Fast Solution of the Poisson-Boltzmann Equation by the Reduced Basis Method and Range-Separated Canonical Tensor Format

Dissertation
zur Erlangung des akademischen Grades

doctor rerum naturalium
(Dr. rer. nat.)

von **M.Sc. Cleophas Muganda Kweyu**

geb. am **30.04.1984** in Kakamega, Kenya

genehmigt durch die Fakultät für Mathematik
der Otto-von-Guericke-Universität Magdeburg

Gutachter: **Prof. Dr. Peter Benner**

Prof. Dr. Benjamin Stamm

PD Dr. Boris Khoromskij

eingereicht am: **30.04.2019**

Verteidigung am: **02.08.2019**

Fast Solution of the Poisson-Boltzmann Equation by the Reduced Basis Method and Range-Separated Canonical Tensor Format

Dissertation submitted for the academic degree of
a doctor of natural sciences

doctor rerum naturalium
(Dr. rer. nat.)

by **M.Sc. Cleophas Muganda Kweyu**

born on **30.04.1984** in Kakamega, Kenya

approved from the Faculty of Mathematics
Otto-von-Guericke-University Magdeburg, Germany

Referees: **Prof. Dr. Peter Benner**

Prof. Dr. Benjamin Stamm

PD Dr. Boris Khoromskij

Submitted on: **30.04.2019**

Defense on: **02.08.2019**

“Young man, in mathematics you don’t understand things. You just get used to them.”

– John von Neumann

PUBLICATIONS

This thesis is a conglomerate of research findings from the following articles, which are either published or submitted for publication.

Chapter 4 is an extended version of the following preprints

[18] Peter Benner, Venera Khoromskaia, Boris Khoromskij, Cleophas Kweyu, and Matthias Stein. Computing electrostatic potentials using regularization based on the range-separated tensor format. *arXiv:1901.09864*, (2019).

[99] Cleophas Kweyu, Venera Khoromskaia, Brois Khoromskij, Matthias Stein, and Peter Benner. Solution decomposition for the nonlinear Poisson-Boltzmann equation using the range-separated tensor format. *in preparation*, (2019).

Chapter 5 on the other hand, is an amalgamation of

[96] Cleophas Kweyu, Martin Hess, Lihong Feng, Matthias Stein, and Peter Benner. Reduced Basis Method for Poisson-Boltzmann Equation. *Eccomas Congress 2016 - Proceedings of the 7th European Congress on Computational Methods in Applied Sciences and Engineering*, 2 (2016), 4187-4195.

[16] Cleophas Kweyu, Lihong Feng, Matthias Stein, and Peter Benner. Fast Solution of the Linearized Poisson-Boltzmann Equation with nonaffine Parametrized Boundary Conditions Using the Reduced Basis Method. *arXiv:1705.08349*, (2018).

The following articles contribute to both of Chapter 4 and Chapter 5

[97] Cleophas Kweyu, Lihong Feng, Matthias Stein, and Peter Benner. Fast Solution of the Nonlinear Poisson-Boltzmann Equation Using the Reduced Basis Method and Range-Separated Tensor Format. *ScienceOpen Posters*, 10.14293/P2199-8442.1.SOP-MATH.OHIYQV.v1, 2018

[98] Cleophas Kweyu, Lihong Feng, Matthias Stein, and Peter Benner. Reduced basis method for the nonlinear Poisson-Boltzmann equation regularized by the range-separated canonical tensor format. *in preparation*, (2019).

ACKNOWLEDGEMENTS

This is a summary of research done over a five year period while a member of the computational methods in systems and control theory (CSC) group and the international Max Planck Research School (IMPRS) for Advanced Methods in Process and Systems Engineering graduate school at the Max Planck Institute (MPI) for Dynamics of Complex Technical Systems, which is affiliated to the Otto-von-Guericke-University Magdeburg, in Germany. First and foremost, I would like to thank the Almighty God for His grace and mercy during the entire journey to this thesis.

Consequently, many people and institutions have provided immense support which enabled me to bring my research to a fruitful conclusion. Thanks. I explicitly thank my thesis advisor, Prof. Dr. Peter Benner for affording me the opportunity to pursue my doctoral studies in his research group, the CSC. Many thanks too for his diligent supervision and guidance besides the funding of my various business trips to conferences and workshops within and outside of Germany. As a research group leader, he epitomizes the spirit of good leardership and kindness.

I am deeply indebted to Prof. Dr. Boris Khoromskij and Dr. Venera Khoromskaia from the Max Planck Institute for Mathematics in the Sciences in Leipzig, Germany, for the collaborations and regular advice on the range-separated (RS) tensor methods and the related Matlab program support. To Dr. Matthias Stein and Dr. Lihong Feng from the MPI in Magdeburg, I am grateful for your regular advice and mentorship in the realms of biomolecular modeling and the reduced basis methods, respectively. All these techniques helped to shape the research scope of this thesis. To my colleagues at the CSC research group, the impromptu discussions we had whenever I ran into trouble with certain scientific computing problems were very significant in solving my various technicalities. I also thank Manuela Hund for translating my abstract into German. Thanks.

I am overwhelmingly grateful to the following institutions: The International Max Planck Research School (IMPRS) for Advanced Methods in Process and Systems Engineering and Max Planck Society for the Advancement of Science (MPG) for the financial and material support during PhD studies and Moi University, Department of Mathematics and Physics, Main Campus, in Eldoret (Kenya), for granting me a generous study leave which enabled me to efficiently pursue my PhD in Germany.

A special profound acknowledgement goes to my lovely wife, Brigid Chimoita Aliero, for her indispensable forbearance, love and emotional support during my doctoral studies. Family and study-life-balance was quite a challenge, but we never tired to this far. To my beloved children Gregory, Pascal and Laura, your endurance and loneliness during my frequent business trips, besides the lack of full attention during my thesis writing was overwhelming. To you I dedicate this thesis.

Finally, I owe wholehearted gratitude to my lovely parents, Dismas Kweyu Malala-

son and Apolonia Muganda Kweyu for inculcating in me the spirit of hard work and determination, and their abundant love and care.

ABSTRACT

We consider the fast and accurate numerical solution of the Poisson-Boltzmann equation (PBE), a three-dimensional second-order nonlinear elliptic parametrized partial differential equation (PPDE), which is ubiquitous in biophysics. It is an implicit solvent continuum model for calculating the electrostatic potential and energies of ionic solvated biomolecules. However, its numerical solution encounters severe challenges arising from: (i) strong singularities caused by the singular source terms and described by Dirac delta distributions; (ii) rapid nonlinearities caused by the exponential nonlinear terms; (iii) three spatial dimensions which lead to a high number of degrees of freedom in the resultant algebraic system of equations of size $\mathcal{O}(10^5) - \mathcal{O}(10^6)$ resulting from large domains, necessary to accommodate large sizes of macromolecules and for accurate approximation of boundary conditions due to the slow polynomial decay of the electrostatic potential in the form of $1/\|\bar{x}\|$ as $\bar{x} \to \infty$; and (iv) computationally expensive PBE simulations, for example, the Brownian dynamics simulations, whereby the PBE requires to be solved multiple times for a large number of system configurations. In this thesis, we for the first time eliminate the effect of strong singularities by applying the novel range-separated (RS) canonical tensor format [15, 17] for the construction of an efficient regularization scheme for the PBE. The RS tensor format allows to derive a smooth approximation to the Dirac delta distribution introduced in [85] in order to construct a regularized PBE (RPBE) model which computes smooth long-range electrostatic potentials, thereby circumventing the building of numerical approximations to the singular solution, resulting in increased accuracy. Consequently, the main computational benefits are due to the localization of the regularized Dirac delta distributions within the molecular region and automatic maintaining of the continuity in the Cauchy data on the solute-solvent interface [85]. The total electrostatic potential is obtained by adding the regularized long-range solution to the directly precomputed short-range potential component determined from the RS splitting of the Newton potential. Furthermore, to accelerate the computations resulting from challenges (iii) and (iv), we for the first time employ the reduced basis method (RBM) to substantially reduce the computational complexity by constructing a surrogate reduced order model (ROM) of typically low dimension, whose solution accurately approximates the original PBE. Moreover, the discrete empirical interpolation (DEIM), is applied to the parametric nonaffine Dirichlet boundary conditions (of Yukawa potential type) in order to significantly reduce the computational complexity of the nonaffine function by interpolation, whereby only a few entries are computed. On examples of several biomolecules, we demonstrate in the numerical experiments, the accuracy and efficacy of the RBM approximation for the nonlinear RPBE and the corresponding computational savings over the classical PBE.

ZUSAMMENFASSUNG

Wir beschäftigen uns mit der schnellen und genauen numerischen Lösung der Poisson-Boltzmann-Gleichung (PBG), einer dreidimensionalen nichtlinearen elliptischen parametrischen partiellen Differentialgleichung (PPDGL) zweiter Ordnung, die in der Biophysik allgegenwärtig ist. Diese ist ein implizites kontinuumsmechanisches Modell um das elektrostatische Potential und die Energie der ionisch gelösten Biomoleküle zu berechnen. Allerdings treten schwierige Herausforderungen beim numerischen Lösen auf: (i) starke Singularitäten, die durch den singulären Quellterm, der durch die Dirac Deltaverteilung beschrieben wird, entstehen; (ii) schnelle Nichtlinearitäten, die durch den exponentiellen, nichtlinearen Term hervorgerufen werden; (iii) drei räumliche Dimensionen, die zu einer Vielzahl an Freiheitsgraden in dem resultierenden algebraischen Gleichungssystem der Größe $\mathcal{O}(10^5) - \mathcal{O}(10^6)$ führen, welches aus einem großen Gebiet resultiert, das nötig ist um große Makromoleküle unterzubringen und um die Randbedingungen aufgrund eines langsamen polynomiellen Abfalls des elektrostatischen Potentials in Gestalt von $1/\|\bar{x}\|$ als $\bar{x} \to \infty$ genau approximieren zu können; und (iv) rechenintensive PBG Simulationen, wie zum Beispiel Brownsche Dynamiksimulationen, bei denen die PBG mehrfach für viele Systemeinstellungen gelöst werden muss. In dieser Dissertation werden wir zum ersten Mal den Effekt der starken Singularitäten entfernen, in dem wir das neuartige bereichstrennende kanonische Tensorformat [15, 17] zur Konstruktion eines effizienten Regularisierungsschemas für die PBG anwenden. Mit diesem Tensorformat kann eine glatte Approximation der Dirac Deltaverteilung, die in [85] eingeführt wurde, hergeleitet werden, um ein regularisiertes PBG (RPBG) Modell zu konstruieren, das glatte elektrostatische Potentiale berechnet und dabei verhindert, dass numerische Approximationen zur singulären Lösung erstellt werden, was zu einer erhöhten Genauigkeit führt. Folglich sind die wichtigsten Rechenvorteile darauf zurückzuführen, dass die regularisierte Dirac Deltaverteilung auf das molekulare Gebiet beschränkt ist und die Stetigkeit in den Cauchy-Daten auf dem Lösungsmittel-Interface [85] automatisch erhalten werden. Das gesamte elektrostatische Potential wird dann durch Hinzufügen der regularisierten weitreichenden Lösung zur vorher berechneten kurzzeitigen Potentialkomponente, die durch das bereichstrennende Aufteilen des Newton-Potentials bestimmt wird, berechnet. Um die Berechnungen, die aus den Herausforderungen (iii) und (iv) resultieren, außerdem zu beschleunigen, haben wir zum ersten Mal die reduzierte Basen Methode (RBM) angewandt, um im Wesentlichen den Rechenaufwand zu reduzieren. Dafür wird ein Modell reduzierter Ordnung (ROM) konstruiert, dessen Lösung die originale PBG exakt approximiert. Darüber hinaus wird die diskrete empirische Interpolation (DEIM) auf die parametrischen nicht-affinen Dirichlet-Randbedingungen (von Yukawa Potentialart) angewandt um den Rechenaufwand der nicht-affinen Funktion durch Interpolation zu verringern, wobei nur wenige Einträge berechnet werden. Anhand von Beispielen verschiedener Biomoleküle werden wir die Genauigkeit und Effizienz der RBM Approximation für die nichtlineare RPBG und die zugehörigen rechnerischen Einsparungen verglichen mit der klassischen PGB aufzeigen.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ALGORITHMS

NOTATIONS

Acronyms and abbreviations

PBE	Poisson-Boltzmann equation
PDE	partial differential equation
LPDE	linear partial differential equation
RPBE	regularized Poisson-Boltzmann equation
NRPBE	nonlinear regularized Poisson-Boltzmann equation
LRPBE	linear regularized Poisson-Boltzmann equation
BVP	boundary value problem
RBM	reduced basis method
EVD	eigenvalue decomposition
SVD	singular value decomposition
FDM	finite difference method
FEM	finite element method
MinRes	minimal residual
MATLAB	Matrix Laboratory software from the MathWorks Inc.
MOR	model order reduction
ODE	ordinary differential equation
PDE	partial differential equation
POD	proper orthogonal decomposition
CCT	cummulated canonical tensor
RS	range-separated
C2T	canonical-to-Tucker
T2C	Tucker-to-canonical

Differentiation and integration

α	a multi-index; the d-tuple $\alpha = (\alpha_1, \ldots, \alpha_d)$, $\alpha_i \in \mathbb{Z}^+$
$\lvert\alpha\rvert$	$\sum\limits_{i=1}^{d} \alpha_i$
$D^\alpha u$	$\frac{\partial^{\lvert\alpha\rvert} u}{\partial x_1^{\alpha_1} \cdots \partial x_d^{\alpha_d}}$

$D_i u$	$\frac{\partial u}{\partial x_i}$
∇	$(D_1, \ldots, D_d)$
$\nabla \cdot \mathbf{u}$	The divergence of a vector function, i.e., $\sum\limits_{i=1}^{d} D_i u_i$
$\int_\Omega u\, dx$	Volume integration of a function $u(x)$ over a set Ω

Sets and spaces

$\mathbb{R}$	field of real numbers
$\mathbb{Z}$	field of integers
$\mathbb{R}^d$	Euclidean d-space
$\bar{x}$	a point $\bar{x} \in \mathbb{R}^d$, $\bar{x} = (x_1, \ldots, x_d)$, where $x_i \in \mathbb{R}$
$\Omega \in \mathbb{R}^d$	Ω is a bounded subset of $\mathbb{R}^d$
$\partial\Omega$	the boundary of $\Omega \subset \mathbb{R}^d$
H	Hilbert space
$L^p(\Omega)$	Lebesgue space of p-th power integrable functions
$C^k(\Omega), C_0^k(\Omega)$	spaces of k-times continuously differentiable functions
$W^k(\Omega)$	space of k-times weakly differentiable functions
$W^{k,p}(\Omega), W_0^{k,p}(\Omega)$	Sobolev spaces associated with $W^k(\Omega)$ and $L^p(\Omega)$
$H^k(\Omega), H_0^k(\Omega)$	Sobolev spaces $W^{k,2}(\Omega)$ and $W_0^{k,2}(\Omega)$

Norms

$\lVert \cdot \rVert_{L^p(\Omega)}$	Norm in $L^p(\Omega)$, i.e., $\lVert v \rVert_{L^p(\Omega)} \equiv \left(\int_\Omega \lvert v \rvert^p dx\right)^{\frac{1}{p}}, \quad 1 \leq p < \infty$
$\lVert \cdot \rVert_{L^\infty(\Omega)}$	$\lVert v \rVert_{L^\infty(\Omega)} \equiv \operatorname*{ess\,sup}\limits_{x \in \Omega} \lvert v(x) \rvert, \quad p = \infty$
$\lVert \cdot \rVert_{H^k(\Omega)}$	norm in $H^k(\Omega)$, i.e., $\lVert v \rVert_{H^k(\Omega)} \equiv \left(\sum\limits_{\lvert \alpha \leq k \rvert} \int_\Omega \lvert D^\alpha v \rvert^2 dx \right)^{\frac{1}{2}}$
$(\cdot, \cdot)_{H^k(\Omega)}$	inner product in $H^k(\Omega)$, i.e., $(w, v) \equiv \sum\limits_{\lvert \alpha \leq k \rvert} \int_\Omega D^\alpha w D^\alpha v dx$
$\lVert \cdot \rVert_{W^{k,p}(\Omega)}$	$\lVert v \rVert_{W^{k,p}(\Omega)} \equiv \left(\sum\limits_{\lvert \alpha \leq k \rvert} \int_\Omega \lvert D^\alpha v \rvert^p dx \right)^{\frac{1}{p}}, \quad 1 \leq p < \infty$
$\lVert \cdot \rVert_{W^{k,\infty}(\Omega)}$	$\lVert v \rVert_{W^{k,\infty}(\Omega)} \equiv \max\limits_{\lvert \alpha \leq k \rvert} \operatorname*{ess\,sup}\limits_{x \in \Omega} \lvert D^\alpha v(x) \rvert, \quad p = \infty$

Matrices

a_{ij}	the i, j-th entry of A
$A(i:j, :)$, $A(:, k:\ell)$	rows $i, \ldots, j$ of A and columns $k, \ldots, \ell$ of A

$A(i:j, k:\ell)$	rows $i, \ldots, j$ of columns $k, \ldots, \ell$ of A
A^T	the transpose of A
A^{-1}	inverse of nonsingular A
A^{-T}	inverse of A^T
$\sigma_{\min}(A)$	minimum singular value of A

CHAPTER 1

INTRODUCTION

Contents

1.1. Motivation

Electrostatic interactions play a significant role in biological processes involving macromolecules such as proteins and nucleic acids in solutions. These interactions help to understand and determine the biological structure, their dynamics, and function of biomolecules. For instance, these long-range interactions are responsible for protein folding and stability, enzyme catalysis, recognition of substrates by receptors, and pH-induced conformational changes [34, 41, 71, 107]. Figure 1.1 shows the structure of the *Chymotrypsin inhibitor 2* (a protein found in the albumin fraction of seeds from the *Hyproly* strain of barley [76]) before and after folding. Note that the folding, which gives the protein its 3-dimensional structure, is due to the electrostatic interactions of the amino acids [48, 107].

Efficient and accurate modeling of these interactions hitherto remains a great challenge in computational biology due to the complexity of biomolecular systems dominated by the effects of solvation on biomolecular processes and by the long-range intermolecular interactions. Precise and faster calculation of electrostatic interaction between molecules has been a challenge until recent times. Techniques based on the Ewald summation method, such as the particle mesh Ewald (PME) technique, are used to efficiently calculate the electrostatic interactions, for example, the interaction energy and the interparticle forces, of a periodic or a pseudoperiodic system [137, 146]. However, these approaches compute the electrostatic potential only at the atomic positions and, therefore, the Ewald sums need to be meshed up over a 3D Cartesian grid onto an $n \times n \times n$ mesh for typical biomolecular simulations [38].

The Poisson-Boltzmann equation (PBE) is one of the physically most profound approaches to calculate the electrostatic potential within and around a solvated biomolecule. However, the following challenges are a bottleneck towards the application of the PBE to biophysical modeling, as far as efficiency and accuracy are concerned.

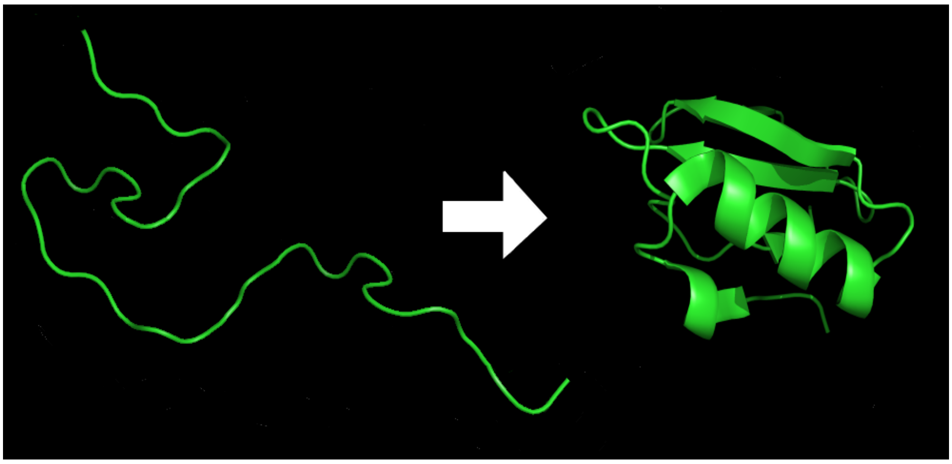


Figure 1.1.: Protein *Chymotrypsin inhibitor 2* before and after folding, (photo courtesy of [2]).

First, the strong singularities caused by the pointwise nuclei and described by the Dirac delta distributions indicate that the numerical solution is not continuous and does not belong to $H^1(\Omega)$, and also introduce significant errors in the numerical approximation of the PBE. The existing PBE solution decomposition techniques apply the splitting of the Laplace operator at the solute-solvent interface using the dielectric coefficients as the cutoff function in order to construct a regularized PBE. However, this approach leads to discontinuities of the solution at the interface, resulting from the truncation of the long-range potential component, thereby creating the need to apply interface conditions for the potential function.

Secondly, a typical macromolecule consists of approximately from $\mathcal{O}(10^3)$ to $\mathcal{O}(10^6)$ atoms (i.e., point charges in the PBE). Coupled with the fact that the electrostatic potential exhibits a slow polynomial decay in $1/\|\bar{x}\|$, large computational domains are necessary for an accurate numerical approximation of boundary conditions, which can be prohibitively expensive and an onerous task to the current computing resources.

Thirdly, incorporating solutions of the full PBE in a typical Brownian dynamics (BD) or molecular dynamics (MD) simulation for molecular association or dissociation could involve $\mathcal{O}(10^7)$ of steps [108]. For an efficient performance of the propagation of molecules and atoms, a single solution of the PBE has to be completed within a split second on the modern PC. Last but not least, the exponential rapid nonlinearities, discontinuous coefficients and infinite domain also contribute essentially to the numerical complexity of the PBE.

This thesis addresses the issue of strong singularities by applying the profound advantage of the range-separated (RS) tensor format introduced in [17] and the RS decomposition of the Dirac delta distribution [85] to the PBE in order to efficiently construct a regularized PBE which is free from the solution singularities. In this regard, we split the long- and short-range components of a multiparticle system in the respective atomic volume rather than at the interface between solute (molecule) and the solvent as done by the existing solution decomposition techniques for the PBE.

The reduced basis method (RBM) and the discrete empirical interpolation method (DEIM), on the other hand, are used to reduce the computational complexity of the

PBE by constructing an accurate surrogate reduced order model (ROM), and of the ROM by interpolating the nonaffine (in parameter) Dirichlet boundary conditions, thereby greatly accelerating the numerical computations of the PBE in a multi-query context (or in multi-parameter simulations).

1.2. Outline of the Thesis

The thesis is structured as follows: In Chapter 2, we first briefly study the significance of the PBE in biophysics. Secondly, we demonstrate the derivation of the PBE model from Gauss's law, the divergence theorem (or Gauss's theorem) and the Boltzmann distribution law, respectively. This is followed by the analysis of the existence and uniqueness of the PBE solutions for both the linearized and the nonlinear models. Thirdly, applications of the electrostatic potential are briefly mentioned and the finite difference discretization of the PBE is provided. Lastly, we highlight the recent advances regarding the regularization techniques for the PBE.

In Chapter 3, we provide an introduction to tensor methods by first outlining the basic concepts of multilinear algebra, which provide the foundation for the rank-structured tensor representations of function related tensors. We also provide some basics of tensor numerical methods, including the canonical-to-Tucker (C2T) algorithm and the generation of a canonical tensor representation for the Newton kernel. Then we discuss the range-separated tensor format, which was developed and analyzed in [15, 17] and describe the construction of the RS tensor format for efficient representation of the collective electrostatic potential of multiparticle systems of general type. The RS tensor format represents the total potential of a large number of charged particles as a sum of a low-rank canonical tensors representing the long-range part of the collective potential (with a rank only logarithmically depending on the number of particles) and a cumulative canonical tensor (CCT) representing the short-range potential of all particles.

Chapter 4 provides the main contribution of this thesis to the PBE theory as far as the solution decomposition techniques for the PBE are concerned. We first construct an efficient regularization scheme for the Poisson equation (PE) by the RS tensor format as a proof of concept [18]. We then extent this efficient regularization approach to both the linearized PBE (LPBE) and the nonlinear PBE (NPBE) in order to construct the corresponding regularized LPBE (LRPBE) and the NRPBE models [99]. In the LRPBE and NRPBE, we solve for the smooth long-range electrostatic potential, by applying only the smooth component of the source terms, obtained by the RS decomposition of the Dirac delta distribution [85]. We finalize this chapter by proving that the electrostatic solvation free interaction energy and forces (for solvated biomolecules) do not depend on the short-range component of the electrostatic potential, but they are completely determined by the long-range electrostatic potential component [99]. Note that the results for the interaction energy derived from the free-space potential, were introduced and proved in [17].

Chapter 5 provides another main contribution of this thesis to the PBE theory. We begin by studying the basics of the reduced basis methods. Then we apply the RBM to the classical linearized PBE as proof of concept and highlight some of the computational difficulties associated with the construction of an efficient reduced order model and how to circumvent these difficulties using the discrete empirical interpolation method (DEIM). The *a posteriori* error estimation for the RB approximation is discussed together with the associated computational challenges [16]. We then dive into the study of the numerical approach of the regularized nonlinear PBE via the Taylor series expansion of the exponential nonlinear term in order to obtain a linear RPBE system at each iteration step [97, 98]. The RBM technique is applied to the linearized RPBE and the computational complexity of the resultant regularized ROM is investigated and resolved via the DEIM. We finalize this section by applying the RBM to the classical nonlinear PBE in order to demonstrate the computational differences between the two models, and the efficacy of the RPBE.

Chapter 6 provides the numerical tests and examples for both the classical and the regularized PBE models. We start off by providing the numerical results for the classical linearized PBE. Then, we demonstrate the accuracy of the nonlinear RPBE (NRPBE) model discretized on fine 3D grids over the classical PBE by its ability to capture the solution singularities accurately. The RBM is then applied to both the NRPBE and the classical NPBE, and numerical differences are highlighted. The computational runtimes for the two models are analyzed and conclusions made.

1.3. Thesis Contributions and Limitations

The principal goal of this thesis is the development of a new approach to the fast and accurate numerical solution of the PBE in order to rapidly accelerate its simulations for varying parameter values and/or other system configurations. First, this thesis contributes to the further improvement of the current PBE mathematical analysis as far as solution decomposition or regularization is concerned. This is realized by extending the results of the RS tensor format for multiparticle modeling [17] and for the splitting of the discretized Dirac delta distributions [85] to the PBE in order to construct the state of the art RS tensor-based regularized PBE model. This results in the efficient splitting of the electrostatic potential into two components, the short-range and long-range, within the molecular and the solvent domains, respectively.

The novelty of this technique is twofold: first, the tensor splitting is accomplished in the respective atomic volumes of the biomolecule, rather than through some cutoff function of the Laplacian operator at the solute-solvent interface as currently done in the existing solution decomposition techniques. Hence, the singular potential component does not contribute to the jump conditions. Secondly, by using the regularized source terms in the RPBE, the numerical approximation to the singular sources is avoided, thereby increasing the accuracy of the PBE solution, which is a major contribution towards its efficient regularization in the PBE theory, and also the main contribution of this thesis.

Secondly, in this thesis, we prove that the electrostatic solvation free interaction energy and forces do not depend on the short-range component of the electrostatic potential. This is the consequence of the corresponding properties for the free-space electrostatic interaction energy derived and verified in [15, 17]. The reason is that the short-range potential components, characterized by cusps, do not communicate (or interact) with their neighbours due to their localization (effective local supports) in the atomic volumes/regions. Instead, the energy is entirely driven by the long-range electrostatic potential. The regularized PBE model is, therefore, fundamental in the efficient computation of electrostatic energies and forces.

Thirdly, we have applied a more efficient numerical method of solving the high-fidelity full order model of the nonlinear regularized PBE (NRPBE), which is based on first linearization via the Taylor series truncation of the nonlinear term, followed by discretization. This approach avoids the computation of the Jacobian of a huge matrix and also converges much faster than the standard Newton iteration.

Lastly, in this thesis, we for the very first time apply the RBM techniques, which is a parametrized model order reduction tool, and DEIM to both the classical and the regularized PBE models in biomolecular simulation in order to provide greatly reduced computational costs in terms of storage and time due to the construction of a small reduced order model (ROM) from the high-fidelity full order model (FOM). However, the differences as far as the accuracy and efficiency are concerned, are quite conspicuous, whereby the regularized PBE model facilitates the construction of a ROM of lower dimension but of higher accuracy than that of the classical PBE because of the absence of the singularities which impede the reduction process.

The benefits of the RBM, or MOR in general, become obvious when the same problem has to be solved for a large number of parameter values. In this thesis, the break-even point is about 10, and thus, the RBM becomes very effective if dozens or more parameter configurations need to be evaluated.

However, this thesis is limited in the following ways: first, we do not provide detailed numerical experiments to corroborate with the actual energies of well-known biomolecules. This requires thorough investigations and provision of supporting literature, which is not in the scope of this thesis; secondly, the error bounds for the RBM are still a critical issue for the PBE. This is because the coefficient matrix A of the PBE system $Au = b$ has small eigenvalues of the order of $\mathcal{O}(10^{-2})$, which impede the construction of tight error bounds. In this thesis, we use the norm of the residual as the corresponding *a posteriori* error estimator.

CHAPTER 2

THE POISSON-BOLTZMANN EQUATION THEORY

Contents

2.1. Introduction

In this chapter, we begin in Section 2.2 by providing a general overview of electrostatic interactions in biomolecules and their significance in biological processes and why the Poisson-Boltzmann equation (PBE) is best suited for the numerical modeling and approximation. We then dive into the mathematical formulation of the PBE in Section 2.2.2 followed by the various applications of the electrostatic potential in Section 2.4. In Section 2.5, we provide a brief description of the classical PBE discretization using finite difference methods. Last but not least, we provide an overview of the recent advances in the PBE theory in Section 2.6, whereby the PBE is treated as an interface problem and highlight some of the solution decomposition techniques available in the literature. Consequently, we point out the drawbacks inherent in these techniques and propose the new regularization scheme based on the range-separated tensor format, which is the main focus in this thesis.

2.2. Electrostatic interactions in biomolecular systems

Biomolecular simulation is ubiquitous in biology and has gained popularity as a biophysical method for understanding various fundamental biological functions, dynamics and structures of biomolecules, enzyme catalysis, molecular recognition, and biomolecular encounter or association rates [48, 112]. In principle, the energetic properties of a biomolecule are ascertained by combining both the short- and long-range forces. On the one hand, short-range forces incorporate various components, for example, the van der Waals forces, angular forces, bonding forces, and torsional forces. On the other hand, long-range forces are usually influenced by electrostatic interactions, which are ubiquitous for any system of polar or charged molecules, for instance, biomolecules (nucleic acids, proteins, sugars, lipid bilayers) in their aqueous environment [5, 31], see Figure 2.1.

Due to their slow polynomial decay over distance (i.e., in the form of $1/\|\bar{x}\|$), electrostatic interactions are in principle, long-ranged and, therefore, they can neither be disregarded nor truncated in biomolecular modeling because they contribute remarkably to molecular interactions at all length scales [5, 37, 38, 70, 106, 120, 135]. Analysis of molecular solvation and electrostatics is vital to research in chemistry, biophysics and medicine, and can be categorized into quantitative analysis for thermodynamic or kinetic observables and qualitative analysis for general characteristics of biomolecular solvation[31, 50]. A significant challenge in biomolecular simulation has been to model these interactions accurately and efficiently. As such, methods that overcome this difficulty are of paramount importance and, therefore, desirous.

In a biomolecular simulation, the typical behaviour of electrostatic interactions is mostly determined by the following factors: molecular charge distributions, atomic radii of the solute (or biomolecule), mobile ion species, and the solvent [5]. There are two main groups of computational approaches which are used to model electrostatic interactions based on how the ionic solvent is treated during a simulation: explicit and implicit methods. Explicit methods, as their name suggests, treat the solvent and mobile ions as explicit particles around the biomolecule. Generally, they offer

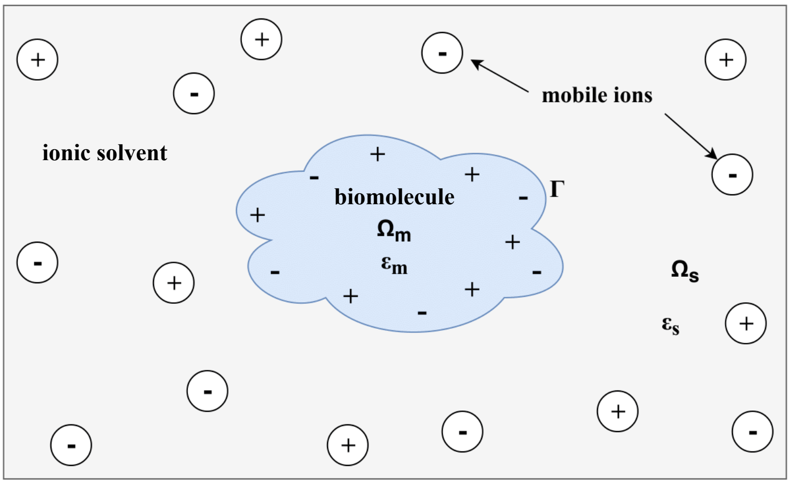


Figure 2.1.: Two-dimensional view of the three-dimensional Debye-Hückel model.

substantial detail and potential for accuracy in molecular simulation. However, the explicit solvent and mobile ions often account for more than 90% of the total atoms in a simulation, which can tremendously increase the computational costs for determining the kinetic and thermodynamic properties [5]. Therefore, these approaches are highly encouraged to be used by researchers in circumstances where the detailed interactions between the solvent and the solute are significant, for instance, in the saturation of solvent polarization near membranes [5].

On the other hand, implicit methods sacrifice the molecular details of the solvent by considering the ionic solvent particles as a continuum. This, in return, leads to the sampling of fewer degrees of freedom in the simulation, which results in a substantial decrease in the computational costs [5, 48, 143]. Therefore, implicit solvent methods enhance much better sampling of larger biomolecular systems than the explicit solvent techniques because of their reduced computational requirements. However, they have a lower accuracy level due to the simplistic treatment of water and ions that may underperform in some circumstances, for example, in high local ion densities and high ion valencies [5, 31].

2.2.1. Implicit solvent methods

As aforementioned, implicit solvent techniques are approximations aimed at reducing the degrees of freedom in a simulation by simplifying the description of the aqueous environment around and in the vicinity of the biomolecule. In this section, we briefly review a few descriptions of polar interactions in an implicit solvent setting. We refer interested readers to [5] for the case of nonpolar interactions.

The first example of implicit solvent models is the Debye-Hückel law [21], which provides the basic description of the electrostatic potential $\phi(x)$ due to a point charge of magnitude q located at position x_0 in a homogeneous polarizable medium of dielectric (permittivity) constant ϵ, i.e.,

$$\phi(x) = \frac{q e^{-\kappa \|x - x_0\|}}{\epsilon \|x - x_0\|}. \tag{2.1}$$

The ionic strength I of the solution, a function of the concentration of mobile ion species, is represented by the screening parameter

$$\kappa^2 = \frac{8\pi I}{1000\epsilon RT}, \tag{2.2}$$

where R is the gas constant and T is the absolute temperature (in Kelvin).

At zero ionic strength, that is, as $\bar{\kappa} \to 0$, the Debye-Hückel law reduces to the Coulomb's law, which provides a description of charges at infinite dilution in a polarizable continuum. A significant observation for the Debye-Hückel law and the Coulomb's law is that they both obey the superposition property, which states that the potential determined by a sum of charges is equivalent to the sum of the potentials of the isolated charges [5].

However, most biological systems of interest cannot be represented as a homogeneous dielectric medium. This is because the interiors of biomolecules often have substantially lower polarizabilities than their aqueous neighbourhoods. Therefore, Debye-Hückel or Coulomb equations cannot be used to model typical biomolecular electrostatics [5].

The second example is the generalized Born (GB) model [40, 117], which is based on the Born ion, a canonical electrostatic model that describes the electrostatic potential and solvation energy of a spherical ion. In this model, an analytical expression based on the Born ion model is used to approximate the electrostatic potential and solvation energy of small molecules. In this model, the electrostatic contribution to the free energy of solvation is

$$\Delta G_{\text{pol}} = -\frac{1}{2}\left(1 - \frac{1}{\epsilon_w}\right)\sum_{i,j}\frac{q_i q_j}{f_{\text{GB}}}, \tag{2.3}$$

where ϵ_s is the solvent dielectric constant, q_i and q_j are partial charges, and f_{GB} is a function which interpolates between an effective Born radius, in cases where the distance r_{ij} between atoms is short, and r_{ij} itself at large distances [40].

It is a computationally fast method for calculating approximate forces and energies for solvated molecules, hence very popular in high-throughput applications such as the molecular dynamics simulations. However, it fails to capture all the details of the molecular structure and ion distributions thereby rendering it heuristic [5, 31].

The final and the most significant of the implicit solvent models, which is also an extension of the Debye-Hückel model, is the Poisson-Boltzmann equation (PBE). It is also the main model under study in this thesis. The PBE offers a compromise between faster, but more approximative models, for instance, the GB, and more comprehensive explicit solvent and integral equation approaches [5]. We derive the PBE model in Section 2.2.2.

In a nutshell, continuum approaches which are motivated by the Debye-Hückel theory, that is, treatment of electrostatic effects of solvated biomolecules, may be specially suited for Brownian dynamics simulations (BD) or molecular dynamics (MD), in case the electrostatic force is predominant in determining the behaviour of the system [67]. We briefly describe the notions of MD and BD.

Definition 2.1 (Molecular dynamics [67]):

It is well known that the motions of macrobiomolecules, for example, proteins, are a particular instance of the N-body problem, which obey the laws of classical mechanics. The dynamics of such a biomolecule can be described by the Newton's second law, i.e.,

$$\mathbf{f} = M\mathbf{a},$$

where M is an $N \times N$ diagonal matrix containing the masses m_i arranged along the diagonal. Note that the vector $\mathbf{f}$ describing the electrostatic forces or chemical bonds can be represented by the potential function $\phi(\mathbf{r})$ as

$$\mathbf{f} = -\nabla\phi(\mathbf{r}).$$

The potential function ϕ is considered to be the sum of various distinct potential functions, i.e.,

$$\phi = \sum_{i=1}^{k} \phi_i, \tag{2.4}$$

where, for instance, the rapidly decaying near field potential ϕ_1 may include the Lennard-Jones 6-12 potential of noble gases

$$\phi(\mathbf{r}) = \sum_{i=1}^{N} 4\epsilon \left[\left(\frac{\sigma_i(\mathbf{r})}{|\mathbf{r} - \mathbf{r}_i|} \right)^{12} - \left(\frac{\sigma_i(\mathbf{r})}{|\mathbf{r} - \mathbf{r}_i|} \right)^{6} \right], \tag{2.5}$$

which decays as $|\mathbf{r}|^{-6}$, or the van der Waals potential of chemical physics.

The external field ϕ_2 might include, for instance, externally magnetic fields while the far field ϕ_3 might include the electrostatic potential of a charged particle system or the gravitational potential. The far field is characterized by slow polynomial decay in $|\mathbf{r}|^{-1}$, for example, the electrostatic potential of a charged multiparticle system

$$\phi(\mathbf{r}) = \sum_{i=1}^{N} \frac{q_i}{\epsilon|\mathbf{r} - \mathbf{r}_i|}. \tag{2.6}$$

Biomolecules mostly occur in ionic solutions, and as such, the equations of motion must include large scale electrostatic effects of the solvent molecules for the accurate approximation of the far field. This is computationally expensive and infeasible for large biomolecules, hence a tractable alternative for the electrostatic force calculations can only be offered by continuum representations of the solvent molecules [67]. ◊

Definition 2.2 (Brownian dynamics simulations [67]):

It refers to the simulation of an interacting system of particles by combining both the deterministic effects of the Newton's second law of motion (i.e., dynamics) and the stochastic effects (i.e., Brownian motion). This approach can be fundamental in investigating the binding properties of antibody or proteins once the electrostatic potential has been computed. ◊

2.2.2. The Poisson-Boltzmann equation

The PBE is one of the most popular implicit solvent models which describes the solvent in a continuum model through the Boltzmann distribution. It describes the electrostatic potential in the entire domain which comprises both the molecule and the solvent. From this potential, further information can be obtained at various regions of interest and for different applications. Firstly, the electrostatic potential at the biomolecular surface, commonly known as electrostatic surface potential, can provide insights into possible docking sites for other small or large molecules. Secondly, the potential outside the biomolecule can provide information about the free energy of interaction of small molecules at different positions in the vicinity of the biomolecule. Thirdly, free energy of a biomolecule can be determined, which provides information about the molecule's stability. Finally, the electrostatic field can be estimated from which the mean atomic forces can be derived. More information can be found in [47, 48, 68, 132].

There are numerous ways for and reviews on the derivation of the PBE. The simplest stems from the Poisson equation [75, 112], which can be derived from the Gauss' law and the divergence theorem (or Gauss' theorem) [36]. Invoking the integral form of the Gauss' law, which relates the electric displacement field over a closed Gaussian surface to the enclosed charge, we obtain the following general conservation relation

$$\int_{\partial\Omega} \epsilon(s)\mathbf{E}(s) \cdot ds = \int_{\Omega} \frac{\rho(\bar{x})}{\epsilon_0} d\bar{x}, \tag{2.7}$$

where $\epsilon(s)\mathbf{E}(s)$ is the electric displacement field, $\mathbf{E}$ is the electric field, $\epsilon(s)$ is the permittivity of the enclosed surface, ϵ_0 is the electric constant (or vacuum permittivity), and ρ is the total electric charge density.

The divergence theorem, also in integral form, relates the electric field flux across the boundary of a closed surface with the enclosed divergence, i.e.,

$$\int_{\partial\Omega} \mathbf{v}(s) \cdot ds = \int_{\Omega} \nabla \cdot \mathbf{v}(\bar{x}) d\bar{x}, \tag{2.8}$$

is applied to the left-hand side of (2.7) to obtain

$$\int_{\partial\Omega} \epsilon(s)\mathbf{E}(s) \cdot ds = \int_{\Omega} \nabla \cdot \epsilon(\bar{x})\mathbf{E}(\bar{x}) d\bar{x}. \tag{2.9}$$

Substituting the results of (2.9) into (2.7), that is, the divergence of the electric displacement field equals the charge density, we obtain, in differential form, the Poisson equation for electrostatics (in SI units)

$$-\nabla \cdot (\epsilon(\bar{x})\nabla u(\bar{x})) = \rho(\bar{x}), \quad \text{in} \quad \Omega \in \mathbb{R}^3, \tag{2.10}$$

which describes the electrostatic potential $u(\bar{x})$ at a point $\bar{x} = (x, y, z) \in \Omega$. Note that here, $\mathbf{E}(\bar{x}) = -\nabla u(\bar{x})$.

The term $\rho(\bar{x})$ is the charge distribution which generates the potential in a region with a position-dependent and piecewise constant dielectric function $\epsilon(\bar{x})$. Equation (2.10) is generally solved in a finite domain Ω subject to Dirichlet boundary conditions $u(\bar{x}) = g(\bar{x})$ on $\partial\Omega$. Usually, $g(\bar{x})$ employs an analytic and asymptotically correct form of the electrostatic potential and therefore, the domain must be sufficiently large to ensure an accurate approximation of the boundary conditions [41].

To obtain the PBE from equation (2.10), we consider two contributions to the charge distribution $\rho(\bar{x})$: the "fixed" solute charges $\rho_f(\bar{x})$ and the aqueous "mobile" ions in the solvent $\rho_m(\bar{x})$. The N_m partial atomic point charges (z_i) of the biomolecule are modeled as a sum of delta distributions at each atomic center $\bar{x}_i$, for $i = 1, \ldots, N_m$, that is,

$$\rho_f(\bar{x}) = \frac{4\pi e_c^2}{\kappa_B T} \sum_{i=1}^{N_m} z_i \delta(\bar{x} - \bar{x}_i). \tag{2.11}$$

Here, $e_c/\kappa_B T$ (or reciprocal of the thermal voltage) is the scaling coefficient which ensures that the electrostatic potential is dimensionless, where e_c is the electron charge and $\kappa_B T$ is the thermal energy of the system and is comprised of the Boltzmann constant κ_B and the absolute temperature T. The total charge of each atom is $e_c z_i$.

On the other hand, the solvent is modeled as a continuum through the Boltzmann distribution which leads to the mobile ion charge distribution

$$\rho_m(\bar{x}) = \frac{4\pi e_c^2}{\kappa_B T} \sum_{j=1}^{m} c_j q_j e^{-q_j u(\bar{x}) - V_j(\bar{x})}, \tag{2.12}$$

where we have m mobile ion species with charges q_j and bulk concentrations c_j. The term $V_j(\bar{x})$ is the steric potential which prevents an overlap between the biomolecule and the counterions.

Substituting (2.11) and (2.12) for ρ in the Poisson equation (2.10) we obtain the following classical nonlinear PBE (NPBE)

$$-\nabla \cdot (\epsilon(\bar{x})\nabla u(\bar{x})) - C \sum_{j=1}^{m} c_j q_j e^{-q_j u(\bar{x}) - V_j(\bar{x})} = C \sum_{i=1}^{N_m} z_i \delta(\bar{x} - \bar{x}_i), \quad \text{in} \quad \Omega \in \mathbb{R}^3, \tag{2.13}$$

where $C = \frac{4\pi e_c^2}{\kappa_B T}$. This PBE model is best suited for polyvalent (or asymmetric) electrolytes.

However, for monovalent (or symmetric) electrolytes, whose ions are in a $1:1$ ratio, for example, $NaCl$, we assume similar steric interactions for each species with the solute. Moreover, $m = 2$, $c_j = c$, $q_j = (-1)^j$ and therefore, the two-term exponential

Table 2.1.: Parameters of the PBE (2.15) in centimeter-gram-second (CGS) units.

Parameter	Value	Unit (abbr.)	Name
e_c	4.8032424e-10	electrostatic units (esu)	Fundamental charge
N_A	6.0221367e+23	-	Avogadro's number
T	298.15	Kelvin (K)	Absolute temperature
κ_B	1.3806581e-16	energy/Kelvin (erg/K)	Boltzmann's constant

series function (2.12) simplifies as follows:

$$\left.\begin{aligned} \rho_m(\bar{x}) &= \frac{4\pi e_c^2}{\kappa_B T}\sum_{j=1}^{2} c_j q_j e^{-q_j u(\bar{x})-V_j(\bar{x})}, \\ &= \frac{4\pi e_c^2}{\kappa_B T}\left[cqe^{-qu(\bar{x})-V(\bar{x})} - cqe^{qu(\bar{x})-V(\bar{x})}\right], \\ &= \frac{4\pi e_c^2}{\kappa_B T} cqe^{-V(\bar{x})}\left[e^{-qu(\bar{x})} - e^{qu(\bar{x})}\right], \\ &= -\frac{8\pi e_c^2}{\kappa_B T} cqe^{-V(\bar{x})}\sinh(qu(\bar{x})), \\ &= -\bar{\kappa}^2(\bar{x})\sinh(u(\bar{x})), \end{aligned}\right\} \tag{2.14}$$

where $\bar{\kappa}^2 = 8\pi N_A e_c^2 I/1000\epsilon\kappa_B T$ is a piecewise constant function as defined in (2.18), which describes both the ion accessibility through $e_c^{-V(\bar{x})}$ and the bulk ionic strength (or concentration) $I = 1/2\sum_{j=1}^{N_{ions}} c_j q_j^2$ [5].

We eventually obtain the PBE by combining the two expressions for the charge distributions in (2.11) and (2.14) with the Poisson equation (2.10) for a monovalent electrolyte,

$$-\nabla\cdot(\epsilon(\bar{x})\nabla u(\bar{x})) + \bar{\kappa}^2(\bar{x})\sinh(u(\bar{x})) = \sum_{i=1}^{N_m} q_i\delta(\bar{x}-\bar{x}_i), \tag{2.15}$$

subject to

$$u(\bar{x}) = g(\bar{x}) \quad \text{on} \quad \partial\Omega, \tag{2.16}$$

where

$$u(\infty) = 0. \tag{2.17}$$

In equation (2.15), $q_i = \frac{4\pi e_c^2}{\kappa_B T} z_i$, $u(\bar{x}) = e_c\psi(\bar{x})/\kappa_B T$ is the dimensionless potential scaled by $e_c/\kappa_B T$ and $\psi(\bar{x})$ is the original electrostatic potential in centimeter-gram-second (cgs) units at $\bar{x} \in \mathbb{R}^3$. The terms $\epsilon(\bar{x})$ and $\bar{\kappa}^2(\bar{x})$ are discontinuous functions at the interface between the charged biomolecule and the solvent, and at an ion exclusion region (Stern layer) surrounding the molecule, respectively, i.e.,

$$\epsilon(\bar{x}) = \begin{cases} \epsilon_m = 1 & \text{if } \bar{x} \in \Omega_m \\ \epsilon_s \ = 78.54 & \text{if } \bar{x} \in \Omega_s/\Omega_{ex} \end{cases}, \quad \bar{\kappa}(\bar{x}) = \begin{cases} 0 & \text{if } \bar{x} \in \Omega_m/\Omega_{ex} \\ \sqrt{\epsilon_s}\kappa & \text{if } \bar{x} \in \Omega_s \end{cases}, \tag{2.18}$$

where Ω_m is the region occupied by the protein molecule, Ω_s is the region occupied by the ionic solution and Ω_{ex} is the ion exclusion layer.

Remark 2.3:

Here and in the rest of the thesis, we shall denote by $\bar{\kappa}(\bar{x})$ the position-dependent piecewise constant κ function in order to distinguish it from the scalar variable κ. Note also that in the numerical tests, we consider the commonly used empirical value of the solvent dielectric constant to be $\epsilon_m = 2$. ◇

The function $g(\bar{x})$ represents the Dirichlet boundary conditions which are discussed in detail in Section 2.5.4 and are nonaffine in the parameter I. Some of the selected parameters of the PBE are given in Table 2.1.

Details on mapping $\epsilon(\bar{x})$ and $\bar{\kappa}^2(\bar{x})$ onto a computational grid can be found in [7]. The PBE (2.15) poses severe computational challenges in both analytical and numerical approaches due to the infinite (unbounded) domain in (2.17), delta distributions, rapid nonlinearity, and discontinuous coefficients [66, 67].

The PBE (2.15) can be linearized in the following two ways. One way is to consider the solvent charge density ρ_m in (2.12) and assume that the steric interactions for each species with solute are similar and that the local electrostatic energies are very small. Then we apply the Taylor series expansion to the exponential function and assume electroneutrality

$$\sum_j c_j q_j = 0$$

of the bulk solution to obtain the linear approximation of ρ_m as follows

$$\left.\begin{aligned} \rho_m(\bar{x}) &= \frac{4\pi e_c^2}{\kappa_B T} \sum_{j=1}^{2} c_j q_j e^{-q_j u(\bar{x}) - V_j(\bar{x})}, \\ &\approx \frac{4\pi e_c^2}{\kappa_B T} e^{-qu(\bar{x})} \sum_j c_j q_j \left[1 - q_j u(\bar{x})\right], \\ &= -\left[\frac{4\pi e_c^2}{\kappa_B T} e^{-qu(\bar{x})} \sum_j c_j q_j^2\right] u(\bar{x}), \\ &= -\bar{\kappa}^2(\bar{x}) u(\bar{x}), \end{aligned}\right\} \tag{2.19}$$

In the second method, we make the assumption that the electrostatic potential is very small relative to the thermal energy $\kappa_B T$ [48]. Therefore, the nonlinear function $\sinh(u(\bar{x}))$ can be expanded into a Taylor series

$$\sinh(u(\bar{x})) = u(\bar{x}) + \frac{u(\bar{x})^3}{3!} + \frac{u(\bar{x})^5}{5!} + \dots, \tag{2.20}$$

and only the first term is retained. We obtain the linearized PBE (LPBE) given by

$$-\nabla \cdot (\epsilon(\bar{x}) \nabla u(\bar{x})) + \bar{\kappa}^2(\bar{x}) u(\bar{x}) = q_i \delta(\bar{x} - \bar{x}_i). \tag{2.21}$$

Remark 2.4:

Both the LPBE and NPBE are referred to as interface problems, because of the discontinuities in $\epsilon(\bar{x})$ and $\bar{\kappa}(\bar{x})$, which represent material interfaces in the physical problem. ◊

Usually, proteins are not highly charged, and it suffices to consider the linearized PBE (LPBE). One can still obtain accurate results because the higher order terms in (2.20) do not provide a significant contribution. However, we must note that the LPBE can give inaccurate results for highly charged biomolecules such as the DNA and RNA (nucleic acids), phospholipid membranes, and polylysine [68]. More information about the PBE, including its derivation from first principles, can be found in [67].

In this thesis, we consider a protein molecule immersed in ionic solution at physiological concentration, and determine the electrostatic potential triggered by the interaction between the two particles, see Figure 2.1. The electrolyte here is of monovalent type, implying that the ionic strength is equivalent to the concentration of the ions. The ionic strength is a physical parameter of the PBE, and we aim to accurately and efficiently determine the electrostatic potential under variation of this parameter.

2.2.3. Analytical solutions of the PBE

Analytical solutions of the PBE are only possible under the assumption that the biomolecules of interest have regular shapes, for example, spheres or cylinders, and are only valid for the linearized PBE [66, 67]. And even if these solutions exist, they are still quite complex. Here, we only discuss the analytical solution due to a spherical molecule with uniform charge immersed in ionic solvent as depicted in Figure 2.2. Another model, which we shall not discuss here, is based on the complete solvent penetration whereby rod-shaped molecules are considered [67]. The idea is to construct the analytical solutions separately in the different model regions and then concatenate them by invoking continuity conditions at the interfaces.

Let us consider Figure 2.2 in which the solvated spherical molecule has total surface charge q. The coordinate system is defined to be centered at the molecule so that we denote by R the radius of the molecule. Since the ions do not come into contact with the molecule, they are barred by the stern (or ion-exclusion) layer, $a - R$. Therefore, the ionic solvent covers the region outside of a. Further, we denote by ϵ_m and ϵ_s the molecular and solvent dielectric constants, respectively. We obtain the following expressions for the analytical solution in the three regions [67]:

$$\left.\begin{aligned} u(r) &= \frac{q}{\epsilon_m R}\left(1 - \frac{R\bar{\kappa}}{1+\bar{\kappa}a}\right) & r &\leq R, \\ u(r) &= \frac{q}{\epsilon_w r}\left(1 - \frac{r\bar{\kappa}}{1+\bar{\kappa}a}\right) & R \leq r &\leq a, \\ u(r) &= \frac{q e^{-\bar{\kappa}(r-a)}}{\epsilon_w(1+\bar{\kappa}a)r} & r &\geq R \end{aligned}\right\} \tag{2.22}$$

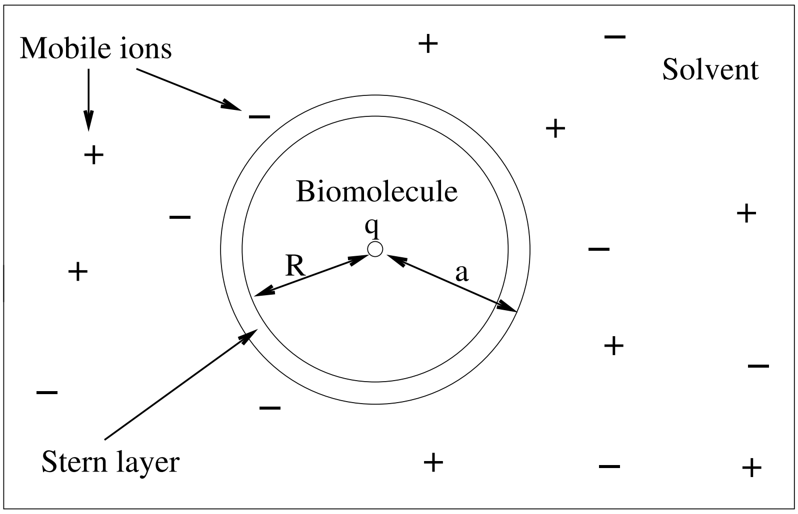


Figure 2.2.: Two-dimensional view of a spherical molecule with spherically symmetric charge.

However, these analytical models are not realistic because biomolecules have irregular shapes or geometries and charge distributions [41, 67]. Therefore, numerical solutions must be computed in order to obtain accurate and realistic solutions to the PBE models, and we discuss these in Section 2.2.4.

2.2.4. Numerical solutions of the PBE

The shortcomings of the aforementioned analytical solutions makes it necessary to apply numerical techniques to the PBE and the first of such methods were introduced in [145] where the electrostatic potential was determined at the active site of a protein (or enzyme). The most popular numerical techniques in this regard are based on discretization of the domain of interest into small regions and employ the finite difference methods (FDM) [7, 143], the finite element methods (FEM) [7, 66], the boundary element methods (BEM) [22, 154], or the domain decomposition method [122]. Here, we briefly discuss these numerical techniques and highlight their strengths and weaknesses as far as the PBE discretization is concerned. A detailed review of these numerical methods for solving the PBE can be found in [108].

2.2.4.1. Finite difference methods (FDM)

Finite difference methods (FDM) offer simpler implementation than the FEM or the finite volume methods because the mesh generation and refinement is trivial [108], see Figure 2.3. This advantage makes the FDM techniques the most popular numerical methods for solving the PBE in biomolecular simulations. In principle, the FDM solves the PBE on a uniform Cartesian mesh, where a Taylor series expansion is employed to transform the differential operator in the PBE, i.e., $-\nabla \cdot (\epsilon \nabla)$, into a sparse difference matrix. A manifold of matrix algebra techniques are then used to solve the resulting matrix equations [5].

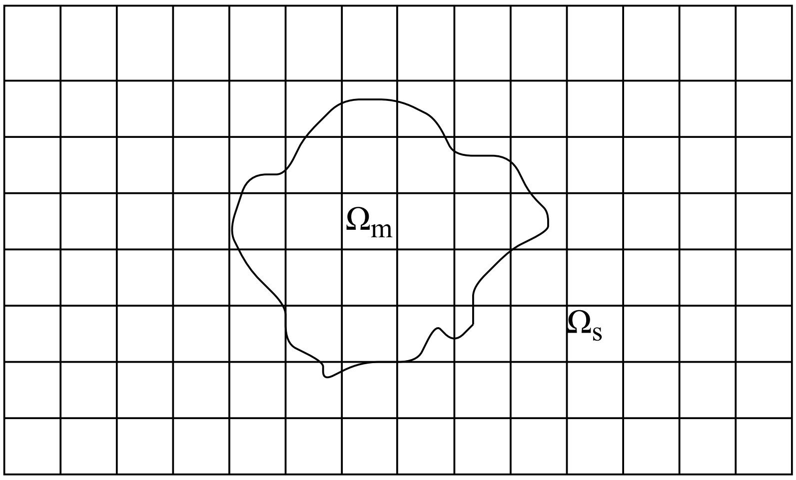


Figure 2.3.: Two-dimensional view of the finite difference Cartesian grids for the PBE.

However, the FDM techniques also experience some numerical drawbacks when applied to the PBE. First, in biomolecular electrostatic calculations, typical grid spacings (Δh) range between 0.2Å to 1Å. Therefore, for large biomolecular systems of hundreds of angstroms in size, the number of degrees of freedom $\mathcal{N} = n^3$ in the resultant discretized system easily approaches $\mathcal{O}(10^6) - \mathcal{O}(10^9)$, which can be an onerous task for the current computing resources. Secondly, despite their simple problem setup, the FD grids provide minimal control over the placement of the unknowns in the solution domain. Thirdly, in general, linear system solvers are of low efficiency thereby leading to increased computational runtimes for the PBE [5, 108].

2.2.4.2. Boundary element methods (BEM)

In principle, the Boundary element methods (BEM) techniques recast the linear Poisson-Boltzmann PDEs as certain boundary integral equation in the solution u by invoking Green's theorem [108]. This is realized by triangular simplicial discretization of the molecular surface, and the solution represented for an induced surface charge. The solution is then convolved with the Green's function (or Coulomb potential) to give the desired solution. The BEM provides a highly efficient computational tool for the LPBE due to the calculation of a detailed description of the macromolecular geometry with a low number of degrees of freedom [5].

However, the BEM have the following severe drawbacks: (1) they are only applicable to the LPBE because of inexistence of Green's functions for the nonlinear PBE, thereby limiting their general use in biomolecular electrostatics and (2) their accuracy and or stability is affected by singular boundary integrals [5, 108].

2.2.4.3. Finite element methods (FEM)

Finite element methods (FEM), like the BEM, employ simplicial discretization for the mesh generation and construct the electrostatic potential from piecewise polynomial basis functions which are associated with mesh vertices [5]. FEM in general, provides more flexibility for local mesh refinement and handling of nonlinear equations, more rigorous convergence analysis and more selections of efficient iterative system solvers than the FDM and the BEM [108]. However, when applied to the PBE with strong singularities, it was not until in [103] that its rigorous solution and approximation theory were established.

All of the aforementioned numerical methods have one major advantage in common. It is possible to employ "electrostatic focussing", which enables users to apply relatively coarse grids for the entire calculations and very fine grids in regions of great interest such as the binding or active sites of macrobiomolecules. This adaptivity provides highly accurate local solutions to the PBE at reduced computational costs [5].

2.2.4.4. Multilevel solvers

Efficient solution of the algebraic system of equations derived from the discretization of the PBE with either the FDM or the FEM can be obtained by multilevel solvers [26]. Matrix equations are often solved by iterative approaches, in which an initial guess u_0 is provided as the starting point, then a set of operations are repeatedly applied to improve this initial guess until desired convergence is reached. However, the inability to rapidly reduce long-range error in the solution limits the speed of conventional iterative methods. This problem can be circumvented by the projection of the discretized system onto grids or meshes at multiple resolutions. The error in the gradually converging low-frequency solution components is rapidly reduced by the projection onto the finest mesh through the solutions on the coarser levels of the system [5].

A "multilevel" solver algorithm is derived from the coupling of scales, in which the system of algebraic equations is directly solved on the coarsest level, and the results used to accelerate solutions on finer mesh levels [5]. The multilevel hierarchy is assembled depending on the underlying discretization method. For example, for FDM techniques, the so-called "multigrid" methods are used with minimal additional work required to assemble the hierarchy. On the other hand, the "algebraic multigrid" methods are employed for the adaptive finite element discretizations [5]. Multigrid or multilevel techniques for both the FDM and FEM discretizations of the PBE have been implemented, for example, in the APBS software package [78].

2.2.4.5. Parallel methods

Some PBE systems can be extremely large to be solved sequentially, i.e., on one processor, regardless of the scalability of the numerical algorithm [5]. For instance, studies on macromolecular assemblages or complexes may include between $\mathcal{O}(10)$--$\mathcal{O}(10^5)$ of residues such as ribosomes, microtubules, among others, which may be prohibitively expensive on most sequential platforms. Therefore, multiprocessor computing platforms are necessary in such cases in order to solve the PBE in a parallel manner [5]. Parallel methods together with the multigrid approaches have also been implemented in the APBS software package, for example, the finite difference parallel multigrid (PMG) solver [78].

Numerous software packages have been developed to solve the PBE and some of the major ones include the adaptive Poisson-Boltzmann solver (APBS) [7] and Delphi [127]. There are also recent developments regarding the PBE theory which include, the treatment of the biomolecular system as an interface problem, the extensive studies on the nonlinear PBE, among others, see Section 2.6 for more details.

Due to the limited computational memory and speed, solving the PBE efficiently is still computationally challenging and affecting the accuracy of the numerical solutions. This is due to the following reasons. Firstly, electrostatic interactions are long-ranged and therefore, the electrostatic potential decays exponentially over large distances, see equation (2.17). This requires an infinite domain which is infeasible in practice. Secondly, biomolecules of interest comprise thousands to millions of atoms which require a large domain to accommodate both the biomolecule and the solvent.

To circumvent these challenges, it is customary to choose a truncated domain of at least three times the size of the biomolecule so as to accurately approximate boundary conditions [67]. Nonetheless, this still leads to a very large algebraic system consisting of several hundreds of thousands to millions of degrees of freedom. It becomes even more difficult if the PBE is incorporated in a typical dynamics simulation which involves millions of time steps or in a multi-query task where the solution is solved many times for varying parameter values such as the ionic strength [143].

Remark 2.5:

The computational complexity arising from the resultant high-dimensional system can be greatly reduced by applying model order reduction (MOR) techniques. The main goal of MOR is to construct a reduced-order model (ROM) of typically low dimension, whose solution retains all the important information of the high-fidelity system at a greatly reduced computational effort. Because the PBE is a parametrized PDE (PPDE), we apply the reduced basis method (RBM) in Chapter 5, which falls into the class of parametrized MOR (PMOR) techniques [14]. However, it is important to note that the RBM is not an independent numerical technique; hence its accuracy depends on that of the underlying technique which is used to discretize the PBE [14, 42]. $\Diamond$

2.3. Theory for the PBE solution

In this section, we provide the results for the existence and uniqueness of solutions to both the linearized and the nonlinear PBE (NPBE) models, independently. For the linearized PBE case, we show that a unique weak solution exists in either $H^1(\Omega)$ or $L^2(\Omega)$, depending on whether or not the source terms are approximated by $L^2(\Omega)$ functions, respectively [67]. On the other hand, for the NPBE, we prove that the unique solution to the boundary value problem of the NPBE extremizes the energy [30, 67, 105]. Note that some standard definitions and theorems from functional analysis, which are significant in the proofs of the results in Section 2.3.1 and Section 2.3.2 are available in Appendix A.

2.3.1. Existence and uniqueness theorems for the LPBE

Let us first consider the strong, divergence form of the general second order linear elliptic equations, in which the LPBE is a member [67]:

$$\left.\begin{aligned} \nabla\cdot(a\nabla u)+bu&=f \quad \text{in}\,\Omega\subset\mathbb{R}^d,\\ u&=g_D \quad \text{on}\,\Gamma_D,\\ (a\nabla u)\cdot n+cu&=g_N \quad \text{on}\,\Gamma_N, \qquad \Gamma=\Gamma_D\cup\Gamma_N \quad \text{and} \quad \Gamma_D\cup\Gamma_N=\emptyset,\end{aligned}\right\} \tag{2.23}$$

where the functions $b(\bar{x}), f(\bar{x}), u(\bar{x}) : \Omega \mapsto \mathbb{R}$, $g_D(\bar{x}) : \Gamma_D \mapsto \mathbb{R}$, $g_N(\bar{x}) : \Gamma_N \mapsto \mathbb{R}$, and the matrix function $a(\bar{x}) : \Omega \mapsto \mathbf{L}(\mathbb{R}^d, \mathbb{R}^d)$. Equation (2.23) is often written in the abstract form as

$$\mathcal{L}u = f.$$

The equation is elliptic if the matrix $a(\bar{x}) = [a_{ij}(\bar{x})]$ is positive definite for all $\bar{x} \in \Omega$ and strongly elliptic if the following condition holds:

$$\exists\,\lambda > 0 \quad \text{such that} \sum_{ij} a_{ij}\eta_i\eta_j \geq \lambda|\eta|^2, \quad \forall\bar{x}\ \in\Omega,\ \eta\ \in\ \mathbb{R}^d. \tag{2.24}$$

Consequently, (2.23) has the following weak form. Find $u \in H_0^1(\Omega)$ such that

$$A(u,v) = F(v) \qquad \forall v\ \in\ H_0^1(\Omega), \tag{2.25}$$

where

$$A(u,v) = \int_\Omega (a\nabla u\cdot\nabla v + buv)d\bar{x}, \qquad F(v) = \int_\Omega fv\,d\bar{x}, \qquad \Omega\ \subset\mathbb{R}^d, \tag{2.26}$$

with $g = \operatorname{tr} w$ and $\Omega \in \mathcal{C}^{0,1}$ (i.e., bounded domains with locally Lipschitz boundary). Consequently, the problem coefficients satisfy [67]:

(a) $0 < a_{ij}(\bar{x}) \leq c_1 < \infty$, $\forall\bar{x} \in \Omega$, $\quad i, j = 1, \ldots, d$.

(b) $0 \leq b(\bar{x}) \leq c_2 < \infty$, $\forall\bar{x} \in \Omega$.

(c) $f(\bar{x}) \in L^2(\Omega)$.

(d) $w(\bar{x}) \in H^1(\Omega)$, $g(\bar{x}) \in H^{1/2}(\Gamma)$, $g = \operatorname{tr} w$.

(e) The differential operator is strongly elliptic, i.e., (2.24).

The following preliminary result can be proven given the very weak assumptions on the coefficients in the problem (2.25)-(2.26). We shall use this result to prove the existence of a unique solution to the LPBE. See [67] for more details.

Lemma 2.6 (Elliptic LPDE: Existence and uniqueness, see Lemma 2.10 in [67]): There exists a unique weak solution $u \in H_0^1(\Omega)$ to the elliptic linear PDE (LPDE) problem (2.25)-(2.26). ♢

Proof. A detailed proof can be found in [67], where it is shown that the existence and uniqueness of a weak solution $u \in H_0^1(\Omega)$ to (2.25)-(2.26) follows from the application of the Lax-Milgram Theorem, whose conditions are met by the assumptions on the problem coefficients. The main idea is to show that $A(\cdot,\cdot)$ is bounded and coercive, and $F(\cdot)$ is bounded. □

With this background on elliptic PDEs, we are in a good position to tackle the solution theory for the LPBE. Consider the LPBE (2.21), where very broad ranges of temperatures and ionic strengths are allowed, that is,

$$T \in [200K, 400K], \qquad I_s \in [0, 10].$$

Consequently, we assume that the set of discrete charges $\{z_1, \dots, z_{N_m}\}$ which represent the biomolecule lies well within the domain Ω, and therefore, far from the boundary Γ of Ω. It is easy to show that the LPBE (2.21) has the form of (2.23), with the equivalent weak formulation (2.25)-(2.26). It is shown in [67] that for the given temperature and ionic strength ranges, the problem coefficients satisfy the following bounds:

$$\left.\begin{array}{ll}
(a) & a : \Omega \mapsto L(\mathbb{R}^3, \mathbb{R}^3), \quad a_{ij}(\bar{x}) = \delta_{ij}\epsilon(\bar{x}), \quad 2 \leq \epsilon(\bar{x}) \leq 80, \quad \forall \bar{x} \in \Omega. \\
(b) & b : \Omega \mapsto \mathbb{R}, \quad b(\bar{x}) = \bar{\kappa}^2(\bar{x}), \quad 0 \leq \bar{\kappa}^2(\bar{x}) \leq 127, \ \forall \bar{x} \in \Omega. \\
(c) & f : \Omega \mapsto \mathbb{R}, \ f(\bar{x}) = C \cdot \sum_{i=1}^{N_m} z_i \delta(\bar{x} - \bar{x}_i), \ 5249 \leq C \leq 10500, \\
 & \qquad\qquad -1 \leq z_i \leq 1, \ \forall \bar{x} \in \Omega. \\
(d) & g : \Gamma \mapsto \mathbb{R}, \quad g(\bar{x}) = \frac{C}{4\pi\epsilon_s} \cdot \sum_{i=1}^{N_m} \frac{z_i e^{-\bar{\kappa}(\bar{x})|\bar{x}-\bar{x}_i|/\sqrt{\epsilon_s}}}{|\bar{x}-\bar{x}_i|}, \quad \epsilon_s = 80, \quad \forall \bar{x} \in \Gamma.
\end{array}\right\} \tag{2.27}$$

Remark 2.7:

From the aforementioned assumption that the charges z_i do not lie in the vicinity of Γ, which will always hold when we choose our domain and the boundary, it is evident that the boundary function $g(\bar{x})$ is a well-behaved continuously differentiable function of $\bar{x}$, i.e., $g \in C^1(\Gamma)$. ◊

Remark 2.8:

It is important to note that if a quantum mechanical description of the solute is used, parts of the assumptions formulated in (2.27) are no longer true. For instance, there is no quantum mechanical operator which corresponds to an atom, because atoms are not well-defined in molecules. This renders the definition of atomic charge ambiguous [34, 144]. ◊

Theorem 2.9 (LPBE: Existence and uniqueness, see Theorem 2.11 in [67]):

There exists a unique weak solution $u \in H^1(\Omega)$ to the LPBE if the source terms are approximated by $L^2(\Omega)$ functions. ◊

Proof. The proof is straightforward as outlined in [67]. To apply Lemma 2.6, we have to verify the assumptions on the problem coefficients. To begin with, the assumptions on $a(\bar{x})$ and $b(\bar{x})$ are clearly satisfied, see (2.27). Consequently, since the source functions $\delta(\bar{x}-\bar{x}_i)$ are approximated with functions $f_i(\bar{x}-\bar{x}_i) \in L^2(\Omega)$, via, for example, the cubic spline interpolation, then the composite function $f(x)$ is also clearly in $L^2(\Omega)$. From Remark 2.7, we have that $g \in C^1(\Gamma) \subset H^1(\Omega)$ and it is well known that $\exists w \in H^1(\Omega)$ such that $g = \operatorname{tr} w$. Finally, from the lower bound on $a_{ij}(\bar{x})$, the strong ellipticity assumption is satisfied. Therefore, the theorem now follows from Lemma 2.6. □

Remark 2.10:

The Lax-Milgram Theorem cannot be used to show the existence and uniqueness of solutions to the LPBE if the function f consists of Dirac delta distributions representing point charges, therefore, $f \notin L^2(\Omega)$. This is because the resulting linear functional F in (2.26) is no longer bounded [67]. ◊

2.3.2. Existence and uniqueness theorems for the NPBE

Let us first consider the strong, divergence form of the general second order semi-linear elliptic equations, in which the NPBE is a member [67]:

$$\left.\begin{aligned} \nabla\cdot(a\nabla u) + b(\bar{x},u) &= f \quad \text{in } \Omega \subset \mathbb{R}^d,\\ u &= g_D \quad \text{on } \Gamma_D,\\ (a\nabla u)\cdot n + cu &= g_N \quad \text{on } \Gamma_N, \qquad \Gamma = \Gamma_D \cup \Gamma_N \quad \text{and} \quad \Gamma_D \cup \Gamma_N = \emptyset, \end{aligned}\right\} \tag{2.28}$$

where now the function $b(\bar{x},u) : \mathbb{R} \mapsto \mathbb{R}$. We shall here denote the equation (2.28) in the abstract form as

$$\mathcal{N}(u) = f.$$

Furthermore, ellipticity is defined as in the linear case.

Consider the following boundary value problem of the NPBE (2.13) [105]:

$$\left.\begin{aligned} \text{NPBE}: \quad \nabla\cdot\epsilon\nabla u - N(u) &= -\rho_f, \quad \text{in } \Omega,\\ \text{BC}: \quad u = g, &\quad \text{on } \partial\Omega\\ N(u) &= C\sum_{j=1}^{m} c_j q_j e^{-q_j u(\bar{x})}. \end{aligned}\right\} \tag{2.29}$$

The electrostatic free energy is defined by

$$E[\phi] = \int_\Omega \left[\frac{\epsilon}{2}|\nabla\phi|^2 - \rho_f\phi + N^*(\phi)\right] dV, \qquad N^*(\phi) = C\sum_{j=1}^{m} c_j\left(e^{-q_j\phi} - 1\right) \tag{2.30}$$

on

$$H_g^1(\Omega) = \{\phi \in H^1(\Omega) : \ \phi = g \quad \text{on } \partial\Omega\}. \tag{2.31}$$

Theorem 2.11 (NPBE: Existence and uniqueness, see Theorem 2.1 in [105]):

(a) The functional $E(\phi) : H^1_g(\Omega) \mapsto \mathbb{R}$ has a unique minimizer $u \in H^1_g(\Omega)$.

(b) The minimizer is bounded in $L^\infty(\Omega)$ uniformly in $\epsilon \in [\epsilon_{\min}, \epsilon_{\max}]$.

(c) The minimizer is the unique solution to the BVP of the NPBE. ◊

Proof. (a). Existence and uniqueness of minimizer. By invoking the Poincaré inequality and the fact that $N : \mathbb{R} \mapsto \mathbb{R}$ is nonnegative, there exists constants $C_1, C_2 \geq 0$ such that

$$E[\phi] \geq C_1 \|\phi\|^2_{H^1(\Omega)} - C_2 \qquad \forall\, \phi \in H^1_g(\Omega) \quad \text{and} \quad C_1, C_2 \geq 0. \tag{2.32}$$

Let $\alpha = \inf_{\phi \in H^1_g(\Omega)} E[\phi]$. Then α is finite. There exists $u_k \in H^1_g(\Omega)$, $k = 1, 2, \ldots$, such that

$$E[u_k] \to \alpha.$$

Then by the lower bound, we have that $\{u_k\}$ is bounded in $H^1(\Omega)$ and hence it has a subsequence (which is not relabeled) such that

$$u_k \to u$$

weakly in $H^1(\Omega)$ and a.e. in Ω for some $u \in H^1_g(\Omega)$. The weak convergence and Fatou's lemma imply that

$$\alpha = \lim_{k \to \infty} E[u_k] \geq E[u] \geq \alpha.$$

Finally, the uniqueness of the minimizer u follows from the strict convexity of $E[\cdot]$:

$$E[\lambda v + (1-\lambda) w] \leq \lambda E[v] + (1-\lambda) E[w] \qquad (0 < \lambda < 1).$$

(b). The boundedness of the minimizer u. Let $\phi_g \in H^1_g(\Omega)$ be such that

$$\nabla \cdot \epsilon \lambda \phi_g = -\rho_f.$$

Then ϕ_g is bounded in $L^\infty(\Omega)$ uniformly in ϵ. Let $u_0 \in H^1_g(\Omega)$ be the unique minimizer in $H^1_0(\Omega)$ of

$$H[\phi] = \int_\Omega \left[\frac{\epsilon}{2} |\nabla \phi|^2 + N^*(\phi_g + \phi) \right] dV.$$

Then $u = u_0 + \phi_g$. Prove $\|u_0\|_{L^\infty(\Omega)} \leq C$ uniform in ϵ.

Since $N(\pm\infty) = \pm\infty$, there exists $\lambda > 0$ with $N(\phi_0 + \lambda) \geq 1$ and $N(\phi_0 - \lambda) \geq -1$ a.e. in Ω. Note λ is uniform in ϵ. Define u_λ by

$$u_\lambda(\bar{x}) = \begin{cases} -\lambda & \text{if } u_0(\bar{x}) < -\lambda, \\ u_0(\bar{x}) & \text{if } |u_0(\bar{x})| \leq \lambda, \\ \lambda & \text{if } u_0(\bar{x}) > \lambda. \end{cases}$$

Then $u_\lambda \in H^1_g(\Omega)$. Therefore, we have $H[u_0] \leq H[u_\lambda]$ and $|\nabla u_\lambda| \leq |\nabla u_0|$. Hence

$$\int_\Omega N^*(\phi_g + u_0)dV \leq \int_\Omega N^*(\phi_g + u_\lambda)dV.$$

Consequently, we have by the convexity of $N^* : \mathbb{R} \to \mathbb{R}$ that

$$\begin{aligned} 0 \geq \int_{\{u_0>\lambda\}} \left[N^*(\phi_g+u_0) - N^*(\phi_g+\lambda)\right] dV + \int_{\{u_0<-\lambda\}} \left[N^*(\phi_g+u_0) - N^*(\phi_g-\lambda)\right] dV \\ \geq \int_{\{u_0>\lambda\}} N(\phi_g+\lambda)(u_0-\lambda)dV + \int_{\{u_0<-\lambda\}} N(\phi_g-\lambda)(u_0+\lambda)dV \\ \geq \int_{\{u_0>\lambda\}} (u_0-\lambda)dV - \int_{\{u_0<-\lambda\}} (u_0+\lambda)dV \\ = \int_{\{|u_0|>\lambda\}} (|u_0|-\lambda)dV \geq 0. \end{aligned} \quad (2.33)$$

Hence the Lebesgue measure of the set $V \in \Omega_+ : \{|u_0(V)| > \lambda\} = 0$. Therefore, $|u_0| \leq \lambda$ a.e. Ω_+.

(c). The minimizer is the unique solution to the BVP of the NPBE. By routine calculations, we obtain

$$\delta E[u][\eta] := \left.\frac{d}{dt}\right|_{t=0} E[u + t\eta] = 0 \qquad \forall \eta \in C^1_c(\Omega).$$

Since $u \in L^\infty(\Omega)$, we have

$$\int_\Omega \left[\epsilon \nabla u \cdot \nabla \eta - \rho_f \eta + N(u)\eta\right] dV = 0 \qquad \forall \eta \in H^1_0(\Omega).$$

Hence u is a weak solution to the BVP of the NPBE. If $\phi_0 \in H^1_g(\Omega)$ is another solution to the BVP of the NPBE (2.29), then

$$\int_\Omega \{\epsilon_\Gamma \nabla(u_0 - \phi_0) \cdot \nabla \eta + [N(u_0) - N(\phi_0)]\eta\} dV = 0$$

for any $\eta \in H^1_0(\Omega)$. Choosing $\eta = u_0 - \phi_0 \in H^1_0(\Omega)$ and invoking the convexity of $N^* : \mathbb{R} \to \mathbb{R}$, yields $u_0 = \phi_0$, which proves the uniqueness. □

Remark 2.12:

In [67], the existence and uniqueness of solutions to the NPBE was proved for the monovalent NPBE variant in (2.15) wherein the idea of the proof was based on the identification of a convex functional for which the weak form of the NPBE is the corresponding gradient mapping. Then the Ekland-Temam theorem [44] was invoked. ♢

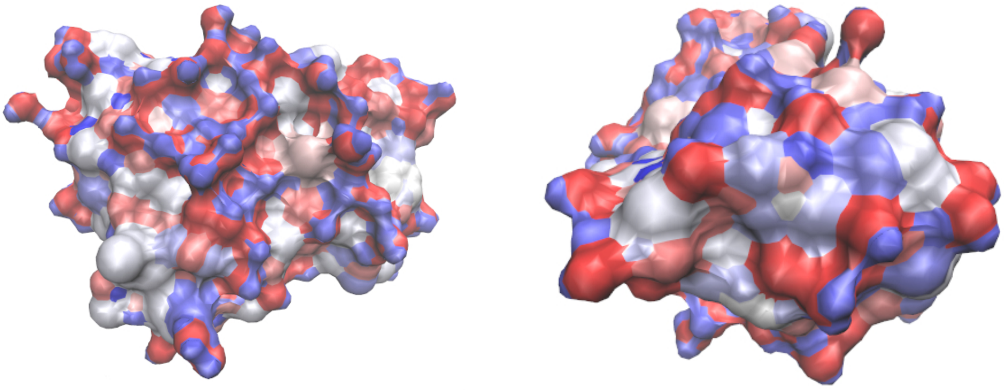


Figure 2.4.: Electrostatic potential mapped onto protein surface of *fasciculin 1* toxin CPDB entry 1FAS.

2.4. Applications of the electrostatic potential

The resultant electrostatic potential for the entire system can be used to calculate electrostatic free energies and electrostatic forces. The electrostatic free energy represents the work needed to assemble the biomolecule and is obtained by integration of the potential over a given domain of interest [41, 140]. For the LPBE, this energy is given by

$$G_{\text{elec}}[u(\bar{x})] = \frac{1}{2}\int_{\Omega} \rho_f u(\bar{x}) d\bar{x} = \frac{1}{2}\sum_{i=1}^{N_m} z_i u(\bar{x}_i), \tag{2.34}$$

where $u(\bar{x}_i)$ is the mean electrostatic potential acting on an atom i located at position $\bar{x}_i$ and carrying a charge z_i. The integral in (2.34) can be seen as the integral of polarization energy which is equivalent to the sum of interactions between charges and their respective potentials. On the other hand, it is also possible to differentiate the energy functional in (2.34) with respect to atomic positions to obtain the electrostatic force on each atom [5, 41, 55].

The electrostatic potential can also be evaluated on the surface of the biomolecule (electrostatic surface potential). It is used to provide information about the interaction between the biomolecule and other biomolecules or ligands or ions in its vicinity. Figure 2.4 shows the electrostatic potential mapped onto the surface of the protein *fasciculin 1* and was generated by the Visual Molecular Dynamics (VMD) software at different orientations [69]. The electrostatic potential is computed by our FDM solver. The red colour represents regions of negative potential, the blue colour represents regions of positive potential, and the white colour represents neutral regions.

2.4.1. Similarity index (SI) analysis of proteins

Similarity indices (SIs) are quite significant for the following reasons. Firstly, they are used in quantum mechanical calculations to compare the electron densities and electrostatic potentials of small organic compounds. The comparison can be used to derive quantitative structure-activity relationships (QSARs) [142]. Secondly, they are used for comparison of molecular electrostatic potentials generated by the PBE. In general, similarity analysis can be used to compare the interaction properties of related proteins which provides information about binding to other particles [142].

If so many protein samples are considered, then it becomes a severe computational issue. On the other hand, self-similarity indices can also be calculated by rotation of individual proteins about an axis, tasks which can be handled more conveniently by the RBM. In this case, the angle of rotation becomes the useful parameter. This is our next research focus where we shall apply solution decomposition technique (the range-separated canonical tensor format) [17] in order to modify the PBE in (2.15) so as to improve on the accuracy and to reduce the computational costs.

2.4.2. Brownian dynamics simulation (BD) and ionic strength dependence on reaction rates

Brownian dynamics simulation technique has a myriad of applications in biological systems. It may be used for example, to determine protein association rates, simulate protein-protein encounter, among others [48, 51]. Protein association rates highly depend on the ionic strength of the solution in which the interaction takes place. For instance, high ionic strengths dampen or attenuate the effect of electrostatic forces and energies of proteins, hence reducing the rates of association and vice versa. The dependence on ionic strength of the solution is an indicator of the significance of long-range electrostatic forces and hence diffusion control [51].

Works of several researchers corroborate this dependence of protein association rates on ionic strength and we here mention a few of these findings. In their research about ionic strength dependence of protein-polyelectrolyte interactions, Seyrek *et al* [131] investigated the effect of univalent electrolyte concentration on protein-polyelectrolyte complex formation. They observed that the addition of salt screened repulsions, as well as attractions, thus reduced the binding of the complex.

Pasche *et al* [118] examined the effect of ionic strength and surface charge on protein adsorption at PEGylated surfaces. They observed that at high grafting density and high ionic strength, the net interfacial force was determined by the steric barrier properties of PEG (polyethylene glycol). On the other hand, at low ionic strength, the electrical double layer thickness exceeded that of the PEG layer, therefore, the protein interactions with PLL-g-PEG coated surfaces were influenced by the surface charges shining through the PEG double layer.

In [125], the electrostatic influence on the kinetics of ligand binding to acetylcholinesterase (AChE) was investigated and distinctions between active center ligands and fasciculin were made. It was observed that reaction rates for the cationic ligands showed a strong dependence on ionic strength. Furthermore, fasciculin 2 (FAS2) showed greater ionic strength dependence than TFK$^+$ (m-trimethylammoniotrifluoroacetophenone) which is consistent with its multiple net positive charges.

The reduced basis method (RBM), which is discussed in Chapter 5, can be quite useful for such multi-parametric systems, whereby a reduced order model (ROM) can be obtained for varying ionic strengths and positions of the molecule under investigation. This ROM can make the BD computations much cheaper than using the full order model (FOM).

2.5. Discretization of the classical LPBE

2.5.1. Finite difference discretization

The LPBE in (2.21) is discretized with a centered finite differences scheme to obtain the algebraic linear system as below,

$$
\begin{aligned}
&-\frac{H}{dx^2}\epsilon^{x}_{i+\frac{1}{2},j,k}(u_{i+1,j,k}-u_{i,j,k})+\frac{H}{dx^2}\epsilon^{x}_{i-\frac{1}{2},j,k}(u_{i,j,k}-u_{i-1,j,k})-\frac{H}{dy^2}\epsilon^{y}_{i,j+\frac{1}{2},k}(u_{i,j+1,k}-u_{i,j,k})\\
&+\frac{H}{dy^2}\epsilon^{y}_{i,j-\frac{1}{2},k}(u_{i,j,k}-u_{i,j-1,k})-\frac{H}{dz^2}\epsilon^{z}_{i,j,k+\frac{1}{2}}(u_{i,j,k+1}-u_{i,j,k})+\frac{H}{dz^2}\epsilon^{z}_{i,j,k-\frac{1}{2}}(u_{i,j,k}-u_{i,j,k-1})\\
&\qquad\qquad+H\bar{\kappa}^2_{i,j,k}u_{i,j,k}=HCq_{i,j,k},
\end{aligned}
\tag{2.35}
$$

where $H = dx \times dy \times dz$ is a scaling factor, $q_{i,j,k}$ is the discretized molecular charge density and $C = 4\pi e_c^2/\kappa_B T$.

It is important to choose efficient algorithms and parameters to be used in the discretization of the charge density distribution, the kappa, and the dielectric functions that appear in the LPBE for the accuracy of the mean electrostatic potential solution. An efficient method is usually chosen to partition the domain into regions of solute (or biomolecule) and the solvent dielectric. Some of the key methods employed in APBS are the molecular surface and cubic-spline surface methods [140]. In the following subsections, we provide some insights into these discretizations.

2.5.2. Calculation of dielectric constant distribution and kappa function

We notice that the dielectric constant ϵ in equation (2.35), is discretized at half grid, and therefore, we use a staggered mesh which results in three arrays (in x, y, and z directions) representing the shifted dielectric values on different grids. The dielectric coefficients and kappa functions which are piecewise constant, are mapped according to (2.18).

Techniques used to map the dielectric and kappa functions onto the grid include, among others, the molecular surface, and the smoothed molecular surface, which are calculated using the Connolly approach [35] and the cubic-spline surface. For more information see [140]. The cubic-spline surface method, which is our method of choice, is more suitable than the other two because it is possible to evaluate the gradient of the mean electrostatic potential such as in the determination of the solvated or polar forces. This method introduces an intermediate dielectric region at the interface between the solute and the solvent because the kappa and dielectric maps are built on a cubic-spline surface. This smoothes the transition of the functions to circumvent discontinuities inherent in them [7, 140].

2.5.3. Calculation of charge densities

The molecular charge density (right-hand side of the LPBE (2.21)) can be obtained from any file with atomic coordinates, charges, and radii. However, these atomic coordinates may not coincide with any of our grid points. Therefore, it is necessary to find an efficient method of spreading the point charges (summation term in LPBE) to the grid points.

Several methods are available to map or spread the charges onto the grid points, e.g. in the APBS software package. Trilinear interpolation (or linear spline) in which charges are mapped onto nearest-neighbour grids, results in potentials which are very sensitive to the grid resolution. Cubic B-spline interpolation where charges are mapped to two layers of grid points, has an average sensitivity to the grid setup, and quintic B-spline interpolation has the lowest sensitivity to grid spacing because charges are spread out to three layers of the grid points [7].

In this study, we use the cubic B-spline interpolation (basis spline) method which maps the charges to the nearest and next-nearest grid points. Although computationally expensive, this method provides softer or smoother distributions of charges which subsequently reduces the sensitivity of the mean electrostatic potential solutions to the grid spacing [140].

2.5.4. Dirichlet boundary conditions

Analytical solutions to the LPBE can only be obtained for systems with simple geometries, for example, spherical and cylindrical systems. Equation (2.36) shows an analytical solution for a spherical molecule with uniform charge (Born ion) [67]. From this equation, we can obtain two different kinds of Dirichlet boundary conditions, the single Debye-Hückel (SDH) and multiple Debye-Hückel (MDH). For the former, we assume that all the atomic charges are collected into a single charge at the center of the solute approximated by a sphere. This kind of boundary condition is suitable when the boundary is sufficiently far from the biomolecule. On the other hand, the latter assumes the superposition of the contribution of each atomic charge (i.e. multiple, non-interacting spheres with point charges) with respective radius. This kind of boundary condition is more accurate than SDH for closer boundaries but can be computationally expensive for large biomolecules.

In this study, we employ the MDH type [7, 126],

$$u(x) = (\frac{e_c^2}{\kappa_B T}) \sum_{i=1}^{N_m} \frac{z_i e^{-\kappa(d_i - a_i)}}{\epsilon_w (1 + k a_i) d_i} \quad \text{on} \quad \partial\Omega, \quad d_i = |x - x_i|. \tag{2.36}$$

Here, z_i are the point partial charges of the protein, ϵ_w is the solvent dielectric, $\kappa = \bar{\kappa}/\sqrt{\epsilon_w}$ is a function of the ionic strength of the solution, a_i are the atomic radii, and N_m is the total number of point partial charges in the protein.

Remark 2.13:

The presence of a highly singular right-hand side of (2.15) which is described by a sum of Dirac delta distributions, introduces significant errors in the numerical solution of the PBE. For instance, for every singular charge z_i in (2.15), the electrostatic potential $u(\bar{x})$ exhibits degenerate behaviour at each atomic position $\bar{x}_i$ in the molecular region Ω_m. To overcome this problem, the PBE theory has recently received a major boost by the introduction of solution decomposition techniques which have been developed, for example, in [30, 32, 110, 151]. The idea behind these techniques is the avoidance of building numerical approximations corresponding to the Dirac delta distributions by treating the biomolecular system (see Figure 2.1) as an interface problem. This is coupled with the advantage that analytical expansions in the molecular sub-region are possible, by the Newton kernel. We discuss some of these regularization techniques in detail in Section 2.6. ♢

2.6. Recent advances in the Poisson-Boltzmann theory

We here discuss and acknowledge the recent developments of the PBE theory. Firstly, the biomolecular system has been considered as an interface problem which requires solution decomposition techniques to get rid of the solution singularities caused by the Dirac-delta distributions on the right hand side of (2.21) or (2.15). This has been discussed, for example in [54, 103, 110, 149] where the PBE has been modified into an interface problem.

Typical solution decomposition techniques for the PBE entail a coupling of two equations for the electrostatic potential in the molecular (Ω_m) and solvent (Ω_s) regions, through the boundary interface [30, 32]. The equation inside Ω_m is simply the Poisson equation, due to the absence of ions, i.e.,

$$-\nabla \cdot (\epsilon_m \nabla u) = \sum_{i=1}^{N_m} q_i \delta(\bar{x} - \bar{x}_i) \quad \text{in } \Omega_m, \tag{2.37}$$

where $q_i = \frac{4\pi e_c^2}{\kappa_B T} z_i$. On the other hand, there is absence of atoms in Ω_s. Therefore, the density is purely given by the Boltzmann distribution

$$-\nabla \cdot (\epsilon_s \nabla u) + \bar{\kappa}^2 \sinh(u) = 0 \quad \text{in } \Omega_s. \tag{2.38}$$

The two equations (2.37) and (2.38) are coupled together through the interface boundary conditions

$$[u]_\Gamma = 0, \quad \text{and} \quad \left[\epsilon \frac{\partial u}{\partial n_\Gamma}\right]_\Gamma = 0, \tag{2.39}$$

where $\Gamma := \partial\Omega_m = \partial\Omega_s \cap \Omega_m$ and $[f]_\Gamma = \lim\limits_{t \longrightarrow 0} f(\bar{x} + tn_\Gamma) - f(\bar{x} - tn_\Gamma)$ is the jump across the boundary. Here, n_Γ denotes the unit outward normal direction of the interface Γ.

We now briefly describe some of the different solution decomposition techniques which form the state of the art representation of the PBE. The first solution decomposition by [151] is generally given by

$$\left.\begin{aligned} -\epsilon_m \Delta u(\bar{x}) &= C\sum_{i=1}^{N_m} z_i \delta(\bar{x}-\bar{x}_i), && \bar{x}\in D_m,\\ -\epsilon_s \Delta u(\bar{x}) + \bar{\kappa}^2 \sinh(u(\bar{x})) &= 0, && \bar{x}\in D_s,\\ u(s^+) = u(s^-), \quad \epsilon_s \frac{\partial u(s^+)}{\partial n(s)} &= \epsilon_m \frac{\partial u(s^-)}{\partial n(s)}, && s\in \Gamma,\\ u(s) &= g(s), && s\in\partial\Omega, \end{aligned}\right\} \tag{2.40}$$

where $C = 4\pi e_c^2/\kappa_B T$, D_m the protein domain, D_s the solvent domain and Γ the interface between the protein and the solvent. The solution $u(\bar{x})$ is decomposed as follows,

$$u(\bar{x}) = G(\bar{x}) + \tilde{\phi}(\bar{x}) + \tilde{u}(\bar{x}). \tag{2.41}$$

The corresponding components of $u(\bar{x})$ include the analytical solution $G(\bar{x})$ of the Poisson equation in the molecular domain,

$$G(\bar{x}) = \frac{C}{4\pi\epsilon_m}\sum_{i=1}^{N_m}\frac{z_i}{|\bar{x}-\bar{x}_i|}, \qquad \bar{x}\in D_m, \tag{2.42}$$

the solution of the linear interface boundary value problem

$$\left.\begin{aligned} &\Delta\tilde{\phi}(\bar{x}) = 0, && \bar{x}\in D_m\cup D_s,\\ &\tilde{\phi}(s^+) = \tilde{\phi}(s^-),\ \epsilon_s\frac{\partial\tilde{\phi}(s^+)}{\partial n(s)} = \epsilon_m\frac{\partial\tilde{\phi}(s^-)}{\partial n(s)} + (\epsilon_m-\epsilon_s)\frac{\partial G(s)}{\partial n(s)}, && s\in\Gamma,\\ &u(s) = g(s) - G(s), && s\in\partial\Omega, \end{aligned}\right\} \tag{2.43}$$

and the solution of the nonlinear interface boundary value problem

$$\left.\begin{aligned} \Delta\tilde{u}(\bar{x}) &= 0, && \bar{x}\in D_m,\\ -\epsilon_s\Delta\tilde{u}(\bar{x}) + \bar{\kappa}^2\sinh(\tilde{u}(\bar{x}) + \tilde{\phi}(\bar{x}) + G(\bar{x})) &= 0, && \bar{x}\in D_s,\\ \tilde{u}(s^+) = \tilde{u}(s^-),\ \epsilon_s\frac{\partial\tilde{u}(s^+)}{\partial n(s)} &= \epsilon_m\frac{\partial\tilde{u}(s^-)}{\partial n(s)}, && s\in\Gamma,\\ u(s) &= 0, && s\in\partial\Omega. \end{aligned}\right\} \tag{2.44}$$

Secondly, we have the solution decomposition by [110] which takes the form $u(\bar{x}) = \hat{u}(\bar{x}) + \tilde{u}(\bar{x})$. The short-range part $\hat{u}(\bar{x})$ is given by

$$\hat{u}(\bar{x}) = \begin{cases} G(\bar{x}) + u^0(\bar{x}) & \text{if} \quad \bar{x}\in D_m,\\ 0 & \text{if} \quad \bar{x}\in D_s, \end{cases} \tag{2.45}$$

where $u^0(\bar{x})$ is a harmonic function which compensates for the discontinuity on the interface and satisfies the following conditions

$$\left.\begin{aligned} \Delta u^0(\bar{x}) &= 0 && \text{if} \quad \bar{x}\in D_m,\\ u^0(s) &= -G(s), && s\in\Gamma. \end{aligned}\right\} \tag{2.46}$$

The regular part $\tilde{u}(\bar{x})$ is represented by

$$\left.\begin{aligned} -\nabla\cdot(\epsilon(\bar{x})\nabla\tilde{u}(\bar{x}))+\bar{\kappa}^2(\bar{x})\sinh(\tilde{u}(\bar{x}))=0,\\ [\tilde{u}(s)]_\Gamma=0,\ [\epsilon\nabla\tilde{u}(s)\cdot\mathbf{n}]_\Gamma=-\epsilon_m\nabla(G(s)+u^0(s))\cdot\mathbf{n}|_\Gamma.\end{aligned}\right\} \tag{2.47}$$

Thirdly, the solution decomposition in [30] is as follows; $u(\bar{x})=G(\bar{x})+u^r(\bar{x})$, where $G(\bar{x})$ is as in (2.42) and the regular part is given by

$$\left.\begin{aligned} -\nabla\cdot(\epsilon\nabla u^r)+\bar{\kappa}^2(\bar{x})\sinh(u^r+G)=\nabla\cdot((\epsilon-\epsilon_m)\nabla G) &\quad \text{in } \Omega\\ u^r=g-G &\quad \text{on } \partial\Omega.\end{aligned}\right\} \tag{2.48}$$

We can also further decompose (2.48) into the linear and nonlinear components as follows $u^r(\bar{x})=u^l(\bar{x})+u^n(\bar{x})$, i.e.,

$$\left.\begin{aligned} -\nabla\cdot(\epsilon\nabla u^l)=\nabla\cdot((\epsilon-\epsilon_m)\nabla G) &\quad \text{in } \Omega\\ u^l=0 &\quad \text{on } \partial\Omega,\end{aligned}\right\} \tag{2.49}$$

and

$$\left.\begin{aligned} -\nabla\cdot(\epsilon\nabla u^n)+\bar{\kappa}^2(\bar{x})\sinh(u^n+u^l+G)=0 &\quad \text{in } \Omega\\ u^n=g-G &\quad \text{on } \partial\Omega.\end{aligned}\right\} \tag{2.50}$$

Lastly, a two component matched interface and boundary (MIB) regularization approach for charge singularities in implicit solvation was considered in [54], whereby local higher order interpolation together with the interface jump conditions are used to capture the discontinuities of the solution and the coefficients (dielectric and kappa) of the PBE.

Remark 2.14:

The solution decomposition techniques in (2.48) provides the motivation for the RS tensor format for regularizing the PBE, which is discussed in Chapter 4. It is also implemented as a solver option for the PBE solution in the well-known adaptive Poisson-Boltzmann software (APBS) package using the FEM [8]. ◊

However, the following computational challenges are inherent in the aforementioned techniques. First, all these techniques do not efficiently separate the long- and short-range components in each of the atomic volumes of the biomolecule. Rather, they split the Laplacian operator at the solute-solvent interface using the dielectric coefficient as a cutoff function, thereby creating the need to incorporate the so-called interface (or jump) conditions for the potential function at the interface in order to eliminate the solution discontinuity (e.g., Cauchy data) at the interface of complicated sub-domain shapes. Consequently, the long-range components of the free space potential are not completely decoupled from the short-range parts at each atomic radius, in the “so-called” singular function $G(\bar{x})$, in the molecular domain Ω_m.

Secondly, in the solution decomposition techniques, see, for instance, [151], multiple algebraic systems for the linear and nonlinear boundary value problems have to be solved, thereby increasing the computational costs. Thirdly, the system matrix is modified because of incorporating the interface conditions and also, for instance, the smooth function (G), in the Boltzmann distribution term in (2.48).

In this thesis, we apply a new solution decomposition approach for the regularization of the PBE by using the RS canonical tensor format which was developed in [17]. This technique relies on the independent grid-based low-rank tensor representation of the long- and short-range components in the total sum of Newton kernels discretized on a fine 3D $n^{\otimes 3}$ Cartesian grid Ω_n in the computational box $\Omega \subset \mathbb{R}^3$ [17]. This representation is based on the splitting of a single reference potential, defined by the Newton kernel, $p(\|\bar{x}\|) = 1/\|\bar{x}\|$, into a sum of localized (in the respective atomic volumes) and global low-rank canonical tensors both represented on the computational grid Ω_n.

The long-range part in the total potential of a multiparticle system is approximated by a low-rank canonical/Tucker tensor whose rank only logarithmically depends on the number of particles in the system [17]. The rank reduction algorithm is accomplished by the canonical-to-Tucker (C2T) transform through the reduced higher order singular value decomposition (RHOSVD) [82] with a subsequent Tucker-to-canonical (T2C) decomposition, see Chapter 3 for a detailed discussion. On the other hand, the short-range contributions to a multiparticle system are parametrized by a single low-rank canonical reference tensor of local support, accomplished by a list of particle coordinates and their respective charges [17].

In general, the splitting technique employed in this thesis is based on the RS tensor decomposition of the discretized Dirac delta distribution [85], which allows avoiding the nontrivial matrix reconstruction as in (2.48) and in [151]. The only requirement in this approach is a simple modification of the singular charge density of the PBE in the molecular region Ω_m, which does not change the FEM/FDM system matrix. The singular component in the total potential is recovered explicitly by the short-range component in the RS tensor splitting of the Newton potential. The main computational benefits of this approach are the localization of the modified singular charge density within the molecular region and automatic maintaining of the continuity in the Cauchy data on the interface. Furthermore, this computational scheme only includes solving a single system of FEM/FDM equations for the regularized (or long-range) component of the decomposed potential.

Remark 2.15:

The novelty of the new regularization approach for the PBE is the efficient tensor splitting of the Newton potential in the respective atomic volumes of the biomolecule, thereby eliminating the contribution of the singular electrostatic component to the jump-condition at the solute-solvent interface. ♢

2.7. Conclusions

In this chapter, we have discussed the PBE theory and its application to biomolecular simulation. The PBE is hitherto the most efficient and popular implicit solvent method whose analytical solutions for realistic biomolecules are inexistent. Therefore, only numerical techniques can offer accurate solutions to the PBE. However, these numerical techniques face severe computational challenges due to the following major reasons: (1) the rapid or strong singularities caused by the singular sources and (2) the high number of degrees of freedom resulting from the discretizations of large biomolecular systems. We have highlighted the current state of the art remedies to fix the first problem from the literature and provided their shortcomings. In Chapter 4, we develop an efficient and novel technique to construct regularized PBE models based on the range-separated tensor format. This is achieved by the efficient splitting of the Dirac delta distribution in the molecular domain and the construction of a regularized source term which is applied to the PBE in order to compute smooth long-range electrostatic potentials. Next, we tackle the second problem by applying the reduced basis method in Chapter 5 to the PBE in order to construct highly accurate surrogate reduced order models of low dimension in order to accelerate biomolecular simulations.

CHAPTER 3

INTRODUCTION TO TENSOR METHODS

Contents

In this chapter, we first discuss the rank-structured tensor representations from multilinear algebra in Section 3.1 and basic tools from tensor numerical methods in Section 3.2, which form the prerequisites for understanding the range-separated (RS) canonical/Tucker tensor format.

3.1. Multilinear algebra of tensors

In this section, we provide an elementary introduction to higher order tensors (i.e., tensors of order $d \geq 3$). There exist a lot of ideas and algorithms for the low-rank tensor representation of multi-dimensional data by utilizing the canonical (or canonical decomposition/parallel factors (CP)) and the Tucker decompositions, which were originally established in the fields of chemometrics and psychometrics [28, 92]. Later on, these developments were extended to signal processing and experimental data analysis [89, 101]. The multilinear algebra of rank-structured tensors was gestated by the early articles on polyadic (canonical) tensor decomposition [63] and the orthogonal Tucker tensor decomposition [138].

Currently, in the computer science field, a considerable amount of research is performed on tensor decomposition techniques for big data analysis, see for instance [3].

We emphasize that in this field, the tensor decompositions have been employed mostly for quantitative analysis of correlations in the multidimensional data arrays obtained, for example, from experiments. Furthermore, usually, there are no special requirements on the accuracy of the decompositions for these data.

In what follows, we introduce higher order tensors of mode d ($d \geq 3$) and their related properties. This allows us to examine the main rank-structured tensor formats which are useful in the range-separated tensor format [17].

3.1.1. Full format mode d tensors

Definition 3.1 (Tensor):

A mode d tensor is a multidimensional array of data over a d-tuple index set (or multi-indices) $\mathbf{i} = (i_1, \ldots, i_d)$, with $i_\ell \in I_\ell$, $I_\ell = \{1, \ldots, n_\ell\}$, $\ell = 1, \ldots, d$. That is

$$\mathbf{A} = [a_{i_1,\ldots,i_d}] \equiv [a(i_1, \ldots, i_d)] \in \mathbb{R}^I, \tag{3.1}$$

where the Cartesian product of the index set is given by $I = I_1 \times \cdots \times I_d$. A tensor $\mathbf{A}$ is considered as an element of a linear vector space $\mathbb{V}_n = \otimes_{\ell=1}^d \mathbb{R}^{I_\ell}$ of real-(or complex-) valued dth-order tensors. It is equipped with the Euclidean scalar product $\langle \cdot, \cdot \rangle : \mathbb{V}_n \times \mathbb{V}_n \to \mathbb{R}$ defined as

$$\langle \mathbf{A}, \mathbf{B} \rangle := \sum_{(i_1,\ldots,i_d)\in I} a_{i_1,\ldots,i_d} b_{i_1,\ldots,i_d} \quad \text{for } \mathbf{A},\, \mathbf{B} \in \mathbb{V}_n, \tag{3.2}$$

and $\|\mathbf{A}\|_F := \sqrt{\langle \mathbf{A}, \mathbf{A} \rangle}$ is the corresponding Frobenius norm. ◊

When all the dimensions have equal size, $n_\ell = n$, the resultant tensor is referred to as $n^{\otimes d}$ tensor. Unless both d and n are small, $n^{\otimes d}$ is quite large, and in some occasions it may exceed the computer memory due to the exponential scaling in n^d, commonly known as "curse of dimensionality". This exponential growth in the dimension can be gotten rid of by applying the rank-structured tensor representations of multidimensional tensors [17].

Definition 3.2 (Order):

The order (also known as modes or ways) of a tensor is the number of dimensions. We therefore can define a scalar as a zeroth-order tensor, a vector as a mode (order)-1 tensor, a matrix as a mode-2 tensor and tensors of order three or higher are known as higher-order tensors. ◊

Figure 3.1.: Column, row, and tube fibers of a mode-3 tensor.

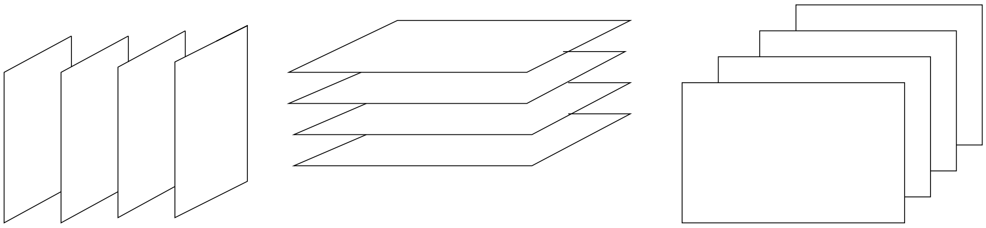


Figure 3.2.: Lateral, horizontal, and frontal slices of a mode-3 tensor.

Definition 3.3 (Fibers):

Fibers are higher-order analogues of matrix rows and columns whereby every index but one is fixed. A matrix column is a mode-1 fiber and a matrix row is a fiber of mode-2. For third-order tensors, we have column, row, and tube fibers which are denoted by $x_{:jk}$, $x_{i:k}$, and $x_{ij:}$, respectively, see Figure 3.1. $\Diamond$

Definition 3.4 (Slices):

Slices are 2D sections of a tensor and are defined by fixing all except two indices. We have the horizontal, lateral, and frontal slices in a mode-3 tensor denoted by $\mathbf{X}_{i::}$, $\mathbf{X}_{:j:}$, and $\mathbf{X}_{::k}$, respectively, see Figure 3.2. $\Diamond$

Definition 3.5 (Rank):

The tensor rank, $R = \operatorname{rank}(\mathcal{X})$ of a tensor $\mathcal{X}$ is defined as the minimum number of rank-one tensors, which are needed to produce $\mathcal{X}$ as their sum. Therefore, we can write a rank-R matrix as $A = \sum_{r=1}^{R} \sigma_r a_r \otimes b_r$ and a rank-R mode-3 tensor as $\mathcal{X} = \sum_{r=1}^{R} \sigma_r a_r \otimes b_r \otimes c_r$. $\Diamond$

Definition 3.6 (Vectorization):

Vectorization is the reordering of a tensor $\mathcal{X}$ into a vector by vertically stacking the columns of $\mathcal{X}$ into a tall vector. For simplicity, consider for example, a matrix $A \in \mathbb{R}^{M\times N}$. The resulting vectorized form is given by

$$\operatorname{vec}(A) = \begin{bmatrix} a_{:1} \\ a_{:2} \\ \vdots \\ a_{:N} \end{bmatrix}. \tag{3.3}$$

◊

Definition 3.7 (Matricization):

Matricization is the transformation of a tensor into a matrix. Also known as unfolding or flattening, it involves reordering of the elements of an d-order tensor into a matrix whose columns are the respective fibers along ℓ-mode, $\ell = 1, \dots, d$. For instance, a $3 \times 2 \times 5$ tensor can be arranged as a 6×5 matrix or 2×15 matrix.

Note that the unfolding of a tensor into a matrix can reduce some multilinear algebraic operations for tensors of order d ($d \geq 3$) to the standard linear algebra. Figure 3.3 shows the unfolding of a 3D tensor for mode $\ell = 1$. Precisely, the unfolding of a tensor $\mathbf{A} \in \mathbb{R}^{I_1\times\cdots I_d}$ along mode ℓ is a matrix of size $n_\ell \times (n_{\ell+1} \cdots n_d n_1 \cdots n_{\ell-1})$, which can be denoted by

$$\mathbf{A}_\ell = [a_{ij}] \in \mathbb{R}^{n_\ell \times (n_{\ell+1}\cdots n_d n_1 \cdots n_{\ell-1})}, \tag{3.4}$$

whose columns are the respective fibers of $\mathbf{A}$ along the ℓth mode.

Consider $x \in \mathcal{X}$ to be an element of a tensor and $m \in \mathbf{M}$ an element of the unfolded tensor. Then the mode-n matricization can be defined through the following mapping

$$x_{i_1,i_2,\cdots,i_N} \mapsto m_{i_n,j} \quad \text{with } j = 1 + \sum_{\substack{k=1\\k\neq n}}^{N} \left((i_k - 1) \prod_{\substack{m=1\\m\neq n}}^{k-1} I_m \right). \tag{3.5}$$

◊

We present a simple example from [124] to demonstrate the formula in (3.5) for the matricization of a mode-3 tensor.

Example 3.1:

Let $\mathcal{X}$ be a tensor with the following frontal slices:

$$X_1 = \begin{bmatrix} 1 & 4 & 7 & 10 \\ 2 & 5 & 8 & 11 \\ 3 & 6 & 9 & 12 \end{bmatrix} \qquad X_2 = \begin{bmatrix} 13 & 16 & 19 & 22 \\ 14 & 17 & 20 & 23 \\ 15 & 18 & 21 & 23 \end{bmatrix}.$$

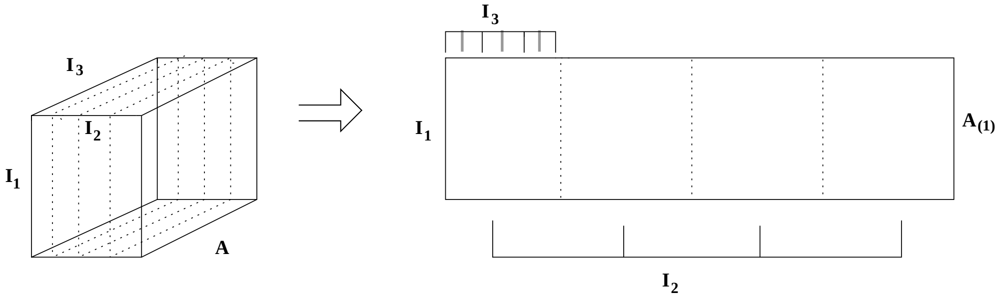


Figure 3.3.: Unfolding of the 3D tensor $\mathbf{A}$ to the $(I_1 \times I_2 I_3)$-matrix $\mathbf{A}_{(1)}$.

Then the three mode-n matricizations are given by:

$$X_{(1)} = \begin{bmatrix} 1 & 4 & 7 & 10 & 13 & 16 & 19 & 22 \\ 2 & 5 & 8 & 11 & 14 & 17 & 20 & 23 \\ 3 & 6 & 9 & 12 & 15 & 18 & 21 & 23 \end{bmatrix} \qquad X_{(2)} = \begin{bmatrix} 1 & 2 & 3 & 13 & 14 & 15 \\ 4 & 5 & 6 & 16 & 17 & 18 \\ 7 & 8 & 9 & 19 & 20 & 21 \\ 10 & 11 & 12 & 22 & 23 & 24 \end{bmatrix}$$

$$X_{(3)} = \begin{bmatrix} 1 & 2 & 3 & 4 & \cdots & 9 & 10 & 11 & 12 \\ 13 & 14 & 15 & 16 & \cdots & 21 & 22 & 23 & 24 \end{bmatrix}. \qquad \Diamond$$

Definition 3.8 (Contracted product):

Contracted product of two tensors is another essential tensor operation which is analogous to the matrix-matrix multiplication. The difference is that for matrices, they are positioned accordingly to allow multiplication over the compatible sizes. On the other hand, the contraction mode ℓ must be determined explicitly for tensors. Therefore, given a tensor $\mathbf{A} \in \mathbb{R}^{I_1 \times \cdots \times I_d}$ and a matrix $B \in \mathbb{R}^{J_\ell \times I_\ell}$, the corresponding mode-ℓ tensor-matrix product is given by

$$C = \mathbf{A} \times_\ell B \in \mathbb{R}^{I_1 \times \cdots \times I_{\ell-1} \times J_\ell \times I_{\ell+1} \times \cdots \times I_d} \tag{3.6}$$

where the sign "$\times_\ell$" denotes the contraction over the mode number ℓ and

$$c_{i_1 \ldots i_{\ell-1} j_\ell i_{\ell+1} \ldots i_d} = \sum_{i_\ell=1}^{n_\ell} a_{i_1 \ldots i_{\ell-1} i_\ell i_{\ell+1} \ldots i_d} b_{j_\ell i_\ell}, \quad j_\ell \in J_\ell.$$

The following sequence of operations can be used to perform a given contraction:

- matrix unfolding (or reshaping) of the tensor;
- matrix-matrix multiplication over the corresponding dimension;
- reshaping of the resultant matrix into a tensor. $\Diamond$

Remark 3.9:

The contracted product between a tensor and a matrix can be applied successively along several modes, and can be shown to be commutative, i.e.,

$$(\mathbf{A} \times_\ell B) \times_m P = (\mathbf{A} \times_m P) \times_\ell B = A \times_\ell B \times_m P, \quad \ell \neq m. \tag{3.7}$$

♢

Remark 3.10:

The unfolding operation in MATLAB is accomplished by the "reshape" command, which is done w.r.t the first given variable. The unfolding w.r.t other variables, a corresponding permutation is necessary by using the "permute" command. See [79, 102], for more details. We demonstrate the corresponding MATLAB implementation of an $8 \times 10 \times 6$ tensor unfolding in the following script. ♢

```
A = rand(nx,ny,nz) % generate a 3D tensor with random coefficients

% Unfolding along mode 1:
A1 = reshape(A,nx,ny*nz);

% Unfolding along mode 2:
B = permute(A,[2,1,3]);
A1 = reshape(B,ny,nz*nx);

% Unfolding along mode 3:
C = permute(A,[3,2,1]);
A1 = reshape(C,nz,nx*ny);
```

3.1.2. Rank-structured tensor formats

The ubiquitous notion of rank-structured tensor formats for the compressed approximation of multidimensional data is commonly understood for instance, in the context of nonlinear parametrization by a small number of parameters which permit low storage costs and efficient multilinear algebra through reduction to univariate operations [17].

Definition 3.11 (Rank-1 tensor):

The basic separable element is the rank-1 tensor given by

$$\mathbf{A} = \mathbf{a}^{(1)} \otimes \cdots \otimes \mathbf{a}^{(d)} \in \mathbb{R}^{n_1 \times \cdots \times n_d}, \tag{3.8}$$

where the entries $a_{i_1,\ldots,i_d} = a^{(1)}_{i_1} \cdots a^{(d)}_{i_d}$ require only $(n_1 + \ldots + n_d) \ll n^d$ bytes to store it. If $n_\ell = n$, then the storage cost is $dn \ll n^d$. ♢

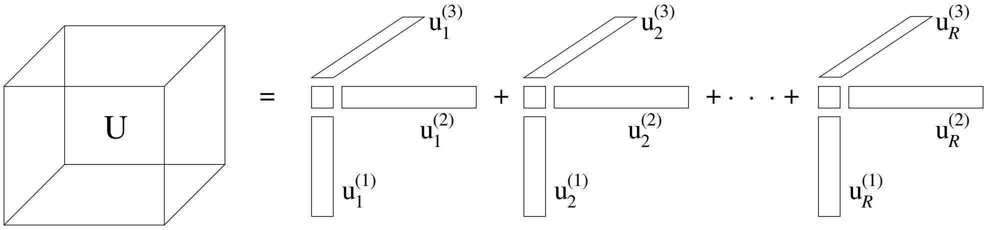


Figure 3.4.: Canonical tensor decomposition of a 3rd order tensor.

3.1.2.1. R-term canonical tensor format

The R-term canonical tensor format, also known as CANDECOMP/PARAFAC (CP), i.e., canonical decomposition and the parallel factors [87] is defined by a finite sum of rank-1 tensors

$$\mathbf{A} = \sum_{k=1}^{R} \xi_k \mathbf{a}_k^{(1)} \otimes \cdots \otimes \mathbf{a}_k^{(d)}, \quad \xi_k \in \mathbb{R}, \tag{3.9}$$

where $\mathbf{a}_k^{(\ell)} \in \mathbb{R}^{n_\ell}$ are normalized vectors, and $R \in \mathbb{R}_+$, the canonical rank. The storage cost is bounded by dRn. The entries of the 3D canonical tensor (3.9) are computed as the sums of elementwise products,

$$a_{i_1,i_2,i_3} = \sum_{k=1}^{R} \xi_k a_{i_1,k}^{(1)} \cdot a_{i_2,k}^{(2)} \cdot a_{i_3,k}^{(3)}. \tag{3.10}$$

The diagrammatic representation of a third-order canonical tensor format is shown in Figure 3.4.

Remark 3.12:

The origins of the CP lie in psychometrics from 1970, but their application has been extended to chemometrics, neuroscience, and in array processing [90]. However, stable algorithms to compute the nearly optimal rank-R canonical representation of a full size tensor are hitherto unavailable (it is an ill-posed problem). To circumvent this drawback, in [17, 80, 82] the authors developed the multigrid accelerated canonical to Tucker (C2T) transform which utilizes the reduced higher order singular value decomposition (RHOSVD) in order to obtain efficiently the low rank approximation to the original full size tensor. ◊

3.1.2.2. Rank-$\mathbf{r}$ orthogonal Tucker format

The Tucker decomposition was pioneered by Tucker in 1963 [139] and like the CP, it goes by several names, for example, HOSVD, N-mode SVD, three-mode (principal component analysis) PCA [90]. The rank-$\mathbf{r}$ orthogonal Tucker format for a tensor $\mathbf{V}$ is

$$\mathbf{V} = \sum\nolimits_{\nu_1=1}^{r_1} \cdots \sum\nolimits_{\nu_d=1}^{r_d} \beta_{\nu_1,\ldots,\nu_d} \mathbf{v}_{\nu_1}^{(1)} \otimes \cdots \otimes \mathbf{v}_{\nu_d}^{(d)} \equiv \boldsymbol{\beta} \times_1 V^{(1)} \times_2 V^{(2)} \ldots \times_d V^{(d)}, \tag{3.11}$$

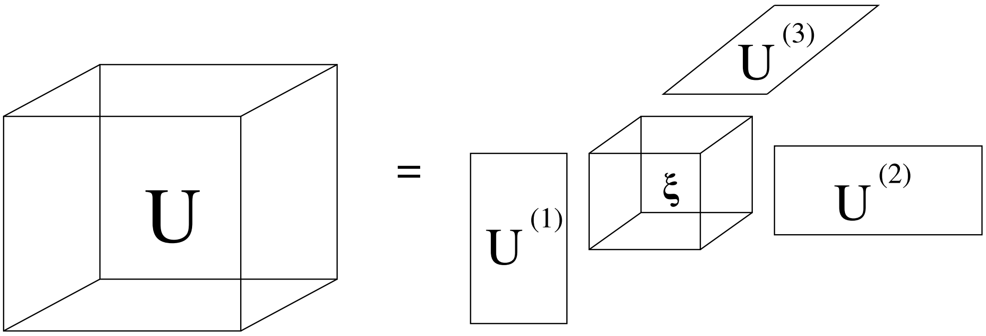


Figure 3.5.: Tucker tensor decomposition of a 3rd order tensor.

where $\{\mathbf{v}_{\nu_\ell}^{(\ell)}\}_{\nu_\ell=1}^{r_\ell} \in \mathbb{R}^{n_\ell}$ is the set of orthonormal vectors for $\ell = 1, \ldots, d$. Here, $\times_\ell$ denotes the contraction along the mode ℓ with the orthogonal matrices $V^{(\ell)} = [\mathbf{v}_1^{(\ell)}, \ldots, \mathbf{v}_{r_\ell}^{(\ell)}] \in \mathbb{R}^{n_\ell \times r_\ell}$. $\boldsymbol{\beta} = \beta_{\nu_1,\ldots,\nu_d} \in \mathbb{R}^{r_1 \times \cdots r_d}$ is the Tucker core tensor. The canonical representation (3.9) can be written as the rank-$(R, \ldots, R)$ (nonorthogonal) Tucker tensor

$$\mathbf{A} = \boldsymbol{\xi} \times_1 A^{(1)} \times_2 A^{(2)} \ldots \times_d A^{(d)}, \tag{3.12}$$

by introducing the side matrices $A^{(d)} = [\mathbf{a}_1^{(\ell)}, \ldots, \mathbf{a}_R^{(\ell)}] \in \mathbb{R}^{n_\ell \times R}$ which are obtained by concatenating the canonical vectors $\mathbf{a}_k^{(\ell)}$, $k = 1, \ldots, R$.

3.2. Some basics of tensor numerical methods

Tensor-structured numerical methods are now becoming popular in scientific computing due to their intrinsic property of reducing the grid-based solution of multidimensional problems to essentially "one-dimensional" computations. These methods evolved from bridging of the traditional rank-structured tensor formats of multilinear algebra [90, 134] with the nonlinear approximation theory based on a separable representation of multidimensional functions and operators [53, 56, 83].

The breakthrough in the tensor numerical methods began when Khoromskij in 2006 proved that for some classes of function-related tensors, the approximation error of Tucker decomposition decays exponentially with respect to the Tucker rank [83], i.e.,

$$\|\mathbf{A}_{(r)} - \mathbf{A}_0\| \leq Ce^{-\alpha r}, \qquad \text{with} \quad r = \min_\ell r_\ell, \tag{3.13}$$

where $\mathbf{A}_0$ and $\mathbf{A}_{(r)}$ are the initial tensor and its "Tucker-rank-r" approximation, respectively, and r_ℓ is the minimal Tucker rank.

One of the ingredients in the development of tensor methods was the canonical-to-Tucker (C2T) algorithm, see Algorithm 3.1 and the reduced higher order singular value decomposition (RHOSVD). The HOSVD allows to find the mapping matrices in (3.11) without constructing full-size tensors [82]. Originally, C2T was invented for the reduction of the ranks of canonical tensors when calculating three-dimensional convolution integrals in computational quantum chemistry [82], see also the monographs [79, 86] and the references therein.

Remark 3.13:

For $d = 2$, the orthogonal Tucker format is equivalent to the singular value decomposition (SVD) of a rectangular matrix. On the other hand, the canonical representation is analogous to the rank-R skeleton matrix decomposition. In general, rank-structured tensor approximations provide rapid multilinear algebra with linear complexity scaling in the dimension d. Algorithm 3.1 provides the procedure for transforming a 3D canonical tensor into the Tucker tensor format in order to determine the corresponding low-rank approximation. The Tucker-to-canonical (T2C) transform can be found in [79]. ♢

Remark 3.14:

It is worth noting that numerical studies in [79] established that the ALS step in the C2T transform in Algorithm 3.1 is usually not required for the case of tensors obtained through the grid-based representation of functions which describe physical quantities in electronic structure calculations. In such cases, only the RHOSVD approximation of the canonical tensor suffices. On the other hand, the C2T transform can be significantly accelerated using the multigrid C2T transform [79, 82]. ♢

3.2.1. Canonical tensor representation of the Newton kernel

Nonlinear separable approximations of the analytic kernels by using sums of Gaussian functions have been presented in the mathematical and chemical literature [23, 24, 25]. The grid-based method for the canonical tensor representation of a spherically symmetric Newton kernel function $p(\|\bar{x}\|) = 1/\|\bar{x}\|$, $\bar{x} \in \mathbb{R}^d$ by using sinc approximation method was introduced in [56]. In [19], the efficient numerical scheme for grid-based low-rank approximation of the Newton kernel in 3D was introduced and implemented. It is based on the projection onto a set of tensor-product piecewise constant basis functions. The canonical representations of the Newton $1/\|\bar{x}\|$, and Yukawa $e^{-\lambda\|\bar{x}\|}/\|\bar{x}\|$ kernels have been presented in [19], where the kernels were discretized by the Galerkin projection onto the tensor-product piecewise polynomial basis functions. In this work, we use the collocation-projection discretization due to their cheap computational costs over the Galerkin methods, [84].

Algorithm 3.1: *Canonical-to-Tucker transform*, [15]

```
Data: Side matrices $U^{(\ell)} = [\mathbf{u_1}^{(\ell)} \dots \mathbf{u_R}^{(\ell)}] \in \mathbb{R}^{n_\ell \times R}$, $\ell = 1,2,3$, comprising of the vectors $\mathbf{u_k}^{(\ell)} \in \mathbb{R}^{n_\ell}$,
      $k = 1,\dots,R$; maximal Tucker-rank parameter $\mathbf{r}$; maximal number of the ALS iterations $m_{\max}$ (usually
      a small number).
Result: The Tucker core tensor $\beta$ and the orthogonal side matrices $\tilde{Z}^\ell$, $\ell = 1,2,3$.
1 Compute the SVD of the side matrices:
      $$U^{(\ell)} = Z^{(\ell)} D^{(\ell)} V^{(\ell)}, \quad \ell = 1,2,3.$$
  Truncate the singular vectors in $Z^{(\ell)}$ and the corresponding singular values up to the given rank threshold to
  obtain small orthogonal matrices $Z_0^{(\ell)} \in \mathbb{R}^{n_\ell \times r_\ell}$, and $D_{\ell,0} \in \mathbb{R}^{r_\ell \times r_\ell}$, $\ell = 1,2,3$.
2 Project $U^{(\ell)}$ onto the orthogonal basis set defined by $Z_0^{(\ell)}$
      $$U^{(\ell)} \mapsto \tilde{U}^{(\ell)} = (Z_0^{(\ell)})^T U^{(\ell)} = D_{\ell,0} {V_0^{(\ell)}}^T, \quad \tilde{U}^{(\ell)} \in \mathbb{R}^{r_\ell \times R}, \quad \ell = 1,2,3$$
  and compute $\mathbf{U}^0_{(r)}$ using the formula $\mathbf{U} \mapsto \mathbf{U}^0_{(r)} = \beta \times_1 U_0^{(1)} \times_2 U_0^{(2)} \times_3 U_0^{(3)}$.
3 (Finding dominating spaces). Implement the ALS iteration in step (4) $m_{\max}$ times at most, starting from the
  RHOSVD initial guess $\mathbf{U}^0_{(r)}$
4 for $\ell = 1,2,3$: ALS iteration do
      • For $\ell = 1$: construct the partially projected image of the full tensor,
            $$\mathbf{U} \mapsto \tilde{\mathbf{U}}_1 = \sum_{k=1}^{R} c_k \mathbf{u}_k^{(1)} \otimes \tilde{\mathbf{u}}_k^{(2)} \otimes \tilde{\mathbf{u}}_k^{(3)}, \quad c_k \in \mathbb{R}. \qquad (*)$$
        Here, $\mathbf{u}_k^{(1)} \in \mathbb{R}^{n_1}$ is in the physical space for mode $\ell = 1$, while $\tilde{\mathbf{u}}_k^{(2)} \in \mathbb{R}^{r_2}$ and $\tilde{\mathbf{u}}_k^{(3)} \in \mathbb{R}^{r_3}$, the
        column vectors of $\tilde{U}^{(2)}$ and $\tilde{U}^{(3)}$, respectively, live in the index sets of $Z^\ell$-projections.
      • Reshape the tensor $\tilde{\mathbf{U}}_1 \in \mathbb{R}^{n_1 \times r_2 \times r_3}$ into a matrix $M_{U_1} \in \mathbb{R}^{n_1 \times (r_2 r_3)}$, representing the span of
        the optimized subset of mode-1 columns of the partially projected tensor $\tilde{\mathbf{U}}_1$.
        Compute the SVD of $M_{U_1}$:
            $$M_{U_1} = Z^{(1)} S^{(1)} V^{(1)},$$
        and truncate the set of singular vectors in $Z^{(1)} \mapsto \tilde{Z}^{(1)} \in \mathbb{R}^{n \times r_1}$, according to the restriction on
        the mode-1 Tucker rank, $r$.
      • Update the current approximation to the mode-1 dominating subspace, $Z^{(1)}_{r_1} \mapsto \tilde{Z}^{(1)}$.
      • Implement the single step of the ALS iteration for mode $\ell = 2$ and $\ell = 3$.
5 Repeat the complete ALS iteration $m_{\max}$ times to obtain the optimized Tucker orthogonal side matrices $\tilde{Z}^{(1)}$,
  $\tilde{Z}^{(2)}$, $\tilde{Z}^{(3)}$, and final projected image $\tilde{\mathbf{U}}_3$.
6 Project the final iterated tensor $\tilde{\mathbf{U}}_3$ in $(*)$ using the resultant basis set[1] in $\tilde{Z}^{(3)}$ to obtain the core tensor,
  $\beta \in \mathbb{R}^{r_1 \times r_2 \times r_3}$.
```

---

[1] A basis set is a group of linearly independent vectors which span a given vector space [100].

Definition 3.15 (Newton potential):

The Newton potential of an integrable function (or a Radon measure) f with compact support in $\mathbb{R}^3$ is defined as the convolution

$$u(\bar{x}) = \Gamma_N * f(\bar{x}) = \int_{\mathbb{R}^3} \Gamma_N(\bar{x} - \bar{y}) f(\bar{y}) dy, \tag{3.14}$$

where the Newton kernel $\Gamma_N = 1/\|\bar{x}\|$, has a mathematical singularity at the origin, and $\bar{y} \in \mathbb{R}^3$ [116]. The Newton potential $u(\bar{x})$ satisfies the Poisson equation

$$-\Delta u = f, \qquad \text{in } \mathbb{R}^3. \tag{3.15}$$

♢

Consider the single particle Newton potential (or the Newton kernel) $1/\|\bar{x}\|$, $\bar{x} \in \mathbb{R}^3$, which is a fundamental solution to the Poisson equation. It is well known that determining a weighted sum of interaction potentials (or Newton kernels), $P_N(\bar{x})$ in a large N_m-particle system, with the particle locations at $\bar{x}_i \in \mathbb{R}^3$, $i = 1, ..., N_m$, i.e.,

$$P_N(\bar{x}) = \sum_{i=1}^{N_m} \frac{q_i}{\epsilon_m \|\bar{x} - \bar{x}_i\|}, \qquad \bar{x}_i, \bar{x} \in \Omega = [-b, b]^3, \tag{3.16}$$

is quite computationally demanding. The Newton kernel exhibits a slow polynomial decay in $1/\|\bar{x}\|$ as $\|\bar{x}\| \to \infty$. Consequently, it has a singularity at $\bar{x} = (0, 0, 0)$, making its accurate grid representation difficult. The RS tensor format [17] can be exploited to construct an efficient technique for the grid-based calculation of $P_N(\bar{x})$ in multiparticle systems.

Remark 3.16:

Notice that the Newton potential $P_N(\bar{x})$ in (3.16) is a special case of (3.14) for a non-compact function

$$f(x) = \frac{1}{\epsilon_m} \sum_{i=1}^{N_m} q_i \delta(\bar{x} - \bar{x}_i). \tag{3.17}$$

♢

To obtain the canonical tensor representation of the Newton kernel, we follow [17] and consider the computational domain $\Omega = [-b, b]^3$, and introduce the uniform $(n^{\otimes 3})$[2] rectangular Cartesian grid Ω_n with mesh size $h = 2b/n$ (n even). Let $\{\psi_i\}$ be a set of tensor-product piecewise constant basis functions as in Definition 3.17, for the 3-tuple index $i = (i_1, i_2, i_3)$, $i_\ell \in I_\ell = \{1, ..., n\}$, $\ell = 1, 2, 3$. Here, the symbol $\prod$ is the pi-product for elementwise multiplication. The goal is to discretize the Newton kernel by

---

[2] $n^{\otimes 3} = n \times n \times n$ is a tensor representation of the 3D Cartesian grid.

its projection onto $\{\psi_i\}$ as follows

$$\mathbf{P} := [p_i] \equiv [p(i_1, i_2, i_3)] \in \mathbb{R}^{n^{\otimes 3}}, \quad p_i = \int_{\mathbb{R}^3} \frac{\psi_i(\bar{x})}{\|\bar{x}\|} \, \mathrm{d}\bar{x}, \tag{3.18}$$

where p_i is obtained from the vectors of the canonical tensor representation of the Newton kernel.

Definition 3.17 (Tensor-product basis functions):

The tensor-product piecewise constant basis functions $\psi_\mathbf{i}$ associated with the equidistant tensor-product lattice $\omega_\mathbf{d} := \omega_1 \times \cdots \times \omega_d$ of size $h = 2b/n$ are given by

$$\psi_\mathbf{i}(\bar{x}) = \prod_{\ell=1}^{d} \psi_{i_\ell}^{(\ell)}(x_\ell), \quad \text{with} \quad x_\ell \in \Omega_\ell, \quad \ell := 1, \ldots, d, \tag{3.19}$$

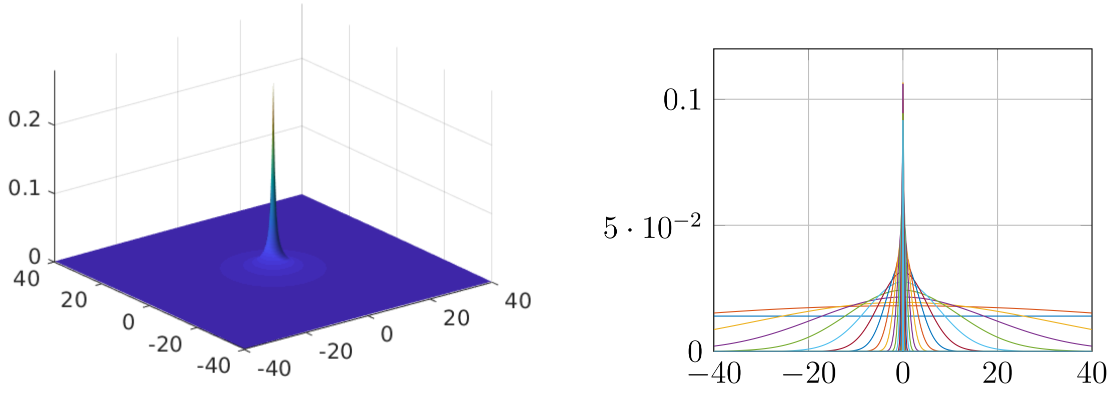
for the 3-tuple index $\mathbf{i} = (i_1, i_2, i_3)$, $i_\ell \in I = \{1, \ldots, n\}$. Here, $\psi_{i_\ell}^{(\ell)} = \chi_{\Omega_{i_\ell}}$ is the characteristic function of Ω_{i_ℓ} [84]. ♢

We provide in Algorithm 3.2 the procedure for determining the low-rank canonical decomposition of $\mathbf{P}$ based on the exponentially convergent sinc-quadrature approximation of the Gaussian (or Laplace-Gauss) transform to the Newton kernel [17, 81].

Note that the skeleton vectors $\mathbf{p}_k^{(\ell)} \in \mathbb{R}^n$ can be renumerated by $k \mapsto q = k + M + 1$, $\mathbf{p}_k^{(\ell)} \mapsto \mathbf{p}_q^{(\ell)}$, where $q = 1, \ldots, R$ and $\ell = 1, 2, 3$. The canonical tensor $\mathbf{P}_R$ (3.27) approximates the discretized 3D symmetric Newton kernel $\frac{1}{\|\bar{x}\|}$, centered at the origin such that $\mathbf{p}_q^{(1)} = \mathbf{p}_q^{(2)} = \mathbf{p}_q^{(3)}$, $(q = 1, \ldots, R)$. In this case, the terms $\mathbf{p}_k^{(\ell)}$ and $\mathbf{p}_{-k}^{(\ell)}$ are equivalent and hence the sum in (3.27) reduces to $k = 0, 1, \ldots, M$, which implies that $R = M + 1$.

The canonical vectors in the tensor representation (3.27) for the Newton kernel along the x-axis from a set $\{\mathbf{p}_q^{(1)}\}_{q=1}^{R}$ are displayed in Figure 3.6. It is clearly evident that there are canonical vectors representing the short-, intermediate-, and the long-range components of the total electrostatic potential. This fascinating feature was recognized in [81], for the rank-structured tensor approximation of the grid-based lattice summation of electrostatic potentials.

In the next section, we present the procedure of splitting (separating) the vectors of the canonical tensor representation into the short- and long-range parts. The Newton (Coulomb) electrostatic potential can then be obtained by transforming the respective canonical vectors back to the original tensor format, as in Figure 3.6 (left).



(a) Newton potential.

(b) Total canonical vectors.

Figure 3.6.: The Newton electrostatic potential (a), and the vectors of the canonical tensor representation (b) for the Newton kernel for $n = 1024$ and $R = 20$.

3.3. Range-separated (RS) canonical/Tucker tensor format

In the recent years, tensor-based approaches were suggested as new methods for the calculation of multiparticle long-range interaction potentials [79, 81]. For a given non-local generating kernel $p(\|\bar{x}\|)$, $\bar{x} \in \mathbb{R}^3$, the calculation of the weighted sum of interaction potentials in an N-particle system, with the particle locations at $\bar{x}_\nu \in \mathbb{R}^3$, $\nu = 1, ..., N$,

$$P(\bar{x}) = \sum\nolimits_{\nu=1}^{N} z_\nu \, p(\|\bar{x} - \bar{x}_\nu\|), \quad z_\nu \in \mathbb{R}, \quad \bar{x}_\nu, \bar{x} \in \Omega = [-b, b]^3, \tag{3.20}$$

is computationally demanding for large N.

Since the generating radial basis function $p(\|\bar{x}\|)$ exhibits a slow polynomial decay in $1/\|\bar{x}\|$ as $\|\bar{x}\| \to \infty$, it follows that each individual term in (3.20) contributes essentially to the total potential at each point in the computational domain Ω. This predicts the $\mathcal{O}(N)$ complexity for a straightforward summation at every fixed space point $\bar{x} \in \mathbb{R}^3$. Moreover, in general, the radial function $p(\|\bar{x}\|)$ has a singularity or a cusp at the origin, $\bar{x} = 0$, making its accurate grid representation problematic. An efficient numerical scheme for the grid-based calculation of $P(\bar{x})$ in multiparticle systems can be constructed by using the RS tensor format [17].

3.3.1. RS tensor splitting of a single Newton kernel

Following the definition of the quadrature in (3.22) and (3.23), we can clearly notice that the full set of Gaussian (quadrature) approximations include two classes of functions. The first set contains the short-range functions which are characterized by their small "effective support". The second set are the smooth long-range functions which are characterized by their global support. The range-separated tensor format is a new scheme which was recently developed in [15, 17] with the aim of independent grid-based low-rank tensor representation of the localized and global parts in the target tensor allowing for the efficient numerical approximation of N-particle interaction potentials.

Algorithm 3.2: *Low-rank canonical decomposition of $\mathbf{P}_R$ in (3.18)* [17, 86]

Data: The Newton kernel $1/\|\bar{x}\|$, tensor product basis functions $\psi_i(\bar{x})$ and the computational domain Ω.
Result: $\mathbf{P}_R$.

1 Determine the Gaussian transform to the Newton kernel in $\mathbb{R}^3$,

$$\frac{1}{\|\bar{x}\|} = \frac{2}{\sqrt{\pi}} \int_{\mathbb{R}} e^{-t^2\|\bar{x}\|^2} \mathrm{d}t = \frac{1}{\sqrt{\pi}} \int_{\mathbb{R}} \prod_{\ell=1}^{3} e^{-t^2(x_\ell)^2} \mathrm{d}t \quad \text{for} \quad \|\bar{x}\| > 0. \tag{3.21}$$

2 For any fixed $\bar{x} = (x_1, x_2, x_3) \in \mathbb{R}^3$, apply sinc-quadratures, see [19, 56] for details, to approximate the Gaussian transform in (3.21), i.e.,

$$\frac{1}{\|\bar{x}\|} = \frac{2}{\sqrt{\pi}} \int_{\mathbb{R}+} e^{-t^2\|\bar{x}\|^2} \mathrm{d}t \approx \sum_{k=-M}^{M} g_k e^{-t_k^2\|\bar{x}\|^2} = \sum_{k=-M}^{M} g_k \prod_{\ell=1}^{3} e^{-t_k^2 \bar{x}_\ell^2}. \tag{3.22}$$

The quadrature parameters (quadrature points and weights) in (3.22) are given by

$$t_k = k\mathfrak{h}_M, \quad g_k = 2\mathfrak{h}_M/\sqrt{\pi}, \quad \text{with } \mathfrak{h}_M = C_0 \log(M)/M, \quad C_0 \approx 3. \tag{3.23}$$

Under the assumption that $0 < a \leq \|x\| \leq A < \infty$, the quadrature provides an exponential convergence rate in M for a class of analytic functions $p(\|x\|)$, i.e.,

$$\left| p(\|x\|) - \sum_{k=-M}^{M} g_k e^{-t_k^2\|x\|^2} \right| \leq \frac{C}{a} e^{-\beta\sqrt{M}} \quad \text{with some} \quad C, \beta \in \mathbb{R}_+. \tag{3.24}$$

3 If we combine (3.18) and (3.22) and consider the separability of the Gaussian basis functions, we obtain the low-rank approximation to each entry of the tensor $\mathbf{P}$:

$$p_{\mathbf{i}} \approx \sum_{k=-M}^{M} g_k \int_{\mathbb{R}^3} \psi_{\mathbf{i}}(x) e^{-t_k^2\|x\|^2} dx = \sum_{k=-M}^{M} g_k \prod_{\ell=1}^{3} \int_{\mathbb{R}} \psi_{\mathbf{i}_\ell}^{\ell}(x_\ell) e^{-t_k^2 x_\ell^2} dx_\ell. \tag{3.25}$$

Let us recall that $g_k > 0$ and define the vector

$$\mathbf{p}_k^{(\ell)} = g_k^{1/3} [b_{i_\ell}^{(\ell)}(t_k)]_{i_\ell}^{n_\ell} \in \mathbb{R}^{n_\ell}, \quad b_{i_\ell}^{(\ell)}(t_k) = \int_{\mathbb{R}} \psi_{\mathbf{i}_\ell}^{\ell}(x_\ell) e^{-t_k^2 x_\ell^2} dx_\ell. \tag{3.26}$$

4 Then the 3rd order tensor $\mathbf{P}$ can be approximated by the R-term ($R = 2M + 1$) canonical representation,

$$\mathbf{P} \approx \mathbf{P}_R = \sum_{k=-M}^{M} g_k \bigotimes_{\ell=1}^{3} \mathbf{b}^{(\ell)}(t_k) = \sum_{k=-M}^{M} \mathbf{p}_k^{(1)} \otimes \mathbf{p}_k^{(2)} \otimes \mathbf{p}_k^{(3)} \in \mathbb{R}^{n \times n \times n}. \tag{3.27}$$

For some threshold $\epsilon > 0$, M can be chosen as the minimal number such that the following max-norm holds,

$$\|\mathbf{P} - \mathbf{P}_R\| \leq \epsilon \|\mathbf{P}\|. \tag{3.28}$$

Due to the symmetry of the Newton kernel, the sum in (3.27) reduces to $k = 0, \ldots, M$. From (3.23), we notice that the sequence of quadrature points $\{t_k\}$ can be split into two subsequences i.e., $\mathcal{T} := \{t_k | k = 0, \ldots, M\} = \mathcal{T}_l \cup \mathcal{T}_s$ with

$$\mathcal{T}_l := \{t_k | k = 0, 1, \ldots, R_l\}, \quad \text{and} \quad \mathcal{T}_s := \{t_k | k = R_l + 1, \ldots, M\}, \tag{3.29}$$

where $\mathcal{T}_l$ includes quadrature points t_k which are condensed "near" zero, thereby generating the smooth and global long-range Gaussians, the so-called low-pass filters [17]. On the other hand, $\mathcal{T}_s$ accumulates the increasing in $M \to \infty$ sequence of "large" sampling points t_k with the upper bound $C_0^2 \log^2(M)$ which corresponds to the localized short-range Gaussians, the so-called high-pass filters. The quasi-optimal choice of $C_0 \approx 3$ was determined in [19].

The additive decomposition of the canonical tensor $\mathbf{P}_R$ generated by the splitting in (3.29), i.e., $\mathbf{P}_R = \mathbf{P}_{R_s} + \mathbf{P}_{R_l}$, where

$$\mathbf{P}_{R_s} = \sum_{t_k \in \mathcal{T}_s} \mathbf{p}_k^{(1)} \otimes \mathbf{p}_k^{(2)} \otimes \mathbf{p}_k^{(3)}, \quad \mathbf{P}_{R_l} = \sum_{t_k \in \mathcal{T}_l} \mathbf{p}_k^{(1)} \otimes \mathbf{p}_k^{(2)} \otimes \mathbf{p}_k^{(3)}. \tag{3.30}$$

Remark 3.18:

The choice of the critical number $R_l = \#\mathcal{T}_l - 1$ (or equivalently, $R_s = \#\mathcal{T}_s = M - R_l$) that specifies the splitting $\mathcal{T} = \mathcal{T}_l \cup \mathcal{T}_s$ is determined by the active support of the short-range part of the total electrostatic potential, such that the functions $\mathbf{p}_k(x)$, $t_k \in \mathcal{T}_s$ vanish outside of the sphere B_σ of radius $\sigma > 0$ subject to a certain threshold $\delta > 0$ [17]. ♢

Following [17], for a given σ, we recall two basic criteria, corresponding to the max-norm and L^1-norm estimates which can be applied in order to obtain R_l. For the max-norm criterion, we have

$$\mathcal{T}_s = \left\{ t_k : g_k e^{-t_k^2 \sigma^2} \leq \delta \right\} \iff R_l = \min \left\{ k : g_k e^{-t_k^2 \sigma^2} \leq \delta \right\}. \tag{3.31}$$

On the other hand, the criterion corresponding to the L^1-norm estimate reads

$$\mathcal{T}_s = \left\{ t_k : g_k \int_{B_\sigma} e^{-t_k^2 x^2} dx \leq \delta \right\} \iff R_l = \min \left\{ k : g_k \int_{B_\sigma} e^{-t_k^2 x^2} \leq \delta \right\}. \tag{3.32}$$

The quantitative estimates of R_l are determined by invoking the explicit equation (3.23) for the quadrature parameters. For instance, in the case $C_0 = 3$ and $g(t) = 1$, the criterion in (3.31) implies that R_l solves the equation

$$\left(\frac{3 R_l \log M}{M} \right)^2 \sigma^2 = \log \left(\frac{\mathfrak{h}_M}{\delta} \right). \tag{3.33}$$

Remark 3.19:

Note that the criteria (3.31) and (3.32) may be modified and tailored to particular applications [17]. For instance, the parameter σ may be associated with the van der Waal radii (interatomic distance) in the molecular system of interest in electronic structure calculations. ♢

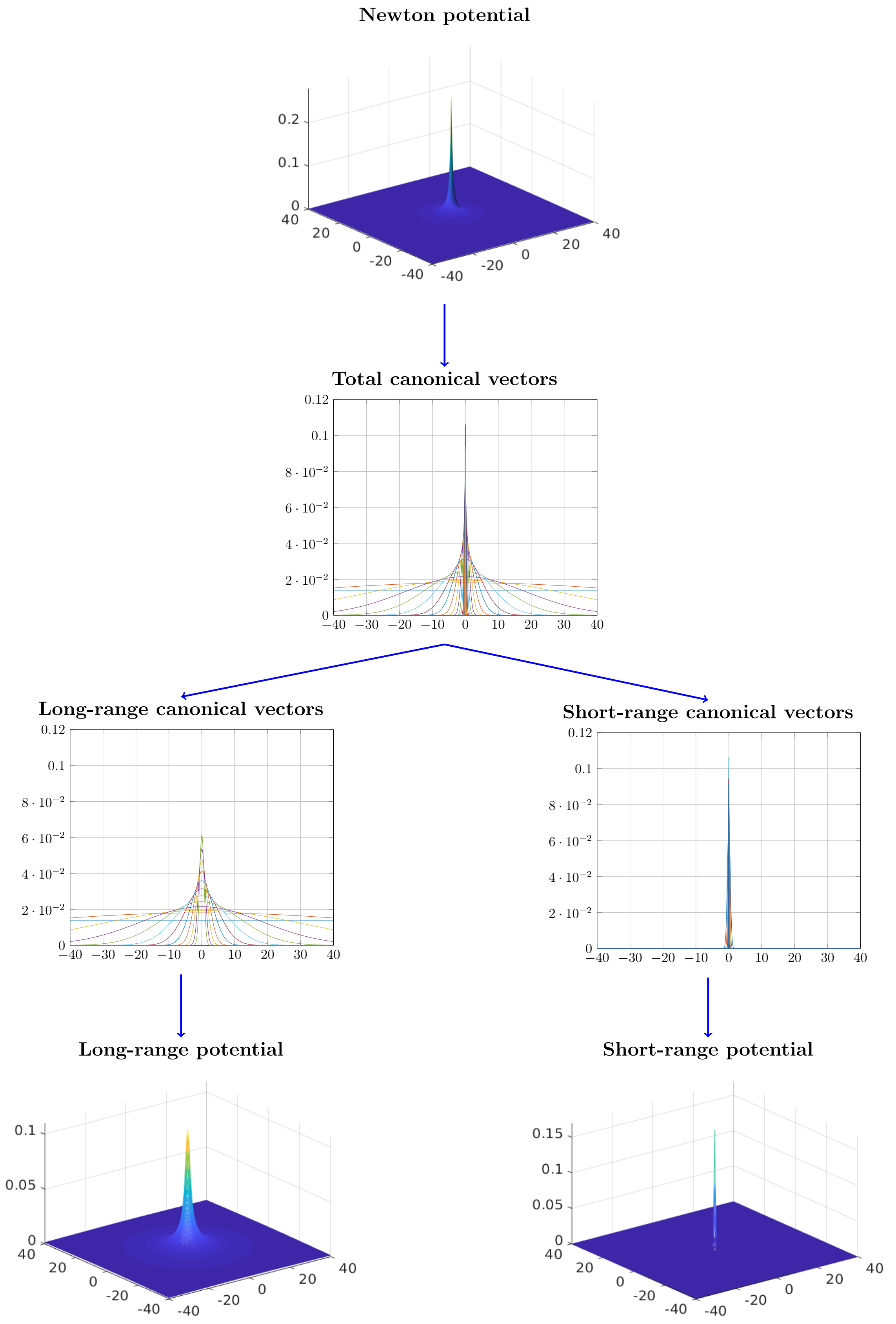
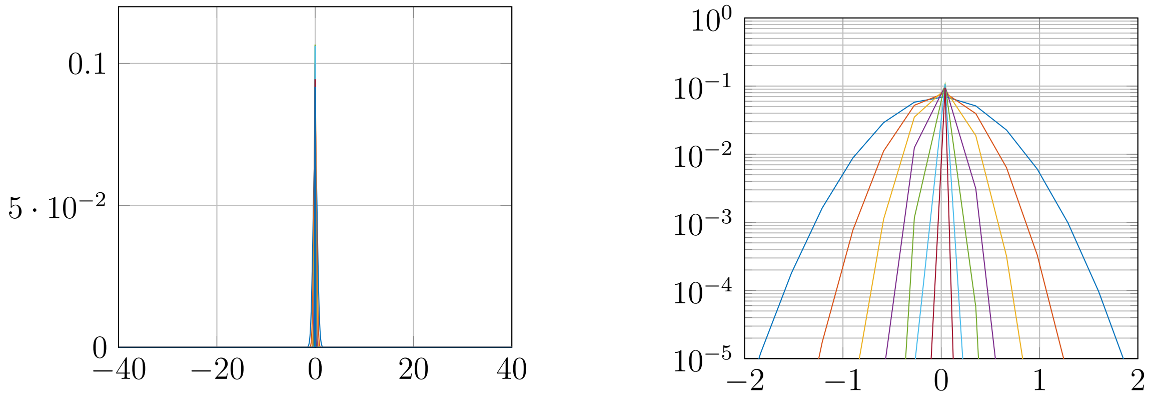


Figure 3.7.: The canonical tensor representation of the single Newton kernel $(1/\|\bar{x}\|)$ and its subsequent splitting into the long-and short-range components by the range-separated (RS) tensor format, and their corresponding transformations into the full tensors [17].



(a) Short-range vectors.

(b) Log plot for Figure 3.8a.

Figure 3.8.: The short-range vectors of the canonical tensor representation for $n = 1024$, $R = 20$, and $R_l = 12$ [17].

As demonstrated in [17], consider the splitting of the tensor $\mathbf{P}_R = \mathbf{P}_{R_s} + \mathbf{P}_{R_l}$ in (3.30) for the representation of the Newton kernel $\frac{1}{\|\bar{x}\|}$ on the $40 \times 40 \times 40$ domain size with the parameters $R = 20$, $R_l = 12$, and $R_s = 8$, respectively. Employing the criterion (3.31) and considering $\delta \approx 10^{-5}$, the effective support for this splitting is obtained by $\sigma = 2$ as shown in Figure 3.7 and Figure 3.8. Details can be found in [15, 17].

Remark 3.20:

Notice from Figure 3.8a, that the short-range vectors are located in an effective support of $\sigma \approx 2$Å. Outside this radius, intermediate vectors begin to scatter into the long-range regime, see Figure 3.7. A domain of $[-40, 40]^3$ was used to compute the results in Figure 3.7 and Figure 3.8. ◊

The aforementioned results are only valid for a single particle potential (or the Newton kernel, $1/\|\bar{x}\|$). In the case of a potential sum generated by a multiparticle system, for example, in (3.16), the two components in (3.30) are treated independently due to their differences as far as their effective supports are concerned [17]. The following is an overview of the RS canonical tensor representation of $P_N(\bar{x})$ in (3.16).

3.3.2. RS tensor splitting of multiparticle systems

Here, we provide an overview of the techniques for modeling of the long-range interaction potential in multiparticle systems of general type by the RS tensor format which were introduced in [17]. First, we introduce the $n \times n \times n$ grid Ω_n in $\Omega = [-b, b]^3$, as well as the auxiliary $2n \times 2n \times 2n$ grid $\widetilde{\Omega}_{2n}$, where $\widetilde{\Omega}_{2n} = 2\Omega$. We denote the canonical rank-R representation of $1/\|\bar{x}\|$ on $\widetilde{\Omega}_{2n}$, and its restriction onto Ω_n by $\widetilde{\mathbf{P}}_R$ and $\mathbf{P}_R$, respectively.

Then the optimal splitting (3.30) is applied to the reference canonical tensor $\mathbf{P}_R$ and to its accompanying version $\widetilde{\mathbf{P}}_R$, such that

$$\widetilde{\mathbf{P}}_R = \widetilde{\mathbf{P}}_{R_s} + \widetilde{\mathbf{P}}_{R_l} \in \mathbb{R}^{2n\times 2n\times 2n}. \tag{3.34}$$

To mitigate the complexity related to the linear growth of the ranks with respect to the number of particles because of the short-range components (which are non-reducible in both canonical and Tucker tensor formats), we treat the two components in (3.34) independently [17].

Therefore, we first consider the global tensor decomposition of only the long-range part $\mathbf{P}_{R_l} \in \mathbb{R}^{n^{\otimes 3}}$ which can be constructed by a direct sum of shift-and-windowing transforms, $\mathcal{W}_\nu$, of the reference tensor $\widetilde{\mathbf{P}}_{R_l}$. Interested readers are referred to [17, 81] for more details. We have

$$\mathbf{P}_l = \sum_{\nu=1}^{N_m} z_\nu\, \mathcal{W}_\nu(\widetilde{\mathbf{P}}_{R_l}) = \sum_{\nu=1}^{N_m} z_\nu\, \mathcal{W}_\nu\Big(\sum_{k\in\mathcal{K}_l} \widetilde{\mathbf{p}}_k^{(1)} \otimes \widetilde{\mathbf{p}}_k^{(2)} \otimes \widetilde{\mathbf{p}}_k^{(3)}\Big). \tag{3.35}$$

The shift-and-windowing transform $\mathcal{W}_\nu$ maps a reference tensor $\widetilde{\mathbf{P}}_{R_l} \in \mathbb{R}^{2n\times 2n\times 2n}$ onto its sub-tensor of smaller size $n^{\otimes 3}$, obtained by first shifting the center of $\widetilde{\mathbf{P}}_{R_l}$ to the grid-point $\bar{x}_\nu$, and then restricting (windowing) the result onto the computational grid Ω_n, see [81]. The particle charges are denoted by z_ν. The canonical rank of the tensor sum $\mathbf{P}_l$, of rank RN_m, was proven in [17] to depend only logarithmically on the number of particles N_m involved in the summation. Therefore, it is expected that the large initial rank of $\mathbf{P}_l$ to be significantly reduced to some value $R_* \ll RN_m$, which is almost independent of N_m.

Remark 3.21:

It is worth noting that for large biomolecules, the rank RN_m and the $n^{\otimes 3}$ Cartesian grid can be very large due to large N_m. In such cases, the canonical-to-Tucker (C2T) and the Tucker-to-canonical (T2C) transforms can be applied in order to obtain a low rank canonical tensor representation which accurately approximates the original tensor. The C2T transform employs the reduced higher order singular value decomposition (RHOSVD) to accomplish the rank reduction process [82]. ◊

On the other hand, the short-range component of the total potential is represented in the RS format, by a single small size tensor, referred to as the cumulated canonical tensors (CCT), which is supplemented by a list of the particles' coordinates. Before we define the CCT, we first define the concept of well-separability of the point set, which controls the trade-off between the numerical efficiency and the approximation quality of the short-range component in the collective potential sum. Interested readers are referred to [17] for detailed information.

Definition 3.22 (Well-separable point distribution, see Definition 3.2 in [17].): Given a constant $\sigma_* > 0$, a set $S = \{\bar{x}_\nu\}$ of points in $\mathbb{R}^d$ is called σ_*-separable if

$$d(\bar{x}_\nu, \bar{x}_{\nu'}) := \|\bar{x}_\nu - \bar{x}_{\nu'}\| \geq \sigma_* \quad \text{for all} \quad \nu \neq \nu'. \tag{3.36}$$

◊

The definition in (3.36) holds for non-bonded atoms, for instance in protein and crystals, where the van der Waal's radii are well defined. The CCT relies on this property in order to provide an accurate approximation of the short-range potential sum.

Definition 3.23 (CCT, see Definition 3.4 in [17]): Given a set of sources $\mathcal{J} = \{\mathbf{j}^{(\nu)} := (j_1^{(\nu)}, j_2^{(\nu)}, \ldots, j_d^{(\nu)})\}$, $j_\ell^{(\nu)} \in I_\ell$, and $\sigma_* > 0$, choose the width parameter $\gamma \in \mathbb{N}$ from the relation $\gamma h \leq \sigma_*$ s.t. $\mathcal{J}_\gamma^{(\nu)} \cap \mathcal{J}_\gamma^{(\nu')} = \varnothing$, $\nu \neq \nu'$. A rank-R_0 CCT tensor $\mathbf{P}_s \in \mathbb{R}^I$, associated with $\mathcal{J}$ and γ, is defined as

$$\mathbf{P}_s = \sum_{\nu=1}^{N_m} z_\nu \mathbf{U}_\nu \quad \text{with} \quad \operatorname{rank}(\mathbf{U}_\nu) \leq R_0, \tag{3.37}$$

where canonical tensors $\mathbf{U}_\nu = [u_j]$ are vanishing beyond the γ-vicinity of $\mathbf{j}^{(\nu)}$: $u_{\mathbf{j}} = 0$ for $\mathbf{j} \subset \mathcal{I} \setminus \mathcal{J}_\gamma^{(\nu)}$, $\nu = 1, \ldots, N_m$. ◊

Definition 3.24 (Uniform CCT, see Definition 3.7 in [17]): We refer to a CCT tensor in (3.37) as uniform if all components $\mathbf{U}_\nu$ are generated by a single rank-R_0 tensor $\mathbf{U}_0 = \sum_{m=1}^{R_0} \mu_m \hat{\mathbf{u}}_m^{(1)} \otimes \cdots \otimes \hat{\mathbf{u}}_m^{(d)}$ s.t. $\mathbf{U}_\nu|_{\mathcal{J}_\delta^{(\nu)}} = \mathbf{U}_0$. ◊

For the given separation parameter $\gamma \in \mathbb{N}$, which is used to partition the canonical vectors into the short-and long-range parts, the RS-canonical tensor format specifies the class of d-tensors $\mathbf{A} \in \mathbb{R}^{n_1 \times \cdots \times n_d}$, which can be represented as a sum of a rank-R canonical tensor $\mathbf{P}_l = \sum_{k=1}^{R} \xi_k \mathbf{u}_k^{(1)} \otimes \cdots \otimes \mathbf{u}_k^{(d)} \in \mathbb{R}^{n_1 \times \ldots \times n_d}$ and a uniform CCT $\widehat{\mathbf{P}}_s = \sum_{\nu=1}^{N_m} z_\nu \mathbf{U}_\nu$ generated by $\mathbf{U}_0$ with $\operatorname{rank}(\mathbf{U}_0) \leq R_0$,

$$\mathbf{A} = \sum_{k=1}^{R} \xi_k \mathbf{u}_k^{(1)} \otimes \cdots \otimes \mathbf{u}_k^{(d)} + \sum_{\nu=1}^{N_m} z_\nu \mathbf{U}_\nu, \tag{3.38}$$

where $\operatorname{diam}(\operatorname{supp}\mathbf{U}_\nu) \leq 2\gamma_i$. Here, γ_i is the atomic radius of each atom in the biomolecule.

Remark 3.25:

Notice that for biomolecules whose atoms have varying radii, we adjust the computation of the short- and long-range range electrostatic potential accordingly by assigning the corresponding vectors from Figure 3.7 and Figure 3.8 to atomic clusters of similar radii [99]. ♢

3.4. Conclusions

In this chapter, we have briefly discussed the fundamentals of multilinear algebra followed by an extensive discussion of the novel RS tensor format, which was then applied to the splitting of both the single Newton kernel and to the sum of Newton kernels arising from multiparticle potential systems. The efficient construction of a regularization scheme for the PBE by the RS tensor format is discussed in Chapter 4.

CHAPTER 4

REGULARIZATION OF THE POISSON-BOLTZMANN EQUATION USING THE RANGE-SEPARATED TENSOR FORMAT

Contents

4.1. Introduction

This chapter provides the main contribution of this thesis to the theory of the Poisson-Boltzmann equation (PBE), as far as solution decomposition techniques are concerned, by efficiently constructing a regularization scheme for the PBE in order to obtain a regularized PBE (RPBE) model which only solves for the long-range electrostatic potential, thereby avoiding the numerical approximation of the singular sources, which increases the accuracy. This is accomplished by the splitting of the Dirac delta distribution and the substitution of the singular sources by a smooth and regularized function. We begin the chapter by first applying the range-separated tensor format from Chapter 3 in order to construct an efficient regularization scheme for the Poisson equation in Section 4.2 as a proof of concept. In Section 4.3, we extend the RS tensor based regularization scheme to both the linear and the nonlinear PBE and provide existence and uniqueness of solutions for the nonlinear PBE. We wind up the chapter by discussing the profound advantage of the RS tensor format for the efficient and accurate computation of electrostatic energies and forces in Section 4.4.

4.1.1. Application of RS tensor format for solving the PBE

The RS tensor format can be gainfully applied in computational problems which include functions with multiple local singularities or cusps, Green kernels with essentially non-local behavior, as well as in various approximation problems treated by means of radial basis functions. In what follows, we describe the new approach for the construction of computationally effective boundary/interface conditions and right-hand sides in the Poisson-Boltzmann equation (PBE) describing the electrostatic potential of a biomolecule in gas phase and in solvent by solving the FEM/FDM discretization of the regularized PBE. This approach is based on the use of the RS decomposition of the Dirac delta distribution which was introduced in [85]. The main advantage of our approach is due to complete avoidance of the direct FEM/FDM approximation (interpolation) of the highly singular right-hand sides in the traditional formulation of the PBE and, at the same time, preventing the modification of the stiffness matrix and/or the continuity conditions across the interface.

4.2. The RS tensor based splitting scheme for the Poisson equation (PE)

In this section, we present the new splitting scheme which is based on the range separated representation of the Dirac δ-distribution [85], which comprises of the highly singular right-hand side in the target PBE (2.15). Following [18], we consider the PE as proof of concept and validate the numerical results in Chapter 6. The idea is to modify the right-hand side ρ_f in such a way that the short-range part in the solution u can be pre-computed independently by the direct tensor decomposition of the free space potential, and the initial elliptic equation applies only to the long-range part of the total potential. The latter is a smooth function, hence the FDM/FEM approximation error can be reduced dramatically even on relatively coarse grids in 3D.

To fix the idea, we consider the simplest case of the single atom with unit charge located at the origin, such that the exact electrostatic potential reads $u(\bar{x}) = \frac{1}{\|\bar{x}\|}$, $\bar{x} \in \mathbb{R}^3$. Recall that the Newton kernel discretized by the R-term sum of Gaussian type functions living on the $n \times n \times n$ tensor grid Ω_n is represented by a sum of short- and long-range tensors,

$$\frac{1}{\|\bar{x}\|} \rightsquigarrow \mathbf{P}_R = \mathbf{P}_{R_s} + \mathbf{P}_{R_l} \in \mathbb{R}^{n\times n\times n},$$

where by definition

$$\mathbf{P}_{R_s} = \sum_{k=R_l+1}^{R} \mathbf{p}_k^{(1)} \otimes \mathbf{p}_k^{(2)} \otimes \mathbf{p}_k^{(3)}, \quad \mathbf{P}_{R_l} = \sum_{k=1}^{R_l} \mathbf{p}_k^{(1)} \otimes \mathbf{p}_k^{(2)} \otimes \mathbf{p}_k^{(3)}. \tag{4.1}$$

Let us formally discretize the exact equation for the Newton potential, $u(\bar{x}) = \frac{1}{\|\bar{x}\|}$,

$$-\Delta \frac{1}{\|\bar{x}\|} = 4\pi\, \delta(\bar{x}),$$

by using the FDM/FEM Laplacian matrix A_Δ instead of Δ and via substitution of the canonical tensor decomposition $\mathbf{P}_R$ instead of $u(\bar{x}) = \frac{1}{\|\bar{x}\|}$. This leads to the grid representation of the discretized Dirac delta [85]

$$\delta(\bar{x}) \rightsquigarrow \boldsymbol{\delta}_h := -\frac{1}{4\pi} A_\Delta \mathbf{P}_R,$$

that will be applied in the framework of our RS tensor based discretization scheme.

We remind that the 3D finite difference Laplacian matrix A_Δ, defined on the uniform rectangular grid, takes the form

$$A_\Delta = \Delta_1 \otimes I_2 \otimes I_3 + I_1 \otimes \Delta_2 \otimes I_3 + I_1 \otimes I_2 \otimes \Delta_3, \tag{4.2}$$

where $-\Delta_\ell = h_\ell^{-2}\text{tridiag}\{1, -2, 1\} \in \mathbb{R}^{n_\ell \times n_\ell}$, $\ell = 1, 2, 3$, denotes the discrete univariate Laplacian, such that the Kronecker rank of A_Δ equals to 3. Here I_ℓ, $\ell = 1, 2, 3$, is the identity matrix in the corresponding single dimension.

Now we are in the position to describe the RS tensor-based splitting scheme. To that end, we use the splitting of the discretized δ-distribution into short- and long-range components in the form [18, 85],

$$\boldsymbol{\delta}_h = \boldsymbol{\delta}_s + \boldsymbol{\delta}_l, \tag{4.3}$$

where

$$\boldsymbol{\delta}_s := -\frac{1}{4\pi} A_\Delta \mathbf{P}_{R_s}, \quad \text{and} \quad \boldsymbol{\delta}_l := -\frac{1}{4\pi} A_\Delta \mathbf{P}_{R_l}. \tag{4.4}$$

We observe that by construction, the short range part vanishes on the interface Γ, hence it satisfies the discrete Poisson equation in Ω_m with the respective right-hand side in the form $\boldsymbol{\delta}_s$ and zero boundary conditions on Γ. Then we deduce that this equation can be subtracted from the full discrete linear system, such that the long-range component of the solution, $\mathbf{P}_{R_l}$, will satisfy the same linear system of equations (same interface conditions for the coefficients), but with a modified right-hand side corresponding to the weighted sum of the long-range tensors $\boldsymbol{\delta}_l$ only. In the simple example of the single charge, we arrive at the particular discrete Poisson equation for the long-range part in the full potential $\mathbf{P}_R$, $\mathbf{U}_l = \mathbf{P}_{R_l}$,

$$-A_\Delta \mathbf{U}_l = \boldsymbol{\delta}_l, \tag{4.5}$$

which can be solved by an appropriate iterative method.

Figure 4.1 illustrates the modified right hand side representing the long-range part of the discrete Dirac delta $\boldsymbol{\delta}_l$. It it worth noting that the FEM approximation theory could be applied to this formulation since the input data (i.e., the right-hand side) are regular enough. However, it is not the case for the initial formulation with the highly singular Dirac delta distribution in the right-hand side.

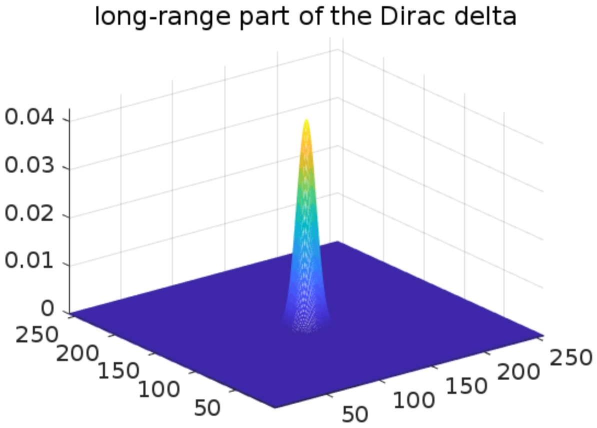


Figure 4.1.: The long-range part of the Dirac delta $\boldsymbol{\delta}_l$ on an $n^{\otimes 3}$ 3D grid, $n = 256$.

This scheme can be easily extended to the case of many-atomic systems just by additive representation of the short- and long-range parts in the total free space potential,

$$-A_\Delta \mathbf{P}_{R_l} = \boldsymbol{\delta}_{R_L}, \tag{4.6}$$

where we suppose that R_L is the rank of the long-range part $\mathbf{P}_{R_l}$ of the corresponding RS tensor of type (3.38), and $\boldsymbol{\delta}_{R_L}$ is calculated as shown in (4.8).

4.2.1. Benefits of the RS tensor based regularization for the PE

We summarize the following benefits of the RS tensor based solution decomposition scheme for the PE.

- Most important is that due to efficient splitting of the short- and long-range parts in the target tensor representing both the single Newton kernel and the total free-space potential, the singular component (short-range part) does not contribute to the jump condition at the interface.

- A remarkable advantage is that the long-range part in the RS tensor decomposition of the Dirac-delta distribution [85] vanishes at the interface and, hence, the modified right-hand side generated by this long-range component remains localized in the ”linear” solute region.

- The boundary conditions are obtained from the long-range part in the tensor representation of the collective electrostatic potential which reduces the computational costs involved, in contrast to solving the equation with some analytical function used to define the boundary condition.

- Only a single system of algebraic equations is solved for the smooth long-range (i.e., regularized) part of the collective potential discretized with controllable precision on a relatively coarse grid, which is then added to the directly precomputed (avoiding PDE solutions) low-rank tensor representation for the short-range contribution.

Notice that the MATLAB representation of the matrix A_Δ (say, the FD matrix) can be easily described in terms of kron operations as follows

$$A_\Delta = \mathrm{kron}(\mathrm{kron}(\Delta_1, I), I) + \mathrm{kron}(\mathrm{kron}(I, \Delta_1, I) + \mathrm{kron}(\mathrm{kron}(I, I), \Delta_1), \tag{4.7}$$

applied to a long vector of size n^3 representing the Newton potential.

Then the rank-structured calculation of the "collective" right-hand side $\boldsymbol{\delta}_{R_L}$ in (4.6) is reduced to one-dimensional operations,

$$-\boldsymbol{\delta}_{R_L} = \sum_{k=1}^{R_L} \xi_k \Delta_1 \mathbf{a}_k^{(1)} \otimes \mathbf{a}_k^{(2)} \otimes \mathbf{a}_k^{(3)} + \sum_{k=1}^{R_L} \xi_k \mathbf{a}_k^{(1)} \otimes \Delta_1 \mathbf{a}_k^{(2)} \otimes \mathbf{a}_k^{(3)} + \sum_{k=1}^{R_L} \xi_k \mathbf{a}_k^{(1)} \otimes \mathbf{a}_k^{(2)} \otimes \Delta_1 \mathbf{a}_k^{(3)}, \tag{4.8}$$

where $\mathbf{a}_k^{(1)}$ are the canonical vectors and R_L is the canonical rank of the long-range part of the collective electrostatic free space potential of a biomolecule computed in the RS tensor format (3.38).

This is the tensor ansatz to be used as the right-hand side in the equation (4.5), which we apply in numerical experiments. With a subsequent usage of the canonical-to-full tensor transform and after reshaping a three-tensor into a long vector, $\boldsymbol{\delta}_{R_L}$ is applied in a standard PBE iterative solver as the RHS for the long-range part.

Remark 4.1:

Another advantage of our scheme is that the short-range part of the solution in the PBE (2.15) is obtained for free, since it is merely incorporated as the set of short-range parts of the respective Newton potentials for every particle in a biomolecule. That corresponds to a set of tensors in the second term of the collective electrostatic potential in the RS tensor format (3.38). ◊

4.3. The RS tensor based regularization scheme for the PBE

In this section, we extend the new regularization scheme based on the RS tensor format to both the linearized and nonlinear PBE, which are characterized by the presence of inhomogeneous (or jumping) coefficients. The idea is to modify the right-hand side of (2.15), in such a way that the short-range part in the solution u can be pre-computed independently by the direct tensor decomposition of the free space potential, and the initial elliptic equation (or the RPBE) applies only to the long-range component of the total potential. The latter is a smooth function, hence the FDM/FEM approximation error can be reduced dramatically even on relatively coarse grids in 3D.

To fix the idea, we first recall the weighted sum of interaction potentials in a large N-particle system as in (3.16) generated by the Newton kernel, $1/\|\bar{x}\|$, at each charge location $\bar{x}_i$, $\bar{x} \in \mathbb{R}^3$, i.e.,

$$P_N(\bar{x}) = \sum_{i=1}^{N_m} \frac{q_i}{\epsilon_m} \frac{1}{\|\bar{x} - \bar{x}_i\|}, \tag{4.9}$$

We also recall that the sum of Newton kernels for a multiparticle system discretized by the R-term sum of Gaussian type functions living on the $n^{\otimes 3}$ tensor grid Ω_n is represented by a sum of long-range tensors in (3.35) and a CCT tensor in (3.37), respectively.

We now have sufficient information to facilitate the construction of the NRPBE based on the simple splitting of the Dirac delta distribution [85]. To fix the idea, from Remark 3.16, the weighted sum of interaction potentials in a large N_m-particle system as in (3.16) is also the analytical solution to the Poisson equation (PE), i.e.,

$$-\epsilon_m \Delta P_N(\bar{x}) = \sum_{i=1}^{N_m} q_i \delta(\bar{x} - \bar{x}_i) \quad \text{in } \mathbb{R}^3. \tag{4.10}$$

Consider the RS tensor splitting of the multiparticle Newton potential into a sum of long-range tensors $\mathbf{P}_l$ in (3.35) and a CCT tensor $\mathbf{P}_s$ in (3.37), i.e.,

$$\mathbf{P}_N(\bar{x}) = \mathbf{P}_s(\bar{x}) + \mathbf{P}_l(\bar{x}). \tag{4.11}$$

Substituting each of the components of (4.11) into the discretized Poisson equation, we derive the respective components of the molecular charge density (or Dirac-delta distributions) as follows

$$f^s := -A_\Delta \mathbf{P}_s, \quad \text{and} \quad f^l := -A_\Delta \mathbf{P}_l, \tag{4.12}$$

where A_Δ is the 3D finite difference Laplacian matrix defined in (4.2).

Figure 4.2 depicts the behaviour of the modified representations of both the smooth and singular components of the Dirac delta distributions using the formula in (4.12). The charge density data is obtained from protein *fasciculin 1*, an anti-acetylcholinesterase toxin from green mamba snake venom [104]. Notice from the highlighted data cursors, that both the functions are localized within the molecular region, with values dropping to zero outside this region. Furthermore, the Figure 4.2a represents the function in (4.12) which we utilize to derive a regularized PBE model (RPBE) in the next step.

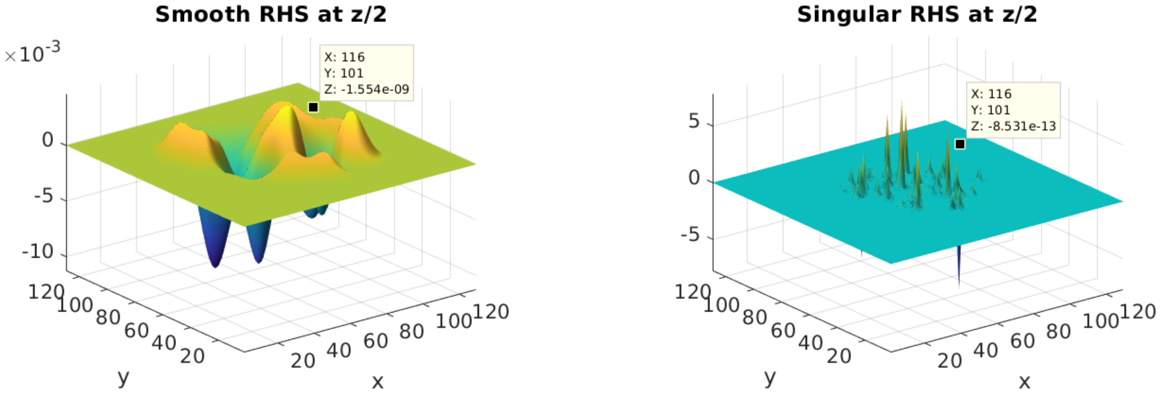
The nonlinear regularized PBE (NRPBE) can now be derived as follows [99]. First, the unknown solution (or target electrostatic potential) u to the PBE (2.15) can be decomposed as $u = u^s + u^r$, where u^s is the known singular function (or short-range component) and u^l is the unknown long-range component to be determined. Therefore, the PBE (2.15) can be rewritten as

$$\left.\begin{aligned} -\nabla \cdot (\epsilon \nabla (u^s + u^r)) + \bar{\kappa}^2 \sinh(u^s + u^r) &= f^s + f^l \qquad &&\text{in } \mathbb{R}^3, \\ u^s + u^r &= g, &&\text{on } \partial\Omega, \end{aligned}\right\} \tag{4.13}$$

where g is the Dirichlet boundary conditions defined in (2.36).

It was proved in [85] and demonstrated in [18] that the function f^s and the corresponding short-range potential u^s are localized within the molecular region Ω_m and vanishes on the interface Γ. Moreover, in the PBE (2.15), the function $\bar{\kappa}$ is piecewise constant as defined in (2.18), and $\bar{\kappa} = 0$ in Ω_m. Therefore, we can rewrite the Boltzmann distribution term in (4.13) as

$$\bar{\kappa}^2 \sinh(u^s + u^r) = \bar{\kappa}^2 \sinh(u^r), \quad \text{because} \quad u^s = 0 \quad \text{in } \Omega_s. \tag{4.14}$$



(a) Long-range part of charge density.

(b) Short-range part of charge density.

Figure 4.2.: The long- and short-range parts of the charge density for protein Fasciculin on $129^{\otimes 3}$ grid.

Consequently, following the splitting of the Dirac-delta distributions in (4.12), the short-range component of the potential satisfies the Poisson equation, i.e.,

$$-\nabla \cdot (\epsilon_m \nabla u^s) = f^s \quad \text{in } \mathbb{R}^3. \tag{4.15}$$

Subtracting (4.15) from (4.13) and substituting (4.14), we obtain the nonlinear regularized PBE (NRPBE) as follows

$$-\nabla \cdot (\epsilon \nabla u^r(\bar{x})) + \bar{\kappa}^2(\bar{x}) \sinh(u^r(\bar{x})) = f^l, \quad \text{in } \Omega, \tag{4.16}$$

subject to the Dirichlet boundary conditions in (2.36). The total solution to the NRPBE is therefore, obtained by $u(\bar{x}) = u^s(\bar{x}) + u^r(\bar{x})$.

We conclude with the following result.

Lemma 4.2:

Let the short- and long-range components of the electrostatic potential of the PBE (2.15) be given by u^s and u^r, respectively. Then the following statements are true

(a) $u^s = \mathbf{P}_s$.

(b) $u^r \neq \mathbf{P}_l$, but u^r satisfies the NRPBE in (4.16). ◊

Proof. (a) Recall the decomposed PBE (4.13) and the NRPBE (4.16). Subtracting the two equations and applying (4.14), we obtain the PE model for the short-range potential in (4.15). Substituting the first equation of (4.12) into the right-hand side of (4.15), yields

$$-\nabla \cdot (\epsilon_m \nabla u^s) = -A_\Delta \mathbf{P}_s, \tag{4.17}$$

where $\mathbf{P}_s$ is the CCT tensor in (3.37), which represents the precomputed short-range potential sum. We conclude that $u^s = \mathbf{P}_s$, which completes the proof of (a).

(b) Here, we subtract the PE (4.15) from the decomposed PBE (4.13) to obtain the NRPBE (4.16). Substituting the second equation of (4.12) into the right-hand side of (4.16) yields

$$-\nabla \cdot (\epsilon \nabla u^r) + \bar{\kappa}^2 \sinh(u^r) = -A_{\Delta}\mathbf{P}_l, \tag{4.18}$$

which implies that $u^r \neq \mathbf{P}_l$, thereby completing the proof of (b). □

The total solution to the NRPBE is therefore obtained by $u(\bar{x}) = u^s(\bar{x}) + u^r(\bar{x})$.

The LPBE in (2.21) can also be regularized in a similar way as in (4.16) to obtain the following LRPBE

$$-\nabla \cdot (\epsilon \nabla u^r(\bar{x})) + \bar{\kappa}^2(\bar{x}) u^r(\bar{x}) = f^l, \quad \text{in } \Omega, \tag{4.19}$$

subject to the Dirichlet boundary conditions in (2.36).

4.3.1. Existence and uniqueness of solutions to the NRPBE

We here briefly discuss the existence and uniqueness of the solution to the nonlinear regularized (continuous) PBE (NRPBE) for monovalent electrolytes. This is an extension of the results in [30] wherein a different regularization scheme was applied to the NPBE (see (2.48) in Section 2.6). We begin by considering the weak formulation of the NRPBE via integration by parts, on the assumption that f^l is far away from the boundary $\partial\Omega$ (i.e., f^l is localized within the molecular region), see Figure 4.2. Hence, we seek to find

$$u \in M := \{z \in H^1(\Omega) \quad \text{s.t.} \quad e^z, e^{-z} \in L^2(\Omega_s), \quad \text{and} \quad z = g \text{ on } \partial\Omega\}$$

such that

$$A(u, v) + (B(u), v) + \langle f^l, v\rangle = 0, \qquad \forall v \ \in H^1(\Omega), \tag{4.20}$$

where

- $A(u, v) = (\epsilon \nabla u, \nabla v)$,
- $(B(u), v) = (\bar{\kappa}^2 \sinh(u), v)$, and
- $\langle f^l, v\rangle = \int_\Omega f^l v dV$.

Next, we now define the electrostatic free energy functional on M [30]:

$$E[\phi] = \int_\Omega \left[\frac{\epsilon}{2}|\nabla\phi|^2 + \langle f^l, \phi\rangle + \bar{\kappa}^2 \cosh(\phi)\right] dV, \qquad \phi \in M. \tag{4.21}$$

The following result establishes the existence and uniqueness of solutions to the NRPBE.

Lemma 4.3 (NRPBE: Existence and uniqueness, [30]):

There exists a unique solution $u \in M \subset H^1(\Omega)$ which minimizes the electrostatic free energy functional E, that is,

$$E(u) = \inf_{\phi \in M} E[\phi].$$

◊

Proof. From (4.21), it is easy to notice that $E(\phi)$ is differentiable in M, i.e.,

$$\langle DE(u), v\rangle = A(u, v) + (B(u), v) + \langle f^l, v\rangle.$$

In order to prove the existence of the minimizer, we need only to verify the following:

(a) M is a convex set,

(b) E is convex on M, and

(c) $\lim\limits_{\|v\|_1 \to \infty} E(v) = \infty$.

Notice that condition (a) is easy to verify because $M \subset H^1(\Omega)$, and the Sobolev space containing $L^2(\Omega)$ is well-known to be convex [39]. Therefore, M is convex. The verification of condition (b) inherits the convexity of the functions x^2 and $\cosh(x)$. In fact, E was proved in Theorem 2.11 to be strictly convex. Finally, the condition (c) is a consequence of the Poicaré inquality:

$$E(v) \geq C(\epsilon, \bar{\kappa})\|v\|_1^2 + C(f^l, g), \tag{4.22}$$

which we can prove as following. Invoking the Young's inquality, we have for any $\delta > 0$

$$\langle f^l, v\rangle \leq \epsilon_s \|f^l\|_{\Omega_s} \|v\|_{\Omega_s} \leq \frac{1}{\delta}\|f^l\|_{\Omega_s}^2 + \delta\epsilon_s^2 \|v\|_{\Omega_s}^2.$$

However, since $\cosh(x) \geq 0$, we have that

$$E(v) \geq C(\epsilon, \bar{\kappa})\|v\|^2 - \frac{1}{\delta}\|f^l\|_{\Omega_s}^2,$$

where we can guarantee that $C(\epsilon, \bar{\kappa}) > 0$ for sufficiently small δ. Then, we apply norm equivalence on M to obtain (4.22). Finally, we obtain the uniqueness of the minimizer from the strict convexity property of E. ☐

4.3.2. The FDM scheme for the regularized nonlinear PBE

Consider the uniform 3D $n^{\otimes 3}$ rectangular grid in $\Omega = [-b, b]^3$ with the mesh parameters $dx, dy, dz < 0.5$. The finite difference discretization of the RPBE in (4.16) leads to the algebraic system of equations [99]

$$\begin{aligned}
&-\frac{1}{dx^2}\epsilon^x_{i+\frac{1}{2},j,k}(u^r_{i+1,j,k} - u^r_{i,j,k}) + \frac{1}{dx^2}\epsilon^x_{i-\frac{1}{2},j,k}(u^r_{i,j,k} - u^r_{i-1,j,k}) - \frac{1}{dy^2}\epsilon^y_{i,j+\frac{1}{2},k}(u^r_{i,j+1,k} - u^r_{i,j,k}) \\
&+ \frac{1}{dy^2}\epsilon^y_{i,j-\frac{1}{2},k}(u^r_{i,j,k} - u^r_{i,j-1,k}) - \frac{1}{dz^2}\epsilon^z_{i,j,k+\frac{1}{2}}(u^r_{i,j,k+1} - u^r_{i,j,k}) + \frac{1}{dz^2}\epsilon^z_{i,j,k-\frac{1}{2}}(u^r_{i,j,k} - u^r_{i,j,k-1}) \\
&+ \bar{\kappa}^2_{i,j,k}\sinh(u^r_{i,j,k}) = f_r,
\end{aligned} \tag{4.23}$$

where the dielectric coefficient ϵ and kappa function $\bar{\kappa}$ are assigned according to the definition of the solvent accessible surface which is calculated by the Connolly approach [35, 140].

The coefficient matrix of the NRPBE is a sum of two matrices. The first matrix, A_1, emanates from the Laplacian operator in (4.23) and is assembled from ϵ, resulting into a $N_x \times N_y \times N_z$ by $N_x \times N_y \times N_z$ seven banded symmetric positive definite matrix containing the following nonzero elements. The main diagonal elements are given by

$$d_0(p) = \left[\frac{1}{dx^2}\epsilon^x_{i+\frac{1}{2},j,k} + \frac{1}{dx^2}\epsilon^x_{i-\frac{1}{2},j,k} + \frac{1}{dy^2}\epsilon^y_{i,j+\frac{1}{2},k} + \frac{1}{dy^2}\epsilon^y_{i,j-\frac{1}{2},k} + \frac{1}{dz^2}\epsilon^z_{i,j,k+\frac{1}{2}} + \frac{1}{dz^2}\epsilon^z_{i,j,k-\frac{1}{2}}\right], \tag{4.24}$$

where $p = (k-1)(N_x)(N_y)+(j-1)(N_x)+i$ is a natural ordering in which $i = 1, \ldots, N_x$, $j = 1, \ldots, N_y$ and $k = 1, \ldots, N_z$. The next lower band diagonal which is shifted in one column from the first column is given by

$$d_{-1}(p) = \left[-\frac{1}{dx^2}\epsilon^x_{i-\frac{1}{2},j,k}\right]. \tag{4.25}$$

The second lower band diagonal which is shifted N_x columns from the first column

$$d_{-2}(p) = \left[-\frac{1}{dy^2}\epsilon^y_{i,j-\frac{1}{2},k}\right]. \tag{4.26}$$

The third lower band diagonal which is shifted $N_x \times N_y$ columns from the first column

$$d_{-3}(p) = \left[-\frac{1}{dz^2}\epsilon^z_{i,j,k-\frac{1}{2}}\right]. \tag{4.27}$$

The remaining elements of the sparse lower triangular matrix A are set to zero. The upper triangular diagonal elements are obtained by symmetry.

The second matrix, A_2, is diagonal, and is generated from the Boltzmann distribution term. The resultant nonlinear system of equations in matrix form is given by

$$A_1 u^r + A_2(u^r) = f^l + b_c, \tag{4.28}$$

where $A_2(u^r) = \bar{\kappa}^2(\bar{x})\sinh(u^r(\bar{x}))$ is nonlinear, b_c is the vector of boundary conditions, and u^r is the regularized potential to be computed.

Notice that the system (4.28) can also be solved using several other methods, for example, the nonlinear relaxation methods [127], the nonlinear multigrid (MG) method [115], and the nonlinear conjugate gradient (CG) method [109]. However, when the aforementioned methods were attempted to solve the FDM version of the nonlinear PBE, the following observations were made: the convergence of the relaxation methods cannot be guaranteed, the MG method may diverge on certain applications, and due to the substantial evaluations of the exponential function, the CG method is quite slow. These drawbacks, are perhaps the impetus for the widespread popularity of the inexact Newton methods besides its proven convergence capabilities [121].

However, in this thesis, we employ a more efficient iterative approach of solving the RPBE in (4.16), in which the RPBE is first linearized and then dicretized. See Section 5.6 for more details.

4.3.3. Discussion of the computational scheme

The main computational tasks for the RS tensor based scheme for regularizing the PBE are summarized in Algorithm 4.1.

Algorithm 4.1: RS tensor format for constructing the NRPBE, see also Algorithms RSDD and LRPBE in [18].

```
Data: PQR[1] file and computational domain $\Omega = [-b, b]^3$.
Result: Long-range potential, $\mathbf{P}_l$ in (3.35), the short-range potential $\mathbf{P}_s$ in
        (3.37) and the regularized Dirac delta distribution $f^l$ in (4.12).
Extract the atomic positions, charges and radii from the PQR file.
Expunge the atoms with zero radii.
Sort the atoms with similar atomic radii.
Generate the canonical tensors via the sinc quadrature representation.
Compute separately via shifting and windowing transform, the long-and
  short-range canonical vectors [81].
Reduce the rank of the long-range component by the RHOSVD.
Compute the regularized Dirac delta distribution from the long-range
  component.
Convert both the Dirac delta distribution and the short-range canonical parts
  to full tensors.
Substitute $f^l$ into the right-hand side of the NPBE (2.15) to obtain the NRPBE
  (4.16) subject to the boundary condition in (2.36).
Discretize the RPBE (4.16) to obtain the following nonlinear system of
  equations
      $A(u^r) = f^l + b_c,$
  which can be solved by any nonlinear system solver.
Obtain the final NRPBE solution $u$ by the sum
      $u = u^r + u^s,$
  where $u^s = \mathbf{P}_{R_s}$ is the precomputed short-range component of the Newton
  potential sum.
```

The RS splitting scheme allows to reduce the initial equation for the solution of the system with modified right-hand side by using the new splitting via range-separated representation of the discretized Dirac delta distribution. In this new approach the problem is reduced to computation of the short-range part in the collective free space electrostatic potential of the system and to the subsequent solution of the PBE equation for the long-range part only by the simple modification of the right-hand side. The

---

[1] A PQR (or Position, charge (Q), and Radius) file is a protein data bank (PDB) file with the temperature and occupancy columns replaced by columns containing the per-atom charge (Q) and radius (R) using the pdb2pqr software. PQR files are used in several computational biology packages, including APBS [8].

advantage is that the PBE applies to the smooth part of the total potential and hence a controllable FDM/FEM approximation error on moderate size 3D grids can be achieved.

4.4. Efficient computation of electrostatic energies and forces for the PBE

4.4.1. Electrostatic solvation energy for the PBE

An important application of the PBE's electrostatic potential is the electrostatic solvation free energy, which is useful in biophysics and biomedicine [108, 140]. It is defined as the free energy required to transfer a biomolecule from a uniform dielectric continuum to an inhomogeneous medium, which is generally divided into nonpolar and polar terms [41]. The polar contribution to the solvation free energy is given by

$$\Delta G_{\text{solv}}^{\text{polar}} = G_{\text{elec}}^{\text{solv}} - G_{\text{elec}}^{\text{ref}}, \tag{4.29}$$

where $G_{\text{elec}}^{\text{ref}}$ (reference energy) is the total biomolecular electrostatic free energy in the reference or vacuum state (solute homogeneous dielectric medium) and $G_{\text{elec}}^{\text{solv}}$ (solvated energy) is that in the solvated state (inhomogeneous dielectric medium, for instance, a protein in aqueous medium) [41, 140]. The electrostatic energy by definition, represents the work required to assemble the biomolecule, and is given by

$$G_{\text{elec}} = \frac{1}{2}\sum_{i=1}^{N_m} q_i u(\bar{x}_i), \tag{4.30}$$

where $u(\bar{x}_i)$ is the mean electrostatic potential acting on the atom located at $\bar{x}_i$ with charge q_i [140].

The APBS software package, for example, has an option which implements the multigrid or multilevel finite difference calculations using PMG [2] [8, 30], solve the PBE twice for the components of the solvation free energy in (4.29). This is computationally demanding, especially if the biomolecule under investigation is very large, such that the computational domain is chosen large enough in order to accurately approximate the boundary conditions, leading to high degrees of freedom $\mathcal{O}(10^6)$ in the discretized PBE [67, 108]. Notice that by the RS tensor format, we solve only for the solvated energy ($G_{\text{elec}}^{\text{solv}}$) component numerically using the linearized variant of RPBE in (4.16) while the reference energy ($G_{\text{elec}}^{\text{ref}}$) is determined directly from the precomputed long-range component of the free space electrostatic potential $\mathbf{P}_l$ in (3.35).

However, since the emergence of [30], options to calculate the regularized linear and nonlinear PBE as in (2.48) using the finite element toolkit (FEtk), have also been incorporated in the APBS, which yields the solvation energy without the need for reference calculations. This is a positive step in the history of APBS, although, as we observed earlier, the regularization scheme in (2.48) suffers from the inability to efficiently separate the long- and short-range components.

---

[2] **PMG** is a Parallel algebraic MultiGrid code for general semilinear elliptic equations[147].

A striking feature of the electrostatic energy of interaction is that it is entirely driven by the long-range electrostatic potential. This is because the short-range components do not communicate with their neighbours due to their localization (effective local support) in the atomic volumes. To justify this claim, we prove in Lemma 4.4, that the solvation free energy calculation depends only on the long-range components of the free-space potential and that in the solvated state.

Lemma 4.4:

Let the total free-space (reference) electrostatic potential in (4.9) be given by the sum $u^{\mathrm{tot}}_{\mathrm{ref}}(\bar{x}) = u^{s}_{\mathrm{ref}}(\bar{x}) + u^{r}_{\mathrm{ref}}(\bar{x})$ and that of the solvated state of the PBE decomposition in (4.16) by $u^{\mathrm{tot}}_{\mathrm{solv}}(\bar{x}) = u^{s}_{\mathrm{ref}}(\bar{x}) + u^{r}_{\mathrm{solv}}(\bar{x})$, $\bar{x} \in \mathbb{R}^3$, using the RS tensor splitting scheme. Then the solvation free energy in (4.29) is given by the regularized form:

$$\Delta G^{r}_{\mathrm{solv}} = \frac{1}{2}\sum_{i=1}^{N_m} q_i u^{r}_{\mathrm{solv}}(\bar{x}_i) - \frac{1}{2}\sum_{i=1}^{N_m} q_i u^{r}_{\mathrm{ref}}(\bar{x}_i). \tag{4.31}$$

◊

Proof. We substitute the components of the total electrostatic potential calculations for the reference and solvated states of the biomolecule

$$\left.\begin{aligned}
\Delta G^{\mathrm{polar}}_{\mathrm{solv}} &= G^{\mathrm{solv}}_{\mathrm{elec}} - G^{\mathrm{ref}}_{\mathrm{elec}},\\
&= \frac{1}{2}\sum_{i=1}^{N_m} q_i u^{\mathrm{tot}}_{\mathrm{solv}}(\bar{x}_i) - \frac{1}{2}\sum_{i=1}^{N_m} q_i u^{\mathrm{tot}}_{\mathrm{ref}}(\bar{x}_i),\\
&= \frac{1}{2}\sum_{i=1}^{N_m} q_i (u^{s}_{\mathrm{ref}}(\bar{x}_i) + u^{r}_{\mathrm{solv}}(\bar{x}_i)) - \frac{1}{2}\sum_{i=1}^{N_m} q_i (u^{s}_{\mathrm{ref}}(\bar{x}_i) + u^{r}_{\mathrm{ref}}(\bar{x}_i)),\\
&= \frac{1}{2}\sum_{i=1}^{N_m} q_i \left(u^{r}_{\mathrm{solv}}(\bar{x}_i) - u^{r}_{\mathrm{ref}}(\bar{x}_i)\right)\\
&= \Delta G^{r}_{\mathrm{solv}}.
\end{aligned}\right\} \tag{4.32}$$

□

The result for the electrostatic potential energy in (4.30), which is based on the effective support of the short-range part of the total potential, was presented in (Lemma 4.2 in [17]). Therefore, by applying the RS tensor format to the PBE, we can leverage this property to greatly reduce the computational costs and errors, which are inherent in the current software packages. Other electrostatic energies, such as the electrostatic free energy, the folding free energy, and the binding free energy can be calculated in a similar way [41]. We leave this for future work.

4.4.2. Electrostatic forces for the PBE

Electrostatic forces on atoms are critical quantities for a range of biomolecular simulations, for example, the molecular structure optimization and equilibrium molecular dynamics [146]. Electrostatic or polar solvation forces, which are obtained by the derivative of the electrostatic energy with respect to the interatomic distance [55, 72], can also be computed efficiently using the results in Lemma 4.4. The traditional computation of these forces is given by

$$\mathbf{f}_i = -\frac{\partial G(r)}{\partial r_i}, \tag{4.33}$$

where r_i is the atomic position and G is the electrostatic energy in (4.30).

Since the electrostatic forces are long-ranged by nature [15, 18, 48], it suffices to compute these forces from the resulting electrostatic solvation energies obtained in (4.32). We claim this in the following result.

Lemma 4.5:

Let the regularized electrostatic solvation energy be derived as in (4.32) via the long-range electrostatic potential component. Then, the corresponding regularized electrostatic force of interaction $\mathbf{f}^r$ can be computed from the derivative of the regularized solvation energy ΔG^r_{solv}. ◊

Proof. Substituting the regularized solvation energy ΔG^r_{solv} into (4.33), we obtain the required regularized electrostatic forces of interaction, i.e.,

$$\mathbf{f}^r_i = -\frac{\partial(\Delta G^r(\mathbf{r}))}{\partial \mathbf{r}_i}, \tag{4.34}$$

which completes the proof. □

4.5. Conclusions

In this chapter, we have applied the novel RS tensor format to the Poisson-Boltzmann equation in order to efficiently construct a regularized PBE model, via the splitting of the Dirac delta distribution and the substitution of the singular sources by a smooth and regularized function. The regularized PBE model enables the efficient computation of only the long-range electrostatic potential, thereby avoiding the numerical approximation of the singular sources, which increases the accuracy. We also prove that the electrostatic energy can be computed a lot more efficiently and accurately by using the long-range potential. Electrostatic forces follow suit.

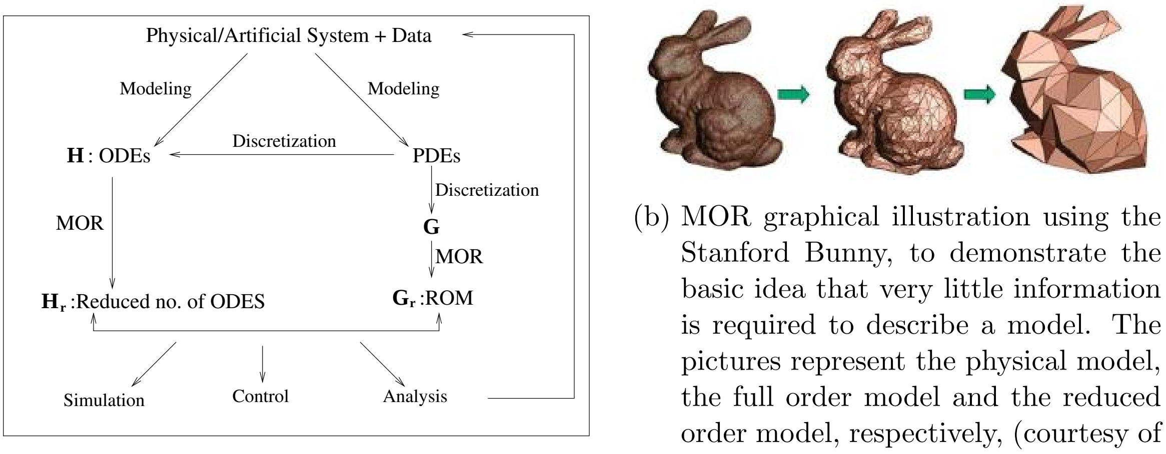
CHAPTER 5

REDUCED BASIS METHODS

Contents

5.1. Introduction

In this chapter, we first review the general model order reduction (MOR) techniques in Section 5.2 in order to provide a better understanding of their applications and relevance in various contexts. Then we discuss the reduced basis method (RBM) for both the classical linearized PBE (LPBE) in Section 5.3 and the classical nonlinear PBE (NPBE) in Section 5.6.2, and highlight some of the computational challenges encountered therein and how to circumvent them. In Section 5.6.1 we apply the RBM



(a) The big picture of MOR [129].

(b) MOR graphical illustration using the Stanford Bunny, to demonstrate the basic idea that very little information is required to describe a model. The pictures represent the physical model, the full order model and the reduced order model, respectively, (courtesy of [130]).

Figure 5.1.: The big picture representation of MOR 5.1a, and its graphical illustration (i.e., easy-to-understand) on the Stanford Bunny in 5.1b.

to the range-separated tensor-based regularized PBE (RPBE), which is more accurate and computationally cheaper than the classical versions.

5.2. Model order reduction techniques in a nutshell

The significance of numerical simulations have tremendously increased in the scientific and engineering disciplines. Real time numerical capabilities are essential in predicting the behaviour of complex systems in most engineering realms [11]. Modeling of these systems usually results in differential systems (ODEs or PDEs) of higher orders, whose prediction and optimization requires a computationally expensive numerical approximation, due to the high number of degrees of freedom involved. Therefore, it is vital to construct accurate and efficient low dimensional surrogate reduced order models (ROMs) for these systems in order to perform rapid design and optimization at a fraction of the original computational costs [11]. The aforementioned process is what is generally referred to as model order reduction (MOR).

Figure 5.1 demonstrates the general steps involved in MOR, whereby the discretized systems of equations can be of the order of $\mathcal{O}(10^5) - \mathcal{O}(10^9)$ degrees of freedom. Note that the reduction can yield small ROMs of the order of $\mathcal{O}(10)$.

5.2.1. MOR techniques in different fields

MOR techniques have been developed in different fields since the 1960s. Here, we highlight some of these fields and their respective MOR approaches [20, 130].

(1) **Structural dynamics.** This is the field where the earliest MOR techniques were developed in the 1960s. In this field, the dynamic analysis of structures is of major interest and the major objectives are the identification of eigenfrequencies or the computation of frequency response functions [20].

  - Mode displacement methods are based on the free vibration modes of the structure. These modes can be obtained via a time-harmonic representation of the displacement of the unforced system.
  - Mode acceleration method is a computational variant of the static correlation method [20] whose objective is to take into account the contribution of the omitted modes. It is also an improvement of the mode displacement method.
  - Modal truncation augmentation method is an extension of the mode acceleration approach whose aim is to utilize the static correlation as an additional direction for the truncation expansion.

(2) **Systems and control.** MOR for the analysis of dynamical systems and the design of feedback controllers have been of major interest since the 1980s. The main techniques here include:

  - Balanced truncation, whose main purpose is the approximation of input-output behaviour of a system. Here, the idea is to keep those 'states' in the ROM that contribute significantly to transferring energy from input to output.
  - Optimal Hankel norm approximation provides a ROM best approximating the original system in the Hankel semi-norm.

(3) **Scientific computing (numerical mathematics).** Numerically efficient MOR techniques in this field have been developed since the 1990s. Some of the techniques here include:

  - Krylov subspace based methods, for example, rational interpolation or moment matching, whose main purpose is the approximation of the system transfer function that accurately describes the input-output bahaviour in some range of the frequency domain. Some of the applications of these techniques are large electronic circuits involving large linear subnetworks of components [20].
  - Proper Orthogonal Decomposition (POD) is a data based reduction technique in which time is considered as the varying parameter and the snapshots of the field variable (or the state) are computed numerically at different times. Singular value decomposition (SVD) is applied to the snapshot matrix in order to retain only a few modes which correspond to the largest singular values. The ROM is then obtained via a Galerkin projection onto the space spanned by the retained vectors [141].
  - The reduced basis method is a parametrized MOR (PMOR) technique that provides rapid and reliable approximation of solutions to parametrized PDEs (PPDEs) for parameter space exploration, for example, in design and optimization, control, and parameter estimation.

MOR techniques are further divided into two major groups: the parametrized MOR (PMOR) and the non-parametrized MOR. In Section 5.2.2, we briefly highlight the popular PMOR techniques among which the reduced basis method (RBM), considered in this thesis, falls under.

5.2.2. Parametrized MOR techniques

Most engineering systems are always parametrized in order to enable variations in material, shape, loading, and boundary conditions during their design and analysis. Therefore, PMOR techniques aim at reflecting this parameter dependency in the ROM, thereby enabling expeditious optimization, control, uncertainty quantification and design space exploration cycles via the parametric simulations in the reduced space [10, 11, 14]. The following are the popular PMOR techniques in the general MOR community.

- Reduced basis methods (RBM) [59, 119, 123, 128]
- Proper orthogonal decomposition (POD) [62, 93, 94, 95, 141]
- Multi-dimensional moment matching [13, 43, 45, 49]
- System interpolation [9, 152]
- Empirical cross Gramian [60, 73, 74, 153]

In this thesis, we focus on the RBM technique, and we now dive into the details of the RBM in the following Section 5.3.

5.3. Reduced basis method for the classical LPBE

The Reduced basis method (RBM) is a projection-based parametrized model order reduction (PMOR) technique. The main goal is to generate a parametric ROM which accurately approximates the original full order model (FOM) of high dimension over varying parameter values [14, 42, 128]. The RBM leverages an offline/online procedure which ensures an accurate approximation of the high-fidelity solution in a rapid and inexpensive manner and is widely applicable in real-time and many-query scenarios. For a thorough review, see [128].

We consider a physical domain $\Omega \subset \mathbb{R}^3$ with boundary $\partial\Omega$, and a parameter domain $D \subset \mathbb{R}$. The LPBE (2.21) is discretized with the centered finite difference scheme (2.35) on Ω and Dirichlet boundary conditions (2.16) obtained from (2.36) are applied. The resultant discrete problem of the LPBE becomes, for any $\mu \in D$, find $u^{\mathcal{N}}(\mu) \in \mathbb{R}^{\mathcal{N}}$ that satisfies the linear system

$$A(\mu)u^{\mathcal{N}}(\mu) = f(\mu), \quad \mu \in D, \tag{5.1}$$

where $A(\mu) \in \mathbb{R}^{\mathcal{N}\times\mathcal{N}}$ and $f(\mu) \in \mathbb{R}^{\mathcal{N}}$. We also assume that the matrix $A(\mu)$ can be written as a parameter-affine matrix,

$$A(\mu) = \sum_{i=1}^{Q} \Theta_i(\mu)A_i, \tag{5.2}$$

where $Q \in \mathbb{N}$, Θ_i are scalar coefficient functions, and A_i are the parameter independent matrices.

The $\mathcal{N} \times \mathcal{N}$ system is indeed computationally expensive to be solved for an accurate approximation of $u(\mu)$ because the dimension $\mathcal{N}$ is approximately 2×10^6 for our problem. Therefore, we apply the RBM to save computational costs by providing an accurate approximation of $u^{\mathcal{N}}(\mu)$ at a greatly reduced dimension of $N \ll \mathcal{N}$. The ROM is given by (5.5).

However, as detailed in Section 5.3.2, we encounter some computational complexity in the online phase of RBM which is caused by the nonaffine parameter dependence in the right-hand side vector $f(\mu)$ from the boundary condition (2.36). The parameter, the ionic strength, resides in the kappa term κ in the exponential function. This violates one of the key assumptions of the RBM which requires that all the system matrices and vectors must be affinely dependent on the parameter so that the offline/online decomposition is natural [59]. To circumvent this problem, we propose to apply an empirical interpolation method to reduce the complexity of the the online phase by avoiding the high-dimensional computation related to the vector $f(\mu)$. We provide some details in Section 5.4.

5.3.1. The solution manifold and the greedy algorithm

Another key assumption of RBM is the existence of a typically smooth and very low dimensional solution manifold which almost covers all the high-fidelity solutions of (5.1) under variation of parameters [42, 128],

$$\mathcal{M}^{\mathcal{N}} = \{u^{\mathcal{N}}(\mu) : \mu \in D\}. \tag{5.3}$$

The RB approximation space is then built upon this solution manifold and is given by the subspace spanned by the snapshots of the FOM. In other words, it is the subspace spanned by the high-fidelity $u^{\mathcal{N}}(\mu)$ solutions corresponding to a number of samples of the parameters, that is,

$$\text{range}(V) = \text{span}\{u^{\mathcal{N}}(\mu_1), ..., u^{\mathcal{N}}(\mu_l)\}, \quad \forall \mu_1, ..., \mu_l \in D. \tag{5.4}$$

The greedy algorithm as given in Algorithm 5.1 is used to generate the reduced basis space (5.4) through an iterative procedure where a new basis is computed at each iteration [57]. The RB space can be thought of as being nested or hierarchical such that the previous basis set is a subset of the next and so on.

Algorithm 5.1: Greedy algorithm for the classical LPBE [58]

```
Data: A training set $\Xi \subset D$ including samples of $\mu$ covering the parameter
      domain $D$, i.e., $\Xi := \{\mu_1, \dots, \mu_l\}$.
Result: RB basis represented by the projection matrix $V$.
Choose $\mu^* \in \Xi$ arbitrarily.
Solve FOM (5.1) for $u^{\mathcal{N}}(\mu^*)$.
$V_1 = [u^{\mathcal{N}}(\mu^*)]$, $N = 1$
while $\max_{\mu \in \Xi} \Delta_N(\mu) \geq \epsilon$ do
    $\mu^* = \arg\max_{\mu \in \Xi} \Delta_N(\mu)$.
    Solve FOM (5.1) for $u^{\mathcal{N}}(\mu^*)$.
    $V_{N+1} = [V_N \quad u^{\mathcal{N}}(\mu^*)]$.
    Orthonormalize the columns of $V_{N+1}$.
    $N = N + 1$.
    Update $\Delta_N(\mu)$.
```

The RB approximation is then formulated as, for any given $\mu \in D$, find $u_N(\mu) \in X_N$ which satisfies

$$A_N(\mu) u_N(\mu) = f_N(\mu), \tag{5.5}$$

where $A_N = V^T A V$ and $f_N(\mu) = V^T f(\mu)$. V is the orthonormal matrix computed from the greedy algorithm. From the fact that $N \ll \mathcal{N}$, solving the small dimensional reduced order model (ROM) is much cheaper than solving the high-fidelity model, the FOM (5.1) [42]. However, one problem still remains when computing the ROM. The computational complexity of evaluating the nonaffine function $f_N(\mu)$ still depends on the dimension of the FOM, as illustrated in Section 5.3.2. Efficient implementation of Algorithm 5.1 depends on an efficient error estimation $\Delta_N(\mu)$ of the ROM, which is discussed in Section 5.5.

5.3.2. Computational complexity of the reduced order model (ROM)

To demystify the issue of computational complexity in the ROM, we can first rewrite (5.1) explicitly to illustrate the affine parameter decomposition on the left-hand side and the nonaffine right-hand side,

$$(A_1 + \mu A_2) u^{\mathcal{N}} = \rho + b(\mu), \quad \mu \in D, \tag{5.6}$$

where the matrix A_1 comes from the Laplacian operator term, A_2 is a diagonal matrix from the *kappa* term, ρ represents the charge density term and $b(\mu)$, the boundary conditions obtained from the analytical solution in (2.36). We can clearly notice the affine parameter decomposition of the matrix A in (5.1) into A_1 and μA_2 in (5.6). However, the right-hand side function $b(\mu)$ is nonaffine in the parameter and therefore it cannot be decomposed in such a manner. Consider the ROM which is obtained by the greedy algorithm approach in Algorithm 5.1 and a Galerkin projection,

$$(\underbrace{\hat{A}_1}_{N\times N}+\mu\underbrace{\hat{A}_2}_{N\times N})\underbrace{u^N}_{N\times 1}=\underbrace{\hat{\rho}}_{N\times 1}+\underbrace{V^T}_{N\times\mathcal{N}}\underbrace{b(\mu)}_{\mathcal{N}\times 1}, \tag{5.7}$$

where $\hat{A}_1 = V^T A_1 V$, $\hat{A}_2 = V^T A_2 V$, $\hat{\rho} = V^T \rho$, and $N \ll \mathcal{N}$.

It is clear from (5.7) that the last term of the right-hand side (RHS) still depends on the dimension $\mathcal{N}$ of the FOM while all the other matrices and vectors depend only on the dimension N of the ROM, with $N \ll \mathcal{N}$. Therefore, the reduced order matrices on the left-hand side and the first vector on the right-hand side of (5.7) can be precomputed and stored during the offline phase, thereby providing a lot of computational savings. However, the term $V^T b(\mu)$ cannot be precomputed because of the aforementioned nonaffine parameter dependence and therefore, the Galerkin projection involving matrix-vector products which are dependent on the dimension $\mathcal{N}$, has to be computed in the online phase of solving the ROM.

In principle, we require $\mathcal{O}(2\mathcal{N}N)$ flops for these matrix-vector products and a full evaluation of the nonaffine analytical function (2.36) to obtain $V^T b(\mu)$. This can be computationally expensive for a large $\mathcal{N}$, especially during the *a posteriori* error estimation (computing $\Delta_N(\mu)$), where the residual is computed l times for varying parameter values μ_i, $i = 1, \ldots, l$ for a single iteration of the greedy algorithm. The discrete empirical interpolation method (DEIM) is an approach to circumvent this problem in order to reduce the computational complexity of the nonaffine function. We discuss this technique at length in Section 5.4.

5.4. Discrete empirical interpolation method (DEIM)

DEIM is a complexity reduction technique that was proposed in [29] to overcome the drawback of the proper orthogonal decomposition (POD) approach for approximating a nonaffine (or nonlinear) parametrized function in the ROM during the online phase. The main idea of DEIM is to interpolate the nonlinear/nonaffine function by computing only a few entries of it, which dramatically reduces the computational complexity [29, 148].

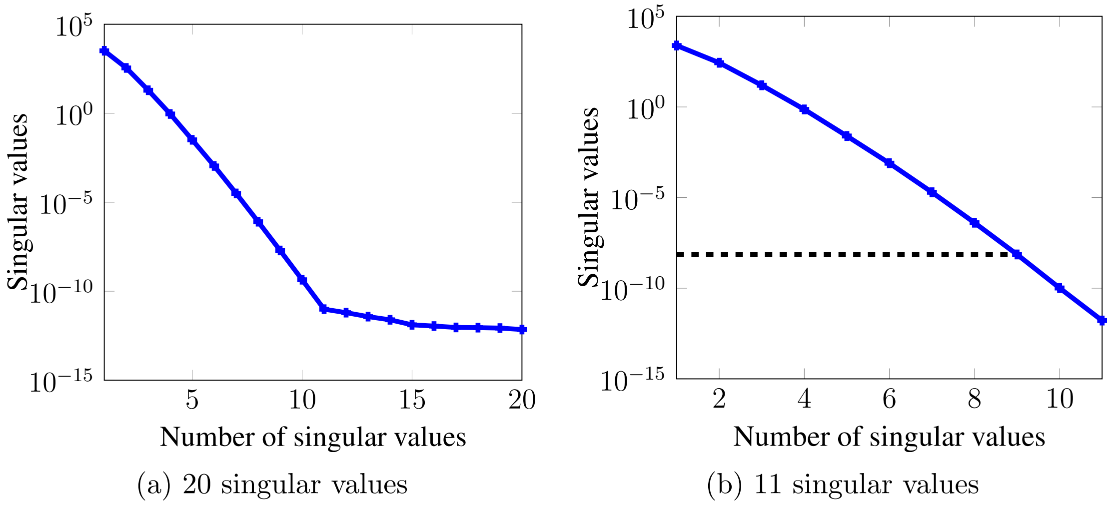
We provide a brief overview on using the singular value decomposition (SVD) to obtain the interpolation basis vectors. Firstly, we compute snapshots of the function $b(\mu)$ at a set of parameters μ in the training set $\Xi = \{\mu_1, \ldots, \mu_l\} \subset D$ and construct the snapshot matrix,

$$F = [b(\mu_1), \ldots, b(\mu_l)] \in \mathbb{R}^{\mathcal{N}\times l}. \tag{5.8}$$

Secondly, we compute its singular value decomposition (SVD),

$$F = U_F \Sigma W^T, \tag{5.9}$$

where $U_F \in \mathbb{R}^{\mathcal{N}\times l}$, $\Sigma \in \mathbb{R}^{l\times l}$, and $W \in \mathbb{R}^{l\times l}$. Note that the matrices U_F and W are orthogonal, that is, $(U_F)^T U_F = W^T W = I_l$, $I_l \in \mathbb{R}^{l\times l}$ and $\Sigma = \mathrm{diag}(\sigma_1, \ldots, \sigma_l)$, with $\sigma_1 \geq \ldots \geq \sigma_l \geq 0$.



(a) 20 singular values (b) 11 singular values

Figure 5.2.: Decay of singular values of Σ in (5.9).

Figure 5.2 shows the decay of the singular values of Σ for the protein *fasciculin 1*. Figure 5.2a shows the behaviour of 20 singular values with almost no decay from the $11th$ singular value. We discard these non-decaying singular values to obtain those in Figure 5.2b. From the latter, we can actually truncate the singular values by selecting only the r largest ones that correspond to some required degree of accuracy. In this case, $l = 11$ and $r = 9$ which corresponds to an accuracy of $\epsilon_{svd} = O(10^{-10})$ in (5.11). The number r plays an important role in selecting the basis set $\{u_i^F\}_{i=1}^r$ of rank r from U_F which solves the minimization problem [141],

$$\arg\min_{\{\tilde{u}_i\}_{i=1}^r} \sum_{j=1}^{l} \|F_j - \sum_{i=1}^{r} \langle F_j, \tilde{u}_i\rangle \tilde{u}_i\|_2^2, \quad \text{s.t.} \ \langle \tilde{u}_i, \tilde{u}_j\rangle = \delta_{ij}, \tag{5.10}$$

where F_j is the jth column of the snapshot matrix F, and δ_{ij} is the usual Kronecker delta.

The following criterion is used to truncate the largest singular values from Figure 5.2 based on some desired accuracy, ϵ_{svd}.

$$\frac{\sum\limits_{i=r+1}^{l} \sigma_i}{\sum\limits_{1=1}^{l} \sigma_i} < \epsilon_{svd}, \tag{5.11}$$

where $\sigma_i, i = 1, \ldots, l$, are the nonzero singular values of F. The dotted horizontal black line corresponds to $r = 9$ singular values at $\epsilon_{svd} = 10^{-10}$ and the corresponding singular vectors $\{u_i^F\}_{i=1}^r$ are used in the DEIM approximation.

DEIM overcomes the problem mentioned in Section 5.3.2 by determining an interpolation of the nonaffine function $b(\mu)$. This is realized by approximating $b(\mu)$ with the linear combination of the basis vectors $U_F = [u_1^F, \ldots, u_r^F] \in \mathbb{R}^{\mathcal{N}\times r}$, i.e.

$$b(\mu) \approx U_F c(\mu), \tag{5.12}$$

where $c(\mu) \in \mathbb{R}^r$ is the corresponding coefficient vector, and can be determined by assuming that $U_F c(\mu)$ interpolates $b(\mu)$ at r selected interpolation points, then,

$$P^T b(\mu) = P^T U_F c(\mu), \tag{5.13}$$

where P is an index matrix given by

$$P = [e_{\wp_1}, \ldots, e_{\wp_r}] \in \mathbb{R}^{\mathcal{N}\times r}, \tag{5.14}$$

which consists of unit vectors $e_{\wp_i}$, $i = 1, \ldots, r$, where the indices $\wp_i$, are the DEIM interpolation points which are selected iteratively with a greedy algorithm. Suppose that $P^T U_F \in \mathbb{R}^{r\times r}$ is nondegenerate, then $c(\mu)$ can be determined from (5.13) by

$$c(\mu) = (P^T U_F)^{-1} P^T b(\mu). \tag{5.15}$$

Therefore, the function $b(\mu)$ in (14) can be approximated as

$$b(\mu) \approx U_F c(\mu) = U_F (P^T U_F)^{-1} P^T b(\mu), \tag{5.16}$$

so that the ROM in (5.7) with DEIM approximation becomes,

$$(\underbrace{\hat{A}_1}_{N\times N} + \mu \underbrace{\hat{A}_2}_{N\times N}) \underbrace{u^N(\mu)}_{N\times 1} = \underbrace{\hat{\rho}}_{N\times 1} + \underbrace{V^T U_F (P^T U_F)^{-1}}_{N\times r} \underbrace{P^T b(\mu)}_{r\times 1}. \tag{5.17}$$

The interpolant $V^T U_F (P^T U_F)^{-1} P^T b(\mu)$ can be computed a lot cheaper than $V^T b(\mu)$ because we can precompute $V^T U_F (P^T U_F)^{-1}$ independent of the parameter μ. Alternatively, we can also compute only those entries in $b(\mu)$ that correspond to the interpolation indices $\wp_i, i = 1, \ldots, r$, $r \ll \mathcal{N}$, i.e., $P^T b(\mu)$ instead of the entire $\mathcal{N}$ entries in $b(\mu)$.

For the actual numerical implementation of the interpolation (5.16), the matrix P needs not be explicitly applied. Instead, only the interpolation indices $\wp_i, i = 1, \ldots, r$, need to be applied to the matrix U_F or the nonaffine function $b(\mu)$. This implies that $P^T U_F$ merely consists of the rows of U_F which correspond to the interpolation indices $\wp_i, i = 1, \ldots, r$. Similarly, $P^T b(\mu)$ is a condensed vector composed of a few entries of $b(\mu)$ which correspond to the same indices. Algorithm 5.2 provides a brief overview of the DEIM procedure.

Algorithm 5.2: DEIM algorithm [29, 46]

```
   Data: POD basis $\{u_i^F\}_{i=1}^r$ for $F$ in (5.9).
   Result: DEIM basis $U_F$ and indices $\vec{\wp} = [\wp_1, \dots, \wp_r]^T \in \mathbb{R}^r$.
1  $\wp_1 = \arg \max_{j \in \{1,\dots,\mathcal{N}\}} |u_{1j}^F|$, where $u_1^F = (u_{11}^F, \dots, u_{1\mathcal{N}}^F)^T$.
2  $U_F = [u_1^F]$, $P = [e_{\wp_1}], \vec{\wp} = [\wp_1]$.
3  for $i$ = 2 to $r$ do
4      Solve $(P^T U_F)\alpha = P^T u_i^F$ for $\alpha$, where $\alpha = (\alpha_1, \dots, \alpha_{i-1})^T$.
5      $r_i = u_i^F - U_F \alpha$.
6      $\wp_i = \arg \max_{j \in \{1,\dots,\mathcal{N}\}} |r_{ij}|$, where $r_i = (r_{i1}, \dots, r_{i\mathcal{N}})^T$.
7      $U_F \leftarrow [U_F\ u_i^F]$, $P \leftarrow [P\ e_{\wp_i}]$, $\vec{\wp} \leftarrow \begin{bmatrix} \vec{\wp} \\ \wp_i \end{bmatrix}$.
```

Note that in Algorithm 5.2, the POD basis $\{u_i^F\}_{i=1}^r$ is of great significance as an input basis for the DEIM procedure in two ways. First, a set of interpolation indices $\wp_i$ are constructed inductively based on this basis through a greedy algorithm. Secondly, an error analysis in [29] indicates that the ordering of this basis according to the dominant singular values makes it the right choice for this algorithm. In step 1, the process selects the first interpolation index $\wp_1$ which corresponds to the location of the entry in u_1^F with the largest magnitude. The subsequent indices in step 6, $\wp_i, i = 2, \dots, r$, are selected in such a way that each of them corresponds to the location of the entry in r (step 5) with the largest magnitude.

5.4.1. DEIM approximation error

We compute the error due to the DEIM interpolation which is to be included into the residual in the *a posteriori* error estimation. This error was first proposed in [148] for nonlinear dynamical systems and has also been used in [46] in the context of a nonlinear population balance systems. We extend this idea to parametrized elliptic PDEs where the DEIM error is given by,

$$e_{\text{DEIM}} = b(\mu) - \tilde{b}(\mu) = \Pi_2 (I - \Pi) b(\mu), \tag{5.18}$$

where Π and Π_2 are oblique projectors defined as follows,

$$\Pi = U_F (P^T U_F)^{-1} P^T, \tag{5.19}$$

and

$$\Pi_2 = (I - \Pi)\tilde{U}_F (\tilde{P}^T (I - \Pi)\tilde{U}_F)^{-1} \tilde{P}^T. \tag{5.20}$$

In equation (5.19), $U_F = (u_1^F, \dots, u_r^F) \in \mathbb{R}^{\mathcal{N} \times r}$ and $P \in \mathbb{R}^{\mathcal{N} \times r}$ are the current DEIM basis and interpolation index matrix obtained from Algorithm 5.2.

To obtain Π_2 in (5.20), we assume that $r^* (\geq r)$ DEIM basis vectors $U_F^* = (u_1^F, \dots, u_{r^*}^F)$ interpolate $b(\mu)$ exactly, i.e.

$$b(\mu) = U_F^* ((P^*)^T U_F^*)^{-1} (P^*)^T b(\mu), \tag{5.21}$$

where P^* is the corresponding index matrix with r^* columns. Finally, $\tilde{U}_F = U_F^*(:, r+1 : r^*)$ and $\tilde{P} = P^*(:, r+1 : r^*)$ such that $U_F^* = [U_F, \tilde{U}_F]$ and $P^* = [P, \tilde{P}]$, where $M(:, r+1 : r^*)$, using MATLAB notation [46]. In the next subsection, we introduce an *a posteriori* error estimation derived from the residual of the approximate RB solution and the DEIM approximation error.

5.5. A Posteriori error estimation

A posteriori error estimators are computable indicators which provide an estimate to the actual solution error. An efficient error estimator is required to possess three major characteristics, namely: it is required to be as sharp as possible (close to the unknown actual error), asymptotically correct (tend to zero with increasing RB space dimension N, at a similar rate as the actual error), and computationally cheap (because it is computed in the online phase). Therefore, these estimators guarantee both reliability and efficiency of the reduction process [123].

In Section 5.5.1 and Section 5.5.2, we briefly introduce the concepts of error estimators which are related to the solution vector (i.e., electrostatic potential) and the output (electrostatic energy), respectively.

5.5.1. Error estimator for the solution vector

We first compute the residual after introducing DEIM interpolation;

$$r_N^{\mathrm{DEIM}}(u_N; \mu) = (\rho + \tilde{b}(\mu)) - A^{\mathcal{N}}(\mu) u_N(\mu), \tag{5.22}$$

where $\tilde{b}(\mu) = \Pi b(\mu)$ is the DEIM interpolation of $b(\mu)$ and $u_N(\mu) := V u^N(\mu)$ is the RB solution transformed back to the high-fidelity space $\mathcal{N}$. Then the final residual is obtained by including the DEIM approximation error derived in Section 5.4.1 as follows;

$$\begin{aligned} r_N(u_N; \mu) &= (\rho + b(\mu)) - A^{\mathcal{N}}(\mu) u_N(\mu) \\ &= (\rho + \tilde{b}(\mu)) - A^{\mathcal{N}}(\mu) u_N(\mu) + b(\mu) - \tilde{b}(\mu) \\ &= r_N^{\mathrm{DEIM}}(u_N; \mu) + \underbrace{b(\mu) - \tilde{b}(\mu)}_{:= e_{\mathrm{DEIM}}} \\ &= r_N^{\mathrm{DEIM}}(u_N; \mu) + e_{\mathrm{DEIM}}. \end{aligned} \tag{5.23}$$

The *a posteriori* error estimation is then derived from the residual in (5.23). Rewriting the first equation of (5.23), we obtain

$$\begin{aligned} r_N(u_N; \mu) &= A^{\mathcal{N}}(\mu) u^{\mathcal{N}}(\mu) - A^{\mathcal{N}}(\mu) u_N(\mu) \\ &= A^{\mathcal{N}}(\mu) e(\mu), \end{aligned} \tag{5.24}$$

where the error of the solution vector $e(\mu) := u^{\mathcal{N}}(\mu) - u_N(\mu)$ is given by

$$e(\mu) = (A^{\mathcal{N}}(\mu))^{-1} r_N(u_N; \mu). \tag{5.25}$$

We obtain an upper bound for the 2-norm of the error by taking the 2-norm on both sides of equation (5.25), i.e.

$$\|e(\mu)\|_2 \leq \|(A^{\mathcal{N}})^{-1}(\mu)\|_2 \|r_N(u_N;\mu)\|_2 = \frac{\|r_N(u_N;\mu)\|_2}{\sigma_{\min}(A^{\mathcal{N}}(\mu))} =: \tilde{\Delta}_N(\mu), \tag{5.26}$$

where $\sigma_{\min}(A^{\mathcal{N}}(\mu))$ is the smallest eigenvalue of the symmetric matrix $A^{\mathcal{N}}(\mu)$ [123]. The quantity $\tilde{\Delta}_N(\mu)$ is a rigorous error bound, and can be used to select snapshots within the greedy algorithm in the offline stage and consequently to measure the accuracy of the RB approximation [57]. For efficient computation of the norm of the residual and error bounds, see [59, 123].

It is computationally expensive to compute $\sigma_{\min}(A^{\mathcal{N}}(\mu))$ in the online phase as it entails solutions of large-scale eigenvalue problems [57]. Nevertheless, in our computations, we use the norm of the residual as our error estimator because the coefficient matrix A of the PBE system $Au = b$ has small eigenvalues of the order of $\mathcal{O}(10^{-2})$, which impede the construction of tight error bounds. This error estimation provides an estimation of the true error that works well for our problem. It also provides rapid convergence as depicted in the numerical results in Figure 6.2. It is given by

$$\|e(\mu)\|_2 \approx \|r_N(u_N;\mu)\|_2 := \Delta_N(\mu). \tag{5.27}$$

5.5.2. Output error estimator

When the output becomes interesting, one can also use the output error bounds (or estimators) to measure the output error. For the PBE model, the output of interest is given by $s(\mu) = u(\mu)^T f(\mu)$, which represents the electrostatic free energy of the system. We here briefly describe the output error estimator for a compliant problem, in which the output functional is equivalent to the load/source functional, see [59] for more details. Additionally, the coeffient matrix of the system should be symmetric for any parameter $\mu \in D$. These properties are fulfilled by the PBE system. According to the derivation in [59], the output error bound is given by

$$\|s(\mu) - s_N(\mu)\|_2 \leq \Delta'_s(\mu) := \frac{\|r_N(u_N;\mu)\|_2^2}{\sigma_{\min}(A^{\mathcal{N}}(\mu))}, \tag{5.28}$$

where $s_N(\mu) = u_N(\mu)^T f_N(\mu)$ is the output computed from the ROM, and $u(\mu)$ and $u_N(\mu)$ are the solutions of the FOM in (5.1) and ROM in (5.5), respectively. We here also avoid the use of $\sigma_{\min}(A^{\mathcal{N}}(\mu))$ due to the small eigenvalues of the coefficient matrix A.

5.6. Numerical approach to solving the NRPBE

Let us consider a physical domain $\Omega \subset \mathbb{R}^3$ with boundary $\partial\Omega$, and a parameter domain $D \subset \mathbb{R}$ which represents the variation in ionic strength $I = 1/2 \sum_{j=1}^{N_{ions}} c_j z_j^2$, which is a function of the ionic concentration c_i, of the salt solution. It resides in $\bar{\kappa}^2 =$

$8\pi e^2 I/1000\epsilon\kappa_B T$. One standard way of solving the NRPBE in (4.16) is that it is first discretized in space to obtain a nonlinear system in matrix-vector form

$$A(u^r_{\mathcal{N}}(\mu)) = b^r(\mu), \quad \mu \in D, \tag{5.29}$$

where $A(u^r_{\mathcal{N}}(\mu)) \in \mathbb{R}^{\mathcal{N}\times\mathcal{N}}$, $b^r(\mu) \in \mathbb{R}^{\mathcal{N}}$, $\mu = I \in D$, and $u^r_{\mathcal{N}}(\mu)$ is the discretized solution vector. Here, $\mathcal{N}$ is of order $\mathcal{O}(10^6)$.

Then system (5.29) can be solved using several existing techniques. For example, the nonlinear relaxation method has been implemented in the Delphi software [127], the nonlinear conjugate gradient (CG) method has been implemented in University of Houston Brownian Dynamics (UHBD) software [109], the nonlinear multigrid (MG) method [115] and the inexact Newton method have been implemented in the adaptive Poisson-Boltzmann solver (APBS) software [65].

In this study, we apply a different approach of solving (4.16) [77, 110, 133]. In particular, an iterative approach is first applied to the continuous NRPBE in (4.16), where at the $(n+1)$th iteration step, the NRPBE is approximated by a linear equation via the Taylor series truncation. The expansion point of the Taylor series is the continuous solution $(u^r(\mu))^n$ at the nth iteration step.

Consider $(u^r(\mu))^n$ as the approximate solution at the nth iterative step, then the nonlinear term $\sinh((u^r(\mu))^{n+1})$ at the $(n+1)$th step is approximated by its truncated Taylor series expansion as follows

$$\sinh((u^r(\mu))^{n+1}) \approx \sinh((u^r(\mu))^n) + ((u^r(\mu))^{n+1} - (u^r(\mu))^n)\cosh((u^r(\mu))^n). \tag{5.30}$$

Substituting the approximation (5.30) into (4.16), we obtain

$$\begin{aligned} -\nabla\cdot(\epsilon(\bar{x})\nabla(u^r(\mu))^{n+1}) + \bar{\kappa}^2(\bar{x})\cosh((u^r(\mu))^n)(u^r(\mu))^{n+1} &= -\bar{\kappa}^2(\bar{x})\sinh((u^r(\mu))^n) \\ &+ \bar{\kappa}^2(\bar{x})\cosh((u^r(\mu))^n)(u^r(\mu))^n + b^r(\mu). \end{aligned} \tag{5.31}$$

The equation in (5.31) is linear, and can then be numerically solved by first applying spatial discretization. In this regard, we first define

$$\cosh\odot(u^r_{\mathcal{N}}(\mu))^n =: w = \begin{bmatrix} w_1 \\ w_2 \\ \vdots \\ w_{\mathcal{N}} \end{bmatrix}, \tag{5.32}$$

where $\odot$ is the elementwise operation on a vector.

Then, we construct the corresponding diagonal matrix from (5.32) of the form

$$B = \operatorname{diag}(w_1, w_2, \dots, w_{\mathcal{N}}).$$

Finally, we obtain the following linear system for the (n+1)st iteration

$$A_1(u^r_{\mathcal{N}}(\mu))^{n+1} + \mu A_2 B(u^r_{\mathcal{N}}(\mu))^{n+1} = -\mu A_2 \sinh\odot(u^r_{\mathcal{N}}(\mu))^n + \mu A_2 B(u^r_{\mathcal{N}}(\mu))^n + b^r_1 + b_2(\mu), \tag{5.33}$$

where A_1 is the Laplacian matrix and A_2 is a diagonal matrix containing the net $\bar{\kappa}^2$ function (i.e., $\bar{\kappa}^2/\mu$). Note that the diagonal matrix B changes at each iteration step, therefore, it cannot be precomputed. The vectors b_1^r and $b_2(\mu)$ are the regularized approximation of the Dirac delta distributions and the nonaffine (in parameter) Dirichlet boundary conditions, respectively.

Let

$$A(\cdot) = A_1 + \mu A_2 B \tag{5.34}$$

and

$$F : \mathrm{right-hand\ side\ of}\ (5.33), \tag{5.35}$$

we obtain

$$A((u_{\mathcal{N}}^r(\mu))^n)(u_{\mathcal{N}}^r(\mu))^{n+1} = F((u_{\mathcal{N}}^r(\mu))^n), \quad n = 0, 1, \ldots . \tag{5.36}$$

Then, at each iteration, system (5.36) is a linear system w.r.t. $(u_{\mathcal{N}}^r)^{n+1}$, which can be solved by any linear system solver of choice. In this study, we employ the aggregation-based algebraic multigrid method (AGMG) [1] [113]. Algorithm 5.3 summarizes the detailed iterative approach of solving (5.36). This approach of first linearization, then discretization is shown to be more efficient than the standard way of first discretization and then linearization, via, for example, the Newton iteration. The advantage of the proposed approach is that it avoids computing the Jacobian of a huge matrix. It is observed that it converges faster than the standard Newton approach.

Algorithm 5.3: Iterative solver for the NRPBE

```
Data: Initialize the potential $(u_{\mathcal{N}}^r(\mu))^0$, e.g., $(u_{\mathcal{N}}^r(\mu))^0 = 0$ and the tolerance
      $\delta^0 = 1$.
Result: The converged NRPBE solution $(u_{\mathcal{N}}^r(\mu))^n$ at $\delta^n \leq 10^{-8}$.
while $\delta^n \geq 10^{-8}$ do
    Solve the linear system (5.36) for $(u_{\mathcal{N}}^r(\mu))^{n+1}$ using AGMG.
    $\delta^{n+1} \leftarrow \|(u_{\mathcal{N}}^r(\mu))^{n+1} - (u_{\mathcal{N}}^r(\mu))^n\|_2$.
    $(u_{\mathcal{N}}^r(\mu))^n \leftarrow (u_{\mathcal{N}}^r(\mu))^{n+1}$.
```

5.6.1. The reduced basis method for the NRPBE

It is prohibitively expensive to solve the $\mathcal{N} \times \mathcal{N}$ system in (5.36) for an accurate approximation of $u_{\mathcal{N}}^r(\mu)$ because the dimension $\mathcal{N}$ ranges between $\mathcal{O}(10^6)$ and $\mathcal{O}(10^8)$ for typical biomolecules of interest.

The greedy procedure for hierarchically constructing the reduced basis space is summarized in Algorithm 5.4. The residual in Algorithm 5.4 is derived from (5.36) and the ROM solution $(\widehat{u}_{\mathcal{N}}^r(\mu))^n = V_N(u_N^r(\mu))^n$ lifted into the high-fidelity space of dimension $\mathcal{N}$, i.e.,

$$r_N((\widehat{u}_{\mathcal{N}}^r(\mu))^{n+1}) = F((\widehat{u}_{\mathcal{N}}^r(\mu))^n) - A((\widehat{u}^r(\mu))^n)(\widehat{u}_{\mathcal{N}}^r(\mu))^{n+1}. \tag{5.37}$$

---

[1] **AGMG** implements an aggregation-based algebraic multigrid method, which solves algebraic systems of linear equations, and is expected to be efficient for large systems arising from the discretization of scalar second order elliptic PDEs [113].

Algorithm 5.4: Greedy algorithm for the NRPBE [58]

```
Data: Training set $\Xi := \{\mu_1, \dots, \mu_l\} \subset D$, tolerance $\epsilon_0 = 1$, and potential
      $(u_N^r(\mu))^0$.
Result: RB basis represented by $V$ and the ROM in (5.38).
Choose $\mu^* \in \Xi$ arbitrarily.
Solve (4.16) for $u_N^r(\mu^*)$ using Algorithm 5.3.
$V_1 = [u_N^r(\mu^*)]$, $N = 1$.
Orthonormalize $V_1$.
while $\max_{\mu \in \Xi} \Delta_N(\mu) \geq \epsilon$ do
    Compute $u_N^r(\mu)$ from (5.38) using Algorithm 5.5, and calculate
      $\Delta_N(\mu) = \|r_N(\widehat{u}_N^r(\mu))\|_2$ in (5.37), $\forall\, \mu \in \Xi$,
    $\mu^* = \arg\max_{\mu \in \Xi} \Delta_N(\mu)$,
    Solve (4.16) for $u_N^r(\mu^*)$,
    $V_{N+1} \leftarrow [V_N \quad u_N^r(\mu^*)]$.
    Orthonormalize the columns of $V_{N+1}$.
    $N \leftarrow N + 1$.
```

The ROM for the system (5.36), is therefore, formulated as follows. Given any $\mu \in D$, and an initial guess $(u_N^r(\mu))^0 \in \mathbb{R}^N$, the RB approximation $(u_N^r(\mu))^{n+1}$, at the future iteration step $n+1$ satisfies the equation

$$A_N((u_N^r(\mu))^n)(u_N^r(\mu))^{n+1} = F_N((u_N^r(\mu))^n), \quad n = 0, 1, \dots, \tag{5.38}$$

where $(u_N^r(\mu))^0$ is the zero initial guess in this study and A_N and F_N are defined explicitly as

$$A_N := \widehat{A}_1(u_N^r(\mu))^{n+1} + \mu \widehat{A}_2 \widetilde{B} V_N (u_N^r(\mu))^{n+1},$$

and

$$F_N := -\mu \widehat{A}_2 \sinh \odot (\widehat{u}_N^r(\mu))^n + \mu \widehat{A}_2 \widetilde{B} V_N (u_N^r(\mu))^n + b_N^r + V_N^T b_2(\mu),$$

where $\widetilde{B} = \operatorname{diag}(\widetilde{w}_1, \widetilde{w}_2, \dots, \widetilde{w}_{\mathcal{N}})$ and

$$\cosh \odot (\widehat{u}_N^r(\mu))^n =: \widetilde{w} = \begin{bmatrix} \widetilde{w}_1 \\ \widetilde{w}_2 \\ \vdots \\ \widetilde{w}_{\mathcal{N}} \end{bmatrix}, \quad \text{where} \quad (\widehat{u}_N^r(\mu))^n = V_N (u_N^r(\mu))^n. \tag{5.39}$$

The resulting ROM is given by

$$\begin{aligned} \widehat{A}_1(u_N^r(\mu))^{n+1} + \mu \widehat{A}_2 \widetilde{B} V_N (u_N^r(\mu))^{n+1} = {} & -\mu \widehat{A}_2 \sinh \odot (\widehat{u}_N^r(\mu))^n \\ & + \mu \widehat{A}_2 \widetilde{B} V_N (u_N^r(\mu))^n + b_N^r + V_N^T b_2(\mu), \end{aligned} \tag{5.40}$$

where $(u_N^r(\mu))^{n+1}$ is the unknown solution to the ROM and $\odot$ is the elementwise operation on a vector.

The reduced matrices $\widehat{A}_1 := V_N^T A_1 V_N$ and $\widehat{A}_2 := V_N^T A_2$ and the reduced vector $b_N^r = V_N^T b_1^r$ (see (5.33)) are determined via projection with the RB basis V_N and can be precomputed in the offline phase of the greedy algorithm. However, the matrix $\widetilde{B}$ and vector $b_2(\mu)$ are updated and or changed at each iteration and for varying parameter values, respectively, hence, they cannot be precomputed. This leads to a partial offline-online decomposition scenario, whereby Galerkin projections to some terms have to be computed in the online phase.

Note that $V_N^T b_2(\mu)$ in (5.40) is computed by first evaluating a long vector $b_2(\mu)$, then projecting it onto the low dimensional space N using V_N. This is time consuming when $b_2(\mu)$ needs to be evaluated many times for many values of μ. In Section 5.6.1.1, we propose to apply DEIM technique to further reduce the computational complexity of $V_N^T b_2^r(\mu)$. Details can be found in [16], where DEIM was applied to a similar PBE problem.

5.6.1.1. Computational complexity of the regularized ROM

It is well known that another key assumption of the RBM, besides the low dimensionality of the solution manifold, is the parameter affine property, which ensures the efficiency of the offline-online decomposition by eliminating the dependency of the ROM on the dimension $\mathcal{N}$ of the truth high-fidelity FOM [59]. However, note that on the one hand, (5.33) is actually parameter nonaffine with respect to the Yukawa-type boundary conditions, represented by F in (5.35). On the other hand, the matrix A_2 requires updates at each iteration, hence Galerkin projections are unavoidable in the online phase.

Therefore, we apply DEIM to the parametric nonaffine boundary conditions, the term $b_2(\mu)$ as discussed in Section 5.4 to obtain the following approximation

$$b_2(\mu) \approx U_F c(\mu) = U_F (P^T U_F)^{-1} P^T b_2(\mu). \tag{5.41}$$

The ROM in (5.40) with DEIM approximation becomes

$$\begin{aligned}\widehat{A}_1 (u_N^r(\mu))^{n+1} + \mu \widehat{A}_2 \widetilde{B} V_N (u_N^r(\mu))^{n+1} = -\mu \widehat{A}_2 \sinh \odot (\widehat{u}_N^r(\mu))^n + \mu \widehat{A}_2 \widetilde{B} V_N (u_N^r(\mu))^n \\ + b_N^r(\mu) + V_N^T U_F (P^T U_F)^{-1} P^T b_2(\mu).\end{aligned} \tag{5.42}$$

Note that at each iteration, only a small ROM in (5.42) is solved. With its small size $N \ll \mathcal{N}$, the system (5.42) can be solved using a direct solver rather than the iterative solver (AGMG), which is applied to to the FOM in (5.31). The iterative approach of obtaining an approximate solution $V_N(u_N^r(\mu))^{n+1}$ to (5.29) using the ROM (5.42) is summarized in Algorithm 5.5.

Algorithm 5.5: Iterative solver for the regularized ROM in (5.42)

```
Data: Initialize the potential $(u_N^r(\mu))^0$, e.g., $(u_N^r(\mu))^0 = 0$ and the tolerance
      $\delta^0 = 1$.
Result: The converged ROM solution $(u_N^r(\mu))^n$ at $\delta^n \leq 10^{-8}$.
1 Precompute $\widehat{A}_1$, $b_N^r$ in (5.42) and $U_F$ and $\vec{\wp}$ in Algorithm 5.2.
2 while $\delta^n \geq 10^{-8}$ do
3     Assemble the ROM in (5.42) using the precomputed quantities in Step 1.
4     Solve the regularized ROM (5.42) for $(u_N^r(\mu))^{n+1}$.
5     $\delta^{n+1} \leftarrow \|(u_N^r(\mu))^{n+1} - (u_N^r(\mu))^n\|_2$.
6     $(u_N^r(\mu))^n \leftarrow (u_N^r(\mu))^{n+1}$.
```

Remark 5.1:

The total electrostatic potential is obtained by lifting the reduced order long-range surrogate solution into the high-fidelity space $\mathcal{N}$ and adding to the parameter independent analytically precomputed short-range component $\mathbf{P}_s$ in (3.37), i.e.,

$$u(\mu) = \mathbf{P}_s + \widehat{u}_N^r(\mu), \tag{5.43}$$

where $\widehat{u}_N^r(\mu) = V_N u_N^r(\mu)$. ◊

5.6.2. The reduced basis method for the classical NPBE

In this section, we apply RBM to the classical nonlinear PBE (NPBE), and highlight the associated computational shortcomings as compared with RBM being applied to the NRPBE. We begin by considering the FOM of the classical NPBE in (2.15) after discretization in space, i.e.,

$$A(u^{\mathcal{N}}(\mu)) = f(\mu), \quad \mu \in D, \tag{5.44}$$

where $f(\mu)$ includes both the singular sources from the right-hand side of (2.15) and the parameter non-affine Dirichlet boundary conditions from (2.36). The corresponding classical ROM is defined as

$$\widehat{A}(u_N(\mu)) = \widehat{f}(\mu), \tag{5.45}$$

where $\widehat{A}(u_N(\mu)) = V_N^T A(V_N u_N(\mu))$ and $\widehat{f} = V_N^T f$. Here, V_N can be constructed using the greedy algorithm in Algorithm 5.4 by replacing the snapshots in Step 3 and Step 9 with the solutions to (5.44).

Note that the FOM of the classical NPBE is solved iteratively in a similar way like the NRPBE using Algorithm 5.3. The corresponding iterative form of (5.44) is given by

$$A_1(u^{\mathcal{N}}(\mu))^{n+1} + \mu A_2 B_2 (u^{\mathcal{N}}(\mu))^{n+1} = -\mu A_2 \sinh \odot (u^{\mathcal{N}}(\mu))^n + \mu A_2 B_2 (u^{\mathcal{N}}(\mu))^n + f(\mu), \tag{5.46}$$

where all the quantities except B_2, f and the solution $(u^{\mathcal{N}}(\mu))^{n+1}$ are equivalent to those in the NRPBE (5.33). Here B_2 is defined as

$$B_2 = \mathrm{diag}(v_1, v_2, \dots, v_{\mathcal{N}}),$$

and is constructed from

$$\cosh \odot (u^{\mathcal{N}}(\mu))^n =: v = \begin{bmatrix} v_1 \\ v_2 \\ \vdots \\ v_{\mathcal{N}} \end{bmatrix}. \tag{5.47}$$

The ROM of (5.46) is straightforward, i.e., given any $\mu \in D$, and an initial potential distribution $(u_N(\mu))^0$, the RB approximation $(u_N(\mu))^{n+1}$, at the subsequent iteration steps $n+1$ satisfies

$$\begin{aligned} \widehat{A}_1(u_N(\mu))^{n+1} + \mu\widehat{A}_2\widetilde{B_2}V_N(u_N(\mu))^{n+1} = -\mu\widehat{A}_2 \sinh \odot (\widehat{u}^{\mathcal{N}}(\mu))^n \\ + \mu\widehat{A}_2\widetilde{B_2}V_N(u_N(\mu))^n + \widehat{f}(\mu), \end{aligned} \tag{5.48}$$

where $(\widehat{u}^{\mathcal{N}}(\mu))^n = V_N(u_N(\mu))^n$ and $\widetilde{B_2} = \mathrm{diag}(\widetilde{v}_1, \widetilde{v}_2, \dots, \widetilde{v}_{\mathcal{N}})$ is constructed from

$$\cosh \odot (\widehat{u}^{\mathcal{N}}(\mu))^n =: \widetilde{v} = \begin{bmatrix} \widetilde{v}_1 \\ \widetilde{v}_2 \\ \vdots \\ \widetilde{v}_{\mathcal{N}} \end{bmatrix}. \tag{5.49}$$

The process of iteratively solving (5.48) is similar to that of (5.38), which is provided in Algorithm 5.5.

5.7. Summary of the RBM workflow

In this ssection, we provide a succint summary of the RBM workflow for parametrized linear PDEs in Algorithm 5.3. Note that in this workflow, we assume an affine in parameter (or solution) system. Otherwise, one needs to emply the DEIM or EIM techniques for complexity reduction in the online phase as well as for the versatility of the ROM in the parameter domain.

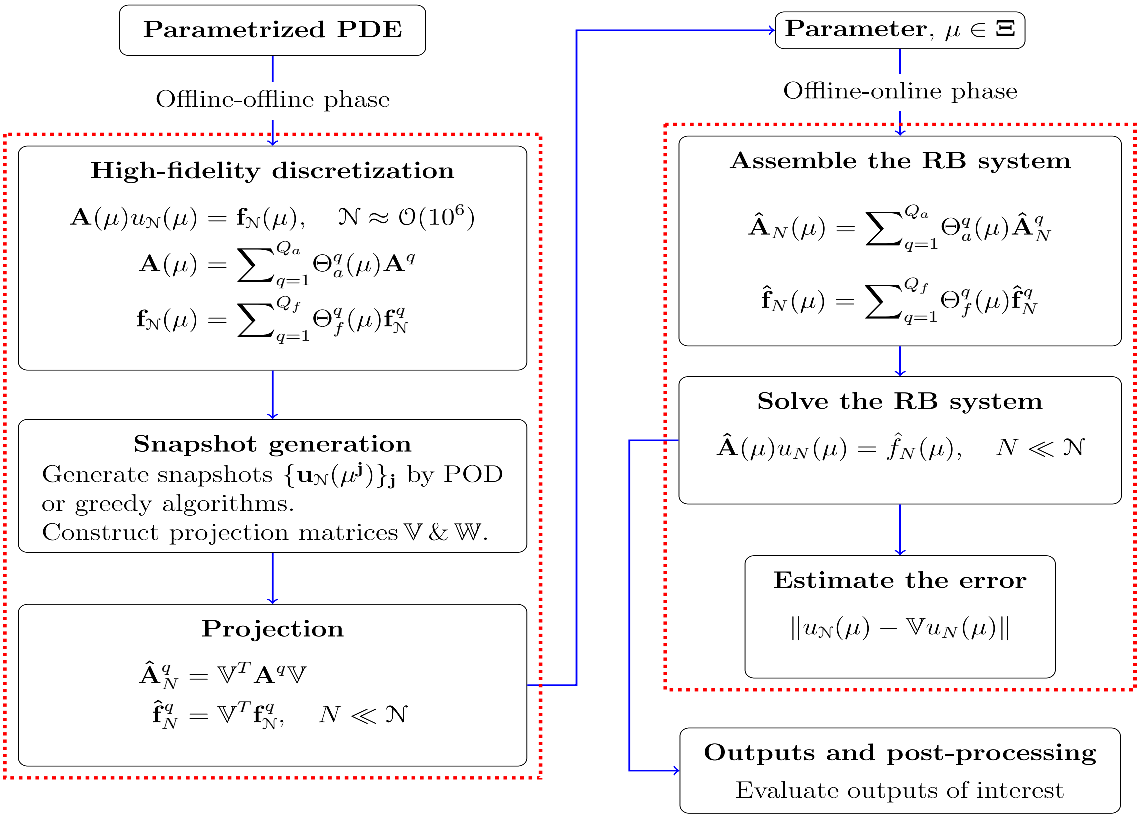


Figure 5.3.: A summary of the RBM workflow.

5.8. Conclusions

In this chapter, we have first briefly studied the general model order reduction techniques. Thereafter, we discussed the reduced basis method and applied it to the classical linearized PBE. The various computational challenges were analyzed and techniques to circumvent them were discussed. Then the nonlinear regularized PBE was considered and the RBM was applied in order to construct a surrogate regularized ROM (RROM). Last but not least, the nonlinear classical PBE was also briefly considered in order to facilitate the comparison with and to demonstrate the efficiency and efficacy of the regularized variant in Chapter 6.

CHAPTER 6

NUMERICAL EXAMPLES

Contents

6.1. Introduction

In this chapter, we first present the numerical results for the classical linearized Poisson-Boltzmann equation (LPBE), which encompasses the various FDM tests using different biomolecules in Section 6.2. The reduced basis approximations for these results are presented in Section 6.3. Then we present the numerical tests for the range-separated (RS) tensor-based regularized PBE in Section 6.4, wherein the accuracies of both the free space potential modeled by the regularized Poisson equation (RPE), and the electrostatic potential of solvated biomolecules in a medium of inhomogeneous (jumping) dielectric coefficients modeled by the RPBE, are shown. The reduced basis results for the regularized nonlinear RPBE are discussed in Section 6.5 and its accuracy and efficacy over the classical NPBE is demonstrated therein.

6.2. Numerical results for the classical LPBE

6.2.1. Finite difference results

We consider the LPBE (2.21), a parameter domain $D = [0.05, 0.15]$, and a cubic grid of 129 points and a box length of 60 Å centered at the protein position. The parameter domain is chosen for a feasible physiological process and μ resides in the second term in the kappa function. Information about the molecular charge density is obtained from a PQR [1] file which contains 1228 atoms of the protein *fasciculin 1* toxin CPDB entry 1FAS. We discretize the LPBE with a centered finite difference scheme and the resulting parametrized linear system (5.29) has more than 2×10^6 degrees of freedom. This FOM is solved by the aggregation-based algebraic multigrid (AGMG) method, where a tolerance of 10^{-10} and a zero initial guess are used [113, 114].

It is worth to note that in the absence of ions (that is, at $\mu = I = 0$), the resultant free space electrostatic potential experiences a slow polynomial decay in $1/\|\bar{x}\|$. This is attributed to the large force constant (332 kcal/mol) of the long-ranged electrostatic interactions. However, in the presence of ions (that is, $\mu > 0$), they are damped or screened, thereby experiencing an exponential decay [48]. The computational time taken to obtain the high-fidelity solution $u^{\mathcal{N}}(\mu)$ is approximately 28 seconds on average and varies depending on the value of the ionic strength used. Figure 6.1 shows the lower cross-sections of the z-axis of the electrostatic potential $u(x, y, 1)$.

6.2.2. Accuracy of FDM

We demonstrate the accuracy and reliability of the FDM before applying the RBM for the solution of the PBE. This is because the accuracy of the RBM depends on that of the underlying discretization technique. In this study, we consider six test examples to validate the FDM which include a Born ion and five proteins consisting of between 380

---

[1] A PQR (or Position, charge (Q), and Radius) file is a protein data bank (PDB) file with the temperature and occupancy columns replaced by columns containing the per-atom charge (Q) and radius (R) using the pdb2pqr software. PQR files are used in several computational biology packages, including APBS [6].

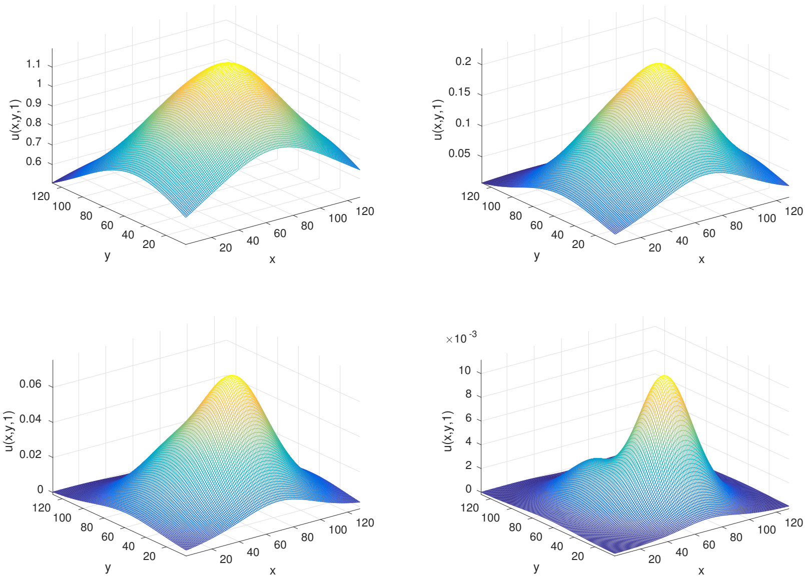


Figure 6.1.: High-fidelity solutions ($u^{\mathcal{N}}(\mu)$) at varying ionic strengths (i.e., $\mu = \{0,$ $0.05,\ 0.15,\ 0.5\}$), respectively.

and 3400 atoms, respectively. We compare the FDM results with those of APBS for electrostatic solvation free energy at different mesh refinements. Firstly, we consider the Born ion which is a canonical example for polar solvation and whose analytical solution is well known.

This analytical solution gives the polar solvation energy which results from the transfer of a non-polarizable ion between two dielectrics [1], i.e.,

$$\Delta_p G_{\text{Born}} = \frac{q^2}{8\pi\epsilon_0 r}\left(\frac{1}{\epsilon_s} - \frac{1}{\epsilon_m}\right), \tag{6.1}$$

where q is the ion charge, r is the ion radius, ϵ_s is the external dielectric coefficient (e.g., water) and ϵ_m is the internal dielectric coefficient (e.g., vacuum). This model assumes zero ionic strength. We consider a Born ion of unit charge, 3Å radius and located at the origin ($(0,0,0)$). Here, $\epsilon_m = 1$ and $\epsilon_s = 78.54$. With these parameters, the analytical solution in (6.1) is

$$\Delta_p G_{\text{Born}} = -691.85\left(\frac{q^2}{r}\right) = -230.62\text{kJ/mol}. \tag{6.2}$$

We compare numerical computations using equation (2.34) for solvation energies in a homogeneous ($\epsilon_m = \epsilon_s = 1$) and heterogeneous ($\epsilon_m = 1, \epsilon_s = 78.54$) dielectric environment with the analytical solution [1]. We use the following additional parameters. We consider two different mesh sizes (or Δx), which result in different numbers of degrees of freedom (or $\mathcal{N}$) as shown in Table 6.1. Numerical results using FDM are compared with those of the exact solution (6.1) and APBS (which uses FEM). The results show that the FDM method gives solutions which are consistent with those of the exact solutions, as well as those of the APBS software package.

Table 6.1.: Comparison of Born ion solvation energies in kJ/mol.

Δx	$\mathcal{N}$	Solver	Numerical	Analytical	Relative error
0.33	97^3	APBS	-229.59	-230.62	4.47×10^{-3}
		FDM	-232.86	-230.62	9.71×10^{-3}
0.25	129^3	APBS	-230.00	-230.62	2.69×10^{-3}
		FDM	-230.42	-230.62	8.67×10^{-4}

Secondly, we compare the accuracy of FDM for the LPBE with the following set of typical examples of use of LPBE and APBS in particular: Calculation of the total electrostatic energy (including self-interaction energies) of a 22 residue, α-helical peptide from the N protein of phage λ which binds to its cognate 19 nucleotide box B RNA hairpin [52], *fasciculin 1*, an anti-acetylcholinesterase toxin from green mamba snake venom [104], the electrostatic potential of a minimized FKBP protein from binding energy calculations of small ligands [27], a 180-residue cytokine solution NMR structure of a murine-human chimera of leukemia inhibitory factor (LIF) [61], and the binding energy of a balanol ligand to the catalytic subunit of the CAMP-dependent protein kinase A, here the apo form of the enzyme [111]. The proteins and or complexes have the following number of atoms (379, 1228, 1663, 2809, and 3423), respectively.

The electrostatic solvation free energies, ΔE are computed and shown in Table 6.2 for varying grid resolutions Δx. However, we here do not have the analytical electrostatic energies for these proteins but rely on the accuracy of the APBS software for validation. A compute cluster with 4 Intel Xeon E7-8837 CPUs running at 2.67 GHz (8 cores per CPU) and 1 TB RAM, split into four 256 GB parts (each CPU controls one part) is used to carry out the computations which require a huge amount of memory, so that it allows for solving large-scale problems with $\mathcal{N} \geq (3 \times 10^6)$.

Remark 6.1:

From Table 6.2, we can clearly see that the results of the FDM method agree well with those of APBS in terms of convergence with respect to mesh refinement. Hence, we conclude that we can test the RBM in conjunction with our FDM solver reliably. We expect no differences when using a FEM solver like APBS, which would require intruding the software. $\Diamond$

Table 6.2.: Comparison of electrostatic solvation free energies ΔE, between FDM and APBS for different proteins.

Δx	$\mathcal{N}$	ΔE, FDM	ΔE, APBS	Relative error
1. Solvation energies of a 22 residue, α-helical peptide from the N protein of phage λ which binds to its cognate 19 nucleotide box B RNA hairpin in kJ/mol. (379 atoms)				
0.375	129^3	-4557.71	-4546.52	2.46×10^{-3}
0.320	161^3	-4541.48	-4532.76	1.92×10^{-3}
0.260	193^3	-4522.48	-4516.85	1.24×10^{-3}
2. Solvation energies of *fasciculin 1*, an anti-acetylcholinesterase toxin from green mamba snake venom in kJ/mol. (1228 atoms)				
0.465	129^3	-5870.54	-5845.86	4.22×10^{-3}
0.375	161^3	-5684.85	-5664.85	3.53×10^{-3}
0.320	193^3	-5629.20	-5611.25	3.20×10^{-3}
3. Solvation energies of a minimized FKBP protein from binding energy calculations of small ligands in kJ/mol. (1663 atoms)				
0.465	129^3	-4419.04	-4403.88	3.44×10^{-3}
0.375	193^3	-4344.55	-4331.10	3.11×10^{-3}
0.320	225^3	-4292.54	-4288.08	1.04×10^{-3}
4. Solvation energies of a 180-residue cytokine solution NMR structure of a murine-human chimera of leukemia inhibitory factor (LIF) in kJ/mol. (2809 atoms)				
0.450	161^3	-9317.76	-9293.98	2.56×10^{-3}
0.375	193^3	-9270.05	-9247.28	2.46×10^{-3}
0.280	257^3	-9153.95	-9134.29	2.15×10^{-3}
5. Solvation energies of CAMP-dependent protein kinase A, here the apo form of the enzyme, in kJ/mol. (3423 atoms)				
0.465	129^3	-19742.36	-19681.32	3.10×10^{-3}
0.375	161^3	-19332.66	-19296.63	1.87×10^{-3}
0.320	193^3	-19039.86	-19014.04	1.36×10^{-3}

6.3. Reduced basis approximation for the classical LPBE

In this section, we evaluate the accuracy of the reduced basis method (RBM) for the approximation of the high-fidelity solutions generated by the FDM for the five proteins which were investigated in Section 6.2.2. We consider a cubic domain of 129 points and a box length of 60 Å centered at the protein position for all the computations.

Figure 6.2 shows the decay of the error estimator and the true error during the greedy algorithm at the current RB dimension $i = 1, \ldots, N$. They corroborate the asymptotic correctness property stated in Section 5.5, and it is evident that the error estimator is

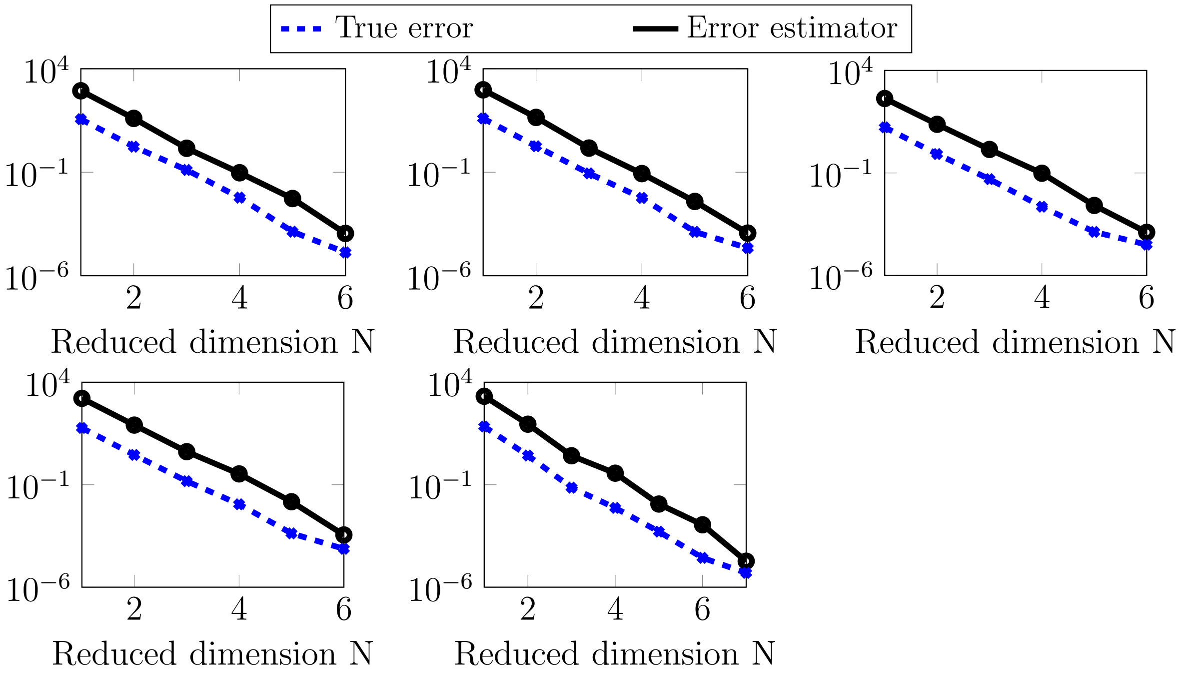


Figure 6.2.: Comparison of maximal error estimator and true error for the proteins in Table 6.2, respectively.

an upper bound to the true error. We also observe a high convergence rate of the error estimator with up to two orders of magnitude and the RB space is rich enough at only six iterations of the greedy algorithm for the five proteins.

These error estimators are the maximal residual and relative maximal residual, respectively, and are defined as, $\Delta_N^{\max} = \max_{\mu \in \Xi} \|r_N(u_N; \mu)\|_2$, and $\Delta_N^{\max}/\|u_N(\mu^*)\|_2$, where $\mu^* = \arg\max_{\mu \in \Xi} \|r_N(u_N; \mu)\|_2$.

Note that we here set the inf-sup constant, $\sigma_{\min}(A^{\mathcal{N}}(\mu))$ in (5.26) to unity $\forall\ \mu \in \Xi$ as in (5.27), because they are of order $\mathcal{O}(10^{-2})$. This makes the norm of the residual in (5.27) a better error estimator than (5.26) for this specific problem.

In the greedy algorithm, we apply an error tolerance of $\epsilon = 10^{-3}$ and a training set Ξ consisting of $l = 11$ samples of the parameter. From Figure 6.2, it is evident that both the error estimator and the true error fall below the prescribed tolerances at the final dimension of the ROM (i.e., $N = 6$).

A standard measure to determine the efficiency and the quality of the error estimator is the so-called effectivity index [4] given by

$$\text{eff} := \frac{\Delta_N(\mu)}{\|u^{\mathcal{N}}(\mu) - u_N(\mu)\|_2}, \tag{6.3}$$

where $\Delta_N(\mu)$ is the error estimator and $\|u_{\mathcal{N}}(\mu) - u_N(\mu)\|_2$ is the true error. The effectivity index in (6.3) is required to be ≥ 1 for rigorosity and as close as possible to unity for sharpness of the error estimator.

In Figure 6.3, we present the effectivity indices to demonstrate the quality and efficiency of the error estimator in Figure 6.2. Since they are of order $\mathcal{O}(10)$ at the final

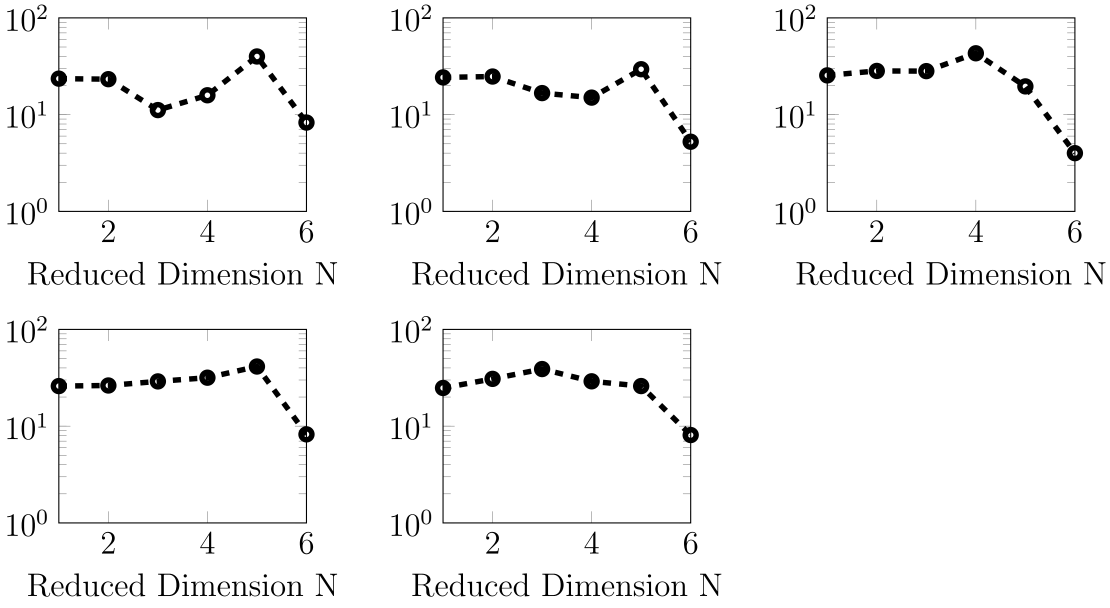


Figure 6.3.: Effectivity indices to demonstrate the quality of the error estimators in Figure 6.2, respectively.

RB dimension, we can claim that the error estimator is of good quality.

Figure 6.4 shows the error estimator and the true error of the finally constructed ROM over $\mu_i = \Xi$, for $i = 1, ..., 11$ samples for each protein as in Table 6.2, respectively. It is evident that the error estimator for the final RB approximations of dimension $N = 6$ is indeed an upper bound of the true error and a trend that both quantities behave similarly is clearly visible from the graphs. Consequently, the error estimators fall below the greedy tolerance of 10^{-3}.

Figure 6.5 is used to validate the true error in Figure 6.4, whereby 20 random values of the parameter domain D which are different from those in the training set Ξ are used. A common observation from these figures is that the true errors fall below $\mathcal{O}(10^{-4})$, which is approximmately an order of magnitude below the error estimator. The computational time taken to obtain the approximate solution $u_N(\mu)$ in the online phase is approximately 4.97×10^{-3} seconds on average, for any parameter $\mu \in D$.

Figure 6.6 demostrates the output error estimators for the generalized Born ion and some of the protein molecules which were introduced in Section 6.2.2. We observe that the dimension of the ROMs obtained by these output estimators are slighlty smaller than those obtained in Figure 6.2.

6.3.1. Runtimes and computational speed-ups

Before we dive into the runtimes of the various phases of the RBM, we would like to clarify some key notions of the two phases of the greedy algorithm, i.e., the offline and online phases, respectively. The offline phase is subdivided into two parts, the offline-offline phase, and the offline-online phase [59]. The offline-offline phase involves computation of the snapshots and pre-computing the parameter-independent quanti-

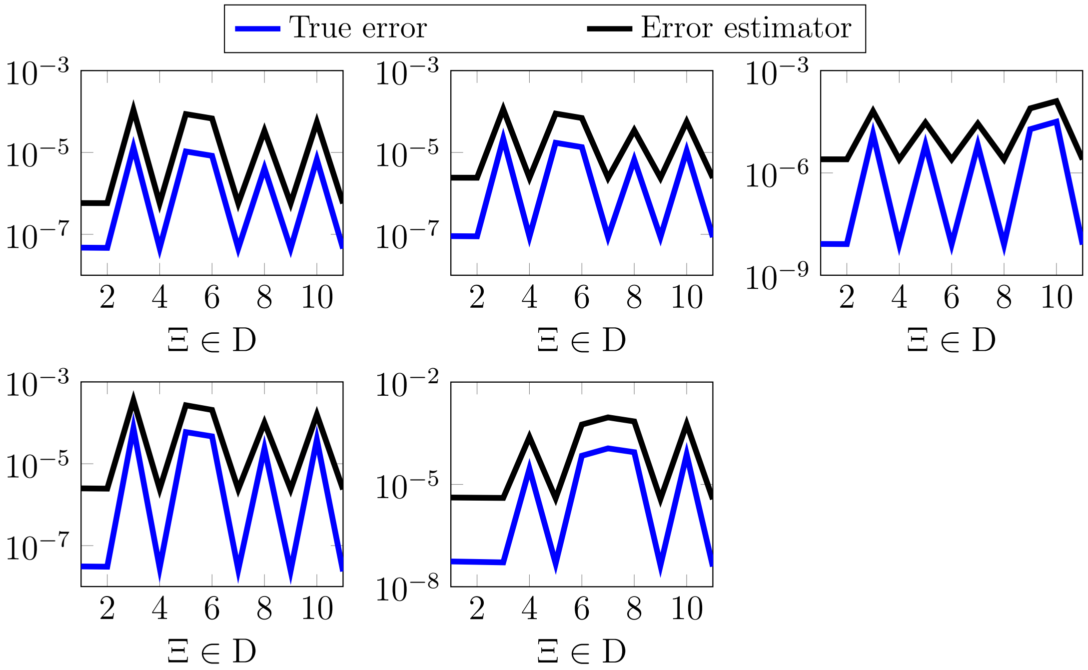


Figure 6.4.: Comparison of error estimator and true error for the final ROM for $\Xi \in D$ and for the proteins in Table 6.2, respectively.

Table 6.3.: Runtimes and speed-ups due to DEIM.

Runtime (seconds) and speed-up			
	Without DEIM	With DEIM	Speed-up
Offline-online phase	96.29	4.84	20
Assemble and solve ROM	8.36	9.91×10^{-3}	844

ties. The offline-online phase involves computation of the error estimator and the RB approximation. On the other hand, the pure online phase is where the final ROM has been constructed after the accuracy of the reduced basis is fulfilled, and is independent of the greedy algorithm. In this phase, the ROM can be solved for any parameter value in the parameter domain, including those which are different from the training set.

Table 6.3 shows the runtimes and computational speed-ups obtained with the use of DEIM approximation during the offline-online phase of the RBM at a single iteration of the greedy algorithm and with the use of the RBM in solving the linear system. We use a modest PC with the following specifications: Intel (R) Core (TM)2 Duo CPU E8400 @ 3.00GHz with 8GB RAM. In this section, the PBE is applied to the protein *fasciculin 1*.

Table 6.4 shows the runtimes of computing the FOM and the ROM at a given parameter value, respectively. The runtimes at different phases of the RBM are also presented. Speed-up factors induced by solving the ROM are listed to visualize the big difference between the FOM and the ROM. The ROM is much faster and takes a split second to assemble and solve for any parameter value. In the offline phase of the RBM,

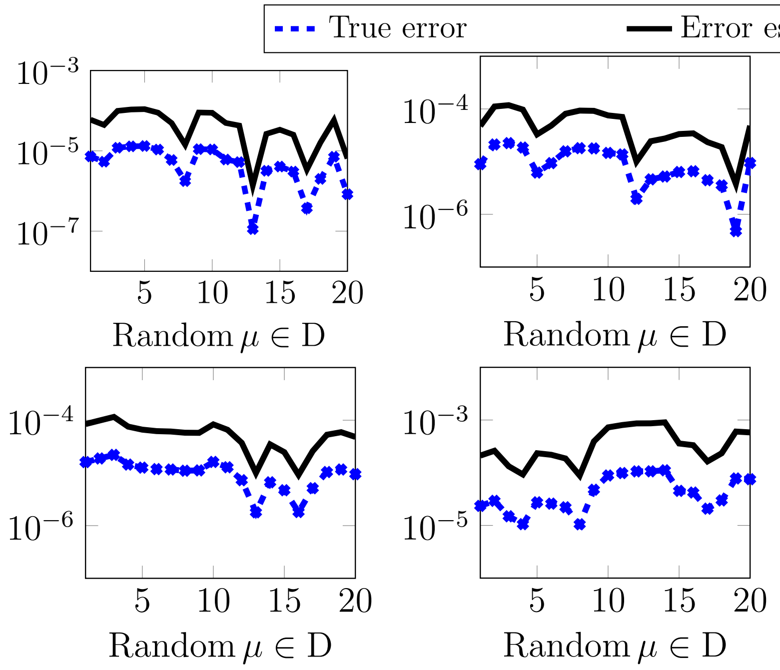


Figure 6.5.: Comparison between the true error $\|u^{\mathcal{N}}(\mu) - u_N(\mu)\|_2$ and the error estimator at the final ROM for random parameters $\mu \in D$ for the proteins in Table 6.2, respectively.

which comprises the greedy algorithm, the dominating cost is that of solving the linear system of the FOM by AGMG (i.e., computing the snapshots) at every iteration of the greedy algorithm. Miscellaneous in this case refers to the runtime to initialize the FDM, including assembling the FOM. The total RBM runtime includes the miscellaneous and offline runtimes.

Table 6.5 shows the runtimes of APBS and RBM for solving the FOM and the ROM at any given parameter value, respectively. The speed-up factor of RBM w.r.t. the APBS is also shown for different numbers of parameter values. It is evident that RBM is much more efficient than APBS when solving the system for many input parameter values (i.e. in a multi-query context). This is because we only need to solve a small system of order $N = 6$ once the final ROM model has been constructed which takes approximately 9.91×10^{-3} seconds for each parameter value, whereas APBS solves the FOM besides the initial system setup.

In a nutshell, to solve the LPBE for any parameter value with APBS, it takes 22.893 seconds, because the solver has to reconstruct the linear system. This implies that it takes approximately $2,289.3$ seconds to compute the potential for 100 parameter values (neglecting the runtime to modify the input files). This is more expensive than the total RBM time of 96.12 seconds. On the other hand, it takes the RBM approximately 9.91×10^{-1} seconds to solve the ROM of the LPBE for the same number of parameters values (i.e., 100).

The RBM only solves the FOM N times during the expensive offline phase as stated in Algorithm 5.4. Moreover, the RBM utilizes the precomputed system matrices and vectors and only solves the ROM for the new parameter value, thus saving a significant

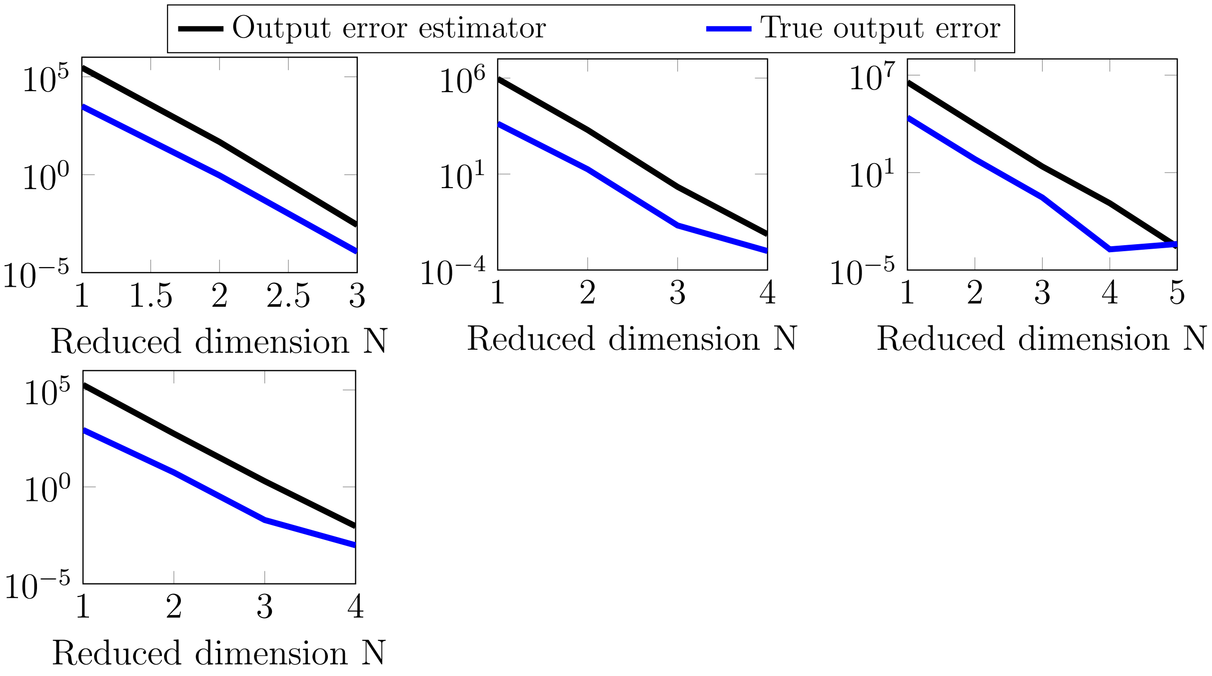


Figure 6.6.: Comparison of error estimator and true output error for the Born ion, proteins *fasciculin 1*, a 180-residue cytokine solution NMR structure of a murine-human chimera of leukemia inhibitory factor, and a minimized FKBP protein, respectively.

amount of computational costs during the online phase. This efficient implementation of a new mathematical approach to solve the PBE holds great promise towards reducing computational costs in a multi-query scenario and molecular dynamics simulation.

6.4. Numerical results for the regularized PBE

In this section, we first consider the free space electrostatic potential computed by the modified Poisson equation PE and the RS tensor format based splitting scheme for the Newton kernel. We compare the results with those of the traditional PE for various biomolecules. In this case, the PBE can be reduced to the PE by considering the zero ionic strength which implies that the function $\bar{\kappa}^2(\bar{x}) = 0$, hence the Boltzmann distribution term in (2.15) is annihilated. Consequently, homogeneous dielectric constants of $\epsilon_m = \epsilon_s = 1$ are considered. Then we consider the numerical tests for the regularized linear and nonlinear PBE such that $\bar{\kappa}^2(\bar{x}) > 0$ and $\epsilon_s/\epsilon_m \approx 40$.

We compute the electrostatic potentials using $n \times n \times n$ 3D Cartesian grids, in a box $[-b, b]^3$ with equal step size $h = 2b/(n+1)$. Conventional computations by the PBE/PE solver are limited to $n = 257$, on a PC with 8GB RAM due to the storage needs of the order of $\mathcal{O}(n^3)$.

Table 6.4.: Runtimes and speed-ups for FOM, ROM and RBM.

Runtime (seconds) and speed-up			
	FOM	ROM	Speed-up
Solve linear system	11.88	$4.97\times10-3$	7,616
Assemble and solve linear system	27.82	9.91×10^{-3}	5,500
Runtime (seconds) for RBM phases			
Miscellaneous	Offline	Online	Total RBM
10.58	85.54	9.91×10^{-3}	96.12

Table 6.5.: Runtimes for APBS and RBM.

Runtime (seconds) and speed-up for APBS and RBM			
No. of parameters	APBS	RBM	Speed-up
1	22.893	≈ 96.12	0.24
10	228.93	≈ 96.12	2.38
100	2,289.3	≈ 96.12	24
1000	22,893	≈ 106.12	215.75

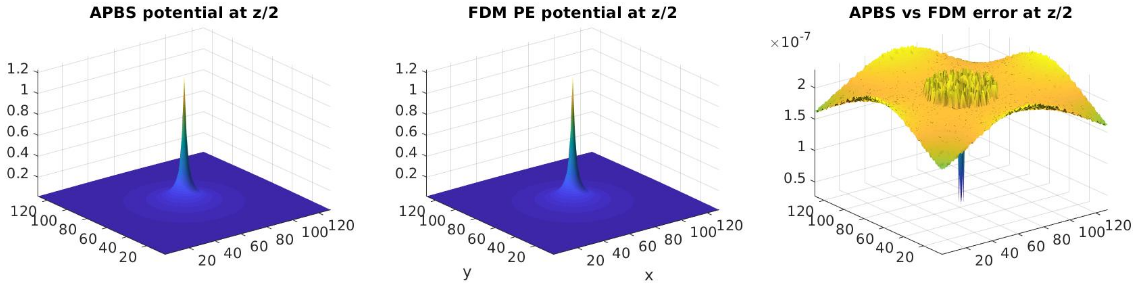


Figure 6.7.: The free space potential for the Born ion computed by the APBS (left), the FDM solver (middle) and the corresponding error (right).

6.4.1. Validating the accuracy of RS splitting for the free space potential

First, we validate the FDM solver for the PE by comparing its solution with that of the adaptive Poisson-Boltzmann solver (APBS) using the finite difference multigrid calculations with PMG [2] option [8, 30]. Here, we consider a Born ion of unit charge, unit (Å) radius and located at the origin ($(0,0,0)$). Figure 6.7 shows the single Newton kernel for the Born ion, approximated by the PE on an $n\times n$ grid surface with $n = 129$ at the cross section of the volume box in the middle of the z-axis computed by the APBS and the FDM solver and the corresponding error between the two solutions. The results show that the FDM solver provides as accurate results as those of the APBS with a

---

[2] PMG is a Parallel algebraic Multigrid code for General semilinear elliptic equations.

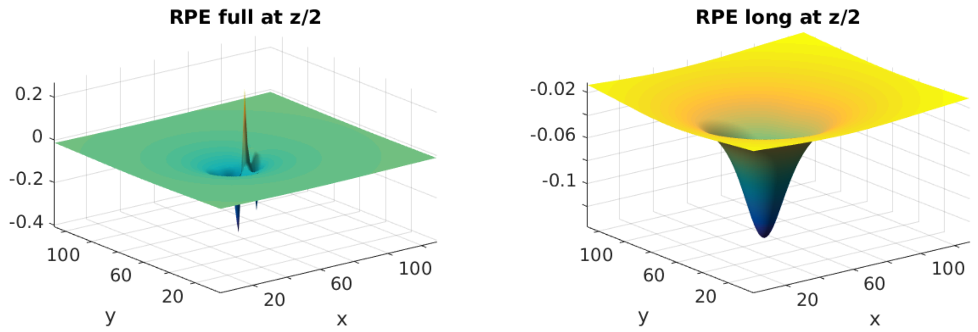


Figure 6.9.: The total free space electrostatic potential (left) and its long-range component (right) computed by the regularized PE.

discrete L_2 error of $\mathcal{O}(10^{-7})$ in the full solution. Similar results for varying proteins are illustrated in [16].

Secondly, we compare the accuracy of the traditional PE model and of the PE model modified by the RS tensor format for the approximation of the single Newton kernel. Figure 6.8 shows the single Newton kernel on an $n \times n$ grid surface with $n = 129$ at the cross section of the volume box in the middle of the z-axis computed by the canonical tensor approximation obtained by sinc-quadratures and the corresponding errors by the traditional PE model and the modified PE model computed by the FDM solver. We notice that the solution of the modified PE model is of higher accuracy than that of the traditional PE model because it captures the singularities exactly, see the central region of Figure 6.8 (right).

In a similar vein, we consider the *acetazolamide* compound consisting of 18 atoms and determine the accuracy of the traditional PE model vis a vis the PE model modified by the RS tensor format. Figure 6.9 shows the total free space electrostatic potential computed by the regularized PE and its long-range component, respectively, while Figure 6.10 shows the error between the exact Newton potential and the classical PE (left) and the regularized PE (right) computed on an $n \times n$ grid surface with $n = 129$. It is clearly shown that the modified PE model provides highly accurate solutions as compared to those of the traditional PE due to the accurate treatment of the solution singularities by the RS tensor format.

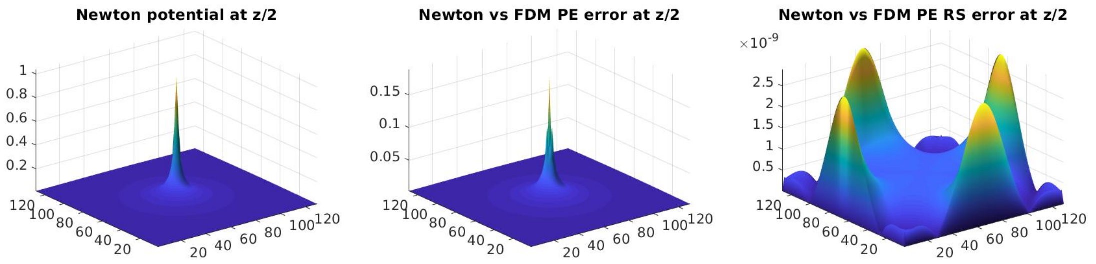


Figure 6.8.: The Newton potential computed by the canonical tensor decomposition (left), the error of its computation on the same grid by using the classical PE (middle) and by the modified PE (right).

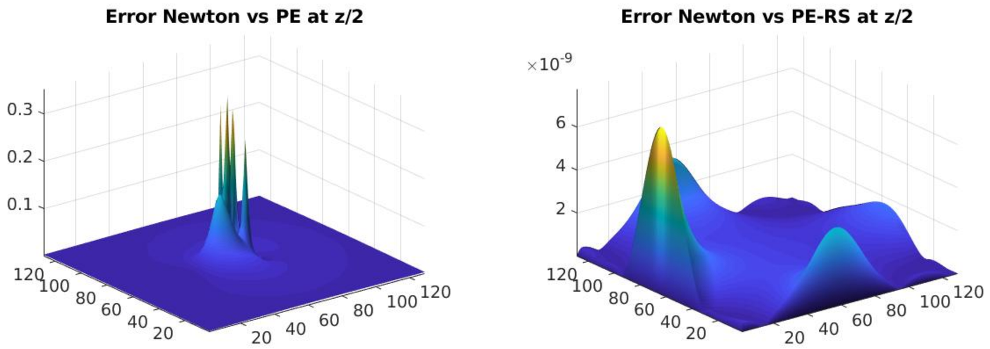


Figure 6.10.: The error of the free space potential between the exact Newton potential sums and the classical PE solution (left) and between the Newton potential sums and the regularized PE solution (right).

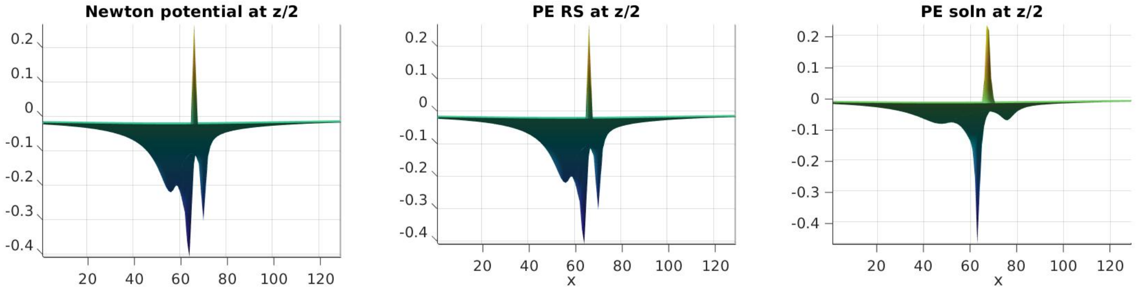


Figure 6.11.: Demonstration of the solution singularities for the *acetazolamide* molecule captured by the canonical tensor approximation (left), by the modified PE model (middle) and by the classical PE (right).

We notice, in addition, that the classical PE model does not capture accurately the singularities in the electrostatic potential due to the numerical errors introduced by the Dirac delta distribution and partly due to the smoothing effect caused by the spline interpolation of the charges onto the grid. The modified PE model, on the other hand, is able to capture the singularities due to the independent treatment of the singularities by the RS tensor technique. This is clearly demonstrated in Figure 6.11.

6.4.2. The regularized Poisson equation (PE) on a sequence of fine grids

Here, we illustrate the accuracy of the modified PE by calculating the free space electrostatic potential on a sequence of fine grids and compare with the solution of the exact Newton potential determined by the canonical tensor representation. We first consider the aforementioned Born ion case and show the absolute error for the finest Cartesian grid and the discrete L_2 norm of the error for a sequence of Cartesian grids. Figure 6.12 shows the absolute error of $\mathcal{O}(10^{-11})$ obtained on a 257^3 Cartesian grid and 32 Å box length. Table 6.6 shows the decay of the discrete L_2 norm of the error for a sequence of grid refinements.

Next, we consider the *acetazolamide* molecule with 18 atoms. This molecule is used as

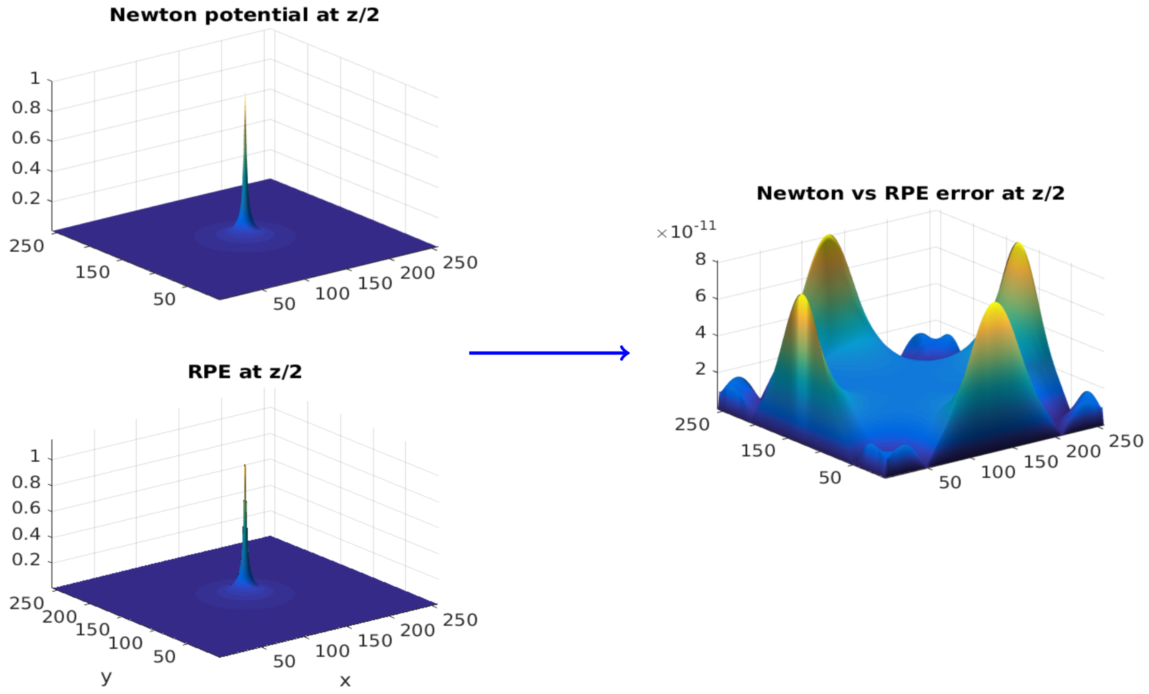


Figure 6.12.: Absolute error between the solutions of the Newton potential and the modified Poisson equation for the Born ion.

n	97	129	257
Discrete L_2 norm	4.1176×10^{-7}	9.5516×10^{-8}	2.7975×10^{-9}

Table 6.6.: The discrete L_2 norm of the error with respect to grid size for the Born ion.

a ligand in the human carbonic anhydrase (hca) protein-ligand complex for the calculation of the binding energy in the adaptive Poisson-Boltzmann software (APBS) package [64] and the MATLAB program for biomolecular electrostatic calculations, (MPBEC) [140]. The electrostatic potential is computed as in the previous case, employing the same grid properties. The absolute error is shown in Figure 6.13 and the error behavior with respect to mesh refinements is shown in Table 6.7.

Finally, we consider the protein *fasciculin 1*, with 1228 atoms, an anti-acetylcholinesterase toxin from the green mamba snake venom [104]. Again, we compute the electrostatics potential as in the previous case, but with 60 Å box length and a 257^3 as the minimum Cartesian grid because of a larger molecular size. The results provided in Figure 6.14 and Table 6.8 illustrate a similar trend of accuracy as in the previous test examples.

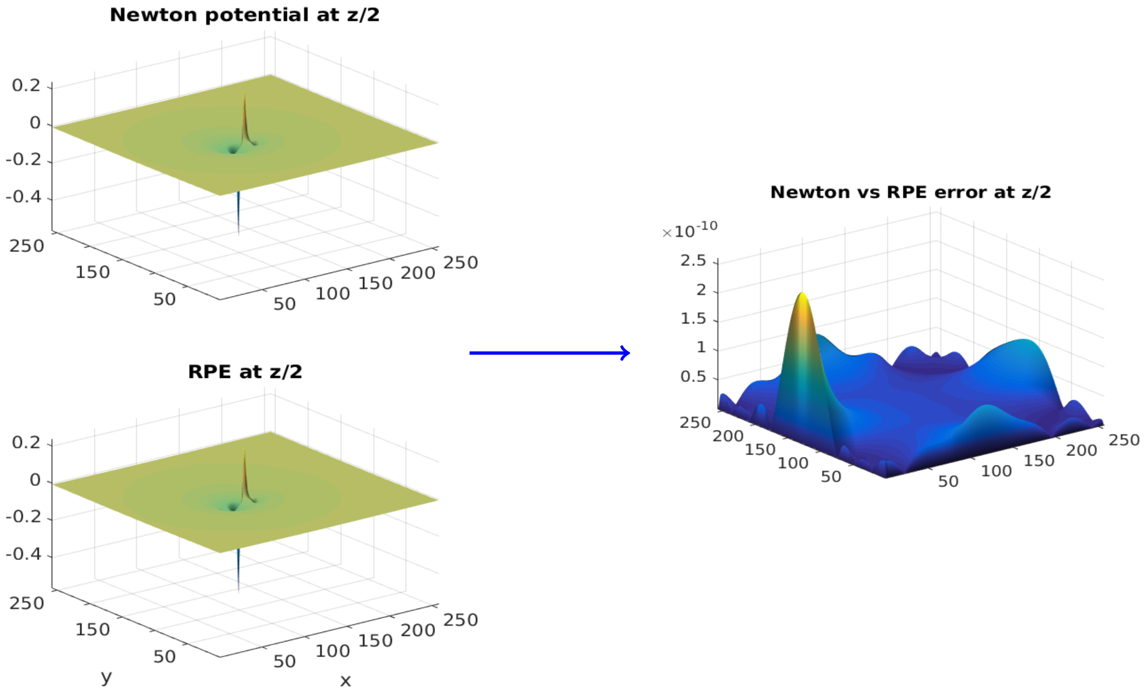


Figure 6.13.: Absolute error between the solutions of the Newton potential sums and the modified Poisson equation for the *acetazolamide* molecule.

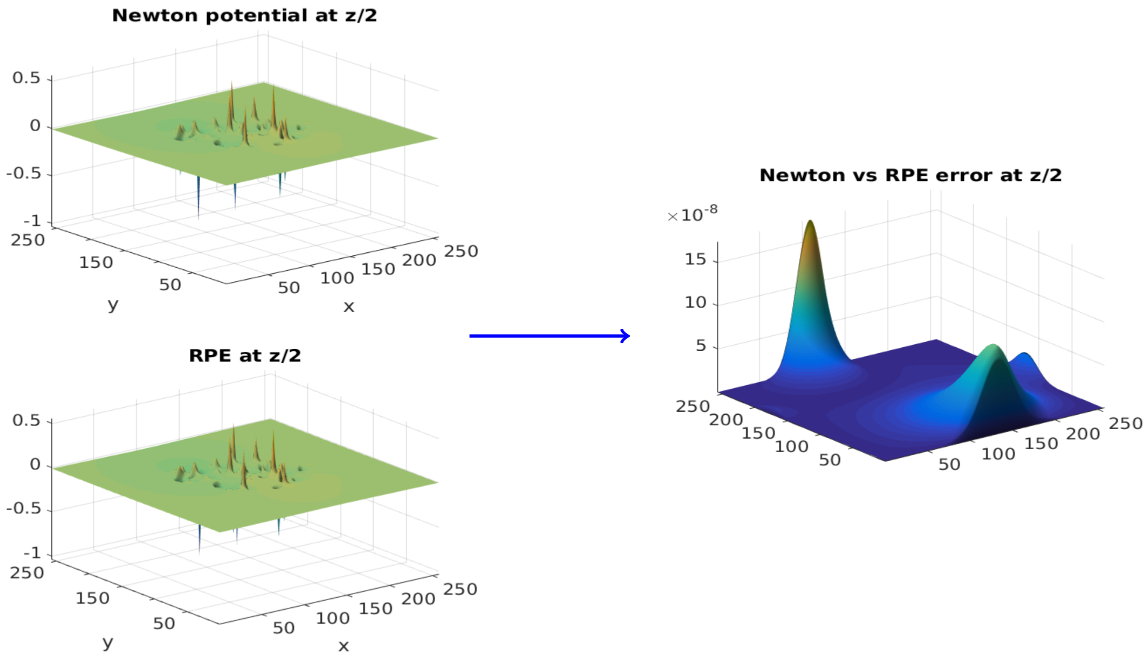


Figure 6.14.: Absolute error between the solutions of the Newton potential sums and the modified Poisson equation for the protein *fasciculin 1*.

n	97	129	257
Discrete L_2 norm	4.1176×10^{-7}	1.1936×10^{-7}	3.7003×10^{-9}

Table 6.7.: The discrete L_2 norm of the error and the relative error with respect to grid size for the *acetazolamide* molecule.

n	129	193	257
Discrete L_2 norm	1.2919×10^{-6}	1.7395×10^{-7}	4.3060×10^{-8}

Table 6.8.: The discrete L_2 norm of the error and the relative error with respect to grid size for the protein *fasciculin 1*.

6.4.3. Accurate representation of the long-range electrostatic potential by the RS tensor format

Here, we highlight the advantages of the RS tensor format in the low-rank approximation of the long-range component in the total potential sum. For this purpose, the RHOSVD within the multigrid C2T transform [82] is used which provides computation of the low-rank canonical/Tucker tensor representation of the long-range part at the asymptotic cost of $\mathcal{O}(Nn)$. Here, N is the number of charges in the molecule while n represents the grid dimension in a single direction.

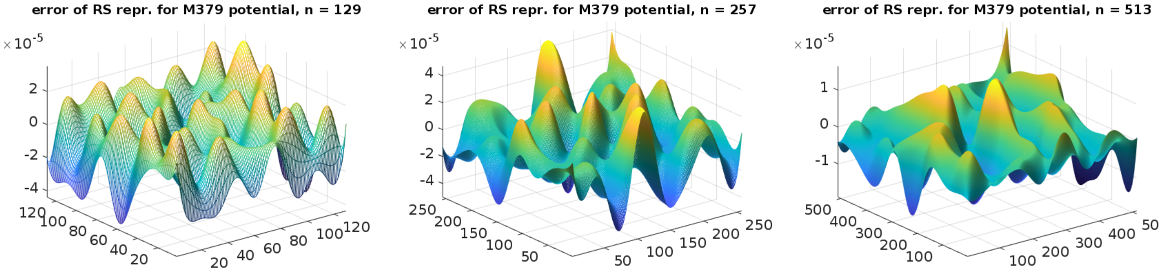


Figure 6.15.: The error due to the low-rank approximation of the long-range component for the 379 atomic molecule for $n = 129^3$ (left), $n = 257^3$ (middle) and $n = 513^3$ (right) grids.

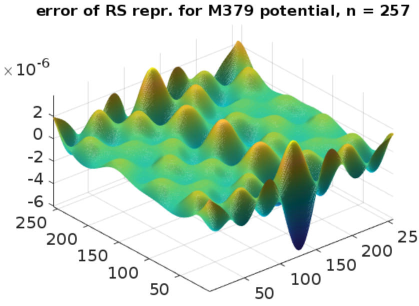


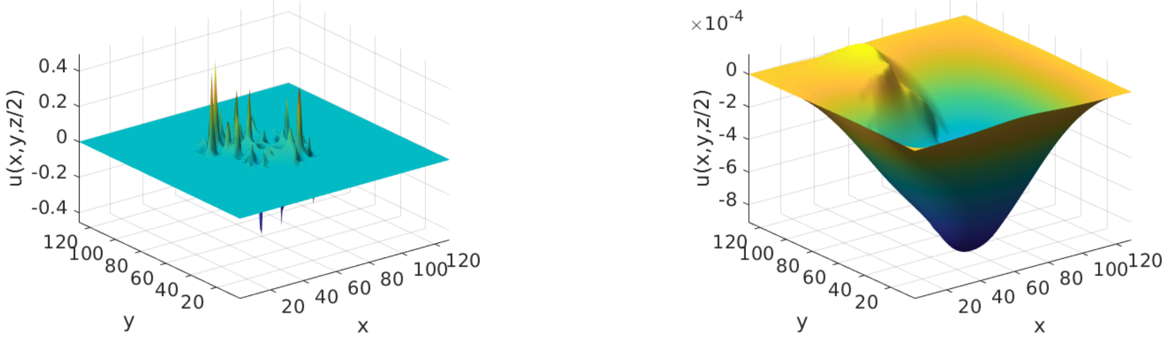
Figure 6.16.: The error due to the low-rank approximation of the long-range component for the 379 atomic molecule for $n = 257^3$ grid at a lower tolerance.

Figure 6.15 shows an error of $\mathcal{O}(10^{-5})$ of the RS tensor format approximation compared with the full size representation at various grid dimensions. These data correspond to the long-range RS-rank equal to 10, with ε-truncation threshold chosen as $\mathcal{O}(10^{-6})$ for the reference Newton kernel and $\mathcal{O}(10^{-7})$ for the C2T transform.

Figure 6.16 shows that the error is reduced by one order of magnitude, i.e. to $\mathcal{O}(10^{-6})$, if we take stronger rank truncation criterion ε of an order less for both the Newton kernel (i.e., $\mathcal{O}(10^{-7})$) and for the C2T transform, $\mathcal{O}(10^{-8})$.

6.4.4. Finite difference results for the classical LPBE

In this section, we consider $n^{\otimes 3}$ 3D uniform Cartesian grids, in a box $[-b, b]^3$ with equal step size $h = 2b/(n-1)$ for computing the electrostatic potentials of the PBE on a modest PC with the following specifications: Intel (R) Core (TM) $i7-4790$ CPU @ 3.60GHz with 8GB RAM. The FDM is used to discretize the PBE in this section and the numerical computations are implemented in the MATLAB software, version R2017b.



(a) LPBE solution by the FDM solver.

(b) APBS vs FDM solution error.

Figure 6.17.: The electrostatic potential for the protein *fasciculin 1* computed by the FDM solver (left) and the error between the APBS and FDM solutions (right) on $129^{\otimes 3}$ grid, at $0.15M$ ionic strength.

First, we validate our FDM solver for the classical LPBE by comparing its solution with that of the APBS software package (version 1.5-linux64), which uses the multigrid

(PMG) accelerated FDM [8]. Here, we consider the protein *fasciculin 1*, with 1228 atoms. The Figure 6.17 shows the electrostatic potential of the PBE on a $n \times n$ grid surface with $n = 129$ at the cross-section of the volume box (60 Å) in the middle of the z-axis computed by the FDM solver and the corresponding error between the two solutions. Here, we use the ionic strength of $0.15M$. The results show that the FDM solver provides accurate results as those of the APBS with a discrete L_2 error of $\mathcal{O}(10^{-4})$ in the full solution.

The results for solvation free energy of protein varieties are presented in [16]. To validate the claim in Remark 6.2, we provide in the Table 6.9, the comparison between the total electrostatic potential energies ΔG_{elec} in kJ/mol, between the LPBE and the nonlinear PBE (NPBE) computations on a sequence of fine grids using the APBS software package.

Remark 6.2:

We reiterate that the solutions obtained from the LPBE and the nonlinear PBE are very close to each other, even when the linearization condition does not hold [48]. This is especially manifested in protein molecules whose charge densities are small. However, in biomolecules with large charge densities, for example, the DNA, significant differences might be observed at the solute-solvent interface [47, 48]. Moreover, the solution of the LPBE is usually used as the initial condition for the nonlinear PBE. ◊

h	$\mathcal{N}$	ΔG_{elec}, LPBE	ΔG_{elec}, NPBE	Relative error
0.465	129^3	91 228.0575	91 227.8354	2.44×10^{-6}
0.375	161^3	130 606.0630	130 605.8448	1.67×10^{-6}
0.320	193^3	170 154.4401	170 154.1862	1.49×10^{-6}

Table 6.9.: Comparison of the total electrostatic potential energies ΔG_{elec} in kJ/mol, between the LPBE and the NPBE computations on a sequence of fine grids for the protein *fasciculin 1*.

Remark 6.3:

Notice from the Table 6.9 that the electrostatic potential energies ΔG_{elec} increase with decreasing grid/mesh size, h. This is caused by the short-range electrostatic potential behaviour as the distance, $\|\bar{x}\| \to 0$, in $1/\|\bar{x}\|$. ◊

6.4.5. Numerical tests for linearized RPBE (LRPBE)

Here, we first demonstrate the change in the electrostatic potential between the Poisson equation, which solves for the free space electrostatic potential, and the LRPBE which solves for the electrostatic potential in a medium consisting of jumping dielectric coefficients.

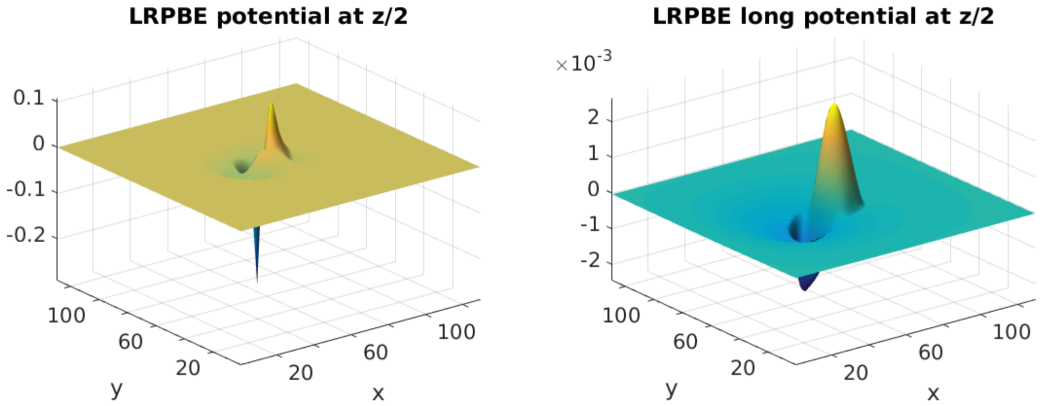


Figure 6.18.: The full electrostatic potential (left) and its long-range component (right) computed by the LRPBE at 0.15 ionic strength for the *acetazolamide* molecule.

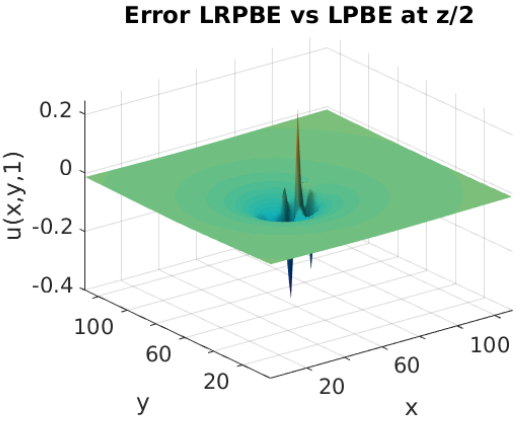


Figure 6.19.: The difference between the potential from linear PBE compared with that calculated by the new RS regularized scheme at 0.15 ionic strength for the *acetazolamide* molecule.

Figure 6.18 illustrates the full electrostatic potential computed for the *acetazolamide* molecule using the linear RS-regularized PBE (left) computed by using the RS tensor decomposition for the Dirac delta, and the long-range part of this potential (right).

Figure 6.19 shows the difference between the potential from linear PBE compared with that calculated by the new RS regularized scheme, for the case of 0.15 ionic strength. We observe that the computational error for the traditional regularization scheme indicated in Figure 6.10 (left) for the case of free space collective potential is almost of the same order of magnitude as that presented in Figure 6.19 for the case of linearized PBE.

Mesh size h	Number of mesh points	Discrete L^2 error
2.5×10^{-1}	110 592	5.11×10^{-2}
1.25×10^{-1}	884 736	1.25×10^{-2}
6.25×10^{-2}	7 077 888	3.04×10^{-3}
3.125×10^{-2}	56 623 104	7.32×10^{-4}

Table 6.10.: Discrete L^2 norm of the error between the solution to the LRPBE in (4.16) and the analytical solution in (6.4) for a Born ion of unit charge and radius $R = 3$ on a sequence of fine grids.

Remark 6.4:

We demonstrate the conspicuous difference between the RPE and the LPBE solutions in Figure 6.9 and Figure 6.18, respectively, due to the effect of the inhomogeneous (or jumping) dielectric coefficient and the ionic strength in the latter. As a result of these coefficients, we can see that the short-range potential component of the LPBE in Figure 6.18 (left) is halfed due to the change of ϵ_m from 1 in free space to 2 in the solvated state, while the long-range potential component in Figure 6.9 scales by approximately 1/80 in addition to the effect of the ionic strength as a damping coefficient. ◊

6.4.5.1. Validation test for the LRPBE

To validate the RS tensor based LRPBE solver, we consider the analytical solution of the Born ball model similar to that in [150]. However, in our model, we use the centimeter-gram-second (CGS) units, see [67], instead of the SI (International standard) units, in addition to the scaling differences for the constant 4π,

$$\psi(\bar{x}) = \begin{cases} \frac{\alpha z}{\epsilon_m d} + \frac{\alpha z}{R}\left(\frac{1}{\epsilon_s} - \frac{1}{\epsilon_m}\right), & \text{if} \quad \bar{x} \in \Omega_m, \\ \frac{\alpha z}{\epsilon_s d}, & \text{if} \quad \bar{x} \in \Omega_s, \end{cases} \tag{6.4}$$

where $\Omega = \Omega_m \cup \Omega_s = [-6, 6]^3$, $d = \|\bar{x}\|$, $\alpha = 4\pi^2 e_c \times 10^8$, $\epsilon_m = 2$, $\epsilon_s = 78.54$, $z = 1$ is a unit charge, and $R = 3$ is the atomic radius of the Born ion.

The Table 6.10 illustrates the validation results for our LRPBE solver for the Born ball model on a sequence of fine grids. The discrete L^2 norm of the error demonstrates good convergence of our LRPBE model. Note that Editha, one of the compute clusters at the MPI in Magdeburg, was used to carry out the last computation in Table 6.10 due to its large memory requirement, i.e., 56 623 104 mesh points.

Remark 6.5:

A more complex validation test would be the Kirkwood model [88] in which a single sphere containing multiple distributed point charges is considered. However, due to

Atomic radii in Å							
Atomic radius	0.00	1.00	1.40	1.50	1.70	1.85	2.00
Number of atoms	322	333	195	82	104	10	182

Table 6.11.: Atomic radii and the corresponding number of atoms for the constituent atoms of the protein *fasciculin 1*.

the unrealistic treatment of the atoms (i.e. without atomic radii), the RS tensor format cannot be applied to this case. Note that the RS tensor approach requires explicit treatment of atoms in a molecule with non-intersecting atomic radii [17]. ♢

6.4.6. Accuracy of the nonlinear RPBE based on the RS tensor format

Here, we provide the results for the calculation of electrostatic potential for the nonlinear RPBE (NRPBE) based on the RS tensor format and compare the results with those of the traditional NPBE for various proteins. This is because the nonlinear RPBE implementation in the APBS package does not currently function for reasons not specified. On the other hand, the LRPBE variant is not optimized for the APBS package and users are recommended to use the multigrid-based methods (which use PMG) for improved performance. Therefore, we cannot compare the results for the regularized PBE models between APBS and the RS tensor-based model.

First, we consider the protein *fasciculin 1* consisting of 1228 atoms of varying atomic radii as shown in the Table 6.11. Notice that 322 of the total atoms have zero radius (e.g. polar or non-polar hydrogen atoms), which implies that we must annihilate them from the RS tensor format calculations so that they are not assigned Newton kernels. Therefore, we consider the smallest atom in the protein as that with 1 Å radius, (i.e., the Hydrogen atom). Recall from the Figure 3.7 that the short-range component of the canonical vectors has effective support in a unit Å radius.

We provide the comparisons between the electrostatic potential of NRPBE, based on the RS tensor format, with that of the traditional NPBE. The Figure 6.20 shows the solutions from the two models and the corresponding error on $129^{\otimes 3}$ uniform Cartesian grid and a 60Å domain length, at $0.15M$ ionic strength.

Remark 6.6:

Notice that the error is predominant within the molecular region, where the solution is singular. However, in the solute region, which is dominated by the long-range regime, the error is small, in $\mathcal{O}(10^{-5})$. ♢

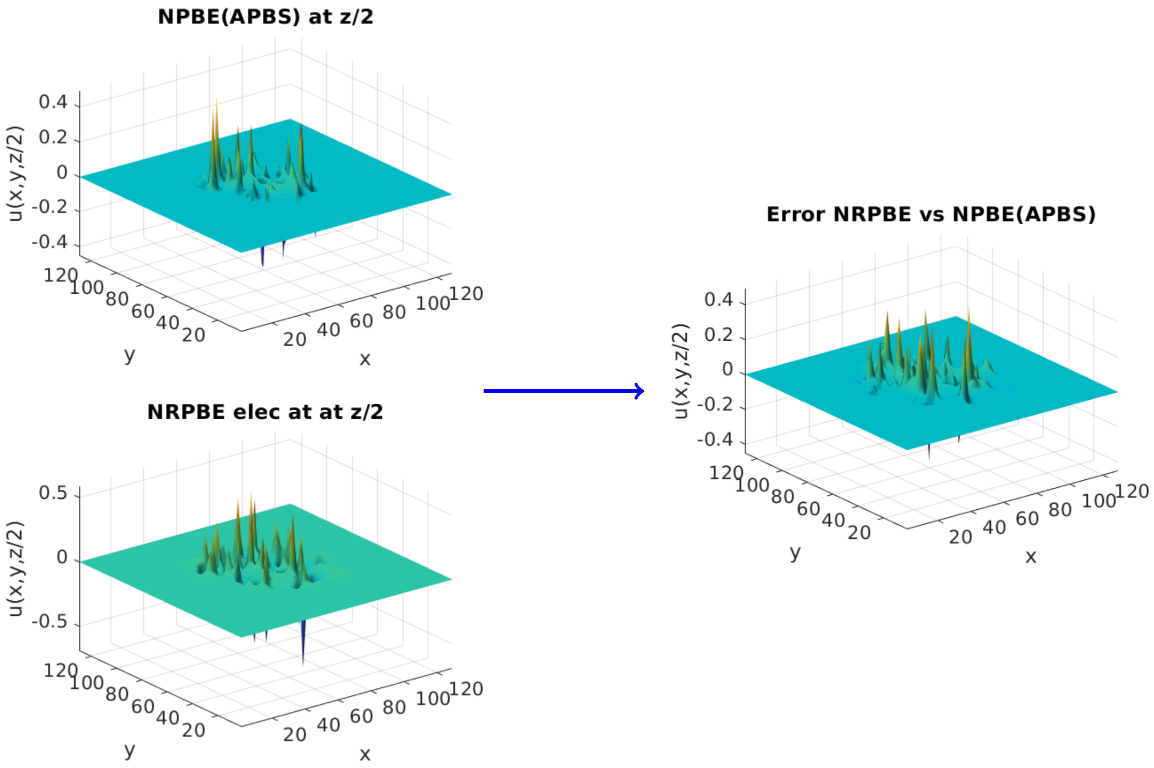


Figure 6.20.: Absolute error between the solutions of the traditional NPBE and the NRPBE for the protein *fasciculin 1*.

The Figure 6.21 provides the cross-sectional view of the electrostatic potential shown in the Figure 6.20 in order to demonstrate the accuracy of the solution singularities inherent in the NRPBE model over the traditional NPBE model. Notice that the NRPBE is capable of capturing exactly, the short-range component of the total potential sum because this part is precomputed analytically thereby avoiding the numerical errors occasioned by the traditional NPBE.

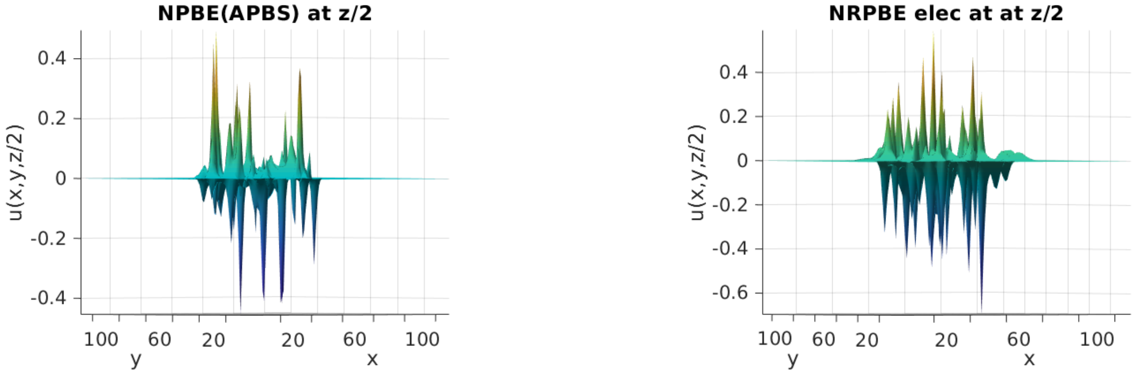
Remark 6.7:

The Figure 6.21b contains densely populated singularities/cusps as a result of explicit treatment of each atomic charge by the Newton kernel, whereas the Figure 6.21a, as a result of the smoothing/smearing effect of the atomic charges by the cubic spline interpolation, contains sparsely populated singularities most of which are not sharp.$\Diamond$

Secondly, we provide results for a 180-residue cytokine solution NMR structure of a murine-human chimera of leukemia inhibitory factor (LIF) [61] consisting of 2809 atoms. The corresponding variation in atomic radii and their frequencies are shown in Table 6.12. Figure 6.22 shows the comparison between the electrostatic potential of NRPBE, with that of the classical NPBE and the corresponding error on a $129^{\otimes 3}$ grid and a 65Å domain length, at $0.15M$ ionic strength.

Atomic radii in Å							
Atomic radius	0.2245	0.4500	0.9000	1.3200	1.3582	1.4680	≥ 1.7000
Frequency	315	6	6	1032	54	6	1390

Table 6.12.: Atomic radii and frequencies for the constituent atoms of a 180-residue cytokine solution NMR structure of a murine-human chimera of leukemia inhibitory factor (LIF).



(a) Cross-sectional view of NPBE solution in the Figure 6.20.

(b) Cross-sectional view of NRPBE solution in the Figure 6.20.

Figure 6.21.: The cross-sectional view of the electrostatic potentials in the Figure 6.20.

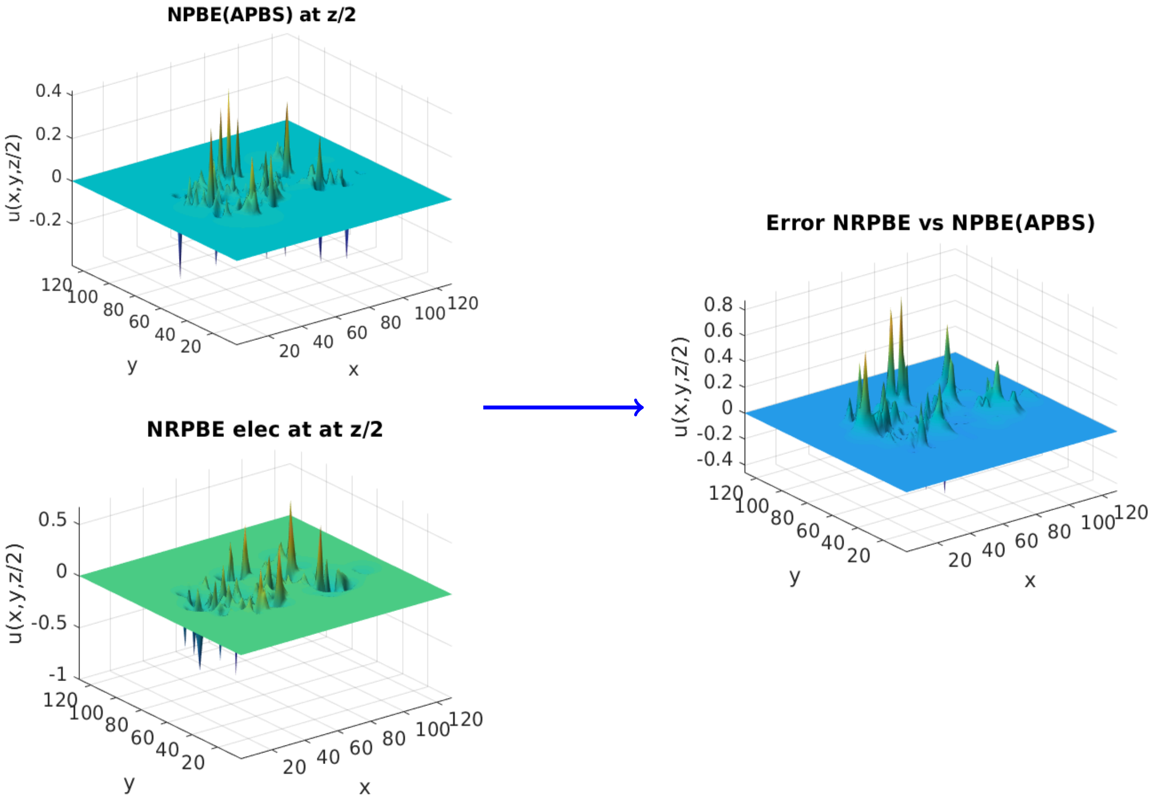
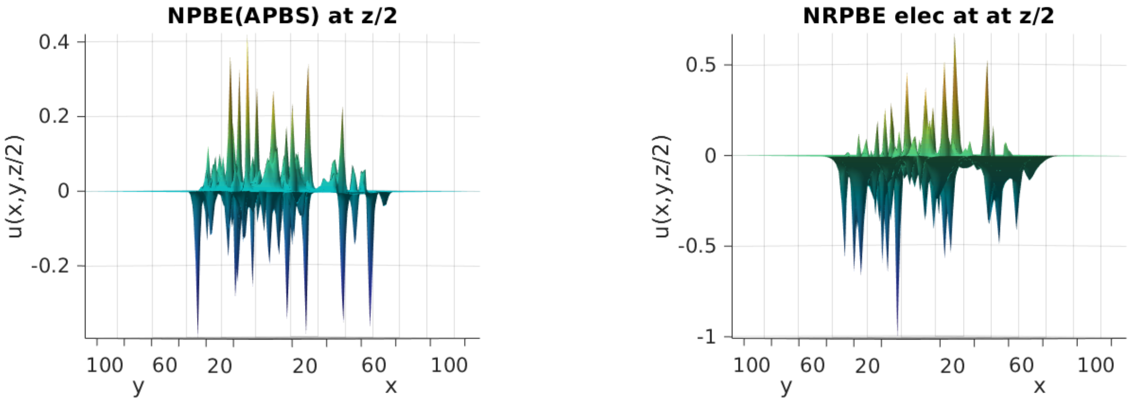


Figure 6.22.: Absolute error between the solutions of the traditional NPBE and the NRPBE for a 180-residue cytokine solution NMR structure of a murine-human chimera of leukemia inhibitory factor (LIF).



(a) Cross-sectional view of NPBE solution in the Figure 6.20.

(b) Cross-sectional view of NRPBE solution in the Figure 6.20.

Figure 6.23.: The cross-sectional view of the electrostatic potentials in the Figure 6.22.

Remark 6.8:

In a similar vein, we notice in the Figure 6.23 that the error is predominant within the molecular region, where the solution is singular. The scaling of the total potential demonstrates that the solution singularity in the NRPBE model is more accurately captured than in the traditional NPBE model. It is also worth mentioning that the small atomic radii, (< 0.9Å), in the Table 6.12 are treated independently in terms of the RS tensor splitting of the short- and long-range potentials. $\Diamond$

6.4.7. Runtimes and computational speed-ups

We compare the runtimes of computing both the classical and regularized PBE models in the Table 6.13 for the protein *fasciculin 1* in an $n^3 = 129^3$ domain of 60 Å length at an ionic strength of $0.15M$. Notice that the runtimes for the LPBE and the LRPBE are almost equal because the linear systems are solved by the same solver (i.e., AGMG). On the other hand, the runtime for the NRPBE is less compared to the NPBE because the former system converges (for all values of the ionic strength) in the AINSOLV solver when the residual and Jacobian of (4.28) are used as inputs.

Runtime (seconds) and speed-up			
	LPBE	LRPBE	Speed-up
Solve linear system	5.2608	6.3421	≈ 1
Total runtime	15.2507	16.4715	≈ 1
	NPBE	NRPBE	Speed-up
Solve nonlinear system	17.9663	6.3000	2.852
Total runtime	34.4014	20.6976	1.662

Table 6.13.: Runtimes and speed-ups for LPBE, LRPBE, NPBE, and NRPBE.

6.5. Reduced basis approximation for the NRPBE

Here, we determine the accuracy and computational efficacy of the RBM for the approximation of the truth high-fidelity solutions of the NRPBE for biomolecular modeling. We set the solute and solvent dielectric coefficients as $\epsilon_m = 2$, and $\epsilon_s = 78.54$, respectively, and employ the parameter values from the training set $\Xi \in D = [0.05, 0.15]$ with a sample size of $l = 11$, greedy tolerance $\text{tol} = 10^{-10}$ for Algorithm 5.4 to generate the projection matrix V_N.

In the numerical tests, the molecular charge density function (singular source term) for the classical NPBE and for the regularized Dirac density function for the NRPBE are obtained from PQR files which are generated from the following biomolecules with varying sizes that depend on the number of atoms:

(a) The *acetazolamide* molecule consisting of 18 atoms, which is used as a ligand in the human carbonic anhydrase (hca) protein-ligand complex for the calculation of the binding energy [64, 140].

(b) *fasciculin 1*, an anti-acetylcholinesterase toxin from green mamba snake venom [104] consisting of 1228 atoms.

(c) A 180-residue cytokine solution NMR structure of a murine-human chimera of leukemia inhibitory factor (LIF) [61] consisting of 2809 atoms.

Remark 6.9:

Since the solution of the PBE has a slow polynomial decay in $1/\|\bar{x}\|$, it is paramount that large domains, approximately 3-times the size of the biomolecule be used in order to accurately approximate the boundary conditions [67]. In this regard, we use domains of lengths $(32\text{Å})^3$, $(60\text{Å})^3$ and $(65\text{Å})^3$, respectively, for the aforementioned biomolecules. Here, Å denotes the angstrom unit of length. ◊

To begin with, we demonstrate the solution components of the full order model (FOM) of the NRPBE via the RS tensor format for the protein *fasciculin 1* in case (b), in a uniform Cartesian grid of $129^{\otimes 3}$ and a 60Å domain length. Figure 6.24 shows

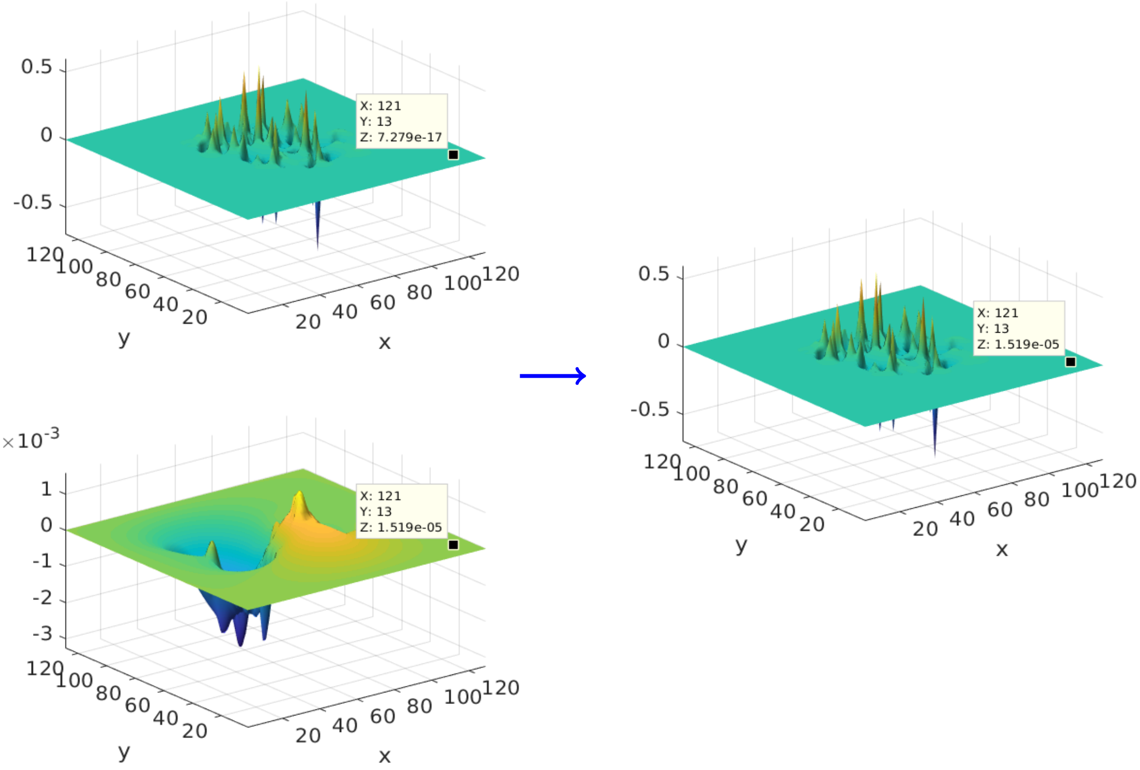


Figure 6.24.: The long-range (bottom left), short-range (top left), and total electrostatic potentials (right) for *fasciculin 1*.

the short- and long-range components of the target electrostatic potential, which are computed analytically from the CCT tensor (3.37), and numerically via the NRPBE in (4.16), respectively, and the corresponding total electrostatic potential.

The behaviour in the scaling among the three electrostatic potentials in Figure 6.24 shows that the total potential on the right-hand side inherits the largest potential value of 0.5 from the short-range component (top left), while the smallest value of $\mathcal{O}(10^{-5})$ is obtained from the long-range component (bottom left) towards the boundary of the domain.

Remark 6.10:

The main computational advantage of applying the RBM technique to the NRPBE is that the RB approximation is only applied to the smooth long-range component of the potential, see Figure 6.24 (bottom left), thereby avoiding the singularities inherent in the short-range component that are known to cause numerical difficulties. Hence the resultant RB approximation is expected to be of higher accuracy. ◇

Consider the NRPBE system generated by all the three cases (a), (b) and (c), in uniform Cartesian grids of $97^{\otimes 3}$ for case (a) and $129^{\otimes 3}$ for cases (b) and (c), respectively.

We show in Table 6.14, the decay of the maximal error estimator, defined as

$$\Delta_N^{\max}(\mu) = \max_{\mu \in D} \|r_N(\widehat{u}_N^r; \mu)\|_2,$$

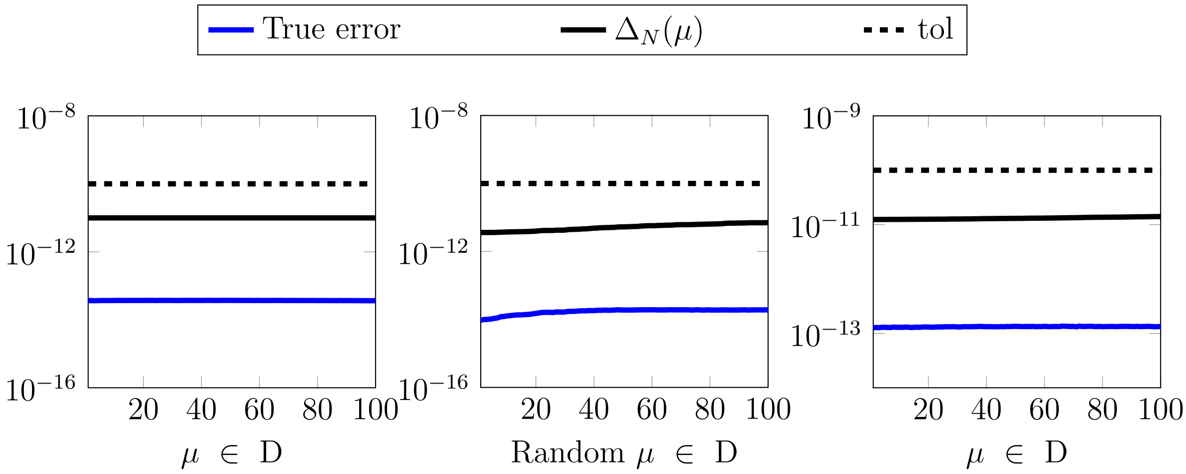
and the true error $\|u_N^r(\mu) - \widehat{u}_N^r(\mu)\|_2$, during the greedy algorithm at the current RB dimension $i = 1, \dots, N$ for all of these cases.

Note that from Table 6.14, the ROM provides highly accurate approximations, close to machine precision ($\mathcal{O}(10^{-15})$) for the NRPBE, for all the biomolecular cases as demonstrated by the true error in the second iteration. This is due to the smoothness of the long-range electrostatic potential, which enhances rapid and accurate model reduction process and facilitates, in general, low-rank approximation.

Next, we validate the final ROM at 100 random $\mu \in D$ in Figure 6.25. It is clear that the true error of the ROM is still below the tolerance for all 100 $\mu \in D$.

Biomolecule	Error at iteration 1		Error at iteration 2		N
	$\Delta_N^{\max}(\mu)$	True error	$\Delta_N^{\max}(\mu)$	True error	
Case (a)	5.06×10^{-6}	1.27×10^{-8}	3.03×10^{-12}	3.04×10^{-15}	2
Case (b)	1.07×10^{-5}	8.92×10^{-8}	3.69×10^{-12}	2.02×10^{-14}	2
Case (c)	3.26×10^{-5}	1.45×10^{-7}	2.06×10^{-11}	3.20×10^{-14}	2

Table 6.14.: The comparison between the maximal error estimator $\Delta_N^{\max}(\mu)$ and the true error for the NRPBE during the greedy iteration at the current RB dimension $i = 1, \dots, N$ for the biomolecules in cases (a) to (c).



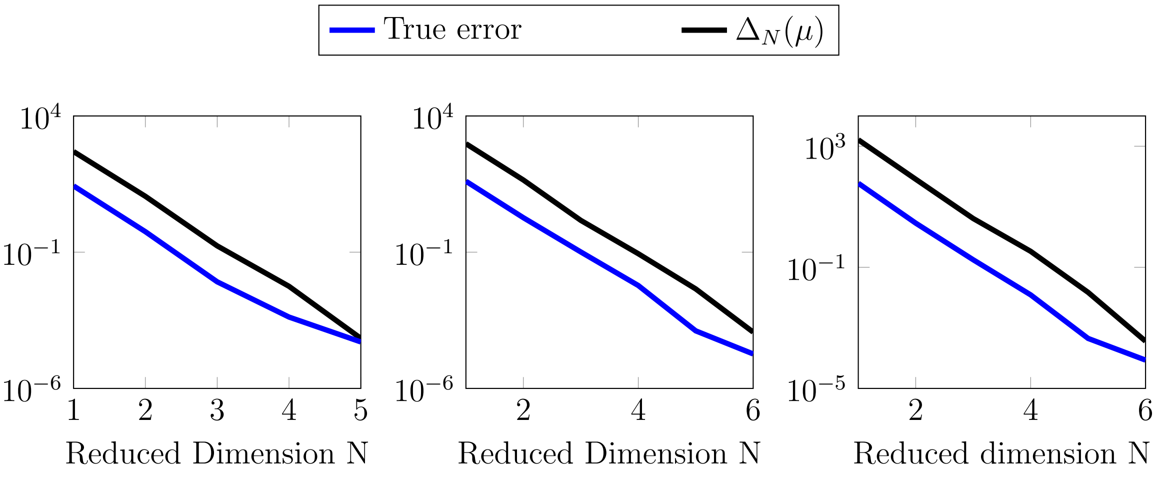
(a) Error for case (a). (b) Error for case (b). (c) Error for case (c).

Figure 6.25.: Comparison between the error estimator and the true error for the NRPBE for the cases (a) to (c) for the final ROM at 100 random (varying) parameter values $\mu \in D$.

6.5.1. Comparison of the RB approximation accuracy between the NRPBE and the NPBE

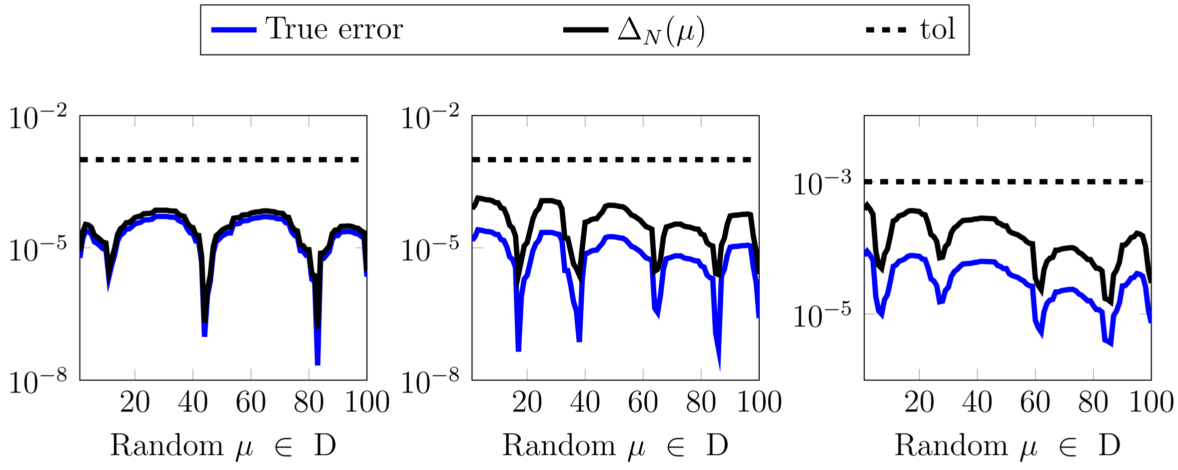
In this section, we demonstrate via the RB approximation, that the NRPBE model is more accurate and computationally efficient than the classical NPBE. In a similar style as in Section 6.5, we consider the biomolecules in cases (a) to (c) on the corresponding domain lengths and grid dimensions. We demonstrate the accuracy of the RB approximation for the classical NPBE model in order to compare it with the NRPBE model.

We begin by demonstrating in Figure 6.26, the comparison of the error decay between the maximal error estimator $\Delta_N^{\max}(\mu)$ and the true error for the classical NPBE during the greedy iteration at the current RB dimension $i = 1, \dots, N$ for the biomolecules in cases (a) to (c).



(a) Error for case (a). (b) Error for case (b). (c) Error for case (c).

Figure 6.26.: The comparison between the maximal error estimator $\Delta_N^{\max}(\mu)$ and the true error for the classical NPBE during the greedy iteration at the current RB dimension $i = 1, \dots, N$ for the biomolecules in cases (a) to (c).



(a) Error for case (a). (b) Error for case (b). (c) Error for case (c).

Figure 6.27.: Comparison between the error estimator and the true error for the classical NPBE for the cases (a) to (c) for the final ROM at 100 random (varying) parameter values $\mu \in D$.

We notice that the RBM constructs a small ROM (i.e., $N = 2$) of high accuracy ($\mathcal{O}(10^{-12})$) for the NRPBE in Table 6.14 because of the regularized nature of the model, the RBM applied to the classical NPBE, nevertheless, generates a ROM of dimension $N = 6$ at the accuracy of $\mathcal{O}(10^{-4})$ for most biomolecules in Figure 6.26 [16, 96]. This is because in the latter case, the short-range component of the electrostatic potential impedes the reduction process due to the sharp cusps or singularities which are hard to capture in the ROM. Furthermore, case (a) has a slightly smaller ROM dimension due to its small number of atoms as compared to the rest, hence its small number of solution singularities (cusps) to be captured in the ROM.

The accuracy of the RB approximation of the classical NPBE in Figure 6.27 is much lower than that of the NRPBE in Figure 6.25 due to the inaccurate approximation of the short-range component inherent in the former. This demonstrates the efficacy of the regularization scheme based on the RS tensor technique. The oscillations in the error in Figure 6.27 also justifies the irregularity of the singular solution, which impedes the model reduction process.

6.5.1.1. Runtimes and computational speed-ups

We compare the computational runtime of computing both the classical and regularized NPBE models as well as that of the corresponding ROM (using the RBM) in Table 6.15. The respective PBE models were applied to the protein *fasciculin 1*. Given a fixed value of the parameter μ, Table 6.15 compares the runtimes for solving the FOM (using the FDM), constructing the ROM (using the RBM), and solving the ROM (using direct methods), for the classical LPBE and NPBE with those of the regularized LRPBE and NRPBE, respectively. It is clear that the RBM spends more time in the offline phase of the greedy algorithm to compute snapshots for the classical NPBE than on the RPBE model, see Figure 6.26 and Table 6.14. This is mainly because of the presence of rapid

Runtime (seconds) for the PBE and the RPBE				
	LPBE	NPBE	LRPBE	NRPBE
FOM	17.68	34.40	22.83	28.30
RBM	107.98	238.78	90.04	76.38
ROM	2.22×10^{-2}	2.40×10^{-2}	2.10×10^{-3}	6.59×10^{-3}

Table 6.15.: Runtimes for the FOM, RBM and ROM for the linear and nonlinear variants of both the classical and the regularized PBE models.

Runtime (seconds) and speed-up using the FDM and the RBM			
No. of parameters	FOM for NRPBE	RBM for NRPBE	Speed-up
1	28.30	76.38	0.37
10	≈ 283	≈ 76.44	3.70
100	≈ 2830	≈ 77.04	36.73
1000	≈ 28300	≈ 82.97	341.09

Table 6.16.: Comparison of the runtimes and speed-ups between the FOM and the RBM for the NRPBE in a multiparameter context.

singularities in the PBE solution, which provides an onerous task in the construction of the ROM.

Consequently, Table 6.16, shows that solving the FOM for the NRPBE by the classical numerical techniques (in this case, the FDM) is sufficient and computationally efficient only for a single parameter value. However, for many varying parameter values, the RBM is more efficient because it constructs only a small ROM once, which can then be solved fast to obtain the solutions at any values of the parameter. For instance, for 1000 different parameter values, the ROM runtime is $\approx 6.59 \times 10^{-3}$sec, leading to a total runtime of $\approx$ 82.97sec to solve the NRPBE using the RBM technique instead of $\approx$ 28300sec by the FDM solver. Note that the runtimes for the 1000 varying parameter values for the FOM are mere approximations based on that of the single parameter value, since simulating the FOM for so many times is impractical.

6.6. Conclusions

In this chapter, we have presented the numerical approximation of both the classical and the regularized Poisson-Boltzmann equations and demonstrated that the latter is a very accurate model due to the avoidance of the rapid singularities in the numerical approximation. The reduced basis approximation also demonstrates higher accuracies and reduced computational costs (i.e., runtimes) for the regularized PBE model.

CHAPTER 7

CONCLUSIONS AND OUTLOOK

Contents

7.1. Conclusions

This thesis has been devoted to the efficient numerical solution of the Poisson-Boltzmann equation for calculation of electrostatics in large molecular systems. The main results of this work can be broadly categorized into two parts. First, we have demonstrated that the range-separated (RS) tensor format can be gainfully applied as a solution decomposition technique for the PBE by eliminating the numerical challenge arising from the strong singularities, described by the Dirac delta distributions. In this treatise, we emphasize that this problem can be efficiently circumvented by replacing the Dirac delta distribution with a smooth long-range function. Furthermore, as the only requirement in this approach is a simple modification of the singular charge density of the PBE in the molecular region Ω_m, it does not change the FEM/FDM system matrix.

The efficacy of the new tensor-based regularization scheme for the PBE is established on the unprecedented properties of the grid-based RS tensor splitting of the Dirac delta distribution. The main computational benefits are due to the localization of the modified right-hand side within the molecular region and automatic maintaining of the continuity of the Cauchy data on the solute-solvent interface. Another advantage is that our computational scheme only includes solving a single system of algebraic equations for the smooth long-range (i.e., regularized) part of the collective potential discretized by FDM. The total potential is obtained by adding this solution to the directly precomputed low-rank tensor representation for the short-range contribution, resulting in a total potential sum of high accuracy. The various numerical tests illustrate the main properties of the presented scheme. For example, it is demonstrated in the numerical results that the classical PBE model does not accurately capture the solution singularities which emanate from the short-range component of the total target electrostatic potential in the numerical approximation.

Secondly, we apply the reduced basis method (RBM) and the discrete empirical interpolation method (DEIM) to leverage this significant property of the regularized PBE by constructing a regularized ROM at extremely low computational costs as compared to that of the classical variant. This avoids constructing a ROM which comprises of the highly singular component of the electrostatic potential, thereby reducing the errors in the numerical approximation. The total potential is obtained by adding the regularized component (solution of the ROM), which is lifted (by projection) to the high-fidelity space, $\mathcal{N}$, to the directly precomputed canonical tensor representation of the short-range component of the Newton potential sum.

7.2. Outlook

Finally, we summarize that the regularization scheme presented in this thesis has capabilities for various generalizations which can be effectively implemented with minor changes in the RS tensor decompositions. We notice the following directions:

First, we do not provide detailed experiments to corroborate with the actual electrostatic free energies and forces of well-known biomolecules. This requires thorough investigations and provision of supporting literature, which is not in the scope of this thesis.

Secondly, the error bounds for the RBM are still a critical issue for the PBE. This is because the coefficient matrix A of the PBE system $Au = b$ has small eigenvalues of the order of $\mathcal{O}(10^{-2})$, which impedes the construction of tight error bounds. In this thesis, we use the norm of the residual as the corresponding *a posteriori* error estimator.

APPENDIX A

APPENDICES

Standard definitions

In this appendix, we state some standard important definitions and theorems that have been used in Section 2.3.1 and Section 2.3.2. For more details and proofs, see [12, 33, 91, 136].

Definition A.1 (Norm):

A norm on a vector space X is a function

$$\|\cdot\| : X \to \mathbb{R}^+ := [0,\infty); \quad x \mapsto \|x\|$$

that satisfies, $\forall\, x, y \in X$ and $\alpha \in \mathbb{F}$,

(i) $\|x\| = 0 \iff x = 0$ (faithfulness),

(ii) $\|\alpha x\| = |\alpha| : \|x\|$ (homogeneity),

(iii) $\|x + y\| \leq \|y\| + \|y\|$ (subadditivity).

A seminorm on X is a function $p : X \to \mathbb{R}^+$ that satisfies condions (ii) and (iii) above. ♢

Definition A.2 (A normed vector space):

A vector space X with a norm $\|\cdot\|$. We denote the norm on the vector space X by $\|\cdot\|_X$. ♢

Definition A.3 (Banach space):

A normed vector space $(E, \|\cdot\|)$ that is complete, that is, every Cauchy sequence in E is convergent, where E is equipped with the metric

$$d(x, y) := \|x - y\|$$

. ◊

Definition A.4 (Bounded (linear) operator):

Let X be a normed vector space; $\forall r \in \mathbb{R}^+$ let X_r denote the closed ball in X with radius r and center the origin:

$$X_r := \{x \in X \ \|x\| \ \leq \ r\}.$$

A bounded linear operator from X to Y is a linear transformation $T : X \to Y$ such that the operator norm $\|T\|$ is finite, where

$$\begin{aligned} \|T\| &:= \inf\{M \in \mathbb{R}^+ \ : \|Tx\| \leq M\|x\| \quad \forall\, x \in X\} \\ &= \sup\{\|Tx\| \ : x \in X_1\} \\ &= \sup\{\|Tx\| \ : x \in X, \quad \|x\| = 1\}. \end{aligned}$$ ◊

Definition A.5 (Convex set):

A subset of a vector space is convex if $tC+(1-t)C \subseteq C$, $\forall t \in (0,1)$, i.e., $tx+(1-t)y \in C$, $\forall x, y \in C$ and $t \in (0,1)$. This condition implies geometrically that every line segment with endpoints in C lies in C. ◊

Definition A.6 (Functional):

Let S be a set of functions. Let $f : S \to \mathbb{R}$ be a real valued function. Such functions are known as functionals. Simply put, a functional is a real valued function whose domain is a set of functions. ◊

Definition A.7 (Extremizing function):

A function for which the functional f is maximum or minimum. The value of the functional at the extremizing function is known as extremum. ◊

Definition A.8 (Spaces of continuous functions):

Let $k \in \mathbb{Z}^+$. We define the space $C^k(\bar{\Omega})$ as

$$C^k(\bar{\Omega}) \equiv \{v | D^\alpha v \text{ is bounded and uniformly continuous on } \Omega,\ \forall\, \alpha \text{ s.t. } 0 \leq |\alpha| \leq k\}.$$

Then $C^k(\bar{\Omega})$ is a Banach space (i.e., a complete normed linear space) with norm

$$\|v\|_{C^k(\bar{\Omega})} = \max_{0\leq|\alpha|\leq k} \sup_{x\in\Omega} |D^\alpha v(x)|.$$

◊

Definition A.9 (Lebesgue spaces):

Let $p \geq 1$. The Lebesgue space $L^p(\Omega)$ is defined as

$$L^P(\Omega) \equiv \{v \text{ s.t. } \|v\|_{L^P(\Omega)} < \infty\}.$$

◊

Definition A.10 (Sobolev spaces):

Let $k \in \mathbb{Z}^+$ and $p \geq 1$. The Sobolev space $W^{k,p}(\Omega)$ is defined as

$$W^{k,p}(\Omega) = \{v \text{ s.t. } D^\alpha v \ \in L^p(\Omega), \qquad \forall\, \alpha \text{ s.t. } |\alpha| \leq k\}.$$

◊

Remark A.11:

- Note that $L^2(\Omega)(\equiv H^0(\Omega)$ is the only Lebesgue space that is a Hilbert space.
- For the case $k = 0$, the Lebesgue spaces are included in the Sobolev spaces, that is, $W^{0,p}(\Omega) \equiv L^p(\Omega)$ and the case $p = 2$ corresponds to a family of Hilbert spaces.

◊

Definition A.12 (Poincaré inequality):

Let p be such that $1 \leq p < \infty$ and Ω a subset with at least one bound. Then there exists a constant C depending only on p and Ω such that for every $u \in W_0^{1,p}(\Omega)$

$$\|u\|_{L^p(\Omega)} \leq C\|\nabla u\|_{L^p(\Omega)}.$$

◊

Standard theorems

Theorem A.13 (Riesz Represenation Theorem):

Suppose l be a bounded linear functional on a Hilbert space H. Then there exists a unique vector $g \in H$ such that

$$l(f) = \langle g, f\rangle \,\forall f \in H, \quad \text{and } \|l\|_* = \|g\|. \qquad \lozenge$$

Theorem A.14 (Lax-Milgram Theorem):

Let H be a real Hilbert space, let the bilinear form $A(u, v)$ be bounded and coercive on $H \times H$, and let $F(u)$ be a bounded linear functional on H. Then there exists a unique solution to the problem:

$$\text{Find } u \in H \text{ s.t. } A(u, v) = F(v) \qquad \forall\, v\, \in H. \qquad \lozenge$$

Lemma A.15 (Fatou's lemma):

Given a measure space (Ω, Σ, μ) and a set $X \in \Sigma$, let $\{f_n\}_{n=1}^{\infty}$ be a sequence of $(\Sigma, \mathcal{B}_{\mathbb{R}_{\geq 0}})$- measurable nonnegative functions $f_n : X \mapsto [0, +\infty]$. Define the function $f : X \mapsto [0, +\infty]$ by setting

$$f(x) = \liminf_{n\to\infty} f_n(x),$$

for every $x \in X$. Then f is $(\Sigma, \mathcal{B}_{\mathbb{R}_{\geq 0}})$- measurable, and

$$\int_X f d\mu \leq \liminf_{n\to\infty} \int_X f_n d\mu. \qquad \lozenge$$

BIBLIOGRAPHY

[1] The Born ion. http://www.poissonboltzmann.org/examples/The_Born_ion/, . Accessed: 2016-11-02.

[2] File:protein folding.png. https://en.wikipedia.org/wiki/File:Protein_folding.png, . Accessed: 2019-04-23.

[3] E. Acar, T. G. Kolda, and D. M. Dunlavy. A scalable optimization approach for fitting canonical tensor decompositions. *J. Chemom.*, 25(2):67–86, 2011.

[4] Ivo Babuška, Ricardo Durán, and Rodolfo Rodríguez. Analysis of the efficiency of an a posteriori error estimator for linear triangular finite elements. *SIAM J. Numer. Anal.*, 29(4):947–964, 1992.

[5] N. A. Baker. *Biomolecular Applications of Poisson-Boltzmann Methods*, volume 21 of *Reviews in Computational Chemistry*. John Wiley & Sons, Hoboken, NJ, USA, 2005. doi:10.1002/0471720895.ch5.

[6] N. A. Baker. PQR molecular structure format. https://apbs-pdb2pqr.readthedocs.io/en/latest/formats/pqr.html, 2019.

[7] N. A. Baker, M. J. Holst, and F. Wang. The adaptive multilevel finite element solution of the Poisson-Boltzmann equation on massively parallel computers. *IBM J. Res. Devel.*, 45:427–438, 2001.

[8] N. A. Baker, D. Sept, S. Joseph, M. J. Holst, and J. A. McCammon. Electrostatics of nanosystems: application to microtubules and the ribosome. *Proc. Nat. Acad. Sci. U.S.A.*, 98(18):10037–10041, 2001. doi:10.1073/pnas.181342398.

[9] U. Baur, C. A. Beattie, P. Benner, and S. Gugercin. Interpolatory projection methods for parameterized model reduction. *SIAM J. Sci. Comput.*, 33(5):2489–2518, 2011. doi:10.1137/090776925.

[10] Ulrike Baur, Peter Benner, Bernard Haasdonk, Christian Himpe, Immanuel Martini, and Mario Ohlberger. Comparison of Methods for Parametric Model Order Reduction of Time-Dependent Problems. In Peter Benner, Albert Cohen, Mario Ohlberger, and Karen Willcox, editors, *Model Reduction and Approximation: Theory and Algorithms*, chapter 9, pages 377–407. SIAM Computational Science and Engineering, 2017. doi:10.1137/1.9781611974829.ch9.

[11] Mohammad Abid Bazaz, Mashuq un Nabi, and S. Janardhanan. A Review of Parametric Model Order Reduction Techniques. In *2012 IEEE International*

Conference on Signal Processing, Computing and Control, volume 2, pages 1–6, Waknaghat Solan, India, 2012. IEEE. doi:10.1109/ISPCC.2012.6224356.

[12] Alexander Belton. Functional Analysis Lecture Notes. Lecture Notes, University of Oxford, 2006.

[13] P. Benner and T. Breiten. Two-sided moment matching methods for nonlinear model reduction. *SIAM J. Sci. Comput.*, 37(2):B239–B260, 2015. doi:10.1137/14097255X.

[14] P. Benner, S. Gugercin, and K. Willcox. A survey of model reduction methods for parametric systems. *SIAM Review*, 57(4):483–531, 2015. doi:10.1137/130932715.

[15] P. Benner, V. Khoromskaia, and B. N. Khoromskij. Range-separated tensor formats for numerical modeling of many-particle interaction potentials. *arXiv:1606.09218v3*, pages 1–38, 2016.

[16] P. Benner, L. Feng, C. Kweyu, and M. Stein. Fast solution of the Poisson-Boltzmann equation with nonaffine parametrized boundary conditions using the reduced basis method. *arXiv:1705.08349*, 221, 2017.

[17] P. Benner, V. Khoromskaia, and B. N. Khoromskij. Range-separated tensor format for many-particle modeling. *SIAM J. Sci. Comp.*, (2):A1034–A1062, 2018. doi:10.1137/16M1098930.

[18] Peter Benner, Venera Khoromskaia, Boris Khoromskij, Cleophas Kweyu, and Matthias Stein. Computing electrostatic potentials using regularization based on the range-separated tensor format. *arXiv:1901.09864*, 2019.

[19] C. Bertoglio and B. N. Khoromskij. Low-rank quadrature-based tensor approximation of the Galerkin projected Newton/Yukawa kernels. *Comp. Phys. Comm.*, 183(4):904–912, 2012. doi:10.1016/j.cpc.2011.12.016.

[20] B. Besselink, U. Tabak, A. Lutowska, N. van de Wouw, H. Nijmeijer, D.J. Rixen, M.E. Hochstenbach, and W.H.A. Schilders. A comparison of model reduction techniques from structural dynamics, numerical mathematics and systems and control. *J Sound Vib.*, 332:4403–4422, 2013. doi:10.1016/j.jsv.2013.03.025.

[21] J.O. Bockris and A.K.N. Reddy. *Modern Electrochemistry: Ionics*. Plenum Press, New York, 1998.

[22] A. H. Boschitsch and M. O. Fenley. Hybrid boundary element and finite difference method for solving the nonlinear Poisson-Boltzman equation. *J. Comput. Chem.*, 25(7):935–955, 2004. doi:10.1002/jcc.20000.

[23] S.F. Boys, G.B. Cook, and C.M. Reeves. Automatic fundamental calculations of molecular structure. *Nature*, 178:1207–1209, 1956.

[24] D. Braess. *Nonlinear Approximation Theory*. Springer-Verlag, Berlin, 1986.

[25] D. Braess. Asymptotics for the approximation of wave functions by exponential sums. *J. Approx. Theory.*, 83:93–103, 1995.

[26] W.L. Briggs. *A Multi-grid Tutorial.* Society for Industrial and Applied Mathematics, Philadelphia, Pennsylvania, 1987.

[27] P. Burkhard, P. Taylor, and Walkinshaw M. D. X-ray structures of small ligand-FKBP complexes provide an estimate for hydrophobic interaction energies. *J. Mol. Biol.*, 295:953–962, 2000. doi:10.1006/jmbi.1999.3411.

[28] J. D. Carrol and J. Chang. Analysis of individual differences in multidimensional scaling via an N-way generalization of 'Eckart-Young' decomposition. *Psychometrika*, 35:283–319, 1970.

[29] S. Chaturantabut and D. C. Sorensen. Nonlinear model reduction via discrete empirical interpolation. *SIAM J. Sci. Comput.*, 32(5):2737–2764, 2010. doi:10.1137/090766498.

[30] L. Chen, M.J. Holst, and J. Xu. The finite element approximation of the nonlinear Poisson-Boltzmann equation. *SIAM J. Numer. Anal.*, 45(6):2298–2320, 2007. doi:10.1137/060675514.

[31] Z. Chen, N. A. Baker, and G. W. Wei. Differential geometry based solvation model I: Eulerian formulation. *J. Comput. Phys.*, 229(22):8231–8258, 2010. doi:10.1016/j.jcp.2010.06.036.

[32] I-Liang Chern, Jian-Guo Liu, and Wei-Cheng Wang. Accurate evaluation of electrostatics for macromolecules in solution. *Methods Appl. Anal.*, 10(2):309–328, 2003.

[33] Philippe Ciarlet. *The Finite Element Method for Elliptic Problems.* North-Holland, New York, NY, 1978.

[34] G. Andrés Cisneros, Mikko Karttunen, Pengyu Ren, and Celeste Sagui. Classical electrostatics for biomolecular simulations. *Chem Rev.*, 114(1):779–814, 2014. doi:10.1021/cr300461d.

[35] M. L. Connolly. Solvent-accessible surfaces of proteins and nucleic acids. *Science*, 221(4612):709–713, 1983. doi:10.1126/science.6879170.

[36] Halliday David and Resnick Robert. *Fundamentals of Physics, 9th Edition.* John Wiley & Sons, 1970. ISBN 78-0-470-46908-8.

[37] Malcolm E. Davis and Andrew McCammon. Electrostatics in biomolecular structure and dynamics. *Chem. Rev.*, 90(3):509–521, 1990.

[38] M. Deserno and C. Holm. How to mesh up Ewald sums. I. A theoretical and numerical comparison of various particle mesh routines. *J. Chem. Phys.*, 109 (18):7678–7693, 1998.

[39] Lars Diening, Petteri Harjulehto, Peter Hästö, and Michael Růžička. *Lebesgue and Sobolev Spaces with Variable Exponents*. Springer International Publishing, 2010.

[40] Bashford Donald and Case David A. Generalized born models of macromolecular solvation effects. *Annu. Rev. Phys. Chem.*, 51:129–152, 2000.

[41] F. Dong, B. Oslen, and N. A. Baker. Computational methods for biomolecular electrostatics. *Methods Cell Biol*, 84(1):843–870, 2008. doi:10.1016/S0091-679X(07)84026-X.

[42] J. L. Eftang. *Reduced Basis Methods for Parametrized Partial Differential Equations*. Ph.D. Thesis, Norwegian University of Science and Technology, Trondheim, Norway, 2011.

[43] R. Eid. *Time domain model reduction by moment matching*. Dissertation, Technische Universität München, Munich, Germany, 2009.

[44] I. Ekland and R. Temam. *Convex Analysis and Variational Problems*. North-Holland, New York, 1976.

[45] L. Feng and P. Benner. *Reduced Order Methods for modeling and computational reduction, MS&A Series*, volume 9, chapter 6: A robust algorithm for parametric model order reduction based on implicit moment matching, pages 159–186. Springer-Verlag, Berlin, Heidelberg, New York, 2014.

[46] Lihong Feng, Michael Mangold, and Peter Benner. Adaptive POD-DEIM basis construction and its application to a nonlinear population balance system. *AIChE Journal*, 63:3832–3844, 2017. ISSN 1547-5905. doi:10.1002/aic.15749.

[47] F. Fogolari, P. Zuccato, G. Esposito, and P. Viglino. Biomolecular electrostatics with the linearized Poisson-Boltzmann equation. *Biophys. J.*, 76(1):1–16, 1999. doi:10.1016/S0006-3495(99)77173-0.

[48] F. Fogolari, A. Brigo, and H. Molinari. The Poisson-Boltzmann equation for biomolecular electrostatics: a tool for structural biology. *J. Mol. Recognit.*, 15 (6):377–392, 2002. doi:10.1002/jmr.577.

[49] R. W. Freund. Model reduction methods based on Krylov subspaces. *Acta Numer.*, 12:267–319, 2003. ISSN 0962-4929.

[50] R. R. Gabdoulline, M. Stein, and R. C. Wade. qPIPSA: relating enzymatic kinetic parameters and interaction. *BMC Bioinformatics*, 8(373):1–16, 2007.

[51] Razif R. Gabdoulline and Rebecca C. Wade. Brownian dynamics simulation of protein-protein diffusional encounter. *Methods*, 14(3):329–341, 1998. ISSN 1046-2023. doi:10.1006/meth.1998.0588.

[52] C. Garcia-Garcia and D. Draper. Electrostatic interaction in a Peptide-RNA Complex. *J. Mol. Biol.*, 331:75–88, 2003. doi:10.1016/S0022-2836(03)00615-6.

[53] I. P. Gavrilyuk, W. Hackbusch, and B. N. Khoromskij. Hierarchical tensor-product approximation to the inverse and related operators in high-dimensional elliptic problems. *Computing*, 74:131–157, 2005.

[54] Weihua Geng and Shan Zhao. A two-component matched interface and boundary (MIB) regularization for charge singularity in implicit solvation. *J. Comput. Phys.*, 351:25–39, 2017. doi:10.1016/j.jcp.2017.09.026.

[55] M. K. Gilson, M. E. Davis, B. A. Luty, and J. A. McCammon. Computation of electrostatic forces on solvated molecules using the Poisson-Boltzman equation. *J. Phys. Chem.*, 97(14):3591–3600, 1993. doi:10.1021/j100116a025.

[56] W. Hackbusch and B. N. Khoromskij. Low-rank Kronecker product approximation to multi-dimensional nonlocal operators. Part I. Separable approximation of multi-variate functions. *Computing*, 76:177–202, 2006.

[57] M. W. Hess and P. Benner. Fast evaluation of time-harmonic Maxwell's equations using the reduced basis method. *IEEE Trans. Microw. Theory Techn.*, 61(6): 2265–2274, 2013. doi:10.1109/TMTT.2013.2258167.

[58] J. S. Hesthaven, B. Stamm, and S. Zhang. Efficient greedy algorithms for high-dimensional parameter spaces with applications to empirical interpolation and reduced basis methods. *ESIAM Math. Modell. Numer. Anal.*, 48(1):259–283, 2014. doi:10.1051/m2an/2013100.

[59] J.S. Hesthaven, G. Rozza, and B. Stamm. *Certified Reduced Basis Methods for Parametrized Partial Differential Equations*. Springer International Publishing, 2016. ISBN 978-3-319-22469-5.

[60] C. Himpe and M. Ohlberger. The empirical cross Gramian for parametrized nonlinear systems. In *Proceedings of the 8th Vienna International Conference on Mathematical Modelling*, volume 48(1), pages 727–728, 2015. doi:10.1016/j.ifacol.2015.05.163.

[61] M.G. Hinds, T. Maurer, Zhang J., and N.A. Nicola. Solution structure of Leukemia inhibitory factor. *J. Biol. Chem.*, 273:13738–13745, 1998. doi:10.1074/jbc.273.22.13738.

[62] M. Hinze and S. Volkwein. Proper orthogonal decomposition surrogate models for nonlinear dynamical systems: Error estimates and suboptimal control. In P. Benner, V. Mehrmann, and D.C. Sorensen, editors, *Dimension Reduction of Large-Scale Systems*, volume 45 of *Lect. Notes Comput. Sci. Eng.*, pages 261–306. Springer-Verlag, Berlin/Heidelberg, Germany, 2005.

[63] F. L. Hitchcock. The expression of a tensor or a polyadic as a sum of products. *J. Math. Phys.*, 6:164–189, 1927.

[64] M. Holst and F. Saied. Multigrid solution of the Poisson-Boltzmann equation. *J. Comput. Chem.*, 14:105–113, 1993.

[65] M. Holst and F. Saied. Numerical solution of the nonlinear Poisson-Boltzmann equation: Developing more robust and efficient methods. *J. Comput. Chem.*, 16: 337–364, 1995.

[66] M. Holst, N. A. Baker, and F. Wang. Adaptive multilevel finite element solution of the Poisson-Boltzmann equation: algorithms and examples. *J. Comput. Chem.*, 21:1319–1342, 2000. doi:10.1002/1096-987X(20001130)21:15<1319::AID-JCC1>3.0.CO;2-8.

[67] M. J. Holst. *Multilevel Methods for the Poisson-Boltzmann Equation.* Ph.D. Thesis, Numerical Computing group, University of Illinois, Urbana-Champaign, IL, USA, 1994.

[68] B. Honig and A. Nicholls. Classical electrostatics in biology and chemistry. *Sci., New Series*, 268(5214):1144–1149, 1995. doi:10.1126/science.7761829.

[69] W. Humphrey, A. Dalke, and K. Schulten. VMD- visual molecular dynamics. *J. Mol. Graph.*, 14(1):33–38, 1996. doi:10.1016/0263-7855(96)00018-5.

[70] P. H. Hünenberger and J. A. McCammon. Effect of artificial periodicity in simulations of biomolecules under Ewald boundary conditions: a continuum electrostatics study. *Biophys. Chemistry*, 78:69–88, 1999.

[71] Philippe Hünenburger, Ulf Börjesson, and Roberto Lins. Electrostatic interactions in biomolecular systems. *Chimia*, 55:861–866, 2001.

[72] Wonpil Im, Dmitrii Beglov, and Benoit Roux. Continuum solvation model: Computation of electrostatic forces from numerical solutions to the Poisson-Boltzmann equation. *Comput. Phys. Commun.*, 8:59–75, 1998.

[73] T. C. Ionescu and J. M. A. Scherpen. Cross Gramians for Nonlinear Systems. In *TC7 Conference on System Modeling and Optimization*, volume 23, pages 322–323, 2007. URL `http://ifip2007.agh.edu.pl/abstracts/48.pdf`.

[74] T. C. Ionescu and J. M. A. Scherpen. Nonlinear Cross Gramians and Gradient Systems. In *Proc. IEEE Conf. Decis. Control*, volume 46, pages 3745–3750, 2007. doi:10.1109/CDC.2007.4434707.

[75] J. D. Jackson. *Classical Electrodynamics.* Wiley, New York, 1975.

[76] Sophie E. Jackson and Alan R. Fersht. Folding of Chymotrypsin Inhibitor 2. 1. Evidence for a two-state transition. *Biochemistry*, 30:10428–10435, 1991.

[77] Lijie Ji, Yanlai Chen, and Zhenli Xu. A reduced basis method for the nonlinear Poisson-Boltzmann equation. *arXiv:1808.09392*, 2018.

[78] Elizabeth Jurrus, Dave Engel, Keith Star, Kyle Monson, Juan Brandi, Lisa E. Felberg, David H. Brookes, Leighton Wilson, Jiahui Chen, Karina Liles, Minju Chun, Peter Li, David W. Gohara, Todd Dolinsky, Robert Konecny, David R. Koes, Jens Erik Nielsen, Teresa Head-Gordon, Weihua Geng, Robert Krasny,

Guo-Wei Wei, Michael J. Holst, J. Andrew McCammon, and Nathan A. Baker. Improvements to the apbs biomolecular solvation software suite. *Protein Science*, 27(1):112–128, 2017. doi:10.1002/pro.3280.

[79] V. Khoromskaia and B. Khoromskij. *Tensor Numerical Methods in Quantum Chemistry.* Walter de Gruyter GmbH, Berlin/Boston, 2018. ISBN 978-3-11-037015-7.

[80] V. Khoromskaia and B. N. Khoromskij. Low rank Tucker tensor approximation to the classical potentials. *Centr. Europ. J. Math.*, 5(3):1–28, 2007.

[81] V. Khoromskaia and B. N. Khoromskij. Grid-based lattice summation of electrostatic potentials by assembled rank-structured tensor approximation. *Comp. Phys. Communications*, 185(12), 2014.

[82] B. Khoromskij and V. Khoromskaia. Multigrid accelerated tensor approximation of function related multidimensional arrays. *SIAM J. Sci. Comput.*, 31(4):3002–3026, 2009. doi:doi.org/10.1137/080730408.

[83] B. N. Khoromskij. Structured rank-$(r_1, \ldots, r_d)$ decomposition of function-related operators in $\mathbb{R}^d$. *Comp. Meth. Appl. Math*, 6(2):194–220, 2006.

[84] B. N. Khoromskij. Fast and accurate tensor approximation of multivariate convolution with linear scaling in dimension. *J. Comp. Appl. Math.*, 234(11):3122–3139, 2010. doi:10.1016/j.cam.2010.02.004.

[85] B. N. Khoromskij. Range-separated tensor representation of the discretized multidimensional Dirac delta and elliptic operator inverse. *Preprint*, arXiv:1812.02684v1, 2018.

[86] Boris N. Khoromskij. *Tensor Numerical Methods in Scientific Computing.* De Gruyter, Berlin, 2018.

[87] H.A.L. Kiers. Towards a standardized notation and terminology in multiway analysis. *J. Chemom.*, 14:105–122, 2000.

[88] John G. Kirkwood. Theory of solutions of molecules containing widely separated charges with special application to zwitterions. *J. Chem. Phys.*, 2:351–361, 1934.

[89] T. Kolda. Orthogonal tensor decompositions. *SIAM J. Matrix Anal. Appl.*, 23: 243–255, 2001.

[90] T. Kolda and B. Bader. Tensor decompositions and applications. *SIAM Rev.*, 51 (3):455–500, 2009.

[91] Erwin Kreyszig. *Introductory Functional Analysis with Applications.* John Wiley & Sons, 1978. ISBN 0-471-50731-8.

[92] J. B. Kruskal. Three-way arrays: rank and uniqueness of trilinear decompositions, with applications to arithmetic complexity and statistics. *Linear Algebra Appl.*, 18:95–138, 1970.

[93] K. Kunisch and S. Volkwein. Galerkin proper orthogonal decomposition methods for parabolic systems. *Numer. Math.*, 90:117–148, 2001.

[94] K. Kunisch and S. Volkwein. Galerkin proper orthogonal decomposition methods for a general equation in fluid dynamics. *SIAM J. Numer. Anal.*, 40(2):492–515, 2002. ISSN 0036-1429. doi:10.1137/S0036142900382612.

[95] K. Kunisch and S. Volkwein. Proper orthogonal decomposition for optimality systems. *ESAIM: Math. Model. Numer. Anal.*, 42(1):1–23, 2008. ISSN 0764-583X.

[96] C. Kweyu, M. Hess, L. Feng, M. Stein, and P. Benner. Reduced Basis Method for Poisson-Boltzmann Equation. In M. Papadrakakis, V. Papadopoulos, G. Stefanou, and V. Plevris, editors, *ECCOMAS Congress 2016 - Proc. of the VII European Congress on Computational Methods in Applied Sciences and Engineering*, volume 2, pages 4187–4195, Athens, 2016. National Technical University of Athens.

[97] Cleophas Kweyu, Lihong Feng, Matthias Stein, and Peter Benner. Fast solution of the nonlinear Poisson-Boltzmann equation using the reduced basis method and range-separated tensor format. *ScienceOpen Posters*, 2018. doi:10.14293/P2199-8442.1.SOP-MATH.OHIYQV.v1.

[98] Cleophas Kweyu, Lihong Feng, Matthias Stein, and Peter Benner. Reduced basis method for the nonlinear Poisson-Boltzmann equation regularized by the range-separated canonical tensor format. in prep..

[99] Cleophas Kweyu, Venera Khoromskaia, Boris Khoromskij, Matthias Stein, and Peter Benner. Solution decomposition for the nonlinear Poisson-Boltzmann equation using the range-separated tensor format. in prep..

[100] Serge Lang. *Introduction to Linear Algebra*. Springer-Verlag New York Inc., 1986. ISBN 96205-0.

[101] L. De Lathauwer. *Signal Processing based on Multilinear Algebra*. Ph.D. Thesis, Katholeke Universiteit Leuven, 1997.

[102] L. De Lathauwer, B. De Moor, and J. Vandewalle. A multilinear singular value decomposition. *SIAM J. Matrix Anal. Appl.*, 21(4):1253–1278, 2000. doi:10.1137/S0895479896305696.

[103] L.Chen, M. J. Holst, and J. Xu. The finite element approximation of the nonlinear Poisson-Boltzmann equation. *SIAM J. Numer. Anal.*, 45(6):2298–2320, 2007. doi:10.1137/060675514.

[104] M.H. Le Du, P. Marchot, P.E. Bougis, and J.C. Fontecilla-Camps. 1.9 Angstrom resolution structure of fasciculine 1, an anti-acetylcholinesterase toxin from green mamba snake venom. *J. Biol. Chem.*, 267:22122–22130, 1992.

[105] Bo Li, Xiaoliang Cheng, and Zhengfang Zhang. Dielectric boundary force in molecular solvation with the Poisson-Boltzmann free energy: A shape derivative approach. *SIAM J. Appl. Math.*, 71(6):2093–2111, 2011.

[106] F. Lipparini, B. Stamm, E. Cances, Y. Maday, and B. Mennucci. Fast domain decomposition algorithm for continuum solvation models: Energy and first derivatives. *J. Chem. Theor. Comp.*, 9:3637–3648, 2013.

[107] Adam Liwo. *Computational Methods to Study the Structure and Dynamics of Biomolecules and Biomolecular Processes: From Bioinformatics to Molecular Quantum Mechanics*. Springer-Verlag Berlin Heidelberg, 2014. ISBN 978-3-642-28554-7. doi:10.1007/978-3-642-28554-7.

[108] B. Z. Lu, Y. C. Zhou, M. J. Holst, and J. A. McCammon. Recent progress in numerical methods for the Poisson-Boltzmann equation in biophysical applications. *Commun. Comput. Phys.*, 3(5):973–1009, 2008.

[109] Brock A. Luty, Malcolm E. Davis, and J. Andrew McCammon. Solving the finite difference nonlinear Poisson-Boltzmann equation. *J. Comput. Chem.*, 13 (9):1114–1118, 1992. doi:10.1002/jcc.540130911.

[110] M. Mirzadeh, M. Theillard, A. Helgadottir, D. Boy, and F. Gibou. An adaptive, finite difference solver for the nonlinear Poisson-Boltzmann equation with applications to biomolecular computations. *Commun. Comput. Phys.*, 13(1):150–173, 2013. doi:10.4208/cicp.290711.181011s.

[111] N. Narayana, T.C. Diller, K. Koide, M.E. Bunnage, K. C. Nicolaou, L.L. Brunton, N. Xuong, L.F.T Eyck, and S.S. Taylor. Crystal structure of the potent natural product inhibitor Balanol in complex with the catalytic subunit of cAMP-dependent protein Kinase. *Biochem.*, 38:2367–2376, 1999. doi:10.1021/bi9820659.

[112] M. T. Neves-Petersen and S. Petersen. Protein electrostatics: A review of the equations and methods used to model electrostatic equations in biomolecules - applications in biotechnology. *Biotechnol. Annu. Rev.*, 9:315–395, 2003. doi:10.1016/S1387-2656(03)09010-0.

[113] Y. Notay. An aggregation-based algebraic multigrid method. *Electron. Trans. Numer. Anal.*, 37:123–146, 2010.

[114] Y. Notay. Aggregation-based algebraic multigrid for convection-diffusion equations. *SIAM J. Sci. Comput.*, 34:A2288–A2316, 2012.

[115] H. Oberoi and N. M. Allewell. Multigrid solution of the nonlinear Poisson-Boltzmann equation and calculation of titration curves. *Biophys. J.*, 65(1):48–55, 1993. doi:10.1016/S0006-3495(93)81032-4.

[116] Encyclopedia of Mathematics. Newton potential. `http://www.encyclopediaofmath.org/index.php?title=Newton_potential&oldid=33114`. Accessed: 2018-03-12.

[117] A. Onufriev, D.A. Case, and D. Bashford. Effective Born radii in the generalized Born approximation: The importance of being perfect. *J. Comput. Chem.*, 23 (14), 2002.

[118] Stéphanie Pasche, Janos Vörös, Hans J. Griesser, Nicholas D. Spencer, and Marcus Textor. Effects of ionic strength and surface charge on protein adsorption at PEGylated surfaces. *J. Phys. Chem. B*, 109:17545–17552, 2005. doi:10.1021/jp050431+.

[119] A. T. Patera and G. Rozza. *Reduced Basis Approximation and a Posteriori Error Estimation for Parametrized Partial Differential Equations.* MIT, 2007. To appear in (tentative rubric) MIT Pappalardo Graduate Monographs in Mechanical Engineering.

[120] E. L. Pollock and J. Glosli. Comments on P^3M, FMM and the Ewald method for large periodic Coulombic systems. *Comp. Phys. Comm.*, 95:93–110, 1996.

[121] Cai Qin, Hsieh Meng-Juei, Wang Jun, and Luo Ray. Performance of nonlinear finite difference Poisson-Boltzmann solvers. *J. Chem. Theory Comput.*, 6(1):203–211, 2010. doi:10.1021/ct900381r.

[122] C. Quan, B. Stamm, and Y. Maday. A domain decomposition method for the Poisson-Boltzmann solvation models. *SIAM J. Sci. Comp.*, 41(2):B320–B350, 2019.

[123] A. Quarteroni, A. Manzoni, and F. Negri. *Reduced Basis Methods for Parametrized Partial Differential Equations: An Introduction.* Springer International Publishing, 2016. ISBN 978-3-319-15430-5. doi:10.1007/978-3-319-15431-2.

[124] Stephan Rabanser, Oleksandr Shchur, and Stephan Günnemann. Introduction to tensor decompositions and their applications in machine learning. *arXiv:1711.10781*, 2017.

[125] Zoran Radić, Paul D. Kirchhoff, Daniel M. Quinn, J. Andrew McCammon, and Palmer Taylor. Electrostatic influence on the kinetics of ligand binding to acetylcholinesterase: Distinctions between active center ligands and fasciculin. *J. Biol. Chem.*, 272(37):23265–23277, 1997. doi:10.1074/jbc.272.37.23265.

[126] W. Rocchia. Poisson-Boltzmann equation boundary conditions for biological applications. *Math. Comput. Model.*, 41(10):1109–1118, 2005. doi:10.1016/j.mcm.2005.05.006.

[127] W. Rocchia, E. Alexov, and B. Honig. Extending the applicability of the nonlinear Poisson-Boltzmann equation: Multiple dielectric constants and multivalent ions. *J. Phys. Chem.*, 105(28):6507–6514, 2001. doi:10.1021/jp010454y.

[128] G. Rozza, D. B. P. Huynh, and A. T. Patera. Reduced basis approximation and a posteriori error estimation for affinely parametrized elliptic coercive partial differential equations. *Archives of Computational Methods in Engineering*, 15(3): 229–275, 2008. doi:10.1007/s11831-008-9019-9.

[129] Henrik Sandberg, Bart Besselink, and Madhu Belur. Introduction to Model Order Reduction. Lecture notes, KTH Royal Institute of Technology.

[130] Wilhelmus Schilders. Introduction to Model Order Reduction. In Wilhelmus H. A. Schilders, Henk A. van der Vorst, and Joost Rommes, editors, *Model Order Reduction: Theory, Research Aspects and Applications*, pages 3–32. Springer Berlin Heidelberg, Berlin, Heidelberg, 2008. ISBN 978-3-540-78841-6. doi:10.1007/978-3-540-78841-6_1.

[131] Emek Seyrek, Paul L. Dubin, Christophe Tribet, and Elizabeth A. Gamble. Ionic strength dependence of protein-polyelectrolyte interactions. *Biomacromolecules*, 4:273–282, 2003. doi:10.1021/bm025664a.

[132] K. A. Sharp and B. Honig. Electrostatic interactions in macromolecules: theory and applications. *Annu. Rev. Biophys. Chem.*, 19:301–332, 1990. doi:10.1146/annurev.bb.19.060190.001505.

[133] A. I. Shestakov, J. L. Milovich, and A. Noy. Solution of the nonlinear Poisson-Boltzmann equation using pseudo-transient continuation and the finite element method. *Commun. Comput. Phys.*, 247:62–79, 2002. doi:10.1006/jcis.2001.8033.

[134] A.K. Smilde, R. Bro, and P. Geladi. *Multi-way Analysis with Applications in the Chemical Sciences*. Wiley, 2004.

[135] M. Stein, R. R. Gabdoulline, and R. C. Wade. Cross-species analysis of the glycolytic pathway by comparison of molecular interaction fields. *Molecular Biosystems*, 6:162–174, 2010.

[136] Gerald Teschl. *Topics in Real and Functional Analysis*. American Mathematical Society, Providence, Rhode Island, 2019.

[137] A.Y. Toukmaji and J.A. Board. Ewald summation techniques in perspective: a survey. *Comput. Phys. Commun.*, 95:73–92, 1996.

[138] L. R. Tucker. Some mathematical notes on three-mode factor analysis. *Psychometrika*, 31:279–311, 1966.

[139] L.R. Tucker. Implications of Factor Analysis of three-way Matrices for Measurement of Change. In C. W. Harris, editor, *Problems in Measuring Change*, pages 122–137. University of Wisconsin Press, 1963.

[140] S. Vergara-Perez and M. Marucho. MPBEC, a Matlab program for biomolecular electrostatic calculations. *Comput. Phys. Commun.*, 198:179–194, 2016. doi:10.1016/j.cpc.2015.08.029.

[141] S. Volkwein. Model Reduction using Proper Orthogonal Decomposition. Lecture notes, University of Konstanz, 2013.

[142] R.C. Wade, R.R. Gabdoulline, and F. De Rienzo. Protein interaction property similarity analysis. *Int. J. Quantum Chem.*, 83:122–127, 2001.

[143] J. Wang and R. Luo. Assessment of linear finite difference Poisson-Boltzmann solvers. *J. Comput. Chem.*, 31:1689–1698, 2010. doi:10.1016/j.cpc.2015.08.029.

[144] Xianwei Wang, Yang Li, Ya Gao, Zejin Yang, Chenhui Lu, and Tong Zhu. A quantum mechanical computational method for modeling electrostatic and solvation effects of protein. *Scientific Report*, 8(1):1–10, 2018. doi:10.1038/s41598-018-23783-8.

[145] J. Warwicker and H. C. Watson. Calculation of the electric potential in the active site cleft due to α-helix dipoles. *J. Mol. Biol.*, 157(4):671–679, 1982. doi:10.1016/0022-2836(82)90505-8.

[146] B.A. Wells and A.L. Chaffee. Ewald summation for molecular simulations. *J. Chem. Theory Comput.*, 11:3684–3695, 2015.

[147] Wikibooks. Structural biochemistry/proteins/x-ray crystallography. `https://en.wikibooks.org/wiki/Structural_Biochemistry/Proteins/X-ray_Crystallography`, 2016.

[148] D. Wirtz, D. C. Sorensen, and B. Haasdonk. A posteriori error estimation for DEIM reduced nonlinear dynamical systems. *SIAM J. Sci. Comput.*, 36(2):A311–A338, 2014. doi:10.1137/120899042.

[149] D. Xie. New solution decomposition and minimization schemes for Poisson-Boltzmann equation in calculation of biomolecular electrostatics. *J. Comput. Phys.*, 275:294–309, 2014. doi:10.1016/j.jcp.2014.07.012.

[150] D. Xie and J. Ying. A new box iterative method for a class of nonlinear interface problems with application in solving Poisson-Boltzmann equation. *J. Comput. Appl. Math.*, 307:319–334, 2016. doi:10.1016/j.cam.2016.01.005.

[151] Dexian Xie. New solution decomposition and minimization scheme for Poisson-Boltzmann equation in calculation of biomolecular electrostatics. *J. Comput. Phys.*, 275:294–309, 2014.

[152] Yao Yue, Lihong Feng, and Peter Benner. Interpolation of Reduced-Order Models Based on Modal Analysis. In *2018 IEEE MTT-S International Conference on Numerical Electromagnetic and Multiphsics Modeling and Optimization*, pages 1–4. IEEE, 2018.

[153] M. Zhanfeng and H. Chao. Structure-preserving balanced truncation for flexible spacecraft using cross gramian. *Journal of Beijing University of Aeronautics and Astronautics*, 34(12):1437–1440, 2008. URL `en.cnki.com.cn/Article_en/CJFDTOTAL-BJHK200812017.htm`.

[154] H. X. Zhou. Boundary element solution of macromolecular electrostatics: Interaction energy between two proteins. *Biophys. J.*, 65(2):955–963, 1993. doi:10.1016/S0006-3495(93)81094-4.

EHRENERKLÄRUNG

Ich versichere hiermit, dass ich die vorliegende Arbeit ohne unzulässige Hilfe Dritter und ohne Benutzung anderer als der angegebenen Hilfsmittel angefertigt habe; verwendete fremde und eigene Quellen sind als solche kenntlich gemacht. Ich habe insbesondere nicht wissentlich:

- Ergebnisse erfunden oder widersprüchliche Ergebnisse verschwiegen,
- statistische Verfahren absichtlich missbraucht, um Daten in ungerechtfertigter Weise zu interpretieren,
- fremde Ergebnisse oder Veröffentlichungen plagiiert oder verzerrt wiedergegeben.

Mir ist bekannt, dass Verstöße gegen das Urheberrecht Unterlassungs- und Schadenersatzansprüche des Urhebers sowie eine strafrechtliche Ahndung durch die Strafverfolgungsbehörden begründen kann.

Die Arbeit wurde bisher weder im Inland noch im Ausland in gleicher oder ähnlicher Form als Dissertation eingereicht und ist als Ganzes auch noch nicht veröffentlicht.

DECLARATION OF HONOR

I hereby declare that I produced this thesis without prohibited assistance and that all sources of information that were used in producing this thesis, including my own publications, have been clearly marked and referenced. In particular I have not wilfully:

- Fabricated data or ignored or removed undesired results.
- Misused statistical methods with the aim of drawing other conclusions than those warranted by the available data.
- Plagiarised data or publications or presented them in a disorted way.

I know that violations of copyright may lead to injunction and damage claims from the author or prosecution by the law enforcement authorities.

This work has not previously been submitted as a doctoral thesis in the same or a similar form in Germany or in any other country. It has not previously been published as a whole.

Magdeburg, 30.04.2019

Cleophas Muganda Kweyu