the RBC and the background plasma are $\varepsilon(\mathbf{r}) = 1.9768$ and $\varepsilon_b = 1.8089$, respectively [24]. Note that while absorption can also be handled by the implemented MOT-TDVIE scheme, for the excitation frequencies considered, it is very small and thus is neglected [24, 25]. The excitation is an $\hat{\mathbf{x}}$ polarized plane wave with a modulated Gaussian time signature, which propagates in the $\hat{\mathbf{z}}$ direction; the incident electric field is then expressed as

$$\mathbf{E}_0(\mathbf{r},t) = \hat{\mathbf{x}} E_0 G(t - z/c_b), \tag{13}$$

$$G(t) = \cos\left[2\pi f_o (t - t_p)\right] \exp\left[-(t - t_p)^2 / \varsigma^2\right], \tag{14}$$

where $E_0 = 1$ V/m is the electric field amplitude and $G(t)$ represents a Gaussian pulse, $f_o = 637.2$ THz, $\varsigma = 3/(2\pi f_{bw})$, $f_{bw} = f_0/15$, and $t_p = 6\varsigma$ are its modulation frequency, duration, bandwidth, and delay, respectively. The grid space is chosen as $\Delta d = 0.045\ \mu\text{m}$, which produces $N_e = 1031550$ discretization elements within the volume of the RBC. The time step $\Delta t = 0.15$ fs and the simulation is carried out for $N_t = 2666$ time steps.

The amplitude of the electric field at the center of the RBC, $\mathbf{E}(\mathbf{r}_o, t)$, $\mathbf{r}_o = (0,0,0)$, which is recorded during the simulation, is presented in Fig. 7(c); the figure clearly demonstrates the late-time stability of the scheme. Additionally, frequency-domain scattered farfields on the xz-plane are computed. For this purpose, at $f = 473.8$ THz ($\lambda = 0.6328 \mu m$, in the plasma), the Fourier transform of the currents induced in the RBC volume is computed during time marching using a running discrete Fourier transform (DFT) summation. Note that the DFT of the currents is normalized by the Fourier transform of $G(t)$ to produce the time-harmonic currents. Then, as a post-processing step, the farfields are easily computed from the frequency-domain currents. Figure 7(d) plots the normalized amplitude of the electric farfield on the xz-plane. The results agree well with those generated in [24] using the finite difference time domain method.

V. CONCLUSIONS

Two distributed-memory schemes are proposed to efficiently parallelize the predictor-corrector based MOT-TDVIE solver on the IBM Blue Gene/P platform. The first scheme distributes the computationally dominant step of the tested field computations among the processors using the MPI standard. To achieve an easy-to-implement and highly-scalable parallelization scheme, the time history of the source fields are stored simultaneously on all processors. Within each multi-core processor, OpenMP standard is used to further accelerate the computation of the tested fields. Obviously, the fundamental limitation of this scheme is its high memory requirement due to the storage of the time history of all the source fields on each processor.

The second scheme alleviates this limitation by distributing the time history of the source fields as well as all steps of the tested field computations among the processors. Even though the implementation of scheme 2 is slightly more complicated than scheme 1, numerical results demonstrate that scheme 2 scales as well as scheme 1 on the IBM Blue Gene/P platform. The memory efficient scheme 2 allows the predictor-corrector based MOT-TDVIE solver to simulate transient electromagnetic wave interactions on electrically large structures discretized using more than three million spatial elements.

It should be emphasized here again that both methods benefit from the IBM Blue Gene/P platform's superior tree network dedicated to global communications and optimized MPI library since they heavily utilize "MPI_Allgatherv" and "MPI_Reduce" subroutines. Other computing platforms do not have this level of hardware and software support for MPI global communications. Development of the extensions to the method proposed here, which would scale well on other platforms, is underway.

ACKNOWLEDGEMENT

The authors would like to thank the King Abdullah University of Science and Technology Supercomputing Laboratory for providing the required computational resources.

This work was supported in part by an Academic Excellence Alliance program awarded from the King Abdullah University of Science and Technology Global Collaborative Research under the title “Energy Efficient Photonic and Spintronic Devices”.

REFERENCES

[1] N. T. Gres, A. A. Ergin, E. Michielssen, and B. Shanker “Volume-Integral-Equation-Based Analysis of Transient Electromagnetic Scattering from Three-Dimensional Inhomogeneous Dielectric Objects,” *Radio Sci.*, vol. 36, no. 3, pp. 379-386, May 2001.

[2] B. Shanker, K. Aygun, and E. Michielssen, “Fast Analysis of Transient Scattering from Lossy Inhomogeneous Dielectric Bodies,” *Radio Sci.*, vol. 39, pp. 1-14, Mar. 2004.

[3] G. Kobidze, J. Gao, B. Shanker, and E. Michielssen, “A Fast Time Domain Integral Equation Based Scheme for Analyzing Scattering from Dispersive Objects,” *IEEE Trans. Antennas Propag.*, vol. 53, no. 3, pp. 1215-1226, Mar. 2005.

[4] Al-Jarro, P. Sewell, T. M. Benson, A. Vukovic, and J Paul, “Transient Time-Dependent Electric Field of Dielectric Bodies using the Volterra Integral Equation in Tthree Dimensions,” *Progress Electromag. Res.*, vol. 110, pp. 179-197, 2010.

[5] A. Al-Jarro and H. Bagci, “A Predictor-Corrector Scheme for Solving the Volterra Integral Equation,” *The XXX General Assembly and Scientific Symposium of URSI*, Istanbul, Aug. 2011.

[6] A. Al-Jarro, M. A. Salem, H. Bagci, T. M. Benson, P. Sewell, and A. Vukovic, “Explicit Solution of the Time Domain Volume Integral Equation using a Stable Predictor-Corrector Scheme,” submitted for publication, 2011.

[7] P. P. Silvester and R. L. *Ferrari, Finite Elements for Electrical Engineers*. Cambridge, U.K, Cambridge University Press, 1990.

[8] F. L. Teixeira, “A Summary Review on 25 Years of Progress and Future Challenges in FDTD and FETD Techniques,” *Applied Computational Electromagnetics Society (ACES) Journal*, vol. 25, no. 1, pp. 1-14, Jan. 2010.

[9] A. Taflove and Susan C. Hagness, *Computational Electrodynamics: The Finite Difference Time Domain Method*, Artech House, 2005.

[10] A. E. Yilmaz, J. M. Jin, and E. Michielssen, “Time Domain Adaptive Integral Method for Surface Integral Equations,” *IEEE Trans. Antennas Propag.*, vol. 52, no. 10, pp. 2692-2708, Oct. 2004.

[11] A. E. Yilmaz, J. M. Jin, and E. Michielssen, “A Parallel FFT Accelerated Transient Field-Circuit Simulator,” *IEEE Trans. Microw. Theory Tech.*, vol. 53, no. 9, pp. 2851-2865, Sep. 2005.

[12] H. Bagci, A. E. Yilmaz, J.-M. Jin, and E. Michielssen, “Fast and Rigorous Analysis of EMC/EMI Phenomena on Electrically Large and Complex Cable-Loaded Structures,” *IEEE Trans. Electromagn. Comp.*, vol. 49, no. 9, pp. 361-381, May 2007.

[13] H. Bagci, A. E. Yilmaz, and E. Michielssen, “An FFT-Accelerated Time-Domain Multiconductor Transmission Line Simulator,” *IEEE Trans. Electromagn. Comp.*, vol. 52, no. 1, pp. 199-214, Feb. 2010.

[14] G. Manara, A. Monorchio, and R. Reggiannini, “A Space-Time Discretization Criterion for a Stable Time-Marching Solution of the Electric Field Integral Equation,” *IEEE Trans. Antennas Propag.*, vol. 45, no. 3, pp. 527-532, Mar. 1997.

[15] D. S. Weile, G. Pisharody, N.-W. Chen, B. Shanker, and E. Michielssen, “A Novel Scheme for the Solution of the Time-Domain Integral Equations of Electromagnetics,” *IEEE Trans. Antennas Propag.*, vol. 52, no.1, pp. 283-295, Jan. 2004.

[16] Y. Shi, M. Xia, R. Chen, E. Michielssen, and M. Lu, “Stable Electric Field TDIE Solvers via Quasi-Exact Evaluation of MOT Matrix Elements,” *IEEE Trans. Antennas Propag.*, vol. 59, no. 2, pp. 574-585, Feb. 2011.

[17] H. A. Ulku and A. A. Ergin, “Application of Analytical Retarded-Time Potential Expressions to the Solution of Time Domain Integral Equations,” *IEEE Trans. Antennas Propag.*, vol. 59, no. 11, pp. 4123- 4131, Nov. 2011.

[18] F. P. Andriulli, H. Bagci, F. Vipiana, G. Vecchi, and E. Michielssen, “A Marching-on-in-Time Hierarchical Scheme for the Time Domain Electric Field Integral Equation,” *IEEE Trans. Antennas Propag.*, vol. 55, no. 12, pp. 3734-3738, Dec. 2007.

[19] F. P. Andriulli, H. Bagci, F. Vipiana, G. Vecchi, and E. Michielssen, “Analysis and Regularization of the TD-EFIE Low-Frequency Breakdown,”

IEEE Trans. Antennas Propag., vol. 57, no. 7, pp. 2034-2046, July 2009.

[20] H. Bagci, F. P. Andriulli, F. Vipiana, G. Vecchi, and E. Michielssen, "A Well-Conditioned Integral-Equation Formulation for Efficient Transient Analysis of Electrically Small Microelectronic Devices," *IEEE Trans. Adv. Packag.*, vol. 33, no. 2, pp. 468-480, May 2010.

[21] F. Wei and A. E. Yilmaz, "A Hybrid Message Passing/Shared Memory Parallelization of the Adaptive Integral Method for Multi-Core Clusters," *Parallel Comp.*, vol. 37, no. 6-7, pp. 279-301, June-July 2011.

[22] X. Duan, X. Chen, K. Huang, and H. Zhou, "A High Performance Parallel FDTD Based on Winsock and Multi-Threading on a PC-Cluster," *Applied Computational Electromagnetics Society (ACES) Journal*, vol. 26, no. 3, pp. 241-249, March 2011.

[23] J. E. Lump, S. K. Mazumder, and S. D. Gedney, "Performance Modeling of the Finite-Difference Time-Domain Method on Parallel Systems," *Applied Computational Electromagnetics Society (ACES) Journal*, vol. 19, no. 2, pp. 147-159, July 1998.

[24] J. He, A. Karlsson, J. Swartling, and S. Andersson-Engels, "Light Scattering by Multiple Red Blood Cells," *J. Opt. Soc. Am. A*, vol. 21, no. 10, pp. 1953-1961, Oct. 2004.

[25] J. Q. Lu, P. Yang, and X.H. Hu, "Simulations of Light Scattering from a Biconcave Red Blood Cell using the Finite-Difference Time-Domain Method," *J. Biomed Opt.*, vol. 10, no. 2, pp. 024022-10, Mar./Apr. 2005.

[26] O. Ergul, A. Arslan-Ergul, and L. Gurel, "Computational Study of Scattering from Healthy and Diseased Red Blood Cells using Surface Integral Equations and the Multilevel Fast Multipole Algorithm," *J. Biomed. Opt.*, vol. 15, no. 4, pp. 045004-8, July/Aug. 2010.

[27] J. Waldvogel, "The Newtonian Potential of Homogeneous Cube," *J. Applied Math. Phys.*, vol. 27, no. 6, pp. 867-871, 1979.

Ahmed Al-Jarro received the B.Eng. degree in Electronic Engineering with Spanish and the Ph.D. degree in Electrical and Electronic Engineering from the University of Nottingham, UK, in 2001 and 2004, respectively. From 2004 to 2009, he was a Research Assistant and Research Fellow at the George Green Institute for Electromagnetics Research (GGIEMR), University of Nottingham, UK. In 2010, he was awarded with the Knowledge Transfer Secondment from the University of Nottingham, UK, in collaboration with Photon Design Ltd., Oxford, UK. In 2011, he joined the Division of Physical Sciences and Engineering at King Abdullah University of Science and Technology (KAUST), Saudi Arabia, as a Postdoctoral Research Fellow. Dr. Al-Jarro was the recipient of the Japan Society for the Promotion of Science, JSPS, Fellowship Award in 2010.

Dr. Al-Jarro's research interests are in the field of computational electromagnetics for the analysis of photonic and optical devices. He is currently working on the development of explicit and stable marching-on-in-time based schemes for solving the time domain volume integral equation, and their efficient parallelization.

Mark Cheeseman received the B.Sc. degree in Physics from the Memorial University of Newfoundland, Canada, in 1999 and the M.Sc. degree in Earth and Atmospheric Science from the University of Alberta, Canada, in 2003. He has been a computational scientist at the Supercomputing Laboratory of the King Abdullah University of Science and Technology (KAUST), Saudi Arabia, since 2009. He possesses over eight years of experience at various HPC centers; including the Swiss National Supercomputing Center, Switzerland, and the National Oceanography Center in Southampton, UK.

The majority of his work concerns the optimization and re-engineering of existing scientific applications for efficient use on massively parallel platforms. These applications range from combustion models, to acoustic wave simulation, and distributed 3D Fast Fourier Transform libraries. His technical interests are particularly focused on IO optimization and the scalability of present-day mathematical libraries.

Hakan Bağcı received the B.Sc. degree in Electrical and Electronics Engineering from the Bilkent University, Ankara, Turkey, in 2001 and the M.Sc. and Ph.D. degrees in Electrical and Computer Engineering from the University of Illinois at Urbana-Champaign (UIUC), Urbana, in 2003 and 2007, respectively. From June 1999 to July 2001, he worked as an Undergraduate Researcher at the Computational Electromagnetics Group, Bilkent University. From 2001 to 2007, he was a Research Assistant at the Center for Computational Electromagnetics and Electromagnetics Laboratory, UIUC. From 2007 to 2009, he worked as a Research Fellow at the Radiation Laboratory, University of

Michigan. In 2009, he joined the Division of Physical Sciences and Engineering at the King Abdullah University of Science and Technology (KAUST), Saudi Arabia, as an Assistant Professor.

His research interests include various aspects of computational electromagnetics with emphasis on time-domain integral equations and their fast marching-on-in-time based solutions, well-conditioned integral-equation formulations, and development of fast hybrid methods for analyzing statistical EMC/EMI phenomena on complex and fully loaded platforms.

Dr. Bağcı was the recipient of the 2008 International Union of Radio Scientists (URSI) Young Scientist Award and the 2004–2005 Interdisciplinary Graduate Fellowship from the Computational Science and Engineering Department, UIUC. His paper titled "Fast and rigorous analysis of EMC/EMI phenomena on electrically large and complex structures loaded with coaxial cables" was one of the three finalists (with honorable mention) for the 2008 Richard B. Schulz Best Transactions Paper Award given by the IEEE Electromagnetic Compatibility Society. He authored and co-authored three finalist papers and another paper, which is awarded honorable mention, in the student paper competitions at the 2005, 2008, and 2010, IEEE Antennas and Propagation Society International Symposiums.