A Meshless, High-Order Integral Equation Method for Smooth Surfaces, with application to Biomolecular Electrostatics

by

Shih-Hsien Kuo

Submitted to the Department of
Electrical Engineering and Computer Science
in partial fulfillment of the requirements for the degree of

Doctor of Philosophy in Electrical Engineering and Computer Science

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

September 2006

Author .....
Department of Electrical Engineering and Computer Science
August 31, 2006

Certified by .....
Jacob K. White
Professor of Electrical Engineering and Computer Science
Thesis Supervisor

Certified by .....
Bruce Tidor
Professor of Biological Engineering and Computer Science
Thesis Supervisor

Accepted by .....
Arthur C. Smith
Chairman, Department Committee on Graduate Students

A Meshless, High-Order Integral Equation Method for Smooth Surfaces, with application to Biomolecular Electrostatics

by

Shih-Hsien Kuo

Submitted to the Department of Electrical Engineering and Computer Science
on August 31, 2006, in partial fulfillment of the
requirements for the degree of
Doctor of Philosophy in Electrical Engineering and Computer Science

Abstract

In this thesis, we develop methods for efficient simulation of biomolecular electrostatics based on Poisson-Boltzmann equation. Current techniques using finite-difference solution of differential formulation have many drawbacks. We present an integral formulation that resolves these difficulties and enables an efficient implementation using a recently developed fast solver. The new approach can solve practical engineering problems with good accuracy, but only with an aid of a high quality mesh generator, and sometimes require a a large number of panels to discretize a surface. To this end, a novel approach to discretize singular integral equations is proposed. Unlike the traditional boundary element method using panel discretization, the new method is meshless and capable of achieving spectral convergence: numerical errors decrease exponentially fast with increasing size of basis set. We will describe a number of techniques in our approach, including the use of global, high order basis, quadrature-based panel integration, and innovative surface representation. The biomolecular problem is particularly suited for this method because molecular surfaces are typically smooth and can be represented globally using spherical harmonics. The use of flat panels in the traditional approach would incur significant geometrical distortion, in addition to much slower convergence rate. Computational results demonstrate that for a practical problem at engineering accuracy (a tolerance of 10^{-3}) this new approach requires one to two orders of magnitude fewer unknowns than a flat panel method. For a more stringent tolerance of 10^{-6}, a comparison to an analytically solvable problem reveals that an improvement more than three orders of magnitude has been achieved.

Thesis Supervisor: Jacob K. White
Title: Professor of Electrical Engineering and Computer Science

Thesis Supervisor: Bruce Tidor
Title: Professor of Biological Engineering and Computer Science

To my parents,

Su-Fang Chen and Chao-Hsiang Kuo

Acknowledgments

First and foremost, I would like to thank Professor Jaocb White for his guidance and patience over the years. Without his willingness to take a chance in me right from the beginning and his continuing support when I was struggling with my research, this thesis would not be possible.

I am grateful to Professor Bruce Tidor for his insight in the biomolecule problem. I learnt a lot and had fun from being a TA in the first offering of the course "Foundations of Algorithms and Computational Techniques in Systems Biology" which Bruce and Jacob co-taught.

I am grateful to Professor Dennis Freeman for his constant encouragement. I can still remember the warm words he offered after my RQE (Research Qualification Exam). Incidentally, he also witnessed the last presentation I gave at MIT as a graduate student. I hope I showed some improvement.

I benefited from discussions with Professor Luca Daniel at two turning points of my research. In both occasions, his insight and suggestions helped me gain valuable perspectives on where my project might stand.

In the first part of the thesis, I worked closely with Michael Altman and Jaydeep Bardhan. Our implementation is an extension of the code Zhenhai Zhu and Ben Song wrote. Without their code, our development would have been much slower. I am also thankful to Bjarne Büchmann who was my early mentor and helped me understand the pre-corrected FFT algorithm.

I enjoy exchanging ideas with Jung Hoon Lee and Carlos Pinto Coelho about research or non-research topics. During my time in the group, I also had opportunities to interact with many other people: Thomas Klemas, Bo Kim, Bradley Bond, Xin Wang, Michal Rewienski, David Willis, Homer Reid, Joseph Kanapka, Xin Hu, Anne Vithayathil, John Rockway, Dmitry Vasilyev, Kin Cheong Sou, Lian Hing Tan, Lei Zhang and Deepak Ramaswamy, some of them I have shared office space with. Thank you for your friendship.

Last but not least, I would like to thank my parents and brother. I would not have made this far without their love, support and encouragement. This thesis is dedicated to them.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Motivation

Biomolecular structure and interactions in an aqueous environment are determined by a complicated interplay between physical and chemical forces including solvation, electrostatics, van der Waals forces, the hydrophobic effect, and covalent bonding. Electrostatic forces have received a great deal of study due to their long-range nature and the tradeoff between desolvation and interaction effects [15, 20, 22, 77]. In addition, electrostatic interactions play a significant role within a biomolecule as well as between biomolecules, making the balance between the two vital to the understanding of macromolecular systems. As a result, much effort has been devoted to accurate modeling and simulation of biomolecule electrostatics. One important application of this work is to compute the strength of electrostatic interactions for a biomolecule in an electrolyte solution, as well as the potential that the molecule generates in space. There are two valuable uses for these simulations. First, it provides a full picture of the electrostatic energetics of a biomolecular system, improving our understanding of how electrostatics contribute to stability, function, and molecular interactions [47]. Second, these simulations serve as a tool for molecular design, since electrostatic complementarity is an important feature of interacting molecules [48]. Through examination of the electrostatics and potential field generated by a protein molecule, for example, it may be possible to suggest improvements to other proteins or drug molecules that interact with it, or perhaps even design new interacting molecules *de novo* [29, 36, 37].

There are two approaches to simulating a protein macromolecule in an aqueous solution with nonzero ionic strength. Discrete, atomistic approaches based on Monto-Carlo or molecular dynamics simulations treat the macromolecule and solvent explicitly at the atomic level [11, 33, 51, 56, 66, 77]. An enormous number of solvent molecules are often required to provide reasonable accuracy, particularly when the electric fields of interest are far away from the macroscopic surface. In addition, free ions within the solvent are difficult to model with this approach. In this work, we adopt instead a mixed discrete-continuum approach based on combining a continuum description of the macromolecule and solvent with a discrete description of the atomic charges [32, 52, 77, 81, 87].

Solutions to the mixed discrete-continuum model are mostly computed numerically, using schemes based on finite-difference discretization of the model's underlying partial differential equations [20, 21, 42, 61, 67, 68]. One of the contributions of this thesis is an efficient procedure based on combining a carefully chosen integral formulation of the mixed discrete-continuum model with the pre-corrected FFT [63] fast solver algorithm. As opposed to volume discretization in the finite-difference approach, the integral formulation requires only a surface discretization and therefore enables reduction in the number of unknowns needed for an accurate solution.

The second and main contribution of this thesis is address the problem that the commonly-used discretization, a piecewise constant basis, is low order, which means memory requirement for a practical engineering problem can still be excessive. Instead, we develop a novel, high-order discretization technique that exhibits spectral convergence and does not require surface triangulation. The meshless method allows a significant reduction in problem size while preserving many advantages associated with an integral formulation.

1.2 Dissertation Outline

In the following chapter we will introduce the background information including the mixed discrete-continuum model of biomolecule electrostatics, the molecular surface definition and two standard numerical techniques that will be useful in this application: the finite-difference and boundary element methods. The finite-difference method is currently the

more popular approach of the two, but the boundary element method will be the focus of this thesis. The advantages of the latter approach are rooted in the integral formulation which will be described in Chapter 3, together with the fast solver implementation using pre-corrected FFT algorithm. The standard boundary element method uses piecewise constant basis and flat panels, so numerical errors are introduced in the form of unknown and geometrical representation respectively. In addition, the use of low order basis results in a method for which accuracy improves slowly with increasing number of unknowns. In Chapter 4, we demonstrate that accuracy can be improved with curved panel discretization, either through an analytical mapping function or a quadratic interpolating polynomial. Similarly, higher order polynomials can be used as basis functions. An isoparametric formulation using higher order polynomials of the same degree in both unknown and geometrical representation results in better accuracy as well as convergence. However, when coupled with a fast solver approach, the higher order method has a much higher cost for the same degree of freedom than the standard low order discretization. In order to develop a higher order method that is competitive in cost and superior in accuracy than the fast solver approach, we propose a spectral method in Chapter 5. The novel scheme uses numerically orthogonal polynomials defined on carefully chosen quadrature points. Unlike the approach using a more standard higher order basis, errors for our method decay exponentially with number of unknowns. In other words, improvement in accuracy accelerates with increasing degrees of freedom. In order to employ our spectral method on a practical problem, however, one needs to define local patches where good quadrature points, and associated mapping functions to the actual surface, are available. In Chapter 6, we show that recently developed spherical harmonic approaches for biomolecule surface representations result in simple and efficient implementations of the higher-order algorithm. We will describe the procedures for constructing a spherical harmonic expansion using a set of surface points and discuss some limitations of this approach. In Chapter 7, some analysis of an efficient implementation using matrix-implicit scheme is described, and finally, we conclude in Chapter 8. Computational results are given throughout the thesis in the appropriate chapters, and a discussion, if needed, is included at the end of a chapter.

Chapter 2

Problem Formulation

The application of the methods developed in this thesis focus on the simulation of biomolecular electrostatics. Although electrostatics is the only significant long-range interaction of a protein macromolecule with its aqueous environment, it is by no means the only force field present. In this chapter, we introduce the model commonly used to capture the phenomenon and address its limitations, as well as briefly describe standard numerical methods that can be used to obtain a solution. As illustrated in Figure 2-1, the system consists of a macromolecule made up of individual atoms that are covalently bonded, and a salt solution containing water molecules and charged ions. The force fields present in such a system include covalent interaction and non-bonded interaction. For the problem addressed in this thesis, however, we will not model explicitly the vibrational energies associated with covalent bonding. That is, in calculating solvation energy, we will assume the macromolecule's vibrational energies do not change between a solvated and non-solvated state. The non-bonded interaction consists of the typical columbic forces for electrostatics which we do model, and the extremely short-range van der Waals forces, which is only significant at the atomic length scale and cannot be modeled effectively with a continuum description.

In the continuum description, the biomolecule is modeled as a rigid body with fixed molecular shape. The electron cloud is modeled as a uniform dielectric medium, and positive charges are assumed fixed at atomic centers. The interior region therefore satisfies the Poisson equation with the right hand side being the potential due to point charges. The exterior region is also modeled as a uniform dielectric medium, but with a much larger

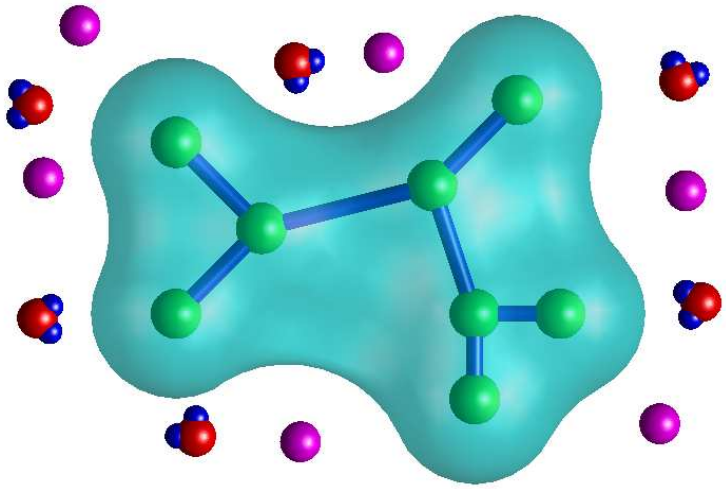


Figure 2-1: A sketch of a biomolecule in a salt solution.

dielectric constant to account for the polarization effect of water. The potential in the presence of mobile ions satisfies the Poisson-Boltzmann equation, but the salt concentration is assumed dilute enough that we can ignore the nonlinearities and use linearized Poisson-Boltzmann equation. Lastly, two boundary conditions have to be satisfied at the interface: the potentials are continuous across boundary, and the normal derivatives have a jump that is related to the relative dielectric constant.

In the following section, we describe the procedures used to generate a molecular surface which defines the interface between two dielectric mediums. The mathematical equations for the continuum formulation will be described in Section 2.2. Finally in Section 2.3, we introduce two standard numerical approaches that can be used to obtain a solution: one for the differential formulation in Section 2.2, and the other for the integral formulation that will be developed in the next chapter.

2.1 Molecular Surface

There are several surfaces biologists consider in modeling a macromolecule: the van der Waals surface, the solvent-accessible surfaces [46] and the molecular surface [14, 65]. In all of these, solid spheres are used to model individual atoms that make up a molecule and another probe sphere is used to model solvent particles so that the geometry can be represented analytically. The van der Waals surface is the surface of the union of spherical atoms of a macromolecule and is independent of the probe sphere. The solvent-accessible surface can be defined similarly as a van der Waals surface, but with radii of spherical atoms expanded by radius of a probe sphere. Alternatively, the solvent-accessible surface, as well as the molecular surface, can be defined by the trace of a rolling probe sphere as it moves around a collection of spherical atoms. The boundary defined by the motion of the probe sphere center is defined as the solvent-accessible surface, while the contiguous surface with which the probe sphere is always in contact with one or more atoms is defined as the molecular surface. The molecular surface, also known as the solvent-excluded surface, defines an interior such that solvent particles are excluded from. As illustrated in Figure 2-2, the molecular surface can be described by three analytical shapes: a spherical triangle defined by the reentrant surface of the probe sphere when it is in simultaneous contact with three atoms, part of a torus defined by the reentrant surface of the probe sphere when it is in simultaneous contact with two atoms, and part of spherical atoms where the probe sphere can come into direct contact with. The reentrant surface is defined by the inward-facing surface of the probe sphere as it is in simultaneous contact with more than one atom, and the

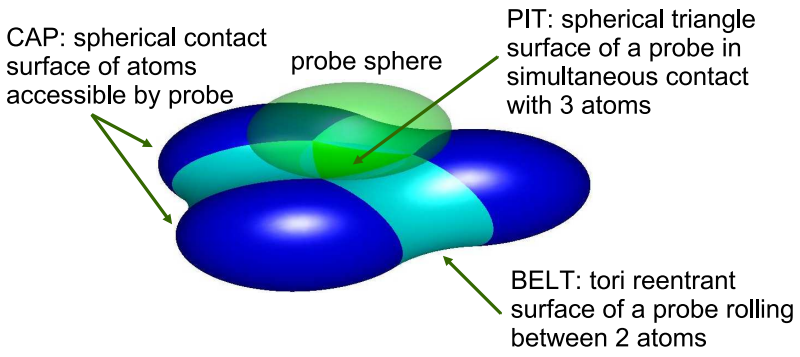


Figure 2-2: Molecular Surface.

contact surface is part of the van der Waals surface. Software program such as MSMS [76] takes inputs of atomic centers and radii, as well as probe radius to generate a triangulation of molecular surface, together with other surface properties such as area and normals.

2.2 Mixed Discrete-Continuum Model

One commonly used simplified model for biomolecule electrostatics was introduced by Tanford and Kirkwood in 1957 [81]. In this model the interior of a protein molecule is approximated as a collection of point charges in a uniform dielectric material, where the dielectric constant is typically two to four times larger than the permittivity of free space. Any surrounding solvent is modeled as a much higher permittivity electrolyte whose behavior is described by the Debye-Hückel theory. The interface between the protein and the solvent is defined by determining how close the solvent molecules can approach the biomolecule [14, 65].

The Tanford and Kirkwood model for a single protein in a solvent is depicted in Figure 2-3, which is the continuum description of the physical system in Figure 2-1, with Region I corresponding to the interior of the protein and Region II corresponding to the surrounding solvent. The electrostatic behavior in Region I is governed by a Poisson equation:

$$\nabla^2 \varphi_1(\vec{r}) = -\sum_{i=1}^{n_c} \frac{q_i}{\varepsilon_1} \delta(\vec{r} - \vec{r}_i) \quad \text{(Region I)} \tag{2.1}$$

where φ_1 is the electrostatic potential, $\vec{r}$ is an evaluation position, $\vec{r}_i$ is the location of the i^{th} protein point charge, q_i is the point charge strength, n_c is the number of point charges, and ε_1 is the dielectric constant in the protein interior. Note also that δ is the standard Dirac-Delta function.

To determine the electrostatic potential in the solvent, Debye-Hückel theory suggests that the electrostatic potential should satisfy a nonlinear Poisson-Boltzmann equation. However, the nonlinearity generates an unnecessarily complicated model when the solution is dilute and charge density within the macromolecule is not extremely high. Instead, the simpler linearized Poisson-Boltzmann equation, which is also a Helmholtz equation, is more

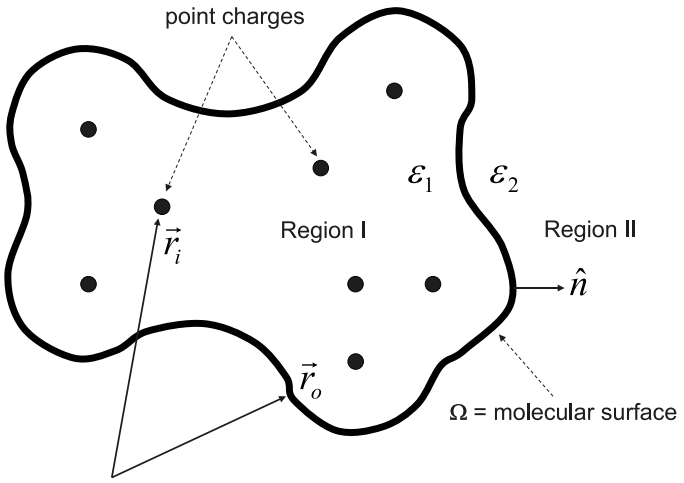


Figure 2-3: The continuum model of a solvated protein.

commonly used, and has been shown to accurately predict biomolecular properties under a variety of conditions [20, 42]. Therefore, the electrostatic potential in the solvent, Region II of Figure 2-3, is presumed to satisfy the Helmholtz equation:

$$\nabla^2\varphi_2(\vec{r}) - \kappa^2\varphi_2(\vec{r}) = 0 \quad \text{(Region II)} \tag{2.2}$$

where

$$\kappa = \sqrt{\frac{8\pi e^2 I}{\varepsilon_2 k_B T}} \tag{2.3}$$

is the inverse Debye screening length, relating to the ionic strength of solvent I, exterior dielectric constant ε_2, Boltzmann constant k_B and the absolute temperature T. For an ionic strength of 0.145 M, and an inner and outer dielectric constants of 4 and 80 respectively, $\kappa = 0.124$ Å^{-1} at room temperature. A typical protein macromolecule consists of hundreds of atoms and measure about 50 Å in size, and a probe radius of 1.4 Å is typically used to model water molecules in solvent. In comparison, the Debye length is about the thickness of a few layers of water molecules.

In addition to regions I and II, sometimes a third region known as the Stern layer is

also used to account for the difference in size between salt ions and water molecules in solvent. It is a layer exterior to the interface in Figure 2-3 within which exterior dielectric constant ε_2 and interior equation (2.1) are used to model the presence of water molecules but absence of salt ions. A thickness of 2 Å is typically used for the Stern layer.

2.3 Numerical Solution

2.3.1 Finite-Difference Approach

A wide variety of numerical techniques can be used to compute solutions to the combination of (2.1) and (2.2). For the biomolecule application, the most commonly used approach is based on the finite-difference method for discretizing partial differential equations, with researchers frequently making use of the DelPhi software package [20, 21, 42, 61, 67, 68]. In this approach, a three-dimensional grid that contains a molecule is used as a computational domain. The differential equations in (2.1) and (2.2) are approximated with algebraic equations using finite differences. A dielectric constant is associated with each grid point depending on its relative location to a molecular interface. Point charges, which usually do not fall exactly on a grid, have to be represented using nearby grid points. The boundary condition at infinity is approximated by the border of a truncated grid. The potential at every grid point can then be computed using a variety of iterative technique [74]. To compensate for the approximated boundary condition at infinity and regularity of a grid, techniques such as focussing and rotational averaging [21] have been used.

Although finite-difference methods have proven to be effective, there are several characteristics of the biomolecule application which are problematic for such methods. As shown in Figure 2-4, inaccuracies can be generated when projecting the discrete charges, which appear in (2.1), on to finite-difference grids. The problem is particularly troublesome when attempting to compute reaction forces at those point charge locations [31]. In addition, the large jump in dielectric constant across the irregularly-shaped protein-solvent boundary must be treated carefully. Finally, the solvent region is unbounded, at least formally, and must be somehow truncated before applying a finite-difference method. Modifications of

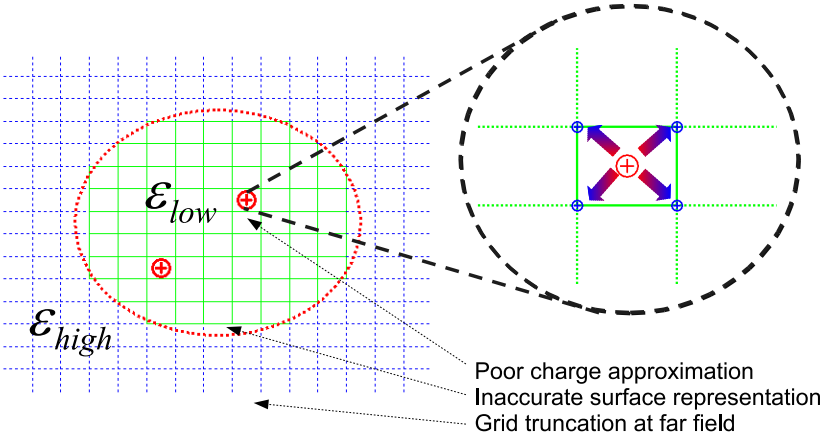


Figure 2-4: Inaccuracies associated with finite-difference solution.

the basic finite-difference method have been developed to resolve many of these difficulties [21, 42, 68, 69, 92], though often at considerable computational cost.

2.3.2 Boundary Element Method

The boundary element method [27, 30] has become a standard method in solving three-dimensional engineering problems. In this section, a brief review is presented so that we can emphasize some aspects of the standard approach our method can improve upon. The following integral equation will be used as our model problem:

$$\phi(\vec{r}) = \int_{\Omega} G(\vec{r},\vec{r}')\sigma(\vec{r}')\,dS' \tag{2.4}$$

where Ω is the surface boundary of a three-dimensional region of interest on which we would like to solve for the unknown quantity σ given an arbitrary ϕ, assumed given anywhere on the same surface. The kernel $G(\vec{r},\vec{r}')$, also known as the Green's function, is specified for a particular physical problem that one is modelling. For the problems [44, 64, 94]

we are interested in, G can take the following forms:

$$G(\vec{r},\vec{r}') = \frac{1}{r}, \frac{e^{ikr}}{r}, \frac{e^{kr}}{r} \qquad k \in \mathbf{R} \tag{2.5}$$

where $r = |\vec{r} - \vec{r}'|$ is the Euclidean distance between a source and target point.

Discretization

In order to numerically solve for σ, two representations are typically used: one for the surface geometry, and the other for the solution itself. A triangular mesh is commonly used to discretize the geometry, and a basis set, $\{B_i : i = 1,2,\ldots,n\}$, is usually defined on the same mesh, with B_i's being non-zero only on a few triangles. The basis is used to discretize the unknown as in

$$\sigma(\vec{r}) = \sum_{i=1}^{n} a_i B_i(\vec{r}) \qquad \vec{r} \in \Omega^{mesh} \tag{2.6}$$

where Ω^{mesh} may be an exact or approximate geometry of the original domain Ω. The use of panels is so common that boundary element methods are sometimes referred to as panel methods. With the representation in (2.6), equation (2.4) can be written as:

$$\phi(\vec{r}) = \sum_{i=1}^{n} a_i \int_{support} G(\vec{r},\vec{r}') B_i(\vec{r}')\, dS' \tag{2.7}$$

$$= \sum_{i=1}^{n} a_i \sum_{\substack{all\ panels \\ in\ B_i's\ support}} \int_{panel} G(\vec{r},\vec{r}') B_i(\vec{r}')\, dS', \tag{2.8}$$

where in (2.7), integration is over support of ith basis and in (2.8), integration is over individual panels within B_i's support. Once the weighting coefficients a_i's are determined, the solution on Ω^{mesh} can be written as a weighted sum of individual basis functions.

Galerkin and Collocation Formulation

In order to solve for the set of coefficients a_i's from equation (2.7), a Galerkin or collocation scheme is commonly used, both of which belong to a general class of methods known as weighted residual minimization methods. The residual measures numerical error intro-

duced by basis function discretization. In the Galerkin methods, orthogonality is enforced between the residual and the basis function as in

$$\int_{support} B_j(\vec{r}) \underbrace{\left(\phi(\vec{r}) - \sum_{i=1}^{n} a_i \int_{support} G(\vec{r},\vec{r}')B_i(\vec{r}')dS' \right)}_{residual} dS = 0 \tag{2.9}$$

for $j = 1,2,\ldots,n$. This is the projection approach since the residual is projected to a space orthogonal to the one spanned by the basis. In other words, the resultant solution is the best one representable by the set of basis. From (2.9), the unknown coefficients can be solved from a $n \times n$ matrix equation:

$$\sum_{i=1}^{n} a_i \underbrace{\int_{support} B_j(\vec{r}) \int_{support} G(\vec{r},\vec{r}')B_i(\vec{r}')\, dS' dS}_{A_{ji}^{Galerkin}} = \int_{support} B_j(\vec{r})\phi(\vec{r})\, dS \tag{2.10}$$

where each entry in the Galerkin matrix involves two integrations over the support of a basis function: one for the source basis function, and the other for the test basis function. Alternatively, one may choose to minimize the residual at n collocation points:

$$\sum_{i=1}^{n} a_i \underbrace{\int_{support} G(\vec{r}_j,\vec{r}')B_i(\vec{r}')\, dS'}_{A_{ji}^{Collocation}} = \phi(\vec{r}_j) \tag{2.11}$$

where $\vec{r}_j$'s are their position vectors. This is known as the collocation formulation, or the interpolation approach, since residual is exactly zero at those points. In contrast to the Galerkin approach, each entry in the collocation matrix involves only one integration over the support of a source basis function. The outer integral of the Galerkin formulation is typically done with quadrature, therefore it is computationally more expensive. Although convergence theory is better developed for the Galerkin approach [2], in practice collocation is often used for its simplicity and efficiency.

Chapter 3

Fast Solver Approach

When boundary element methods [27, 30] are used to solve Laplace or Helmholtz problems associated with complicated three-dimensional geometries, the associated integral equation is typically discretized using a piecewise constant basis, and a system of equations is generated using either a Galerkin or a collocation scheme. The resulting matrix is then solved iteratively using acceleration [7, 23, 25, 39, 59, 63, 70]. This approach has become the method of choice for exterior problems and is used in diverse applications such as interconnect extraction [94], MEMS and fluidic simulation [64, 84].

In calculating bimolecular solvation energy, however, methods based on finite-difference approach have been more popular. As this chapter will make clear, numerical methods based on solving an integral formulation of (2.1) and (2.2) can treat point charges, irregularly shaped regions with large jumps in parameters, unbounded domains, and the reaction force computation much more naturally than finite-difference methods. For this reason, a number of researchers have developed integral formulations [8, 9, 35, 44, 88, 89, 91]. However, even though integral formulations have many advantages for this application, they are not often used. The available numerical techniques for solving integral equations were too computationally expensive to use on complicated problems, but recently developed fast algorithms have changed that situation considerably. Earlier works [8, 9, 91] have been based on fast multipole methods [23, 70] for acceleration. In this chapter, we present an alternative approach using pre-corrected FFT algorithm [63] which is particularly well suited for problems with multiple domains governed by various Green's functions.

In the following section we will describe an integral formulation for the biomolecular electrostatic problem, and in Section 3.2 we will describe the standard discretization using piecewise-constant basis on flat panels. In Section 3.3, we introduce the fast numerical technique based on pre-corrected FFT algorithm for computing the integral equation solutions. An appropriate preconditioner that can be used in an iterative solver is discussed in Section 3.4. Computational results are presented in Section 3.5 and finally, we conclude with a discussion in Section 3.6.

3.1 Integral Equation Formulation

To begin the formulation derivation, first consider that the well-known fundamental solutions to (2.1) and (2.2) are, respectively,

$$G_1(\vec{r};\vec{r}') = \frac{1}{4\pi|\vec{r}-\vec{r}'|} \tag{3.1}$$

$$G_2(\vec{r};\vec{r}') = \frac{e^{-\kappa|\vec{r}-\vec{r}'|}}{4\pi|\vec{r}-\vec{r}'|}. \tag{3.2}$$

The two fundamental solutions can be combined with Green's second theorem to generate an integral equations for the potential and its normal derivative. In particular in Figure 2-3, the integral equation for Region I is

$$\varphi_1(\vec{r}) = \int_\Omega \left[G_1(\vec{r};\vec{r}')\frac{\partial\varphi_1}{\partial n}(\vec{r}') - \varphi_1(\vec{r}')\frac{\partial G_1}{\partial n}(\vec{r};\vec{r}') \right] d\vec{r}' + \sum_{i=1}^{n_c} \frac{q_i}{\varepsilon_1} G_1(\vec{r};\vec{r}_i), \tag{3.3}$$

and the equation for Region II is

$$\varphi_2(\vec{r}) = \int_\Omega \left[-G_2(\vec{r};\vec{r}')\frac{\partial\varphi_2}{\partial n}(\vec{r}') + \varphi_2(\vec{r}')\frac{\partial G_2}{\partial n}(\vec{r};\vec{r}') \right] d\vec{r}', \tag{3.4}$$

where $\vec{n}$ is the outward pointing normal at the interface, and the domain of integration for the integrals, Ω, is the boundary surface separating the low permittivity protein interior from the high permittivity solvent.

The potentials φ_1 and φ_2 must satisfy a pair of matching conditions on the boundary

surface Ω. In particular, the electric potential is continuous and the normal derivative of the potential jumps by an amount related to the ratio of the dielectric constants,

$$\varphi_1(\vec{r}_o) = \varphi_2(\vec{r}_o) \tag{3.5}$$

$$\frac{\partial\varphi_1}{\partial n}(\vec{r}_o) = \varepsilon\frac{\partial\varphi_2}{\partial n}(\vec{r}_o), \tag{3.6}$$

where $\vec{r}_o \in \Omega$, and $\varepsilon = \varepsilon_2/\varepsilon_1$ is the relative dielectric constant of the two regions. To enforce these matching boundary conditions, take the limit of equation (3.3) as $\vec{r} \to \Omega$ from the inside, and use the limit of equation (3.4) as $\vec{r} \to \Omega$ from the outside. In this limit, G_1, G_2, $\frac{\partial G_1}{\partial n}$, and $\frac{\partial G_2}{\partial n}$ are kernels with integrable singularities, so care must be taken in carrying out the integrations. Note that that the potential due to a monopole layer is continuous across the layer, while the potential due to a dipole layer is discontinuous across the layer [79]. The results generated by applying the limiting processes to (3.3) and (3.4) yields

$$\begin{aligned}\varphi_1(\vec{r}_o) &= \lim_{\vec{r}\to\vec{r}_o}\varphi_1(\vec{r}) \\ &= \int_\Omega\left[G_1(\vec{r}_o;\vec{r}')\frac{\partial\varphi_1}{\partial n}(\vec{r}') - \varphi_1(\vec{r}')\frac{\partial G_1}{\partial n}(\vec{r}_o;\vec{r}')\right]d\vec{r}' + \frac{1}{2}\varphi_1(\vec{r}_o) \\ &\quad + \sum_{i=1}^{n_c}\frac{q_i}{\varepsilon_1}G_1(\vec{r}_o;\vec{r}_i)\end{aligned} \tag{3.7}$$

and

$$\begin{aligned}\varphi_2(\vec{r}_o) &= \lim_{\vec{r}\to\vec{r}_o}\varphi_2(\vec{r}) \\ &= \int_\Omega\left[-G_2(\vec{r}_o;\vec{r}')\frac{\partial\varphi_2}{\partial n}(\vec{r}') + \varphi_2(\vec{r}')\frac{\partial G_2}{\partial n}(\vec{r}_o;\vec{r}')\right]d\vec{r}' + \frac{1}{2}\varphi_2(\vec{r}_o)\end{aligned} \tag{3.8}$$

where $\vec{r}_o$ is the position vector of some point on the boundary Ω and the integrals are taken to be principal value integrals.

Substituting equations (3.5) and (3.6) into (3.7) and (3.8) yields a coupled pair of integral equations for φ_1 and $\frac{\partial\varphi_1}{\partial n}$ on the dielectric interface,

$$\frac{1}{2}\varphi_1(\vec{r}_o) + \int_\Omega\left[\varphi_1(\vec{r}')\frac{\partial G_1}{\partial n}(\vec{r}_o;\vec{r}') - G_1(\vec{r}_o;\vec{r}')\frac{\partial\varphi_1}{\partial n}(\vec{r}')\right]d\vec{r}' = \sum_{i=1}^{n_c}\frac{q_i}{\varepsilon_1}G_1(\vec{r}_o;\vec{r}_i), \tag{3.9}$$

and

$$\frac{1}{2}\varphi_1(\vec{r}_o)+\int_\Omega\left[-\varphi_1(\vec{r}')\frac{\partial G_2}{\partial n}(\vec{r}_o;\vec{r}')+G_2(\vec{r}_o;\vec{r}')\frac{1}{\varepsilon}\frac{\partial\varphi_1}{\partial n}(\vec{r}')\right]d\vec{r}'=0. \tag{3.10}$$

Equations (3.9) and (3.10) can be used to compute φ_1 and $\frac{\partial\varphi_1}{\partial n}$ on Ω. Then those surface potentials and their normal derivatives can be used in (3.3), (3.4), (3.7), and (3.8) to compute the potentials anywhere. Therefore, to compute the reaction potentials at the charge locations, which are needed to determine energy changes, one need only evaluate

$$\varphi^{REAC}(\vec{r}_i)=\int_\Omega\left[G_1(\vec{r}_i;\vec{r}')\frac{\partial\varphi_1}{\partial n}(\vec{r}')-\varphi_1(\vec{r}')\frac{\partial G_1}{\partial n}(\vec{r}_i;\vec{r}')\right]d\vec{r}'. \tag{3.11}$$

3.2 Discretization

A standard piecewise-constant centroid-collocation scheme is used to discretize (3.9) and (3.10). In the piecewise constant collocation method, the surface is first discretized into a set of panels, and a piecewise constant basis function, B_k, is associated with each panel. Then, the potentials are represented as a weighted combination of the panel basis functions. That is,

$$\varphi_1(\vec{r}_o) \approx \sum_k a_k B_k(\vec{r}_o) \tag{3.12}$$

$$\frac{\partial\varphi_1}{\partial n}(\vec{r}_o) \approx \sum_k b_k B_k(\vec{r}_o) \tag{3.13}$$

where k is the panel index, and a_k and b_k are weights of individual basis functions.

The basis function weights are determined by insisting that when (3.12) and (3.13) are substituted for the potential and its normal derivative in (3.9) and (3.10), the resulting equations are exactly satisfied for those values of $\vec{r}_o$ which correspond to panel centroids. The resulting system of equations can be denoted as a matrix of the form

$$\begin{bmatrix} \frac{1}{2}I+\int_{panel_k}\frac{\partial G_1}{\partial n}d\vec{r} & -\int_{panel_k}G_1 d\vec{r} \\ \frac{1}{2}I-\int_{panel_k}\frac{\partial G_2}{\partial n}d\vec{r} & \frac{1}{\varepsilon}\int_{panel_k}G_2 d\vec{r} \end{bmatrix}\begin{bmatrix} a_k \\ b_k \end{bmatrix}=\begin{bmatrix} \sum_{i=1}^{n_c}\frac{q_i}{\varepsilon_1}G_1 \\ 0 \end{bmatrix} \tag{3.14}$$

where n_c is the total number of charges inside the protein and $\int_{panel_k}$ corresponds to an integration over the kth panel surface. Note that the matrix is only a function of protein geometry and is independent of charge magnitudes, making it possible to construct the matrix operator once and use it repeatedly to solve for optimal charge configurations [3].

3.3 Precorrected-FFT Acceleration

Although the matrix equation in (3.14) can be readily solved with Gaussian elimination, and is used for the smaller test cases to demonstrate the validity of this formulation and to examine convergence properties, Gaussian elimination is too computationally expensive to solve practical examples of interest. An alternative approach to Gaussian elimination is to use an iterative solver such as GMRES [75], and recent advances in fast algorithms have made this approach very appealing. Most of these fast methods take advantage of the fact that an iterative solver is a matrix-implicit algorithm. No explicit matrix has to be formed or stored; only the calculation of matrix-vector products is required. An existing precorrected-FFT algorithm [63] is particularly well suited for this problem and will be described here.

As demonstrated in the above formulation, the boundary element method often involves the solution of an integral equation of the following form:

$$\varphi(\vec{r}) = \int K(\vec{r};\vec{r}')\sigma(\vec{r}')d\vec{r}', \tag{3.15}$$

where $K(\vec{r};\vec{r}')$ is a known kernel. Given a potential distribution $\varphi(\vec{r})$, one desires to find the corresponding charge distribution $\sigma(\vec{r})$. In the context of matrix-implicit iterative methods, what is important is the ability to efficiently compute the potential distribution for some charge distribution $\sigma(\vec{r})$. Although charge-potential terminology has been used here, this is for illustration purposes only; they can be any general variables, such as those in the matrix equation (3.14), and the kernel $K(\vec{r};\vec{r}')$ does not have to be the usual $\frac{1}{|\vec{r}-\vec{r}'|}$ implied by the charge-potential relationship.

The biomolecule electrostatic model has two integral equations with different kernels,

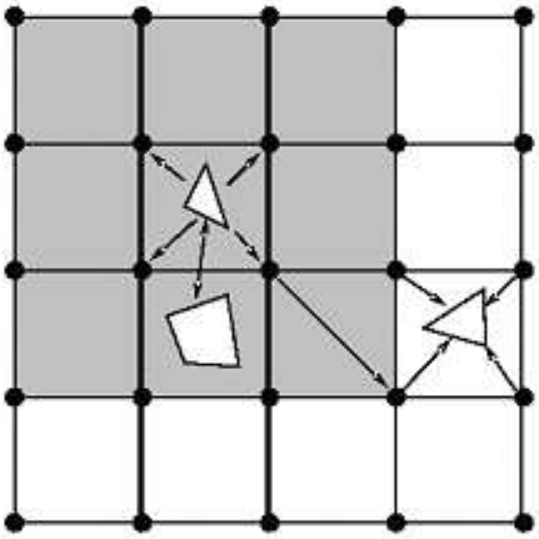


Figure 3-1: A pictorial representation of the precorrected FFT algorithm (image courtesy of J. Phillips)

and therefore the fast method for computing matrix-vector products must be kernel independent. Kernel independence is a key feature of the precorrected-FFT algorithm, and it is a property not shared with the more commonly used versions of the fast multipole method [23, 24, 59].

The algorithm can be summarized in four steps, as shown in Figure 3-1, where a given set of panels from a discretized surface are superimposed on a uniform grid. First, panel charges are projected onto their associated grid points, in what is called the projection step. Second, given the distribution of grid charges, the grid potential can be calculated using a convolution of the Green's function (the kernel) and the grid charges; this convolution is efficiently computed using the fast Fourier transform (FFT). Third, grid potentials are interpolated back onto the panels, a step known as interpolation. In the fourth step, called precorrection, nearby interactions are computed directly, with a correction factor that removes the contributions from the grid. All four steps—projection, interpolation, FFT convolution, and precorrection—possess sparse representations, so the algorithm is very efficient in both speed (roughly $O(n \log n)$) and memory (roughly $O(n)$), where n is the number of panels. This is a tremendous improvement over traditional methods for discretizing the integral equations, which generate dense matrices and therefore require n^2 memory and n^2

operations for matrix-vector multiplication.

3.4 Preconditioner

As discussed in the previous section, an iterative solver combined with fast matrix-vector multiplication techniques is more efficient than a direct solver using Gaussian elimination or an iterative solver without matrix sparsification. To take full advantage of an iterative solver, however, it is important to keep the number of iteration small compared to the size of a problem so that the overall complexity is still $O(n \log n)$. A good preconditioner which is cheap to factor (thus incurring minimal overhead cost) and resembles the original matrix (thus improving matrix condition) is therefore essential for an iterative solver. Fortunately for the matrix equation in (3.14), it is not very difficult to devise an effective preconditioner. Since all elements in the 2×2 block diagonals correspond to self-term single or double layer potentials, they are the largest in magnitude in the corresponding columns. So a preconditioner based on the four block diagonal entries in (3.14):

$$P = \begin{bmatrix} \ddots & \ddots \\ \ddots & \ddots \end{bmatrix} \tag{3.16}$$

is used in an iterative solver which is sparse and cheap to factor.

3.5 Computational Results

3.5.1 Analytical Reaction Potential of a Spherical Molecule

A spherical molecule of radius 1 Å, in aqueous salt solution, with a single charge located at various radial distances, was simulated. A dielectric constant of 1 was used inside the molecule, and a dielectric constant of 20 was used externally; $\kappa = 3\text{Å}^{-1}$ in this example. The reaction potential calculated at the charge location was compared with the analytic result [41] for three cases, at radial distances r_c of 0 Å, 0.5 Å, and 0.9 Å, as shown in Figure 3-2. As the charge moved closer to the molecular surface, the relative error also

increased. All three cases exhibited reasonable convergence properties as the discretization was refined.

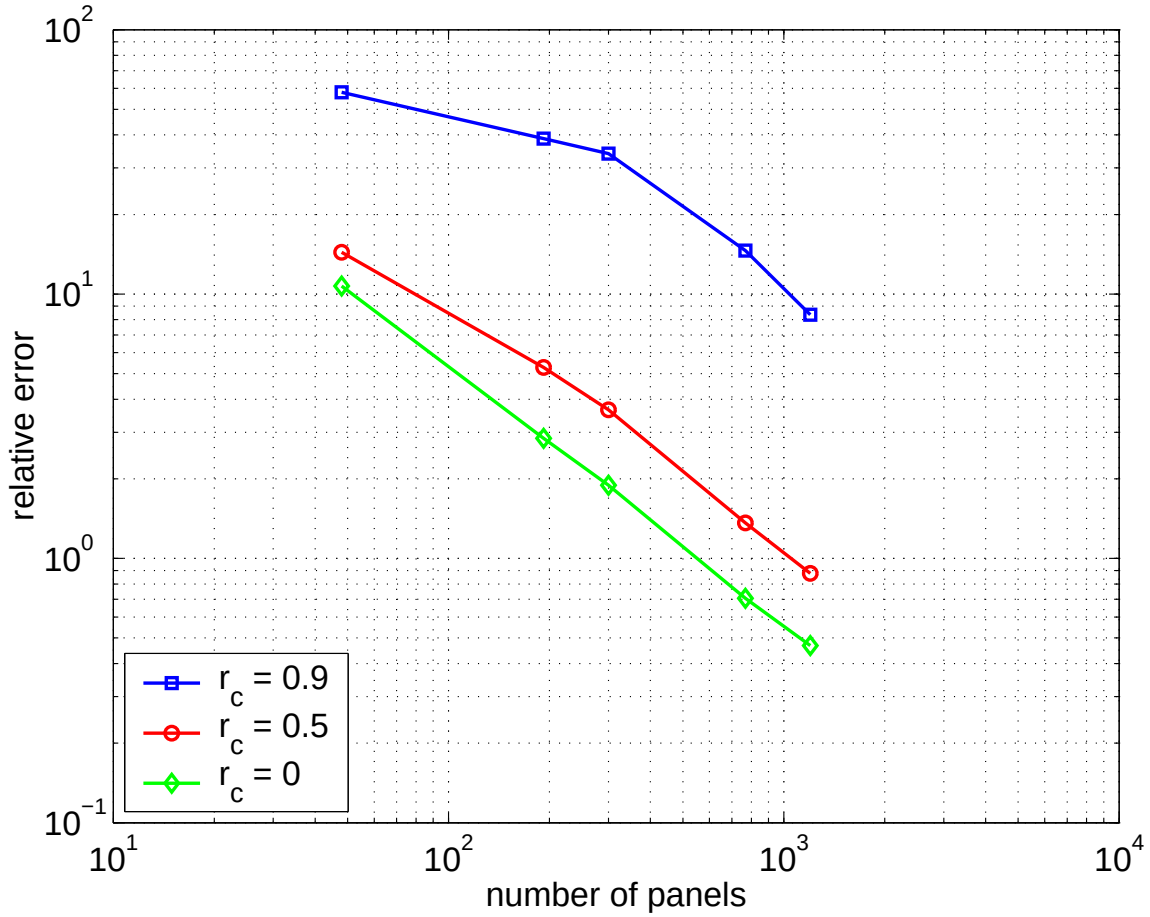


Figure 3-2: Convergence of the reaction potential of a spherical molecule to the analytical result as the discretization is refined.

The number of iterations required to reach convergence with pre-corrected FFT acceleration is shown in Table 3.1, for two charge locations, at $r_c = 0$ Å and $r_c = 0.9$ Å. Although no preconditioner was used in these test cases, GMRES converged reasonably quickly and the iteration count remained fairly constant as the number of panels increased. The conditioning of this formulation is evident and adoption of a preconditioner will further improve performance.

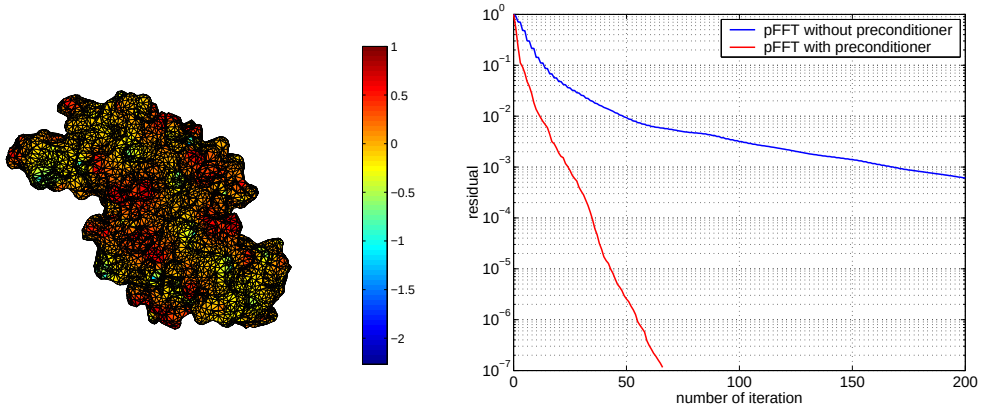
Number of Surface Panels	GMRES iteration count	
	$r_c = 0.0$ Å	$r_c = 0.9$ Å
192	8	25
972	19	43
4800	16	44
6912	16	46
10800	18	48

Table 3.1: GMRES convergence of pFFT-accelerated implementation

3.5.2 Comparison to a Finite-Difference Solver

Three realistic examples including a water molecule (H_2O), an organic molecule (QSI) and an *E. coli* chorismate mutase (ECM) protein macromolecule are used to verify the numerical results of the integral formulation obtained with a fast solver. The geometry of water used is based on the TIP3P model [34]. The geometry of the ECM molecule was taken directly from an X-ray crystal structure [45], and can be obtained from the Protein Data Bank (PDB) [5] as accession number 1ECM. The molecular surface triangulation were obtained with the program MSMS [76] using a probe radius of 1.4 Å. An ionic strength of 0.145 M was used, equivalent to $\kappa = 0.124$ Å^{-1} at 25° C. A dielectric constant of 4 was used for the interior of a molecule, and a dielectric constant of 80 was used externally. A Stern layer of 2 Å was also used in all three cases. Figure 3-3(a) shows a sample solution of the potential distribution on the ECM molecular surface using a relatively coarse mesh of 21221 panels. As indicated in Figure 3-3(b), the use of a preconditioner like (3.16) greatly reduces the number of iteration required to reach convergence at some prescribed tolerance.

The solvation free energy, which is simply one half of the inner product of the charge values with the vector of the potentials at the charge points, is compared with those obtained from the finite-difference solver DelPhi [20, 21, 42, 61, 67, 68]. DelPhi is a popular finite-difference scheme based simulation tool for solving the linearized Poisson-Boltzmann equation, and is used both in academic and industry settings. Table 3.2 compares the results for the three molecules in solvent described previously. The number of discretization panels for the dielectric interface, and the salt interface (i.e., the Stern layer) are listed for the pre-corrected FFT solver. The discretization used in DelPhi was 257 grids per Angstrom. The two solvers agree to within 1%.



(a) Surface potential (b) Improving matrix condition

Figure 3-3: Numerical solution of an ECM macromolecule obtained with pre-corrected FFT implementation.

Protein			$E_{solvation}$ (kcal/mol)	
	# of dielectric panels	# of salt panels	pFFT	DelPhi
H_2O	17204	9330	-3.14	-3.17
QSI	34114	5842	-34.62	-34.75
ECM	82868	18596	-646.42	-653.88

Table 3.2: Solvation free energies calculated by pFFT-accelerated solver and DelPhi.

3.5.3 Electrostatic Binding Energy of Protein Macromolecules

One practical use of the integral equation solver (as well as the finite-difference solver DelPhi) is in the calculation of binding energy between two protein macromolecules, which may enable one to design a drug molecule with specific charge configuration for a target receptor. The well-known barnase-barstar complex is used as an illustrative example. To compute the binding energy, one needs to compute the solvation free energies of three configurations: two for each of the binding molecules and one for the compound. The binding energy can then be taken as a sum of the three energies, taking appropriate sign for each term. Table 3.3 shows the binding energy as well as the constituent solvation energies for the barnase-barstar complex.

Protein				Energy Calculated (kcal/mol)	
	# atoms	# of dielectric panels	# of salt panels		
Barnase	1107	43298	21284	$E_{desolvation}$	51.06
Barstar	839	35978	17434	$E_{desolvation}$	40.11
Barnase-Barstar Complex	1946	68592	31728	$E_{interaction}$	-82.65
				$E_{binding}$	8.53

Table 3.3: Solvation free energies calculated by pFFT solver.

3.6 Discussion

In this chapter we presented an integral-equation based approach for computing numerical solutions to the mixed discrete-continuum model of biomolecule electrostatics. The new approach combines a carefully chosen integral formulation of the mixed discrete-continuum model with a kernel-independent precorrected-FFT accelerated integral equation solver. Computational results from our new approach, on both simple and more complicated geometries, were compared to analytic results and to the widely used finite-difference based DelPhi program. The results are encouraging and indicate a potential application of this formulation. On the other hand, accuracy and convergence of an integral equation solver relies on the availability of a good quality mesh. Unfortunately, software tools for molecular surface triangulation are limited. The MSMS program we used is mainly for visualization purpose and some post-processing may be needed to prune badly-formed triangles. In addition, the number of triangles required to faithfully represent the geometry of a complicated macromolecule may be very high, therefore an integral equation solver can be more computationally expensive than a finite-difference based solver like DelPhi.

Nevertheless, there are fundamental advantages associated with an integral formulation compared to a differential formulation as shown in Table 3.4. The need to only discretize a two-dimensional surface in the integral formulation compared to a three-dimensional volume in the differential formulation has ramifications in both accuracy and efficiency. In addition, the use of fundamental solutions as Green's functions ensure boundary conditions at infinity are satisfied in the integral formulation. The forcing term in the right hand side of integral formulation uses exact representation of points charges, while in the differential

	Boundary Element	**Finite Difference**
Boundary Condition at Interface	Approximation with surface Mesh	Approximation with volume grid
Boundary Condition at Infinity	Exact	Approximation with truncated grid
Point Charge Representation	Exact	Approximation with nearby grid points

Table 3.4: Comparison between integral and differential formulation of biomolecular electrostatic problem.

formulation they are approximated with projected grid charges. In order to take advantage of these attractive features with better computational cost than current implementation, there is a need to develop novel geometry representation as well as higher order methods so that the size of problem can be kept small for a given accuracy. In Chapters 5 and 6, we will propose a highly accurate approach to discretize the integral formulation developed in this chapter. But let's first look at the standard approach to improve convergence in the next chapter.

Chapter 4

Higher Order Method

In the fast solver implementation in Chapter 3, discretization with a piecewise constant basis on flat panels are used. The discretization with a triangular mesh has many advantages: first, accuracy can be controlled by tuning the number of panels, perhaps adaptively, without modifying the general representation. Second, the basis functions are kept fairly linearly independent by restricting support to one or a few panels (if piecewise linear basis are used). Finally, the formidable task of integrating over an arbitrary boundary Ω is broken down into integrating over a flat triangle. Panel integration is still difficult however, because simple quadrature schemes do not work well for self or nearby terms. The existence of analytical formulae [30, 60] for integrating products of $\frac{1}{r}$ with polynomials over flat triangles allows a relatively straightforward implementation of the boundary element method. On the other hand, no such formula is available for $\frac{e^{ikr}}{r}$ or $\frac{e^{kr}}{r}$ kernels. And if curved panels are to be used for surface representation, the task of panel integration becomes even more difficult.

For problems with planar boundaries, the process of meshing does not introduce significant geometrical discretization error, and most numerical errors are associated with how well a solution can be represented by a chosen set of basis. In the case of piecewise constant basis, any smooth function other than constant cannot be represented exactly. Although accuracy improves with number of panels, its convergence is limited to the lowest order. Many engineering problems, however, have curvilinear boundaries, and discretization with flat panels introduces significant approximation in the geometry. While the mesh

converges to the actual surface in the same manner as piecewise constant basis, in practice a large number of panels are needed to achieve good accuracy. And the use of higher order basis alone will not improve convergence without adopting simultaneously a better representation of the curved surface.

In this chapter, we address this inefficiency and demonstrate better accuracy and faster convergence when curved panels are used together with higher order basis. Higher order panels using B-splines [53] or Taylor series expansion of local surface curvature [72] have been used. While analytical expressions of singular integrals are used in both cases, such formulae are not always possible for curved surfaces that cannot be described by polynomials. In addition, such formulate are cumbersome to derive and may not generalize to polynomials of all orders: that is, derivation has to be carried out repeatedly for all possible degrees of polynomials involved in the geometry and basis function representation. We take an alternative approach to represent the geometry with mapping functions and numerically evaluate the integrals. This procedure can be generalized to any curvilinear surface if a mapping function from a local reference panel can be found. In the case when exact geometrical representation is not possible, an isoparametric approximation of a mapping function in terms of polynomial basis functions can also be used, as will be described in Section 4.3. While this can be a general approach to achieve better accuracy and faster convergence in the standard boundary element method implementation, more importantly the techniques we introduce here can also be applied to the spectral method that will be discussed in later Chapters 5 and 6.

In the following section, we give a brief survey on the common techniques of panel integration and describe our approach. The particular technique we have chosen enables the implementation of higher order representation on curved panels and basis functions in Section 4.2. In cases where exact geometrical representation is not feasible, we describe in Section 4.3 a general approach based on isoparametric formulation. In Section 4.4, computational results on a sphere are shown and we conclude in Section 4.5 with a discussion.

4.1 Panel Integration

In the standard boundary element method implementation, one has to integrate a singular Green's function over some surface in forming a matrix or computing a matrix-vector product. This is one main reason for the common use of flat panels in representing a curved surface as it simplifies the task in a consistent manner to integrating over a flat triangle. The procedure, commonly known as panel integration, is a crucial step in the formulation and much research [17, 26, 30, 40, 60, 82, 93] has been devoted to the accurate and efficient evaluation of the integral. If non-constant basis are used, the integrand will also include a basis function, usually of polynomial form, according to (2.8). The main difficulty arises from the singularity of the Green's function integrand because straightforward quadrature techniques are either too computationally costly or insufficiently accurate. Therefore, various techniques have been devised to carry out the integration analytically [30, 60] or semi-analytically [93]. Other approaches include specialized quadrature rules for singular integrals [71, 80] or removal of singularity by variable transformation before applying standard quadratures [17, 26, 40, 82].

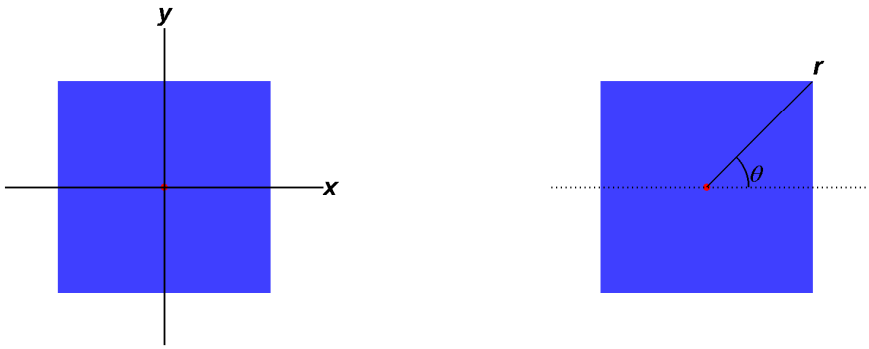
While various approaches are available for integration over flat triangles, such techniques are much more limited for curved panels, partly because a curved surface can have arbitrarily complicated geometry and is very hard to generalize. Table 4.1 gives a few examples of panel integration techniques appropriate for the Green's functions of interest. The analytical and semi-analytical formulae [30, 60, 93] used in the fast solver implementation in Chapter 3 are not available to curved panels. While an approach [86] has been proposed to generalize these formulae to approximate integration over a curved panel, it is most effective for constant basis functions and does not extend to double layer potential. Unless one uses polynomial approximation of curve panels as in [53], panel integration for an arbitrary curved panel would most likely have to be based on numerical quadrature [28, 43].

Our method is inspired by [12] and in accent similar to [86] in that both introduce a new variable that will de-singularize the integrand and both make use of a flat reference panel. The variable chosen has to goes to 0 as Green's function becomes singular so that the

	Flat Panel	Curved Panel
$\frac{1}{r}$	analytical formulae [30, 60]	semi-analytical with 2D polynomial fit [86]
$\frac{e^{ikr}}{r}, \frac{e^{kr}}{r}$	semi-analytical with 1D quadrature [93]	

Table 4.1: Common approaches for panel integration.

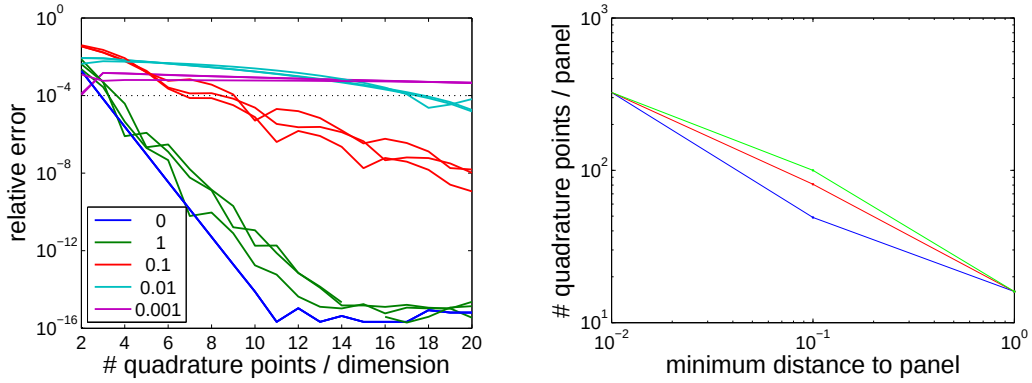
de-singularized integral can be approximated with numerical quadrature. For the $1/r$-typed Green's functions, ideally one can choose a new coordinate system such that the radial variable in the polar coordinates exactly cancels out the denominator of the Green's function. This can be easily done for flat panels, as shown in Figure 4-1, where integrand becomes a



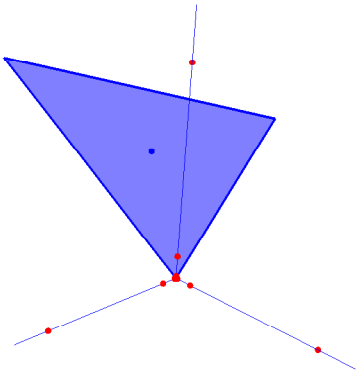
(a) $\iint \frac{1}{\sqrt{(x^2+y^2)}} dxdy$ (b) $\iint \frac{1}{r} r dr d\theta$

Figure 4-1: De-singularization of Green's function in polar coordinates.

constant. The resultant integral can be computed using two-dimensional quadrature in (r, θ) coordinates. While this is not necessary for flat panels since better techniques using analytical or semi-analytical formulae are possible, the same approach can be applied to curved panels. But before we proceed, it may be useful to compare this purely numerical approach to the analytical formulae [30, 60] for $1/r$ kernel on flat panels. It can give some insight to the accuracy and efficiency with this approach when applied to a curved surface. Figure 4-2(a) shows relative errors introduced by the quadrature scheme for various locations of evaluation points in the configuration in Figure 4-2(c). In the test panel in Figure 4-2(c), a vertex is chosen as the origin of polar coordinates for evaluation points in region defined by three axes originating from the vertex: a normal to the panel and perpendicular lines to



(a) (b)



(c)

Figure 4-2: Accuracy of numerical panel integration using quadrature in polar coordinates.

the two edges. To sample the region, we take evaluation points along the three axes since they are closest to the panel and presumably the corresponding panel integrations are most difficult to compute accurately. For various distances of an evaluation point to the vertex (relative to the distance from vertex to panel centroid), a set of three lines in Figure 4-2(a) corresponds to locations of evaluation points along the three axes. As shown in the figure, while the cost of numerical quadrature (for the same accuracy) increases as an evaluation point approaches a panel, the de-singularization technique is very effective when an evaluation point is on the panel itself and the cost is similar to the case where an evaluation point is about one panel size away. Figure 4-2(b) plots an estimate of the total number of quadrature points required to obtain an accuracy of 10^{-4} versus distance of an evaluation point to panel. The three distinct lines correspond to sample points along three axes.

For curved panels, it may be very difficult to define a coordinate system that cancels the denominator of the Green's function exactly. In addition, such coordinate system would not generalize to arbitrary curved surfaces anyway. On the other hand, if one defines a polar coordinate system on the flat triangle that shares the same vertices as a curved triangle, as shown in Figure 4-3, one may still generalize the above approach to a smooth but otherwise arbitrary surface, as long as a mapping function from the flat surface to the curved surface can be found. The key observation is that the origin of polar coordinates has to be chosen to correspond to the singular point of the Green's function, so that the resultant integrand, although not constant, is sufficiently smooth to be integrated with quadrature.

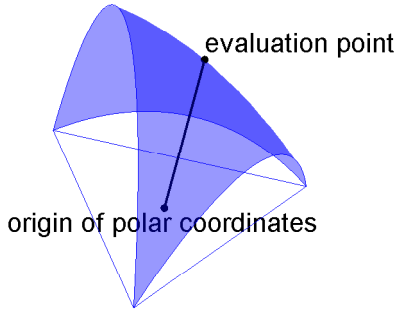


Figure 4-3: Mapping from flat to curved panel.

4.2 Higher Order Basis on Curved Panels

The strategy for panel integration outlined in the previous section relies on the availability of a one-to-one mapping function from a flat panel of local coordinates (u,v) to a curved panel of global coordinates (x,y,z):

$$P : \vec{r}_{flat}(u,v) \rightarrow \vec{r}_{curve}(x,y,z). \tag{4.1}$$

The mapping function on a reference panel serves three purposes: first is to define the origin and corresponding polar coordinates on the flat panel when an evaluation point is on the curved panel so that the integral is de-singularized. When an evaluation point is out-

side the curved panel itself, the Green's function is not singular so there is more freedom in choosing an origin. One strategy is to choose an origin on the flat panel whose corresponding point on the curved panel is among the closest points on the source panel to the evaluation point. The hope is that for nearby terms (evaluation points close to a curved panel), this can alleviate the steep slope associated with $1/r$ decay when r is small and thus minimize number of quadrature points required for a good approximation. However, as Figure 4-2(b) indicates, for arbitrarily close nearby terms, this approach may still be too costly. On the other hand, in settings typically encountered in boundary element method, evaluation points are at least about one panel size away for non-self terms, so this can still be a plausible approach for many practical problems.

The second purpose is to define two-dimensional quadrature points on the flat reference panel using product of one-dimensional Gauss points in polar coordinates. The trade-off for not having to define a separate curvilinear coordinate system for each curved panel is the need for a mapping function. However once it is defined, the de-singularization of Green's function integral and quadrature points can be set up in a consistent manner. In fact, an integral over the curved panel can be written as an integral over the flat panel with an appropriate Jacobian:

$$\int_{curve\ panel} G(\vec{r}_{eval}, \vec{r}')\, dS' = \int_{flat\ panel} G(\vec{r}_{eval}, P(\vec{r}'))|J(\vec{r}')|\, dS' \tag{4.2}$$

where $G(\vec{r}, \vec{r}')$ is Green's function of evaluation and source coordinates, and $|J| = \left|\frac{\partial P(\vec{r}_{flat})}{\partial \vec{r}_{flat}}\right|$ is the Jacobian of the mapping function in 4.1. The integral in the right-hand-side of equation (4.2) is carried out by numerical quadrature in polar coordinates. In general, the origin of polar coordinates is in the interior of a panel, but it can be translated to the case in Figure 4-2(c) by breaking it up into a few triangles and summing up individual contribution. This step is necessary because integral across a vertex is not smooth and cannot be accurately computed using quadrature.

Last but not least, in order to improve convergence, one has to use higher order basis than piecewise constant functions. Representing actual surface with mapping function and carrying out panel integration as such will improve accuracy over flat panel discretization,

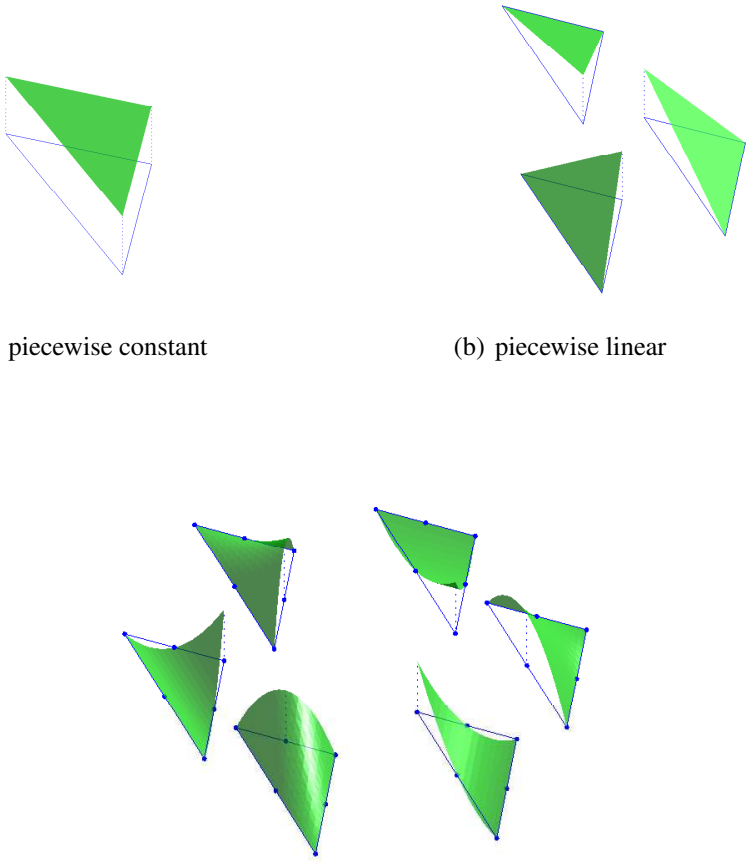
but only by a constant factor independent of discretization level. And if the higher cost of numerical panel integration over analytical formulate available only for flat panels exceeds this factor of improvement in accuracy, it may be hard to justify this approach. In other words, if one can get away with using more flat panels to achieve the same accuracy with same or less computation time than using fewer curved panels, the curved panel discretization is not necessarily superior. And one hopes to improve the trade-off by coupling curved panel representation with higher order basis, thus having a greater margin of improvement in accuracy as one refines discretization. The standard piecewise constant, linear and quadratic bases are shown in Figure 4-4 whose supports are defined on a few panels of a triangle mesh. The figure shows, however, not the entire support of a basis function, rather a few basis functions that share their supports on a panel. As opposed to piecewise constant basis whose supports do not overlap, three piecewise linear functions and six piecewise quadratic functions, both of them nodal basis functions, can be defined on a panel. As a result, orthogonality is not maintained for piecewise linear and piecewise quadratic bases defined on nearby panels. While all bases are defined on flat panels, the mapping function in (4.1) allows one to utilize these standard bases also on curved panels. They may not have the linear or quadratic form in curvilinear coordinates of a curved panel, but it is nevertheless a consistent and convenient way to define higher order bases on an arbitrary curved surface. The panel integration procedure can be easily modified to accommodate an additional term in the integrand:

$$\int_{curve\ panel} G(\vec{r}_{eval},\vec{r}')B(\vec{r}')\,dS' = \int_{flat\ panel} G(\vec{r}_{eval},P(\vec{r}'))|J(\vec{r}')|B(\vec{r}')\,dS' \qquad (4.3)$$

where $B(\vec{r}')$'s are basis functions defined on a reference panel.

4.3 Isoparametric Formulation

Ideally if a mapping function in (4.1) can be found to represent an exact surface, there will be no discretization error in geometry representation. However, in practice when this is not possible, a mapping function can still be found to represent an approximate surface. As



(a) piecewise constant

(b) piecewise linear

(c) piecewise quadratic

Figure 4-4: Standard piecewise polynomial basis defined on flat panels.

long as error introduced in the geometry discretization is no greater than the basis function discretization error, convergence rate consistent with order of basis can still be achieved. In this section we describe a general method to do this so that the approach in the previous section can again be adopted.

The quadratic basis in Figure 4-4(c) can be written as quadratic functions of local coordinates (u,v) on a flat panel:

$$B(\vec{r}_{flat}(u,v)) = \beta_0 + \beta_1 u + \beta_2 v + \beta_3 uv + \beta_4 u^2 + \beta_5 v^2. \tag{4.4}$$

An isoparametric representation can be obtained whereby the same quadratic polynomials

are used to approximate the geometry. The set of coefficients that uniquely determine a quadratic panel can be obtained by solving a set of 6x6 matrix equations:

$$\begin{bmatrix} \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & u & v & uv & u^2 & v^2 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \end{bmatrix} \begin{bmatrix} \alpha_0 \\ \alpha_1 \\ \alpha_2 \\ \alpha_3 \\ \alpha_4 \\ \alpha_5 \end{bmatrix} = \begin{bmatrix} \vdots \\ \vdots \\ x \\ \vdots \\ \vdots \end{bmatrix}, \begin{bmatrix} \vdots \\ \vdots \\ y \\ \vdots \\ \vdots \end{bmatrix}, \begin{bmatrix} \vdots \\ \vdots \\ z \\ \vdots \\ \vdots \end{bmatrix} \tag{4.5}$$

where the matrix entries are local coordinates (u,v) evaluated at six points on a flat panel and the right hand sides are global coordinates (x,y,z) of the corresponding points on an actual surface. Typically, those six points are chosen to be on vertices and edge midpoints as shown in Figure 4-4(c). The global coordinates of the three vertices are easy to obtain as they presumably coincide with the three vertices of a flat panel, but those of three edge midpoints may require additional knowledge of the actual curved geometry. On the other hand, in cases where analytical mapping function is not possible, an interpolated surface using discrete points can be a useful approach without sacrificing order of convergence. Once an approximated mapping function in terms of quadratic polynomials is obtained, an analytical Jacobian:

$$|J| = \sqrt{\left(\frac{\partial x}{\partial u}\frac{\partial y}{\partial v} - \frac{\partial y}{\partial u}\frac{\partial x}{\partial v}\right)^2 + \left(\frac{\partial y}{\partial u}\frac{\partial z}{\partial v} - \frac{\partial z}{\partial u}\frac{\partial y}{\partial v}\right)^2 + \left(\frac{\partial x}{\partial u}\frac{\partial z}{\partial v} - \frac{\partial z}{\partial u}\frac{\partial x}{\partial v}\right)^2} \tag{4.6}$$

can be easily obtained from partial derivatives of quadratic functions.

4.4 Computational Results on Sphere

A unit sphere in an infinite fluid potential flow problem, which has an analytical solution [58, 63] is used to demonstrate the convergence properties of higher order bases. The integral equation is

$$\phi(\vec{r}) = \int_\Omega \frac{\sigma(\vec{r}')}{|\vec{r} - \vec{r}'|} dS' \tag{4.7}$$

where $\phi(\vec{r})$ is given. The potential and corresponding solution for the sphere example are

$$\phi(\vec{r}) = -\frac{z}{2\|\vec{r}\|^3} \qquad \|\vec{r}\| \geq 1 \tag{4.8}$$

$$\sigma(\vec{r}) = -\frac{3z}{8\pi} \qquad \|\vec{r}\| = 1. \tag{4.9}$$

A plot of the analytical solution σ is shown in Figure 4-5 and Figure 4-6 shows convergence properties of various discretization using bases of increasing order. Accuracy is assessed

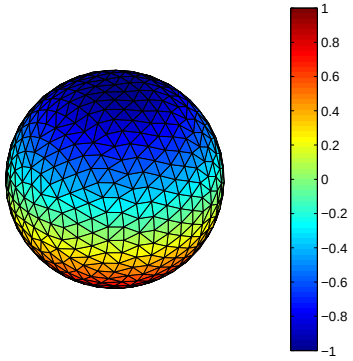


Figure 4-5: Solution distribution of potential flow problem on sphere.

in terms of integrated error which is approximated by a weighted sum of absolute errors at discrete points. For piecewise constant basis, the obvious choice of these points is the set of centroidal collocation points:

$$\Sigma_{panel} = \sum_{i=1}^{N} a_i \left|\sigma_{num}(\vec{r}_{centroid}) - \sigma_{exact}(\vec{r}_{centroid})\right| \tag{4.10}$$

where a_i's are areas associated with each panel and N is total number of panels. For higher order bases, however, the integrated error can alternatively be approximated from a set of collocation points which do not fall on panel centroids:

$$\Sigma_{node} = \frac{4\pi R^2}{n} \sum_{i=1}^{n} \left|\sigma_{num}(\vec{r}_{colloc}) - \sigma_{exact}(\vec{r}_{colloc})\right| \tag{4.11}$$

where R is radius of sphere and n is total number of collocation points. Note that the sum is weighted by an average area in this node-based approximation instead of panel area in

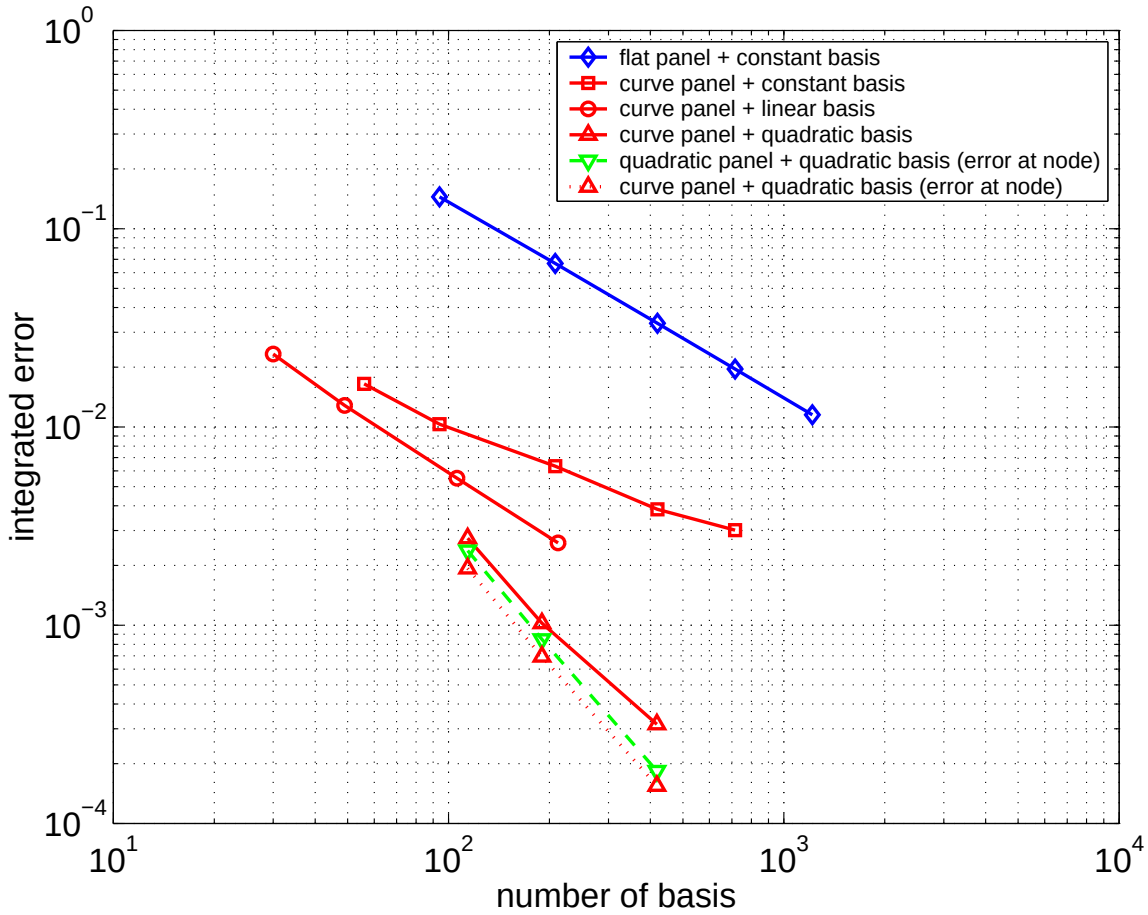


Figure 4-6: Convergence results for a potential flow problem on sphere

(4.10). In Figure 4-6, solid lines correspond to panel-based integrated errors calculated for all bases. For piecewise linear and piecewise quadratic bases, post-processing is needed to evaluate nodal basis functions at panel centroids, thereby making a consistent comparison to the piecewise constant basis. Alternatively, the last two broken lines show integrated errors computed at collocation points for the piecewise quadratic basis according to (4.11): the dashed line corresponds to an isoparametric representation of the curved geometry and the dotted line, like all other solid lines except flat panel discretization, uses mapping functions to represent the exact geometry. As can be seen from the flat panel versus curved panel discretization in the piecewise constant basis, a better representation of geometry improves accuracy but not convergence. In fact, based on the integrated error comparison, convergence seems to deteriorate when curved panels are used. This is perhaps due to superconvergence [78] such that convergence rate of linear basis is achievable with constant

basis when centroidal collocation points are used, as is in the case of flat panel discretization. For curved panel discretization, the centroids do not fall on panels and the collocation points on panels (radially project from centroids of flat panels that share the same vertices) have been used in the calculation. In order to improve convergence, a better representation in terms geometry as well as basis function discretization have to be used. As can be seen from the plot, both accuracy and convergence improve with the use of higher order basis. The isoparametric representation in Section 4.3 is an effective approach to approximate geometry and is capable of achieving convergence consistent with order of basis functions used. Also, the calculations of integrated error according to (4.10) and (4.11) both give similar results.

4.5 Discussion

In this chapter, an approach to improve convergence through accurate representation of geometry and use of higher order bases is presented. The notion of mapping function is introduced to parameterize a curved surface by local coordinates of a flat reference panel on which the standard higher order bases can be defined. Furthermore, the integration of the product of Green's function, basis functions and Jacobian of a mapping function on a flat panel is carried out using quadrature points in polar coordinates whose origin has been carefully chosen to desingularize the integrand. The approach is in contrast to earlier methods [53, 72] where polynomial approximation of a curved surface is assumed and analytical formulae for panel integration are derived. In cases when mapping functions from flat to curve panels are exact, no geometrical error is introduced. On the other hand, when an exact mapping function is not available, an isoparametric approximation can be used. So long as geometrical and quadrature errors are no greater than error introduced by the basis function discretization, convergence consistent to the order of basis functions can be achieved. However, based on the author's experience, integration by quadrature for singular functions, even with desingularization technique described above, demands higher cost compared with analytical formulae available for flat panels [30, 60]. It is not clear if the improvements in accuracy and convergence can sufficiently offset the higher

computational cost.

In addition, the availability of matrix sparsification techniques [59, 63] utilized in Chapter 3 further complicates the issue. It is clear that the same matrix-implicit iterative solver approach can be used for the higher order basis. However, such multi-resolution techniques rely on approximation of faraway interactions. Since higher order basis have larger supports and therefore less faraway interaction, the amount of speedup with acceleration is less than compared with piecewise constant basis. Furthermore, the amount of overhead in either multipole- [59] or FFT-based [63] approach is likely to be higher for higher order bases as well for the same size as a piecewise constant basis. This is because in calculating moments of a basis function used in approximating faraway interaction, contribution from all panels within its supports has to be summed up whereas for piecewise constant basis, it only needs to be done once per panel. It is this trade off between accuracy and speed that motivates the development of a method with even faster convergence in the next chapter.

Chapter 5

Spectral Method

Traditional boundary element methods using piecewise-constant basis are low order, therefore large numbers of unknowns are needed to achieve high accuracy. While acceleration techniques [7, 23, 25, 39, 59, 63, 70] make it possible to solve such problems, memory is often a bottleneck. As a result, there is much interest in developing higher order methods [12, 13, 38, 53] that can achieve faster convergence and reduce problem size. In [13, 53], the use of a higher order basis based on B-splines resulted in faster algebraic convergence, while in [12, 38], the aim was to attain spectral convergence. In this chapter, we propose a new approach to discretize a singular integral equation using global, numerically orthogonal basis and demonstrate spectral convergence (error decays exponentially with number of unknowns). Our method differs from [12, 38] in that we use an explicit high order basis in our approach.

Higher order basis such as piecewise linear or piecewise quadratic polynomials in the previous chapter can improve convergence, but they are still a local approximation to the solution. As will be made clear in this chapter, our method takes a more global approach in an attempt to create a high order approximation over a large region. The improvement in accuracy is significant: the method exhibits a spectral convergence rates. It is well know that the Nyström method can attain spectral convergence for second kind integral equations with non-singular kernels by using collocation points as quadrature points [1]. However, the method breaks down for Laplace and Helmholtz problems, since the kernels are singular. Our approach can be applied to singular integrals and can be considered as an extension

of the Nyström method. On the other hand, our method can have a different interpretation as an interpolation approach, similar to [12]. This interpretation allows an efficient implementation of matrix-vector multiplication that can be used in an iterative solution of the resultant matrix equation. With this implementation, the speed of our approach without acceleration may still be competitive with low-order accelerated methods for the same accuracy, with orders of magnitude lower memory use.

In the following section, we will describe the basis we use in the proposed method. In Section 5.2, we use the panel integration techniques in the previous chapter in the current context for integration over curved surfaces. In Section 5.3, the equivalence between Galerkin and collocation formulation for the particular choice of basis is described. Numerical results are presented in Section 5.4 and finally in Section 5.5, we conclude with some discussion.

5.1 Numerically Orthogonal Basis

An alternative basis for discretization is a set of numerically orthogonal polynomials defined by carefully chosen quadrature points. In contrast to a panel-based representation whereby orthogonality is only partially maintained by spatial separation of basis function' supports, a quadrature based approach ensures good orthogonality for arbitrarily high order basis, though their supports have significant overlap. To demonstrate how such basis functions can be defined, consider a global surface that can be partitioned into a few regions and each of which can be associated with a one-to-one mapping function as in (4.1) defined on a local patch. In a complicated geometry where exact mapping is not possible, a high order approximation, consistent with the basis order, has to be used since convergence is limited by the lower accuracy of the two. A second requirement is the availability of good quadrature points associated with each patch. For example, in a rectangular patch a tensor product of one-dimensional Gauss-Lobatto quadrature points is used, and basis set can be similarly defined as polynomials that take on unit value at one of the grid points but zero at all other grid points. In one dimension, these are the Lagrangian interpolating polynomials

[6]:

$$\ell_i(u) = \frac{\prod_{k=1,k\neq i}^{m} (u - u_k)}{\prod_{k=1,k\neq i}^{m} (u_i - u_k)} \tag{5.1}$$

such that

$$\ell_i(u_k) = \delta_{ik} \qquad i,k = 1,\ldots,m \tag{5.2}$$

where u_k is coordinate of kth quadrature point. The basis functions on a patch can therefore be written as a product of two one-dimensional polynomials as in

$$B_{(i*j)}(u,v) = \ell_i(u)\ell_j(v). \tag{5.3}$$

Therefore, if m quadrature points are used along each dimension, there will be m^2 basis functions. A good set of quadrature points ensures orthogonality as the inner product over a patch approximated by the same quadrature points is always zero by design:

$$\iint_{patch} B_{IJ}(u,v)B_{I'J'}(u,v)\,du\,dv \approx \sum_{i,j} B_{IJ}(u_i,v_j)B_{I'J'}(u_i,v_j) = \delta_{II'}\delta_{JJ'}. \tag{5.4}$$

For the basis function associated with a boundary node, its support will span across nearby patches so that computed solutions will be continuous along patch boundaries. This is an improvement over panel-based representation since all derivatives are continuous within a patch which does not shrink in size as number of basis increases. The use of such a numerically orthogonal basis was proposed in the spectral element method [62] and is well known in the finite element community. However, to authors' knowledge, it has not yet been applied to the boundary element method, perhaps hindered by the difficulty of panel integration, the subject of the next section.

5.2 Integration over Curved Surfaces

Once the mapping function (4.1) and basis functions (5.3) have been defined on a patch, integration of products of Green's and basis functions over the actual surface can be per-

formed using parametric coordinates (u,v). For an evaluation point at $\vec{r}(x,y,z)$:

$$\int\limits_{(x,y,z)} G(\vec{r},\vec{r}')B_i(\vec{r}')\,dS' = \int\limits_{(u,v)} G(\vec{r},P(\vec{r}_f{}'))B_i(\vec{r}_f{}')|J(\vec{r}_f{}')|\,dS' \tag{5.5}$$

where $\vec{r}_f{}'$ is such that $P(\vec{r}_f{}') = \vec{r}'$ (see (4.1)), $|J|$ is the Jacobian of the mapping function P. Note that the basis function, originally defined on the local patch, is also being used to represent the solution in the global surface through the mapping function:

$$B_i(\vec{r}'(x,y,z)) = B_i(P(\vec{r}_f{}'(u,v))). \tag{5.6}$$

An analytical expression for (5.5) is not generally available as the Jacobian can be very complicated, and straightforward quadrature in (u,v) coordinates is not sufficiently accurate for evaluation points on or close to the surface associated with a source patch. It is shown in Section 4.1 that, however, the integral can be de-singularized in appropriately chosen polar coordinates in place of u and v. The key is to locate the polar coordinates origin such that the radial coordinate ρ goes to zero at the singular point. The resulting integrand is smooth and can be evaluated using Gauss quadrature points in (ρ,ϑ). In particular, if the evaluation point $\vec{r}_j$ is such that $P^{-1}(\vec{r}_j)$ is on the patch, then

$$(u_o,v_o) = P^{-1}(\vec{r}_j(x,y,z)) \tag{5.7}$$

and

$$\begin{aligned} u-u_o &= \rho\cos\vartheta \\ v-v_o &= \rho sin\vartheta \end{aligned}. \tag{5.8}$$

is the appropriate change of variables to apply to (5.5).

5.3 Equivalence between Galerkin and Collocation Formulation

For the Galerkin formulation in (2.10), the outer integral with the target basis function is smooth and is typically approximated by quadrature. Since the support of a basis function in (5.3) have been defined on patches associated with quadrature points, one can use the same points to approximate the outer integral:

$$A_{ji}^{Galerkin} \approx \sum_{k=1}^{\#points} w_{j,k} B_j(\vec{r}_{j,k}) |J(\vec{r}_{j,k})| \int_{support} G(P(\vec{r}_{j,k}), \vec{r}') B_i(\vec{r}') dS' \tag{5.9}$$

where i and j are again global indices of all nodes, and k is the index of quadrature points associated with target basis function's support, which may be on one or a few patches. So $\vec{r}_{j,k}$ is the position vector of kth quadrature point on B_j's support with corresponding quadrature weights $w_{j,k}$. Since each basis function is chosen to be non-zero at only one quadrature point:

$$B_j(\vec{r}_{j,k}(u,v)) = \delta(|\vec{r}_j(x,y,z) - P(\vec{r}_{j,k}(u,v))|), \tag{5.10}$$

each row of the Galerkin matrix in (5.9) reduces to the corresponding row of the collocation matrix scaled by some constant. Each entry of the right hand side in the Galerkin formulation in (2.10) is also equal to the corresponding entry of the right hand side in the collocation formulation in (2.11) scaled by the same constant, namely $w_{j,k}|J(\vec{r}_{j,k})|$. As a result, the Galerkin formulation for the particular choice of basis in (5.3) is equivalent to the collocation formulation. One can simultaneously take advantage of Galerkin scheme's convergence property and collocation scheme's computational efficiency.

5.4 Computational Results using Exact Geometry

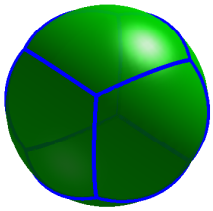
5.4.1 Potential Flow on Sphere

The sphere example in Section 4.4 is used here to demonstrate the convergence property of the proposed approach, and to make a comparison with the standard panel-based higher or-

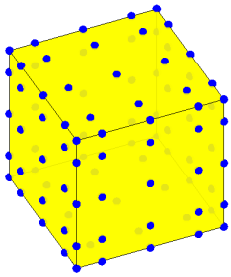
der basis in the previous chapter. In order to describe the spherical geometry, local patches on six faces of a cube centered at the origin as shown in Figure 5-1 is used, and a mapping function is defined by radially projecting any point on the cube to sphere. The Jacobian of the mapping is given by:

$$|J| = \frac{hR^2}{d^3} \tag{5.11}$$

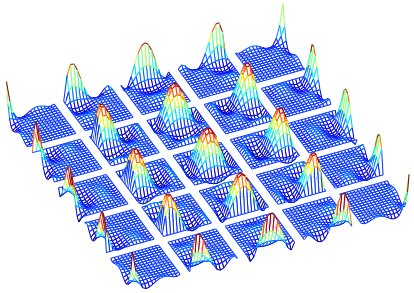
where d is the distance from center of sphere to a local point on a patch, h is perpendicular distance from center of sphere to the patch and R is radius of sphere. An $m \times m$ Gauss-Lobatto grid is used on on each face of a cube, a basis set is defined on the grid and $2m \times 2m$ quadrature points in polar coordinates are found to be sufficiently accurate in this case to evaluate the integral in (5.5). Both direct and iterative solvers are used to obtain a numerical solution and accuracy is assessed in terms of integrated error defined in 4.11.



(a) partition on sphere



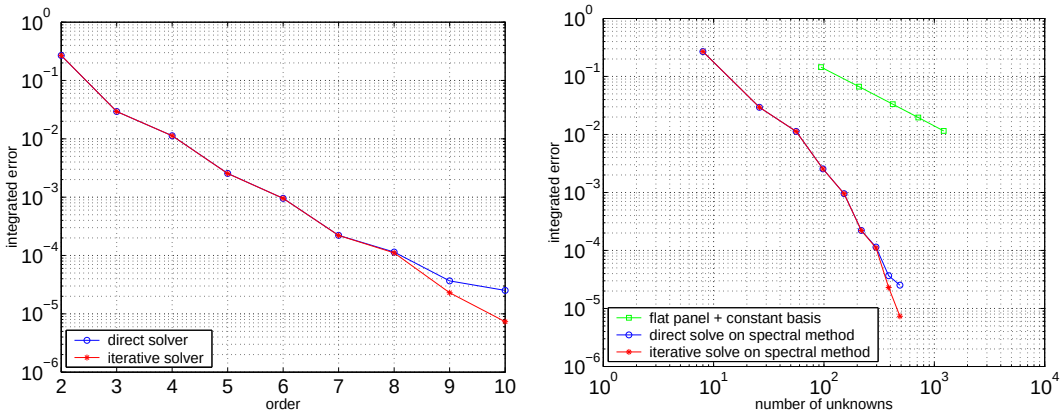
(b) Gauss-Lobatto points on six local patches



(c) two-dimensional Lagrangian basis

Figure 5-1: Mapping from cube to sphere.

Figures 5-2 show the results in two different scales, one plotted against order m and the other plotted against number of unknowns. It can be seen that both direct and iterative



(a) (b)

Figure 5-2: Potential flow problem on sphere.

solution have similar accuracies, except when errors approach the 10^{-6} tolerance used for the iterative solver. Both approaches, however, show similar convergence behavior: a straight line in a log-linear plot and a curve in a log-log plot indicates spectral convergence, and the error decays exponentially with number of unknowns. This is compared to the standard panel method in Figure 5-2(b). The improvement over the traditional approach is clear: not only is the accuracy better for the same number of degrees of freedom, or fewer unknowns are needed for the same accuracy, but the method's advantage grows with increasing problem size or more stringent error tolerance. For the sphere problem, the spectral method is able to achieve six digits of accuracy with about 500 unknowns, which in our MATLAB® implementation, takes less than ten minutes in a 3GHz Intel® Xeon machine. By extrapolating the straight line in Figure 5-2(b), one can estimate that at least a million panels are needed for the standard method to achieve the same accuracy. Table 5.1 shows matrix condition number and the number of iterations required to converge to 10^{-6} tolerance using GMRES [75], without any pre-conditioner. Note that the condition number and number of iterations are growing slowly with problem size. Figure 5-3 summarizes the key results to shows a comparison to the standard panel method and the standard panel-based higher order method. It can be seen that even with quadratic basis discretization, the spectral method does better beyond 200 unknowns.

order	2	3	4	5	6	7	8	9	10
number of unknowns	8	26	56	98	152	218	296	386	488
condition number	11	32	57	92	136	189	254	327	408
number of iterations	2	5	6	8	8	7	7	5	4

Table 5.1: Matrix condition of direct and iterative solver.

5.4.2 Capacitance of a Cube

The sphere example in the previous section demonstrates that the spectral method is very effective for problems with smooth geometry and smooth underlying solution. In the case where solution is not smooth or contains singularities, polynomial approximation cannot achieve spectral convergence and the spectral method as presented in this chapter is not necessarily better than the traditional low order approach. The capacitance calculation of a cube is used to demonstrate this idea. The integral equation is the same as (4.7) but the potential is set to constant. The capacitance, by definition, is sum of charge distribution in the corresponding solution. It is well-known that the charge distribution (at constant potential) is singular in the presence of geometrical discontinuity: in this case along 12 edges of a cube. That is why in the traditional panel method, a thin strip of panels known as edge cells [73] are typically used along surface boundary while more uniform-sized panels are used in the interior. While piecewise constant basis cannot faithfully capture a singularity, the sum of charges as in the capacitance calculation can still be well-approximated.

In the spectral method, while the geometrical discontinuity in itself does not present a problem since local patches have been defined to align with the boundary, the fact that the underlying solution is singular prohibits spectral convergence when polynomial basis functions are used. Figure 5-4 shows capacitance calculation of a cube using both the spectral method and standard panel method implemented in FastCap [59]. Although analytical formulae for a cube capacitance is not available, a numerical result of 0.660678 nF as given in [54] is used as a reference to compute the relative errors. As shown in the plot, the convergence rate for the spectral method is not significantly better and accuracy using both methods is comparable. In the implementation of the spectral method, Lagrangian basis defined on Gauss-Lobatto grid as in 5-1 are used so that solution has continuity across patch boundary. However, in this case where solution become singular along edges, enforc-

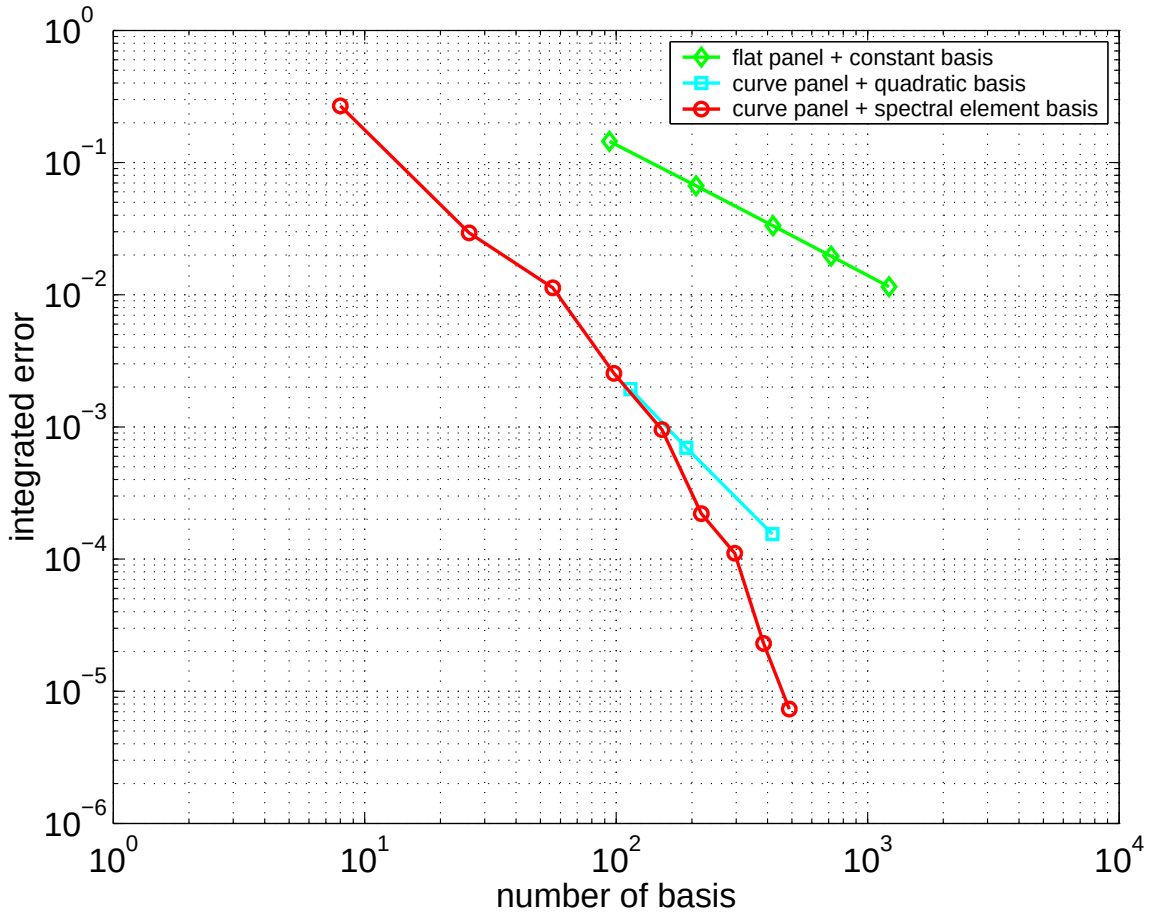


Figure 5-3: Comparison of convergence properties between standard and spectral methods.

ing continuity is not very meaningful. Alternatively, the basis defined on regular Gauss-Legendre grid can be used instead. As shown in the figure, accuracy is somewhat improved although the same convergence behavior remains. Further improvement in accuracy can be obtained in the spectral method if edge cells are also used along boundary, but in order to increase the convergence rate, specialized basis (possibly singular) tailored to the underlying charge distribution has to be used.

5.4.3 Capacitance of an Ellipsoid

The capacitance calculation is repeated for an ellipsoid. It has a smooth geometry as in the sphere case, but the solution is not constant which allows a meaningful comparison

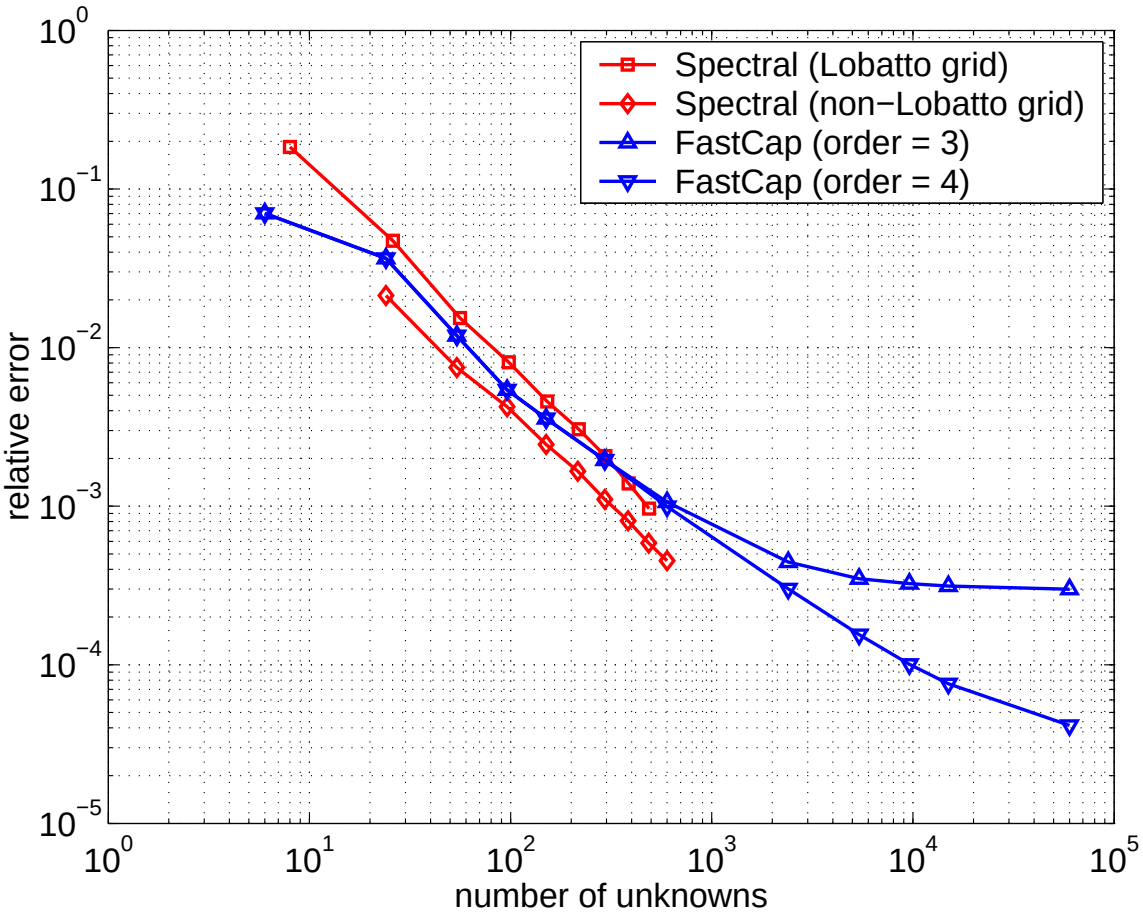


Figure 5-4: Capacitance calculation of a unit cube.

between the spectral and standard methods. The equation of an ellipsoid is given by:

$$\frac{x^2}{a^2}+\frac{y^2}{b^2}+\frac{z^2}{c^2}=1 \tag{5.12}$$

where a, b and c are lengths along its three axes. For simplicity, we will consider the case when two of the lengths are the same ($b=c$), which is a special case of an ellipsoid known as a spheroid. It corresponds to surface of revolution of an ellipse about one of its principal axes. Depending on the axis of rotation, an spheroid can be of cigar-shaped (a prolate spheroid) or disk-shaped (an oblate spheroid). Figure 5-5 shows the test example of a prolate spheroid in the capacitance problem. In the spectral method, local patches is again defined on the six faces of a cube and the mapping Jacobian from a cube to ellipsoid is similar to (5.11) but the radial distance R is no longer a constant. The capacitance of a

spheroid has an analytical formula [16]:

$$C_{prolate} = 4\pi\varepsilon_0 \frac{\sqrt{a^2 - b^2}}{\ln \frac{a+\sqrt{a^2-b^2}}{b}} \tag{5.13}$$

and Figure 5-6 shows relative errors in the capacitance calculation of an ellipsoid ($a = 3$, $b = c = 1$) using both spectral method and the standard panel method. As demonstrate in the sphere case, spectral convergence is again validated here from the linear decay in the log-linear plot. The curve in the log-log plot indicates exponential decay in the error versus number of unknowns. One can compare the improvement in accuracy between both methods in Figures 5-6 and 5-2, and note that in the sphere example, the spectral method seems to have a greater improvement over the standard panel method. However, one should also note that the error metrics used in both examples are not equivalent. While the integrated error in the potential flow problem is a sum of absolute errors, the relative error in the capacitance calculation is an error of sums. So depending on the physical quantity of interest, one may achieve a slightly different factor of improvement, but the superior accuracy and convergence in the spectral method is demonstrate in both cases.

5.5 Discussion

In this chapter a novel approach is proposed to discretize an integral equation. As opposed to the standard panel method, local patches are used to define geometry through mapping

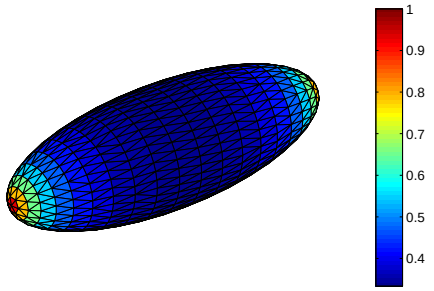
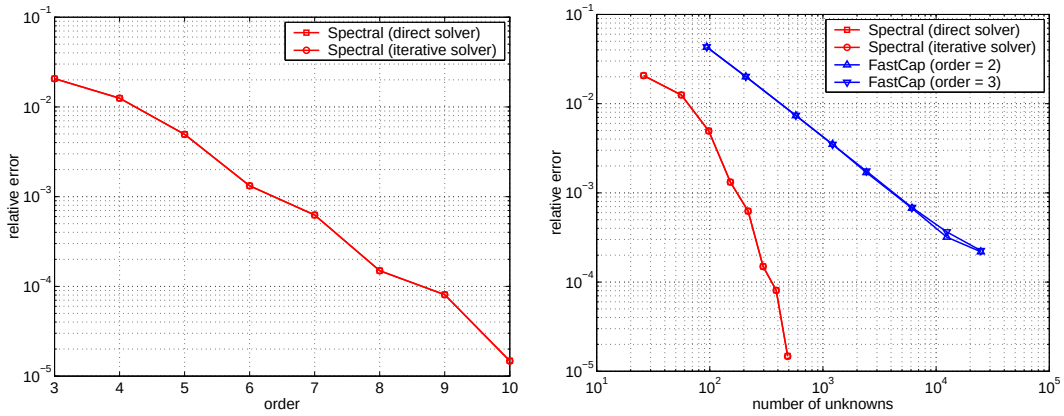


Figure 5-5: Charge distribution on an ellipsoidal surface of equal potential.



(a) (b)

Figure 5-6: Capacitance calculation of a spheroid.

functions and the number of patches is kept constant as discretization increases. The use of progressively higher order bases with discretization results in spectral convergence. The superior accuracy and convergence behavior would make this approach more attractive than the standard higher order method in the previous chapter. In addition, the use of common patches where an increasing number of basis functions share their supports on as discretization increases has ramification on the efficiency as well. While a discussion of an efficient implementation is deferred to Chapter 7, the cost of panel integration can be minimized by taking advantage of the fact that quadrature points defined on a patch can be shared among many basis functions. This would not be possible in the traditional panel-based discretization. On the other hand, the use of large patches can mean that the use of multi-resolution techniques [7, 23, 25, 39, 59, 63, 70], commonly used with the panel method, may not be very effective when coupled with the proposed approach. This particular issue, together with two interpretations of the spectral method, are discussed in the following subsections. The extension to a more general molecular surface than sphere will be described in the next chapter.

5.5.1 Multi-resolution Techniques

Traditional approach with piecewise-constant basis requires a lot of panels for accuracy and one typically uses multi-resolution techniques to accelerate the solution procedure by

approximating faraway interaction. Such techniques can also be used for the proposed approach. In fact, for a distant evaluation point, the influence of a basis function on a patch is well represented by a point charge since the basis are chosen to be non-zero at only one quadrature point. So fast multipole method [23] can be readily applied to approximate the effect of all basis functions on a patch. On the other hand, large patches should be used whenever possible to define the geometrical mapping in order to take full advantage of the proposed method. The reasons are twofold: better accuracy because solution has only first order continuity across patch boundary but smooth anywhere else, and improved efficiency because quadrature points in polar coordinates can be shared among all basis functions on the same patch. So it is not clear if total number of patches will be large enough to make multi-resolution techniques useful. At the same time, multi-resolution approximation may interfere with the spectral convergence behavior the method seeks to achieve.

5.5.2 Two Interpretations

As mentioned in the beginning of this chapter, our approach is similar to the Nyström method in that it is also capable of achieving spectral convergence. For an integral equation with non-singular kernels, the Nyström approach uses quadrature points as collocation points and reduces to a set of algebraic equations. For singular kernels, however, our approach shows that the same spectral convergence can be obtained if explicit Lagrangian basis functions defined on quadrature points can be integrated sufficiently accurate. And the basis functions we use are the same as those in the spectral element method [62].

Alternatively, our approach can be seen as a more global interpolation of the underlying function, and carrying out the integration on patches. This is in contrast to the standard approach where accuracy and convergence is limited by panel-wise or local approximation. We also demonstrated that good accuracy and a well-conditioned matrix equation can be achieved by using a set of good interpolation points defined on patches. In this view, our method is similar to [12] although in that approach, uniform grids are used and a set of partition-of-unity weighting functions are introduced to make the underlying representation periodic.

Chapter 6

Spherical Harmonic Surface Representation

The previous chapter demonstrates how local patches defined on six faces of a cube and the associated Lagrangian polynomials defined on a Gauss-Lobatto grid can be useful in the discretization of integral equations. While this is a radically different approach in representing geometry from the panel method and the examples shown involve only simple geometries where analytical mapping functions can be found, this chapter seeks to explain how the spectral method can be extended to general molecular surfaces using the same cube-based discretization. In fact, it will be extraordinarily difficult to define mapping functions from a given triangulation of flat panels to the actual surface. Fortunately, there has been work done in global representation of molecular surfaces [10, 18, 49, 50, 55, 57] using spherical harmonics, which are defined on a unit sphere. Coupled with the approaches used in the sphere example in Section 5.4.1, a closed 3-D surface can be defined via mapping from cube to sphere, and from sphere to a spherical harmonic surface. In this chapter, it will be made clear how a biomolecule problem is no more difficult than a sphere problem with a substituted Jacobian. The computational cost, therefore, would be similar to that of a sphere example, with additional overhead in evaluating spherical harmonic functions.

In the next section, we establish the notation and formulae used for the spherical harmonics. In Section 6.2, we describe how a spherical harmonic representation can be generated using least squares fitting to points distributed on a molecular surface. In Section 6.3,

the Jacobian for the mapping from a cube to the approximated global molecular surface is computed. The effectiveness of spherical harmonic representation is verified using area calculation in a few examples in Section 6.4. Finally, computational results of the spectral method using spherical harmonic surfaces are shown in Section 6.5 and we conclude with a discussion in Section 6.6.

6.1 Spherical Harmonics

Spherical harmonics $Y_n^m(\theta,\phi)$ can represent solution of Laplace's, Helmholtz's and Schrödinger equations in angular coordinates: $0 \leq \theta \leq \pi$ is polar angle from positive z-axis and $0 \leq \phi < 2\pi$ is azimuthal angle from positive x-axis. Physically, they correspond to the shape of electron orbitals and therefore seem like a natural choice for describing molecular surfaces. Mathematically, they are orthogonal functions defined on the surface of a unit sphere. The set of basis is complete such that any sufficiently smooth function can be represented as a sum of spherical harmonics:

$$f(\theta,\phi) = \sum_{n=0}^{\infty} \sum_{m=-n}^{n} c_n^m Y_n^m(\theta,\phi) \tag{6.1}$$

whose coefficients, due to orthogonality, can be calculated by integrating $f(\theta,\phi)$ with an individual basis function over surface of a unit sphere:

$$c_n^m = \int_0^{2\pi} \int_0^{\pi} f(\theta,\phi) Y_n^m(\theta,\phi) \sin\theta \, d\theta \, d\phi. \tag{6.2}$$

The spherical harmonics are commonly written as complex functions, but an equivalent set of real functions can be used by taking linear combinations of the complex version. For our purpose, it is more convenient to work with real spherical harmonics:

$$Y_n^m(\theta,\phi) = \begin{cases} \alpha_n^m P_n^m(cos\theta)\cos m\phi & \text{if } m \geq 0 \\ \alpha_n^{|m|} P_n^{|m|}(cos\theta)\sin |m|\phi & \text{if } m < 0 \end{cases} \tag{6.3}$$

where $P_n^m(\theta,\phi)$ are associated Legendre functions, and

$$\alpha_n^m = (-1)^m \sqrt{\frac{(2-\delta_{0m})(2n+1)}{4\pi}\frac{(n-m)!}{(n+m)!}} \tag{6.4}$$

are normalization constants chosen such that the basis are orthonormal:

$$\int_0^{2\pi}\int_0^{\pi} Y_n^m(\theta,\phi)Y_{n'}^{m'}(\theta,\phi)\sin\theta\, d\theta\, d\phi = \delta_{nn'}\delta_{mm'}. \tag{6.5}$$

For each order n, there are $2n+1$ distinct basis functions corresponding to $-n \leq m \leq n$. The first three orders of the spherical harmonic functions are shown in Figure 6-1.

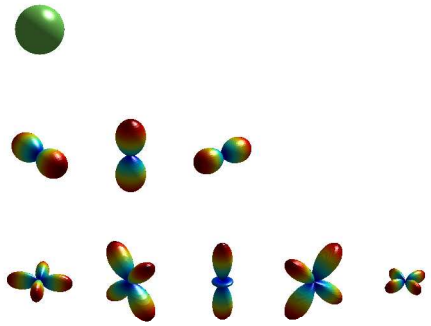


Figure 6-1: Spherical harmonics functions $|Y_n^m(\theta,\phi)|^2$ of the first three orders.

6.2 Surface Approximation by Least Squares

Molecular surface representation by spherical harmonics has been proposed by various authors [10, 18, 49, 50, 55, 57]. In such approach, a set of coefficients is generated from a point distribution or triangulation of a molecular surface, typically obtained from another program such as MSMS [76]. The coefficients, together with spherical harmonic basis, represent an analytical approximation to the surface geometry, and can be differentiated. The simplest strategy for generating the spherical harmonic representation is to first pick a ”molecular center” to be the origin of a spherical coordinate system, and then represent

each surface point using a spherical harmonic expansion as

$$r(\theta,\phi) \approx \sum_{n=0}^{N} \sum_{m=-n}^{n} \widetilde{c_n^m} Y_n^m(\theta,\phi) \tag{6.6}$$

where N is the expansion order, r, θ and ϕ are the spherical coordinates of points on the surface. The approximation is a truncation of the series in (6.1) if and only if the coefficients are calculated exactly as in (6.2). Depending on the numerical error and algorithm used to generate the coefficients, $\widetilde{c_n^m}$ and c_n^m might not be equal for $n \leq N$. On the other hand, the approximation should improve with N as the basis set is enriched. It does have the limitation that the surface has to be starlike [18], which means that there exists an origin within the molecule such that an outgoing ray intersects the molecular surface exactly once. Other techniques that avoid this restriction are available [10, 18, 49] for surfaces that are topologically equivalent to a sphere.

The coefficients $\widetilde{c_n^m}$ can be calculated by forming the inner product integral in (6.2). Instead, we adopt a more easily implemented least squares approach [10]. Given a set of k points (r,θ,ϕ) on a molecular surface, we look for a set of coefficients $\{a_j\} = \{\widetilde{c_n^m}\}$ such that $||r - Aa||_2$ is minimized, where each element of A are spherical harmonics evaluated at (θ,ϕ) coordinates:

$$A_{i,j} = A_{i,n^2+m+n+1} = Y_n^m(\theta_i,\phi_i) \tag{6.7}$$

for $1 \leq i \leq k$ and $1 \leq j \leq (N+1)^2$. To solve the least-squares problem, we used the singular value decomposition, $A = U\Sigma V^T$ [83]. The set of coefficients in (6.6) can then be obtained from

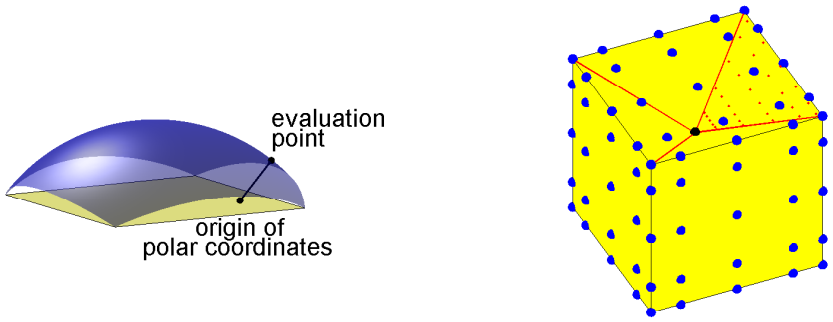
$$a = V\Sigma^{-1}U^T r \tag{6.8}$$

where $a_j = a_{n^2+m+n+1} = \widetilde{c_n^m}$.

6.3 Jacobian of Spherical Harmonic Surface

In order to incorporate the spherical harmonic representation into the spectral method, one must be able to integrate over the molecular surface given by (6.6). In addition, in order to

de-singularize the integral involving the Green's function, our approach is to carry out the integration patch-wise on six faces of a cube by setting up appropriate polar coordinates on each face. As shown in Figure 6-2(a), the origin of the polar coordinates is chosen to coincide with Green's function's singularity through an appropriate mapping function. The two sets of quadrature points in the spectral method are depicted in Figure 6-2(b): the red dots correspond to integration points in polar coordinates in which integrands are evaluated, and the blue nodes correspond to Gauss-Lobatto points on each patch used to define Lagrangian basis. Once the Jacobian of the mapping functions from cube to a spherical harmonic surface is obtained, the spectral method described in the previous chapter can be readily applied to solve a general biomolecular problem.



(a) Desingularization in polar coordinates (b) Basis function nodes and quadrature points

Figure 6-2: Integration domain defined on six faces of a cube.

Consider the area integral in order to figure out the appropriate Jacobian. Given a molecular surface parameterized by θ and ϕ, the normal vector is given by:

$$\hat{N} = \frac{\overrightarrow{R}_\theta \times \overrightarrow{R}_\phi}{|\overrightarrow{R}_\theta \times \overrightarrow{R}_\phi|} \tag{6.9}$$

and the area integral is given by:

$$\int dS = \int_0^{2\pi} \int_0^{\pi} |\overrightarrow{R}_\theta \times \overrightarrow{R}_\phi| \, d\theta \, d\phi \tag{6.10}$$

where

$$\overrightarrow{R} = r(\theta,\phi)\sin\theta\cos\phi\hat{x} + r(\theta,\phi)\sin\theta\sin\phi\hat{y} + r(\theta,\phi)\cos\theta\hat{z} \tag{6.11}$$

is position vector of any point on the molecular surface, $\overrightarrow{R}_\theta$ and $\overrightarrow{R}_\phi$ are the partial derivative of $\overrightarrow{R}$ with respect to θ and ϕ respectively. Alternatively, the cross product in (6.10) can be expressed in spherical coordinates [4, 19]:

$$\overrightarrow{R}_\theta \times \overrightarrow{R}_\phi = r^2\sin\theta\hat{r} - rr_\theta\sin\theta\hat{\theta} - rr_\phi\hat{\phi} \tag{6.12}$$

so that

$$\int dS = \int_0^{2\pi}\int_0^{\pi} r\sqrt{r^2\sin^2\theta + r_\theta^2\sin^2\theta + r_\phi^2}\,d\theta\,d\phi \tag{6.13}$$

where $r = r(\theta,\phi)$ in (6.6), r_θ and r_ϕ are derivatives of radius coordinate with respect to θ and ϕ respectively.

In order to carry out the surface integral on a reference patch on each face of a cube, one needs the Jacobian for the change of variables:

$$d\theta\,d\phi = |J_{map}|\,du\,dv \tag{6.14}$$

which corresponds to a mapping from a flat surface parameterized by (u,v) to angular coordinates (θ,ϕ). Consider the sphere example where $r = r_o$ is constant so that

$$\int\limits_{sphere} dS = \int_0^{2\pi}\int_0^{\pi} r_o^2\sin\theta\,d\theta\,d\phi = \iint\limits_{cube} \frac{hr_o^2}{(u^2+v^2+h^2)^{3/2}}\,du\,dv \tag{6.15}$$

where h is perpendicular distance from center of sphere to a cube face, one can deduce that

$$|J_{map}| = \frac{h}{\sin\theta(u^2+v^2+h^2)^{3/2}} \tag{6.16}$$

when mapping is along radial direction from center of cube which coincides with center of sphere. We are now in a position to carry out surface integral on reference patches of a cube by combining equations (6.13), (6.14) and (6.16) where radius function in (6.6) is represented by spherical harmonics. The evaluation of r and r_ϕ is according to the definition

of real spherical harmonics in (6.3) and r_θ can be calculated using the following relation:

$$\frac{dP_n^m(\cos\theta)}{d\theta} = \frac{(n-|m|+1)P_{n+1}^{|m|} - (n+1)\cos\theta P_n^{|m|}}{\sin\theta}. \tag{6.17}$$

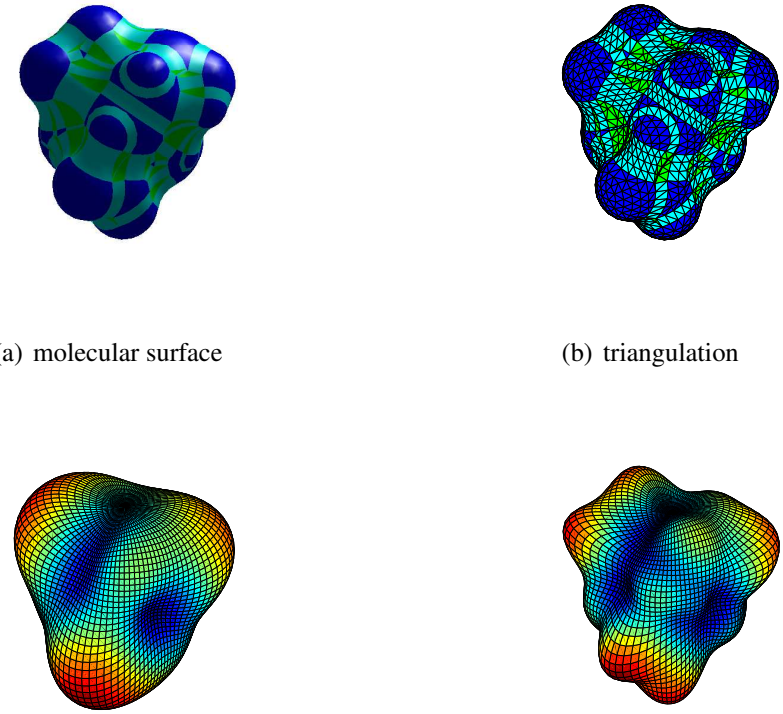
Therefore, when spherical harmonics are used to represent a smooth surface, the Jacobian in (5.5) is given by

$$|J| = |\overrightarrow{R}_\theta \times \overrightarrow{R}_\phi||J_{map}|. \tag{6.18}$$

6.4 Computational Results on Geometry Representation

6.4.1 A Biomolecule

In order to verify the above method for a general molecular surface, we use the example of a small organic molecule with 26 atoms, the transition state analog (TSA) of the protein enzyme chorismate mutase. The geometry of this small molecule was taken directly from an X-ray crystal structure [45], and can be obtained from the Protein Data Bank (PDB) [5] as accession number 1ECM. The radii used were 1.0 Å for hydrogens, 1.4 Å for oxygens, 2.0 Å for aliphatic carbons, and 1.7 Å for carbonyl or vinyl carbons. The surface of the TSA molecule was triangulated with the program MSMS [76], using a probe radius of 1.4 Å for water. A spherical harmonic representation is obtained by least squares fit to vertices of the triangulation. Fig. 6-3(c) and 6-3(d) shows an order 5 approximation with 36 coefficients and an order 10 approximation with 121 coefficients respectively for the surface using 3359 points, and a triangulation of 6714 panels. The color in the spherical harmonic surface correspond to the radial distance from the center of expansion, while the color in the triangulated surface correspond to the tori-reentrant, spherical-reentrant and contact surface in the definition of molecular surface. The area of a spherical harmonic surface can be calculated using (6.13), and is compared to analytical area given by MSMS for increasing order of approximation. In Fig. 6-4, the area convergence versus number of coefficients is shown for three sets of point distributions. The data demonstrated that 5 to 10 times as many points as coefficients can generate a reasonable approximation. For the



(a) molecular surface (b) triangulation

(c) order 5 expansion (d) order 10 expansion

Figure 6-3: Geometrical discretization of the TSA molecular surface.

order 10 expansion, the area incurs less than 0.3% relative error. The convergence seems to stagnate beyond 10^{-3} because MSMS data have precision of three decimal places, which is also the precision given in the experimental data of atomic coordinates.

6.4.2 An Ellipsoid

While one can represent an ellipsoidal surface analytically as in Section 5.4.3, we would like to investigate the effectiveness of spherical harmonic approximation, especially of an elongated shape. Figure 6-5 shows area convergence of spherical harmonic approximation of an ellipsoid with various aspect ratios. As shown in the plot, the larger the aspect ratio, the poorer the spherical harmonic representation for a given order and more basis functions have to be used in generating a reasonable approximation. In addition, more surface points will have to be used in generating an expansion of higher order, thus incurring higher

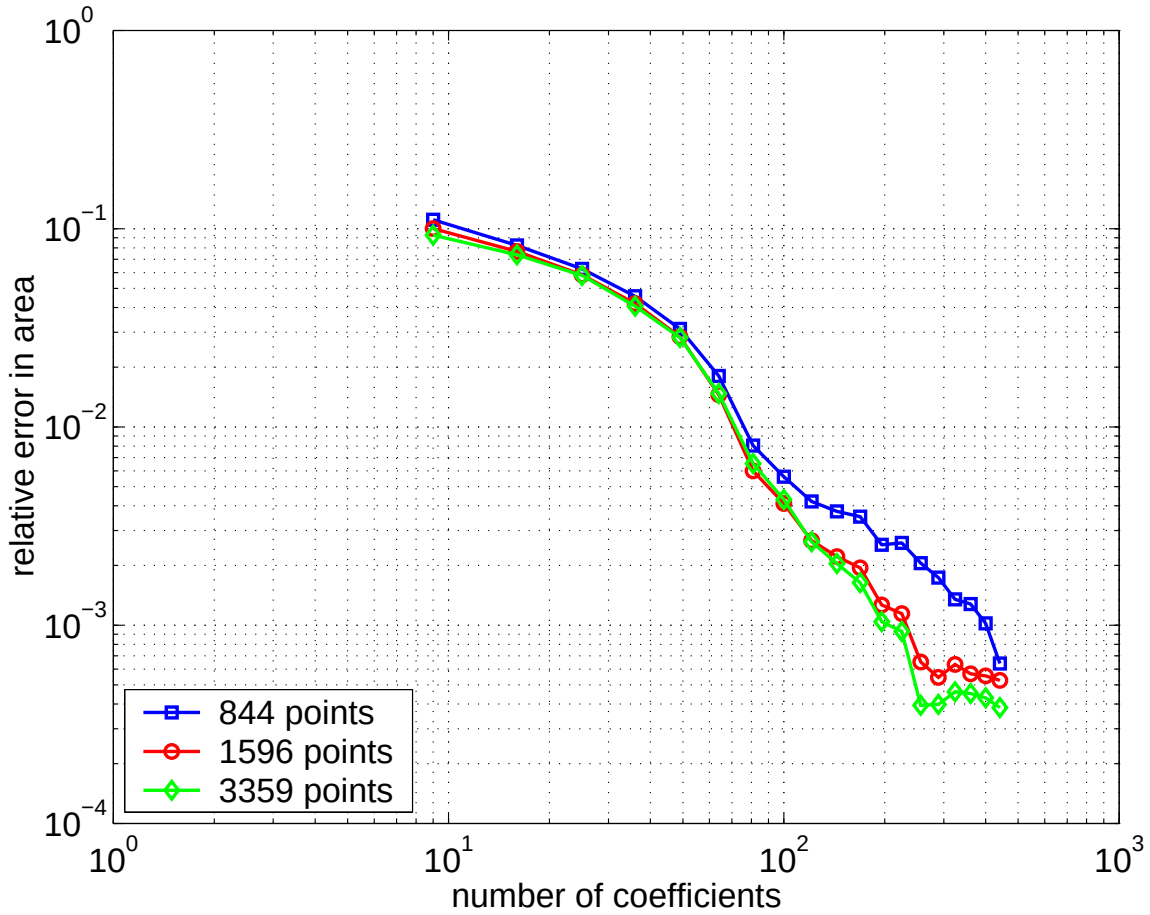


Figure 6-4: Area convergence of a TSA molecule using spherical harmonic approximation.

computational cost. On the other hand, different strategies [18] from approximating the radial distance as in (6.6), or alternative basis such as ellipsoidal harmonics may be useful in representing an elongated surface. But detail studies of those are left for future work.

6.4.3 Cusps

In the definition of molecular surface in 2.1, cusps can sometimes be generated when a probe sphere's reentrant surface intersects with itself. Such an example is shown in Figure 6-6: a singular edge results when two spherical triangular surfaces intersect each other and two singular vertices result when a tori-reentrant surface intersects with itself. In the case where geometrical singularity exists at a point where normal is not well-defined, the global representation using spherical harmonics will not be very effective. This is demon-

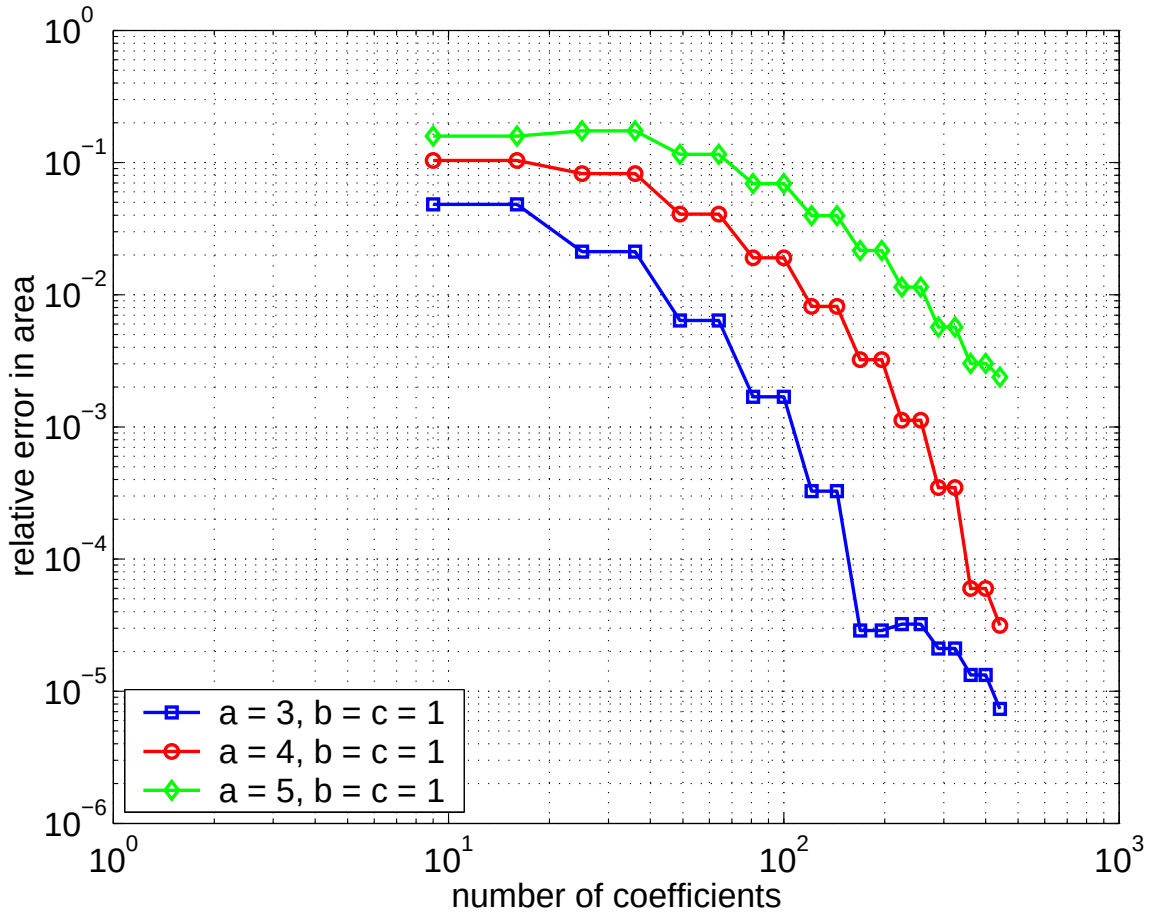


Figure 6-5: Area convergence of a spheroid using spherical harmonic approximation.

strated in Figure 6-7 where in the presence of a singular point in an otherwise smooth surface, the use of triangular panels gives better geometrical approximation than global spherical harmonic representation. As the order of expansion increases, spherical harmonic approximation becomes somewhat better in the smooth region, although ripples, similar to the Gibbs phenomenon, may place a limit on an achievable error tolerance. Furthermore, the expansion fails to capture the singular peak. It is important to note that such features are non-physical as they do not correspond to actual shapes of electron orbital (which are representable by spherical harmonics), but rather artifacts from algorithms of molecular surface generation. Nevertheless, if one wants to model such surface, additional pre-processing [85, 90] steps have to be carried out before using spherical harmonic expansion.

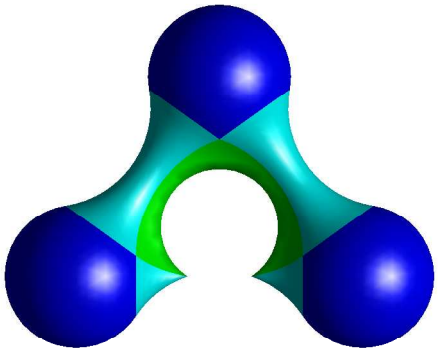


Figure 6-6: Geometrical singularities in a molecular surface.

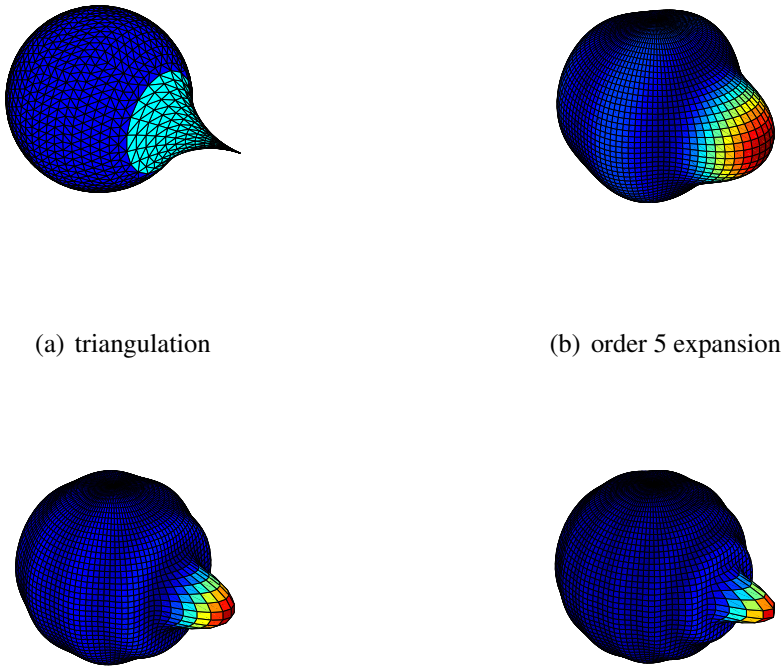
6.5 Computational Results using Approximate Geometry

6.5.1 Capacitance of an Ellipsoid

The same capacitance calculation as in Section 5.4.3 is carried out, but instead of representing geometry exactly with a mapping Jacobian, an approximate spherical harmonic surface is used instead. Figure 6-8 shows relative errors using the spectral method with exact and approximate geometry, as well as the standard panel method. The plots in Figure 5-6(b) have been reproduced here for a comparison. An order 15 expansion with 256 coefficients has been used to generate a surface representation which, according to Figure 6-5, has a relative error in area less than 10^{-4}. However, as shown in Figure 6-8, the amount of discretization error in geometry still incurs significant error in the capacitance calculation. Depending on the desired accuracy, a higher order surface representation may be obtained but the associated computational cost of adopting it in the spectral method will be higher.

6.5.2 Capacitance of a Biomolecule

Once a spherical harmonic surface is obtained, we can apply the spectral method to solve the integral equation in (4.7), where for the capacitance problem, the potential is set to unity. We can therefore compare our method to the standard panel method implemented in FastCap [59]. The spherical harmonic surface in Fig. 6-3(d) is used for geometrical representation in the spectral method, and triangulation from MSMS is used to generate



(a) triangulation (b) order 5 expansion

(c) order 10 expansion (d) order 15 expansion

Figure 6-7: Spherical harmonic expansion of a smooth surface with one singular vertex.

input files for the FastCap program. The capacitance calculation for the two solvers with increasing discretization is shown in Fig. 6-9. For the spectral method, the number of unknowns correspond to number of global lagrangian basis used while for the standard panel method, the number of unknowns correspond to number of panels in the triangulation. For the spectral method, the result converge to three significant figures with 386 unknowns while in the standard method, the same convergence can only be achieved with 27742 unknowns. The spectral method requires almost two orders of magnitude fewer unknowns for a tolerance of 10^{-3}, which is consistent with the sphere example in Figure 5-3.

6.5.3 Solvation Energy of a Biomolecule

For the same TSA molecule, we would like to calculate the solvation energy when the molecule is in an ionic solution. We use the formulation in Section 3.1 to obtain solution of

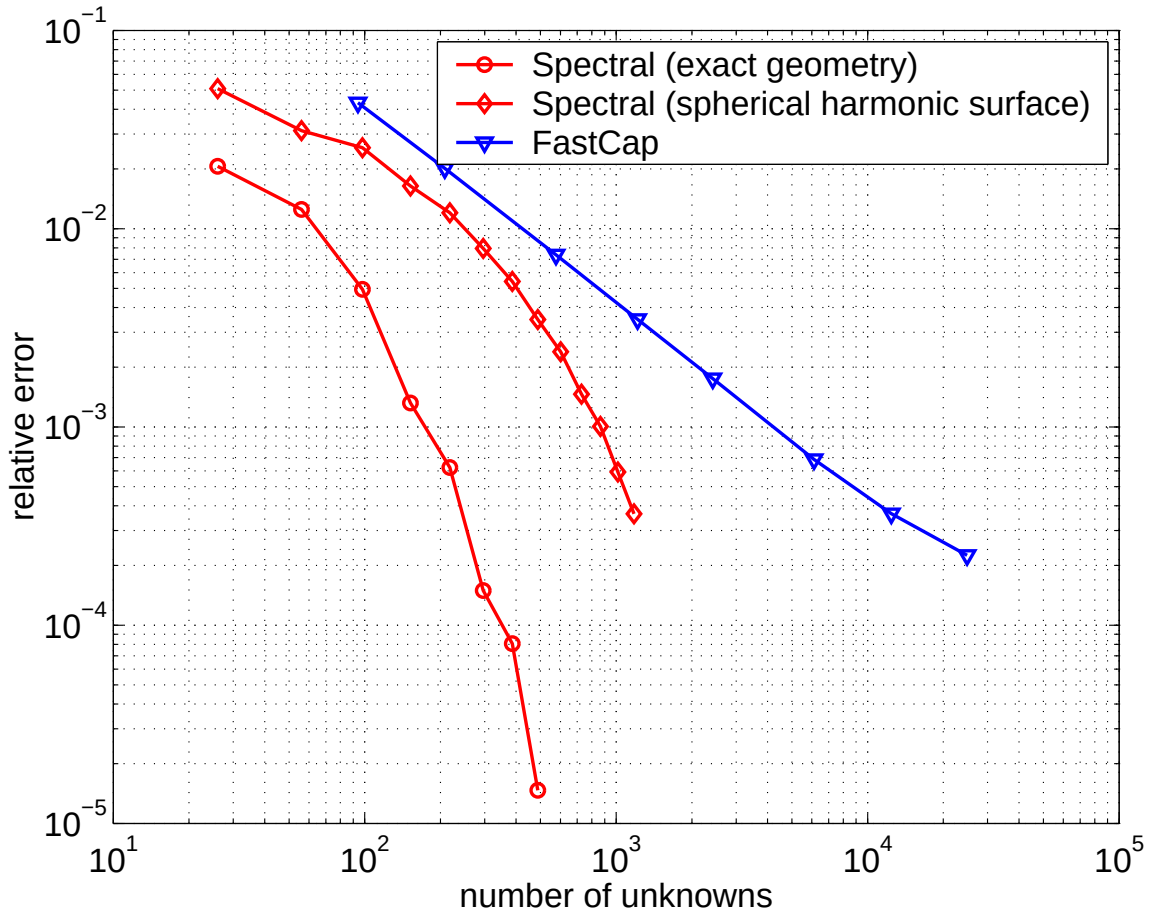


Figure 6-8: Capacitance calculation of a spheroid.

a linearized Poisson-Boltzmann equation. The coupled integral equations of interest are:

$$\begin{aligned} \frac{1}{2}\varphi_1(\vec{r}_o) \quad &+ \quad \int_\Omega \left[\varphi_1(\vec{r}') \frac{\partial G_1}{\partial n}(\vec{r}_o;\vec{r}') - G_1(\vec{r}_o;\vec{r}') \frac{\partial \varphi_1}{\partial n}(\vec{r}') \right] d\vec{r}' \\ &= \quad \sum_{i=1}^{n_c} \frac{q_i}{\varepsilon_1} G_1(\vec{r}_o;\vec{r}_i) \end{aligned} \tag{6.19}$$

and

$$\begin{aligned} \frac{1}{2}\varphi_1(\vec{r}_o) \quad &+ \quad \int_\Omega \left[-\varphi_1(\vec{r}') \frac{\partial G_2}{\partial n}(\vec{r}_o;\vec{r}') + G_2(\vec{r}_o;\vec{r}') \frac{1}{\varepsilon} \frac{\partial \varphi_1}{\partial n}(\vec{r}') \right] d\vec{r}' \\ &= \quad 0 \end{aligned} \tag{6.20}$$

where the unknown quantities are potential φ_1 at the dielectric interface and its normal derivative $\frac{\partial \varphi_1}{\partial n}$ on the inner surface. The normal derivative at the interface has a jump that is

related to the relative dielectric constant ε. The free charges, q_i, are derived from quantum mechanical calculations. The Green's functions are:

$$G_1(\vec{r};\vec{r}') = \frac{1}{4\pi|\vec{r}-\vec{r}'|} \tag{6.21}$$

$$G_2(\vec{r};\vec{r}') = \frac{e^{-\kappa|\vec{r}-\vec{r}'|}}{4\pi|\vec{r}-\vec{r}'|} \tag{6.22}$$

where $\kappa = 0.124$ Å^{-1}, equivalent to an ionic strength of 0.145 M at 25° C was used. A dielectric constant of 4 ε_0 was used inside the TSA molecule and a dielectric of 80 ε_0 was used externally. Once the potential and its normal derivative are computed on the molecular surface, potentials everywhere can be calculated. In particular, the potential at each charge location, known as the reaction potential, is given by

$$\varphi^{REAC}(\vec{r}_i) = \int_\Omega \left[G_1(\vec{r}_i;\vec{r}')\frac{\partial \varphi_1}{\partial n}(\vec{r}') - \varphi_1(\vec{r}')\frac{\partial G_1}{\partial n}(\vec{r}_i;\vec{r}') \right] d\vec{r}'. \tag{6.23}$$

The solvation energy can be calculated by multiplying these potentials with corresponding charge magnitudes.

The spectral method is again compared with the standard panel method implemented with precorredted-FFT acceleration [44, 63, 94]. The results are shown in Fig. 6-10. Note that the size of matrix equation is twice the size of the basis set shown on the x-axis, since there are two sets of unknowns in the coupled integral equations. This problem is also more challenging due to the presence of double layer potentials. To converge to three significant figures, the spectral method requires 488 basis functions while 8502 panels are needed, a factor of 20 improvement.

6.6 Discussion

This chapter extends the novel approach in discretizing integral equations with singular kernels, such as those associated with electrostatic analysis of molecular surfaces. While in the panel method both geometrical discretization and basis functions supports are defined on a mesh, they are decoupled in the proposed spectral method. On the other hand, mapping

functions are required to describe the geometry. In simulation of molecular electrostatics, spherical harmonics can be a good candidate for generating an analytic representation of molecular surfaces. Once a mapping from a cube becomes available, the solution on a general surface is no more difficult to obtain than that on a sphere. On the other hand, since these basis are smooth functions, molecular surfaces that contain cusps cannot be well approximated if one insists on representing these non-physical structures. Also, the expansion based on radial distances as described in this chapter is not possible for non-star geometries. Instead, more sophisticated approach based on past [18] or future work will have to be carried out.

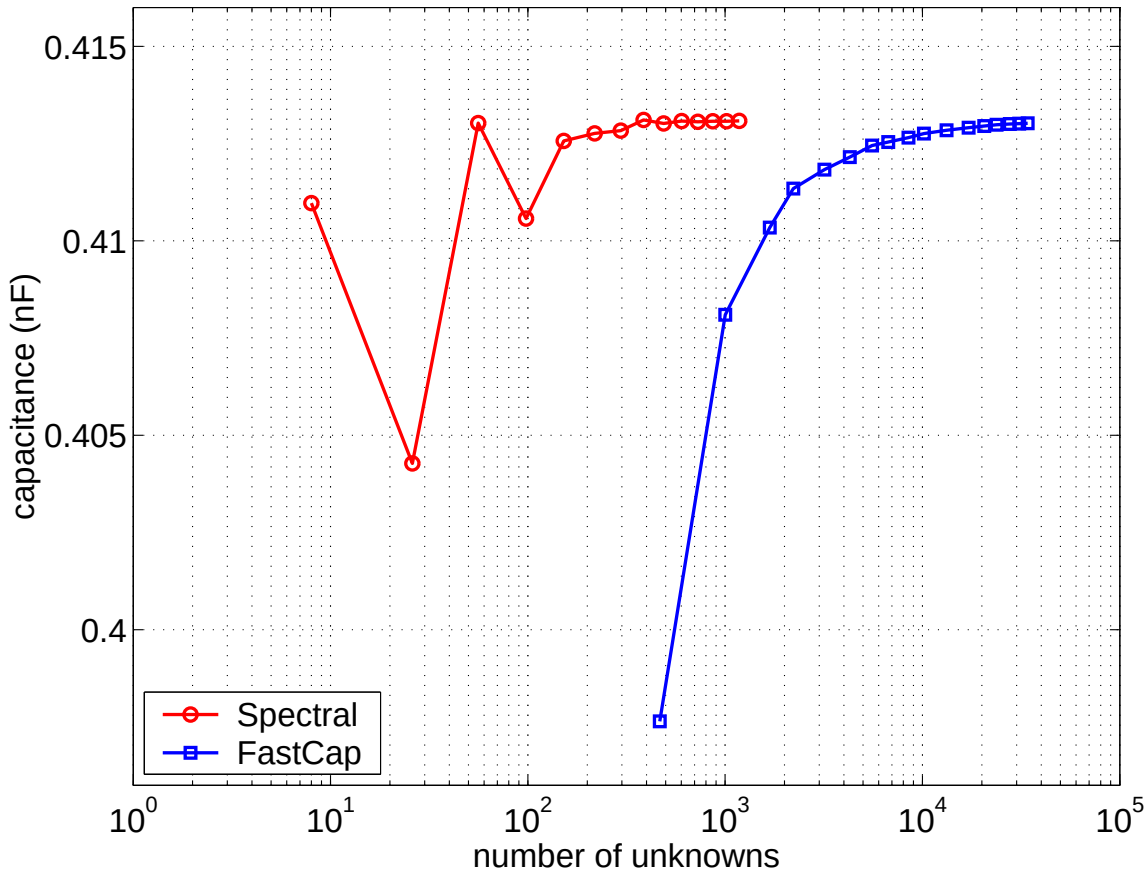


Figure 6-9: Capacitance calculation of the TSA molecule.

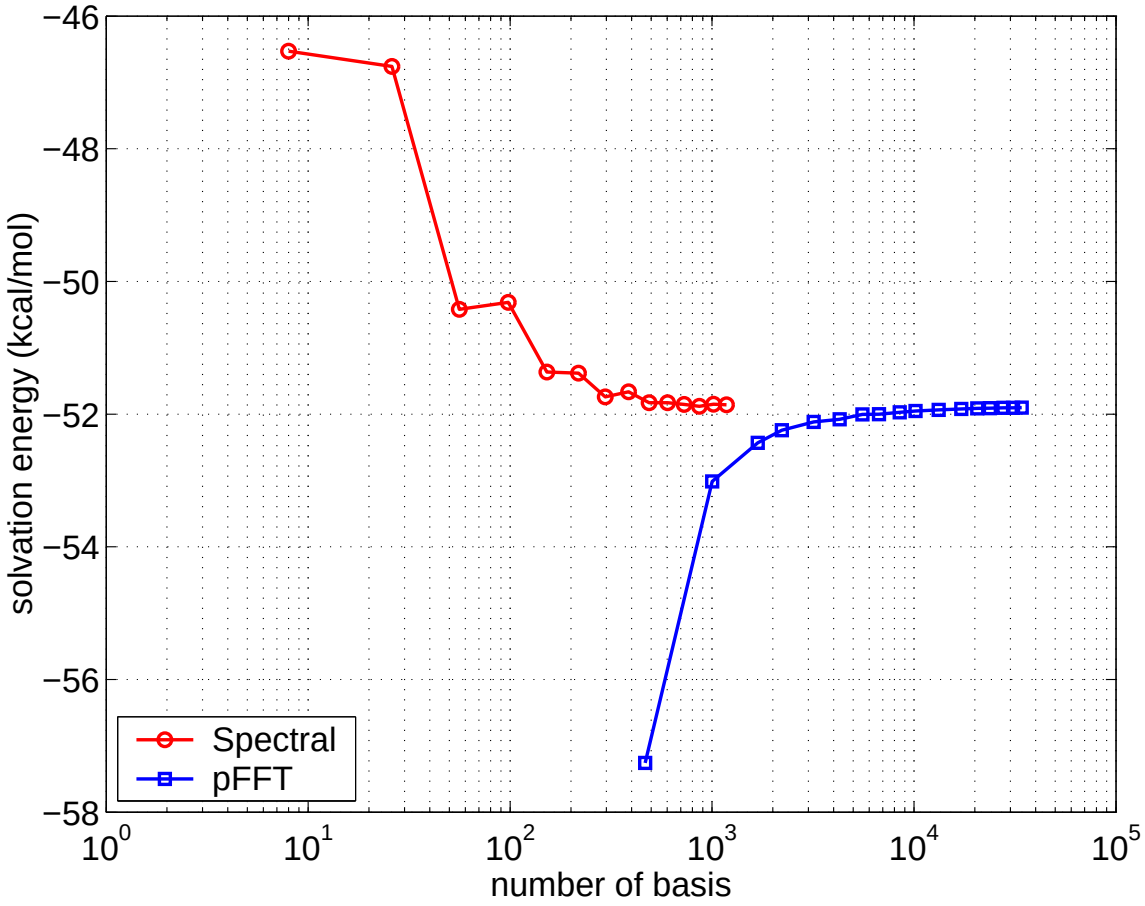


Figure 6-10: Solvation energy calculation of the TSA molecule.

Chapter 7

Implementation Details

7.1 Iterative Solver

The cost of constructing the collocation matrix depends on the number of quadrature points used in the polar coordinates. And the use of higher order basis requires a similarly higher order quadrature scheme in order to accurately approximate the integral. The use of quadrature points in computing panel integration tends to dominate the computation time. Since many basis functions share their supports on a patch, an efficient implementation should recycle quadrature points defined on (ρ, θ) among them. This is most easily illustrated by an iterative solver approach:

$$\begin{aligned}
\phi(\vec{r}_j) &= \sum_{i=1}^{n} A_{ji}^{Collocation} \sigma_i && (7.1)\\
&= \sum_{i=1}^{n} \sigma_i \left(\iint G(\vec{r}_j, \vec{r}'(\rho,\theta)) B_i(\rho,\theta) |J(\rho,\theta)| \, \rho \, d\rho \, d\theta \right) && (7.2)\\
&= \iint G(\vec{r}_j, \vec{r}'(\rho,\theta)) \left(\sum_{i=1}^{n} \sigma_i B_i(\rho,\theta) \right) |J(\rho,\theta)| \, \rho \, d\rho \, d\theta && (7.3)
\end{aligned}$$

where $\sigma_i = \sigma(\vec{r}_i)$ is test solution at collocation points. The summation over all patches within a basis function' support in implicity assumed here. As opposed to an direct solver whereby integration over patches is done for individual basis functions in (7.2), at each iteration step, a weighted sum of all basis functions in (7.3) is integrated instead. This

is equivalent to first interpolating on each patch via a set of Gauss-Lobatto points, then integrating the interpolated function over the corresponding global surface. In addition to the computing efficiency, an iterative solver uses less memory than a direct solver so larger problems may be solved.

The same efficiency may be achieved with a direct solver, but would require storage of all quadrature points used for panel integration shared among all basis functions on a patch. Since distribution of quadrature points changes depending on location of an evaluation point, this would require storing six sets of quadrature points (for six faces of a cube) for each evaluation point. On the other hand, pre-computing and storing the quadrature points could further improve the speed of an iterative solver since no redundant computation needs to be done at each iteration step. Further computational studies on the trade-off between memory and speed have to be carried out in a low-level language such as C or C++, but below we give preliminary performance results based on a code implemented in MATLAB® and optimized for memory: that is, no pre-computing and storing quadrature points for both direct and iterative solvers. Figure 7-1 shows computation time required for direct and iterative solvers of the sphere example in Section 5.4.1. The number of iteration required for GMRES [75] to converge is shown in Table 5.1. Iteration counts stay fairly constant as problem size increases even though no preconditioners are used for the iterative solver. As shown in the figure, computational time for a direct solver grows like $O(n^2)$ where n is number of basis or unknowns. This is because in the implementation in MATLAB®, the cost of panel integration dominates that of Gaussian elimination which has been pre-compiled. On the other hand, if quadrature points are used to integrate a sum of basis functions once per patch per collocation instead of an individual basis function repeatedly, the plot for the iterative solver shows that the cost of panel integration is less than $O(n^2)$. And even though such integration has to be done redundantly at each iteration step, an iterative solver is still faster than a direct solver for all but the smallest problems, when optimized for memory.

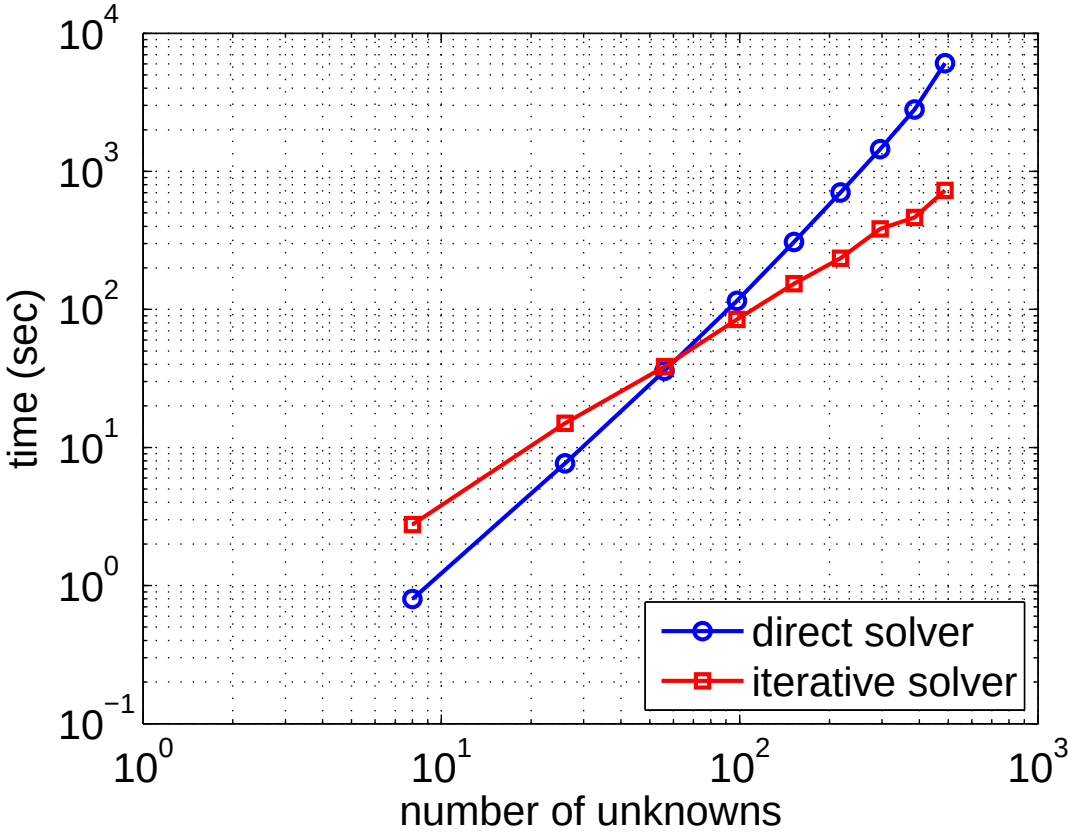


Figure 7-1: Efficiency comparison between direct and iterative solver.

7.2 Algorithm Steps

Below we give a summary of all the steps involved in the matrix-vector multiplication used in an iterative solver. We will assume that a spherical harmonic representation of geometry has been obtained, and the basis are defined using Gauss-Lobatto grids on each face of a cube. Given σ_i at collocation points, potentials at evaluation points can be computed as follows:

```
for each collocation point
  for each patch
```

1. Choose origin of polar coordinates on a patch according to (5.7), if evaluation point is on patch. Otherwise, choose the nearest point on patch as the origin.

2. Partition patch into triangles by connecting the origin to all vertices. Set up quadra-

ture points in polar coordinates for each triangle.

3. Evaluate basis at quadrature points by (5.1) and (5.3). The interpolated function at quadrature points are given by (2.6).

4. Evaluate Jacobian at quadrature points by (6.18), (6.16), (6.12) and (6.6).

5. Evaluate Green's function at quadrature points via projection of quadrature points according to (6.11) and (6.6).

6. Integrate on a reference triangle using the above functional evaluations at quadrature points and appropriate quadrature weights.

7. Calculate the integral on a patch in (7.3) by summing up contribution from each triangle.

```
  end
end
```

7.3 Complexity Analysis

To facilitate the complexity discussion, let the number of nodes per side of a square patch be m, so the basis are a set of two-dimensional polynomials of degrees $m-1$ in each of local (u,v) coordinates. And let the number of quadrature points in polar coordinates used for flat panel integration be k per triangle. Depending on the location of an evaluation point relative to a local patch, the integration may be carried out as a sum over 2, 3 or 4 triangles. The number of quadrature points required for a good approximation to the integral could depend on the order m of the basis functions, as well as the order of an spherical harmonic expansion in surface representation. To simplify the discussion, however, we will distinguish it with a separate variable and implicitly assume here that k has been chosen large enough to compute the integral to sufficient accuracy.

The dominant cost in the algorithm steps in Section 7.2 are those associated with polynomial interpolation in Step 3 and $O(k)$ function evaluations at integration points in Steps 4

and 5. As shown in [6], a one-dimensional Lagrangian interpolation costs $O(m)$ per evaluation point. Since the interpolation points are defined on a two-dimensional grid and there are k evaluation points, total cost associated with Step 3 is $O(km^2)$, which is the dominant cost per evaluation point per patch. The number of patches is kept constant at six for mapping from a cube and is small compared to the total number of collocation points, n, so the complexity of the algorithm is $O(knm^2) = O(kn^2)$, same as a straightforward matrix-vector multiplication procedure.

In comparison to a fast solver approach, the asymptotic complexity of the proposed method is less attractive. On the other hand, the number of unknowns needed to achieve good accuracy is a lot fewer with spectral convergence rates than the standard panel method. In such cases, the constant factor associated with the complexity is often more important. A distinct feature of the proposed spectral method is that quadrature points used for panel integration can be shared among many basis functions on a patch, and the number of patches is kept small and constant as number of unknowns increases. This allows the number of calls to a panel integration routine, a costly operation in any boundary element method implementation, to grow only with $O(n)$ as size of basis set increases. In addition, the method's efficiency can be further improved by pre-computing and storing almost all values associated with integration quadrature points (except those interpolated from grid nodes) at the expense of higher memory cost.

Chapter 8

Conclusion

In simulation of biomolecular electrostatics, the traditional and popular method is based on finite difference solution of differential formulation. This approach is relatively easy to implement and does not require a triangulation of molecular surface. However, these advantages come at the expense of inaccurate treatment of boundary conditions, namely those at infinity and at the interface between two dielectric mediums, as well as poor representation of point charges via nearby grid points. While much effort has been developed over the years to remedy these effects, a boundary element method approach based on integral formulation can resolve these difficulties much more elegantly.

Therefore, the first part of this thesis has focused on developing an integral formulation appropriate for the biomolecule application, namely those that can model the multiple-domain problem where each medium is governed by a distinct Green's function. While such a development is not entirely new and similar ideas can be borrowed from other areas, for example in interconnect simulation, most of earlier formulations in this application deal with non-ionic solutions. That is, the governing Green's function is the same in either side of a dielectric interface. The subsequent implementation with pre-corrected FFT implementation demonstrates that an numerical solution based on integral formulation is a viable alternative to finite difference solution.

However, discretization errors exist in both geometry and basis function representation in boundary element method. While the use of a surface mesh is an improvement over a volume grid, the number of flat panels required to accurately represent an inherently smooth

surface may still be excessive. The lack of high quality mesh generator in triangulating a molecular surface and the complexity involved in describing such a surface based on a seemingly straightforward definition also present another hurdle. Therefore, an attempt is made to discretize the integral formulation with curved panels and higher order basis functions. The standard approach using higher order polynomials in representing surface and unknowns can improve accuracy and convergence, but at higher computational cost. Panel integration is more expensive for curved panels. Furthermore, fast solver algorithms become less effective when higher order basis with larger supports are used. The trade-off is not necessarily in favor of using higher order discretization in the panel method, and most existing software tools based on boundary element method use low order discretization coupled with a fast solver implementation.

The main contribution of this thesis is in the development of the spectral method. It presents a novel approach to discretizing integral equations with singular kernels in the boundary element framework, yet the notion of panels has been replaced by patches. The basis functions are no longer defined on panels, but on a Gauss-Lobatto grid on a patch. Although there is substantial overlap in basis functions' supports, numerical orthogonality is ensured and solution maintains continuity across patch boundaries. The method is capable of achieving spectral convergence and requires many fewer unknowns for a given accuracy than the standard panel method. In order to apply the method to electrostatic analysis of molecular surfaces, a spherical harmonic analytic representation of the surface is generated and used to construct a mapping from local patches on a cube. The differentiable surface is the least-squares fitting to a given set of surface points, which is more readily available than a triangulation. Integration on a patch is done by quadrature in carefully chosen polar coordinates. And the cost associated with panel integration is kept small by using only a small number of patches (6 on a cube) and sharing quadrature points among all basis functions defined on a patch. While a more careful comparison of the computational costs between the spectral method and fast solver approach is yet to be carried out, initial results indicate that the proposed method is very efficient while having superior accuracy. A summary of the comparison between the spectral method and the standard panel method is shown in Table 8.1.

	Panel method	Spectral method
Basis	Piecewise constant Local Orthogonal	Lagrangian polynomial Global Numerically orthogonal
Geometry Representation	Mesh	Mapping functions Meshless
Accuracy	Algebraic convergence	Spectral convergence
Efficiency	O(n) with acceleration	see Section 7.3

Table 8.1: Comparison between panel and spectral methods.

While motivated by the biomolecular electrostatics problem, the spectral method developed can also be applied to other application areas as well. Unlike the panel method, however, the surface description will be in terms of spherical harmonic expansions or other mapping functions. And the method is most effective when underlying solution is smooth.

Bibliography

[1] K. Atkinson. *The Numerical Solution of Integral Equations of the Second Kind*. Cambridge University Press, 1997. 5

[2] K. E. Atkinson. The numerical solution of boundary integral equations. In I. Duff and G. Watson, editors, *The State of the Art in Numerical Analysis*, pages 223–259. Clarendon Press, 1997. 2.3.2

[3] J. P. Bardhan, J. H. Lee, S. S. Kuo, M. D. Altman, B. Tidor, and J. K. White. Fast methods for biomolecule charge optimization. In *Proceedings of the International Conference on Modeling and Simulation of Microsystems*, San Juan, Apr. 2003. 3.2

[4] A. Baxansky and N. Kiryati. Calculating geometric properties of three-dimensional objects from the spherical harmonic representation. Technical Report VIA-2005-6-1, Tel Aviv University, June 2005. 6.3

[5] H. M. Berman, J. Westbrook, Z. Feng, G. Gilliland, T. N. Bhat, H. Weissig, I. N. Shindyalov, and P. E. Bourne. The Protein Data Bank. *Nucleic Acids Research*, 28(1):235–242, 2000. 3.5.2, 6.4.1

[6] J.-P. Berrut and L. N. Trefethen. Barycentric lagrange interpolation. *SIAM Review*, 46(3):501–517, 2004. 5.1, 7.3

[7] G. Beylkin, R. Coifman, and V. Rokhlin. Fast wavelet transforms and numerical algorithms I. *Communications on Pure and Applied Mathematics*, 44:141–183, 1991. 3, 5, 5.5

[8] R. Bharadwaj, A. Windemuth, S. Sridharan, B. Honig, and A. Nicholls. The fast multipole boundary element method for molecular electrostatics: An optimal approach for large systems. *Journal of Computational Chemistry*, 16:898–913, 1995. 3

[9] A. Boschitsch, M. Fenley, and H.-X. Zhou. Fast boundary element method for the linear poisson-boltzmann equation. *Journal of Physical Chemistry B*, 106(10):2741–2754, 2002. 3

[10] C. Brechbühler, G. Gerig, and O. Kübler. Parametrization of closed surfaces for 3-D shape description. *Computer Vision and Image Understanding*, 61(2):154–170, 1995. 6, 6.2, 6.2

[11] C. L. Brooks, M. Karplus, and B. M. Pettitt. *Proteins: A Theoretical Perspective of Dynamics, Structure and Thermodynamics*. John Wiley & Sons, 1988. 1.1

[12] O. Bruno and L. Kunyansky. A fast, high-order algorithm for the solution of surface scattering problems: basic implementation, tests, and applications. *Journal of Computational Physics*, 169(1):80–110, 2001. 4.1, 5, 5.5.2

[13] B. Büchmann. Accuracy and stability of a set of free-surface time-domain boundary element models based on B-splines. *International Journal for Numerical Methods in Fluids*, 33(1):125–155, 2000. 5

[14] M. L. Connolly. Analytical molecular surface calculation. *Journal of Applied Crystallography*, 16:548–558, 1983. 2.1, 2.2

[15] M. E. Davis and J. A. McCammon. Electrostatics in biomolecular structure and dynamics. *Chemical Reviews*, 90:509–521, 1990. 1.1

[16] A. C. M. de Queiroz. Capacitance calculations. `http://www.coe.ufrj.br/~acmq/tesla/capcalc.pdf`. 5.4.3

[17] M. G. Duffy. Quadrature over a pyramid or cube of integrands with a singularity at a vertex. *SIAM Journal on Numerical Analysis*, 19(6):1260–1262, Dec. 1982. 4.1

[18] B. S. Duncan and A. J. Olson. Approximation and characterization of molecular surfaces. *Biopolymers*, 33(2):219–229, Feb. 1993. 6, 6.2, 6.2, 6.4.2, 6.6

[19] E. J. Garboczi. Three-dimensional mathematical analysis of particle shape using x-ray tomography and spherical harmonics: Application to aggregates used in concrete. *Cement and Concrete Research*, 32(10):1621–1638, Oct. 2002. 6.3

[20] M. K. Gilson, A. Rashin, R. Fine, and B. Honig. On the calculation of electrostatic interactions in proteins. *Journal of Molecular Biology*, 183:503–516, 1985. 1.1, 2.2, 2.3.1, 3.5.2

[21] M. K. Gilson, K. A. Sharp, and B. H. Honig. Calculating the electrostatic potential of molecules in solution: Method and error assessment. *Journal of Computational Chemistry*, 9:327–335, 1987. 1.1, 2.3.1, 2.3.1, 3.5.2

[22] J. A. Grant, B. T. Pickup, and A. Nicholls. A smooth permittivity function for Poisson-Boltzmann solvation methods. *Journal of Computational Chemistry*, 22:608–640, 2001. 1.1

[23] L. Greengard. *The Rapid Evaluation of Potential Fields in Particle Systems*. MIT Press, 1988. 3, 3.3, 5, 5.5, 5.5.1

[24] L. Greengard and V. Rokhlin. A fast algorithm for particle simulations. *Journal of Chemical Physics*, 73:325–348, 1987. 3.3

[25] W. Hackbusch and Z. P. Nowak. On the fast matrix multiplication in the boundary element method by panel clustering. *Numerische Mathematik*, 54:463–491, 1989. 3, 5, 5.5

[26] W. Hackbusch and S. Sauter. On numerical cubature of nearly singular surface integrals arising in BEM collocation. *Computing*, 52(2):139–159, 1994. 4.1

[27] R. F. Harrington. *Field Computation by Moment Methods*. MacMillan, 1968. 2.3.2, 3

[28] K. Hayami and C. Brebbia. Quadrature methods for singular and nearly singular integrals in 3-D boundary element method. In C. Brebbia, editor, *Boundary Elements*

X: the 10th International Conference on Boundary Elements, volume 1, pages 237–264. Plenum, Southampton, UK, Sept. 1988. 4.1

[29] Z. S. Hendsch, M. J. Nohaile, R. T. Sauer, and B. Tidor. Preferential heterodimer formation via undercompensated electrostatic interactions. *Journal of the American Chemical Society*, 123:1264–1265, 2001. 1.1

[30] J. L. Hess and A. M. O. Smith. Calculation of nonlifting potential flow about arbitrary three-dimensional bodies. *Journal of Ship Research*, 8(2):22–44, 1964. 2.3.2, 3, 4, 4.1, 4.1, 4.5

[31] R. W. Hockney and J. W. Eastwood. *Computer simulation using particles*. Adam Hilger, 1988. 2.3.1

[32] B. Honig and A. Nicholls. Classical electrostatics in biology and chemistry. *Science*, 268:1144–1149, 1995. 1.1

[33] A. Jean-Charles, A. Nicholls, K. Sharp, B. Honig, A. Tempczyk, T. F. Hendrickson, and W. C. Still. Electrostatic contributions to solvation energies: Comparison of free energy perturbation and continuum calculations. *Journal of the American Chemical Society*, 113:1454–1455, 1991. 1.1

[34] W. L. Jorgensen, J. Chandrasekhar, J. D. Madura, R. W. Impey, and M. L. Klein. Comparison of simple potential functions for simulating liquid water. *Journal of Chemical Physics*, 79:926–935, 1983. 3.5.2

[35] A. H. Juffer, E. F. F. Botta, B. A. M. V. Keulen, A. V. D. Ploeg, and H. J. C. Berendsen. The electric potential of a macromolecule in a solvent: A fundamental approach. *Journal of Computational Physics*, 97:144–171, 1991. 3

[36] E. Kangas and B. Tidor. Electrostatic specificity in molecular ligand design. *Journal of Chemical Physics*, 112:9120–9131, 2000. 1.1

[37] E. Kangas and B. Tidor. Electrostatic complementarity at ligand binding sites: Application to chorismate mutase. *Journal of Physical Chemistry*, 105:880–888, 2001. 1.1

[38] S. Kapur and D. Long. High-order Nyström schemes for efficient 3-D capacitance extraction. In *Proceedings of the IEEE/ACM International Conference on Computer-Aided Design*, pages 178–185, 1998. 5

[39] S. Kapur and D. E. Long. IES^3: A fast integral equation solver for efficient 3-dimensional extraction. In *Proceedings of the IEEE/ACM International Conference on Computer-Aided Design*, pages 448–455, San Jose, CA, 1997. 3, 5, 5.5

[40] M. A. Khayat and D. R. Wilton. Numerical evaluation of singular and near-singular potential integrals. *IEEE Transactoins on Antennas and Propagation*, 53(10):3180–3190, 2005. 4.1

[41] J. G. Kirkwood. Theory of solutions of molecules containing widely separated charges with special application to zwitterions. *Journal of Chemical Physics*, 2:351, 1934. 3.5.1

[42] I. Klapper, R. Hagstrom, R. Fine, K. Sharp, and B. Honig. Focusing of electric fields in the active site of Cu-Zn superoxide dismutase: Effects of ionic strength and amino-acid modification. *Proteins: Structure, Function, and Genetics*, 1:47–59, 1986. 1.1, 2.2, 2.3.1, 2.3.1, 3.5.2

[43] R. Klees. Numerical calculation of weakly singular surface integrals. *Journal of Geodesy*, 70(11):781 – 797, 1996. 4.1

[44] S. S. Kuo, M. D. Altman, J. P. Bardhan, B. Tidor, and J. K. White. Fast methods for simulation of biomolecule electrostatics. In *Proceedings of the IEEE/ACM International Conference on Computer-Aided Design*, San Jose, CA, Nov. 2002. 2.3.2, 3, 6.5.3

[45] A. Y. Lee, P. A. Karplus, B. Ganem, and J. Clardy. Atomic structure of the buried catalytic pocket of escherichia coli chorismate mutase. *Journal of the American Chemical Society*, 117(12):3627 – 3628, 1995. 3.5.2, 6.4.1

[46] B. Lee and F. M. Richards. The interpretation of protein structures: estimation of static accessibility. *Journal of Molecular Biology*, 55:379–400, 1971. 2.1

[47] L.-P. Lee and B. Tidor. Barstar is electrostatically optimized for tight-binding to barnase. *Nature Structural Biology*, 8:73–76, 2001. 1.1

[48] L.-P. Lee and B. Tidor. Optimization of binding electrostatics: Charge complementarity in the barnase-barstar protein complex. *Protein Science*, 10:362–377, 2001. 1.1

[49] S. Leicester, J. Finney, and R. Bywater. A quantitative representation of molecular surface shape. I: Theory and development of the method. *Journal of Mathematical Chemistry*, 16:315–341, 1994. 6, 6.2, 6.2

[50] S. E. Leicester, J. L. Finney, and R. P. Bywater. Description of molecular surface shape using fourier descriptors. *Journal of Molecular Graphics*, 6(2):104–108, 1988. 6, 6.2

[51] V. Lounnas, B. M. Pettitt, L. Findsen, and S. Subramaniam. A microscopic view of protein solvation. *Journal of Physical Chemistry*, 18:7157–7159, 1992. 1.1

[52] J. D. Madura, J. M. Briggs, R. C. Wade, M. E. Davis, B. A. Luty, A. Ilin, J. Antosiewicz, M. K. Gilson, B. Bagheri, L. Ridgway-Scott, and J. A. McCammon. Electrostatics and diffusion of molecules in solution: Simulations with the University of Houston Brownian Dynamics program. *Computer Physics Communications*, 91:57–95, 1995. 1.1

[53] H. D. Manier. *A Three Dimensional Higher Order Panel Method based on B-splines*. PhD thesis, Massachusetts Institute of Technology, Cambridge, MA, 1995. 4, 4.1, 4.5, 5

[54] M. Mascagni and N. A. Simonov. The random walk on the boundary method for calculating capacitance. *Journal of Computational Physics*, 195(2):465–473, 2004. 5.4.2

[55] N. L. Max and E. D. Getzoff. Spherical harmonic molecular surfaces. *IEEE Computer Graphics and Applications*, 8(4):42–50, 1988. 6, 6.2

[56] J. A. McCammon and S. C. Harvey. *Dynamics of Proteins and Nucleic Acids*. Cambridge University Press, Cambridge, 1987. 1.1

[57] R. J. Morris, R. J. Najmanovich, A. Kahraman, and J. M. Thornton. Real spherical harmonic expansion coefficients as 3D shape descriptors for protein binding pocket and ligand comparisons. *Bioinformatics*, 21(10):2347–2355, Feb. 2005. 6, 6.2

[58] K. Nabors, F. T. Korsmeyer, F. T. Leighton, and J. White. Preconditioned, adaptive, multipole-accelerated iterative methods for three-dimensional first-kind integral equations of potential theory. *SIAM Journal on Scientific Computing*, 15(3):713–735, 1994. 4.4

[59] K. Nabors and J. White. FastCap: A multipole accelerated 3-D capacitance extraction program. *IEEE Transactions on Computer-Aided Design*, 10(11):1447–1459, Nov. 1991. 3, 3.3, 4.5, 5, 5.4.2, 5.5, 6.5.2

[60] J. N. Newman. Distributions of sources and normal dipoles over a quadrilateral panel. *Journal of Engineering Mathematics*, 20:113 – 126, 1986. 4, 4.1, 4.1, 4.5

[61] A. Nicholls and B. Honig. A rapid finite difference algorithm, utilizing successive over-relaxation to solve the Poisson-Boltzmann equation. *Journal of Computational Chemistry*, 12:435–445, 1991. 1.1, 2.3.1, 3.5.2

[62] A. T. Patera. A spectral element method for fluid dynamics: Laminar flow in a channel expansion. *Journal of Computational Physics*, 54:468–488, June 1984. 5.1, 5.5.2

[63] J. R. Phillips and J. K. White. A precorrected-FFT method for electrostatic analysis of complicated 3-D structures. *IEEE Transactions on Computer-Aided Design*, 16(10):1059–1072, Oct. 1997. 1.1, 3, 3.3, 4.4, 4.5, 5, 5.5, 6.5.3

[64] D. Ramaswamy, W. Ye, X. Wang, and J. White. Fast algorithms for 3-D simulation. *Journal of Modeling and Simulation of Microsystems*, 1(1):77–82, Dec. 1999. 2.3.2, 3

[65] F. M. Richards. Areas, volumes, packing, and protein structure. *Annual Review of Biophysics and Bioengineering*, 6:151–176, 1977. 2.1, 2.2

[66] S. W. Rick and B. J. Berne. The aqueous solvation of water: A comparison of continuum methods with molecular dynamics. *Journal of the American Chemical Society*, 116:3949–3954, 1994. 1.1

[67] W. Rocchia, E. Alexov, and B. Honig. Extending the applicability of the nonlinear Poisson-Boltzmann equation: Multiple dielectric constants and multivalent ions. *Journal of Physical Chemistry B*, 105:6507–6514, 2001. 1.1, 2.3.1, 3.5.2

[68] W. Rocchia, S. Sridharan, A. Nicholls, E. Alexov, A. Chiabrera, and B. Honig. Rapid grid-based construction of the molecular surface and the use of induced surface charge to calculate reaction field energies: Applications to the molecular systems and geometric objects. *Journal of Computational Chemistry*, 23:128–137, 2002. 1.1, 2.3.1, 2.3.1, 3.5.2

[69] N. K. Rogers and M. J. Sternberg. Electrostatic interactions in globular proteins: Different dielectric models applied to the packing of α-helices. *Journal of Molecular Biology*, 174:527–542, 1984. 2.3.1

[70] V. Rokhlin. Rapid solution of integral equations of classical potential theory. *Journal of Computational Physics*, 60:187–207, 1985. 3, 5, 5.5

[71] V. Rokhlin. End-point corrected trapezoidal quadrature rules for singular functions. *Computers & Mathematics with Applications*, 20(7):51–62, 1990. 4.1

[72] J. E. Romate. *The Numerical Simulation of Nonlinear Gravity Waves in Three Dimensions using a Higher Order Panel Method*. PhD thesis, University of Twente, Netherlands, 1989. 4, 4.5

[73] A. E. Ruehli and P. A. Brennan. Efficient capacitance calculations for three-dimensional multi-conductor systems. *IEEE Transactoins on Microwave Theory and Techniques*, 21(2):76–82, 1973. 5.4.2

[74] Y. Saad. *Iterative methods for sparse linear systems*. Society for Industrial & Applied Mathematics, 2nd edition, 2003. 2.3.1

[75] Y. Saad and M. Schultz. GMRES: A generalized minimal residual algorithm for solving nonsymmetric linear systems. *SIAM Journal of Scientific and Statistical Computing*, 7:856–869, 1986. 3.3, 5.4.1, 7.1

[76] M. Sanner, A. J. Olson, and J. C. Spehner. Reduced surface: An efficient way to compute molecular surfaces. *Biopolymers*, 38:305–320, 1996. 2.1, 3.5.2, 6.2, 6.4.1

[77] K. A. Sharp and B. Honig. Electrostatic interactions in macromolecules: Theory and applications. *Annual Review of Biophysics and Biophysical Chemistry*, 19:301–332, 1990. 1.1

[78] I. Sloan. Superconvergence. In M. Golberg, editor, *Numerical Solution of Integral Equations*, pages 35–70. Plenum, New York, 1990. 4.4

[79] I. Stakgold. *Boundary Value Problems of Mathematical Physics*. MacMillan, 1967. 3.1

[80] J. Strain. Locally-corrected multidimensional quadrature rules for singular functions. *SIAM Journal on Scientific Computing*, 16(4):992–1017, 1995. 4.1

[81] C. Tanford and J. G. Kirkwood. Theory of protein titration curves I. general equations for impenetrable spheres. *Journal of the American Chemical Society*, 59:5333–5339, 1957. 1.1, 2.2

[82] J. C. F. Telles and R. F. Oliveira. Third degree polynomial transformation for boundary element integrals: Further improvements. *Engineering Analysis with Boundary Elements*, 13(2):135–141, 1994. 4.1

[83] L. N. Trefethen and D. Bau, III. *Numerical Linear Algebra*. Society for Industrial & Applied Mathematics, 1997. 6.2

[84] J. C. Vassberg. A fast surface-panel method capable of solving million-element problems. Aerospace Sciences Meeting and Exhibit, 35th, Reno, NV, Jan. 1997. AIAA Paper 97-0168. 3

[85] Y. N. Vorobjev and J. Hermans. SIMS: computation of a smooth invariant molecular surface. *Biophysical Journal*, 73(2):722–732, 1997. 6.4.3

[86] X. Wang, J. Newman, and J. White. Robust algorithms for boundary-element integrals on curved surfaces. In *Proceedings of the International Conference on Modeling and Simulation of Microsystems*, pages 473 – 476, San Diego, Mar. 2000. 4.1, 4.1

[87] J. Warwicker and H. C. Watson. Calculation of the electric potential in the active site cleft due to alpha-helix dipoles. *Journal of Molecular Biology*, 157:671–679, 1982. 1.1

[88] B. J. Yoon and A. M. Lenhoff. A boundary element method for molecular electrostatics with electrolyte effects. *Journal of Computational Chemistry*, 11:1080–1086, 1990. 3

[89] R. J. Zauhar and R. S. Morgan. The rigorous computation of the molecular electric potential. *Journal of Computational Chemistry*, 9:171–187, 1988. 3

[90] R. J. Zauhar and R. S. Morgan. Computing the electric potential of biomolecules: application of a new method of molecular surface triangulation. *Journal of Computational Chemistry*, 11(5):603–622, 1990. 6.4.3

[91] R. J. Zauhar and A. Varnek. A fast and space-efficient boundary element method for computing electrostatic and hydration effects in large molecules. *Journal of Computational Chemistry*, 17:864–877, 1996. 3

[92] Z. Zhou, P. Payne, M. Vasquez, N. Kuhn, and M. Levitt. Finite-difference solution of the Poisson–Boltzmann equation: Complete elimination of self-energy. *Journal of Computational Chemistry*, 17:1344–1351, 1996. 2.3.1

[93] Z. Zhu, J. Huang, B. Song, and J. White. Improving the robustness of a surface integral formulation for wideband impendance extraction of 3D structures. In *Proceedings of the IEEE/ACM International Conference on Computer-Aided Design*, pages 592–597, San Jose, CA, 2001. 4.1

[94] Z. Zhu, B. Song, and J. White. Algorithms in FastImp: A fast and wideband impedance extraction program for complicated 3-D geometries. In *Proceedings of the Design Automation Conference*, pages 712 –717, Anaheim, CA, June 2003. 2.3.2, 3, 6.5.3