A Fast Accurate Boundary Integral Method for Potentials on Closely Packed Cells

Wenjun Ying[1], J. Thomas Beale[2*]

[1] *Shanghai Jiao Tong University, Department of Mathematics and Institute of Natural Sciences, Minhang, Shanghai 200240, P. R. China*
[2] *Duke University, Department of Mathematics, Durham, NC 27708-0320, USA*

Abstract. Boundary integral methods are naturally suited for the computation of harmonic functions on a region having inclusions or cells with different material properties. However, accuracy deteriorates when the cell boundaries are close to each other. We present a boundary integral method which is specially designed to maintain second order accuracy even if boundaries are arbitrarily close. The method uses a regularization of the integral kernel which admits analytically determined corrections to maintain accuracy. For boundaries with many components we use the fast multipole method for efficient summation. We compute electric potentials on a domain with cells whose conductivity differs from that of the surrounding medium. We first solve an integral equation for a source term on the cell interfaces and then find values of the potential near the interfaces via integrals. Finally we use a Poisson solver to extend the potential to a regular grid covering the entire region. A number of examples are presented. We demonstrate that increased refinement is not needed to maintain accuracy as interfaces become very close.

AMS subject classifications: 35J05, 65N06, 65N38

Key words: Laplace equation, boundary integral method, fast multipole method

1 Introduction

A wide range of biological problems lead to models involving a potential function in tissue with a number of closely packed cells. Recent applications include gene transfection [12, 13], electrochemotherapy of tumors [15] and cardiac defibrillation [1]. Our interest in the problem is mainly motivated by studies of the electrical response of biological cells under field stimulation [17], which can be described by harmonic potential functions on a domain consisting of many cells and an extracellular region with different conductivities.

*Corresponding author. *Email addresses:* wying@sjtu.edu.cn (W.-J. Ying), beale@math.duke.edu (J. T. Beale)

Since the potential is a harmonic function inside each cell and in the exterior region, with prescribed boundary conditions at the cell interfaces, it is natural to write the potential as a sum of single and double layer potentials on the cell boundaries Γ_k, $k=1,\ldots,K$, and evaluate the integrals directly. In principle this is routine if the point of evaluation $\mathbf{x}$ is away from Γ_k. It is also not difficult in this two-dimensional setting if $\mathbf{x}\in\Gamma_k$. However, if, for example, Γ_1 and Γ_2 are close and $\mathbf{x}\in\Gamma_1$, then the integral on Γ_2 is nearly singular, so that a standard quadrature rule becomes inaccurate when the distance is small. It is therefore desirable to use a method of quadrature which is accurate, uniformly with respect to the point of evaluation, without requiring a large amount of extra work. A method with these features was developed in [5] and is used here. Briefly, the singularity in the integral kernel is regularized on a scale comparable to the grid size, and a standard quadrature is used for the regularized integral. Analytical corrections are then added for the errors due to regularization and discretization. It is not necessary to use special quadrature points depending on the point of evaluation; the method is almost as efficient as for a smooth integrand. A similar method for layer potentials on surfaces was developed in [4].

To solve for the potential, we first solve an integral equation for a source term on the cell boundaries. We then compute the potential at grid points covering the region of interest. To compute the integrals directly would require a large computational cost, especially if the number of cells is large. For this reason we use a version of the fast multipole method for the interaction of points which are well separated; the effect of the regularization can be neglected in this case. The nearby interactions are summed directly, using the regularized kernel. After solving the integral equation, we evaluate the potential at grid points near the cell boundaries. These values are again given by nearly singular integrals, which are computed in a similar manner using the multipole method. From these we can obtain the values at all grid points using a fast Poisson solver, using a method introduced in [11] and applied in [5]. The operation count for the full method is roughly proportional to the total number of grid points in the region and on the cell boundaries.

The difficulty in computing nearly singular integrals has long been recognized, e.g., [2], Sec. 7.2.1. For boundary element methods many techniques have been introduced for computing needed integrals on triangles, usually using coordinate transformations. A recent, comprehensive mathematical treatment of boundary element methods is given in [14]. The method [16] for computing layer potentials on surfaces, using coordinate charts rather than boundary elements, included a special treatment of the nearly singular case.

The problem is formulated in Section 2. In Section 3 the integral expression for the solution is derived in terms of a source term q on the boundary, and an integral equation for q is stated. In Section 4 formulas are given for the evaluation of the nearly singular integrals. The fast multipole method is described in Section 5. Extensive numerical results are presented in Section 6 and briefly discussed in Section 7.The integral equation is derived in the Appendix.

2 Interface Problem

Let $\Omega_i \subset \mathbb{R}^2$ be a bounded open set with smooth boundary, which may have multiple disconnected components. Let $\Omega_e = \mathbb{R}^2 \setminus \bar{\Omega}_i$ be the unbounded, complementary domain. Let Γ be the interface, the common boundary of Ω_i and Ω_e. When the interface Γ has multiple components, we write $\Gamma = \bigcup_{k=1}^{K} \Gamma_k$, and assume each component Γ_k is a simple closed curve.

Let $\mathbf{x} = (x_1, x_2)^{\mathrm{T}} \in \mathbb{R}^2$ be a point in space. Let $\Phi_i(\mathbf{x})$ and $\Phi_e(\mathbf{x})$ be two unknown potential functions on Ω_i and Ω_e, respectively. Assume the conductivities σ_i and σ_e on Ω_i and Ω_e are constant but distinct. The potential functions satisfy the Laplace equation

$$\triangle \Phi_i(\mathbf{x}) = 0 \qquad \text{in } \Omega_i$$

and

$$\triangle \Phi_e(\mathbf{x}) = 0 \qquad \text{in } \Omega_e.$$

Let

$$\Phi(\mathbf{x}) = \begin{cases} \Phi_i(\mathbf{x}) & \mathbf{x} \in \Omega_i \\ \Phi_e(\mathbf{x}) & \mathbf{x} \in \Omega_e \end{cases}.$$

In general, the function $\Phi(\mathbf{x})$ is discontinuous across the interface Γ. Let

$$\Phi_i(\mathbf{x}) - \Phi_e(\mathbf{x}) = V_m(\mathbf{x}) \qquad \text{on } \Gamma, \tag{2.1}$$

where $V_m(\mathbf{x})$ will be known. Assume that

$$\sigma_i \frac{\partial \Phi_i(\mathbf{x})}{\partial \mathbf{n}_\mathbf{x}} - \sigma_e \frac{\partial \Phi_e(\mathbf{x})}{\partial \mathbf{n}_\mathbf{x}} = 0 \qquad \text{on } \Gamma, \tag{2.2}$$

with $\mathbf{n}_\mathbf{x}$ being the unit normal pointing from the bounded domain Ω_i to the unbounded domain Ω_e. Let $\mathbf{E} = (E_1, E_2)^{\mathrm{T}} \in \mathbb{R}^2$ be a given vector. We assume the potential function $\Phi_e(\mathbf{x})$ satisfies the far field condition

$$\Phi_e(\mathbf{x}) \to -\mathbf{E} \cdot \mathbf{x} \qquad \text{as } |\mathbf{x}| \to \infty.$$

3 Boundary Integral Equation

We will express Φ in terms of double and single layer potentials of the form

$$u(\mathbf{x}) = \int_\Gamma \frac{\partial G(\mathbf{y} - \mathbf{x})}{\partial \mathbf{n}_\mathbf{y}} f(\mathbf{y}) \, ds_\mathbf{y}, \qquad v(\mathbf{x}) = \int_\Gamma G(\mathbf{x} - \mathbf{y}) q(\mathbf{y}) \, ds_\mathbf{y} \tag{3.1}$$

with some density functions f and q. Here $G(\mathbf{x}) = (2\pi)^{-1} \log |\mathbf{x}|$ is the fundamental solution of the Laplace equation in $\mathbb{R}^2$ and $s_\mathbf{y}$ is the arc length parameter of the interface Γ.

We recall that u has a jump discontinuity at Γ,

$$\begin{cases} u_i(\mathbf{x}) &= \frac{1}{2}f(\mathbf{x}) + \int_\Gamma \frac{\partial G(\mathbf{y}-\mathbf{x})}{\partial \mathbf{n}_\mathbf{y}} f(\mathbf{y})\, ds_\mathbf{y} \\ u_e(\mathbf{x}) &= -\frac{1}{2}f(\mathbf{x}) + \int_\Gamma \frac{\partial G(\mathbf{y}-\mathbf{x})}{\partial \mathbf{n}_\mathbf{y}} f(\mathbf{y})\, ds_\mathbf{y} \end{cases}, \tag{3.2}$$

while $\partial u/\partial \mathbf{n}$ is continuous across Γ. Also v is continuous at Γ but $\partial v/\partial \mathbf{n}$ has a jump,

$$\begin{cases} \frac{\partial v_i(\mathbf{x})}{\partial \mathbf{n}_\mathbf{x}} &= -\frac{1}{2}q(\mathbf{x}) + \int_\Gamma \frac{\partial G(\mathbf{x}-\mathbf{y})}{\partial \mathbf{n}_\mathbf{x}} q(\mathbf{y})\, ds_\mathbf{y} \\ \frac{\partial v_e(\mathbf{x})}{\partial \mathbf{n}_\mathbf{x}} &= \frac{1}{2}q(\mathbf{x}) + \int_\Gamma \frac{\partial G(\mathbf{x}-\mathbf{y})}{\partial \mathbf{n}_\mathbf{x}} q(\mathbf{y})\, ds_\mathbf{y} \end{cases}. \tag{3.3}$$

Now, assuming the solution Φ of the interface problem above exists, let

$$q(\mathbf{x}) = \frac{\partial \Phi_i(\mathbf{x})}{\partial \mathbf{n}_\mathbf{x}} - \frac{\partial \Phi_e(\mathbf{x})}{\partial \mathbf{n}_\mathbf{x}} \qquad \text{on } \Gamma. \tag{3.4}$$

Then the potential function $\Phi(\mathbf{x})$ can be represented as

$$\Phi(\mathbf{x}) = \int_\Gamma \frac{\partial G(\mathbf{y}-\mathbf{x})}{\partial \mathbf{n}_\mathbf{y}} V_m(\mathbf{y})\, ds_\mathbf{y} - \int_\Gamma G(\mathbf{x}-\mathbf{y}) q(\mathbf{y})\, ds_\mathbf{y} - \mathbf{E}\cdot\mathbf{x}. \tag{3.5}$$

According to the properties above, this expression for Φ will have the jumps prescribed in (2.1) and (3.4). The unknown density $q(\mathbf{x})$ in (3.5) is determined by the interface condition (2.2).

Let $\mathbf{t}_\mathbf{x} = (x_1'(s), x_2'(s))^T$ be the unit tangent along the interface, so that $\mathbf{n}_\mathbf{x} = (x_2'(s), -x_1'(s))^T$. From the continuity properties of the single and double layer potentials and the interface condition (2.2), we get the boundary integral equation

$$\frac{1}{2}q(\mathbf{x}) + \mu \int_\Gamma \frac{\partial G(\mathbf{x}-\mathbf{y})}{\partial \mathbf{n}_\mathbf{x}} q(\mathbf{y})\, ds_\mathbf{y} = \mu \int_\Gamma \frac{\partial G(\mathbf{x}-\mathbf{y})}{\partial \mathbf{t}_\mathbf{x}} \frac{\partial V_m(\mathbf{y})}{\partial \mathbf{t}_\mathbf{y}} ds_\mathbf{y} - \mu\, \mathbf{E}\cdot\mathbf{n}_\mathbf{x} \tag{3.6}$$

with $\mu = (\sigma_e - \sigma_i)/(\sigma_e + \sigma_i) \in (-1,1)$. Here the second integral is meant in the principal value sense. A brief derivation of this equation is given in the Appendix. The integral equation (3.6) can be re-written concisely as

$$\frac{1}{2}q + \mu \mathcal{K} q = \mu g \qquad \text{on } \Gamma, \tag{3.7}$$

with

$$g(\mathbf{x}) = \mathcal{L} V_m - \mathbf{E}\cdot\mathbf{n}_\mathbf{x} \qquad \text{on } \Gamma,$$

where $\mathcal{K}$ and $\mathcal{L}$ are the integral operators defined on the interface

$$(\mathcal{K}q)(\mathbf{x}) = \int_\Gamma \frac{\partial G(\mathbf{x}-\mathbf{y})}{\partial \mathbf{n}_\mathbf{x}} q(\mathbf{y})\, ds_\mathbf{y} \qquad \text{on}\,\Gamma,$$
$$(\mathcal{L}V_m)(\mathbf{x}) = \int_\Gamma \frac{\partial G(\mathbf{x}-\mathbf{y})}{\partial \mathbf{t}_\mathbf{x}} \frac{\partial V_m(\mathbf{y})}{\partial \mathbf{t}_\mathbf{y}}\, ds_\mathbf{y} \qquad \text{on } \Gamma.$$

We solve the integral equation (3.7) by simple iteration. The spectrum of the operator $\mathcal{K}$ is contained in the interval $-\frac{1}{2} < \lambda \leq \frac{1}{2}$, and consequently the iteration

$$q_{n+1} = (1-\beta) q_n + 2\beta\mu (g - \mathcal{K} q_n)$$

converges to the exact solution for $0 < \beta < 2/(1+\mu)$.

4 Evaluation of Boundary Integrals

To compute the single layer potential in (3.5) we will integrate by parts so that the kernel is the tangential derivative of G. We first note that for each interface component Γ_k

$$\int_{\Gamma_k} q(\mathbf{y})\, ds_\mathbf{y} = 0$$

where q is defined in (3.4). To see this, we first note that the integral of $\partial\Phi_i(\mathbf{y})/\partial\mathbf{n}_\mathbf{y}$ is zero, using the divergence theorem in the interior of Γ_k. But the same is true for $\partial\Phi_e(\mathbf{y})/\partial\mathbf{n}_\mathbf{y}$ because of the flux condition (2.2), and therefore also for q. Thus

$$Q(r) = \int_0^r q(\mathbf{y})\, ds_\mathbf{y}$$

is a periodic function of the arc length, and we can write

$$\int_{\Gamma_k} G(\mathbf{y}-\mathbf{x}) q(\mathbf{y})\, ds_\mathbf{y} = -\int_{\Gamma_k} \frac{\partial G(\mathbf{y}-\mathbf{x})}{\partial \mathbf{t}_\mathbf{y}} Q(s_\mathbf{y})\, ds_\mathbf{y}.$$

Now to evaluate $\Phi(\mathbf{x})$ in (3.5) we will compute boundary integrals of the form

$$u^{(k)}(\mathbf{x}) = \int_{\Gamma_k} \frac{\partial G(\mathbf{y}-\mathbf{x})}{\partial \mathbf{n}_\mathbf{y}} f(\mathbf{y})\, ds_\mathbf{y} \tag{4.1}$$

$$v^{(k)}(\mathbf{x}) = \int_{\Gamma_k} \frac{\partial G(\mathbf{y}-\mathbf{x})}{\partial \mathbf{t}_\mathbf{y}} f(\mathbf{y})\, ds_\mathbf{y} \tag{4.2}$$

on each interface component Γ_k. When $\mathbf{x}$ is far away from Γ_k we replace the integral by a trapezoidal sum and use the fast summation method described in the next section. When $\mathbf{x}$ is close to Γ_k, we compute the integrals using the method developed in [5]. We

summarize the procedure here, with slight differences in notation. The singular integral is regularized with a length parameter δ, usually chosen to be of the order of the grid size. The integral is replaced by a sum, and two corrections are added. The resulting approximation to the integral is accurate to about $O(\delta^3)$.

Assume each Γ_k is parametrized as $\mathbf{y}=\mathbf{y}(\alpha)$, $0\le\alpha\le 2\pi$, and the α-segment is partitioned into J_k segments with grid points $\alpha_j^{(k)}=2\pi j/J_k$, $1\le j\le J_k$. We express $u^{(k)}$ as

$$\begin{aligned} u^{(k)}(\mathbf{x}) &= \int_{\Gamma_k} \frac{\partial G(\mathbf{y}-\mathbf{x})}{\partial \mathbf{n}_\mathbf{y}} f(\mathbf{y})\, ds_\mathbf{y} \\ &= \int_0^{2\pi} N(\alpha)\cdot\nabla G(\mathbf{y}(\alpha)-\mathbf{x})[f(\alpha)-f(\alpha_0^{(k)})]\,d\alpha + \chi(\mathbf{x}) f(\alpha_0^{(k)}). \end{aligned} \tag{4.3}$$

Here $N(\alpha)=(y_2'(\alpha),-y_1'(\alpha))$; $\chi(\mathbf{x})=1$ for $\mathbf{x}$ inside Γ_k, $\chi(\mathbf{x})=0$ for $\mathbf{x}$ outside; and $\mathbf{y}(\alpha_0^{(k)})$ is the closest point on Γ_k to $\mathbf{x}$, so that $\mathbf{x}=\mathbf{y}_0+b\mathbf{n_0}$, for some b, where $\mathbf{y}_0=\mathbf{y}(\alpha_0^{(k)})$ and $\mathbf{n_0}=\mathbf{n}_{\mathbf{y_0}}$. Replacing ∇G with a regularized version

$$\begin{aligned} \nabla G_\delta(\mathbf{y}(\alpha)-\mathbf{x}) &= (1-e^{-r^2/\delta^2})\nabla G(\mathbf{y}(\alpha)-\mathbf{x}) \\ &= \frac{1}{2\pi r^2}(1-e^{-r^2/\delta^2})(\mathbf{y}(\alpha)-\mathbf{x}), \end{aligned} \tag{4.4}$$

with $r=|\mathbf{y}(\alpha)-\mathbf{x}|$, we approximate $u_k(\mathbf{x})$ by

$$u_h^{(k)}(\mathbf{x}) = \frac{2\pi}{J_k} S_{u,1}^{(k)}(\mathbf{x}) - \frac{2\pi}{J_k} f(\alpha_0^{(k)}) S_{u,0}^{(k)}(\mathbf{x}) + \chi(\mathbf{x}) f(\alpha_0^{(k)}) + T_{u,1} + T_{u,2}$$

with

$$S_{u,1}^{(k)}(\mathbf{x}) = \sum_{j=1}^{J_k} N(\alpha_j^{(k)})\cdot\nabla G_\delta(\mathbf{y}(\alpha_j^{(k)})-\mathbf{x}) f(\alpha_j^{(k)}) \tag{4.5}$$

and

$$S_{u,0}^{(k)}(\mathbf{x}) = \sum_{j=1}^{J_k} N(\alpha_j^{(k)})\cdot\nabla G_\delta(\mathbf{y}(\alpha_j^{(k)})-\mathbf{x}). \tag{4.6}$$

The terms $T_{u,1}$ and $T_{u,2}$ are corrections for regularization and discretization, derived in [5]. The first is

$$T_{u,1} = -\delta^2(4\pi)^{-1}\eta\left(\sqrt{\pi}e^{-\eta^2} - \pi|\eta|\operatorname{erfc}|\eta|\right)\left(|\mathbf{y}_0'|^{-2} f_0'' - |\mathbf{y}_0'|^{-4}(\mathbf{y}_0''\cdot\mathbf{y}_0') f_0'\right)$$

where $\eta=b/\delta$, $\mathbf{y}_0'=\mathbf{y}'(\alpha_0^{(k)})$, and similarly for $\mathbf{y}_0''$, f_0', f_0''. The second is

$$T_{u,2} = -\frac{h f_0'\eta\sigma}{2}\sum_{n=1}^{\infty}\sin(2n\pi\alpha_0^{(k)}/h) E^+(\eta, n\pi\sigma)$$

with $\sigma = \delta/(h|\mathbf{x}_0'|)$ and

$$E^{\pm}(\eta,\zeta) = e^{2\eta\zeta}\mathrm{erfc}(\eta+\zeta) \pm e^{-2\eta\zeta}\mathrm{erfc}(-\eta+\zeta)$$

The sum in $T_{u,2}$ is rapidly convergent.

The treatment of $v^{(k)}$ is similar:

$$\begin{aligned} v^{(k)}(\mathbf{x}) &= \int_{\Gamma_k} \frac{\partial G(\mathbf{y}-\mathbf{x})}{\partial \mathbf{t}_\mathbf{y}} f(\mathbf{y})\, ds_\mathbf{y} \\ &= \int_0^{2\pi} \mathbf{y}'(\alpha)\cdot\nabla G(\mathbf{y}(\alpha)-\mathbf{x})[f(\alpha)-f(\alpha_0^{(k)})]\, d\alpha. \end{aligned} \tag{4.7}$$

We approximate $v^{(k)}(\mathbf{x})$ by

$$v_h^{(k)}(\mathbf{x}) = \frac{2\pi}{J_k} S_{v,1}^{(k)}(\mathbf{x}) - \frac{2\pi}{J_k} f(\alpha_0^{(k)}) S_{v,0}^{(k)}(\mathbf{x}) + T_{v,1} + T_{v,2}.$$

with

$$S_{v,1}^{(k)}(\mathbf{x}) = \sum_{j=1}^{J_k} \mathbf{y}'(\alpha_j^{(k)})\cdot\nabla G_\delta(\mathbf{y}(\alpha_j^{(k)})-\mathbf{x}) f(\alpha_j^{(k)}) \tag{4.8}$$

and

$$S_{v,0}^{(k)}(\mathbf{x}) = \sum_{j=1}^{J_k} \mathbf{y}'(\alpha_j^{(k)})\cdot\nabla G_\delta(\mathbf{y}(\alpha_j^{(k)})-\mathbf{x}). \tag{4.9}$$

The corrections $T_{v,1}$ and $T_{v,2}$ are

$$T_{v,1} = -\delta(2\pi)^{-1} f_0' |\mathbf{y}_0'|^{-1} (1+\kappa_0\eta\delta/2)\left(\sqrt{\pi}e^{-\eta^2} - \pi|\eta|\mathrm{erfc}\,|\eta|\right)$$

and

$$T_{v,2} = h f_0' \sum_{n=1}^{\infty} \cos(2n\pi\alpha_0^{(k)}/h)\left(-\frac{\eta\sigma}{2}E^-(\eta, n\pi\sigma) + \frac{\sigma}{\sqrt{\pi}} e^{-\eta^2} e^{-n^2\pi^2\sigma^2}\right)$$

where κ_0 is the curvature at $\mathbf{y}_0$, defined by $\mathbf{y}_{ss} = \kappa_0 \mathbf{n_0}$.

For the integral equation (3.6) we need to compute partial derivatives of the single layer potential. To do this we rewrite them in integrals in the form of $u^{(k)}$ and $v^{(k)}$ above,

$$\begin{aligned} \int_\Gamma \frac{\partial G}{\partial x_1}(\mathbf{y}-\mathbf{x}) q(\mathbf{y})\, ds_\mathbf{y} &= -\int_\Gamma \frac{\partial G(\mathbf{y}-\mathbf{x})}{\partial \mathbf{n}_\mathbf{y}} q(\mathbf{y}) y_2'(s)\, ds_\mathbf{y} \\ &\quad - \int_\Gamma \frac{\partial G(\mathbf{y}-\mathbf{x})}{\partial \mathbf{t}_\mathbf{y}} q(\mathbf{y}) y_1'(s)\, ds_\mathbf{y}, \\ \int_\Gamma \frac{\partial G}{\partial x_2}(\mathbf{y}-\mathbf{x}) q(\mathbf{y})\, ds_\mathbf{y} &= \int_\Gamma \frac{\partial G(\mathbf{y}-\mathbf{x})}{\partial \mathbf{n}_\mathbf{y}} q(\mathbf{y}) y_1'(s)\, ds_\mathbf{y} \\ &\quad - \int_\Gamma \frac{\partial G(\mathbf{y}-\mathbf{x})}{\partial \mathbf{t}_\mathbf{y}} q(\mathbf{y}) y_2'(s)\, ds_\mathbf{y}. \end{aligned}$$

These formulas are easily justified for $\mathbf{x} \notin \Gamma$, and they also hold for $x \in \Gamma$, again in the principal value sense, by averaging the limiting values on the two sides. These integrals can be evaluated in the manner just described.

5 The Fast Multipole Method

For each fixed target point $\mathbf{x}$, the summations (4.5)-(4.6) and (4.8)-(4.9) involve computational work on the order of $J = \sum_{k=1}^{K} J_k$. Thus the evaluation of boundary integrals at $O(J)$ target points requires computational work of $O(J^2)$. The fast multipole method, which was introduced by Rokhlin and Greengard [9] and has been acclaimed as one of the top-ten algorithms of the 20^{th} century [7], is applied to reduce to the complexity of these summations from $O(J^2)$ to $O(J\log J)$.

The sums (4.5) and (4.8) for the different interface components Γ_k can be combined. That is, the total summations

$$S_{u,1}(\mathbf{x}) = \sum_{k=1}^{K} S_{u,1}^{(k)}(\mathbf{x}) = \sum_{k=1}^{K}\sum_{j=1}^{J_k} N_k(\alpha_j^{(k)})\cdot\nabla G_\delta(\mathbf{y}(\alpha_j^{(k)})-\mathbf{x}) f(\alpha_j^{(k)})$$

and

$$S_{v,1}(\mathbf{x}) = \sum_{k=1}^{K} S_{v,1}^{(k)}(\mathbf{x}) = \sum_{k=1}^{K}\sum_{j=1}^{J_k} \mathbf{y}'(\alpha_j^{(k)})\cdot\nabla G_\delta(\mathbf{y}(\alpha_j^{(k)})-\mathbf{x}) f(\alpha_j^{(k)}),$$

each of which is in the form

$$w(\mathbf{x}_i) = \sum_{j=1}^{J} A(\mathbf{y}_j,\mathbf{x}_i) q(\mathbf{y}_j), \tag{5.1}$$

are computed at once by the fast multipole method. However, for the sums (4.6) and (4.9), we have to apply the fast multipole method separately for each component Γ_k, since the multiplication of the sum by $f(\alpha_0^{(k)})$, which depends on the target point $\mathbf{x}$, makes the total sum over the different components fail to be in the form (5.1).

In two space dimensions, the double layer potential (4.1) and the tangential derivative (4.2) of the single layer potential happen to be the real and imaginary parts of a Cauchy integral if the spatial coordinates are treated as complex variables. The fast multipole method uses the multipole expansions for the kernel of the Cauchy integral to group sources that lie close together, but far away from the target point, and treat them as if they are a single source.

In the fast multipole method, each point is associated with a list of near field interaction points and a list of far field interaction points, the latter of which is not explicitly stored. The part of the summation due to contributions from the far field points is computed by the multipole expansions and translations, including far field expansion, multipole to multipole translation from coarse to fine level and multipole to local translation at the same level [9].

Since it is not clear if a simple multipole expansion can be used for the regularized kernel, in this work the standard kernel is used in the far field calculations with the fast

multipole method while the regularized version of the fundamental solution is used only for those grid nodes $\mathbf{y}(\alpha_j)$ which are on the near neighbor interaction list of the target point $\mathbf{x}$. The error due to the approximation should be negligible when the smoothing parameter δ is sufficiently small since in this case the exponential function e^{-r^2/δ^2} with $r = |\mathbf{y}-\mathbf{x}|$ in (4.4) is almost zero for source points which are not in the near neighbor interaction list.

The fast multipole method implemented in this work represents the discrete points by a quad-tree data structure. Each quad-tree node represents a rectangular patch and is allowed to contain at most C source points; C is called the capacity parameter of the quad-tree structure.

Next we give a method to roughly estimate the capacity parameter C that is reasonably large enough for the multipole summation to be accurate as well as efficient. For a source point $\mathbf{y}$ which is not in the near neighbor interaction list of a target point $\mathbf{x}$, its distance $r = |\mathbf{y}-\mathbf{x}|$ away from the target point is on average about $Ch_k/2$ with $h_k = \mathcal{L}_k/J$, assuming the source points locally are aligned nearly along straight lines. Here $\mathcal{L}_k$ is the arc length of the k^{th} interface component Γ_k. In order for the error in replacing the standard kernel with the regularized one (4.4) to be within some tolerance parameter ϵ, the capacity parameter C needs to satisfy

$$\exp\{-(\frac{Ch_k}{2\delta})^2\} < \epsilon$$

or

$$C > \frac{2\delta}{h_k}\sqrt{|\ln\epsilon|}.$$

For $\epsilon = 10^{-8}$, we have $\sqrt{|\ln\epsilon|} \approx 4.29$ and an estimate for the lower bound of the capacity parameter, $C > 8.58 \times \delta/h_k$. This will guide us in the selection of the capacity parameter for the fast multipole summation. However, the practical lower bound of the capacity parameter varies due to the local symmetry and curvature of the interface.

6 Numerical Results

We present numerical results for boundary value problems and interface problems which illustrate the methods just described. Since double and single layer potentials can represent the solution to a Fredholm boundary integral equation of the second kind for the Dirichlet and Neumann boundary value problems, respectively, we begin with examples of both to illustrate the accuracy of the computational method. We then proceed to problems with interfaces having several components.

The first two examples solve the boundary value problems in a general domain, which is embedded into a larger rectangular box. Other examples solve the interface problem with the computational domain again being a rectangular box, with irregular interfaces embedded. In the examples, the rectangular box is the square $-1.5 \le x, y \le 1.5$,

partitioned by a Cartesian grid (see Figure 6.1). The nodes adjacent to the irregular domain boundary or interface (i.e., those whose discrete Laplacian crosses the interface) are called *irregular* grid nodes, and the others are called *regular* grid nodes. The values of the boundary integrals at the irregular grid nodes (the solid marked points in Figure 6.1), their neighboring regular nodes (the other marked points in Figure 6.1) and the boundary nodes of the regular box are computed by the accurate evaluation method together with the fast summation technique, which were introduced in the previous sections. The values of the solution to the boundary value or interface problem at other regular grid nodes are then computed by solving a discrete Poisson equation. The right hand side of the discrete Poisson equation is non-zero only at the irregular grid nodes. The value at an irregular node is the discrete Laplacian of the values of the computed boundary integrals. According to [5] the error in the integrals at the irregular points is about $O(\delta^3)$ if the smoothing parameter δ is not small relative to the grid size. The theory developed in [6] predicts that the resulting error in the values at the regular points should be about $O(\delta^2)$.

The maximum numerical errors at the irregular grid nodes and regular grid nodes are reported in Tables 1-5 and Table 7 for each problem with various grid sizes and numerical parameters. In the tables, C denotes the capacity parameter of the quad-tree node in the fast multipole summation (the case $C = 0$ corresponds to the direct summation); N denotes the number of grid cells along one direction; M denotes the number of nodes on each component of the interface (if the interface has multiple disconnected components); $\|e^{\text{irreg}}\|_\infty$ represents the maximum of the errors of the computed values of the boundary integral(s) at the irregular grid nodes; $\|e^{\text{reg}}\|_\infty$ is the maximum error at the regular grid nodes of the numerical solution, which is obtained by inverting the discrete Laplacian with a fast Poisson solver.

In the experiments, we determine the smoothing parameter δ by choosing constant γ, which is also called a smoothing parameter, and setting

$$\delta = \gamma \frac{2\pi r_{\min}}{J}.$$

so that δ is on the order of $h_k = \mathcal{L}_k / J$ for each interface component. Here, $r_{\min}$ is the minimum of the semi-axes of the ellipse or all ellipses if the interface consists of multiple ellipses.

The computer codes for the numerical examples were written in the C++ computer language and are available to the readers upon request. The programs were run in a desktop computer which has a 2.8GHz Intel Xeon processor.

6.1 Numerical results for boundary value problems

Numerical results for the boundary value problems with the fast and accurate boundary integral method are presented in this subsection.

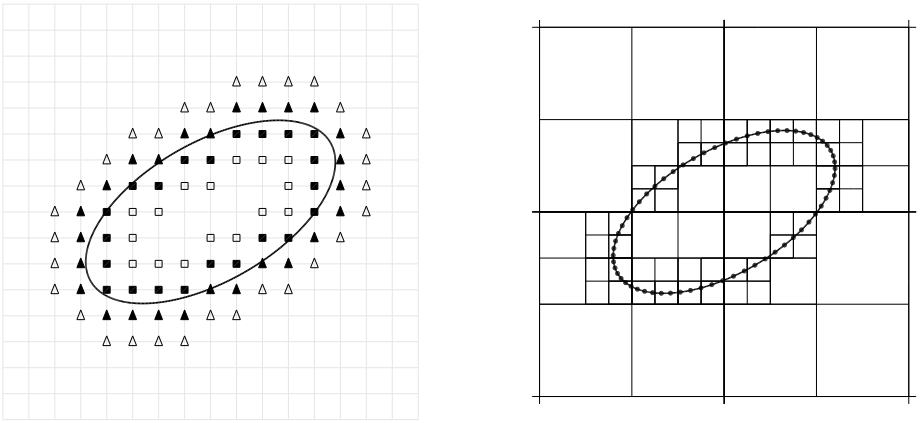


Figure 1: (a) A Cartesian grid with the irregular grid nodes being marked; (b) adaptive grid generated by the fast multipole algorithm

Example 1. Our first example is an interior Dirichlet boundary value problem for the Laplace equation. The computational domain is the interior of the ellipse

$$\{(x,y)\,|\,\frac{x^2}{4}+y^2<1\},$$

rotated by the angle $\pi/6$ counterclockwise about the origin. The exact solution is given by

$$u(x,y)=x^3-3xy^2.$$

The problem is solved by the boundary integral method with u prescribed on the ellipse. Figure 2 shows isolines of the numerical solution to the Dirichlet problem. Table 1 contains errors of the numerical solutions with different parameters. The second and third rows list errors when the numerical quadratures are computed by the direct summation ($C=0$) instead of the fast multipole summation. The fourth and fifth rows list errors when the fast multipole method with capacity parameter $C=20$ is used for the summation while the smoothing parameter δ or γ is the same as before. These two sets of results show that the error due to approximation of the regularized kernel by the standard one, used in the multipole method for points that are well separated, is too small to influence the errors at the irregular and regular grid nodes. Another indication of these cases is that the order of accuracy at irregular grid nodes is only about two instead of three because the smoothing parameter $\gamma=1$ is too small. In contrast, the errors in later rows of Table 1 show that the numerical solution has third order accuracy at irregular grid nodes and nearly second order accuracy at regular grid nodes when the smoothing parameter γ is sufficiently large.

Example 2. Next an interior Neumann boundary value problem of the Laplace equation is solved with the boundary integral method. The computational domain and exact

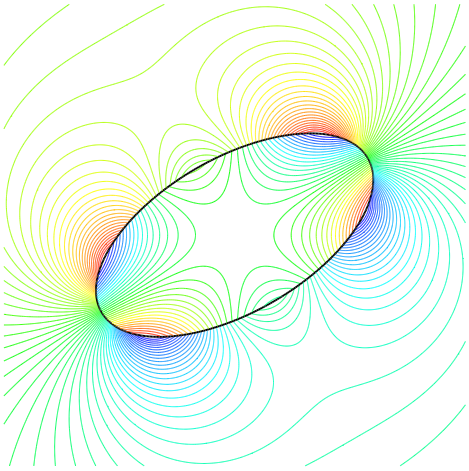


Figure 2: Isolines of a numerical solution to the Dirichlet BVP

Table 1: Numerical errors for the Dirichlet BVP (example 1)

C	γ	$N = M/2$	64	128	256	512	1024	2048
0	1	$\|e^{\text{irreg}}\|_\infty$	6.47E-4	9.00E-5	2.43E-5	5.57E-6	1.52E-6	3.80E-7
0	1	$\|e^{\text{reg}}\|_\infty$	3.64E-3	1.39E-3	5.42E-4	1.63E-4	4.76E-5	1.32E-5
20	1	$\|e^{\text{irreg}}\|_\infty$	6.47E-4	9.00E-5	2.43E-5	5.57E-6	1.52E-6	3.80E-7
20	1	$\|e^{\text{reg}}\|_\infty$	3.64E-3	1.39E-3	5.42E-4	1.63E-4	4.76E-5	1.32E-5
20	2	$\|e^{\text{irreg}}\|_\infty$	5.18E-3	6.61E-4	7.98E-5	9.98E-6	1.24E-6	1.57E-7
20	2	$\|e^{\text{reg}}\|_\infty$	5.50E-3	2.44E-3	8.79E-4	2.55E-4	7.23E-5	1.93E-5
20	4	$\|e^{\text{irreg}}\|_\infty$	4.68E-2	5.44E-3	6.60E-4	8.03E-5	9.99E-6	1.24E-6
20	4	$\|e^{\text{reg}}\|_\infty$	7.55E-3	3.49E-3	1.05E-3	3.01E-4	8.55E-5	2.32E-5

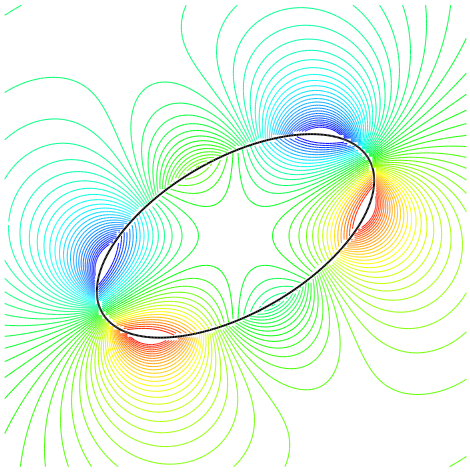


Figure 3: Isolines of a numerical solution to the Neumann BVP

Table 2: Numerical errors for the Neumann BVP (example 2)

C	γ	$N=M/2$	64	128	256	512	1024	2048
0	1	$\|e^{\text{irreg}}\|_\infty$	9.89E-4	2.25E-4	6.11E-5	1.71E-5	4.24E-6	1.04E-6
0	1	$\|e^{\text{reg}}\|_\infty$	3.53E-3	9.41E-4	4.32E-4	1.17E-4	3.80E-5	1.07E-5
20	1	$\|e^{\text{irreg}}\|_\infty$	9.83E-4	2.25E-4	6.12E-5	1.70E-5	4.25E-6	1.04E-6
20	1	$\|e^{\text{reg}}\|_\infty$	3.42E-3	9.39E-4	4.29E-4	1.17E-4	3.80E-5	1.07E-5
20	2	$\|e^{\text{irreg}}\|_\infty$	3.49E-3	4.02E-4	5.92E-5	7.39E-6	9.40E-7	1.17E-7
20	2	$\|e^{\text{reg}}\|_\infty$	4.97E-3	1.40E-3	5.50E-4	1.67E-4	4.77E-5	1.29E-5
20	4	$\|e^{\text{irreg}}\|_\infty$	2.96E-2	3.66E-3	4.75E-4	5.96E-5	7.46E-6	9.30E-7
20	4	$\|e^{\text{reg}}\|_\infty$	2.60E-2	7.92E-3	2.39E-3	6.88E-4	1.86E-4	4.74E-5

solution are the same as for the Dirichlet case. The solution to the Neumann boundary value problem as well as that to the corresponding boundary integral equation is not unique. To fix a solution, we require the density has zero mean and set the value of the numerical solution at the center of the ellipse to be zero. Figure 3 shows isolines of a numerical solution to the Dirichlet boundary value problem. Table 2 contains errors for the numerical solutions. As in the previous example, the second and third rows list errors when the numerical quadratures are computed by the direct summation ($C=0$) instead of the fast multipole summation, and the next two rows list errors when the fast multipole method with capacity parameter $C=20$ is used for the summation while the smoothing parameter, with γ unchanged. Again the difference in these two sets of results is negligible. The orders of accuracy for the irregular and regular grid nodes are similar to those in the Dirichlet example.

6.2 Numerical results for the interface problems

Numerical results for the interface problem with the fast and accurate boundary integral method are presented in this subsection. The interface may have multiple disjoint components. Let K be the number of interface components. Each component Γ_k is assumed to be an ellipse Γ_k, for $k \in \{1,2,\cdots,K\}$. For ellipse Γ_k, its major and minor axes are denoted by a_k and b_k; the coordinates of its center are denoted by $(c_{k,1}, c_{k,2})$.

To solve an interface problem whose exact solution is known and further verify accuracy of the method, the homogeneous interface condition (2.2) is replaced by the non-homogeneous one

$$\frac{1}{\sigma_i+\sigma_e}\left[\sigma_i \frac{\partial \Phi_i(\mathbf{x})}{\partial \mathbf{n}_\mathbf{x}} - \sigma_e \frac{\partial \Phi_e(\mathbf{x})}{\partial \mathbf{n}_\mathbf{x}}\right] = J_m \qquad \text{on}\,\Gamma$$

with J_m be a known function. Correspondingly, instead of the integral equation (3.7), the following one

$$\frac{1}{2}q + \mu \mathcal{K} q = \mu g + J_m \qquad \text{on } \Gamma \tag{6.1}$$

is solved.

The known functions, V_m and J_m, and the vector $\mathbf{E}$ are chosen such that the exact solution of the interface problem is given by

$$\Phi_i(\mathbf{x}) = -\frac{\sigma_e}{\sigma_i+\sigma_e} x_1 \qquad \mathbf{x} \in \Omega_i,$$

and

$$\Phi_e(\mathbf{x}) = -x_1 - \frac{\sigma_e}{\sigma_i+\sigma_e} \sum_{k=1}^{K} \frac{a_k^2 (x_1 - c_{k,1})}{(x_1-c_{k,1})^2 + (x_2-c_{k,2})^2} \qquad \mathbf{x} \in \Omega_e, \tag{6.2}$$

which indicates the field vector $\mathbf{E}$ is given by $\mathbf{E} = (1,0)^{\mathrm{T}}$. The conductivities are fixed to be $\sigma_e = 2$ and $\sigma_i = 1$.

Example 3. In this example, the interface is one ellipse, the same as in the previous two examples for boundary value problems. Figure 4 shows isolines of a numerical solution to the interface problem. Table 3 contains errors of the numerical solutions with different capacity and smoothing parameters. The results show that the expected third order accuracy at irregular grid nodes and second order accuracy at regular grid nodes are observed only when the capacity and smoothing parameters are sufficiently large. In particular, in the case that $\gamma = 3$ and $C = 10$, no convergence is evident, as the capacity parameter C is so small that the approximation of the regularized kernel by the standard one in the fast multipole summation introduces too much error. Table 3 also lists the CPU times used by the computer program for runs with different capacity and smoothing parameters. The timing results show that the computational work by this method is linearly proportional to the number of unknowns or grid nodes on the rectangular box.

Example 4. In this example the interface consists of two ellipses. The first ellipse is centered at point $(c_{1,1}, c_{1,2})^{\mathrm{T}} = (0.375, 0.5)^{\mathrm{T}}$ with semi-axes $(a_1, b_1)^{\mathrm{T}} = (0.75, 0.3)^{\mathrm{T}}$ and rotation

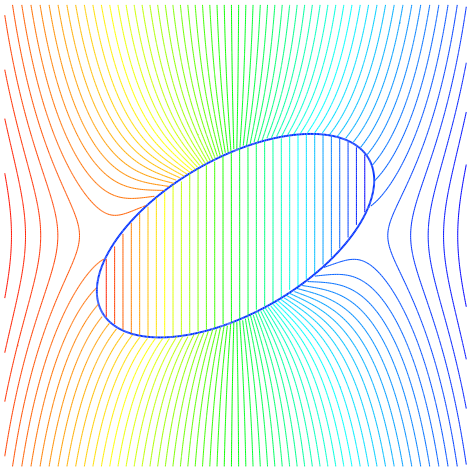
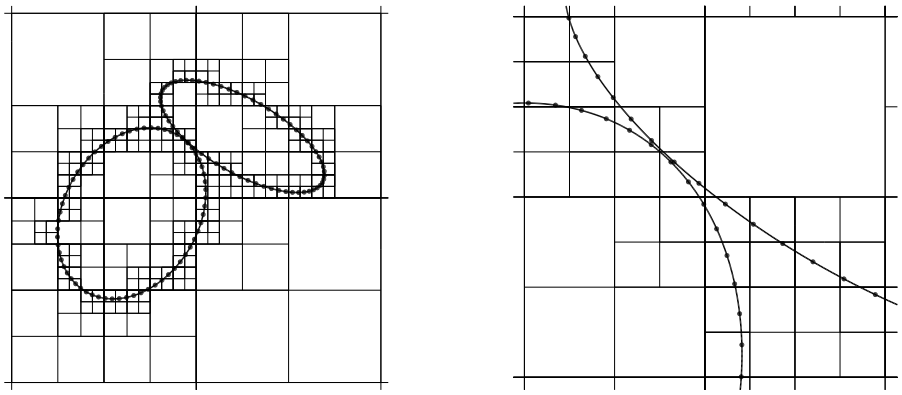


Figure 4: Isolines of a numerical solution to the interface problem with one ellipse

Table 3: Numerical errors for the interface problem with one ellipse (example 3)

C	γ	$N=M$	64	128	256	512	1024	2048
10	3	$\|e_h^{\text{irreg}}\|_\infty$	1.73E-3	3.71E-3	4.84E-3	4.61E-5	5.18E-3	6.81E-3
10	3	$\|e_h^{\text{reg}}\|_\infty$	2.41E-3	2.61E-3	6.41E-3	6.02E-5	9.60E-3	1.02E-2
10	3	t_{cpu} (secs)	0.050	0.12	0.28	0.62	1.4	3.5
20	3	$\|e_h^{\text{irreg}}\|_\infty$	1.73E-3	2.34E-4	2.70E-5	3.23E-6	4.15E-7	1.58E-6
20	3	$\|e_h^{\text{reg}}\|_\infty$	2.41E-3	6.69E-4	1.99E-4	5.10E-5	1.32E-5	3.37E-6
20	3	t_{cpu} (secs)	0.050	0.13	0.26	0.55	1.4	3.3
40	3	$\|e_h^{\text{irreg}}\|_\infty$	1.73E-3	2.34E-4	2.70E-5	3.22E-6	4.15E-7	6.55E-8
40	3	$\|e_h^{\text{reg}}\|_\infty$	2.41E-3	6.69E-4	1.99E-4	5.08E-5	1.32E-5	3.36E-6
40	3	t_{cpu} (secs)	0.060	0.17	0.41	0.66	1.5	3.7
40	5	$\|e_h^{\text{irreg}}\|_\infty$	6.47E-3	9.16E-4	1.20E-4	1.50E-5	1.87E-6	2.46E-7
40	5	$\|e_h^{\text{reg}}\|_\infty$	4.85E-3	1.44E-3	4.28E-4	1.13E-4	3.00E-5	7.72E-6
40	5	t_{cpu} (secs)	0.060	0.17	0.40	0.66	1.5	3.7



(a) (b)

Figure 5: Adaptive grids for the fast multipole summation (two cells)

angle $\theta = -30$ degrees. The second ellipse is centered at $(c_{2,1}, c_{2,2})^{\mathrm{T}} = (-0.525, -0.125)^{\mathrm{T}}$ with semi-axes $(a_2, b_2)^{\mathrm{T}} = (0.7375,\ 0.55)^{\mathrm{T}}$ and rotation angle $\theta = 60$ degrees. These two ellipses are chosen to be very close to each other but not intersecting. The distance of the ellipses is about 0.005. See Figure 5 for the ellipses and a close-up of the interface. Figure 6 shows isolines of a numerical solution to the interface problem. Errors of the numerical solutions with different parameters are in Table 4. The results in Table 4 are consistent with those in the previous example. The expected third order accuracy at irregular grid nodes and second order accuracy at regular grid nodes are observed only when the capacity and smoothing parameters are sufficiently large. Again when γ=3 and $C = 10$, convergence is not found. Run times shown in Table 4 again show computational work proportional to the number of grid nodes.

Example 5. In this example the interface consists of three ellipses. The first ellipse is centered at point $(c_{1,1}, c_{1,2})^{\mathrm{T}} = (0.625, -0.225)^{\mathrm{T}}$ with semi-axes $(a_1, b_1)^{\mathrm{T}} = (0.775, 0.375)^{\mathrm{T}}$ and rotation angle $\theta = -60$ degrees. The second ellipse is centered at $(c_{2,1}, c_{2,2})^{\mathrm{T}} = (-0.125, 0.625)^{\mathrm{T}}$ with semi-axes $(a_2, b_2)^{\mathrm{T}} = (0.625, 0.35)^{\mathrm{T}}$ and rotation angle $\theta = 30$ degrees. The third ellipse is centered at $(c_{3,1}, c_{3,2})^{\mathrm{T}} = (-0.575, -0.375)^{\mathrm{T}}$ with semi-axes $(a_3, b_3)^{\mathrm{T}} = (0.6, 0.4)^{\mathrm{T}}$ and rotation angle $\theta = 60$ degrees. These three ellipses are chosen to be very close to each other but not intersecting. The closest distance between the ellipses is about 0.006. See Figure 7 for the ellipses and a close-up of the interface. Figure 8 shows isolines of a numerical solution to the interface problem. Table 5 contains errors of the numerical solutions with different parameters. The results in Table 5 are consistent with those in the previous two examples. The expected third order accuracy at irregular grid nodes and second order accuracy at regular grid nodes are observed only when the capacity and smoothing parameters are sufficiently large, but not with $\gamma = 3$ and $C = 10$. The run times are again proportional to the number of grid nodes.

Example 6. In this example, the interface consists of twenty ellipses, whose axes and centers are listed in Table 6. The ellipses and a close-up are illustrated in Figure 9.

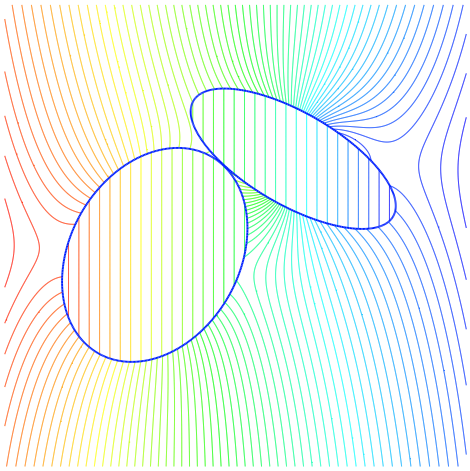
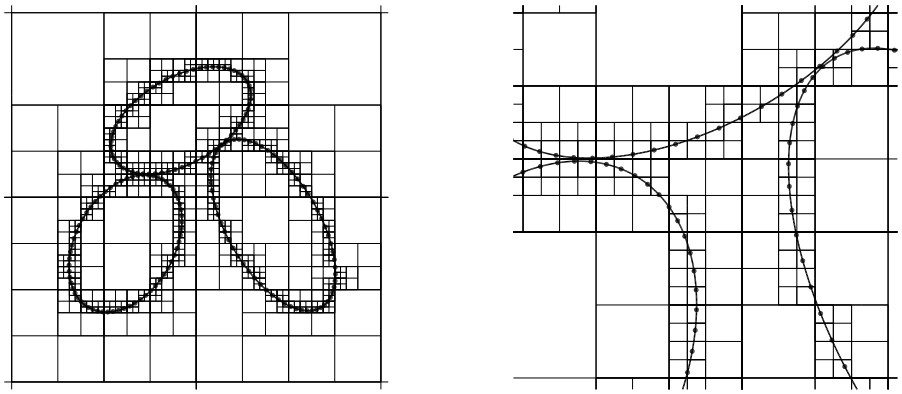


Figure 6: Isolines of a numerical solution to the interface problem with two ellipses

Table 4: Numerical errors for the interface problem with two ellipses (example 4)

C	γ	$N=M$	64	128	256	512	1024	2048
10	3	$\|e_h^{\text{irreg}}\|_\infty$	2.74E-3	4.99E-4	1.23E-3	1.42E-3	1.24E-3	1.56E-3
10	3	$\|e_h^{\text{reg}}\|_\infty$	3.51E-3	1.00E-3	1.32E-3	1.40E-3	1.82E-3	3.91E-3
10	3	t_{cpu} (secs)	0.24	0.49	0.95	1.9	3.9	8.9
20	3	$\|e_h^{\text{irreg}}\|_\infty$	2.74E-3	5.04E-4	6.48E-5	8.51E-6	1.03E-6	1.25E-7
20	3	$\|e_h^{\text{reg}}\|_\infty$	3.64E-3	1.07E-3	2.65E-4	7.18E-5	1.81E-5	4.53E-6
20	3	t_{cpu} (secs)	0.23	0.48	0.83	1.6	3.4	7.8
40	3	$\|e_h^{\text{irreg}}\|_\infty$	2.74E-3	5.04E-4	6.48E-5	8.50E-6	1.03E-6	1.25E-7
40	3	$\|e_h^{\text{reg}}\|_\infty$	3.64E-3	1.07E-3	2.65E-4	7.18E-5	1.81E-5	4.51E-6
40	3	t_{cpu} (secs)	0.29	0.65	1.2	1.9	3.9	8.6
40	5	$\|e_h^{\text{irreg}}\|_\infty$	1.03E-2	1.95E-3	2.96E-4	3.95E-5	4.85E-6	5.86E-7
40	5	$\|e_h^{\text{reg}}\|_\infty$	7.13E-3	1.72E-3	5.22E-4	1.54E-4	4.03E-5	1.03E-5
40	5	t_{cpu} (secs)	0.29	0.64	1.2	1.9	3.9	8.6



(a) (b)

Figure 7: Adaptive grids for the fast multipole summation (three cells)

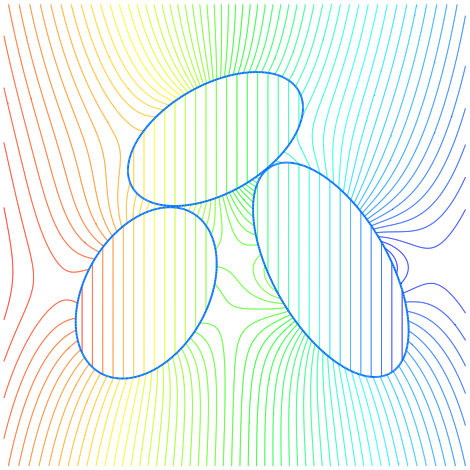


Figure 8: Isolines of a numerical solution to the interface problem with three ellipses

Table 5: Numerical errors for the interface problem with three ellipses (example 5)

C	γ	$N=M$	64	128	256	512	1024	2048
10	3	$\|e_h^{\text{irreg}}\|_\infty$	5.80E-3	1.16E-3	1.65E-3	1.69E-3	1.74E-3	6.93E-3
10	3	$\|e_h^{\text{reg}}\|_\infty$	6.31E-3	1.44E-3	1.32E-3	2.63E-3	4.58E-3	1.20E-2
10	3	t_{cpu} (secs)	0.36	0.68	1.3	2.5	5.3	12.0
20	3	$\|e_h^{\text{irreg}}\|_\infty$	5.80E-3	8.16E-4	1.02E-4	1.41E-5	2.17E-6	2.94E-7
20	3	$\|e_h^{\text{reg}}\|_\infty$	6.01E-3	1.28E-3	3.38E-4	1.02E-4	2.86E-5	6.80E-6
20	3	t_{cpu} (secs)	0.39	0.71	1.2	2.3	4.8	11.0
40	3	$\|e_h^{\text{irreg}}\|_\infty$	5.80E-3	8.16E-4	1.02E-4	1.41E-5	2.17E-6	2.94E-7
40	3	$\|e_h^{\text{reg}}\|_\infty$	6.01E-3	1.28E-3	3.38E-4	1.02E-4	2.86E-5	6.78E-6
40	3	t_{cpu} (secs)	0.43	1.0	1.8	2.7	5.8	12.0
40	5	$\|e_h^{\text{irreg}}\|_\infty$	2.32E-2	3.73E-3	4.85E-4	5.89E-5	8.63E-6	1.30E-6
40	5	$\|e_h^{\text{reg}}\|_\infty$	2.19E-2	3.57E-3	6.52E-4	1.85E-4	5.37E-5	1.48E-5
40	5	t_{cpu} (secs)	0.45	1.0	1.8	2.6	5.7	12.0

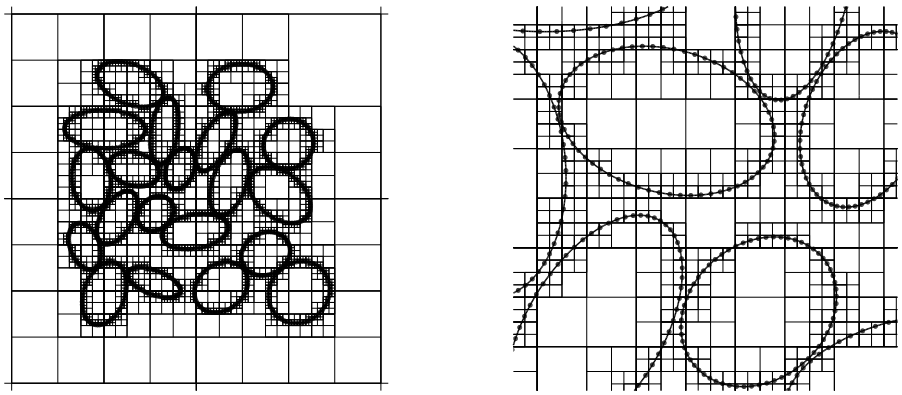
Figure 10 shows isolines of a numerical solution to the interface problem. Table 7 contains errors of the numerical solutions with different parameters. The results are consistent with the previous three examples. With large and compatible capacity and smoothing parameters ($C = 40$, $\gamma = 3$ or $\gamma = 4$), third order accuracy at irregular grid nodes and second order accuracy at regular grid nodes are observed. The timing results again show linear growth.

7 Discussion

This work describes a boundary integral method for potentials on closely packed cells. When portions of the boundary are close to each other, the boundary integrals become nearly singular. The nearly singular integrals are evaluated by a regularization of the integral kernel which admits analytically determined corrections to maintain accuracy. To speed up the dense matrix vector product associated with the boundary integrals, the fast multipole method is used. The combination of the fast multipole method and the regularized boundary integral makes the method fast as well as accurate. However, since the fast multipole method only works with the standard integral kernel instead of the regularized one, the approximation of the regularized kernel by the standard one in the far field computation of the fast multipole method introduces extra errors. To minimize the approximation errors, the capacity and smoothing parameters need to be appropriately selected. Numerical examples for both boundary value and interface problems show that, when the capacity and smoothing parameters are sufficiently large, the boundary integral method presented here yields high order accurate solutions, and the computational work is linearly proportional to the number of unknowns over the computational domain.

Table 6: Twenty ellipses for example 6: major axis a_k, minor axis b_k, center $(c_{k,1},c_{k,2})$ and rotation angle θ (in degree)

k	a_k	b_k	$c_{k,1}$	$c_{k,2}$	θ
1	0.216669	0.141285	−0.526154	0.25008	165.569
2	0.244085	0.193709	0.210888	−0.749594	37.818
3	0.28455	0.115038	−0.269197	0.572747	81.4637
4	0.29436	0.192655	0.708311	0.030127	141.49
5	0.284289	0.144615	−0.0103632	−0.278066	9.396
6	0.250819	0.133865	−0.668486	−0.161434	58.735
7	0.273781	0.246768	0.871021	−0.79315	47.446
8	0.218416	0.205301	0.780105	0.461611	47.5849
9	0.341714	0.162392	−0.773844	0.588366	179.192
10	0.276414	0.174403	−0.77664	−0.802865	70.189
11	0.237011	0.101913	−0.358575	−0.709595	159.004
12	0.180211	0.117365	−0.135826	0.253291	68.7521
13	0.278021	0.202428	0.372457	0.937647	4.0367
14	0.263711	0.163481	−0.889039	0.162802	95.2703
15	0.165073	0.13415	−0.34441	−0.126333	39.7459
16	0.212647	0.177712	0.588054	−0.466227	25.219
17	0.27797	0.117366	0.167368	0.484899	60.798
18	0.204751	0.124421	−0.94228	−0.403608	110.762
19	0.289704	0.142591	0.280349	0.135601	71.381
20	0.294926	0.168056	−0.558163	0.968711	157.659



(a) (b)

Figure 9: Adaptive grids for the fast multipole summation (twenty cells)

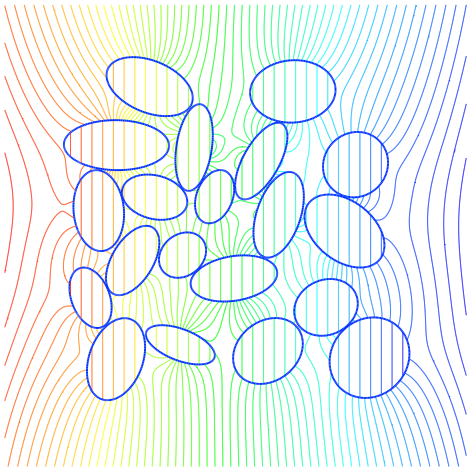


Figure 10: Isolines of a numerical solution to the interface problem with twenty ellipses

Table 7: Numerical errors for the interface problem with twenty ellipses (example 6)

C	γ	$N=M$	64	128	256	512	1024	2048
10	3	$\|e_h^{\text{irreg}}\|_\infty$	9.36E-3	1.20E-2	9.52E-3	9.88E-3	1.06E-2	1.07E-2
10	3	$\|e_h^{\text{reg}}\|_\infty$	9.84E-3	2.37E-2	4.80E-2	1.10E-1	1.12E-1	2.65E-1
10	3	t_{cpu} (secs)	3.3	6.0	11.	22.	44.	93.
20	3	$\|e_h^{\text{irreg}}\|_\infty$	1.25E-3	1.88E-4	2.89E-5	3.63E-6	2.60E-6	2.78E-6
20	3	$\|e_h^{\text{reg}}\|_\infty$	6.76E-3	3.11E-3	7.08E-4	1.74E-4	4.27E-5	1.84E-5
20	3	t_{cpu} (secs)	3.3	5.2	9.2	18.	35.	74.
40	3	$\|e_h^{\text{irreg}}\|_\infty$	1.25E-3	1.88E-4	2.89E-5	3.48E-6	4.08E-7	8.51E-8
40	3	$\|e_h^{\text{reg}}\|_\infty$	6.76E-3	3.12E-3	7.09E-4	1.74E-4	4.44E-5	1.11E-5
40	3	t_{cpu} (secs)	4.8	7.8	11.	19.	37.	76.
40	5	$\|e_h^{\text{irreg}}\|_\infty$	5.89E-3	8.70E-4	1.30E-4	1.68E-5	1.95E-6	2.69E-7
40	5	$\|e_h^{\text{reg}}\|_\infty$	1.33E-2	3.03E-3	6.80E-4	1.70E-4	4.45E-5	1.10E-5
40	5	t_{cpu} (secs)	4.7	7.6	11.	19.	36.	75.

It is possible that the fast multipole method could be replaced with other fast summation techniques such as the Barnes-Hut algorithm [3] so that both near field and far field computations could be done with the regularized kernel. In this way, the fine-tuning of the capacity and smoothing parameters might be avoided. The extension of the method to three space dimensions is straightforward. The corresponding work will be reported separately.

Acknowledgments

Research of the first author was supported in part by the National Science Foundation of the USA under Grant DMS–0915023, the National Natural Science Foundation of China under Grant DMS–11101278 and the Recruitment Program of Global Experts of China. Research of the second author was supported in part by the National Science Foundation of the USA under Grant DMS–0806482.

Appendix

In deriving the integral equation (3.6) we will need the normal derivative of the double layer potential such as (3.1) on Γ. For u as in (3.1), $\partial u/\partial \mathbf{n}$ is continuous at Γ, and

$$\frac{\partial u(\mathbf{x})}{\partial \mathbf{n}_\mathbf{x}} = \int_\Gamma \frac{\partial G(\mathbf{x}-\mathbf{y})}{\partial \mathbf{t}_\mathbf{x}} \frac{\partial f(\mathbf{y})}{\partial \mathbf{t}_\mathbf{y}} ds_\mathbf{y} \tag{7.1}$$

This is a classical fact (e.g. see [10], p. 5 for this formula, or [8], Thm. 2.23, p. 57 for the 3D case), but we sketch the derivation for completeness. Starting with the integral for $u(\mathbf{x})$ in (3.1), for $\mathbf{x} \notin \Gamma$, we can apply $\nabla_\mathbf{x} = -\nabla_\mathbf{y}$ to G inside the integral, use $\Delta_\mathbf{y} G = 0$ to convert the normal derivative to a tangential derivative, and integrate by parts to obtain

$$\nabla u(\mathbf{x}) = \int_\Gamma \left(\frac{\partial}{\partial x_2}, -\frac{\partial}{\partial x_1} \right) G(\mathbf{x}-\mathbf{y}) f'(\mathbf{y}) ds_\mathbf{y}, \qquad \mathbf{x} \notin \Gamma,$$

where $f'(\mathbf{y}(s)) = (\partial/\partial s) f(\mathbf{y}(s))$. Now suppose we extend $\mathbf{t}$ and $\mathbf{n}$ from Γ to a neighborhood as orthogonal vector fields. Then

$$\mathbf{n}_\mathbf{x} \cdot \nabla \mathbf{u}(\mathbf{x}) = \int_\Gamma \mathbf{t}_\mathbf{x} \cdot \nabla_\mathbf{x} G(\mathbf{x}-\mathbf{y}) f'(\mathbf{y}) ds_\mathbf{y}, \qquad \mathbf{x} \notin \Gamma.$$

Since the tangential gradient of the single layer potential is continuous at Γ, we can now let $\mathbf{x}$ approach Γ and obtain (7.1).

Now to derive (3.6), we find the normal derivatives of Φ_i and Φ_e at Γ, applying (3.3) and (7.1) to (3.5). We find that

$$\frac{\partial \Phi_i(\mathbf{x})}{\partial \mathbf{n}_\mathbf{x}} = \int_\Gamma \frac{\partial G(\mathbf{x}-\mathbf{y})}{\partial \mathbf{t}_\mathbf{x}} \frac{\partial V_m(\mathbf{y})}{\partial \mathbf{t}_\mathbf{y}} ds_\mathbf{y} - \int_\Gamma \frac{\partial G(\mathbf{x}-\mathbf{y})}{\partial \mathbf{n}_\mathbf{x}} q(\mathbf{y}) ds_\mathbf{y} + \frac{1}{2} q(\mathbf{x}) + \mathbf{E} \cdot \mathbf{n}_\mathbf{x}$$

and $\partial\Phi_e(\mathbf{x})/\partial\mathbf{n}_\mathbf{x}$ is the same except that the term $\frac{1}{2}q$ is replaced by $-\frac{1}{2}q$. Substituting these two expressions into (2.2) and rearranging, we obtain (3.6).

References

[1] T. Ashihara, T. Yao, T. Namba, M. Ito, T. Ikeda, A. Kawase, S. Toda, T. Suzuki, M. Inagaki, M. Sugimachi, M. Kinoshita, K. Nakazawa, Electroporation in a model of cardiac defibrillation, J. Cardiovasc. Electrophysiol. 12 (12) (2001) 1393–1403.

[2] K. E. Atkinson, The Numerical Solution of Integral Equations of the Second Kind, Cambridge University Press, Cambridge, UK, 1997.

[3] J. Barnes, P. Hut, A hierarchical $O(N\log N)$ force calculation algorithm, Nature 324 (2) (1986) 446–449.

[4] J. T. Beale, A grid-based boundary integral method for elliptic problems in three-dimensions, SIAM J. Numer. Anal. 42 (2004) 599–620.

[5] J. T. Beale, M. C. Lai, A method for computing nearly singular integrals, SIAM J. Numer. Anal. 38 (6) (2001) 1902–1925.

[6] J. T. Beale, A. T. Layton, On the accuracy of finite difference methods for elliptic problems with interfaces, Comm. Appl. Math. Comput. Sci.

[7] B. A. Cipra, The best of the 20th century: Editors name top 10 algorithms, SIAM News 33 (4) (2000) 2.

[8] D. Colton, R. Kress, Integral Equation Methods in Scattering Theory, Wiley, New York, 1983.

[9] L. Greengard, V. Rokhlin, A fast algorithm for particle simulations, J. Comput. Phys. 73 (1987) 325–348.

[10] G. C. Hsiao, W. L. Wendland, Boundary Integral Equations, Springer, 2008.

[11] A. Mayo, Fast high order accurate solution of Laplace's equation on irregular regions, SIAM J. Sci. Statist. Comput. 6 (1985) 144–157.

[12] L. M. Mir, M. F. Bureau, J. Gehl, R. Rangara, D. Rouy, J. M. Caillaud, P. Delaere, D. Branellec, B. Schwartz, D. Scherman, High-efficiency gene transfer into skeletal muscle mediated by electric pulses, Proc. Natl. Acad. Sci. USA 96 (8) (1999) 4262–4267.

[13] E. Neumann, S. Kakorin, K. Toensing, Fundamentals of electroporative delivery of drugs and genes, Bioelectrochem. Bioenerg. 48 (1) (1999) 3–16.

[14] S. A. Sauter, C. Schwab, Boundary Element Methods, Springer Series in Computational Mathematics 39, 2011.

[15] G. Sersa, T. Cufer, M. Cemazar, M. Rebersek, R. Zvonimir, Electro-chemotherapy with bleomycin in the treatment of hypernephroma metastasis: case report and literature review, Tumori 86 (2) (2000) 163–165.

[16] L. Ying, G. Biros, D. Zorin, A high-order 3d boundary integral equation solver for elliptic PDEs in smooth domains, Journal of Computational Physics 219 (1) (2006) 247–275.

[17] W.-J. Ying, C. S. Henriquez, Hybrid finite element method for describing the electrical response of biological cells to applied fields, IEEE Trans. Biomed. Engrg. 54 (4) (2007) 611–620.