INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN ENGINEERING
Int. J. Numer. Meth. Engng 2001; **51**:1133–1156 (DOI: 10.1002/nme.178)

Random walk method for the two- and three-dimensional Laplace, Poisson and Helmholtz's equations

Mandar K. Chati[1,§,‡], Mircea D. Grigoriu[2,¶],
Salil S. Kulkarni[1,§] and Subrata Mukherjee[1,*,†,¶]

[1] *Department of Theoretical and Applied Mechanics, Cornell University, Ithaca, NY 14853, U.S.A.*
[2] *Department of Environmental and Civil Engineering, Cornell University, Ithaca, NY 14853, U.S.A.*

SUMMARY

The random walk method (RWM) is developed here for solving the Laplace, Poisson, and Helmholtz equations in two and three dimensions. The RWM is a *local* method, i.e. the solution at an arbitrary point can be determined without having to obtain the complete field solution. The method is based on the properties of diffusion processes, the Itô formula, the Dynkin formula, the Feynman–Kac functional, and Monte Carlo simulation. Simplicity, stability, accuracy, and generality are the main features of the proposed method. The RWK is inherently parallel and this fact has been fully exploited in this paper. Extensive numerical results have been presented in order to understand the various parameters involved in the method. Copyright © 2001 John Wiley & Sons, Ltd.

KEY WORDS: random walk method; Brownian motion; Laplace; Poisson; Helmholtz; parallel computing

1. INTRODUCTION

1.1. Global and local methods

Global methods providing values of the stress, displacement, and other response functions at all or a finite number of points are generally used to solve mechanics, elasticity, physics, and other engineering problems. These methods can be based on analytical or numerical algorithms. Analytical methods have limited value because few practical problems admit closed-form solutions. The finite element, boundary element and finite difference methods are generally applied to solve practical problems. Some of the possible limitations of these numerical methods are: (1) the computer codes used for solution of boundary value problems are relatively complex

*Correspondence to: Subrata Mukherjee, Theoretical and Applied Mechanics, 212 Kimball Hall, Cornell University, Ithaca, NY 14853, U.S.A.
† E-mail: sm85@cornell.edu
‡ Presently at: General Electric, Corporate Research and Development, Schenectady, NY 12301, U.S.A.
§ Graduate student
¶ Professor

Contract/grant sponsor: NSF; contract/grant numbers: CMS-9610022 and CMS-9912524

Received 21 June 1999
Revised 31 August 2000

and can involve extensive preprocessing to formulate a particular problem in the required format, (2) the numerical algorithms may become unstable in some cases, (3) the order of the errors caused by the discretization of the continuum and the numerical integration methods used in analysis cannot always be bounded and (4) the field solution must be calculated even if the solution is needed at a single point.

A major objective of the present work is to demonstrate the easy implementation of alternative techniques to the traditional numerical methods, referred to collectively as *local methods*, that provide efficient solutions for relevant problems in mechanics. The local methods give the solution of a partial differential equation at an arbitrary point in the domain directly, rather than extracting the response value at this point from the field solution. These methods are based on probabilistic interpretations of certain partial differential equations. The development of the local method presented in the paper is based on the mean value theorem for differential equations, the Itô calculus, properties of Itô diffusion processes, and Monte Carlo simulation. The theoretical considerations supporting the proposed local method are relatively complex and involve concepts of elasticity, applied mathematics, random processes, and stochastic integrals. Some of the details are presented in this paper, but for an elaborate explanation see References [1–4]. The local method discussed in this paper has been used to solve the 1-D Schrödinger equation in Reference [5].

In spite of the mathematical complexity associated with the proposed local method, however, the numerical algorithms have attractive features, such as

- *Simple to program*: The computer codes are extremely simple and are presented in this paper for some of the numerical examples considered.
- *Always stable*: The method is stable irrespective of the input parameters chosen to obtain the numerical results.
- *Accurate*: By appropriate choice of parameters the desired accuracy can be obtained.
- *Local*: The field solution can be calculated only at a desired point, for example, at a point of stress concentration, without having to obtain the global solution by solving a system of linear equations.
- *No meshing*: There is absolutely NO discretization of the domain or the bounding surface required in this method.
- *Ideal for parallel computation*: The method is inherently parallel. This fact has been fully exploited in this paper and extensive parallel computations have been carried out.

The salient features of the local methods are briefly compared below with two popular methods in computational mechanics, namely, the finite element method (FEM) (see Reference [6]) and the boundary element method (BEM) (see Reference [7]).

As is well known, the starting point of the FEM is a weak form (usually obtained from a variational formulation or from weighted residuals) of the governing differential equations of a problem. The weak form is typically an integral defined over the domain of the body. The entire domain is discretized into finite elements, and a piecewise interpolation of the unknown function is carried out over the elements. Upon inserting the interpolants into the weak form, and carrying out the necessary local integrations of the shape functions, one obtains a system of linear algebraic equations of the form

$$\mathbf{Ax} = \mathbf{b} \tag{1}$$

Table I. Comparison of FEM, BEM and LM.

	FEM	BEM	LM
Locality	None (Domain method)	Better (surface method)	Strictly local
Mesh	Complicated (Needs domain mesh)	Simpler (Needs surface mesh)	No mesh at all
$\mathbf{Ax} = \mathbf{b}$	**A** is large, sparse and symmetric	**A** is smaller, dense and non-symmetric	No linear system
Versatility	Very general	Less general	Less general
Parallelization	Possible but needs effort	Easier but still needs effort	Already parallel
Programming effort	Reasonable	Reasonable	Extremely trivial
Exterior problems	Needs work	Works well	Very easy

The vector **x** contains the nodal values of the unknown function throughout the body and the vector **b** is usually obtained from the boundary conditions and non-homogeneous terms, if any. Equation (1) provides the solution at the nodal values and then the solution at any other point can be obtained from the interpolating functions.

The starting point of the BEM is an integral equation defined on the boundary on the body, typically obtained from a mathematical identity such as one due to Green (for potential problems) or Somigliana (for elasticity problems). For linear problems, the BEM requires discretization of the bounding surface of the body into boundary elements, and piecewise interpolation of the unknown function is carried out over these elements. Upon inserting the interpolates into the integral equation, one obtains a system of linear equations of the same generic form as Equation (1).

A comparison between the conventional FEM and BEM with the local methods is presented in Table I.

1.2. Brief literature survey

There is a rich literature on the use of local methods to solve deterministic parabolic and elliptic partial differential equations. A brief (and by no means comprehensive) survey of this work is presented below.

The relationship between stochastic processes and parabolic and elliptic differential equations was demonstrated a long time ago by Lord Rayleigh [8] and Courant [9], respectively. Haji-Sheikh and Sparrow [10] developed a floating random walk method and applied it to obtain solutions to steady-state and transient heat equations. This method, also known as the *walk on sphere* method, is the most important of the more general *walk inside the domain methods*. Other walks, namely walk on ellipsoids, squares, cones and half-spaces also have been developed (see Reference [11]). Another probabilistic method known as the *walk on boundary* [11] or *the surface density technique* [12] also exits. Chorin [13] has also used a probabilistic method to solve the diffusion equation as a part of the solution to the vorticity transport equation. Another local method, called the *random walk method*, is the method of choice in the present paper. This method is briefly described in Section 1.3.3.

1.3. Walk on sphere, boundary walk and random walk methods

Table II summarizes the salient features of the local methods.

1.3.1. Walk on sphere method (WSM). The WSM has been applied to solve the Laplace and Poisson equations in $\Re^d$ with Dirichlet and/or Neumann boundary conditions. The solution is based on the integral representation

$$u(\mathbf{x}) = \frac{1}{a_d} \int_{S(\mathbf{x},r)} k(\mathbf{x},\mathbf{y})\, u(\mathbf{y})\, \mathrm{d}\sigma(\mathbf{y}) \tag{2}$$

of the unknown function u, where $a_d = 2\pi^{d/2} r^{d-1}/\Gamma(d/2)$ denotes the surface area of the sphere $S(\mathbf{x},\mathbf{r}) = \{\mathbf{y} \in \Re^{\mathrm{d}} : \|\mathbf{y} - \mathbf{x}\| \leqslant r\} \subseteq D$ of radius $r > 0$ centred at $\mathbf{x} \in D$, D is an open bounded domain in $\Re^d$, and the kernel k depends on the particular form of the differential operator. An equivalent form of Equation (2) is

$$u(\mathbf{x}) = E[k(\mathbf{x},\mathbf{Y})\, u(\mathbf{Y})], \quad \mathbf{x} \in D \tag{3}$$

where $\mathbf{Y}$ is uniformly distributed on $S(\mathbf{x},r)$ and the expectation is performed relative to this random variable. This observation is used to develop an algorithm for estimating $u(\mathbf{x})$. Suppose, for example, that u is the solution of a Dirichlet boundary value problem defined in an open bounded set $D \subset \Re^d$ satisfying the boundary condition $u(\mathbf{x}) = \xi(\mathbf{x})$, $\mathbf{x} \in \partial D$. The objective is to find an estimate of the value of u at an arbitrary point $\mathbf{x} \in D$. Let $\mathbf{x}_0 = \mathbf{x}$ and $S(\mathbf{x}_0, r(\mathbf{x}_0))$ be the largest sphere centred at $\mathbf{x}_0$ that can be inscribed in D. Let $\mathbf{x}_1(\omega)$ be a randomly selected point on $S(\mathbf{x}_0, r(\mathbf{x}_0))$ from the uniform distribution on this sphere (Equation (3)) and $S(\mathbf{x}_1(\omega), r(\mathbf{x}_1(\omega)))$ the largest sphere centred at $\mathbf{x}_1(\omega)$ that can be inscribed in D. Select a random uniformly distributed point $\mathbf{x}_2(\omega)$ on $S(\mathbf{x}_1(\omega), r(\mathbf{x}_1(\omega)))$ and continue as in the previous step. This algorithm generates a sequence of points $\{\mathbf{x}_0, \mathbf{x}_1(\omega), \mathbf{x}_2(\omega), \ldots\}$, referred to as a *spherical process*, that converges to a point $\mathbf{y}(\omega)$ of ∂D. The value of u at this limit point is known and equal to $\xi(\mathbf{y}(\omega))$ from the boundary conditions, so that

$$u(\mathbf{x}_0, \omega) = \left[\prod_{i=1}^{N(\omega)} k(\mathbf{x}_{i-1}(\omega), \mathbf{x}_i(\omega)) \right] \xi(\mathbf{y}(\omega)) \tag{4}$$

where $N(\omega) + 1$ denotes the number of steps of the spherical process till it reaches ∂D along sample path ω. Then $u(\mathbf{x}_0)$ can be estimated by the average value

$$\hat{u}(\mathbf{x}_0) = \frac{1}{n} \sum_{\omega=1}^{n} u(\mathbf{x}_0, \omega) \tag{5}$$

over n independent sample paths of the spherical process.

1.3.2. Boundary walk method (BWM). The walk on sphere and walk on boundary methods are similar in the sense that both are based on integral representations of the solution of a boundary value problem. However, the representations corresponding to these methods differ significantly. The integral representation for the WSM is an equivalent statement of the mean value theorem and is defined on subsets of the domain of definition D, while the integral representation for the BWM constitutes the Fredholm integral associated with the boundary value problem under consideration, and is defined on the entire domain of definition D.

Table II. Comparison of RWM WSM and BWM.

Methods	Features	Advantages	Limitations
RWM	Based on random walks	Very easy to program	Currently for operators given by Equation (6)
WSM	Based on local integral equation representation and spherical processes	Easy to program	Currently for operators given by Equation (6)
BWM	Based on global integral equations representation on physical boundaries and Neumann series	Versatile, simple probabilistic interpretation of kernels for Laplace Poisson equations Can be applied to solve linear elasticity problems	Currently Neumann series diverges but it can be regularized

1.3.3. Random walk method (RWM). The RWM can be applied to find the local solution of *second-order* partial differential equations of the form

$$\frac{\partial u(\mathbf{x},t)}{\partial t} = \sum_{i=1}^{d} \alpha_i(\mathbf{x}) \frac{\partial u(\mathbf{x},t)}{\partial x_i} + \frac{1}{2} \sum_{i,j=1}^{d} \beta_{ij}(\mathbf{x}) \frac{\partial^2 u(\mathbf{x},t)}{\partial x_i \partial x_j} + q(\mathbf{x},t)\, u(\mathbf{x},t) + p(\mathbf{x},t) \tag{6}$$

where α_i, β_{ij} are real-valued functions defined on $\Re^d$, $d \geqslant 1$ is an integer, and q, p denote real–valued functions defined on $\Re^d \times [0,\infty)$. The domain of definition of Equation (6) is $D \times (0,\infty)$, where $D \subset \Re^d$ is an open bounded set. The solution $u : D \times (0,\infty) \to \Re$ depends on the initial and boundary conditions that need to be specified. The operator of Equation (6) includes a large number of interesting special cases; for example, parabolic, hyperbolic, elliptic partial differential equations in $\Re^2$ correspond to the steady–state version of Equation (6) with $d = 2$ and $\beta_{12}(\mathbf{x})\,\beta_{12}(\mathbf{x}) - \beta_{11}(\mathbf{x})\,\beta_{22}(\mathbf{x}) = 0; > 0; < 0$, respectively. Therefore, for example, the Laplace, Poisson and Helmholtz equations are special cases of Equation (6).

The RWM method can be applied to find the local solution of Equation (6) with Dirichlet and/or Neumann boundary conditions. The solution by this method involves three steps. *First*, a diffusion process $\mathbf{X}$ with generator coinciding with the differential operator of Equation (6) has to be constructed. *Second*, a relationship needs to be established between the value of the unknown function u at $(\mathbf{x},t) \in D \times (0,\infty)$, the boundary conditions, and an expectation depending on the sample paths of $\mathbf{X}$. Properties of diffusion processes, features of stochastic integrals, and Itô's formula can be used to obtain this relationship. *Third*, a Monte Carlo algorithm needs to be developed to estimate the expectation giving $u(\mathbf{x},t)$.

1.4. Outline of this paper

The main contribution in this work is the efficient application of the RWM to solve problems in Laplace, Poisson and Helmholtz's equations in two or three dimensions, and elliptic/parabolic differential equations which are special cases of Equation (6). The emphasis here is to demonstrate the ease of using the RWM for solving this class of problems and to carry out parallel

implementations of this method for several examples. The RWM is already transparently parallel and it is shown that parallel codes based on this method are extremely efficient.

Another important class of problems is linear elasticity. The RWM is not well suited for this class of problems since it has not been yet possible to find a diffusion process whose generator matches the Navier–Stokes operator for linear elasticity. The stress function approach also does not lend itself to the RWM since it involves a fourth-order elliptic partial differential equation. The BWM appears to be best suited for elasticity problems as demonstrated by Shia and Hui [14]. Further work on elasticity problems, using the BWM, is currently in progress.

This paper is organized as follows. Section 2 presents some technical details of the RWM. Section 3 describes the procedure for obtaining local solutions for the Laplace, Poisson and Helmholtz equations using the RWM. Section 4 briefly describes the procedure for obtaining local solutions of the more general Equation (6). Section 5 presents numerical results for selected example problems governed by the Laplace, Poisson, Helmholtz equations respectively; and for an elliptic PDE with varying coefficients. Finally, some conclusions are drawn in Section 6.

2. TECHNICAL DETAILS OF THE RWM

The Itô formula gives a method for calculating the increment of functions of diffusion processes during a given time interval. For our discussion, it is sufficient to consider a special case of diffusion processes called the Brownian motion process. Let $\{\mathbf{B}(t), t \geqslant 0\}$ be a Brownian motion process taking values in $\Re^n$. The process starts at an arbitrary point $\mathbf{x} \in \Re^n$ and is characterized by stationary independent increments, that is, the increments of $\mathbf{B}$ over non-overlapping time intervals are independent and their statistics depend only on the duration of the time increment. Specifically, (1) the increment $\mathbf{B}(t) - \mathbf{B}(s)$ of $\mathbf{B}$ during the time interval (s,t), $s<t$, is an $\Re^n$-valued Gaussian vector with mean zero and covariance matrix $\mathbf{I}(t-s)$, where $\mathbf{I}$ denotes the identity matrix and (2) the increments $\mathbf{B}(t) - \mathbf{B}(v)$ and $\mathbf{B}(u) - \mathbf{B}(s)$, $s<u \leqslant v<t$, are independent.

Consider a function $g : \Re^d \to \Re$ with continuous partial second derivatives. It can be shown that the average rate of change of $g(\mathbf{B}(t))$ at $t=0$ given that $\mathbf{B}(0)=\mathbf{x}$ is

$$\mathscr{A}g(\mathbf{x}) = \lim_{t \downarrow 0} \frac{E^x[g(\mathbf{B}(t))] - g(\mathbf{x})}{t} = \frac{1}{2}\Delta g(\mathbf{x}) \tag{7}$$

where $\Delta = \sum_{i=1}^{n} \partial^2/\partial x_i^2$ is the Laplace operator. The operator $\mathscr{A}$ defined by Equation (7) is called the generator of $\mathbf{B}$. Here, $E^x[\,] = E[|\mathbf{B}(0)=\mathbf{x}]$.

Taking the expectation of the Itô formula applied to the function $g(\mathbf{B}(s))$, conditional on $\mathbf{B}(0)=\mathbf{x}$, gives

$$E^x[g(\mathbf{B}(t))] - g(\mathbf{x}) = \frac{1}{2} E^x\left[\int_0^t \Delta g(\mathbf{B}(s))\,\mathrm{d}s\right] \tag{8}$$

This formula can be generalized by replacing t with a random time. For example, let D be a bounded open subset of $\Re^n$ and suppose that the Brownian motion $\mathbf{B}$ starts at $\mathbf{x} \in D$. Define

$$T = \inf\{t \geqslant 0:\ \mathbf{B}(t) \notin D, \mathbf{B}(0) = \mathbf{x} \in D\} \tag{9}$$

to be the first time when **B**, starting at $\mathbf{x} \in D$, leaves this set. The time T is a random variable because its value depends on the particular sample of the Brownian motion. The averaged Itô formula with t replaced by T becomes

$$E^x[g(\mathbf{B}(T))] - g(\mathbf{x}) = \frac{1}{2} E^x \left[\int_0^T \Delta g(\mathbf{B}(s)) \, \mathrm{d}s \right] \tag{10}$$

This equation is essential for obtaining local solutions of the Laplace and Poisson's equation.

The generalized version of Equation (10) is given by

$$E^x[g(\mathbf{X}(T))] - g(\mathbf{x}) = E^x \left[\int_0^T \mathscr{A} g(\mathbf{X}(s)) \, \mathrm{d}s \right] \tag{11}$$

Here $\{\mathbf{X}(t) \in \Re^n, t \geqslant 0\}$ is a Itô diffusion given by

$$\mathrm{d}\mathbf{X}(t) = \mathbf{b}(\mathbf{X}(t)) \, \mathrm{d}t + \boldsymbol{\sigma}(\mathbf{X}(t)) \, \mathrm{d}\mathbf{B}(t) \tag{12}$$

where the drift **b** and the diffusion $\boldsymbol{\sigma}$ are $(n, 1)$ and (n, m) matrices and **B** is an m-dimensional standard Brownian motion. In Equation (11) $\mathscr{A} g$ is the generator of the diffusion process (12) and is given by

$$\mathscr{A} g = \sum_i b_i \frac{\partial g}{\partial x_i} + \frac{1}{2} \sum_{i,j} (\sigma \, \sigma^{\mathrm{T}})_{i,j} \frac{\partial^2 g}{\partial x_i \partial x_j} \tag{13}$$

Also, T in Equation (11) is defined as

$$T = \inf\{t \geqslant 0 \colon \mathbf{X}(t) \notin D, \mathbf{X}(0) = \mathbf{x} \in D\} \tag{14}$$

and $E^x[\,] = E[\,|\mathbf{X}(0) = \mathbf{x}]$. Equation (11) is referred to as Dynkin's formula. It can be observed that Equation (11) reduces to Equation (10) for $\mathbf{b} = \mathbf{0}$ and $\boldsymbol{\sigma} = \mathbf{I}_n$ where $\mathbf{I}_n$ is an n-dimensional identity matrix.

3. LOCAL SOLUTIONS OF LAPLACE, POISSON, AND HELMHOLTZ EQUATIONS BY THE RWM

Let $u : \Re^n \to \Re$ be the solution of the Poisson equation

$$\Delta u(\mathbf{x}) + p(\mathbf{x}) = 0, \quad \mathbf{x} \in D \tag{15}$$

satisfying the Dirichlet boundary conditions

$$u(\mathbf{x}) = \xi(\mathbf{x}), \quad \mathbf{x} \in \partial D \tag{16}$$

where p and ξ are specified functions.

Suppose that the objective is to find the local solution of Equations (15) and (16) at an arbitrary point **x** of the domain of definition D. Because u has continuous second-order partial derivatives, Equation (10) applies and gives

$$E^x[\xi(\mathbf{B}(T))] - u(\mathbf{x}) = -\frac{1}{2} E^x \left[\int_0^T p(\mathbf{B}(s)) \right] \tag{17}$$

because $u(\mathbf{B}(T)) \in \partial D$ so that it is equal to the boundary value $\xi(\mathbf{B}(T))$ and $\mathbf{B}(s) \in D$ for $s \in (0, T)$ so that $\Delta u(\mathbf{B}(s)) = -p(\mathbf{B}(s))$ by Equation (15). Hence, the value of the unknown function u at $\mathbf{x} \in D$ is

$$u(\mathbf{x}) = E^x[\xi(\mathbf{B}(T))] + \frac{1}{2}E^x\left[\int_0^T p(\mathbf{B}(s))\right] \tag{18}$$

The right-hand side of Equation (18) depends on expectations of known functions, the functions ξ and p, and samples of the Brownian motion $\mathbf{B}$ in the time interval $(0, T)$. Generally, these expectations cannot be obtained analytically. However, they can be estimated by Monte Carlo simulation as demonstrated in the following sections.

If $p = 0$ in Equation (15), the equation considered for solution is a Laplace equation and the local solution of this equation is given by

$$u(\mathbf{x}) = E^x[\xi(\mathbf{B}(T))] \tag{19}$$

that is, Equation (18) with $p = 0$.

Suppose now that $u : \Re \to \Re$ is the solution of the Helmholtz equation

$$\tfrac{1}{2}\Delta u(\mathbf{x}) + q(\mathbf{x})\, u(\mathbf{x}) + p(\mathbf{x}) = 0, \quad \mathbf{x} \in D \tag{20}$$

with the Dirichlet boundary conditions given by Equation (16). The local solution is less simple in this case because the term $q(\mathbf{x})\, u(\mathbf{x})$ does not appear in the generator of the Brownian motion. In order to obtain this solution, it is necessary to augment the Brownian motion process $\mathbf{B}$ with an additional state Z defined by

$$\mathrm{d}Z(t) = q(\mathbf{B}(t))\, Z(t)\, \mathrm{d}t \tag{21}$$

with the initial condition $Z(0) = 1$ so that

$$Z(t) = \exp\left[\int_0^t q(\mathbf{B}(s))\, \mathrm{d}s\right] \tag{22}$$

This integral is known as the Feynman–Kac functional. Equation (10) can be extended to functions $u(\mathbf{B}(t))\, Z(t)$ of the process $(\mathbf{B}, \mathbf{Z})$ and takes the form

$$E^x[u(\mathbf{B}(T))\, Z(T)] - u(\mathbf{x})\, Z(0) = E^x\left[\int_0^T Z(s)\left(\frac{1}{2}\Delta u(\mathbf{B}(s)) + q(\mathbf{B}(s))\, u(\mathbf{B}(s))\right) \mathrm{d}s\right] \tag{23}$$

This formula can be simplified because $Z(0) = 1$, $u(\mathbf{B}(T)) = \xi(\mathbf{B}(T))$ by the boundary conditions, $\frac{1}{2}\Delta u(\mathbf{B}(s)) + q(\mathbf{B}(s))\, u(\mathbf{B}(s)) = -p(\mathbf{B}(s))$ by Equation (20), and the observation that $\mathbf{B}(s) \in D$ for $s \in (0, T)$. The new version of Equation (23) is

$$u(\mathbf{x}) = E^x\left[\xi(\mathbf{B}(T))\, \exp\left(\int_0^T q(\mathbf{B}(s))\, \mathrm{d}s\right)\right] + E^x\left[\int_0^T \exp\left(\int_0^s q(\mathbf{B}(\sigma))\, \mathrm{d}\sigma\right) p(\mathbf{B}(s))\, \mathrm{d}s\right] \tag{24}$$

This formula relates the value of the unknown function u at $\mathbf{x} \in D$ to some expectations depending on known functions, the functions q and p, and samples of the Brownian motion process $\mathbf{B}$. These expectations can be estimated by Monte Carlo simulation. If q is a constant, the formula of Equation (24) simplifies to

$$u(\mathbf{x}) = E^x[\xi(\mathbf{B}(T))\mathrm{e}^{qT}] + E^x\left[\int_0^T \mathrm{e}^{qs}\, p(\mathbf{B}(s))\mathrm{d}s\right] \tag{25}$$

The above discussion, related to Dirichlet boundary conditions, can be extended to Neumann boundary conditions by 'reflecting' the Brownian motion at a Neumann boundary. Details are available in References [1, 15]. Please see a numerical example in Section 5.1.5.

4. LOCAL SOLUTIONS OF MORE GENERAL PARABOLIC/ELLIPTIC PARTIAL DIFFERENTIAL EQUATIONS BY THE RWM

Let $u: \Re^n \to \Re$ be the steady-state solution of Equation (6) satisfying the prescribed Dirichlet conditions. Then Equation (6) can be written as

$$Lu(\mathbf{x}) + q(\mathbf{x})u(\mathbf{x}) + p(\mathbf{x}) = 0, \quad \mathbf{x} \in D \tag{26}$$

satisfying the boundary conditions

$$u(\mathbf{x}) = \xi(\mathbf{x}), \quad \mathbf{x} \in \partial D \tag{27}$$

where p, q and ξ are specified functions and L is a parabolic/elliptic differential operator defined on twice continuously differential functions and is of the form

$$L = \sum_i \alpha_i \frac{\partial}{\partial x_i} + \frac{1}{2} \sum_{i,j} \beta_{i,j} \frac{\partial^2}{\partial x_i \partial x_j} \tag{28}$$

For simplicity, let $q(\mathbf{x}) = 0$. It can be seen the operator given in Equation (28) can be made identical to the generator given in Equation (13) if $\mathbf{b}$ and $\boldsymbol{\sigma}$ are chosen appropriately, i.e. if $\mathbf{b} = \boldsymbol{\alpha}$ and $\boldsymbol{\sigma}\boldsymbol{\sigma}^{\mathrm{T}} = \boldsymbol{\beta}$. Then, making use of Equation (26), the boundary condition (27) and Dynkin's formula (Equation (11)), one has

$$u(\mathbf{x}) = E^x[\xi(\mathbf{X}(T))] + E^x\left[\int_0^T p(\mathbf{X}(s))\,\mathrm{d}s\right] \tag{29}$$

The expectations can now be obtained using Monte Carlo simulations.

To solve for transient solutions or for the cases where $q(\mathbf{x}, t) \neq 0$, the Itô diffusion $\mathbf{X}$ needs to be modified by making use of the Feynman–Kac functional [15].

5. NUMERICAL RESULTS AND DISCUSSION

Numerical results using the random walk method (RWM) have been obtained for two- and three-dimensional problems governed by Laplace, Poisson or Helmholtz equation and for an elliptic partial differential equation with variable coefficients. A study of the parameters involved in the method has been carried out. The two important parameters that influence the performance of the method are:

- time step (Δt),
- number of samples (n).

Also, issues such as imposing of Neumann boundary conditions and the effect of the chosen random number generator on the numerical results obtained, are studied in the following sections.

One of the major drawbacks of the RWM proposed herein is that it can take a long time to solve a problem on a large domain, the reason being that the Brownian motion or the Itô diffusion originating at a point (inside the body) may not go directly to the boundary because the direction of advancement is chosen at random. However, the better side of this disadvantage is that the method is very easily amenable for parallel computation. This fact has been fully exploited in this paper and an extensive comparison between serial and parallel codes is carried out.

Unlike the conventional finite difference, finite element or boundary element methods, the final computer program for obtaining the solution at a point can be extremely simple. In fact, MATLAB codes for solving some problems are presented in this paper.

As a measure of the performance of the method, an L_2 error norm is constructed as

$$\varepsilon = \frac{100}{|\Psi|_{\max}} \sqrt{\frac{1}{N} \sum_{i=1}^{N} (\Psi^{\mathrm{n}} - \Psi^{\mathrm{e}})^2}\,\% \tag{30}$$

where Ψ^{e} is the exact solution, Ψ^{n} is the numerical solution and N is the number of randomly chosen points in the domain where the unknown function Ψ is evaluated.

Numerical solutions using the RWM for the various governing equations are presented below. Section 5.1 describes the application of the RWM to problems governed by Laplace's equation. Section 5.2 describes the solution of various problems governed by Poisson's equation. The solution to Helmholtz's equation using the RWM is presented in Section 5.3. An elliptic partial differential equation with variable coefficients is solved in Section 5.4.

5.1. Laplace's equation

Laplace's equation in n-dimensions can be written as

$$\triangle \Psi = \frac{\partial^2 \Psi}{\partial x_i \partial x_i} = 0, \quad i = 1, 2, \ldots, n \tag{31}$$

where Ψ is the potential. The RWM can be used to solve Laplace's equation in more than three dimensions, however in this paper it is used to solve only two- and three-dimensional problems.

The numerical results presented in this section are organized as follows. In Section 5.1.1 the RWM is used to solve for the potential (Ψ) at various points in a two-dimensional multiply connected domain. The two important parameters of the RWM, namely, the time step (Δt) and the number of samples of the Brownian motion (n), are studied very carefully for a Dirichlet problem on a rectangle in Section 5.1.2. The RWM is applied to a physical problem of flow over a semi-circular bump in Section 5.1.3. Section 5.1.4 presents numerical results obtained by applying the RWM to the problem of an infinite domain with an internal cavity subject to Dirichlet boundary conditions. The RWM for problems with mixed boundary conditions involves the reflection of the Brownian motion at the Neumann boundary. This idea is used to solve a problem with mixed boundary conditions on a cube in Section 5.1.5.

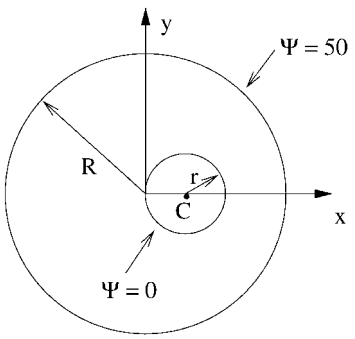

Figure 1. Laplace's equation solved on a multiply connected domain with Dirichlet boundary conditions.

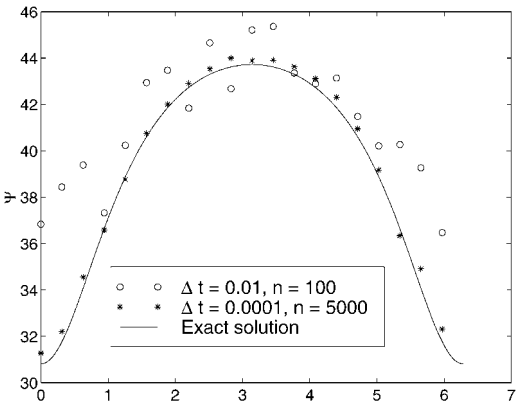

Figure 2. Potential (Ψ) computed along a circle of radius $r_0 = 0.75$ for a two-dimensional multiply connected domain.

5.1.1. Multiply connected domain. Laplace's equation is solved on a two-dimensional multiply connected domain as shown in Figure 1. The problem consists of obtaining the potential within the domain for prescribed potentials on the inner and outer circle. The numerical solution has been obtained by using $R = 1.0$, $r = 0.25$, and for the boundary conditions shown in Figure 1. The exact solution can be easily obtained using conformal mapping (see Reference [16]) and is given as

$$\Psi_{\text{exact}} = 50\left[1 - \frac{1}{2\ln b}\ln\left[\frac{(x-a)^2 + y^2}{(ax-1)^2 + a^2y^2}\right]\right] \tag{32}$$

where $a = b = 2 + \sqrt{3}$.

The RWM is used to compute the potential along a circle of radius $r_0 = 0.75$, centred at the origin. Figure 2 shows a comparison between the numerical solution (using Equation (19)) and the exact solution, using two sets of parameter values of the Brownian motion, namely, $\Delta t = 0.01$, $n = 100$ and $\Delta t = 0.0001$, $n = 5000$. Figure 2 also shows that the accuracy of the solution improves upon *decreasing* the time step (Δt) and *increasing* the number of samples (n).

An important aspect of any new numerical method is the computational burden associated with it. The proposed numerical method is quite computer intensive. However, it is very easily amenable for parallel computation. The basic idea behind the parallel program is to send a sample of the Brownian motion to each slave node. Once a Brownian motion, say at slave node i, reaches the boundary, the value of the potential at the boundary is reported by that slave node i to the Master node. Then the next sample of Brownian motion is given to this slave node i, and so on, until the total number of samples are exhausted. This algorithm is repeated at all the points in D at which the solution is desired. It can be clearly seen that communication between the processors is minimal in the parallel implementation of the RWM.

The parallel RWM code is run on multiple processors using the message passing interface (MPI) standard on the IBM SP2 (R6000 architecture, 120 MHz P2SC Processor). A comparison

Table III. Comparison of wall-clock times between serial and parallel computer codes for obtaining the solution at n_p points along a circle of radius $r_0 = 0.75$ ($\Delta t = 0.0001$, $n = 5000$).

Number of points, n_p	Serial code	Parallel code 4 Procs	8 Procs	12 Procs
15	14 min 27 s	4 min 55 s	2 min 28 s	1 min 42 s
20	19 min 17 s	6 min 29 s	2 min 49 s	1 min 56 s
30	28 min 58 s	9 min 44 s	4 min 38 s	2 min 56 s

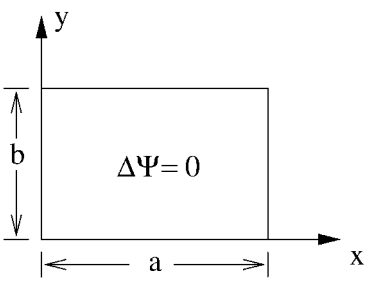

Figure 3. Dirichlet problem on a rectangle.

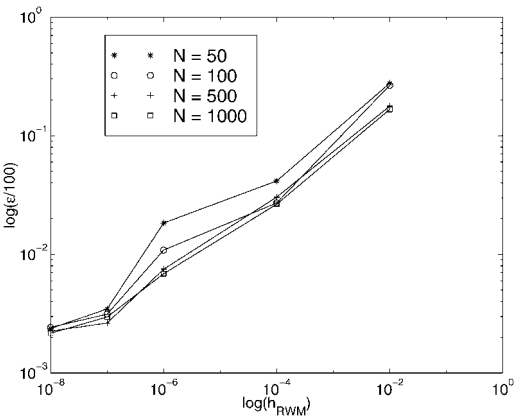

Figure 4. Convergence in the L_2 error for the Dirichlet problem on a rectangle using a cubic exact solution.

of the running times for different number of points along the circle, of radius $r_0 = 0.75$, (centred at the origin) is shown in Table III. It can be clearly seen from Table III that order of magnitude gains are possible by using a parallel version of the serial program.

5.1.2. Dirichlet problem on a rectangle. The RWM is used to solve a Dirichlet problem on a rectangle (Figure 3). The main idea behind choosing such a simple problem is to study two important parameters of the RWM namely, the time step (Δt) and the number of samples (n) of the Brownian motion.

First, a linear exact solution is imposed on the boundary of a square ($a = b = 1.0$). The exact solution is given as

$$\Psi_{\text{exact}} = x + y \tag{33}$$

Table IV shows a comparison in the L_2 error (ε) (from Equation (39)) for various values of the parameters Δt and n. The L_2 error (ε) is computed using $N = 50$, 100, 500, and 1000 points, respectively, randomly chosen within the domain. It can be seen from Table IV that choosing even 100 points gives a good estimate of the performance of the method. Also, an important observation is that the L_2 error decreases by *increasing* the number of samples (n) and *decreasing* the time step (Δt). In fact, a parameter equivalent to the mesh size (h) in the

Table IV. L_2 error in Ψ for a Dirichlet problem on a rectangle using a linear exact solution.

RWM parameters	L_2 error (%) $N=50$	$N=100$	$N=500$	$N=1000$
$\Delta t=0.1,\ n=10$	16.9370	17.2344	13.2835	12.9526
$\Delta t=0.01,\ n=100$	3.0246	1.9061	2.6849	2.4661
$\Delta t=0.001,\ n=1000$	0.7253	0.6947	0.6751	0.6610
$\Delta t=0.0005,\ n=5000$	0.2771	0.2759	0.2776	0.2765
$\Delta t=0.0001,\ n=10000$	0.1829	0.1924	0.2087	0.1995

Table V. Comparison of wall-clock times for Dirichlet problem on a rectangle using $\Delta t=0.0005,\ n=5000$ for various number of points in the domain.

Number of points, N	Serial code	Parallel code 4 Procs	8 Procs	16 Procs
50	3 min 24.3 s	2 min 6.6 s	56.0 s	29.8 s
100	6 min 46.6 s	3 min 53 s	1 min 44.9 s	50.7 s
500	33 min 10.8 s	17 min 14.4 s	7 min 29.5 s	3 min 34.3 s
1000	1 h 6 min 13.8 s	34 min 32.7 s	14 min 50.9 s	7 min

conventional FEM is proposed here. It can be written as

$$h_{\mathrm{RWM}}=\frac{\Delta t}{n} \tag{34}$$

It can be seen from Table IV that the L_2 error decreases with decreasing values of h_{RWM}. However, the proposed parameter h_{RWM} needs to be calculated using reasonable values of the time step (Δt) and number of samples (n) i.e. for $h_{\mathrm{RWM}}=10^{-4}$ using $\Delta t=10^{-4}$ and $n=1$ would certainly give meaningless results!

The RWM is also tested using a cubic function which satisfies the Laplace's equation. The exact solution, imposed on the boundary of the unit square, is

$$\Psi_{\mathrm{exact}}=x^3+y^3-3x^2y-3xy^2 \tag{35}$$

Figure 4 shows the convergence in the L_2 error with respect to the proposed parameter h_{RWM}. Again, it can be seen from Figure 4 that there is hardly any difference in the L_2 error for various numbers of randomly chosen points (N) in the domain.

As mentioned previously, the computational cost associated with a numerical method needs to be computed. The computational burden associated with measuring the performance of the method, i.e. computation of the L_2 error, is tabulated next. Table V shows a comparison of wall-clock times between a serial code and a parallel code run on 4, 8, and 16 Processors for the parameter values $\Delta t=0.0005,\ n=5000$. It can be seen that order of magnitude gains are possible by carrying out a parallel implementation of the serial code.

One of the main features of the proposed local method is the simplicity of the computer program involved for obtaining the solution at a point. To further illustrate this fact, the MATLAB program for obtaining a local solution using the RWM is shown in Table VI. Equations (19) and (33) are used in line 8 of the MATLAB program. The simplicity of the computer program speaks for itself.

Table VI. MATLAB program for obtaining a local solution for the Dirichlet problem on a rectangle for a linear exact solution.

```
1    function psi=prect (x, y, dt, n, a, b, nseed);
2    randn('seed',nseed); psi=0;
3    for k=1:n,
4      xe=x; ye=y;
5      while xe>0 & xe < a & ye > 0 & ye < b,
6          xe=xe+sqrt(dt)*randn;   ye=ye + sqrt(dt)*randn;
7      end
8    psi=psi+(xe + ye);
9    end
10   psi=psi/n;
```

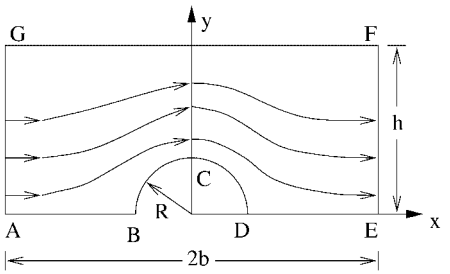

Figure 5. Schematic of fluid flow over a semi-circular bump.

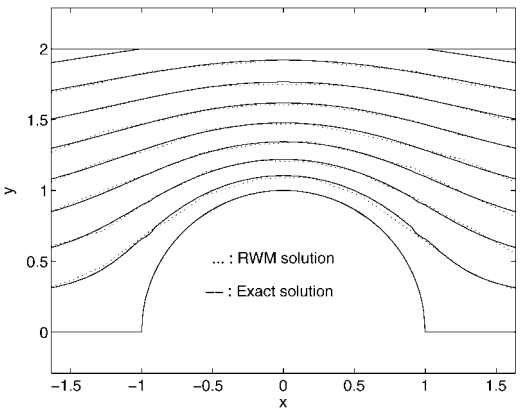

Figure 6. Streamlines for the problem of flow over a semicircular bump.

5.1.3. Flow over a semi-circular bump. An application of the RWM to a physical example is considered here. Figure 5 shows a schematic of the streamlines ($\Psi = \text{constant}$) of an inviscid, incompressible fluid over a semi-circular bump. The problem consists of determining the stream function Ψ and the components of the velocity at various points near the semi-circular bump.

The exact solution for the stream function, again imposed on the boundary ABCDEFGA, is given as (see Reference [16])

$$\Psi(x, y) = U_0 y \left[1 - \frac{R^2}{r^2}\right], \quad r^2 = x^2 + y^2 \tag{36}$$

where U_0 is the x-component of the upstream velocity and R is the radius of the semi-circular bump. The value of the stream function (Ψ) is computed at various points using the RWM (Equation (19)). Figure 6 compares the numerically generated contour plot of the streamlines with the exact solution. Figure 6 shows a good agreement between the RWM solution and

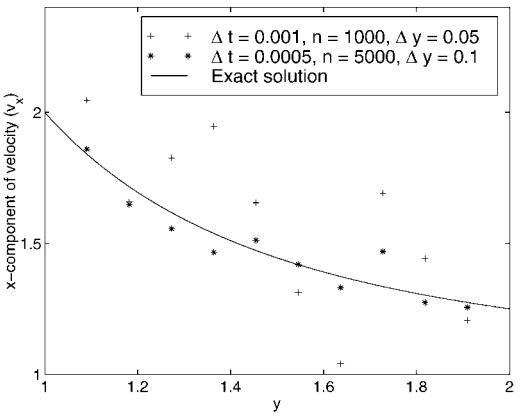

Figure 7. x-component of the velocity (v_x) along the y-axis.

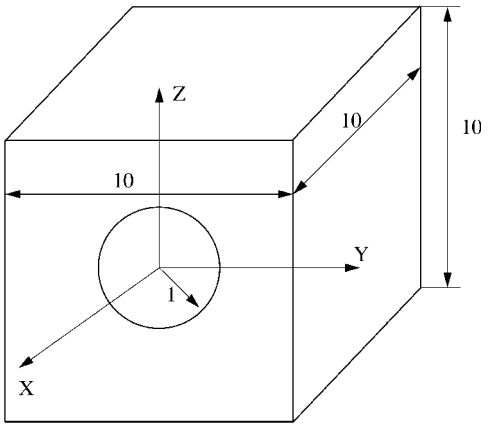

Figure 8. Geometry of the Dirichlet problem of an infinite domain with a spherical cavity.

the exact solution. The following non-dimensional parameters have been used for this problem: $U_0 = 1.0,\ R = b = 1.0,\ h = 2.0$.

The components of the velocity can be obtained from the stream function using the well-known relations (see Reference [17])

$$v_x = -\frac{\partial \Psi}{\partial y}, \quad v_y = \frac{\partial \Psi}{\partial x} \tag{37}$$

The RWM can be directly used only to get only the values of Ψ at various points, and not the velocities. However, in order to obtain the components of the velocity, one could employ a central difference scheme. The central difference formula used to obtain the results can be written as

$$v_x(y) = -\left[\frac{\Psi(0, y + \Delta y) - \Psi(0, y - \Delta y)}{2\Delta y}\right] \tag{38}$$

where Δy is chosen through numerical experiments. The x-component of the velocity (v_x) along the y-axis is shown in Figure 7. It can be clearly seen from Figure 7 that the results are not very accurate for the velocity. It is felt that better techniques need to be developed in order to obtain the components of the velocity i.e. gradient of the potential (Ψ).

5.1.4. Dirichlet problem in an infinite domain (in three dimensions) with a spherical cavity. A three-dimensional example of solving a Laplace problem with Dirichlet boundary conditions with the RWM, is presented next. Consider a three-dimensional problem of finding the temperature distribution is an infinite domain with a spherical cavity subject to a uniform temperature distribution on the surface of the cavity.

The exact solution is given as (see Reference [18])

$$T(r) = T_c\left(\frac{a}{r}\right) \tag{39}$$

where T_c is the temperature of the cavity wall and a is the radius of the cavity.

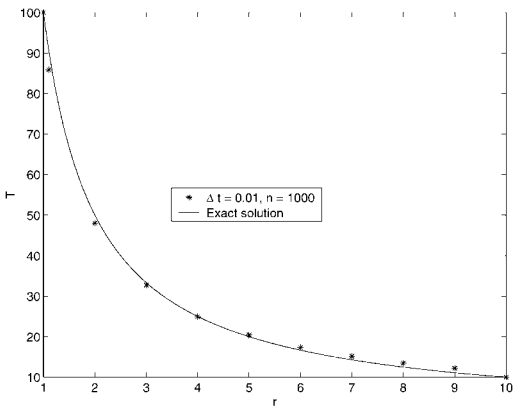

Figure 9. Variation of temperature T with radius r.

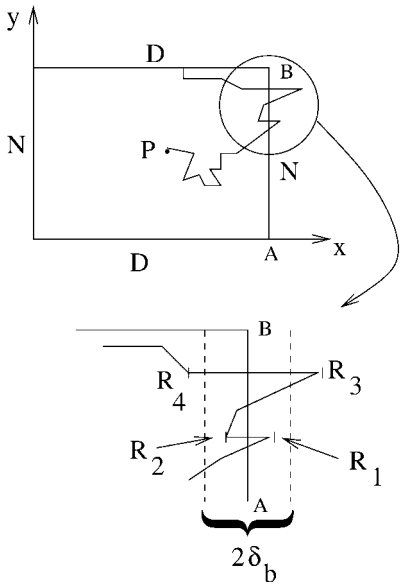

Figure 10. 'Reflection' of the Brownian motion at the Neumann boundary.

The infinite domain with the cavity is modelled as a cube having a cavity at the centre (see Figure 8). The dimension of the cube is taken to be much larger than the radius of the cavity. To be specific, the side of the cube is taken to be 10 units while the radius of the cavity is taken to be 1 unit. A uniform temperature of 100°C is imposed on the cavity surface and a temperature of 10°C is imposed on the sides of the cube.

The RWM is used with $\Delta t = 0.01$ and $n = 1000$. Since the problem is spherically symmetric, 100 random points are selected on the surface of spheres of pre-determined radii (>1) and the temperature at a particular radius is reported as an average of the temperatures of the 100 points on that sphere. Figure 9 shows the comparison of the numerical solution with the analytical one. The numerical solution is found to match the analytical solution quite closely inspite of using a large step size.

5.1.5. Mixed problem on a cube. The RWM is used to solve Laplace's equation in a cube, with mixed boundary conditions. In order to solve problems with mixed boundary conditions using the RWM, one needs to 'reflect' the Brownian motion at the Neumann boundary, i.e. the boundary where the flux $q \equiv \partial u/\partial n$ is prescribed. The details of this procedure can be found in References [1, 15]. However, a simple explanation of the reflection procedure follows.

Figure 10 shows a schematic demonstrating the idea of reflection of the Brownian motion. The main idea is to prevent the Brownian motion, originating inside the domain, from escaping the part of the boundary on which the flux is prescribed. Consider the boundary crossing of the Brownian motion to the point R_1, as shown in Figure 10. Now, the Brownian motion is first reflected (across the boundary AB) to the point R_2 and then continued from there. Now, suppose that the Brownian motion crosses the boundary again to the point R_3. It is

Table VII. L_2 error in Ψ for a problem with mixed boundary conditions on a cube for $\Delta t = 0.001$, $n = 1000$.

	L_2 error (%)		
δ_b	$N = 50$	$N = 100$	$N = 500$
0.1	3.336	3.575	3.515
0.05	3.335	3.586	3.534
0.01	3.332	3.597	3.543
0.001	3.314	3.584	3.548

reflected back to R_4 and so on. Eventually, the Brownian motion reaches a Dirichlet part of the boundary and escapes. This value of the prescribed function, at the boundary crossing point, is recorded as usual.

The importance of the 'boundary layer', of specified thickness δ_b, is that the time spent by the Brownian motion trapped in this boundary layer region is accumulated and is used in the formula for computing the value of Ψ at an internal point (see Reference [1]).

The reflection procedure described above is implemented by employing the following harmonic function:

$$\Psi_{\text{exact}} = x + 2y + 3z \tag{40}$$

Dirichlet boundary conditions (corresponding to Equation (40)) are prescribed on the $y, z = \pm 1$ faces and Neumann conditions (corresponding to Equation (40)) are prescribed on the $x = \pm 1$ faces of the cube.

First, the effect of the boundary layer δ_b is investigated by computing the L_2 error in Ψ at 50, 100, and 500 points in D, for various values of δ_b. Table VII shows that the numerical method is quite insensitive to the boundary layer thickness δ_b chosen to obtain the numerical results.

A convergence study is carried out with respect to the proposed parameter h_{RWM} for the problem with mixed boundary conditions. Figure 11 shows the numerical results obtained. A convergence rate of 0.52 is observed. It is noted in Figure 11 that for $\Delta t = 10^{-6}$, $n = 10^6$, the parallel RWM code takes 2 min to run to completion on 64 Processors, in order to compute the L_2 error for $N = 100$ random points.

5.2. *Poisson's equation*

Poisson's equation governs a wide range of problems in engineering and physics. Poisson's equation in n-dimensions can be written as

$$\triangle \Psi = \frac{\partial^2 \Psi}{\partial x_i \partial x_i} = f, \quad i = 1, 2, \ldots, n \tag{41}$$

where f is a given function of the co-ordinates. The following three examples are considered. Section 5.2.1 describes the solution of Poisson's equation on an ellipsoid. Section 5.2.2 solves a physical problem of torsion of a shaft with a triangular cross-section. Section 5.2.3 obtains the results for a shaft with a circular cross-section with a groove.

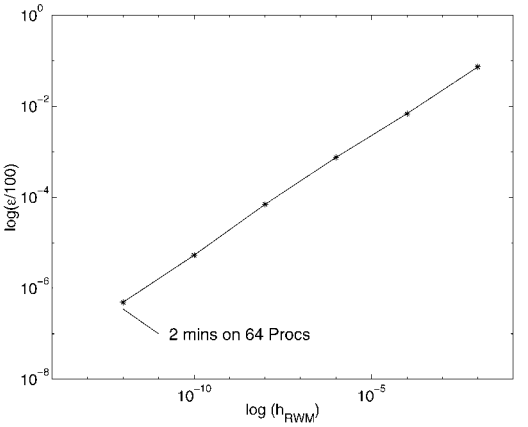

Figure 11. Convergence results for the mixed problem on a cube ($N = 100$).

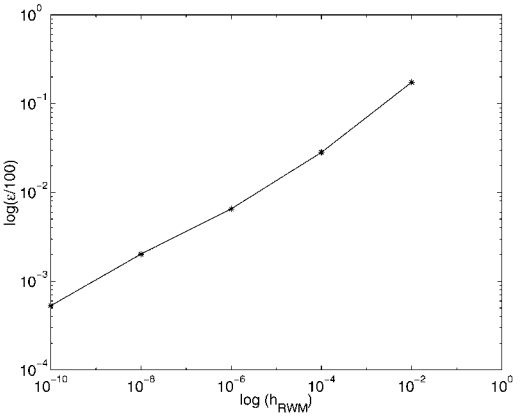

Figure 12. Convergence study for the solution to Poisson's equation on an ellipsoid using $N = 50$ random points.

5.2.1. Poisson's problem on an ellipsoid. The RWM is used to solve Poisson's equation (with Dirichlet boundary conditions) on an ellipsoid. The equation of the ellipsoid can be written as

$$\frac{x^2}{a^2} + \frac{y^2}{b^2} + \frac{z^2}{c^2} = 1 \tag{42}$$

The numerical results have been obtained (using Equation (18)) by imposing a quadratic exact solution, given as

$$\Psi_{\text{exact}} = x^2 + y^2 + z^2 \tag{43}$$

on the surface of the ellipsoid. Thus, for the quadratic solution imposed, $f = 6$ in Equation (41). Figure 12 shows the convergence in the L_2 error with respect to the parameter h_{RWM}. A convergence rate of 0.31 is observed.

As mentioned before, the computer programs involved for obtaining the local solution using the RWM can be very simple. Table VIII shows the MATLAB program that can be used to obtain a local solution (to Poisson's equation) using the RWM. Equations (18) and (43) are used in line 11 of the MATLAB program. It can be clearly seen from Table VIII that the program is extremely easy to read, write and debug! Again, one of the main strengths of the present method is the simplicity of the computer programs involved.

5.2.2. Torsion of a shaft with a triangular cross-section. The problem of torsion of a shaft is governed by Poisson's equation. The governing equation can be written as (see Reference [19])

$$\frac{\partial^2 \Psi}{\partial x^2} + \frac{\partial^2 \Psi}{\partial y^2} = -2G\theta, \quad \Psi = 0 \quad \text{on } \partial D \tag{44}$$

where Ψ is the *stress function*, G is the shear modulus and θ is the angle of twist per unit length. This Poisson's equation is solved on a triangular cross-section shown in Figure 13.

Table VIII. MATLAB program for obtaining a local solution for the Poisson's equation on an ellipsoid for an exact solution in Equation (43).

```
psi=function pellip (x, y, z, dt, n, q, a, b, c, nseed)
randn('seed',1); psi = 0;
for k=1:n,
  xe=x; ye=y; ze=z; time=0;
  rcomp=sqrt(xe*xe/(a*a)+ye*ye/(b*b)+ze*ze/(c*c));
  while rcomp<1,
      xe=xe+sqrt(dt)*randn; ye=ye+sqrt(dt)*randn; ze=ze+sqrt(dt)*randn;
  rcomp=sqrt(xe*xe/(a*a)+ye*ye/(b*b)+ze*ze/(c*c));
  time=time+dt;
  end
psi=psi + (xe*xe+ye*ye+ze*ze)-3.*time;
end
psi=psi/n;
```

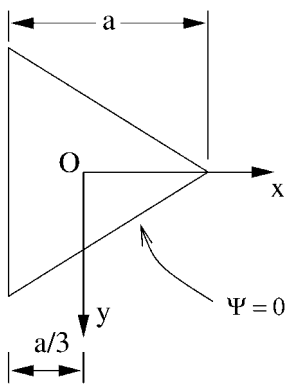

Figure 13. Torsion of a prismatic bar with the cross-section as an equilateral triangle.

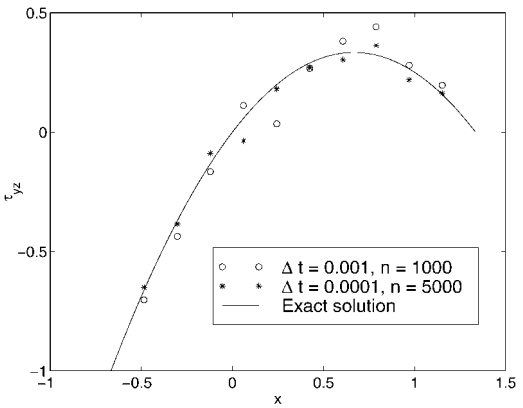

Figure 14. Shearing stress along the x-axis for the torsion problem on an equilateral triangle using $\Delta x = 0.05$.

The exact solution, for the cross-section shown in Figure 13, can be written as (see Reference [19])

$$\Psi_{\text{exact}} = -G\theta\left[\frac{1}{2}(x^2+y^2) - \frac{1}{2a}(x^3-3xy^2) - \frac{2}{27}a^2\right] \tag{45}$$

$\Psi = 0$ is imposed on the boundary of the triangle (Dirichlet boundary conditions). The important physical quantity, the shearing stress, can be obtained from the stress function as

$$\tau_{xz} = \frac{\partial \Psi}{\partial y}, \quad \tau_{yz} = -\frac{\partial \Psi}{\partial x} \tag{46}$$

Since the RWM provides only the values of Ψ at certain points, a central difference scheme is used to obtain the shearing stress. The shearing stress (τ_{yz}) is computed along the x-axis

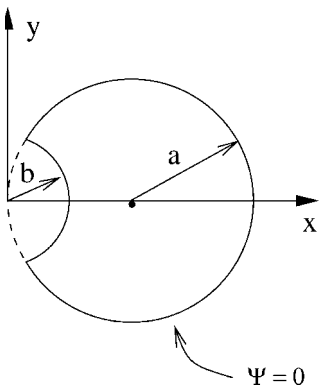

Figure 15. Elastic torsion of a circular shaft with a small grove.

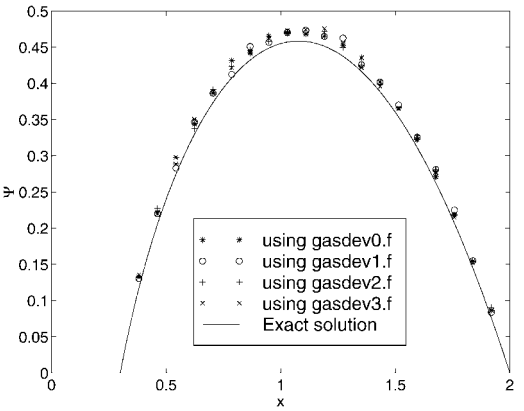

Figure 16. Comparison of Ψ along the x-axis using various random generators for the torsion of a circular shaft with a grove ($\Delta t = 0.0005,\ n = 5000$).

using the central difference formula written as

$$\tau_{yz}(x,0) = \frac{\Psi(x+\Delta x,0) - \Psi(x-\Delta x,0)}{2\Delta x} \tag{47}$$

Figure 14 shows a comparison between the numerical results and the exact solution in the shearing stress (τ_{yz}) along the x-axis. Figure 14 shows that acceptably accurate results can be obtained for the shearing stress in spite of using a central difference scheme in conjunction with the RWM.

5.2.3. Torsion of a circular shaft with a groove. The governing equation for torsion is given in Equation (44). In this section, the torsion problem is solved for a circular shaft with a grove. Figure 15 shows a schematic of the circular cross-section with a grove. The exact solution for Ψ can be written as (see Reference [19])

$$\Psi_{\text{exact}} = -\frac{1}{2}(r^2 - b^2) + a\left(r - \frac{b^2}{r}\right)\cos\theta, \quad \Psi = 0 \quad \text{on } \partial D \tag{48}$$

where (r,θ) are the usual polar co-ordinates. Again, $\Psi = 0$ is prescribed on the boundary of the region shown in Figure 15.

Since the proposed numerical method, the RWM, is based on random numbers, a careful study of the random number generator, used to obtain the numerical results, is in order. The computer programs presented in *Numerical Recipes* [20, chapter 7] are used to generate the random numbers. The book proposes various random number generators as a source of uniform deviates `ran0`, `ran1`, `ran2`, `ran3`. Each of these uniform deviates can be used to generate a normally distributed deviate with zero mean and unit variance. A careful study of the various random number generators, referred to as `gasdev0`, `gasdev1`, `gasdev2`, `gasdev3` is carried out.

Table IX. Comparison of wall-clock times between serial and parallel computer codes for obtaining the solution at n_p points along the x-axis ($\Delta t = 0.0005,\ n = 5000$).

Number of points, n_p	Serial code	Parallel code		
		4 Procs	8 Procs	12 Procs
15	3 min 35.9 s	1 min 11.9 s	32.2 s	22.2 s
20	4 min 43.5 s	1 min 34.2 s	42.4 s	29.2 s
30	7 min	2 min 19.1 sec	1 min	40.3 s

Figure 16 shows a comparison of Ψ along the x-axis for various random number generators. It can be clearly seen that the RWM is insensitive to the random number generator used to obtain the numerical results. Table IX presents a comparison of wall-clock times between serial and parallel codes for obtaining the solution at varying number of points along the x-axis. It is seen that significant gains are obtained by parallelization.

5.3. *Helmholtz's equation*

The Helmholtz equation in n-dimensions can be written as

$$\triangle\Psi + q\Psi = \frac{\partial^2 \Psi}{\partial x_i \partial x_i} + q\Psi = 0, \quad i = 1, 2, \ldots, n \tag{49}$$

The Helmholtz equation is solved on a three-dimensional problem using the RWM. The idea behind choosing this example is to study the variable q appearing in Equation (49). The RWM is used to solve a Dirichlet problem on a unit sphere for various values of q. The exact solution for $q > 0$ and for $q < 0$ can be written as

$$\Psi(x, y, z) = \cos(ax)\cos(ay)\cos(az) \quad \text{for } q > 0 \tag{50}$$

$$\Psi(x, y, z) = \exp(ax)\exp(ay)\exp(az) \quad \text{for } q < 0 \tag{51}$$

The constant a in the exact solution (Equations (50) and (51)) is chosen so as to satisfy the Helmholtz Equation (49). As before, Ψ from Equation (50) or (51) is prescribed on the surface of the unit sphere.

Figure 17 shows a comparison between the exact solution and the RWM solution (using Equation (25)) obtained at the origin of the sphere for various values of q. Now, since $\Psi(0, 0, 0) = 1.0$, for $q < 0$ and $q > 0$, one can evaluate the performance of the numerical method for changing values of q by calculating the value of Ψ at the origin. It can be seen from Figure 17 that the results deteriorate for *large* values of $q > 0$. Figure 17 also shows that the numerical results for $q > 0$ do not improve upon decreasing the time step (Δt) and increasing the number of samples (n).

The observation made in Figure 17 can be easily explained by looking at the computer code used to obtain the numerical results. Table X presents the MATLAB code used to obtain the numerical results for this example (for $q > 0$). In order to compute the value of Ψ at a point inside the domain, the value of Ψ at the boundary where the Brownian motion exits the domain is multiplied by the term $\exp(qt)$. In particular, Equation (25) is used in conjunction with Equation (50) in line 11 of the MATLAB program. Now, for large values of

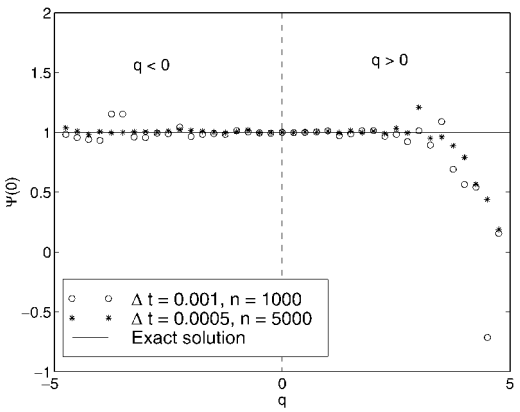

Figure 17. Dirichlet problem on a sphere: Solution to the Helmholtz equation at the origin.

Table X. MATLAB program for obtaining a local solution for the Helmholtz's equation on a sphere ($q>0$).

```
1  function psi=psphere (x, y, z, radius, dt, n, q, nseed);
2  a = sqrt(2*q/3); randn('seed',nseed); psi = 0;
3  for k = 1:n,
4     xe=x; ye=y; ze=z; time=0;
5     rcomp=sqrt(xe*xe + ye*ye + ze*ze);
6     while rcomp < radius,
7        xe=xe + sqrt(dt)*randn; ye=ye + sqrt(dt)*randn; ze=ze + sqrt(dt)*randn;
8        rcomp=sqrt(xe*xe + ye*ye + ze*ze);
9        time = time + dt;
10    end
11 psi= psi + exp(q*time)* cos(a*xe)* cos(a*ye)* cos(a*ze) ;
12 end
13 psi=mean(psi);
```

$q>0$, this term ($\exp(qt)$) can be unbounded. And that is precisely the reason for the failure of the method for large values of $q>0$. However, this situation does not arise when $q<0$ and accurate numerical results are obtained (see Figure 17).

5.4. Elliptic PDE with variable coefficients

Consider the following non-homogeneous elliptic partial differential equation:

$$x^2\frac{\partial^2 g}{\partial x^2} + y^2\frac{\partial^2 g}{\partial y^2} = 2x^2y^2 \tag{52}$$

which is a special case of Equation (6). This equation is solved in the domain $1\leqslant x\leqslant 2$, $1\leqslant y\leqslant 2$ subject to Dirichlet conditions on $x=1,2$ and $y=1,2$. The exact solution is given by

$$g(x,y)=\tfrac{1}{2}x^2y^2$$

Table XI. L_2 error in $g(x, y)$ for the problem defined by Equation (52).

RWM parameters	L_2 error $N = 500$
$\Delta t = 0.01,\ n = 1000$	0.9413%
$\Delta t = 0.001,\ n = 1000$	0.6729%

First, an Itô diffusion whose generator coincides with the operator

$$\frac{1}{2}\left(x^2\frac{\partial^2}{\partial x^2} + y^2\frac{\partial^2}{\partial y^2}\right)$$

is chosen. It can be observed from Equation (13) that the required process is obtained by setting $\mathbf{b} = \mathbf{0}$ and $\boldsymbol{\sigma} = \mathrm{diag}(x, y)$. Here $\mathrm{diag}(x, y)$ is a 2×2 diagonal matrix with x and y being the diagonal entries. Next Equation (12) is solved. For the case considered, the diffusion process can be explicitly found as

$$\begin{bmatrix} X(t) \\ Y(t) \end{bmatrix} = \begin{bmatrix} X(0)\exp\{B_1(t) - \frac{1}{2}t\} \\ Y(0)\exp\{B_2(t) - \frac{1}{2}t\} \end{bmatrix} \tag{53}$$

Here $B_1(t)$ and $B_2(t)$ are one-dimensional standard independent Brownian motions. Using Equations (53), (29) and Monte Carlo simulations, the solution to (52) is obtained. The results are shown in Table XI.

As seen from Table XI, the results obtained from RWM are quite accurate.

6. CONCLUSIONS

A local method, the random walk method (RWM), has been developed and implemented in this paper for the solution of the Laplace, Poisson, and Helmholtz equations in two- and three-dimensions as well as for an elliptic partial differential equations with variable coefficients. Extending this approach to n-dimensions, $n > 3$, is straightforward. Considerable insight has been gained into the various parameters involved in the method. The comments below summarize the numerical results:

- The two main parameters are the time step (Δt) and number of samples (n). It has been observed through numerical experiments that *decreasing* the time step (Δt) and *increasing* the number of samples (n) improves the overall accuracy of the numerical results.
- For problems with mixed boundary conditions, the parameter δ_b, which controls the 'boundary layer', has a marginal effect on the numerical solution.
- The numerical results tend to deteriorate when the derivative of Ψ is computed through a central difference scheme. An alternative technique for the evaluation of gradient needs to be looked into. This is planned for the future.
- The variable q in Helmholtz's equation plays an important role in the numerical solution. It is observed that very poor numerical results are obtained for *large* values of $q > 0$. This is a generic problem for the Helmholtz's equation, irrespective of the numerical method being used.

- Since the proposed numerical method is inherently parallel, significant gains in wall-clock times have been achieved by parallelizing the serial computer codes.

Overall, the random walk method (RWM) is an ideal choice for problems that can be solved by this method, especially in the early stages of the design where the solution is desired only at a few selected points.

ACKNOWLEDGEMENTS

This research has been supported by NSF Grants #CMS-9610022 and CMS-9912524 to Cornell University. The computing for this research was carried out using the resources of the Cornell Theory Center, which receives funding from Cornell University, New York State, the National Center for Research Resources at the National Institutes of Health, the National Science Foundation, the Defense Department Modernization program, the United States Department of Agriculture, and corporate partners. The discussions with John Zollweg have been extremely valuable.

REFERENCES

1. Chung KL, Williams RJ. *Introduction to Stochastic Integration* (2nd edn). Birkhäuser: Boston, 1990.
2. Courant R, Hilbert D. *Methods of Mathematical Physics*, vol. 2. Interscience Publishers: New York, 1966.
3. Durrett R. *Brownian Motion and Martingales in Analysis.* Wadsworth Advanced Books and Software: Belmont, CA, 1984.
4. Øksendal B. *Stochastic Differential Equations* (3rd edn). Springer: New York, 1992.
5. Grigoriu M, Mukherjee S. A local solution of the Schrödinger equation by Monte Carlo simulation. In *Proceedings of Fourth International Conference on Stochastic Structural Dynamics—SSD'98*, Spencer Jr. BF, Johnson EA (eds), Notre Dame, Indiana, 1998; 223–227.
6. Zienkiewicz OC. *The Finite Element Method* (3rd edn). McGraw-Hill: New York, 1977.
7. Banerjee PK. *The Boundary Element Methods in Engineering* (2nd edn). McGraw-Hill: New York, 1994.
8. Lord Rayleigh. On James Benoulli's theorem in probabilities. *Philosophical Magazine* 1899; **47**:246–251.
9. Courant R, Friedrichs K, Lewy H. Über die partiellen differenzen gleichungen der mathematischen Physik. *Mathematische Annalen* 1928; **100**:32–74.
10. Haji-Sheikh A, Sparrow EM. The floating random walk and its applications to Monte Carlo solutions of heat transfer. *Journal of SIAM*, *Applied Mathematics* 1966; **14**:370–389.
11. Sabelfeld K. *Monte Carlo Methods in Boundary Value Problems*. Springer: Berlin, 1991.
12. Nakamura S. *Computational Methods in Engineering and Science*. Wiley-Interscience Publications: New York, 1977.
13. Chorin A. Numerical study of slightly viscous flow. *Journal of Fluid Mechanics* 1973; **57**:785–796.
14. Shia D, Hui C. A Monte Carlo solution method for linear elasticity. *International Journal of Solids and Structures* 2000; **37**:6085–6105.
15. Grigoriu M. Solution of some elasticity problems by the random walk method. *Acta Mechanica* 1998; **125**: 197–209.
16. Greenberg MD. *Foundations of Applied Mathematics*. Prentice-Hall, Inc.: Englewood Cliffs, New Jersey, 1978.
17. Lamb H. *Hydrodynamics*. Cambridge University Press: Cambridge, 1993.
18. Özisik MN. *Boundary Value Problems of Heat Conduction*. International Textbook Company: Pennsylvania, 1968.
19. Timoshenko SP, Goodier JN. *Theory of Elasticity* (3rd edn). McGraw-Hill: New York, 1970.
20. Press WH, Teukolsky SA, Vetterling WT, Flannery BP. *Numerical Recipes in Fortran* (2nd edn). Cambridge University Press: Cambridge, 1992.

