Progress In Electromagnetics Research, PIER 57, 237–252, 2006

A PARALLELIZED 3D FLOATING RANDOM-WALK ALGORITHM FOR THE SOLUTION OF THE NONLINEAR POISSON-BOLTZMANN EQUATION

K. Chatterjee †

Department of Electrical and Computer Engineering
Cooper Union
New York, NY 10003-7120, USA

J. Poggie

Computational Sciences Center
Air Force Research Laboratory
Wright-Patterson Air Force Base
OH 45433-7512, USA

Abstract—This paper presents a new three-dimensional floating random-walk (FRW) algorithm for the solution of the Nonlinear Poisson-Boltzmann (NPB) equation. The FRW method has not been previously used in the numerical solution of the NPB equation (and other nonlinear equations) because of the non-availability of analytical expressions for volumetric Green's functions. In the past, numerical studies using the FRW method have examined only the linearized Poisson-Boltzmann equation, producing solutions that are only accurate for small values of the potential. No such linearization is required for this algorithm. An approximate expression for a volumetric Green's functions has been calculated with the help of a novel use of perturbation theory, and the resultant integral form has been incorporated within the FRW framework. The algorithm requires no discretization of either the volume or the surface of the problem domains, and hence the memory requirements are expected to be lower than approaches based on spatial discretization, such as finite-difference methods. Another advantage of this algorithm is that each random walk is independent, so that the computational procedure is inherently parallelizable and an almost linear increase in computational speed is expected with increase in the number of processors. We have

† Adjunct Scientist at Laboratory for Electromagnetic and Electronic Systems, Massachusetts Institute of Technology, Cambridge, MA 02139-4307, USA

recently published the preliminary results for benchmark problems in one and two dimensions. In this work, we present our results for benchmark problems in three dimensions and demonstrate excellent agreement between the FRW- and finite-difference based algorithms. We also present the results of parallelization of the newly developed FRW algorithm. The solution of the NPB equation has applications in diverse branches of science and engineering including (but not limited to) the modeling of plasma discharges, semiconductor device modeling and the modeling of biomolecular structures and dynamics.

1. INTRODUCTION

The solution of the NPB equation has widespread applications in science and engineering. These applications range from the modeling of the plasma-sheath transition [1], semiconductor device modeling [2] and the modeling of biomolecular structures and dynamics [3]. In this paper, we address the efficient solution of the NPB equation by developing a stochastic algorithm based on the FRW method [4–6].

The FRW method is based on probabilistic interpretations of deterministic equations. The method is completely meshless and requires no discretization of either the volume or the surface of problem domains. As a result, the memory requirements for complicated problem geometries are expected to be significantly lower than for methods based on spatial discretization. Furthermore, each random walk is independent, so that the method is inherently parallelizable. In spite of its many advantages, the FRW method has not been applied in the numerical solution of NPB equation (and other important nonlinear equations) because analytical expressions for volumetric Green's functions [7] are not available. In particular, previous numerical studies using FRW algorithms [8, 9] have examined the linearized Poisson-Boltzmann equation, restricting the applicability of the solution to small values of the potential. Here we present a new technique that eliminates this restriction. We have previously presented the results for a one-dimensional [10] and a two- dimensional [11] floating random-walk algorithm for the NPB equation subject to Dirichlet [7] boundary conditions. In this paper, we extend this algorithm to three-dimensional problem geometries and present a detailed formulation, validation with finite-difference benchmarks, and parallelization of this newly developed algorithm. But before presenting the specifics of the algorithm, we will give an overview of the FRW method.

2. OVERVIEW OF THE FRW METHOD

The FRW method is a generalization of the Monte Carlo integration method [12], a statistical approach to estimating integrals. Given a differential equation, with a differential operator L,

$$L\left[U(\mathbf{r})\right] = f(\mathbf{r}), \tag{1}$$

the solution $U(\mathbf{r})$ is a function of the three-dimensional position vector $\mathbf{r}$. The function $f(\mathbf{r})$ is a source term. The Green's functions for Eq. (1) are the solutions of the differential equation

$$L\left[G(\mathbf{r}|\mathbf{r}_o)\right] = \delta(\mathbf{r} - \mathbf{r}_o), \tag{2}$$

subject to specified boundary conditions. We assume that the operator L is of the Sturm-Liouville [7] form: $L = \nabla \cdot [p(\mathbf{r})\nabla] + q(\mathbf{r})$, where $p(\mathbf{r})$ and $q(\mathbf{r})$ are known functions of $\mathbf{r}$. Using Green's integral representation [7], $U(\mathbf{r})$ can be written as

$$\begin{aligned} U(\mathbf{r}_o) &= \iiint_V dv G(\mathbf{r}|\mathbf{r}_o) f(\mathbf{r}) - \oiint_S \left[d\mathbf{s} \cdot \nabla_{\mathbf{r}} U(\mathbf{r})\right] p(\mathbf{r}) G(\mathbf{r}|\mathbf{r}_o) \\ &\quad + \oiint_S \left[d\mathbf{s} \cdot \nabla_{\mathbf{r}} G(\mathbf{r}|\mathbf{r}_o)\right] p(\mathbf{r}) U(\mathbf{r}). \end{aligned} \tag{3}$$

The first term on the right hand side of Eq. (3) is a volume integral involving the source term in the entire volume V of interest. The second and third terms are vector surface integrals over the surface S enclosing V, where $d\mathbf{s}$ is a vector whose magnitude is equal to that of an infinitesimally small area unit on the surface S and directed normally outward from the center of the area unit. The term $G(\mathbf{r}|\mathbf{r}_o)$ is often referred to as the volume Green's function and the term $\nabla_{\mathbf{r}} G(\mathbf{r}|\mathbf{r}_o)$ is called the surface Green's function. The second integral corresponds to the Neumann [7] boundary condition while the third integral corresponds to the Dirichlet boundary condition.

Eq. (3) forms the mathematical basis of the FRW method. To evaluate the solution to Eq. (1) at a particular point in the domain of interest, we consider [4–6] maximal spheres, cubes, or any geometrical object for which the solution to Eq. (2) is known. We then make random hops to the surface of that geometrical object based on any predefined probability density. The weights for such random hops are determined by sampling the various integrands in Eq. (3). For example, in the case of a Dirichlet problem with no source term [that is, $f(\mathbf{r}) = 0$], the problem reduces to a Monte Carlo [12] integration of

an infinite-dimensional integral as given by:

$$\begin{aligned} U(\mathbf{r}_o) &= \oint_{S_1} d\mathbf{s}_1 K(\mathbf{r}_o|\mathbf{r}_1) \oint_{S_2} d\mathbf{s}_2 K(\mathbf{r}_1|\mathbf{r}_2) \\ &\quad \cdots\cdots \oint_{S_n} d\mathbf{s}_n K(\mathbf{r}_{n-1}|\mathbf{r}_n) U(\mathbf{r}_n), \\ K(\mathbf{r}_{n-1}|\mathbf{r}_n) &= p(\mathbf{r}_n) \left|\nabla_{\mathbf{r}_n} G\left(\mathbf{r}_{n-1}|\mathbf{r}_n\right)\right| \cos\left(\gamma_{n-1,n}\right), \end{aligned} \tag{4}$$

where $\gamma_{n-1,n}$ is the angle between $\nabla_{\mathbf{r}_n} G\left(\mathbf{r}_{n-1}|\mathbf{r}_n\right)$ and $d\mathbf{s}_n$. The successive surface integrals in Eq. (4) relate to successive random hops across the problem domain and the weight factors of the form $K\left(\mathbf{r}_{n-1}|\mathbf{r}_n\right)$ are derived from the third integral term on the right hand side of Eq. (3) that corresponds to the Dirichlet boundary condition. A particular random-walk is terminated at the boundary where the solution is known and the successive weight factors multiplied by the solution at the boundary yields a particular sample of the solution. A numerical solution of Eq. (1) is obtained by averaging over a statistically large number of such samples. A schematic diagram of circular random-walks on a circular problem domain is shown in Fig. 1.

At this point, we observe that this method does not require

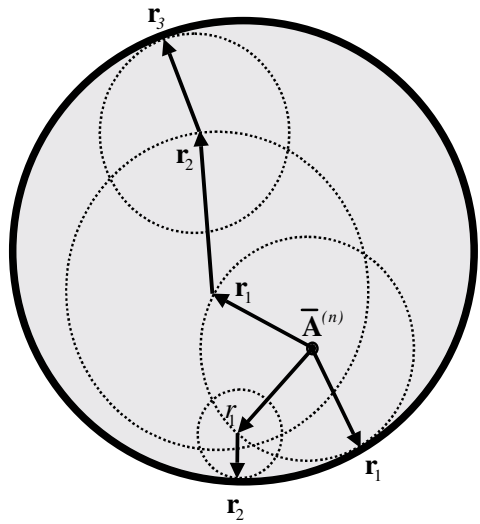

Figure 1. Schematic diagram of circular random-walks on a circular problem domain. One-, two- and three-hop random-walks are represented.

any discretization because the solution can be evaluated at the point of origination of the random walks irrespective of the solution at any other point. We can also note that this method is inherently parallelizable as different random-walks can be performed on different processors and inter-processor communication is required only during the final averaging of the contributions from different walks. In spite of these unique advantages, the floating random-walk method has not being applied to the NPB equation and other important nonlinear equations. This is due to the absence of corresponding analytical expressions for volumetric Green's functions. Early researchers in the area expressed the apprehension that the extension of the stochastic solution methodology to nonlinear problems might not be possible. In a 1954 paper [13], J. R. Curtiss wrote: "So far as the author is aware, the extension of Monte Carlo methods to non-linear processes has not yet been accomplished and may be impossible." Stochastic approaches to solving nonlinear equations (in particular the NPB equation) that have been suggested in literature [14] involve an iterative solution of a series of linear problems. In our proposed approach, an approximate (yet accurate) expression for the Green's function for the nonlinear problem is obtained through perturbation theory, which gives rise to an integral formulation that is valid for the entire nonlinear problem. As a result, our algorithm does not have any iteration steps, and thus has a lower computational cost. The validity of such an integral expression is maintained by restricting the size of a random hop. Increasing the order of perturbation in the Green's function would allow one to increase the hop size, thus increasing computational speed. An approach utilizing a perturbation-based Green's function was used to develop an FRW algorithm for the Helmholtz equation in heterogeneous problem domains (important for IC interconnect analysis at high frequencies) by the first author in Ref. [15, 16], where the idea of extending the approach to nonlinear problems was proposed. Later that idea was extended to the NPB equation in one and two dimensions [10, 11]. In this work, a three-dimensional volumetric Green's function truncated to the first order (with a correspondingly restricted hop size) has been calculated and will be presented in the next section.

3. FORMULATION OF THE ALGORITHM

In our problem of interest, the dependent variable ϕ is governed by the NPB equation given as

$$\nabla^2\phi = \frac{1}{c^2}\left(e^{k\phi(\mathbf{r})} - e^{-k\phi(\mathbf{r})}\right), \quad \mathbf{r} \in W, \tag{5}$$

where $\mathbf{r}(r, \theta, \varphi)$ is the three-dimensional position coordinate, c and k are constants, and W is the three-dimensional problem domain. Dirichlet boundary conditions have been imposed:

$$\phi = g(\mathbf{r}), \quad \mathbf{r} \in \partial W \tag{6}$$

where ∂W is the boundary of the domain W. Eq. (5) can be normalized to

$$\frac{1}{\hat{r}^2}\frac{\partial}{\partial \hat{r}}\left(\hat{r}^2\frac{\partial \hat{\phi}}{\partial \hat{r}}\right)\frac{1}{\hat{r}^2\sin\hat{\theta}}\frac{\partial}{\partial \hat{\theta}}\left(\sin\hat{\theta}\frac{\partial \hat{\phi}}{\partial \hat{\theta}}\right) + \frac{1}{\hat{r}^2\sin^2\hat{\theta}}\frac{\partial^2 \hat{\phi}}{\partial \hat{\varphi}^2} = e^{\hat{\phi}} - e^{-\hat{\phi}}, \tag{7}$$

where $\hat{r} = r/\lambda$, $\hat{\theta} = \theta$ and $\hat{\varphi} = \varphi$; $\hat{\phi} = k\phi$, $\lambda = \frac{c}{\sqrt{k}}$. The value of the solution is assumed to be known on the surface encompassing the problem domain of interest. We will first derive an approximate expression for a volumetric Green's function $G(\hat{\mathbf{r}}|\hat{\mathbf{r}}_o)$ for Eq. (7) on a spherical domain, given a Dirac-delta function centered at $\hat{\mathbf{r}}_o$ inside the domain, and boundary conditions such that the Green's function on the surface of the sphere is zero. Such a Green's function is given as the solution of the equation

$$\nabla^2 G\left(\hat{\mathbf{r}}|\hat{\mathbf{r}}_o\right) - \left(e^{G} - e^{-G}\right) = \delta\left(\hat{\mathbf{r}} - \hat{\mathbf{r}}_o\right) \tag{8}$$

We further normalize the length scales to the radius R of the spherical domain and substitute $\hat{\rho} = \frac{\hat{r}}{R}$ and $\hat{\rho}_o = \frac{\hat{r}_o}{R}$ in Eq. (7). The twice-normalized Green's function equation is written as

$$\nabla^2_{\hat{\boldsymbol{\rho}}} G - R^2\left(e^{G} - e^{-G}\right) = \delta\left(\hat{\boldsymbol{\rho}} - \hat{\boldsymbol{\rho}}_o\right). \tag{9}$$

A zeroth-order approximation for the volumetric Green's function is the solution of equation

$$\nabla^2_{\hat{\boldsymbol{\rho}}} G^{(0)}\left(\hat{\boldsymbol{\rho}}|\hat{\boldsymbol{\rho}}_o\right) = \delta\left(\hat{\boldsymbol{\rho}} - \hat{\boldsymbol{\rho}}_o\right), \tag{10}$$

which is given as [7]

$$\begin{aligned} G^{(0)}\left(\hat{\boldsymbol{\rho}}|\hat{\boldsymbol{\rho}}_o\right) &= \frac{1}{4\pi}\left[\frac{1}{\{1+\hat{\rho}^2\hat{\rho}_o^2-2\hat{\rho}\hat{\rho}_o C\}^{\frac{1}{2}}} - \frac{1}{\{\hat{\rho}^2+\hat{\rho}_o^2-2\hat{\rho}\hat{\rho}_o C\}^{\frac{1}{2}}}\right], \\ C &= \cos\hat{\theta}\cos\hat{\theta}_o + \sin\hat{\theta}\sin\hat{\theta}_o\cos\left(\hat{\varphi} - \hat{\varphi}_o\right). \end{aligned} \tag{11}$$

Eq. (11) can be used to obtain a first-order approximation to the volumetric Green's function and is given as a solution of the equation

$$\nabla^2_{\hat{\boldsymbol{\rho}}} G^{(1)} = \delta\left(\hat{\boldsymbol{\rho}} - \hat{\boldsymbol{\rho}}_o\right) + R^2\left(e^{G^{(0)}} - e^{-G^{(0)}}\right). \tag{12}$$

Based on Eqs. (11) and (12), $G^{(1)}\left(\hat{\boldsymbol{\rho}}|\hat{\boldsymbol{\rho}}_o\right)$ is given by the expression

$$G^{(1)}\left(\hat{\boldsymbol{\rho}}|\hat{\boldsymbol{\rho}}_o\right) = G^{(0)}\left(\hat{\boldsymbol{\rho}}|\hat{\boldsymbol{\rho}}_o\right) + R^2 \int_0^1 \int_0^\pi \int_0^{2\pi} \Big[d\hat{\rho}' d\hat{\theta}' d\hat{\varphi}' \left(\hat{\rho}'\right)^2 \sin\hat{\theta}' \times G^{(0)}\left(\hat{\boldsymbol{\rho}}|\hat{\boldsymbol{\rho}}'\right) f\left\{G^{(0)}\left(\hat{\boldsymbol{\rho}}'|\hat{\boldsymbol{\rho}}_o\right)\right\}\Big]; \quad f\{y\} = e^y - e^{-y}. \tag{13}$$

Based on this approximate expression for the volumetric Green's function and Eq. (3), an expression for normalized potential at a point $\hat{\boldsymbol{\rho}}_o$ is given by a line integral over the circumference of the unit circle and is expressed as

$$\hat{\phi}(\hat{\boldsymbol{\rho}}_o) = \int_0^\pi \int_0^{2\pi} d\hat{\theta} d\hat{\varphi} \sin\hat{\theta} \left[\frac{dG}{d\hat{\rho}}\right]_{\hat{\rho}=1} \times \hat{\phi}\left(1, \hat{\theta}, \hat{\varphi}\right) \tag{14}$$

For the development of the floating random-walk algorithm, we need to estimate $\left[\frac{dG^{(1)}}{d\hat{\rho}}\right]_{\hat{\rho}=1}$ in Eq. (14). Such an estimate is obtained by differentiating Eq. (13), and in the zero-centered notation (i.e., $\hat{\rho}_o = 0$) is given by

$$\left[\frac{dG}{d\hat{\rho}}\right]_{\hat{\rho}=1} = \frac{1}{4\pi} + \frac{R^2}{4\pi} \int_0^1 \int_0^\pi \int_0^{2\pi} \Big[d\hat{\rho}' d\hat{\theta}' d\hat{\varphi}' (\hat{\rho}')^2 \sin\hat{\theta}' \times D \times E\Big] \tag{15}$$

where D and E are given by

$$\begin{aligned} D &= e^H - e^{-H}, \quad H = \frac{1}{4\pi}\left[1 - \frac{1}{\hat{\rho}'}\right], \\ E &= \frac{1}{4\pi}\left[\frac{1 - (\hat{\rho}')^2}{\{1 + (\hat{\rho}')^2 - 2\hat{\rho}' C\}^{3/2}}\right]. \end{aligned} \tag{16}$$

Eqs. (15) and (16), in conjunction with Eq. (14), are used to develop the FRW algorithm for the problem under consideration. In order to calculate the normalized potential at a point of interest, we start our random walks at that point and hop to the surface of a sphere of radius R. The random walks have to be restricted to a small fraction of λ to maintain the validity of the first-order approximation in the perturbation expression for the volumetric Green's function. For every hop there is a weight factor obtained by sampling the multi-dimensional integrand of Eq. (14) (with the help of a random-number generator) according to any pre-determined probability distribution for each of the variables. As explained in the previous section, a particular random

walk, consisting of several such random hops, is terminated on the boundary of the problem domain, where the value of the potential is known. The contribution from a particular random-walk is obtained by multiplying the overall weight factor (which is obtained by multiplying the weight factors of individual hops) with the boundary value, and an estimate $\overline{\hat{\phi}}$ of the potential, at the point of origination of the hops is obtained by averaging over a statistically large number of random walks and given by

$$\overline{\hat{\phi}} = \frac{1}{N} \sum_{n=1}^{N} \hat{\phi}_n. \tag{17}$$

In order to achieve convergence of this algorithm, for $\rho' \leq 0.01$, the term D in Eq. (15) is expanded as a polynomial in H and terms beyond the fourth power in H are dropped.

The error in the result has two components:

1) A deterministic error arising from the truncation of the iterative perturbation based Green's function in Eq. (13), which can be controlled by controlling the radius of the hop.
2) A statistical $1 - \sigma$ error σ_T given by [17]

$$\sigma_T = \frac{\sigma_E}{\sqrt{N}}, \tag{18}$$

where σ_E is the standard deviation of the estimates from different random-walks and N is the number of random-walks. As a result, the statistical error can be controlled by controlling the number of random-walks.

We have chosen two benchmark problems. The first problem (Fig. 2) is characterized by angular symmetry, where a spherical electrode 0.5λ in radius is surrounded by another spherical electrode of radius 1.5λ. In the second problem (Fig. 3), no such angular symmetry exists and a spherical electrode λ in diameter is surrounded by a box of dimensions $3\lambda \times 2\lambda \times 2\lambda$. The boundary conditions imposed in both benchmark problems are such that the normalized potential is unity on the inner electrode and zero on the outer electrode.

The random-walk algorithms were coded in C and run on a Silicon Graphics, Inc. workstation. The finite-difference solver used for validation was written in Fortran and run on the same computing platform. In this work, 20000 random walks were performed per solution point, and the radii of the hops were restricted to two percent of λ to maintain the validity of the first-order approximation to the volumetric Green's function. For the finite-difference solution of the

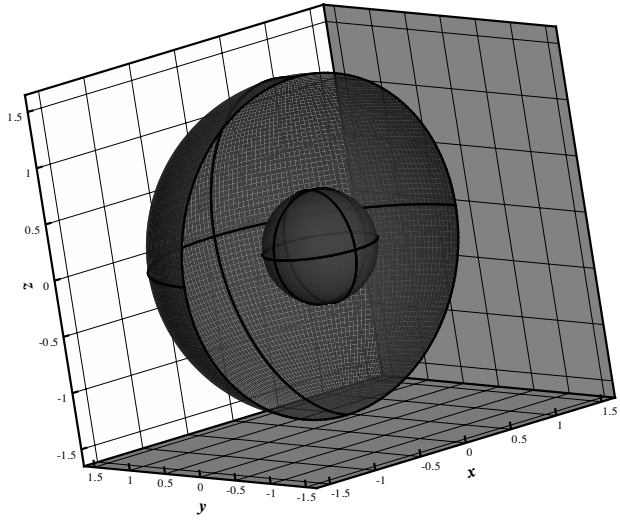

Figure 2. Solution domain enclosed between two concentric spheres maintained at fixed potentials. Half of outer sphere cut away to reveal inner sphere.

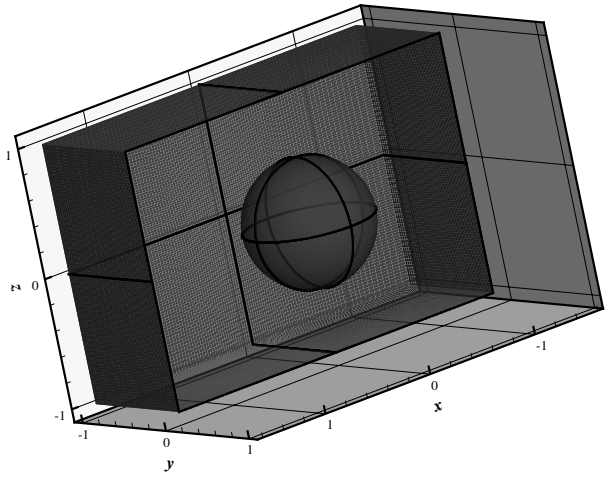

Figure 3. Solution domain enclosed in the space between a sphere and a rectangular parallelepiped maintained at fixed potentials. Half of rectangular outer domain cut away to reveal inner sphere.

first problem, the angular symmetry of the problem was exploited and the finite-difference solution was performed only in the radial dimension, which was discretized into 101 points. For the second problem, the finite-difference calculations were carried out over a $51 \times 51 \times 51$ grid distributed over the positive octant of the problem geometry (Fig. 4).

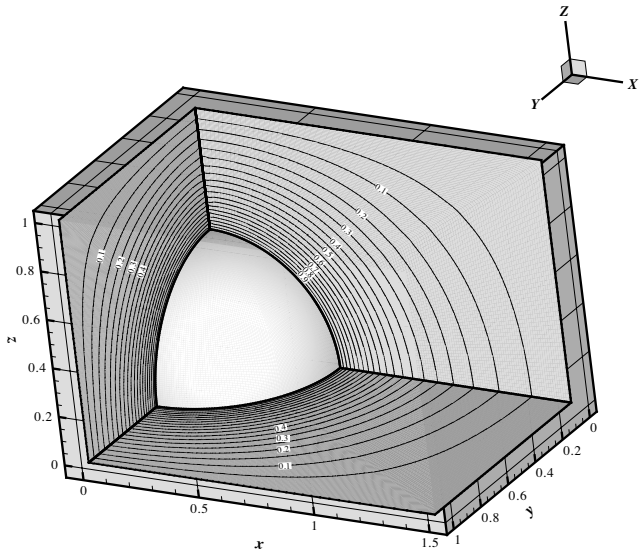

Figure 4. Finite-difference solution for Problem 2. First octant shown.

Table 1 tabulates the statistical error and the mean absolute discrepancy between the random-walk and finite-difference based results for each of the benchmark problems. Solution profiles for the two benchmark problems are plotted in Fig. 5 and Fig. 6, respectively. There is excellent agreement between the random-walk solutions and finite-difference based results. It can also be observed that the absolute discrepancies are around three times larger than the statistical errors. This can be attributed to the truncation of the perturbation-based Green's function in Eq. (13), and also to the truncation errors in the finite-difference based approach.

For a time comparison, both the FRW and finite-difference algorithm was run for Problem 2 on a 1-processor SGI workstation (Specifications: 400 MHz IP30 MIPS R12000 processor, 4 GB RAM). The time required for the finite-difference algorithm was 52 minutes, 15 seconds, whereas the time required for the FRW algorithm was 14 minutes, 10 seconds. Of course, the finite-difference algorithm

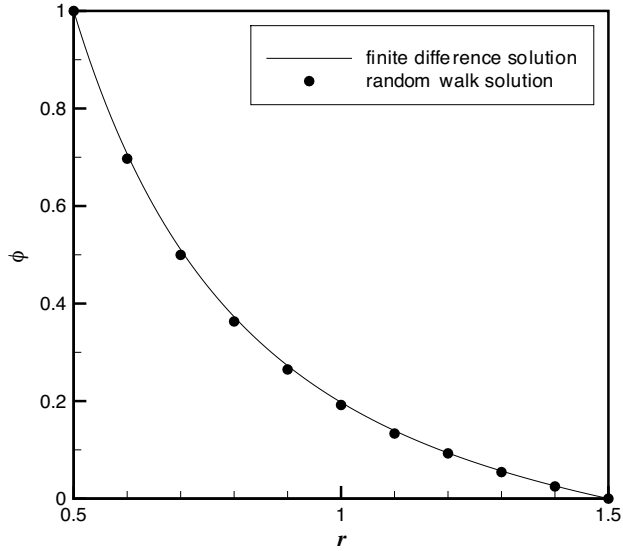

Figure 5. Potential plotted against position in normalized coordinates for Problem 1.

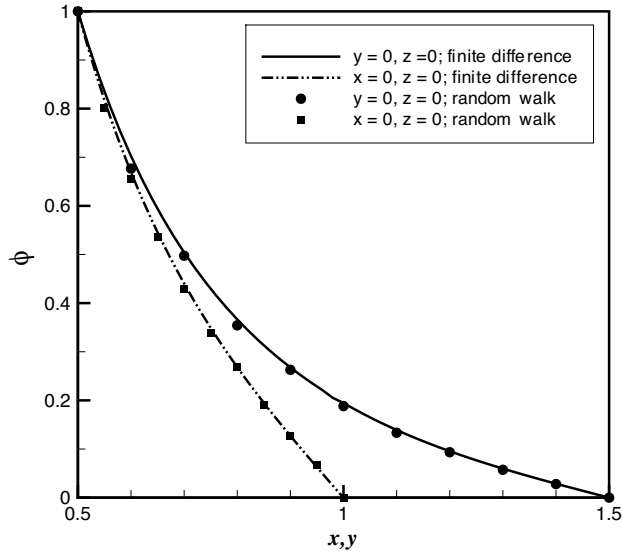

Figure 6. Potential plotted against position in normalized coordinates for Problem 2.

Table 1. Statistical error and mean absolute discrepancy between random-walk and finite-difference based results.

Benchmark Problems	Mean Absolute Discrepancy	Statistical Error
Problem 1	6.0e-3	1.9e-3
Problem 2 (along the centerline positive x-axis)	7.4e-3	2.8e-3
Problem 2 (along the centerline positive y-axis)	7.9e-3	2.5e-3

produced the solution at 132651 points, while the FRW algorithm produced the solution at 22 points. This confirms the well-known fact that stochastic solution methods are superior when the requirement is to know the solution at a relatively few points on the problem domain.

Further, FRW algorithms have a strong advantage on geometrically complicated domains in which a dense finite difference mesh must be generated in order to produce an accurate solution. For example, in Problem 2, several techniques were used to reduce the size of the finite difference mesh: symmetry was exploited and grid clustering was employed near the surface of the sphere and near the corners in order to adequately resolve the large potential gradients in those locations. For extremely complicated problems, as in IC interconnects, finite difference or finite element methods on fully-resolved meshes become intractable, and random walk methods have significant advantages.

The FRW algorithm for Problem 1 was parallelized. Two levels of parallelism are inherent in an FRW algorithm. First, the solutions for different points in the domain (different origins for the random walks) are independent of each other. Second, for a given point of origin, each random walk is independent, and inter-processor communication is required only to sum up the contributions of the walks. For this initial parallel implementation, the test points in the domain were handled serially. The walks were distributed in groups across computer processors, with communication and a reduction operation at the completion of the walks. As mentioned previously, the code was written in C, and the serial version of the code was converted to parallel using the Message Passing Interface (MPI) library. The elegance and inherent parallelism of the FRW algorithm is displayed in the fact that the serial and parallel versions of the code differ by only four function calls, three of which are merely initialization routines. The inter-processor communication is handled by one call

to "MPI_Reduce," which sums the contributions from each walk. The parallel computations were run on a COMPAQ SC-40 machine, based on an 833 MHz EV 6.8 chip. This computer has a shared/distributed memory system with 4 processors per shared-memory node. The results of this parallelization are tabulated in Table 2 and Fig. 7. We have used as many as 32 processors and an almost linear increase in computational speed is observed with the increase in the number of processors.

Table 2. Parallelization results of the FRW algorithm for NPB equation for Problem 1.

Number of processors	*Random-walks/processor/point*	*Total Time required to calculate solution at 11 points (seconds)*
1	20000	334.60
2	10000	168.58
4	5000	85.86
8	2500	44.71
16	1250	23.20
32	625	13.01

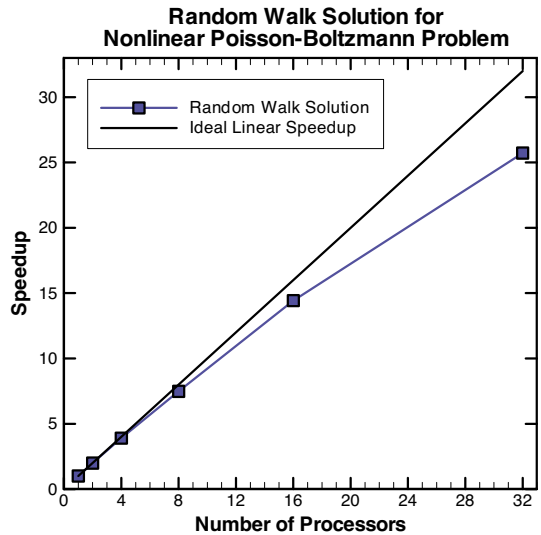

Figure 7. Efficiency of parallelization of the FRW algorithm for Problem 1.

4. CONCLUSION

A new FRW algorithm has been developed for the solution of the NPB equation in three dimensions. Due to the absence of analytical expressions for the volumetric Green's function subject to homogeneous boundary conditions, an approximate expression has been calculated with the help of a novel use of perturbation theory. Excellent agreement was found between the results of random-walk calculations and finite-difference based results. This method has the advantages of being highly parallelizable and requiring no discretization of the problem domain. The approach is general, and can be applied to the numerical solution of other important nonlinear equations.

Our work in the immediate future will investigate the benefits of retaining higher-order perturbation terms in the volumetric Greens function expression and the extension of this algorithm to Neumann and mixed boundary condition problems. The ultimate objective of this research is to develop FRW algorithms for the solution of plasma flow equations and address the efficient implementation of the algorithms on parallel processor computer platforms.

ACKNOWLEDGMENT

Support for Dr. K. Chatterjee was provided under a National Research Council Summer Faculty Fellowship at the Air Force Research Laboratory, Wright-Patterson Air Force Base. Additional support was provided by the Air Force Office of Scientific Research, under grants monitored by Dr. J. Schmisseur and Dr. F. Fahroo. We would like to acknowledge valuable discussions with Dr. M. D. White and Dr. D. V. Gaitonde.

REFERENCES

1. Mitchner, M. and C. H. Kruger, *Partially Ionized Gases,* John Wiley & Sons, New York, 1973.
2. Sze, S. M., *Physics of Semiconductor Devices,* 2nd ed., 366–368, John Wiley & Sons, New York, 1999.
3. Davis, M. E. and J. A. McCammon, "Electrostatics in biomolecular structure and dynamics," *Chemical Reviews,* Vol. 90, 509–521, 1990.
4. Brown, G. M., *Modern Mathematics for Engineers,* E. F. Beckenbach (ed.), McGraw-Hill, New York, 1956.

5. Le Coz, Y. L., H. J. Greub, and R. B. Iverson, "Performance of random walk capacitance extractors for IC interconnects: a numerical study," *Solid-State Electronics,* Vol. 42, 581–588, 1998.
6. Le Coz, Y. L., R. B. Iverson, T. L. Sham, H. F. Tiersten, and M. S. Shepard, "Theory of a floating random walk algorithm for solving the steady-state heat equation in complex materially inhomogeneous rectilinear domains," *Numerical Heat Transfer, Part B: Fundamentals,* Vol. 26, 353–366, 1994.
7. Haberman, R., *Elementary Applied Partial Differential Equations,* 3rd ed., Prentice-Hall, Englewood Cliffs, NJ, 1998.
8. Hwang, C.-O. and M. Mascagni, "Efficient modified 'walk on spheres' algorithm for the linearized poisson-boltzmann equation," *App. Phys. Lett.,* Vol. 78, No. 6, 787–789, 2001.
9. Mascagni, M. and N. A. Simonov, "Monte carlo methods for calculating the electrostatic energy of a molecule," *Proceedings of the 2003 International Conference on Computational Science (ICCS 2003).*
10. Chatterjee, K. and J. Poggie, "A meshless stochastic algorithm for the solution of the nonlinear poisson-boltzmann equation in the context of plasma discharge modeling: 1D analytical benchmark," *Proceedings of the 17th AIAA Computational Fluid Dynamics Conference,* 2005–5339, Toronto, Ontario, Canada, June 6–9, 2005 (American Institute of Aeronautics and Astronautics, Reston, VA, 2005).
11. Chatterjee, K. and J. Poggie, "A two-dimensional floating random-walk algorithm for the solution of the nonlinear Poisson-Boltzmann equation: application to the modeling of plasma sheaths," *Proceedings of the 3rd MIT Conference on Computational Fluid and Solid Mechanics,* 1058–1061, Cambridge, MA, June 14–17, 2005 (Elsevier, Oxford UK, 2005).
12. Sobol, I. M., *A Primer for the Monte Carlo Method,* CRC Press, Boca Raton, 1994.
13. Curtiss, J. H., "Monte carlo methods for the iteration of linear operators," *J. Math. and Phys.,* Vol. 32, 209–232, 1954.
14. Sabelfeld, K. K. and N. A. Simonov, *Random Walks on Boundary for Solving PDEs,* 119–120, VSP, Utrecht, The Netherlands, 1994.
15. Chatterjee, K., "Development of a floating random walk algorithm for solving Maxwell's equations in complex IC-interconnect structures," Rensselaer Polytechnic Institute, May, 2002, UMI Dissertation Services, 300 North Zeeb Road, P.O. Box 1346, Ann Arbor, Michigan 48106-1346, USA, UMI Number: 3045374, Web Address: www.il.proquest.com.

16. Chatterjee, K. and Y. L. Le Coz, "A floating random-walk algorithm based on iterative perturbation theory: solution of the 2D vector-potential maxwell-helmholtz equation," *Applied Computational Electromagnetics Society Journal,* Vol. 18, No. 1, 48–57, March 2003.
17. Hammersley, J. M. and D. C. Handscomb, *Monte Carlo Methods,* 50–54, John Wiley & Sons, New York, NY, 1964.