

A PARALLEL ITERATIVE PROBABILISTIC METHOD FOR MIXED PROBLEMS OF LAPLACE EQUATIONS WITH THE FEYNMAN–KAC FORMULA OF KILLED BROWNIAN MOTIONS*

CUIYANG DING[†], CHANGHAO YAN[‡], XUAN ZENG[§], AND WEI CAI[¶]

Abstract. In this paper, a parallel probabilistic method using the Feynman–Kac formula of killed Brownian motions is proposed to solve the mixed boundary value problems (BVPs) of 3D Laplace equations. To avoid using reflecting Brownian motions and the calculation of their local time $L(t)$ in the Feynman–Kac representation of solutions for Neumann and Robin BVPs, the proposed method uses an iterative approach to approximate the solutions where each iteration will use the Feynman–Kac formula to solve a pure Dirichlet problem, thus only involving killed Brownian motions. First, the boundary of the domain is decomposed with overlapping local patches formed by the intersection of hemispheres superimposed on the domain boundary. The iteration starts with an arbitrary initial guess for the Dirichlet data on the Neumann and Robin boundaries; then, using the Feynman–Kac formula for a pure Dirichlet problem with the current available Dirichlet data on the whole boundary, the solution over the hemispheres can be obtained by the Feynman–Kac formula for the killed Brownian motions, sampled by a walk-on-spheres (WOS) algorithm. Second, by solving a local boundary integral equation (BIE) over each hemisphere and a local patch on the domain boundary, the Dirichlet data on the Neumann and Robin boundaries can be updated. By continuing this process, the proposed hybrid probabilistic and deterministic BIE–WOS method gives a highly parallel algorithm for the global solution of any mixed-type BVPs of the Laplace equations. Numerical results of various mixed interior and exterior BVPs demonstrate the parallel efficiency and accuracy of the proposed method.

Key words. Feynman–Kac formula, killed Brownian motions, Neumann and Robin boundary, walk-on-spheres (WOS), boundary integral equations, Laplace equation

MSC codes. 65C05, 65N99, 78M25, 92C45

DOI. 10.1137/22M1478458

1. Introduction. The Feynman–Kac formula establishes a connection between PDEs of elliptic and parabolic types to Ito diffusions, and this connection gives the possibility of finding PDE solutions locally, which is expressed as a weighted average of the domain boundary conditions. Stochastic methods based on the Feynman–Kac formula have been proposed for the Dirichlet boundary value problems (BVPs) of elliptic PDEs [24], [8], [10]. The probabilistic representation based on the Feynman–Kac formula for the Laplace equations with Dirichlet boundary conditions uses killed Brownian motions on the domain boundary, and averaging of the boundary data is taken over a hitting probability on the boundary. Due to the spatial homogeneity of the Brownian motions, efficient sampling methods using walk-on-spheres (WOS) can

*Submitted to the journal’s Methods and Algorithms for Scientific Computing section February 16, 2022; accepted for publication (in revised form) July 11, 2022; published electronically October 27, 2022.

<https://doi.org/10.1137/22M1478458>

Funding: The work of the first, second, and third authors was supported by National Key R & D program of China grants 2020YFA0711900 and 2020YFA0711904 and by National Natural Science Foundation of China (NSFC) research projects 61974032, 62141407, 61674042, and 61929102.

[†]State Key Laboratory of ASIC and System, Fudan University, Shanghai, 201203, China (20112020052@fudan.edu.cn).

[‡]Joint first author. State Key Laboratory of ASIC and System, Fudan University, Shanghai, 201203, China (yanch@fudan.edu.cn).

[§]Corresponding author. State Key Laboratory of ASIC and System, Fudan University, Shanghai, 201203, China (xzeng@fudan.edu.cn).

[¶]Corresponding author. Dept. of Mathematics, Southern Methodist University, Dallas, TX 75275 USA (cai@smu.edu).

be made to find the hitting probability distribution, resulting in the hitting probability. However, for the Neumann and Robin boundary conditions, reflecting Brownian motions (RBMs) are required, and the local time $L(t)$ of the RBMs, which records the frequency of the RBMs hitting and reflecting from the boundary, will also be needed to give the weight used for averaging the boundary data. The sampling of RBMs and the computation of the local time $L(t)$ pose great challenges for computational accuracy and efficiency, and several recent works [33], [31], [32] have addressed this issue, and other approaches to handling the reflections have also been studied [4], [19], [2], [16], [15], [11]. Finding the solution of the Neumann or Robin boundary condition has many applications in materials sciences, such as magnetic polarization tensors [28] and virtual mass [27] of nanoparticles or polymers of complex shapes as well as their conductivity and elastic moduli [7]. Due to the complex geometries involved, traditional finite element type methods face difficulties, while probabilistic methods using Brownian paths, without the need of complicated structured mesh structures, have been shown to have their advantages [12].

In this paper, we will propose a new probabilistic method for finding the global solution of mixed Laplace BVPs, including Neumann and Robin types, with the Feynman–Kac formula of killed Brownian motions only, while still maintaining the local solution feature of a probabilistic method. The method will be iterative in nature where each iteration will use the Feynman–Kac formula for a pure Dirichlet BVP. More importantly, by using a local boundary integral equation (BIE) approach, the method will be made local and massively parallel. At the convergence of the iterations, the Neumann-to-Dirichlet or Robin-to-Dirichlet mappings will be obtained. All these are achieved by a hybrid local BIE and the WOS method—the BIE-WOS method originally proposed in [29], [30] to obtain Neumann data on a Dirichlet boundary. The local BIE-WOS method is formed over local patches of the domain boundary in each iterative step so the Dirichlet data on patches with prescribed Neumann or Robin data can be updated with increasing accuracy. The iterative method decomposes a global potential field problem into a large number of small BIE problems which can be solved in parallel, allowing the full use of many-core teraflops computing platforms. Once the Dirichlet and Neumann data are obtained over the whole boundary, the potential solution in the whole domain can be obtained by a simple field evaluation, accelerated by fast multipole methods [9], with an integral representation using the boundary Dirichlet and Neumann data.

The rest of the paper is organized as follows. The Feynman–Kac formula and the WOS method are reviewed in section 2. In section 3, a BIE-WOS-iterative method with killed Brownian motions is introduced to solve mixed boundary value problems for Laplace equations. Several implementation details of the BIE-WOS-iterative method are given in section 4, including the pre-computation of the hitting probability of the Brownian motions on boundaries, reusing BIE coefficients and WOS probabilities, updating solutions in the iteration, and quadrature rules for integrals. Section 5 compares the BIE-WOS-iterative method with direct BIE and BIE-WOS methods using reflecting Brownian motions and discusses their different characteristics. Numerical results for various types of mixed boundary conditions for several domains are presented in section 6. Finally, a conclusion is given in section 7.

2. Feynman–Kac formula for boundary value problems and WOS. The Feynman–Kac formula [5], [6] relates the Ito diffusion paths $X_t \in \mathbb{R}^3$, defined by the following Ito stochastic differential equation:

$$(1) \quad dX_t = b(X_t)dt + \alpha(X_t)dB_t,$$

with $B_t(\omega)$ being the Brownian motion with possible reflections on Neumann or Robin boundaries to the solution $u(\mathbf{x})$ of the following BVPs of an elliptic PDE:

$$\begin{aligned}
 (2) \quad & L(u) = f(\mathbf{x}), \quad \mathbf{x} \in \Omega, \\
 & u = \phi_1(\mathbf{x}) \quad \text{on } \Lambda_1, \\
 & \frac{\partial u}{\partial \mathbf{n}} = \phi_2(\mathbf{x}) \quad \text{on } \Lambda_2, \\
 & \frac{\partial u}{\partial \mathbf{n}} - cu = \phi_3(\mathbf{x}) \quad \text{on } \Lambda_3.
 \end{aligned}$$

In the above, L is a uniformly elliptic differential operator,

$$(3) \quad L(u) \equiv \sum_{i=1}^3 b_i(\mathbf{x}) \frac{\partial u}{\partial x_i} - \sum_{i,j=1}^3 a_{ij}(\mathbf{x}) \frac{\partial^2 u}{\partial x_i \partial x_j},$$

and $[a_{ij}] = \frac{1}{2} \alpha(x) \alpha^T(x)$, and there exists a positive number $\mu > 0$ such that

$$(4) \quad \sum_{i,j=1}^3 a_{ij}(\mathbf{x}) \xi_i \xi_j \geq \mu |\boldsymbol{\xi}|^2 \quad \text{if } \mathbf{x} \in \Omega, \quad \boldsymbol{\xi} \in \mathbb{R}^3,$$

and $a_{ij}(\mathbf{x})$ and $b_i(\mathbf{x})$ are uniformly Lipschitz continuous on $\bar{\Omega} = \Omega \cup \partial\Omega$. Also, the boundary $\partial\Omega = \Lambda_1 \cup \Lambda_2 \cup \Lambda_3$ of the domain Ω is assumed to be C^2 , and the boundary data ϕ_i is assumed to be C^0 .

Specifically, a probabilistic representation of $u(\mathbf{y})$ for the mixed boundary value problem of the Laplace operator is given by a Feynman–Kac formula [22], [19]:

$$\begin{aligned}
 (5) \quad u(\mathbf{y}) = & E^x \left\{ \hat{e}_c(\tau_{\Lambda_1}) \phi_1(X_{\tau_{\Lambda_1}}) \right\} + \frac{1}{2} E^x \left\{ \int_0^\infty \hat{e}_c(t) \phi_{2,3}(X_t) dL(t) \right\} \\
 & + E^x \left\{ \int_0^\infty \hat{e}_c(t) f(X_t) dt \right\},
 \end{aligned}$$

where $X_t = B_t$ is the standard reflecting Brownian motion, $L(t)$ is the corresponding local time, the Feynman–Kac functional

$$(6) \quad \hat{e}_c(t) := e^{\frac{1}{2} \int_0^t c(X_t) dL(t)},$$

Λ_i , $i = 1, 2, 3$, denote the Dirichlet, Neumann, and Robin boundary, respectively, and ϕ_i , $i = 1, 2, 3$, are the corresponding boundary values. Moreover, τ_{Λ_1} is the first time a Brownian path originating from the point \mathbf{y} in Ω (interior or exterior region) hits the Dirichlet boundary Λ_1 .

In (5), on the Neumann and Robin boundary, the local time $L(t)$ of the reflecting Brownian motion is used, and the sampling of the RBM and the calculation of the local time have been studied in [33], [31]. In [32], it was demonstrated that (5) can provide an accurate local solution for all three types of BVPs of Laplace equations.

Dirichlet boundary problem. For the Dirichlet boundary problem, equation (5) can be simplified with only killed Brownian motions involved, and the following Feynman–Kac formula gives a probabilistic solution of the Poisson equation over the domain Ω for (2):

$$(7) \quad u(\mathbf{x}) = E^x (\phi(X_{\tau_\Omega})) + E^x \left[\int_0^{\tau_\Omega} f(X_t) dt \right],$$

where the expectation is taken over all sampling paths $X_{t=0}(\omega) = \mathbf{x}$ and τ_Ω is the first boundary hitting time (or exit time) of the domain Ω . In this paper, we consider only the Laplace equations ($f \equiv 0$).

For the Laplace equation, the Ito diffusion of (1) is the Brownian motion. The solution of the Laplace equation can be rewritten with a harmonic measure μ_Ω^x , which measures the probability of the Brownian paths hitting a given boundary area:

$$(8) \quad u(\mathbf{x}) = E^x(\phi(X_{\tau_\Omega})) = \int_{\partial\Omega} \phi(\mathbf{y}) d\mu_\Omega^x,$$

where

$$(9) \quad \mu_\Omega^x(F) = P^x\{\omega | X_{\tau_\Omega}(\omega) \in F, X_0(\omega) = \mathbf{x}\}, \quad F \subset \partial\Omega, \quad \mathbf{x} \in \Omega.$$

The harmonic measure can be shown to be related to the Laplacian Green's function $g(\mathbf{x}, \mathbf{y})$ in the domain Ω with a homogeneous boundary condition, i.e.,

$$(10) \quad \begin{aligned} -\Delta g(\mathbf{x}, \mathbf{y}) &= \delta(\mathbf{x} - \mathbf{y}), \quad \mathbf{x}, \mathbf{y} \in \Omega, \\ g(\mathbf{x}, \mathbf{y})|_{\mathbf{x} \in \partial\Omega} &= 0. \end{aligned}$$

By comparing (8) with the integral representation of the solution of the Laplace equation in terms of the Green's function $g(\mathbf{x}, \mathbf{y})$,

$$(11) \quad u(\mathbf{x}) = - \int_{\partial\Omega} \phi(\mathbf{y}) \frac{\partial g(\mathbf{x}, \mathbf{y})}{\partial \mathbf{n}_y} ds_y,$$

we can see that the hitting probability, now denoted as $p(\mathbf{x}, \mathbf{y}) ds_y = \mu_\Omega^x([\mathbf{y}, \mathbf{y} + ds_y])$, has the following connection to the Green's function of the domain Ω [3]:

$$(12) \quad p(\mathbf{x}, \mathbf{y}) = - \frac{\partial g(\mathbf{x}, \mathbf{y})}{\partial \mathbf{n}_y}.$$

For some simple structures, if the Dirichlet data on the boundary is given, (11) can be used to analyze the electric field directly with traditional boundary element methods (BEMs). However, for complex structures, a simulation method using Brownian paths is a convenient way to find the harmonic measure μ_Ω^x as detailed in section 4.1 and calculate the probabilistic solution $u(\mathbf{x})$ in the domain.

Therefore, if the domain is a ball centered at the \mathbf{x} where a path starts, we have a uniform probability for the path to hit the surface of the ball. This fact is the basis for the design of random walk-on-spheres (WOS), which allows us to describe the Brownian motion and its exit location on $\partial\Omega$ without explicitly finding its trajectory. Specifically, a sequence of walks or jumps over spheres allows the Brownian path to hit the boundary $\partial\Omega$ (for practical purposes, within an absorption ε -shell of $\partial\Omega$ as proposed in [17]). As indicated by (12), the probability of a Brownian path hitting the surface of a sphere is given by the normal derivative of the Green's function for the sphere (with a homogeneous boundary condition). Therefore, if we draw a ball centered at the starting point \mathbf{x} of a Brownian path, the path will hit the surface of the ball with a uniform probability as long as the ball does not intersect with the domain boundary $\partial\Omega$. Thus, we can make a jump for the Brownian particle to a point \mathbf{x}_1 , sampled with a uniform distribution on the surface of the ball. Next, a second ball now centered at \mathbf{x}_1 is drawn again, not intersecting with the domain boundary $\partial\Omega$, and the Brownian particle can make a second jump to \mathbf{x}_2 on the surface of the second ball.

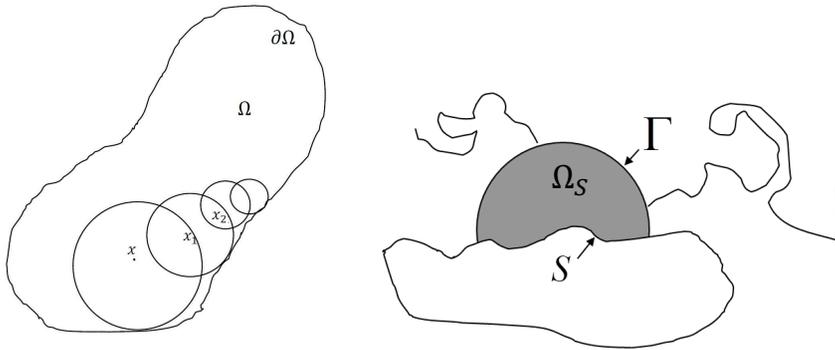


FIG. 1. (Left) WOS sampling of Brownian paths for an interior problem. (Right) BIE-WOS on a patch S for an exterior problem [29].

This procedure (as illustrated in Figure 1 (left), termed walk-on-spheres (WOS)) [8], [20], [18] is repeated until the Brownian particle hits the domain boundary $\partial\Omega$ (within the absorption ε -shell of $\partial\Omega$) [17], and the location of the hitting point is denoted as \mathbf{x}_{τ_Ω} . The value of the boundary data $\phi(\mathbf{x}_{\tau_\Omega})$ is then recorded, and eventually all such data are used to compute the expectation in (8).

Moreover, in applying the Feynman–Kac formula (8) to find the potential in the infinite exterior domain (with a vanishing condition for the potential at the infinity), since some paths go to the infinity, a truncation with a large sphere is used in our simulations of the WOS method such that trajectories outside the large sphere are ignored and considered as ending at the infinity where the potential vanishes. Theoretical estimates on the size of the truncation sphere can be found in [25].

3. An iterative BIE-WOS method with killed Brownian motion for mixed boundary value problems. In this section, we will propose a probabilistic solution for the mixed BVPs of Laplace equations, which only uses killed Brownian motion, therefore avoiding the expensive and delicate computations of reflecting Brownian motions and their local time altogether.

Using the Green’s second identity and both Dirichlet and Neumann data on the boundary, the solution to a Laplace equation in Ω is given by

$$(13) \quad u(\mathbf{x}) = \int_{\partial\Omega} G(\mathbf{x}, \mathbf{y}) \frac{\partial u(\mathbf{y})}{\partial \mathbf{n}_\mathbf{y}} ds_\mathbf{y} - \int_{\partial\Omega} \frac{\partial G(\mathbf{x}, \mathbf{y})}{\partial \mathbf{n}_\mathbf{y}} u(\mathbf{y}) ds_\mathbf{y}, \quad \mathbf{x} \in \Omega,$$

where $\mathbf{n}_\mathbf{y}$ is the outward normal direction of boundary point \mathbf{y} , and $G(\mathbf{x}, \mathbf{y})$ is the fundamental solution of the Laplace operator in the free 3D space

$$(14) \quad G(\mathbf{x}, \mathbf{y}) = \frac{1}{4\pi} \frac{1}{|\mathbf{x} - \mathbf{y}|}.$$

In order to solve (13), traditional BEMs discretize the whole boundary $\partial\Omega$ into small panels, construct a large dense linear system with collocation Nystrom methods, and then solve the resulting linear system for a numerical solution. Because the traditional BEM treats the entire domain boundary together, the dimension of the linear system is determined by the size of the whole domain. Even if a small part of the solution is needed, a large system of equations still needs to be solved.

As shown in Figure 1 (right), the BIE-WOS-iterative method superimposes an imaginary hemisphere upon a relevant portion of the boundary $\partial\Omega$, where a *patch* S is the intersection of $\partial\Omega$ and the hemisphere, the hemispherical surface Γ is the outside part of the hemispherical surface exterior to domain Ω , and the shaded region bounded by $S \cup \Gamma$ is denoted as Ω_S .

Let $G_1(\mathbf{x}, \mathbf{y})$ be the Green's function for a sphere with a homogeneous boundary condition on the whole sphere, which can be obtained with a Kelvin image, i.e.,

$$(15) \quad G_1(\mathbf{x}, \mathbf{y}) = \frac{1}{4\pi} \left[\frac{1}{|\mathbf{x} - \mathbf{y}|} - \frac{a|\mathbf{y}|}{|a^2\mathbf{y} - |\mathbf{y}|^2\mathbf{x}|} \right],$$

where a is the radius of the sphere, the subscript 1 indicates one image charge is used in its definition, i.e., the term $\frac{1}{|\mathbf{x} - \mathbf{y}|}$ attributes to the source point inside the sphere, and the term $\frac{-a|\mathbf{y}|}{|a^2\mathbf{y} - |\mathbf{y}|^2\mathbf{x}|}$ attributes to the image source outside the sphere.

Applying (13) to Ω_S , one can obtain for $\mathbf{x} \in \Omega_S$

$$(16) \quad u(\mathbf{x}) = - \int_{\Gamma} \frac{\partial G_1(\mathbf{x}, \mathbf{y})}{\partial \mathbf{n}_y} u(\mathbf{y}) ds_y + \int_S \left[- \frac{\partial G_1(\mathbf{x}, \mathbf{y})}{\partial \mathbf{n}_y} u(\mathbf{y}) + G_1(\mathbf{x}, \mathbf{y}) \frac{\partial u(\mathbf{y})}{\partial \mathbf{n}_y} \right] ds_y,$$

which can be rewritten as

$$(17) \quad u(\mathbf{x}) = \Sigma_1 + \Sigma_2,$$

where the first part Σ_1 is the integral over Γ and the second part Σ_2 is the one over S . It should be noted that Σ_1 involves only the normal derivative of the Green's function since G_1 vanishes on Γ by construction, and as a result, on Γ only the solution $u(\mathbf{y})$ is needed. On the other hand, both the solution $u(\mathbf{y})$ and its normal derivative $\frac{\partial u(\mathbf{y})}{\partial \mathbf{n}}$ appear in Σ_2 .

The solution $u(\mathbf{y})$ over Γ appearing in Σ_1 can be expressed with the Feynman-Kac formula using the Dirichlet data on $\partial\Omega$. Namely, the Dirichlet data $u(y)$ on the imaginary hemispherical surface Γ in Σ_1 is given by

$$(18) \quad u(\mathbf{y})|_{\Gamma} = \int_{\partial\Omega} p(\mathbf{y}, \mathbf{x}') u(\mathbf{x}') ds_{x'},$$

where the hitting probability $p(\mathbf{y}, \mathbf{x}')$ will be obtained by simulating Brownian motions with the WOS method described in section 4.1.

Therefore, the integral over Γ in (16) can be rewritten as

$$(19) \quad \Sigma_1 = - \int_{\Gamma} \frac{\partial G_1(\mathbf{x}, \mathbf{y})}{\partial \mathbf{n}_y} \left[\int_{\partial\Omega} p(\mathbf{y}, \mathbf{x}') u(\mathbf{x}') ds_{x'} \right] ds_y.$$

The Dirichlet data $u(y)$ on the imaginary hemisphere Γ becomes the data for intermediate calculation of the iterative method, and the formula (16) only involves the boundary information of the original domain Ω . As a result, an iterative method of using BIE and WOS alternately can be established below.

An iterative BIE-WOS algorithm with killed Brownian motion for mixed BVP of Laplace equations. The iterative BIE-WOS method for Laplace equations over the domain boundary consists of the following boundary preparation, Dirichlet data initialization, and two steps for each iteration:

- *Boundary patch preparation.* The boundary $\partial\Omega$ will be covered with M overlapping patches $S_j, j = 1, \dots, M$, obtained by the intersection of half-spheres Ω_{S_j} , i.e., $S_j = \Omega_{S_j} \cap \partial\Omega$.
- *Initialization of Dirichlet data* on $\partial\Omega = \Lambda_1 \cup \Lambda_2 \cup \Lambda_3$, $u^{(0)}|_{\partial\Omega}$. $u^{(0)}|_{\Lambda_1} = \phi_1, u^{(0)}|_{\Lambda_2} = 0, u^{(0)}|_{\Lambda_3} = 0$.
- Given $u^{(n-1)}|_{\partial\Omega}$, for $n < n_{\max}$, or $\max_{\partial\Omega} |u^{(n)} - u^{(n-1)}| > \varepsilon$, with a prescribed number of iterations n_{\max} and an error tolerance ε .

Step 1. At each Gauss point \mathbf{y}_i over each half-spherical surface Γ_j , a WOS solution is obtained for the Laplace equation with a complete Dirichlet data $u^{(n-1)}$ available over the whole boundary $\partial\Omega$, using killed Brownian motion, i.e.,

$$(20) \quad u^{(n)}(\mathbf{y}_i)|_{\Gamma_j} = \int_{\partial\Omega} p(\mathbf{y}_i, \mathbf{x}')u^{(n-1)}(\mathbf{x}')ds_{x'}.$$

Step 2. Over each patch $S_j, j = 1, \dots, M$, the following four kinds of local BIEs will be solved:

Case 1: If S_j is given a Dirichlet datum, i.e., $u|_{S_j} = \phi_1$, solve the following integral equation over S_j for the Neumann data on S_j , denoted as $\frac{\partial u^{(n)}(\mathbf{x})}{\partial \mathbf{n}_{\mathbf{x}}}|_{S_j}$:

$$(21) \quad - \int_{S_j} G_1(\mathbf{x}, \mathbf{y}) \frac{\partial u(\mathbf{y})}{\partial \mathbf{n}_{\mathbf{y}}} ds_{\mathbf{y}} = f_j(\mathbf{x}) - \frac{1}{2} \phi_1(\mathbf{x}) - \int_{S_j} \frac{\partial G_1(\mathbf{x}, \mathbf{y})}{\partial \mathbf{n}_{\mathbf{y}}} \phi_1(\mathbf{y}) ds_{\mathbf{y}}, \mathbf{x} \in S_j,$$

where

$$(22) \quad f_j(\mathbf{x}) = - \int_{\Gamma_j} \frac{\partial G_1(\mathbf{x}, \mathbf{y})}{\partial \mathbf{n}_{\mathbf{y}}} u(\mathbf{y}) ds_{\mathbf{y}},$$

where the integral can be evaluated with a Gaussian quadrature over Γ_j with the value of u at the nodes computed by (20).

Case 2: If S_j is given a Neumann datum, i.e., $\frac{\partial u(\mathbf{x})}{\partial \mathbf{n}_{\mathbf{x}}}|_{S_j} = \phi_2$, solve the following integral equation over S_j for the Dirichlet data on S_j , denoted as $u^{(n)}|_{S_j}$:

$$(23) \quad \frac{1}{2} u(\mathbf{x}) + \int_{S_j} \frac{\partial G_1(\mathbf{x}, \mathbf{y})}{\partial \mathbf{n}_{\mathbf{y}}} u(\mathbf{y}) ds_{\mathbf{y}} = f_j(\mathbf{x}) + \int_{S_j} G_1(\mathbf{x}, \mathbf{y}) \phi_2(\mathbf{y}) ds_{\mathbf{y}}, \mathbf{x} \in S_j.$$

Case 3: If S_j is given a Robin datum, i.e., $\frac{\partial u(\mathbf{x})}{\partial \mathbf{n}_{\mathbf{x}}} - cu(\mathbf{x})|_{S_j} = \phi_3$, solve the following integral equation over S_j for the Dirichlet data on S_j , denoted as $u^{(n)}|_{S_j}$:

$$(24) \quad \begin{aligned} & \frac{1}{2} u(\mathbf{x}) + \int_{S_j} \left[\frac{\partial G_1(\mathbf{x}, \mathbf{y})}{\partial \mathbf{n}_{\mathbf{y}}} - cG_1(\mathbf{x}, \mathbf{y}) \right] u(\mathbf{y}) ds_{\mathbf{y}} \\ & = f_j(\mathbf{x}) + \int_{S_j} G_1(\mathbf{x}, \mathbf{y}) \phi_3(\mathbf{y}) ds_{\mathbf{y}}, \mathbf{x} \in S_j. \end{aligned}$$

Case 4: S_j is given a mixed type of boundary conditions including Dirichlet, Neumann, and Robin type. An integral equation involving the combination of (21), (23), and (24) can be derived. Specific examples are given in the numerical examples in section 6.

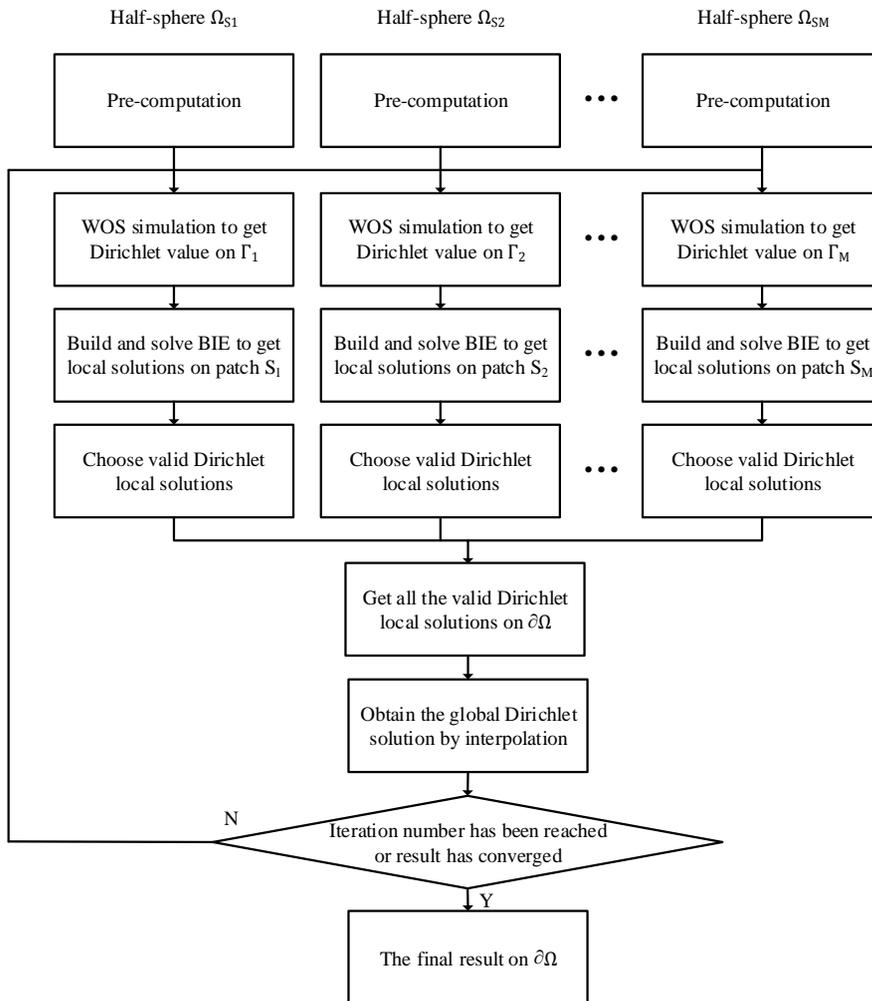


FIG. 2. The implementation flowchart of the iterative BIE-WOS algorithm.

At the end of **Step 2**, the Dirichlet data $u^{(n)}|_{\partial\Omega}$ will have been updated over the whole boundary, from (23) and (24). If $n < n_{\max}$, or $\max_{\partial\Omega} |u^{(n)} - u^{(n-1)}| > \varepsilon$, then return to **Step 1**; otherwise, stop.

4. Algorithm implementation details. A flowchart of the algorithm implementation is shown in Figure 2, and some details will be elaborated in this section.

4.1. Pre-computation of hitting probability $p(y_i, \mathbf{x}')$ for repeated use of Feynman–Kac formula in (20). As the iterative procedure requires the application of the Feynman–Kac formula for a pure Dirichlet problem in (20) for each iteration, it is natural to pre-compute the hitting probability $p(y_i, \mathbf{x}')$ for all starting points y_i —the pre-determined Gauss quadrature points over all the hemispheres.

For a normal pure Dirichlet problem $u(\mathbf{x})|_{\partial\Omega} = \phi(\mathbf{x})$, the Feynman–Kac formula can give an approximate solution of $u(\mathbf{y})$ after the simulation of N_{path} Brownian

particles starting from \mathbf{y} ,

$$(25) \quad u(\mathbf{y}) \simeq \frac{1}{N_{\text{path}}} \sum_{k=1}^{N_{\text{path}}} \phi(\mathbf{e}_k),$$

where \mathbf{e}_k is the location on the domain boundary $\partial\Omega$ where the k th path terminates. However, the updated Dirichlet data on the Neumann or Robin boundary are not given analytically, so we will try to use the Dirichlet data only at the center of each panel. As a result, the formula (18) can be rewritten as a sum of the following terms:

$$(26a) \quad u^{(n)}(\mathbf{y})|_{\Gamma} = \int_{\Lambda_1} p(\mathbf{y}, \mathbf{x}') \phi_1(\mathbf{x}') ds_{x'} + \int_{\Lambda_{2,3}} p(\mathbf{y}, \mathbf{x}') u^{(n-1)}(\mathbf{x}') ds_{x'}$$

$$(26b) \quad \simeq \int_{\Lambda_1} p(\mathbf{y}, \mathbf{x}') \phi_1(\mathbf{x}') ds_{x'} + \sum_{j=1}^{N_{\text{panel(N or R)}}} \int_{S_{\text{panel}_j}} p(\mathbf{y}, \mathbf{x}') u^{(n-1)}(\mathbf{x}') ds_{x'}$$

$$(26c) \quad \simeq \int_{\Lambda_1} p(\mathbf{y}, \mathbf{x}') \phi_1(\mathbf{x}') ds_{x'} + \sum_{j=1}^{N_{\text{panel(N or R)}}} p_{\text{panel}_j} u_{\text{panel}_j}^{(n-1)},$$

where we assume the hitting probability on panel_j will be pre-calculated for any needed starting point \mathbf{y} on a hemisphere Γ ,

$$(27) \quad p_{\text{panel}_j} = \int_{S_{\text{panel}_j}} p(\mathbf{y}, \mathbf{x}') ds_{x'} = p(\mathbf{y}, \mathbf{x}' \in \text{panel}_j),$$

and the potential over panel_j of a reflecting boundary is replaced with the potential at the panel's center $\mathbf{x}_{\text{panel}_j}$, i.e.,

$$(28) \quad u_{\text{panel}_j}^{(n-1)} = u^{(n-1)}(\mathbf{x}_{\text{panel}_j}).$$

In (26c), the sum is divided into two parts; one is the integral over the non-reflecting Dirichlet boundary, and the other is the integral on the reflecting boundary (i.e., Neumann or Robin boundaries). The first part can be directly obtained by the given Dirichlet conditions, while the second part is approximated by the sum of the integral on the panels of the reflecting boundary. With N_{path} simulations of Brownian paths, if N_{panel_j} is the number of paths that fall on the j th panel, then we have

$$(29) \quad p_{\text{panel}_j} = \frac{N_{\text{panel}_j}}{N_{\text{path}}};$$

therefore, (26c) becomes

$$(30) \quad u^{(n)}(\mathbf{y}) \simeq \frac{1}{N_{\text{path}}} \left(\sum_{k=1}^{N_{\text{path D}}} \phi_1(\mathbf{e}_k) + \sum_{j=1}^{N_{\text{panel(N or R)}}} N_{\text{panel}_j} u_{\text{panel}_j}^{(n-1)} \right),$$

where $N_{\text{path D}}$ is the number of the paths absorbed by the Dirichlet boundary, and $\phi_1(\mathbf{e}_k)$ is the given boundary value at the position where the k th path terminates on the Dirichlet boundary. In this formula, $\sum_{k=1}^{N_{\text{path D}}} \phi_1(\mathbf{e}_k)$ and N_{panel_j} can be obtained by the WOS before the iteration, and $u_{\text{panel}_j}^{(n-1)}$ is the variable updated in the iterations.

Pre-computing $p_{panel_j} = p(\mathbf{y}, \mathbf{x}' \in panel_j)$. For each \mathbf{y} , we conduct a WOS simulation with N_{path} paths, starting from \mathbf{y} , until hitting the boundary and being killed there. If the hitting position is on the Dirichlet boundary, the Dirichlet condition of this point is recorded; if it is on the reflecting boundary, the global ID of the panel where the hitting occurred is recorded. When a path does not reach the boundary before being truncated, or goes out of a certain range (common in external problems), no operation is performed. After all N_{path} simulations are carried out, the probability distribution of the WOS paths starting from one point \mathbf{y} arriving at the domain boundary $\partial\Omega$ can then be obtained.

In implementation, each starting point $\mathbf{y} \in \Gamma$ will have its own hash map, which stores the probability of reaching each panel on the reflecting boundary and the sum of the Dirichlet values obtained when reaching the Dirichlet boundary. Since most paths terminate near the starting point, it is sufficient to record the global IDs of the panels hit and the corresponding collision numbers in the form of a hash map. When reusing these maps to complete the WOS calculation in the iterations, we only need to traverse the panel global IDs in the hash map and calculate the weighted sum of the probabilities and the corresponding Dirichlet solution on $\partial\Omega$ produced by the previous iteration. Therefore, the WOS method, which originally took a long time to simulate multiple paths, can be replaced by a simple vector product to obtain the Dirichlet values of the sampling points on the hemispheres in subsequent iterations.

When the domain boundary is exactly the Dirichlet boundary, Dirichlet data of the points on the hemispheres can be obtained, and the local BIEs can be directly solved without iterations. Under this circumstance, no p_{panel_j} needs to be pre-computed and recorded, as described in [29]. However, when a Neumann boundary or Robin boundary is included in a mixed boundary, the Dirichlet data of the points on the hemisphere cannot be easily obtained directly, but can be expressed in the form of a weighted average of the Dirichlet data on the boundary of the domain. Under this circumstance, p_{panel_j} plays an important role as the weights.

In general, using pre-computed WOS with discretized panels may introduce errors and redundant storage. However, based on our numerical experiments in section 6, as long as the discretization is sufficiently fine and regular, the error is negligible and the storage space required is also acceptable.

4.2. Reuse of BIE coefficients and WOS probabilities. As a pre-processing step of the BIE-WOS-iterative algorithm, the reflecting (Neumann or Robin) boundary of the domain is jointly discretized and superimposed by a series of overlapping hemispheres. Local BIEs are then established over the auxiliary hemispheres.

For each hemisphere, once the hemisphere is discretized, the BIE matrix entries containing the interaction through the Green's function between panels can be stored as a feature of the hemisphere. Similarly, in the formula (18) used by the WOS method, given two points \mathbf{y} and \mathbf{x}' , the probability $p(\mathbf{y}, \mathbf{x}')$ is a certain value. It means that for one point \mathbf{y} on the hemisphere, the pre-computed probabilities of WOS reaching the panels on the domain boundary can be stored in the form of a table for later use as mentioned above.

As a result, for the entire algorithm, both the BIE and WOS components of the iterative algorithm incurring the highest computational cost only need to be executed once, and in subsequent iterations, only a small amount of time is required to perform matrix operations with the stored matrices and tables instead. Thus, most of the time is spent on the pre-computations; the numerical result can be improved through more iterations with a negligible increase in overall computation time.

When preparing the data to be reused by the BIEs, it should be noted that since the Green’s function used in the formula (16) is for a spherical domain, it is necessary to convert the coordinates of the panels to the local relative coordinates inside the corresponding hemispheres. The calculated Green’s functions and the derivatives are then saved to form the BIE matrices.

4.3. Updating of solutions during iterations. In each iteration, the Dirichlet data on the boundary of the domain from the previous iteration is used in the Feynman–Kac formula for the Dirichlet problem of the Laplace equations to compute the Dirichlet data on the auxiliary hemispheres as in (26c) with the pre-computed hitting probability (without running WOS any longer), which in turn will be used for computing the new Dirichlet solution on the boundary by solving the local BIEs through matrix operations. In this way, a closed loop is formed so the iterative method can be used to obtain better approximation to the solution. Therefore, the Dirichlet solution on the boundary $\partial\Omega$ is the variable to be updated during the iterations.

For local solutions obtained by BIEs, only the solution within a certain range from each hemisphere center will be considered valid due to the corner edge effect at the hemisphere and domain boundary intersection on the accuracy. Therefore, the area of valid solutions offered by the hemisphere Ω_{S_i} is actually smaller than the patch S_i . Although there are no strict requirements on the size and position of the auxiliary hemispheres, it is necessary to ensure that the entire reflecting boundary is effectively covered so that the solution can be successfully updated.

Moreover, two sets of discretized meshes are adopted in practical applications. A global mesh covering the entire reflecting boundary is used in the WOS simulation, whereas a separate discretized local mesh for the patch S_i under each half-sphere Ω_{S_i} is used when the corresponding BIE system is formed. Once a BIE is solved, the new Dirichlet solution in the valid area will be recorded in a set, which collects all valid local solutions and will be at standby for further processing until all the BIEs during an iteration round are calculated. Then, a local-to-global mapping is established by an inverse distance weighting (IDW) [26], a common spatial interpolation method, given by

$$(31) \quad u_g(\mathbf{x}) = \sum_{i=1}^n w_i u_i, \quad w_i = \frac{d_i^{-\beta}}{\sum_{j=1}^n d_j^{-\beta}};$$

i.e., the interpolation weight is inversely proportional to the power of the distance, and the value of β is usually set as 2. Through this mapping, the Dirichlet solution over the global mesh can be interpolated from the local Dirichlet solutions. With the result of each iteration round, the updated global Dirichlet solution is stored in a list, indexed by the global IDs of the panels for the use of WOS in the next iteration.

In some simple cases, a simplified implementation using only the global mesh can be adopted. The auxiliary hemispheres directly cover the panels of the global mesh, and the corresponding local BIEs are established over the covered panels. If one panel is covered by multiple hemispheres, the average value of the solutions will be taken for updating. It is obvious that the surface formed by the covered panels is slightly different from the actual boundary surface covered by the hemisphere with some jagged irregularities, which may introduce some errors. In the following numerical experiments, Case 2 employs the simplified method, while Case 3 employs the local discretization over each patch.

Downloaded 01/05/23 to 129.119.67.75 . Redistribution subject to SIAM license or copyright; see https://epubs.siam.org/terms-privacy

4.4. Gauss quadratures over Γ . The integral Σ_1 in (17) is computed by a Gauss quadrature. Therefore, the matrix dimension of the local BIE is equal to the number of panels covered by the hemisphere. By using $N_g \times N_g$ Gauss points over the hemispherical surface Γ , we have

$$(32) \quad \Sigma_1 \simeq - \sum_{i,j=1}^{N_g} \omega_i \omega_j \frac{\theta_{max}}{2} \frac{\varphi_{max}}{2} (a^2 \sin \theta_i) \frac{\partial G_1(\mathbf{x}, \mathbf{y}_{i,j})}{\partial \mathbf{n}_{\mathbf{y}_{i,j}}} u(\mathbf{y}_{i,j}) ds_{\mathbf{y}_{i,j}},$$

where

$$(33) \quad \theta_i = \frac{\theta_{max}}{2} (\xi_i + 1), \quad \varphi_j = \frac{\varphi_{max}}{2} (\xi_j + 1), \quad \mathbf{y}_{i,j} = (a, \theta_i, \varphi_j),$$

and ω_i and ξ_i , $1 \leq i \leq N_g$, are the weights and the locations of the Gauss points of the quadrature, respectively. Note that the term $\frac{\theta_{max}}{2} \frac{\varphi_{max}}{2} (a^2 \sin \theta_i)$ in (32) is the area of the surface element in spherical coordinates. $\varphi_{max} = 2\pi$, and θ_{max} is the zenith angle between the normal vector of the hemisphere center and the contact edge with the domain boundary. As an example, when the domain boundary is a plane, $\theta_{max} = \frac{\pi}{2}$ and the term equals $\frac{\pi^2}{4} (a^2 \sin \theta_i)$.

In our experiments, the potential $u(\mathbf{y})$ on Γ is computed by the WOS method for points on a grid generated by evenly discretizing the surface of the hemisphere along with the polar and azimuthal angles. A bilinear interpolation is used for potentials $u(\mathbf{y}_{i,j})$ on Γ used in the quadratures of the formula (32).

5. Comparison with direct BIE and BIE-WOS method with reflecting Brownian motions.

5.1. Comparison with direct BIE. In the iterative process of the BIE-WOS-iterative method, WOS is carried out in the global domain space, while BIE is only used locally to each hemisphere boundary. In fact, WOS also has a certain degree of locality. Most of the locations where WOS paths arrive at the boundary will be near the starting point; very few reach far away. Therefore, the entire BIE-WOS-iterative algorithm has a certain degree of locality, and the dominant contribution comes from boundary data information close to the starting point.

Compared with using the BIE equation directly over the whole boundary of the domain, the proposed BIE-WOS-iterative method can significantly reduce the dimension of the problem. Since the formation of the BIE equation requires calculating the coefficients between every two panels, the complexity of the algorithm is proportional to the square of the number of discrete panels. When the number of panels increases, computing resources, storage, and time consumption will increase tremendously. In contrast, by splitting the huge boundary integral equation into multiple small BIEs over patches under the imaginary hemispheres, the proposed BIE-WOS-iterative method only needs to calculate the coefficients between the panels in a local area, which can effectively eliminate the adverse effects brought about by the quadratic growth of BIE matrix entries. More importantly, these BIEs can be formed and solved in parallel without any communications, using current multicore processors.

5.2. Comparison with BIE-WOS method using reflecting Brownian motions. In the BIE-WOS method using reflecting Brownian motions (RBMs), u is directly obtained by applying the approximate solution (5) under mixed boundary

conditions,

$$(34) \quad u(\mathbf{y}) \simeq \frac{1}{N_{\text{path}}} \sum_{k=1}^{N_{\text{path}}} \left\{ \frac{1}{2} \sum_{j'=0}^{NP} \hat{e}_c(t_j) \phi_{2,3}(X_{t_j}) \delta n_{t_j} \frac{(\Delta x)^2}{3\epsilon} + \hat{e}_c(\tau_{\Lambda_1}) \phi_1(X_{\tau_{\Lambda_1}}) \right\},$$

where $\hat{e}_c(t_j) = e^{\frac{1}{2} \sum_{k=0}^j c(X_{t_k}) \delta n_{t_k} \frac{(\Delta x)^2}{3\epsilon}}$, j' denotes each step of the path, and j denotes the steps where the path hits the reflecting (Robin or Neumann) boundary. More details can be found in [32].

When a domain with mixed boundaries is studied, the Dirichlet data on the reflecting boundary can be solved by using WOS directly starting from the boundary with the formula (34), namely Neumann-to-Dirichlet (NtD) or Robin-to-Dirichlet (RtD). For the Dirichlet problem, our previous work [29] proposed BIE-WOS for Dirichlet-to-Neumann (DtN) mapping, but it can also be extended to the case where the reflecting boundary is involved. The WOS under mixed boundary conditions is used to obtain the Dirichlet values of the points on the hemispheres, and then the local BIEs are performed to obtain the Neumann value on the Dirichlet boundary.

The proposed BIE-WOS-iterative method is more suitable for finding the global solution of the mixed boundary value problem of the Laplace equation in a straightforward parallel manner. After each iteration, the Dirichlet solutions on the entire Neumann and Robin boundary are updated synchronously. More importantly, no RBM is needed in the iterative approach.

The original BIE-WOS method with RBMs, even with no need of iterations, is still much more time-consuming than the BIE-WOS-iterative, as confirmed in the comparative numerical experiments of numerical test Case 2 in section 6. Locating the reflections of RBMs and calculating the local time $L(t)$ in the formula require much more time than finding the absorption by the Dirichlet boundary in the simulation of killed Brownian motions. Moreover, the reflecting boundary makes the path be reflected continuously, thus difficult to truncate, resulting in much longer WOS simulation time. Normally in the simulations, the maximum number of WOS steps for the path N_{step} is smaller than 10^3 for the pure Dirichlet boundary conditions, but reaches 10^5 level with RBMs for the case of mixed boundary for a similar accuracy in the solutions.

In contrast, the proposed BIE-WOS-iterative method applies the WOS method only for Dirichlet boundary conditions with the formula (18), and the path is absorbed once it hits the boundary; even reflecting boundaries are present in the mixed boundary problems.

6. Numerical results.

6.1. Numerical parameters. A list of parameters for the proposed algorithm will be specified first. The size of the grid sampled on each hemisphere for the WOS simulation is denoted by $N_s \times N_s$, and the number of Monte Carlo simulations N_{path} is set for all the grid points on the hemispheres. Results under different N_s and N_{path} settings will be shown below. The longest length of the path N_{step} is set to be 10^3 , which is adequate for WOS under the Dirichlet boundary condition. When the proposed method is applied to external problems, a large sphere with a radius $R_{\text{inf}} = 10^8$ is introduced for WOS. Whenever the Brownian paths move out of this large sphere, they are considered as gone to the infinity [25], and no contribution will be made to the expectation value in the Feynman–Kac formula. The thickness of the absorption shell $\epsilon = 10^{-5}$ is used to determine whether the path has hit the boundary or not.

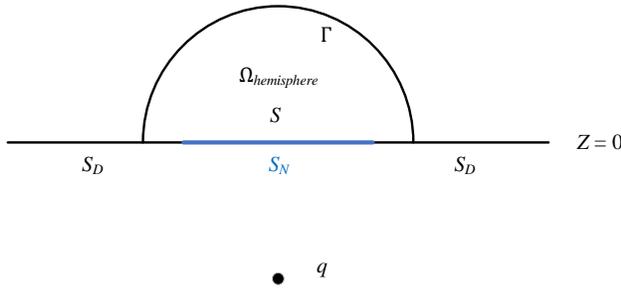


FIG. 3. Case 1: A half-space with a small piece of Neumann boundary.

A constant collocation BEM is used for panels on the patches covered by the hemispheres, and the collocation points are located on the center points of the panels. A Gauss quadrature with 40×40 Gauss points is applied for each integration panel. The Gauss quadrature used to calculate the integral Σ_1 over each hemispherical surface Γ also takes 40×40 points, namely, $N_g = 40$ in the formula (32).

6.2. Computational results. Three examples of different shapes of solution domains with mixed boundary conditions are used to validate the accuracy and efficiency of the proposed iterative BIE-WOS algorithm.

Case 1: A half-space with part of a Neumann boundary. The first case is a half-space with an infinite planar boundary. Except for a small portion of a Neumann boundary, the rest of the half-space boundary is given a Dirichlet condition. Though this case is one of the simplest examples, it is the basis for solving many other complex problems.

As shown in Figure 3, the whole space is divided by the plane $z = 0$ into two dielectric domains. A charge q is located in the lower domain at $r_s = (0, 0, -1)$, and then the analytical solution of the potential in the upper half-space is given by

$$(35) \quad u(\mathbf{r}) = \frac{q}{4\pi\epsilon_0} \frac{1}{|\mathbf{r} - \mathbf{r}_s|}.$$

On the upper surface of the plane, the boundary is given a Neumann condition on a disk of 0.7 from the origin, and outside the disk a Dirichlet condition is given. The goal of the experiment is to solve the Dirichlet value on the Neumann boundary.

A hemisphere with a radius of 1 is formed, and the part of the boundary it intersects is denoted by S , which includes the Neumann boundary part S_N , and the region within the hemisphere is denoted by $\Omega_{hemisphere}$. Thus, the boundary of the half-space S_{plane} is given by

$$(36) \quad S_{plane} = S_N \cup S_D, \quad \partial\Omega_{hemisphere} = \Gamma \cup S, \quad S_N \subset S.$$

$N_{path} = 5 \times 10^4$ WOS paths are performed on each point sampled with the grid $N_s = 40$ on the hemispherical surface Γ and a boundary integral equation is established within the hemisphere. The Neumann boundary is discretized into 6×6 panels in r and θ directions.

The initial values of all Dirichlet variables on the Neumann boundary are set to be 0. In other words, when the first WOS simulation is performed for each point sampled on Γ , the result only takes in the prescribed Dirichlet data sampled by the

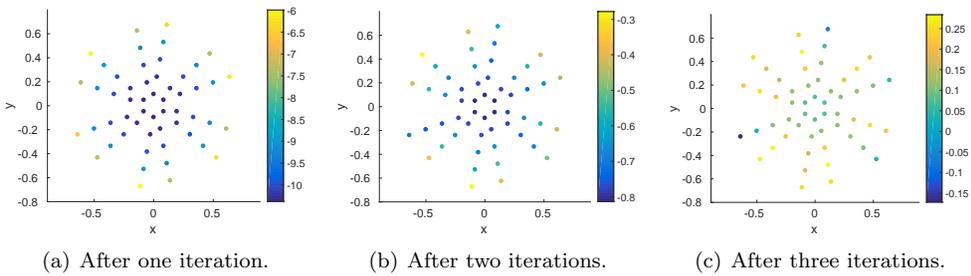


FIG. 4. Case 1: The relative error in percentages versus true value on S_N after three iterations.

paths hitting the Dirichlet boundary. After the BIE is solved, the Neumann panels obtain nonzero Dirichlet data after one iteration, which will affect the result of the WOS in the next iteration. In the subsequent iterative process, the Dirichlet solution on S_N as well as the Dirichlet value on Γ are alternately generated and gradually approach their respective true values.

The relative error of the result for the Dirichlet solution on S_N with respect to the true value after each iteration is displayed in Figure 4. Since most of the plane is a Dirichlet boundary, the iterative result shows a very fast convergence rate. Within only three iterations, the relative error can be reduced to within 0.5%. After the first iteration, the range of relative error on S_N is $-10.37\% \sim -5.99\%$; after the second iteration, $-0.81\% \sim -0.28\%$; after the third iteration, $-0.17\% \sim 0.28\%$.

In this case, the feasibility of pre-computing probabilities is also investigated. Experiments show that replacing the hitting points with the center points of discretized panels in the WOS simulation creates negligible error if the discretization is sufficient and reasonable. Dirichlet values on the hemisphere calculated with the pre-computed probabilities have a maximum relative error (0.38%) similar to that of Dirichlet values (0.47%) on the hemisphere calculated by a direct WOS method.

Case 2: The exterior region of a large sphere with a mixed boundary.

The second case is for the exterior region of a large sphere with a mixed boundary. The schematic diagram of the case is shown in the Figure 5. The entire outer surface of the sphere is separated by the plane $z = 0$; the upper half is the Dirichlet boundary, and the lower half is the Neumann boundary. The radius of the large sphere is 10. A charge q is located in the sphere at $r_s = (0, 0, -5)$, and the analytical solution of the potential outside the sphere is set to be of the same form as the formula (35).

The boundary is arranged to be covered by 6 groups of a total of 64 hemispheres. The first hemisphere is set at the lowest point of the large sphere, and the last group of hemisphere centers is set on the middle dividing line to ensure that the Neumann boundary is completely covered. No hemispheres are needed for the prescribed Dirichlet boundary. The whole Neumann boundary and a part of the Dirichlet boundary (next to the dividing plane $z = 0$) are uniformly discretized in the θ and ϕ directions from the bottom end with $n_\theta = 60$ and $n_\phi = 120$ so that BIEs in all hemispheres can be established. The solution within the distance $R_{valid} = 0.6R_{hemisphere}$ from the center of each hemisphere is accepted as valid. For example, in Figure 5, the solution on the surface within the thick dotted line is an accepted valid solution in the first hemisphere.

As the number of iterations increases, the Dirichlet data on the Neumann boundary changes as shown in Figure 6. In the figure, the results under different N_s settings

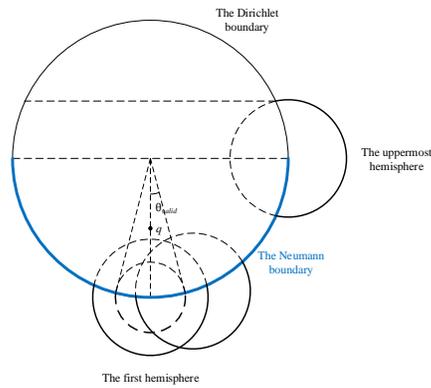


FIG. 5. Case 2: An exterior region of a large sphere with a mixed boundary.

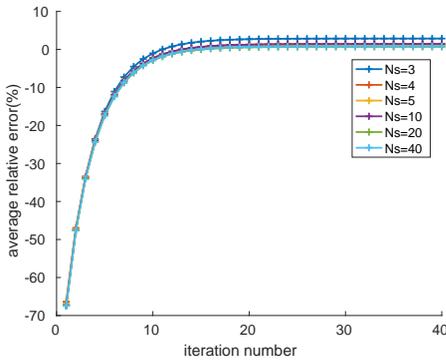
are represented by curves in different colors. In Figure 6(a), we can see that as the number of iterations increases, the solution grows from 0 and finally converges to the true value. Figure 6(b) is a zoomed-in display of the convergence. After about 20 iterations, the rate of solution convergence slows down significantly, and after about 40 iterations, the solution on the Neumann boundary becomes stabilized. Figure 6(c) and Figure 6(d), respectively, show the history of the average and maximum value of the relative error in percentages on all Neumann panels versus the number of iterations (≥ 8). All indicators stabilize after 40 iterations and show good accuracy. The result is relatively better when $N_s = 20, 40$, and the parameter N_{path} used in all experiments in Figure 6 is 5×10^4 .

The contour of the solution on the Neumann boundary at the bottom patch after 40 iterations is shown in Figure 7 with the parameter $N_s = 20, N_{path} = 2 \times 10^5$. Figure 7(a) is the electric potential obtained by the proposed method, and Figure 7(b) is the relative error compared with the theoretical solution. On the Neumann boundary, our solution presents a smooth electric field. Compared with the true solution, the average relative error on all panels is 0.8029%, and the maximal relative error is 1.2842%.

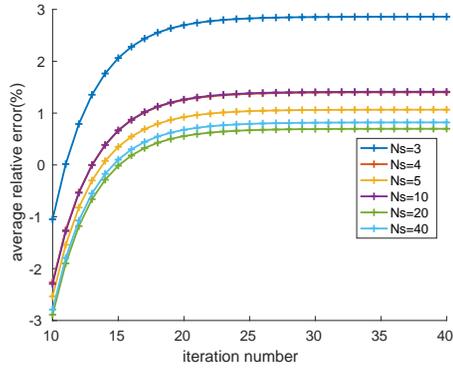
Numerical results under different N_s and N_{path} settings are listed in Table 1. As a contrast, the test was first carried out with theoretical solutions given at the sampling points on the hemispheres. Under this circumstance, the results can be directly obtained by the local BIEs inside the hemispheres without iteration. The relative error caused by local BIEs has an average value 0.1972% and the maximum value 0.7496%, which shows the accuracy of the Feynman–Kac formula based WOS component in the proposed iterative BIE–WOS method.

Effect of sampling points N_s on the hemisphere. In Table 1(a), with N_{path} fixed to be 5×10^4 , test results under different N_s settings are given. In general, larger N_s values yield relatively better results but result in increased WOS runtime and more storage space. According to the experimental results, $N_s = 20$ is sufficient with about 0.8% average error and about 1.5% maximum error. If a fast and rough solution is needed, a smaller value of N_s is also an option. For example, when $N_s = 5$, the maximum relative error is about 2% with only one sixteenth of the WOS workload of the case of $N_s = 20$.

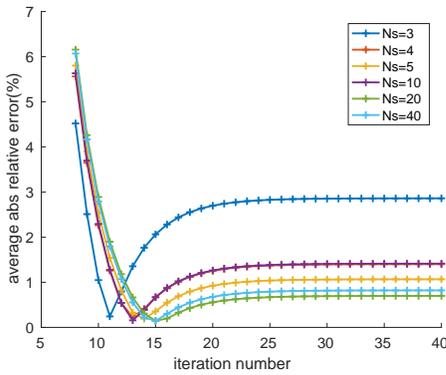
Effect of path numbers N_{path} . In Table 1(b), N_s is fixed to be 20, and



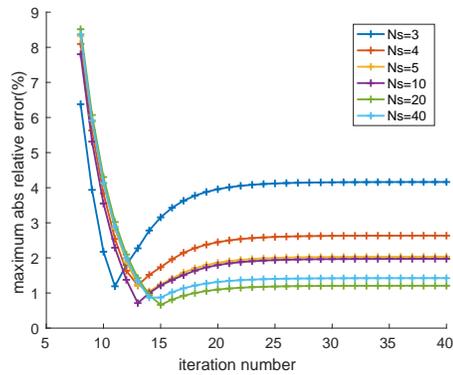
(a) The history of the average relative error in percentages.



(b) (Zoom-in of (a)) The convergence of the average relative error (iteration number ≥ 10).

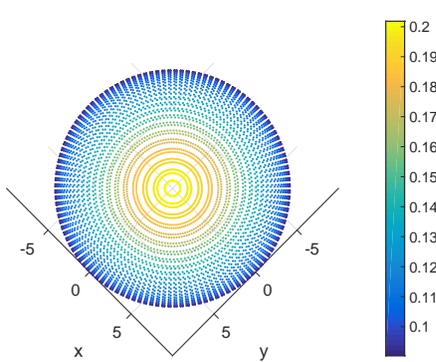


(c) The history of the average relative error in percentages (iteration number ≥ 8).

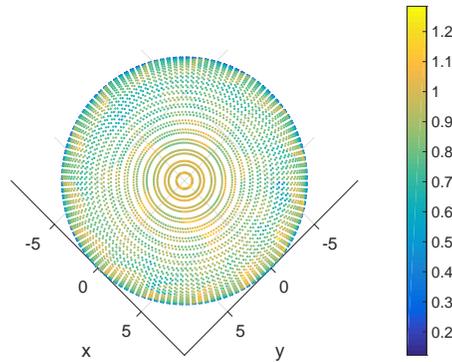


(d) The history of the maximum relative error in percentages (iteration number ≥ 8).

FIG. 6. Case 2: The relative error in percentages on the Neumann boundary versus the number of iterations.



(a) Solution on the Neumann boundary.



(b) Relative error compared with true value.

FIG. 7. Case 2: The contour of the solution on the bottom Neumann boundary ($N_s = 20$, $N_{path} = 2 \times 10^5$).

TABLE 1
Numerical results with different N_s and N_{path} values.

(a) Different N_s				(b) Different N_{path}			
N_{path}	N_s	Avg err%	Max err%	N_s	N_{path}	Avg err%	Max err%
5e4	BIE	0.1972	0.7496	20	1e3	1.5740	3.8496
	3	2.7900	4.0980			1.4648	3.0152
		3.0238	4.3088		1e4	1.8828	2.7555
	4	1.4016	2.6348			1.7415	2.5520
		1.8681	3.1353		5e4	0.6979	1.2067
	5	1.0661	2.0326			0.5482	1.2648
		0.9337	1.8124			0.7552	1.2597
	10	1.4689	2.2153			1.2072	1.7945
		0.7400	1.4215		1.1218	1.7585	
	20	0.6979	1.2067		2e5	0.8029	1.2842
		0.5482	1.2648			0.8799	1.4071
	40	0.8204	1.4235			0.9641	1.4441
		0.9672	1.5310			0.8455	1.2999
						0.9109	1.3990

different N_{path} settings are tested. When the path number of Monte Carlo simulation is too small, the results become rougher. In such a case, some panels with lower hitting probabilities may not even be sampled by the sampling paths and thus not recorded in the hash map of the starting point. When $N_{path} = 5 \times 10^4$, the result is relatively accurate, and the maximum error is small, but there is a certain variance among the results. When $N_{path} = 2 \times 10^5$, the uncertainty of the iterative results is smaller, though it also results in four times the amount of computation time.

Effect of parallelization. The BIE-WOS iterative method is ideal for parallel computing architecture, as validated by our numerical experiments. To achieve faster CPU time, the algorithm can be fully parallelized for all the WOS paths as much as possible. The program runs on 8 Intel(R) Xeon(R) CPU E7-8850, with 10 cores in each CPU, using OpenMP [21] for parallelism. When $N_s = 20$ and $N_{path} = 5 \times 10^4$, on average for each hemisphere, runtime of each part consists of the following tasks: constructing coefficient matrices for BIE: 4.6s; WOS for the sampled points: 66.5s; each subsequent iteration: 0.06s. We can conclude that once the BIE matrices and the hitting probabilities used by WOS are prepared, subsequent iterations take very little time such that the iteration time accounts for less than one-tenth of the total computation time for dozens of iterations. It should be mentioned that storing the hitting probabilities of the sampling points consumes about 16MB of storage space for each hemisphere on average.

As a comparison, the BIE-WOS method with RBMs has also been tested in this case. It takes up to about 8 hours to calculate the solution for a single hemisphere under the same experimental conditions with a similar relative error (mean 1.2120%; max 1.5911%). This reflects the huge time savings of the proposed iterative approach with killed BMs.

Case 3: The exterior or interior region of a large body of revolution. The third case is a large body of revolution formed by rotating the function $f(x) = (x/2)(x/2 - 3)(x/2 - 1.50 - 0.8i)(x/2 - 1.5 + 0.8i)$, $x \in [0, 6]$, along the x -axis. The function $f(x)$ and the constructed rotating body are shown in Figure 8. For the exterior (interior) problem, four charges are located inside (outside) the revolution body asymmetrically and generate the electric fields outside (inside) the body.

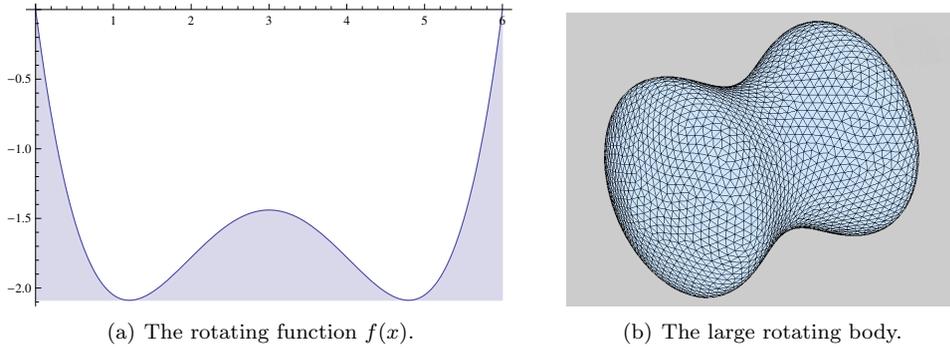


FIG. 8. Case 3: An exterior region of a large body of revolution [30].

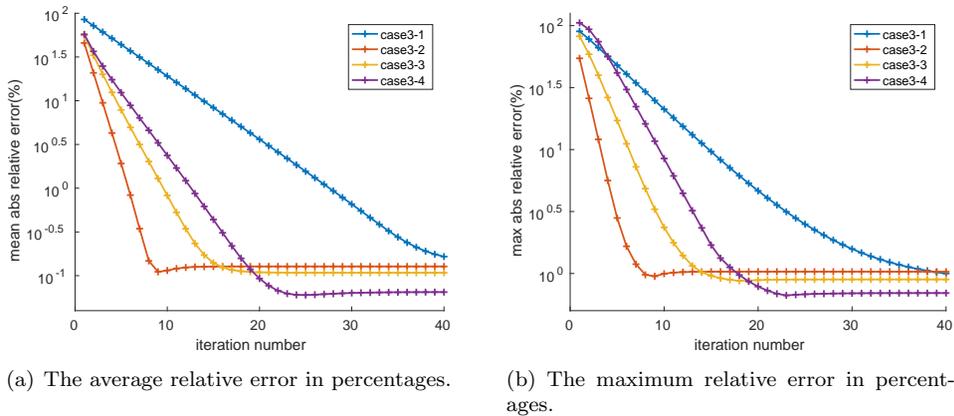


FIG. 9. Case 3: The relative error in percentages of solution versus the number of iterations.

Because the intersections of the hemispheres and the body surface are irregular curves, this case is close to more realistic application situations and more complex than the previous two cases. In order to achieve higher accuracy of the local BIEs, the finer locally discretized covering patches are employed. The entire body surface is covered by 478 local hemispheres with a radius $R = 1$. Both global and local meshes are constructed by Distmesh [23]. The global grid has 23,924 panels. The valid ratio inside each hemisphere is set as 0.5, and the total of the valid local panels is 50,382. The same meshes are used in the exterior and interior problems with opposite outer normals.

To test the performance of our proposed method under normal circumstances, the surface of the body of revolution is given several different types of boundary conditions. The convergence of the iterative solutions and the final results are shown in Figures 9 and 10.

Case 3-1: A pure Neumann boundary (exterior region). The entire surface of the rotating body is given Neumann boundary conditions. It should be noted that a unique solution exists in this case because the electric potential at infinity is set to be 0 for the exterior problem. Therefore, even with no Dirichlet data given on the boundary, the result still converges to the analytical solution after only dozens of iterations.

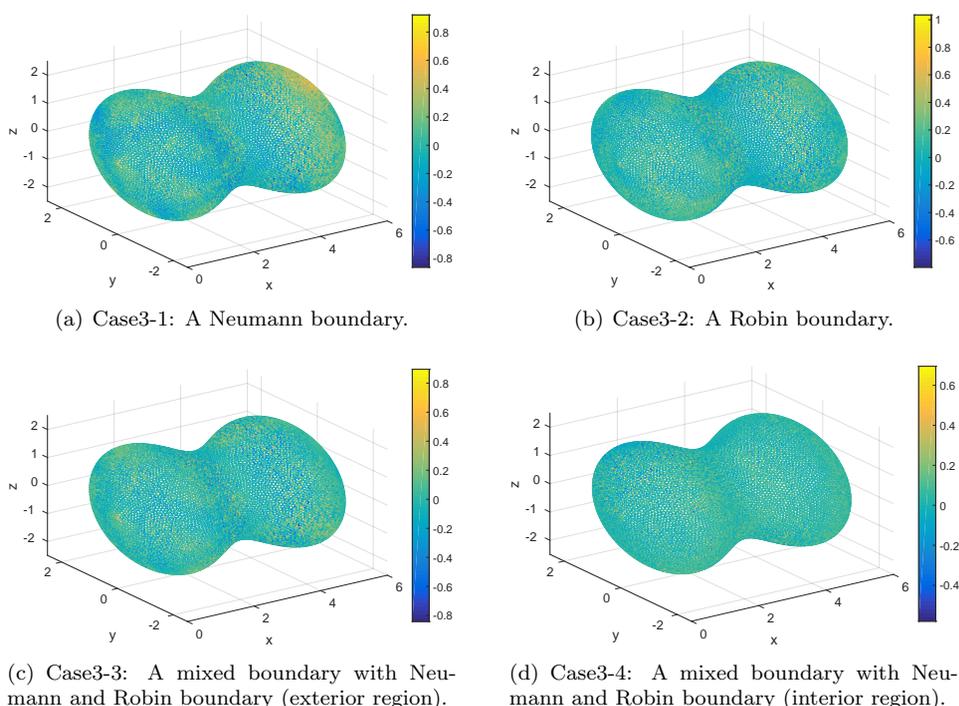


FIG. 10. Case 3: The relative error of the converged solutions.

Similar to the previous examples, the Dirichlet data starts from 0 and eventually converges to the analytical solution as the number of iterations increases. Compared with mixed boundary conditions, the rate of convergence is relatively slower for the pure Neumann boundary conditions. After about 40 iterations, the maximum relative error falls to within 1%, and after about 80 iterations, the iterative solution stabilizes. The relative error between the final result and the analytical solution has an average relative error 0.1495% and a maximum relative error 0.9207%.

Case 3-2: A pure Robin boundary (exterior region). The entire surface of the rotating body is given Robin boundary conditions. The rate of convergence is relatively faster than Case 3-1. After about 20 iterations, the iterative solution becomes stabilized. The average relative error is 0.1275%, and the maximum relative error is 1.0347%.

Case 3-3: A mixed boundary with Neumann and Robin boundaries (exterior region). The surface of the rotating body is given a mixed boundary with Neumann and Robin boundary conditions. Divided along the x -axis, the boundary is given a Robin condition for $x \in [0, 2]$ or $(4, 6]$ and a Neumann condition for $x \in (2, 4]$.

The maximum relative error falls to within 1% after about 15 iterations, and the iterative solution becomes stabilized after about 30 iterations. The final relative error has an average value of 0.1081% and a maximum value of 0.8958%.

Case 3-4: A mixed boundary with Neumann and Robin boundaries (interior region). In this case, we solve the interior potential produced by 4 charges outside the domain and the Laplace equation is given the same boundary type as Case 3-3. Again, the algorithm performs well, and the iterative solution becomes stabilized after about 40 iterations. The converged relative error has an average value

of 0.0650% and a maximum value of 0.6952%.

In terms of CPU time, for Cases 3-1–3-4, when $N_s = 20$ and $N_{path} = 5 \times 10^4$, on average for each hemisphere, the runtime of different components of the method consists of constructing BIE matrices: 2.5s; WOS for the sampled points: 400–480s; each subsequent iteration: 0.04s. In terms of storage, it takes about 27MB for storing the hitting probabilities of the sampling points on each hemisphere.

7. Conclusions and future work. In this paper, a parallel iterative probabilistic method is presented for finding the global solution of Laplace equations with mixed boundary conditions with the Feynman–Kac formula for killed Brownian motions, thus removing the need of expensive computation of local time of reflecting Brownian motions. The proposed method combines a deterministic (local) boundary integral equation (BIE) method and the probabilistic Feynman–Kac formula with killed Brownian motions for mixed BVP solutions of elliptic PDEs in an iterative manner. The algorithm decomposes a large domain boundary computation problem into many small local integral equation problems. The method is shown to perform well under various boundary conditions, especially for large-scale complex problems with mixed boundary conditions. Moreover, this method is well suited for distributed computing with multicore processors due to the high degree of parallelism. At the same time, the scale of the problem can be increased for all boundary conditions with large-scale parallel computers by not being constrained by the size of the solution domain.

Future work on the BIE-WOS iterative method includes nonlinear boundary conditions and efficient WOS algorithms for the Poisson equation. The inhomogeneity from the Poisson equation will pose several challenges in extending the BIE-WOS strategy. First, the local BIE will become a local volume integral equation (VIE), which requires the application of Feynman–Kac formula for the solution at locations inside the hemisphere. Second, the path information of the Brownian motion before reaching each sphere in the WOS process will be required for a path integral of the inhomogeneity in the Feynman–Kac formula. The second issue was addressed in the WOS method for the fractional Poisson equation [13], [14], for obtaining the solution at a single location, as the occupation measure inside a ball of the corresponding stable processes is available [1]. Therefore, making an efficient new VIE-WOS method for the whole solution of the Poisson equation will be a big challenge, requiring further extensive investigations. However, alternatively, for compact supported inhomogeneity of the Poisson equation as in many applications, we could simply remove the inhomogeneity of the Poisson equation by subtracting a special solution in a periodic box containing the inhomogeneity support, thus reducing the Poisson equation to a Laplace equation with a modified boundary condition. The special solution can be efficiently obtained using fast Fourier transforms due to the diagonal form of the Laplace operator in the Fourier space.

Acknowledgment. The authors thank the anonymous reviewers for many constructive comments, which have improved the paper.

REFERENCES

- [1] R. M. BLUMENTHAL, R. K. GETTOOR, AND D. B. RAY, *On the distribution of first hits for the symmetric stable processes*, Trans. Amer. Math. Soc., 99 (1961), pp. 540–554.
- [2] M. BOSSY, E. GOBET, AND D. TALAY, *A symmetrized Euler scheme for an efficient approximation of reflected diffusions*, J. Appl. Probab., 41 (2004), pp. 877–889.
- [3] K. L. CHUNG, *Green, Brown and Probability*, World Scientific, Singapore, 1995.

- [4] C. COSTANTINI, B. PACCHIAROTTI, AND F. SARTORETTO, *Numerical approximation for functionals of reflecting diffusion processes*, SIAM J. Appl. Math., 58 (1998), pp. 73–102, <https://doi.org/10.1137/S0036139995291040>.
- [5] M. I. FREIDLIN, *Functional Integration and Partial Differential Equations*, Princeton University Press, Princeton, NJ, 1985.
- [6] A. FRIEDMAN, *Stochastic Differential Equations and Applications*, Dover, New York, 2006.
- [7] L. V. GIBIANSKY AND S. TORQUATO, *Link between the conductivity and elastic moduli of composite materials*, Phys. Rev. Lett., 71 (1993), 2927.
- [8] J. A. GIVEN, C. O. HWANG, AND M. MASCAGNI, *First- and last-passage Monte Carlo algorithms for the charge density distribution on a conducting surface*, Phys. Rev. E, 66 (2002), 056704.
- [9] L. GREENGARD AND V. ROKHLIN, *A fast algorithm for particle simulations*, J. Comput. Phys., 73 (1987), pp. 325–48.
- [10] C. O. HWANG AND J. A. GIVEN, *Last-passage Monte Carlo algorithm for mutual capacitance*, Phys. Rev. E, 74 (2006), 027701.
- [11] G. J. JIANG AND J. L. KNIGHT, *A nonparametric approach to the estimation of diffusion processes, with an application to a short-term interest rate model*, Econometric Theory, 13 (1997), pp. 615–645.
- [12] D. JUBA, D. J. AUDUS, M. MASCAGNI, J. F. DOUGLAS, AND W. KEYROUZ, *ZENO: Software for calculating hydrodynamic, electrical, and shape properties of polymer and particle suspensions*, J. Res. Natl. Inst. Stand. Technol., 122 (2017), 20.
- [13] A. E. KYPRIANOU, O. ANA, AND S. TONY, *Unbiased ‘walk-on-spheres’ Monte Carlo methods for the fractional Laplacian*, IMA J. Numer. Anal., 38 (2018), pp. 1550–1578.
- [14] A. LISCHKE, G. PANG, M. GULIAN, F. SONG, C. GLUSA, X. ZHENG, Z. MAO, W. CAI, M. M. MEERSCHAERT, M. AINSWORTH, AND G. E. KARNIADAKIS, *What is the fractional Laplacian? A comparative review with new results*, J. Comput. Phys., 404 (2020), 109009.
- [15] S. MAIRE AND M. SIMON, *A partially reflecting random walk on spheres algorithm for electrical impedance tomography*, J. Comput. Phys., 303 (2015), pp. 413–30.
- [16] S. MAIRE AND E. TANRÉ, *Monte Carlo approximations of the Neumann problem*, Monte Carlo Methods Appl., 19 (2013), pp. 201–236.
- [17] M. MASCAGNI AND C. O. HWANG, *ϵ -shell error analysis for “walk on spheres” algorithms*, Math. Comput. Simul., 63 (2003), pp. 93–104.
- [18] M. MASCAGNI AND N. A. SIMONOV, *The random walk on the boundary method for calculating capacitance*, J. Comput. Phys., 195 (2004), pp. 465–473.
- [19] J. P. MORILLON, *Numerical solutions of linear mixed boundary value problems using stochastic representations*, Internat. J. Numer. Methods Engrg., 40 (1997), pp. 387–405.
- [20] M. E. MULLER, *Some continuous Monte Carlo methods for the Dirichlet problem*, Ann. Math. Statist., 27 (1956), pp. 569–589.
- [21] OPENMP, *The OpenMP API specification for parallel programming*, <https://www.openmp.org/>.
- [22] V. G. PAPANICOLAOU, *The probabilistic solution of the third boundary value problem for second order elliptic equations*, Probab. Theory Related Fields, 87 (1990), pp. 27–77.
- [23] P.-O. PERSSON AND G. STRANG, *A simple mesh generator in MATLAB*, SIAM Rev., 46 (2004), pp. 329–345, <https://doi.org/10.1137/S0036144503429121>.
- [24] K. K. SABELFELD, *Monte Carlo Methods in Boundary Value Problems*, Springer Ser. Comput. Phys., Springer, Berlin, 1991.
- [25] K. K. SABELFELD, I. A. SHALIMOVA, AND A. LAVRENTJEVA, *Random walk on spheres process for exterior Dirichlet problem*, Monte Carlo Methods Appl., 1 (1995), pp. 325–331.
- [26] D. SHEPARD, *A two-dimensional interpolation function for irregularly-spaced data*, in Proceedings of the 1968 23rd ACM National Conference, ACM, New York, 1968, pp. 517–524.
- [27] G. I. TAYLOR, *The energy of a body moving in an infinite fluid, with an application to airships*, Proc. Roy. Soc. London Ser. A, 120 (1928), pp. 13–21.
- [28] F. VARGAS-LARA, A. M. HASSAN, E. J. GARBOCZI, AND J. F. DOUGLAS, *Intrinsic conductivity of carbon nanotubes and graphene sheets having a realistic geometry*, J. Chem. Phys., 143 (2015), 204902.
- [29] C. YAN, W. CAI, AND X. ZENG, *A parallel method for solving Laplace equations with Dirichlet data using local boundary integral equations and random walks*, SIAM J. Sci. Comput., 35 (2013), pp. B868–B889, <https://doi.org/10.1137/120875004>.
- [30] C. YAN, W. CAI, AND X. ZENG, *A highly scalable boundary integral equation and walk-on-spheres (BIE-WOS) method for the Laplace equation with Dirichlet data*, Commun. Comput. Phys., 29 (2021), pp. 1446–1468.
- [31] Y. ZHOU AND W. CAI, *Numerical solution of the Robin problem of Laplace equations with a*

- Feynman-Kac formula and reflecting Brownian motions*, J. Sci. Comput., 69 (2016), pp. 107–121.
- [32] Y. ZHOU AND W. CAI, *A Path Integral Monte Carlo (PIMC) Method Based on Feynman-Kac Formula for Electrical Impedance Tomography*, preprint, <https://arxiv.org/abs/1907.13147>, 2019.
- [33] Y. ZHOU, W. CAI, AND (E.) P. HSU, *Computation of local time of reflecting Brownian motion and probabilistic representation of the Neumann problem*, Commun. Math. Sci., 15 (2017), pp. 237–259.