INTERIOR-POINT METHODS FOR MINIMIZATION OF POTENTIAL ENERGY FUNCTIONS OF POLYPEPTIDES

MUTHU SOLAYAPPAN
(M.S., University of Florida)

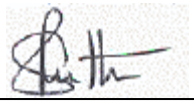
A THESIS SUBMITTED
FOR THE DEGREE OF MASTER OF ENGINEERING
DEPARTMENT OF INDUSTRIAL AND SYSTEMS ENGINEERING
NATIONAL UNIVERSITY OF SINGAPORE

2011

DECLARATION

I hereby declare that this thesis is my original work and it has been written by me in its entirety. I have duly acknowledged all the sources of information which have been used in the thesis.

This thesis has also not been submitted for any degree in any university previously.



MUTHU SOLAYAPPAN
11 April 2013

Acknowledgements

First and foremost, I would like to thank my supervisors, Dr. Ng Kien Ming and Professor Poh Kim Leng for accepting me as their student and giving me an opportunity to pursue my research under their guidance. I am thankful to both of them for having spent time with me discussing research, which often helps me to gain a better perspective of the research problem. I appreciate the freedom that they gave me in my research work and I'll always be indebted to them for that. I also thank my supervisors for providing me an opportunity to work on other research projects. Apart from providing financial support, the experience also helped me to gain some knowledge in other areas of research as well.

I would also like to thank the Department of Industrial and Systems Engineering (ISE) for supporting my research financially. Special thanks to the administrative staff at ISE, especially Ms. Ow Lai Chun for helping me with the administrative work during my candidature at the University.

The computing lab has always provided me with an excellent working atmosphere and I am thankful to my colleagues who made it possible. I have always enjoyed my conversations with Pan Jie, Zhu Zhecheng, and Aldy Gunawan. I couldn't have enjoyed my stay in Singapore more if it wasn't for the friends that I made whilst my stay here. In particular, I appreciate my friendship with Manohar, Murali, Pradeep, Satish and Malik for they always have been a source

of support and encouragement during my stay in Singapore.

My wife and my son has always been a source of emotional support for me over the past years and I thank both of them for their patience, love and care that they continue to shower on me. Lastly, my parents love and support have played a great role in motivating me. I thank them for their patience and the belief they had in me.

Contents

Abstract

Determining the minimum energy conformation of polypeptides from its amino acid sequence is an essential part of the problem of protein structure prediction. Our research focuses on developing *ab initio* methods to minimize the nonlinear, nonconvex potential energy function of proteins constrained by the bounds on dihedral angles. We use the CHARMM energy function which calculates the total potential energy of a protein as a sum of its interaction energies. Two new approaches belonging to the class of interior-point methods have been proposed to solve the above-mentioned problem.

The first approach uses a barrier function to transform the original problem into a sequence of subproblems. A key feature of our method lies in how such subproblems are solved. First-order necessary conditions are used to generate a search direction, which is the direction of descent for the subproblem being solved. In order to determine the steplength we employ the golden section search method. Issues related to the algorithm implementation, parameter initialization and parameter updates are also discussed. The performance of the proposed approach is also shown by applying it to a number of standard test problems from the literature.

The second approach is also based on the barrier function method. However, it does not employ an external function to be used as a barrier function. Utilizing

an external function will only complicate an already complex objective function. Hence, the term for Lennard-Jones 6-12 potential, which is used to model the van der Waals interactions in the CHARMM energy function is used as a barrier function. Thus a hypothetical barrier problem using the Lennard-Jones term is formulated. The Lennard-Jones term satisfies the properties required of a barrier function and hence its usage guarantees at least a good local solution, if not a global one. In order to gauge the performance of the proposed approach, a number of problems in the area of energy minimization of Lennard-Jones clusters are solved.

The two proposed solution approaches have been utilized to solve a number of dipeptide structures of amino acids. The dipeptide structures serve as a good starting point for testing the efficiency of the proposed methods. The ability of the solution methods to handle larger problems is also tested by applying it to several polypeptide structures to determine their minimum energy conformation. The performance of the solution methods is also compared with that of a genetic algorithm implementation. Apart from this, the results obtained are also compared with those available the literature. Based on the comparison, we conclude that the proposed approaches are computationally inexpensive and provide good quality solutions.

List of Tables

List of Figures

Chapter 1

Introduction

Peptides are short polymers of amino acids. They play an important role in physiological and biochemical functions of life. Shorter peptides consisting of two amino acids and joined by a single peptide bond are called dipeptides. A linear chain of 20 or more amino acids joined together by peptide bonds are called polypeptides. One or more polypeptides combine to form proteins. As it is widely believed that the three-dimensional (native) structure of protein is the one which minimizes its potential energy. Hence, determining the minimum energy conformation of proteins form an integral part of protein structure prediction.

1.1 Motivation

The problem of protein structure prediction is one of the prominent problems in the field of molecular biology. In spite of rigorous research done over the past years, the problem still remains an unsolved one. The problem in question is to find the native three-dimensional (stable) structure of the protein from its linear sequence of amino acids. In the following, we discuss the potential applications and importance of solving the problem of protein structure prediction.

Currently, the protein structure is determined through experimental tech-

niques such as X-ray crystallography and nuclear magnetic resonance (NMR) spectroscopy. Though these methods are productive, Wider (2000) mentions that they are extremely time consuming and very expensive. Moreover, the author describes the difficulty of some proteins which cannot be crystallized and hence the X-ray crystallography method cannot be used to study the structure of the protein. For NMR methods to be used, the protein in solution should be of specific density. If the protein of interest, in its solution form does not measure up to the required density levels, then NMR techniques cannot be used. Hence, development of computational techniques to address the problem of protein structure prediction is of high importance.

One of the main applications of protein structure prediction is its usability in *de novo* protein design, i.e. helping to identify the amino acid sequences that fold into proteins with desired functions. As Floudas *et al.* (2006) states, the main goal of protein design is not only to achieve the desired structure but also to render specific functions or properties to the novel protein. Most of the diseases, Alzheimer's disease, Parkinson's disease to name a few, occur due to malfunctioning of proteins or misfolded proteins. Thus, with the artificially designed proteins, we will be able to treat the diseases that occur due to improper functioning of proteins. This is made possible by artificial drug design for which the structure of protein representing the minimum energy is required. The problem of peptide docking, closely related to the protein folding problem, requires identification of equilibrium structures for a macromolecule-ligand complex. By treating it as a protein folding problem, apart from correctly identifying the binding site for the target molecule it also helps to identify a number of equilibrium structures for candidate docking molecules.

The problem of protein structure prediction is similar to the problem of molecular structure prediction. Knowledge of molecular structure is essential for design of molecules for specific applications. Examples of these types of applications provided by Meza & Martinez (1994) include development of enzymes for toxic wastes removal, development of new catalysts for material processing and the design of new anti-cancer agents. The design and development of these drugs depends on the accurate determination of the structure of the corresponding molecules. But for smaller molecules, molecular structure prediction is still an unsolved problem. Molecular Dynamics (MD) simulation, one of the many techniques in the area of computational chemistry, is used to study the macroscopic properties of complex chemical systems. The initial step in the Molecular dynamics studies is to provide a structure of the molecule that minimizes its free energy. Better results are obtained from MD studies with structures that truly represent its global minimum state. As of now, structures for which true global minimum is not known, a set of low-energy conformations, which often represent meta stable states are used (Wilson & Cui, 1988). Thus solution methods that are developed to determine the minimum energy conformation can also easily be adapted to solve the molecular structure prediction problem.

The application of energy minimization problems is not restricted to computational chemistry or structural biology. Moloi & Ali (2005) mentions the applicability of minimizing the potential energy equation in nano-scale devices within the semiconductor industry. Thus the problem of energy minimization, with its wide areas of application and uses, should be dealt in greater detail to provide elaborate, meaningful and efficient solutions that could be put to practical use.

1.2 Current Scenario

Recombinant DNA techniques facilitated rapid determination of DNA sequences which in turn helped in discovering the amino acid sequences of proteins from structural genes. The number of such sequences is increasing almost exponentially whereas the progress on the structure prediction front is on the lower side. The functional properties of proteins depend on their three-dimensional structure. In order to aid the process of protein structure prediction, the National Institute of General Medical Sciences (NIGMS), launched the Protein Structure Initiative (PSI), in 1999. The overall strategy of PSI is to experimentally determine unique protein structures, thereby creating a systematic sampling of major protein families and a large collection of protein structures (National Institute of Health, 1999). Structures thus created will serve as templates for computational modeling of related sequences.

Several methods have been developed to predict the minimum energy conformation of protein structures by comparing the target sequence to a given template. Though success rate has been higher, these methods require a template to which it can compare and predict the structure of the sequence in question. The other class of methods, called *ab initio* methods, predicts the three-dimensional structure directly from the amino acid sequence without resorting to any template. However, such methods require a scoring function which could accurately model the folding pathway of the protein.

1.3 Challenges

Ever since Anfinsen (1973) suggested that the three-dimensional structure of a native protein is the one in which the Gibbs free energy of the whole system is the lowest, several quantitative and qualitative systems for modeling the energy function of proteins has been developed. Anfinsen's hypothesis led to a redefinition of the problem of protein structure prediction to finding the minimum energy conformation of proteins. Such a formulation led to the use of several optimization techniques in search of local as well as global optimal solutions.

The most common optimization techniques employed in this area are simulated annealing (Liu & Beveridge, 2002; Liu & Tao, 2006; Rohl *et al.*, 2004; Son *et al.*, 2012), genetic algorithm (Brain & Addicoat, 2011; de Sancho & Rey, 2008; John & Sali, 2003; Schneider, 2002) and monte carlo simulation (Al-Mekhnaqi *et al.*, 2009; Guvench & MacKerell, 2008; Kolinski & Skolnick, 1994). These methods help in searching of the vast conformational space of the energy hypersurface to find good solution(s). Over the years, different variations of these methods have been tried and good solutions have also been reported. Of the number of exact methods that have been proposed, only alpha Branch and Bound algorithm developed by Maranas *et al.* (1996) have reported encouraging results. The main focus of our research is to develop efficient exact methods to solve the problem of energy minimization. The choice of exact methods has its advantages because of the mathematical basis that it provides to determine the quality of solution obtained. It will help to determine if the solution obtained is local or global optimum, failing which we would at least have an idea of how far it is from the optimum.

1.4 Background

Proteins are arguably the most complex and vital components of life. Proteins are a class of bio-macromolecules that make up the primary constituents of biological organisms. Each protein that we know of has specific functions to perform which is highly dependent on its three-dimensional structure. Functions include, but are not limited to, catalyzing chemical reactions, storage and transport of ligands, and immune response. This section aims to give an overview of proteins and the components that make them, the different structures they adapt, its geometrical representation and the existing methods to predict their structures.

1.4.1 Amino Acids

Amino acids are the basic building blocks of proteins. In nature, there are only 20 different types of amino acids. All the amino acids have a carboxyl group ($COOH$), an amino group (NH_2) and a hydrogen atom attached to the central carbon atom (C_α). However, the difference between the amino acids arises due to the different side chain (R) that is attached to C_α. Figure 1.1 represents a schematic diagram of an amino acid. The amino acids are generally classified

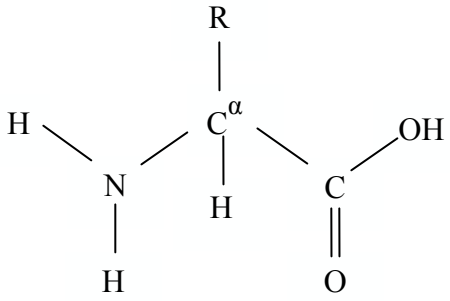


Figure 1.1: Structure of an amino acid

Table 1.1: Amino acid classification and notation

Hydrophobic	Alanine(Ala, A), Valine(Val, V), Phenyalanine(Phe, F) Proline(Pro, P), Methionine(Met, M), Isoleucine(Ile, I) Leucine(Leu, L)
Charged	Aspartic acid(Asp, D), Glutamic acid(Glu, E), Lysine(Lys, K) Arginine(Arg, R)
Polar	Serine(Ser, S), Threonine(Thr, T), Tyrosine(Tyr, Y) Histidine(His, H), Cysteine(Cys, C), Asparagine(Asn, N) Glutamine(Gln, Q), Tryptophan(Trp, W)

according to the side chain attached to the central carbon atom. The side chain could be a simple hydrogen atom or sometimes a complex aromatic ring. Branden & Tooze (1991) classifies amino acids as Hydrophobic, Charged and Polar. Table 1.1 lists the classification of amino acids along with the three letter and single letter notation that are commonly used. As seen in Table 1.1, each protein can be uniquely represented by a sequence of three-letter or one-letter codes. Amino acids are joined end to end during the synthesis of protein. This is made possible by condensation reaction in which a molecule of water is shed and a peptide bond is formed between adjacent amino acids. Thus numerous amino acids are joined end to end to form a polypeptide or a protein. The repeating -$NC_{\alpha}C$- chain of a protein is called its backbone. Hormones are the smallest proteins and have about 25 to 100 amino acid residues, typical globular proteins have about 100 to 500, while fibrous proteins may have more than 3000 residues.

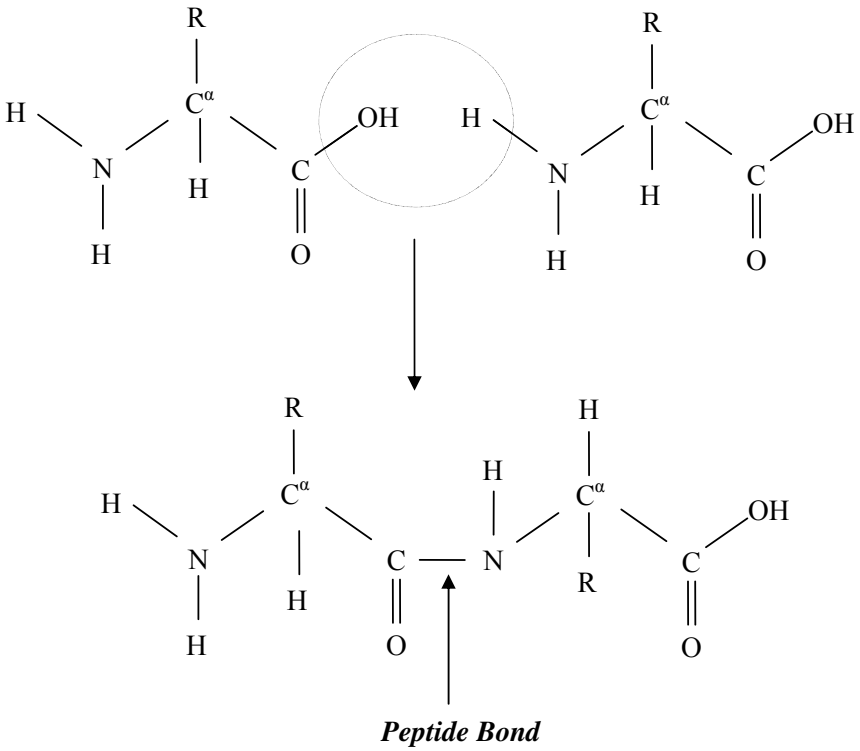


Figure 1.2: Peptide bond formation

1.4.2 Types of Protein Structure

The first X-ray crystallographic structural results on a globular protein molecule, myoglobin, reported in 1958, showcased the lack of symmetry and the complexity that the protein's structure possess. Such irregularity in structure is essential for proteins to fulfill their functions. In spite of the irregularity, there are certain regular features that help to classify protein structures.

The linear chain of amino acids is called the *Primary Structure.* Though, the structure is extremely short-lived, it contains the sequence of amino acids that are required to form the final shape. Figure 1.3 shows the primary structure of a protein.

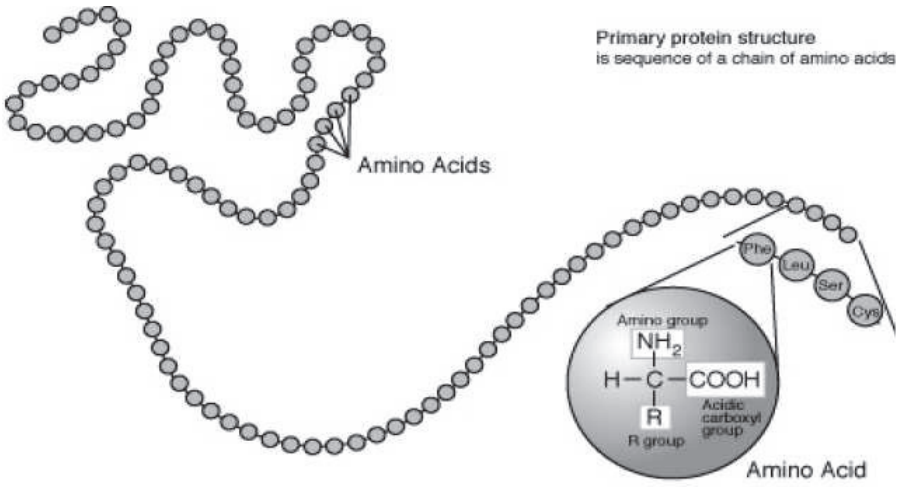


Figure 1.3: Primary structure of a protein

It has been observed that in a folded protein, the interior of the molecule is hydrophobic, whereas the surface is hydrophilic. The side chain components of water-soluble proteins are hydrophobic. In order to minimize the exposure of side chain components to the solvent, the side chains are bought into the core, which helps in stabilizing the folded state. Side chains which are charged and polar are situated on the surface, thereby interacting with the surrounding environment. Apart from the hydrophobic side chains, hydrogen bond formation also helps in stabilizing the protein structure. These hydrogen bond formations lead to what is called the *Secondary Structure* of the protein molecule. Such secondary structure is usually of two types: *Alpha Helices* and *Beta Sheets.* Both types have the main chain NH and CO groups participating in the formation of hydrogen bonds. Figure 1.4 shows the commonly occurring α helix and β sheet structures.

The final specific geometric shape that a protein assumes is called the *Tertiary Structure.* This final shape is determined by a variety of bonding interactions

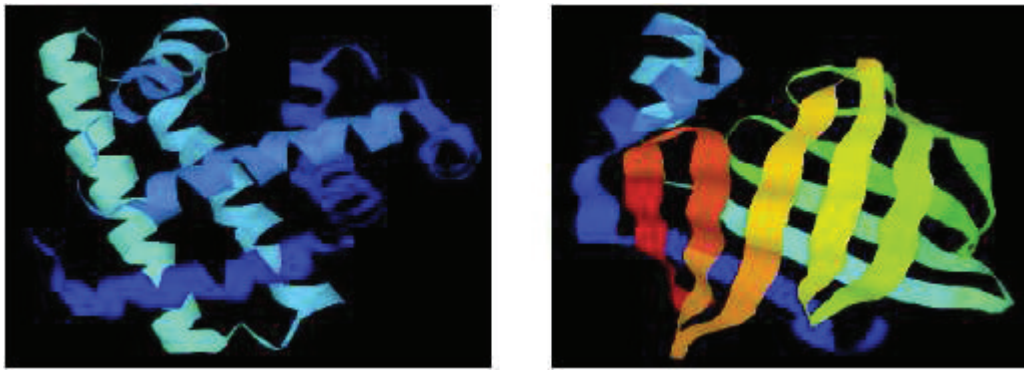


Figure 1.4: Secondary structure of a protein

between the side chains of the amino acids. These interactions between side chains may cause a number of folds, bends, and loops in the protein chain. The interactions could be due to hydrogen bonding, disulfide bond or hydrophobic interactions. It is in this final shape, the proteins perform the function that it was intended to do. Figure 1.5 shows a tertiary structure of Asparagine Synthetase.

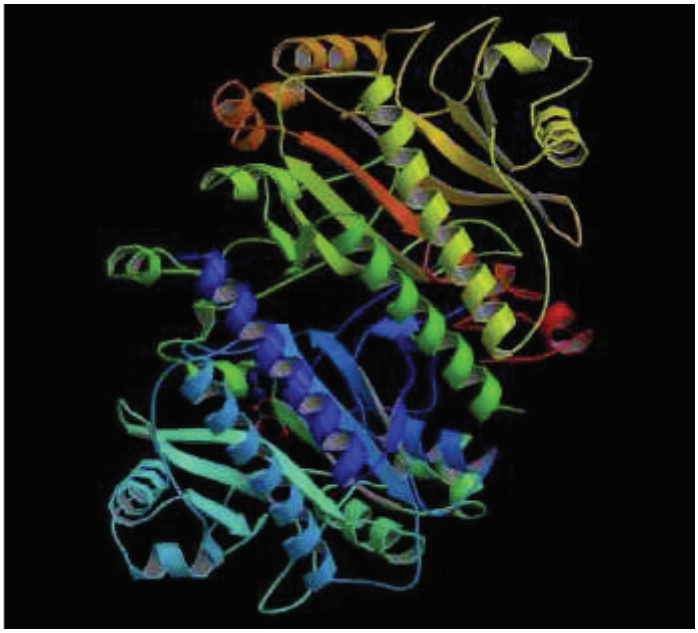


Figure 1.5: Tertiary structure of asparagine synthetase

The fourth level of protein structure, called the *Quaternary Structure,* occurs due to the interaction of two or more polypeptide chains, which associate and form a larger protein molecule. The forces that stabilize a quaternary structure are much the same as those that stabilize the secondary and tertiary structure. Examples of proteins with quaternary structure include hemoglobin, DNA polymerase, and ion channels. Figure 1.6 shows an example of quaternary structure.

Figure 1.6: Quaternary structure of a protein

1.4.3 Protein Structure Prediction

The problem of protein structure prediction lies in determining its tertiary structure from the given sequence (target sequence) of amino acids. As Anfinsen (1973) mentions, the primary sequence of a protein contains the necessary information for determining its conformational arrangement, and thus it is feasible to predict the tertiary structure of a protein based on its sequence alone. This is one of the areas that have been actively researched and still the solution continues to elude the researchers involved. The gap between the protein sequences and its predicted structure continues to increase, highlighting the need for techniques that

could predict the protein structure with considerable accuracy. The growth in the number of protein sequences can be attributed to the various genomic sequencing projects that have been actively undertaken around the world. However, similar results did not surface in the area of protein structure prediction. In order to accelerate the process of structure prediction, researchers have been using the biological knowledge and the available computational techniques to their advantage. Over the years, many protein structure prediction methods have been developed and can broadly be classified into the following three categories, namely, Homology Modeling, Protein Threading and *ab initio* Folding. The first two methods are template based and the third one does not resort to any template.

1.4.3.1 Homology Modeling

Homology Modeling is one of the methods that is known to have a reasonable success in predicting the three dimensional structure of a protein. This method, also known as Comparative Modeling, develops the three dimensional structure of proteins from its sequence based on the structures of homologous proteins, referred to as template. Though, homology primarily means sequence similarity or structural similarity, it is however, not restricted to that. Homologous proteins may also mean that they might have evolved from the same ancestors. Thus the term "homology" is more of qualitative in nature. One important assumption in this method, as mentioned in Chothia & Lesk (1986), is that if two or more proteins are said to be homologous, then their three-dimensional structure are more conserved than their primary sequence. It is this observation that has helped to develop the three-dimensional structure of proteins that has very low sequence similarities.

The first step involved is to determine the homologous protein(s) from available structural databases and identify the sequence similarity. This set of proteins is referred to as the parent template. Next is the sequence alignment phase, wherein the multiple sequence similarities between the target sequences and the homologous proteins are identified. After the known structures are aligned, they are examined to identify the structurally conserved regions from which an average structure, or framework, can be constructed for these regions of the proteins. Variable regions in which each of the known structures may differ in conformation, should be identified so that it could be treated as loops in the finally constructed structure. Once the identification of regions is done, the coordinates of the backbone atoms in the core region is obtained by copying them from the similar atoms in the homologous protein. A side chain rotamer library is used to model the side chain conformations. The variable regions are mostly modeled as loops, while in some cases, if similarity exists, then the coordinates from the homologous protein are copied. In order to improve the accuracy, refinement of the predicted model is done. Various computer programs that helps in structural analysis, such as PROCHECK and 3D-Profiler, can be used. Sometimes, minimizing the energy function is also used as one of the methods to tweak the predicted structure.

1.4.3.2 Protein Threading

Protein Threading, also known as Fold Recognition, is widely used and effective because of its underlying assumption. It is believed that there are a strictly limited number of unique protein folds in nature, mostly as a result of evolution but also due to constraints imposed by the basic physics and chemistry of polypeptide chains. Thus, there is a $70 - 80\%$ chance that a protein which has a similar fold

to the target protein has already been studied either by X-ray crystallography or NMR spectroscopy which can be found in the Protein Data Bank. Hence, these methods are applied to those target sequences which has similar fold as proteins with known structures but do not have homologous proteins.

The basic idea is that the target sequence is compared with the collection of backbone structures of template proteins and a "goodness of fit" score is calculated for each sequence-structure alignment. This goodness of fit is measured mostly in terms of an empirical energy function but many other scoring functions have also been proposed and tried over the years. The most useful scoring functions include both pairwise terms (interactions between pairs of amino acids) and solvation terms. Many different algorithms that incorporate dynamic programming in some form have been proposed for finding the correct threading of a sequence onto a structure.

Jones (1999) reports three problems associated with this method that contribute to its lack of use - slowness of the programs, the requirement of human intervention to interpret the results and the inaccuracy of sequence-structure alignments produced. Though different methods proposed suffer from either of these handicap, the above-mentioned article proposes an algorithm, GenTHREADER, which recognizes protein folds with improved accuracy and reasonably fast. Moreover, the algorithm does not require any kind of human intervention.

1.4.3.3 *Ab Initio* Folding

Though, comparative modeling is the most accurate prediction method, the non-availability of template structures for the majority of proteins makes one to look into alternative methods. For those proteins which do not have templates, the *ab*

initio method serves as the only alternative available now. The *ab initio* method predicts the structure of a protein directly from its given sequence, without resorting to any parental template. This method, however, is limited only to smaller proteins. Major advances in computational power would take this method to the next level.

The thermodynamical hypothesis governing the process of protein folding proposed by Anfinsen (1973) forms the basic principle of *ab initio* methods. The hypothesis states that the native structure of the protein would be at its global free energy minimum. This has paved way for modeling the protein folding problem as an optimization problem. Different versions of the equation that represent the energy of the protein have been derived and used as an objective function which has to be minimized, in order to find its global minimum. Detailed explanation of the energy function can be found in the Section 3.2. This method, which utilizes the energy function of a protein is referred to as the atomic force field approach. Various algorithms have been proposed to locate the minimum point on the complex, nonconvex energy surface.

The other approach, often referred to as the knowledge-based method, relies on simulating the folding pathway to predict the protein tertiary structure. But, due to limited knowledge of the folding pathway and the complex bio-chemical reactions that take place in a fraction of a second, simulation is a highly improbable task. Several algorithmic implementations have been tried and the success stories are very few. During the process of folding, there are a multitude of interactions taking place between the atoms. Since, there are huge number of such interatomic interactions taking place, computational modeling of the system becomes extremely complex. Duan & Kollman (1998), successfully simulated a protein of

36 amino acids for one micro second, with 256 cray processors running for about two months.

1.5 Organization of Thesis

The remainder of the thesis is organized as follows: Chapter 2 is a literature review composed of two distinct parts: Firstly, a literature review of various methods in protein structure prediction is presented. Secondly, various optimization techniques involved in the problem are classified and reviewed accordingly. The problem formulation is described in Chapter 3 along with the protein geometry. Chapter 4 gives a background of interior point methods and discusses the proposed barrier function algorithm. Numerical results for some of the standard test problems are also discussed. Chapter 5 proposes an intrinsic barrier function algorithm to solve the problem of minimum energy determination. The intrinsic barrier function algorithm is applied to the problem of minimum energy conformation of Lennard-Jones clusters to gauge the performance of the algorithm. The proposed algorithms are then applied to polypeptides and the computational experience, along with comparisons to other methods are presented in Chapter 6. An overall conclusion and the scope for future work is detailed in the final Chapter 7.

Chapter 2

Literature Survey

The *ab intio* method of protein structure prediction deals with predicting the native structure of protein given the linear sequence of amino acids. This so-called *protein folding problem* is one of the most challenging problems in the field of bio-chemistry, and as stated in Neumaier (1997), it is a very rich source of interesting problems in mathematical modeling and numerical analysis, requiring an interplay of techniques in eigenvalue calculations, stiff differential equations, stochastic differential equations, local and global optimization, nonlinear least squares, multidimensional approximation of functions, design of experiment, and statistical classification of data. Although, a variety of solution techniques and methods have been proposed, our research focuses on the optimization techniques utilized to solve the problem in question. Hence, the literature review presented here will handle two different topics; Firstly, we will review the studies till date on the problem of protein structure prediction in general and *ab intio* methods in particular. The survey will also cover the different energy functions (force fields) that have been used to calculate the potential energy of a molecule. Secondly, we will give an overview of widely reported optimization solution techniques that have been utilized for solving the problem of protein structure prediction. Focus

will be on both the exact algorithms and heuristics, which would help build our solution method.

2.1 Introductory References

As the area of protein structure prediction is a multi-disciplinary one, it is not uncommon to look for introductory references in this area. Neumaier (1997) serves as an excellent starting point for those from different backgrounds and are willing to further their research in the area of protein structure prediction. For a complete review of the advances in the field of protein structure prediction, the reader is referred to Floudas *et al.* (2006), Floudas (2007) and Zhang (2008). Branden & Tooze (1991) and Brooks *et al.* (1988) are some of the books which provide an introduction to proteins and its structure. Pardalos *et al.* (1994) gives an account of various optimization methods that could be used to solve the energy minimization problem.

2.2 Existing Research on Prediction Methods

In spite of numerous research activities spanning different areas, the problem of protein structure prediction still remains an unsolved one. Since the problem has been in existence for more than three decades, a vast amount of literature pertaining to this problem is available. This section reviews those literature which seems to fit the overall objective of our research.

Ever since Anfinsen (1973) pointed out that the primary sequence of protein contains the necessary information to determine its three-dimensional structure, much attention was devoted to this area. Different classes of methods that were

developed was discussed in Section 1.4.3. This section surveys the existing literature on these methods.

2.2.1 Homology Modeling

Homology modeling, as explained before, deals with the structure prediction of those sequences which has homologous proteins. One of the earlier works in this area, much before Anfinsen's hypothesis, was done by Needleman & Wunsch (1970). They developed a method to determine if significant homology exists between proteins. The protein sequences are compared using a pair of amino acids, each from one protein, using a two-dimensional array. Such methods have been successfully used to identify related proteins. Later, Jurasek *et al.* (1976), successfully built the structure for Streptoyces trypsin-like protein from that of bovine trypsin using the ideas of homology modeling. Greer (1981) modeled eleven structurally unknown proteins which belong to the mammalian serine proteases family. Apart from predicting the structurally conserved region, Greer was also able to find the possible structure of the variable region using the available homologous proteins.

Swindells & Thornton (1991) reviews the methods that were developed until 1991, during which the concentration was only on those proteins which exhibits a considerable similarity in sequence identity. Only later the ideas were extended to those sequences for which the similarity between two proteins were undetectable. Havel & Snow (1991) converted the multiple sequence alignments into distance and chirality constraints and used them in distance calculations. This method provides numerous conformations for the unknown structure, the difference of which can be used as an indicator for the accuracy of predicted structure. The idea

of homology modeling was also extended to the side-chain structure prediction as in Laughton (1994). It calls for a method which involves the comparison of the local environment of each residue whose side-chain conformation is to be predicted with a database of local environments. The method was tested on eight proteins, ranging in size from 46 to 323 amino acid residues, and it predicted 59.8% of all side-chain dihedral angles within ±30 degrees of the crystal structure values.

Markov models were developed by Karplus *et al.* (1998) to find the remote homologs of the protein sequences. The method begins with a single target sequence and iteratively builds a hidden Markov model from the sequence and homologs are found using the HMM for database search. Notredame (2002) advocates multiple sequence alignment methods and identifies the potential strengths and weaknesses of existing methods. Homology modeling generally suffers from the error occurring due to the alignment phase. In order to overcome that John & Sali (2003) has adopted a genetic algorithm approach which starts with a set of initial alignments and then iterates through re-alignment, model building and model assessment to optimize the value of a scoring function. The accuracy in the prediction is said to have increased from 43% to 54%. Tramontano & Morea (2003) provides a recent review of the progress in the area of Homology Modeling.

Some of the research done in this area has been implemented either as automatic or semi-automatic programs to predict the three-dimensional structure of homologous proteins. Šali & Blundell (1993) developed a program called MODELLER, which finds the three-dimensional structure by satisfying the spatial restraints. The spatial restraints are expressed as probability density functions and are derived from the alignment between the sequence and the homologous proteins. SWISS-MODEL, developed by Guex & Peitsch (1997) is a completely

automatic prediction server, which can be used when there is a higher similarity between the sequence and the template. Several variations of the BLAST program has been used to search protein and DNA databases for sequence similarities. Altschul *et al.* (1990) presents one such tool, which is a heuristic that attempts to optimize a specific measure. However, the method has to do a trade-off between the speed and sensitivity. Altschul *et al.* (1997) developed a new heuristic called gapped BLAST that generates gapped alignments and runs at three times the speed of the original. An additional heuristic was also incorporated for automatically combining statistically significant alignments produced by BLAST into a position-specific score matrix and utilize it to search the database. Position-Specific Iterated blast (PSI-BLAST) program was reported to be more sensitive to weak similarities. Sequence Alignment and Modeling Tools, SAMT, a software suite developed by Karplus *et al.* (1998) uses hidden markov models to predict the three-dimensional structure.

2.2.2 Protein Threading

Protein Threading determines the three-dimensional structure of a protein sequence for which homology modeling methods does not provide a reasonable prediction. It is believed that the structure is more conserved than the sequence and that there are only quite a few unique folds compared to the multitude of protein sequences available. While aligning the sequence to the protein structure, the pairwise contact potential can either be ignored or considered. If the pairwise potentials are considered along with the gaps, Lathrop (1994) proved that the threading problem will become NP-hard.

Jones *et al.* (1992), in their work, fitted the target sequences directly onto the backbone coordinates of known protein structures in the full three-dimensional space, incorporating specific pair interactions explicitly. Then they used the dynamic programming approach to predict the final three-dimensional structure. Lathrop & Smith (1994) guarantees to find the optimal threading of a protein sequence using a branch-and-bound algorithm, while including both the pairwise contact potential and amino acid interactions. Lathrop & Smith (1996) considers both the variable-length gaps and the pairwise contact potential, to find the exact global optimum protein threading using the branch-and-bound approach.

Xu & Xu (2000) models the pairwise interaction between the residues as a mean force between residues and the values are derived from already existing structures. They also allow for alignment gaps in the loop regions. Kim *et al.* (2003) suggests running the program without considering the pairwise contact potential in the first stage. The contact potential is inferred from the first stage and later included in the program for further run to globally optimize the scoring function. Xu *et al.* (2004) solves the protein threading problem by adapting branch-and-cut approach. They claim that the linear relaxation of the integer program possesses two well-known cuts in the constraint set and it solves to integral optimal solutions directly. Andonov *et al.* (2004) proposes a mixed-integer programming model to solve the protein threading problem. They decompose the problem into several subproblems and use a efficient parallel algorithm to solve the subproblems.

PROSPECT (PROtein Structure Prediction and Evaluation Computer Toolkit) is a computer program developed by Xu *et al.* (1998) for protein structure prediction. The threading algorithm in PROSPECT employs a divide-and-conquer

strategy and guarantees to find the globally optimal alignment between a query sequence and a template structure, while optimizing a certain energy function. Later Kim *et al.* (2003) developed PROSPECT II, which does not consider the pairwise interaction between the residues initially. It uses a dynamic programming algorithm to solve the alignment problem and only later it includes the interactions as a distance-dependent term in the second phase. PROSPECT II which is much faster than its earlier version did not fair well in the recognition of targets.

Kelley *et al.* (2000) developed 3D-PSSM (three-dimensional position specific scoring-matrix) which utilizes multiple sequence profile to recognize the fold targets. It actually calculates three different alignments between the target and the template and updates the resulting values in a scoring matrix. A dynamic programming algorithm is used to evaluate the optimal alignment. Xu *et al.* (2003) adapted a integer programming approach in their program, RAPTOR: RAPid Protein Threading by Operations Research technique. A branch-and-bound approach was used to solve the linear relaxation model which accounted for both the pairwise contact potential and the gapped penalties. The CAFASP3 evaluation ranked RAPTOR as the No.1 prediction server among individual prediction servers in terms of the recognition capability and alignment accuracy.

The success of protein threading models depends on the recognition of correct templates and generation of accurate sequence-template alignments. In case of protein with low-homology, Peng & Xu (2010) presents a profile entropy scoring function for low-homology protein threading. While most of the protein threading methods use only one template, Peng & Xu (2011) uses multiple template to improve modeling accuracy. The use of multiple templates helps to improve

pairwise sequence-template alignment accuracy, thereby increasing the predictive correctness of the model.

2.2.3 *Ab Initio* Folding

Given the linear sequence of amino acids, the *ab initio* method predicts the native conformation of the protein without any aid from external databases or structural templates. The basic idea in this method lies in searching the entire conformational space of the protein to identify the most stable state. Searching the entire conformational space for proteins with large number of residues is a daunting task even with the computational capability available today. Hence several techniques in this area aim to reduce the search space or reformulate the problem in such way that it can identify the most favorable state.

In order to identify the native structure of the protein one has to minimize its energy function as proposed by Anfinsen (1973). Any of the energy functions discussed in Section 3.2.1 is used to find the native state of the protein considered. However, the energy surface is highly complex and its nonconvex nature makes it one of the hardest problems to solve. Caution is required while using optimization techniques as it may converge to a local optimum point rather than the global optimum. Several global optimum methods have been developed to counter this problem. Since the *ab initio* methods mostly employ optimization techniques, the literature in this area are presented in the Section 2.3 which introduces and presents the work carried out in the area of mathematical optimization pertaining to the problem of protein structure prediction.

2.3 Optimization Methods

With the advent of high speed computers, optimization techniques have become popular among computational biologists. Depending on the problem type, optimization methods help to locate optimal or near-optimal solutions of the problem being pursued. In the area of computational biology, the formulated problems are often nonlinear, and hence global optimization methods tend to be highly relevant.

Global optimization addresses the computation and characterization of global optima of nonconvex functions constrained in a specified domain Floudas (2000). A general global optimization problem statement provided by Pintér (1996): given a bounded set $\mathcal{D}$ in the real n-space, $\mathbb{R}^n$ and a continuous function $f : \mathcal{D} \to \mathbb{R}$, find

$$\begin{aligned} &\min\ f(x) \\ &s.t.\ x \in \mathcal{D}. \end{aligned} \tag{2.1}$$

The general problem statement shown in (2.1) covers almost all specific global optimization problems. Characterizing the global optima for the problem depends very much on the complexity of the function f and the constraint set $\mathcal{D}$. It is the nature of the function and that of the constraint set that dictates the technique to be used. Floudas (2000) details the theoretical and algorithmic advances in deterministic global optimization whereas Pétrowski & Taillard (2006) describes the various metaheuristics available to solve the problem.

2.3.1 Optimization Techniques for Protein Structure Prediction

The primary idea of this section is to elucidate the techniques that have attracted much attention for solving the potential energy minimization problems particularly in the area of *ab intio* methods of Protein Structure Prediction. As mentioned before, these problems often have been formulated as optimization problems to determine the lowest energy conformation. The nonconvex potential energy equation which is used as the objective function for the problem makes it difficult to develop solution techniques that could locate the true global minimum. However, existing techniques have been employed to find good solution(s), if not global ones. This section will review some of the more popular techniques that have been used to handle the problem of protein structure prediction.

2.3.1.1 Simulated Annealing

The dauntingly complex conformational space of large-scale optimization problems inspired Kirkpatrick *et al.* (1983) to develop the method of simulated annealing, which has much in common with the physical annealing process. Heating a metal and cooling it slowly, gives it a uniform crystalline state, which is believed to minimize its free energy (global minimum). One of the earliest applications of simulated annealing in structure prediction can be attributed to Wilson & Cui (1988), who used the idea in their computer program to predict the structure of peptide systems. Later the method was successfully applied to the "dipeptide models" of all the 20 natural amino acids by Wilson & Cui (1990). They produced a Ramachandran-type plot on ϕ/ψ scale tracing the random walk for each run only to find that as the temperature is lowered, the molecule spent more time

in the lowest energy regions making the annealing process converge to the global minimum.

Huber & McCammon (1997) propose a weighted-ensemble simulated annealing technique which uses multiple copies of the system that move independently. As the temperature is lowered, copies that are trapped in high energy system are deleted and those which move in a favorable direction towards the global minimum are duplicated. This facilitates parallel computation and hence lesser computational time. Liu & Beveridge (2002) adapts a similar approach, in which a number of replicas of the initial structure is subjected to individual simulated annealing process. All the back bone torsion angles were allowed to move with equal probability. Fragment assembly methods to predict protein structures often employ simulated annealing as in Rohl *et al.* (2004). The technique was used to randomly combine the identified fragments to form a compact structure which was then minimized using a scoring function. An application of generalized simulated annealing algorithm on *ab initio* protein structure prediction is discussed in Melo *et al.* (2012). The stochastic search algorithm that they employ depend on utilizing the long-range interactions to predict the protein structure.

2.3.1.2 Genetic Algorithm

Genetic algorithm developed by Holland (1973), on the lines of biological evolution, allows mutations and crossing over among the candidate solutions in a hope to derive better ones. Though the genetic algorithms were not employed for tertiary structure prediction initially, Tuffrey *et al.* (1991) used it to assign side-chain rotamer conformations with the known fixed backbone conformation of a protein. Blommers *et al.* (1992) used it to analyze the conformations of

a dinucleotide photodimer. Sun (1993) used genetic algorithm to successfully fold the protein melittin and apamin with a root mean square error of 1.66 Å. Simultaneous optimization of the conformation population was done with the probability set to unity for all the conformations to be replicated in order to achieve maximal accessible search. Pedersen & Moult (1995) applied the ideas of gentic algorithm-based search methods to fold small polypeptides and protein fragments using double crossovers. A 200-step Monte Carlo simulation for each member of the running population between crossovers was performed. Khimasia & Coveney (1997) looks at the genetic algorithm design for the problem of protein structure prediction. For this purpose they use a modified version of Simple Genetic Algorithm Goldberg (1989) and used the Random Energy Function Derrida (1980) as the objective function to be minimized. They postulate that high resolution building blocks attainable by multi-point crossovers and a local dynamics operator to fine tune good conformations are required of the genetic algorithms used to predict the protein structure. The genetic algorithm approach without much change was adapted by Schneider (2002) in order to identify the conformationally invariant and flexible molecules of a protein rather than predicting the actual structure. John & Sali (2003) used genetic algorithm in their program MODELER which was fashioned on the five genetic algorithm operators, namely, single point crossover, two point crossover, gap insertion, gap deletion, and gap shift. Kondov (2013) uses particle swarm optimization to study the low-energy conformations of peptides by applying periodic boundary conditions to the search space.

2.3.1.3 Other Methods

The branch-and-bound method, widely used to solve integer programming problems has numerous applications in a variety of areas. In the area of our concern, it has been mainly used to solve formulations that are encountered in the protein threading problem rather than the *ab initio* methods. In the past, Lathrop & Smith (1994) used this technique to model the pairwise contact potential of the protein threading problem. They divide the entire search space into subsets of possible threading sequences and using a tight lower bound developed, each and every set is scored only to further divide the set which gives the infimum score. Androulakis *et al.* (1995) proposed the much popular and widely adapted variation of the branch-and-bound technique called αBB. The method develops a convex lower bounding function by the addition of a convex separable quadratic term for each variable to the objective function. αBB attains a finite $\epsilon-$convergence to the global minimum by continuous dividing and sub-dividing of the search space based on the lower bound. Maranas *et al.* (1996) exploited this technique to predict the structure of oligopeptides by *ab inito* methods using the ECEPP/3 energy function.

Lathrop & Smith (1996) used branch-and-bound for gapped protein alignment with five different scoring functions, to rank the sequences according to the score calculated. Eyrich *et al.* (1999), in their *ab initio* methods, adapted a variation of αBB algorithm. In fact, they propose three variations - a different quadratic smoothing function, using inter-residue distance instead of dihedral angles as search space and annealing approach to smooth the potential of the volume terms excluded due to repulsion. Moreover, a Monte Carlo minimiza-

tion is done before invoking the αBB algorithm. Lin *et al.* (2002) utilized the branch-and-bound technique to assign NMR peaks to the protein backbone, a key step in studying protein NMR structure. Das *et al.* (2003) formulates the protein structure prediction problem as a nonlinear constrained minimization problem. They use a hybrid global optimization method which combines the α-Branch and Bound approach with the conformational space annealing method.

McAllister & Floudas (2010) applies hybrid methods for large-scale unconstrained optimization of protein models such as Bovine Pancreatic Trypsin Inhibitor(BPTI) and Rnase. A basin-hopping approach to global optimization was used by Hoffmann & Strodel (2013). However, they utilize additional constraints by imposing NMR shift restraints. Bhattacharya & Cheng (2013) propose a method to refine protein structures by bringing the low-resolution predicted models close to high-resolution native structures. This is achieved by optimizing the hydrogen bonding network and applying the atomic-level energy minimization on the optimized model. A parallel implementation of protein structure prediction has been discussed in Tyka *et al.* (2012). Mirzaei *et al.* (2012) discusses the use of energy minimization techniques in protein - protein docking. They utilize LBFGS quasi-Newton method for local optimization since it uses only gradient information to obtain second order information about the energy function. Rodrigues *et al.* (2012) also propose a fast method for protein structure refinement using knowledge-base potential of mean force.

2.3.1.4 Interior-Point Methods

Interior-Point methods, unlike simplex method, travel from the starting point and move through the feasible space in search of the optimal point. It enjoys a

polynomial-time convergence and has been frequently used to solve nonlinear and nonconvex problems. However, the application of these methods in the area of protein structure prediction is virtually non-existent. MELLER *et al.* (2002) addresses the problem of feasibility while modeling the protein threading problem as a linear program. They determine the largest number of constraints that could be satisfied with the available set of data using the method of analytic centers. MaxF heuristic, that they propose, identifies those constraints that are hard to satisfy from the easily satisfiable ones. Though not a direct implementation, Wagner *et al.* (2004) have used interior-point methods to solve the linear programming formulation of a protein threading problem. They have used a publicly available software, PCx, which utilizes the primal-dual predictor-corrector method. Other than these two works, to the best of our knowledge, we are not aware of any other research done in the application of interior-point methods to the problem of protein structure prediction, especially in *ab initio* methods.

2.4 Conclusion

A detailed review in the area of protein structure prediction and that of mathematical techniques to solve optimization problems pertaining to the problem of interest has been given. Studies show that mathematical programming techniques have gained popularity over the years in solving problems that are in the interest of the biologists. Linear Programming and Integer Programming approach has been generously borrowed to tackle the problem of protein threading. Simulated Annealing, Genetic Algorithm and Branch-and-Bound techniques have gained the most attention of researchers working on *ab initio* methods. However, interior-point methods, for unknown reasons has never been thought of in this particular

direction. It is this finding that gives us the scope and iterates the significance of our research.

Chapter 3

Problem Description

The problem of protein structure prediction has been modeled and solved using different methods. Various algorithms for database searching in case of homology modeling, adaptation of optimization techniques to optimize a scoring function in case of protein threading and a variety of optimization solution techniques while dealing with the *ab initio* methods have been proposed and are reviewed in Chapter 2. This chapter describes the protein geometry and gives a detailed account of the potential energy equation of proteins. The problem formulation for the *ab initio* method of protein structure prediction is also presented.

3.1 Protein Geometry

The complete structure of a protein can geometrically be described by a three-dimensional vector assigned to each and every atom in the structure. The mathematical description that follows in this section is based on Maranas *et al.* (1996). Let r_i be the vector representing the position of the i^{th} atom, given as in (3.1).

$$r_i = \begin{pmatrix} x_i \\ y_i \\ z_i \end{pmatrix}, \qquad i = 1, ..., N, \tag{3.1}$$

where N is the total number of atoms in the protein molecule. The bond length between two consecutive atoms i, j is given by the bond vector, r_{ij} as in (3.2). The bond length between two consecutive atoms i, j is given in (3.3).

$$r_{ij} = \begin{pmatrix} x_j - x_i \\ y_j - y_i \\ z_j - z_i \end{pmatrix}, \tag{3.2}$$

$$|r_{ij}| = \sqrt{(x_j - x_i)^2 + (y_j - y_i)^2 + (z_j - z_i)^2}. \tag{3.3}$$

The bond vectors, bond angles and the dihedral angles in a protein are denoted by the same notation throughout the protein community in order to facilitate clarity of thought and communication among different researchers. Figures 3.1 and 3.2, give a pictorial representation of a protein structure along with its bond vectors, angles and dihedrals. θ_{ijk} is the covalent bond angle formed between the

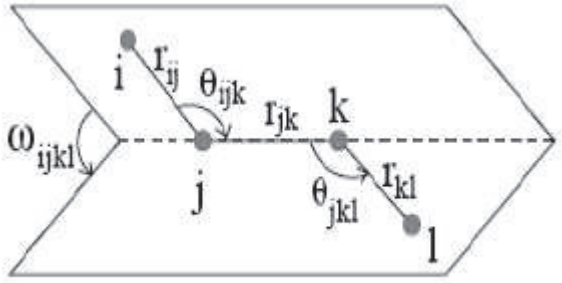


Figure 3.1: Bond vectors and bond angles taken from Maranas *et al.* (1996)

vectors r_{ij} and r_{jk} and can be computed using the dot product and cross product of the associated bond vectors as given in (3.4) and (3.5).

$$\cos(\theta_{ijk}) = \frac{r_{ij}.r_{jk}}{|r_{ij}||r_{jk}|}, \tag{3.4}$$

$$\sin(\theta_{ijk}) = \frac{r_{ij} \times r_{jk}}{|r_{ij}||r_{jk}|}. \tag{3.5}$$

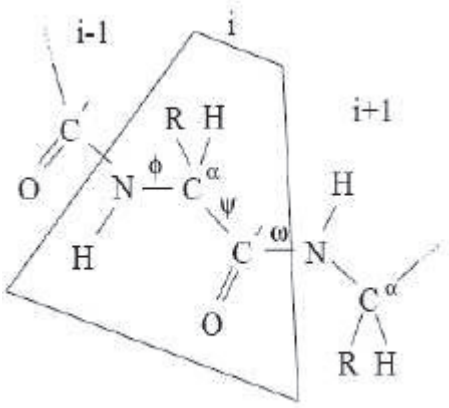


Figure 3.2: Dihedral angles in a protein, taken from Maranas *et al.* (1996)

$\omega_{ijkl} \in [-180, 180]$ is the dihedral angle, which is nothing but the angle between the atom i and the plane formed by the atoms j, k, l. The dihedral angle can also be thought of as the angle formed between the normals of the two planes formed by the atoms i, j, k and j, k, l. The functional form used to calculate the dihedral angle is shown in (3.6) and (3.7). Sometimes, the complementary torsion angle, $180^\circ - \omega$, is also used to measure the relative orientation between a chain of atoms. Apart from the bond lengths, bond angles and dihedral angles, used to determine the structure of a protein, out-of-plane bending or improper torsion angles, $\tau = \sphericalangle(i - j - k - l)$ is also used when the situation warrants.

$$\cos(\omega_{ijkl}) = \frac{(r_{ij} \times r_{jk}).(r_{jk} \times r_{kl})}{|r_{ij} \times r_{jk}||r_{jk} \times r_{kl}|}, \tag{3.6}$$

$$\sin(\omega_{ijkl}) = \frac{(r_{kl} \times r_{ij}).r_{jk}|r_{jk}|}{|r_{ij} \times r_{jk}||r_{jk} \times r_{kl}|}. \tag{3.7}$$

Various dihedral angles in a protein follow a standard nomenclature. As can be seen from Figure 3.2, the dihedral angle between the normals of the planes formed by the atoms $C'_{i-1}N_iC_{\alpha,i}$ and $N_iC_{\alpha,i}C'_i$ respectively is called ϕ_i, where

$i-1$ and i are two adjacent amino acid residues. The angle formed between the planes$R_iC_{\alpha,i}C'_i$ and $C_{\alpha,i}C'_iN_{i+1}$ respectively is called ψ_i, where i and $i+1$ are two adjacent amino acid residues. ω_i is the dihedral angle defined by the planes $C_{\alpha,i}C'_iN_{i+1}$ and $C'_iN_{i+1}C_{\alpha,i+1}$. The letter χ_i is used to denote the dihedral angle associated with the side groups R_i. Though the bond lengths, bond angles and dihedral angles are used to describe the structure of a protein, it often over determines the structure. Under biological conditions, as stated in Maranas *et al.* (1996), the bond lengths and bond angles are fairly rigid and it can be assumed to be fixed at their equilibrium values. Thus, the assumption manifests, that only the backbone dihedral angles is enough to fully determine the geometrical shape of the protein and it also helps in reducing the problem size when compared to that using cartesian coordinates for representing the protein structure.

3.2 Protein Force Fields

In order to adapt any of the above-said methods, a scoring function is required to quantitatively evaluate the appropriateness of the predicted structure. The force field or the potential energy equation developed is a popular candidate among the several scoring functions available. This section gives an overview of the various force fields and their components.

Theoretical studies of biological molecules permit the study of the relationships between structure, function and dynamics at the atomic level. Any study of biological systems as such involves many atoms and hence dealing with them at the electron level becomes much difficult and sometimes may not be feasible. In such cases, the problem becomes more tractable when empirical potential energy functions, called force fields, are used. Effective application of force fields is based

on the accuracy of the developed function. There are numerous approximations that goes into the development of the empirical function and thereby paving way for different forms of empirical functions. This chapter intends to describe the functional form of the force fields used for the study of proteins.

In order to derive the empirical form of the potential energy of a protein, researchers adapt a classical description of molecules. The atoms are considered to be the smallest particle in the calculations. Proteins, generally consist anywhere from 500 to 500,000 or more atoms. Apart from the interaction between these atoms, one should also consider the environment surrounding the protein and the atom's interaction with its environment. If one should consider all the interactions, the problem presents itself as dauntingly complex. However, assumptions such as protein folding in vacuum, absence of long range interactions, a simple mathematical function representing the energy of the protein are commonly used in developing force field equations.

3.2.1 Survey of Energy Functions

The static forces in a molecule can fully be determined by *V(x)* as given in (3.11). Hence, modeling a molecule simply amounts to specifying the contribution of the various interactions to the potential. These models also called as force fields derive their final form from molecular dynamics and different versions of them are available mainly due to the difference in the assumptions that are involved. This section surveys the various force fields that are widely used.

CHARMM developed by Mackerell *et al.* (1998), is an all-atom empirical energy function that has gone through several versions, the latest of them being CHARMM22 and CHARMM27. CHARMM27 has been specifically optimized

for simulating DNA, however, both the versions are almost the same when used for purely protein systems. AMBER force field developed by Cornell *et al.* (1995) emphasizes on the accurate representation of the electrostatics and simple representation of bond and angle energies, while optimizing the electrostatic and van der Waals parameters for condensed phase simulations. GROMOS force field was developed in conjunction with the GROMACS program package by Scott *et al.* (1997). GROMOS force field was mainly designed for proteins, nucleotides, or sugars in aqueous or apolar solvents using the concept of united atoms. It was later extended to an all-atom model applicable only to sugars. Nemethy *et al.* (1992) developed ECEPP/3, the latest and the updated version of the first ECEPP developed by Momany *et al.* (1975). The model developed empirical interatomic potentials for calculating the energetically most favorable conformations of polypeptides and proteins.

Though the above-mentioned force fields used molecular dynamics simulation and parameter optimization, there were also efforts by others to develop force fields using different techniques. Knowledge-based force field was first developed by Tanka & Scheraga (1976) who used Boltzmann distribution to derive them. Later, Lathrop *et al.* (1998) used a Bayesian network approach to deduce the energy function of a protein system while Maiorov & Crippen (1992) used a linear programming approach for determining the force field. With the evolution of so many force fields, high quality decoys were are also developed to test the effectiveness of a force field.

3.2.2 Potential Energy Equation

The energy, V, of a protein is often expressed as a function of its atomic position, R, of all the atoms in the system. The position of the atoms are generally expressed in terms of cartesian coordinates. The total energy of a protein system is thought of as contributions from its bonded terms and non-bonded terms as shown in (3.8) below:

$$V(R) = E_{bonded} + E_{non-bonded}. \tag{3.8}$$

The energy due to atoms that are bonded, E_{bond}, takes into account the interactions between the atoms that are involved in the formation a bond, angle or a dihedral plane. Whereas, the energy derived through non-bonded atoms, $E_{non-bonded}$, represents the interactions due to the partial atomic charges on the atoms and the van der Waals interactions. The energy contributions from the non-bonded interactions are generally much higher when compared to that of the bonded interactions. (3.9) and (3.10) elucidate the above discussion in an empirical fashion.

$$E_{bonded} = E_{bond} + E_{angle} + E_{dihedrals}, \tag{3.9}$$

$$E_{non-bonded} = E_{vanderWaals} + E_{electrostatic}. \tag{3.10}$$

A general form of the equation representing the potential energy, V, of a system as a function of its structure, r, as given in Ponder & Case (2003), is provided below in (3.11).

$$\begin{aligned} V(r) = &\sum_{bonds} k_b(b - b_0)^2 + \sum_{angles} k_\theta(\theta - \theta_0)^2 + \sum_{torsions} k_\phi[cos(n\phi + \delta) + 1] \\ &+ \sum_{\substack{nonbonded \\ pairs}} \left[\frac{q_i q_j}{r_{ij}} + \frac{A_{ij}}{r_{ij}^{12}} - \frac{C_{ij}}{r_{ij}^{6}}\right], \end{aligned} \tag{3.11}$$

where k_b, k_θ, k_ϕ are the bond, angle, and dihedral angle force constants respectively; b, θ, ϕ are the bond length, bond angle and dihedral angle, respectively, with the subscript zero representing the equilibrium terms for the corresponding terms. The first three summations run over bonds (1-2 interactions), angles (1-3 interactions) and dihedral (1-4 interactions). The last summation term runs over all the atom pairs that are involved in the non-bonded interactions. Both, the coulombic or electrostatic and van der Waals interactions contribute to the non-bonded interactions. The constants, q_i, q_j correspond to the partial charges on the atoms and r_{ij} denotes the Euclidean distance between the atoms i and j. Constants, A_{ij} and C_{ij} represent the minimum interaction distance between the atoms.

As mentioned earlier, due to different objectives and hence differing assumptions a variety of force fields have been developed. Each and every force field, thus developed adapt a slightly different empirical form. The most popular force fields that are efficient and currently in use are ECEPP, MM2, ECEPP/2, CHARMM, AMBER and GROMOS to name a few. For explanations and references of these force fields in the literature, refer to Section 3.2.1.

3.3 CHARMM Potential Energy Function

For the purpose of our research, we are using the empirical form of the CHARMM potential energy function, developed by Mackerell *et al.* (1998) as given in (3.12).

$$\begin{aligned} V(r) = & \sum_{bonds} K_b(b-b_0)^2 + \sum_{UB} K_{UB}(S-S_0)^2 + \\ & \sum_{angles} K_\theta(\theta-\theta_0)^2 + \sum_{dihedrals} k_\phi(1+cos(n\phi-\delta)) + \\ & \sum_{\substack{nonbonded \\ pairs}} \left\{ \epsilon \left[\left(\frac{R_{min_{ij}}}{r_{ij}} \right)^{12} - \left(\frac{R_{min_{ij}}}{r_{ij}} \right)^{6} \right] + \frac{q_i q_j}{\epsilon_1 r_{ij}} \right\}, \end{aligned} \tag{3.12}$$

As mentioned in (3.9), the CHARMM potential energy function is calculated as the sum of interaction energies caused by both bonded and nonbonded terms. The following two equations explicitly mention the components involved in both the bonded and nonbonded interaction terms as given by the CHARMM energy function.

$$E_{bonded} = E_{bond} + E_{angle} + E_{improper} + E_{dihedrals}, \tag{3.13}$$

$$E_{nonbonded} = E_{vdW} + E_{elec}. \tag{3.14}$$

3.3.1 Bonded Interactions

The first term in the CHARMM energy equation, E_{bond} represents the interaction between two atoms separated by a covalent bond and is often referred to as either 1,2-interactions or 1,2-pairs. If b is the actual bond length and b_0 is the ideal bond length, the following equation approximates the energy due to displacement from its ideal bond length.

$$E_{bond} = \sum_{bonds} K_b(b-b_0)^2, \tag{3.15}$$

where K_b is a force constant. Both K_b and b_0 are specific to the atoms participating in the bond. Similarly, the bond angle θ may deviate from its ideal bond angle θ_0 and the energy is calculated as shown below

$$E_{angle} = \sum_{angles} K_\theta(\theta - \theta_0)^2, \tag{3.16}$$

where K_θ is a force constant specific to the atoms involved in the angle formation. It may be noted here that the three atoms are separated by two covalent bonds and is referred to as either 1,3-interactions or 1,3-pairs. The potential function which describes the interaction energy of four atoms separated by three covalent bonds (1,4-interactions) is

$$E_{dihedrals} = \sum_{dihedrals} K_\phi(1 + cos(n\phi - \delta)), \tag{3.17}$$

where K_ϕ is a force constant and ϕ is the dihedral angle. The potential due to dihedrals is assumed to be periodic and hence it is modeled using a cosine function with periodicity n and phase δ. The equations (3.18) and (3.19) represent the Urey-Bradley term and the improper term. Energy due to Urey-Bradley is derived out of the distance that separates the three atoms that are involved. E_{imp} is a term used to maintain chirality and planarity.

$$E_{UB} = \sum_{UB} K_{UB}(S - S_0)^2, \tag{3.18}$$

$$E_{imp} = \sum_{impropers} K_{imp}(\varphi - \varphi_0)^2, \tag{3.19}$$

where K_{UB} and K_{imp} are corresponding force constants. S is the Urey-Bradley 1,3-distance and φ is the improper dihedral angle, with the subscript zero representing the equilibrium values for the respective terms.

3.3.2 Nonbonded Interactions

As shown in (3.14), the nonbonded interaction energy consists of van der Waals and electrostatic interaction term . The van der Waals interaction term models the potential energy of two interacting atoms based on the distance of separation. Lennard-Jones 6-12 potential, proposed by Sir John Edward Lennard-Jones is often used to model the van der Waals interaction and is given by the following equation:

$$E_{std-vdW} = 4\epsilon \left[\left(\frac{\sigma}{r}\right)^{12} - \left(\frac{\sigma}{r}\right)^{6} \right], \tag{3.20}$$

where $E_{std-vdW}$ is the intermolecular potential between two atoms, ϵ is the well depth, r is the distance of separation between the atoms involved and σ is the distance at which the intermolecular potential between the two particles is zero. Both attraction and repulsion between atoms involved are empirically described by (3.20). Figure 3.3 shows the intermolecular potential energy as a function of r. At short distances, the first term in (3.20) dominates thereby modeling the repulsion between atoms when they are brought very close to each other. At longer distance, the second term dominates to mimic the force of attraction between atoms. Thus, the van der Waals equation in (3.20) leads to an equilibrium value where the minimum of (3.20) is reached at $r = \sigma$.

In CHARMM energy function a modified Lennard-Jones 6-12 potential is used to model the van der Waals energy component caused by interactions of non-bonded atoms. The empirical form of the modified Lennard-Jones 6-12 potential is shown below

$$E_{vdW} = \sum_{\substack{nonbonded \\ pairs}} \epsilon \left[\left(\frac{R_{min_{ij}}}{r_{ij}}\right)^{12} - \left(\frac{R_{min_{ij}}}{r_{ij}}\right)^{6} \right], \tag{3.21}$$

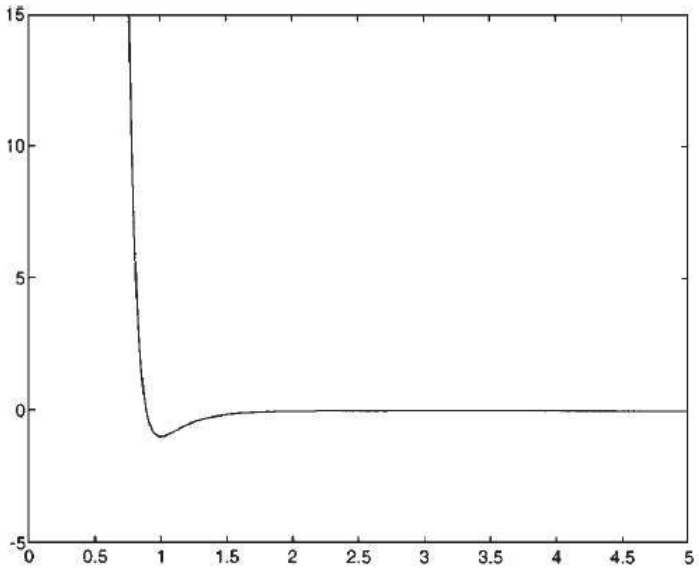


Figure 3.3: Lennard-Jones potential, taken from Gockenbach *et al.* (1997)

where and $R_{min_{ij}}$ is the distance at Lennard-Jones minimum. r_{ij} is the distance between two atoms i and j. The Lennard-Jones parameters between pairs of different atoms are obtained from the Lorentz-Berthelodt combination rules, in which ϵ_{ij} values are based on the geometric mean of ϵ_i and ϵ_j and $R_{min_{ij}}$ values are based on the arithmetic mean between $R_{min_i}$and R_{min_j} (Mackerell *et al.*, 1998). This rule has been designed to reduce the number of parameters associated with the overall energy function.

The electrostatic potential between a pair of atoms is modeled by Coulomb potential as follows

$$E_{elec} = \sum_{\substack{nonbonded \\ pairs}} \frac{q_i q_j}{\epsilon_1 r_{ij}}, \tag{3.22}$$

where q_i and q_j are the partial charges assigned to atoms i and j and ϵ_1 is the effective dielectric constant. In order to obtain a balanced parametrization, particularly for the peptide group, ϵ_1 is set to 1. The partial charges of the

atoms approximate the electrostatic potential of the electron cloud. Thus the energy is a consequence of the distortion of electronic distribution which generates induced electric moments. However, the Coulomb interaction is valid only for a homogeneous dielectric medium.

Thus the total potential energy of a molecule is calculated as the sum of all the energy components described in equations (3.15) to (3.22), as given below

$$E = E_{bond} + E_{angle} + E_{dihedrals} + E_{UB} + E_{imp} + E_{vdW} + E_{elec}. \tag{3.23}$$

Nonbonded interaction terms included for all atoms are separated by three or more covalent bonds. An approximation included in the CHARMM model is that it only considers the pairwise interaction potential of atoms and it does not take into account the simultaneous interaction of three or more atoms.

3.4 Problem Formulation

The thermodynamical hypothesis proposed by Anfinsen (1973) forms the basic premise on which all the problem formulations, especially *ab inito* methods, are based on. Simply stated, the formulation involves the minimization of a free energy function which captures the potential energy interactions of a protein system. Mathematically speaking, it is a nonconvex nonlinear optimization (minimization) problem. Though the structure of the problem formulation has not varied over the years, the difference lies in the solution methods that have been proposed.

The objective function of the problem requires an empirical form of an energy function which has to be minimized. Various potential energy functions have been developed and are discussed in Section 3.2.1. For the purpose of our research we

are using CHARMM energy function for its popularity among the protein community and its efficient parametrization (Mackerell *et al.*, 1998). The CHARMM energy function stated in (3.12) is restated here for clarity. The notations and the variable definitions stay the same here.

$$\begin{aligned} V(r) = & \sum_{bonds} K_b(b-b_0)^2 \; + \sum_{angles} K_\theta(\theta-\theta_0)^2 \; + \\ & \sum_{UB} K_{UB}(S-S_0)^2 \; + \sum_{impropers} K_{imp}(\varphi-\varphi_0)^2 \\ & \sum_{dihedrals} k_\phi(1+cos(n\phi-\delta)) \; + \\ & \sum_{\substack{nonbonded \\ pairs}} \left\{ \epsilon \left[\left(\frac{R_{min_{ij}}}{r_{ij}} \right)^{12} - \left(\frac{R_{min_{ij}}}{r_{ij}} \right)^{6} \right] + \frac{q_i q_j}{\epsilon_1 r_{ij}} \right\}. \end{aligned} \tag{3.24}$$

The CHARMM energy function described in (3.24) computes the potential energy as a function of cartesian coordinates of atoms. In case of problems pertaining to protein structure, the energy function is generally used as a function of internal coordinates, viz. bond lengths, bond angles and dihedral angles. Such a representation also reduces the number of variables involved when compared with the model using cartesian coordinates of atoms for representation. Cartesian coordinates representation requires three variables for each atom in the protein structure which increases the number of variables in the model. The general assumption in the bio-chemistry community is that the energy required to perturb the bond length and the bond angles from their equilibrium values is relatively large and hence the parameters can be assumed to have a fixed value (Byrd *et al.*, 1996). We, in our research espouse the same assumption, thereby formulating the optimization problem as a function of dihedral angles alone. Hence, the objective

function that we consider for our research is stated in (3.25).

$$V(r) = \sum_{dihedrals} k_\phi(1 + cos(n\phi - \delta)) + \sum_{\substack{nonbonded \\ pairs}} \left\{ \epsilon \left[\left(\frac{R_{min_{ij}}}{r_{ij}} \right)^{12} - \left(\frac{R_{min_{ij}}}{r_{ij}} \right)^{6} \right] + \frac{q_i q_j}{\epsilon_1 r_{ij}} \right\}. \tag{3.25}$$

The first four terms of (3.24), which approximates the energy due to displacement from their equilibrium value is ignored in (3.25). Based on the above assumptions and the definitions, the energy minimization problem can be sated as follows:

$$\text{Minimize } V(\Phi)$$

$$\text{Subject to:}$$

$$\begin{aligned} -\pi \quad \leq \quad \phi_{ij} \quad \leq \quad \pi, \quad & i = 2, ..., N-1, \\ & j = 3, ..., N, \\ & j = i + 1, \\ \Phi \in \Re^{N-2}. & \end{aligned} \tag{3.26}$$

V is the expression for the total potential energy of the protein as a function of its dihedral angle as given in (3.25). $\Phi = \{\phi_{ij} : i = 2, ..., N-1,\ j = 3, ..., N,\ j = i+1\} \in \Re^{N-2}$ is a vector of dihedral angles around the atoms i and j, while N is the total number of atoms in the protein considered. As opposed to what is generally followed in the literature, for instance Maranas *et al.* (1996), here we adapt a single variable representation for the dihedral angles irrespective of the atom type involved. Generally, the variable ϕ_i is used to represent the torsion around $C'_{i-1}-N_i-C_{\alpha,i}-C'_i$, ψ_i to represent the torsion around $R_i-C_{\alpha,i}-C'_i-N_{i+1}$ and χ_i to denote the torsion around side chain components, where i represents the amino acid residues. In the formulation (3.26), we have used the sequential atomic numbers, denoted by i and j, to differentiate the various dihedral angles.

This, we feel, is only a matter of convenience and has no effect, whatsoever, on the problem as such. The objective function, V, accounts for both the bonded and the non-bonded interactions. However, in some cases non-bonded interactions consider only those atoms that are separated only by two other atoms. Long-range interactions are not considered owing to the fact that the potential energy due to such long-range interactions is considerably low as atoms become farther apart.

The energy function V, is a nonconvex function of dihedral angles. Therefore, a number of local minima exists even for molecules of modest size. These local minima correspond only to the metastable states of the molecules (Maranas *et al.*, 1996). Hence the solution method developed should identify the energetically most favorable state, bypassing the multitude of local minima points.

Chapter 4

Interior Point Methods

A number of algorithms which involve perturbation of sufficiency conditions for a point to be a local constrained minimum of a nonlinear programming problem (NLP) has been proposed. The term interior point method was originally proposed by Fiacco & McCormick (1968) to describe any algorithm that computes a local minimum of a nonlinear programming problem by solving a sequence of unconstrained minimization problems. This method searches for the local minimum within the interior of the feasible region of the NLP problem.

4.1 Interior Point Unconstrained Minimization

Consider the following inequality constrained problem

$$\begin{aligned} &\text{minimize } f(x) \\ &\text{subject to } g_i(x) \geq 0, \quad i = 1, ..., m, \end{aligned} \tag{4.1}$$

where $f(x)$ and $g_i(x)$ are C^2 functions. Fiacco and McCormick propose to solve the problem (4.1) as a series of unconstrained minimization problems by defining two scalar valued functions $I(x)$ and $s(r)$ with specific properties as illustrated below.

Definition 4.1. *$I(x)$ is a scalar valued function with the following properties:*

Property 1 $I(x)$ is continuous in the region $R^0 = \{x \mid g_i(x) > 0, i = 1, \ldots, m\}$.
Property 2 If $\{x^k\}$ is any infinite sequence of points in R^0 converging to x_B such that $g_i(x_B) = 0$ for at least one i, then $\lim_{k\to\infty} I(x) = +\infty$.

Definition 4.2. *$s(r)$ is a scalar valued function of the single variable r with the following properties:*

Property 1 If $r_1 > r_2 > 0$, then $s(r_1) > s(r_2) > 0$.
Property 2 If $\{r_k\}$ is an infinite sequence of points such that $\lim_{k\to\infty} r_k = 0$, then $\lim_{k\to\infty} s(r_k) = 0$.

Given the functions, $I(x)$ and $s(r)$ as in Definitions 4.1and 4.2, the interior unconstrained minimization function, as defined by Fiacco & McCormick (1968) is

$$U(x, r_k) = f(x) + s(r_k)I(x). \tag{4.2}$$

Starting from a point $x^0 \in R^0$, the unconstrained function $U(x, r_1)$ is solved to yield a local minimum $x(r_1) \in R^0$. Subsequently, the function $U(x, r_2)$ is solved to find its local minimum, with $x(r_1)$ as its initial point. Continuing in this fashion, a local minimum of $U(x, r_k)$, $x(r_k)$ is found starting from $x(r_{k-1})$. Under appropriate assumptions, Fiacco and McCormick prove that the sequence of local minima exists and converges to a local minimum of the original problem (4.1).

Theorem 4.1. *Assuming functions f, $g_1, \ldots, g_m$ are continuous and function U defined as in 4.2, where $I(x)$ and $s(r)$ satisfies the properties as defined in 4.1 and 4.2, then the problem (4.1) has at least one local minimum in the closure of R^0, and $\{r_k\}$ is a strictly decreasing null sequence. Moreover, there exists a sequence*

of points $\{x(r_k)\}$ *such that* $\lim_{k\to\infty} f[x(r_k)] = f(x^*)$*, where* x^* *is an isolated local minimizer of the problem (4.1).*

Proof. See Theorem 8 in Fiacco & McCormick (1968). □

4.2 Barrier Function

In the context of interior point methods, barrier functions are used to transform a constrained problem into an unconstrained problem or into a sequence of unconstrained problems. Given that the solution methods starts from the interior of the feasible region, these functions set a barrier against leaving the feasible region. Two types of barrier function are often used when interior point methods are utilized to solve an optimization problem. Let

$$I(x) = -\sum_{i=1}^{m} \ln(g_i(x)) \text{ and } s(\mu_k) = \mu_k. \tag{4.3}$$

Using (4.12), the constrained nonlinear programming problem (4.1) can be transformed into the following interior unconstrained minimization function.

$$U_L(x, \mu_k) = f(x) - \mu_k \sum_{i=1}^{m} \ln(g_i(x)). \tag{4.4}$$

The function U_L in (4.4) is referred to as the *logarithmic barrier function.* In order to illustrate the other type of barrier function,let

$$I(x) = \sum_{i=1}^{m} \frac{1}{g_i(x)} \text{ and } s(\mu_k) = \mu_k^2. \tag{4.5}$$

Using the above definitions of $I(x)$ and $s(\mu)$, the transformation of (4.1) is

$$U_I(x, \mu_k) = f(x) + {\mu_k}^2 \sum_{i=1}^{m} \frac{1}{g_i(x)}. \tag{4.6}$$

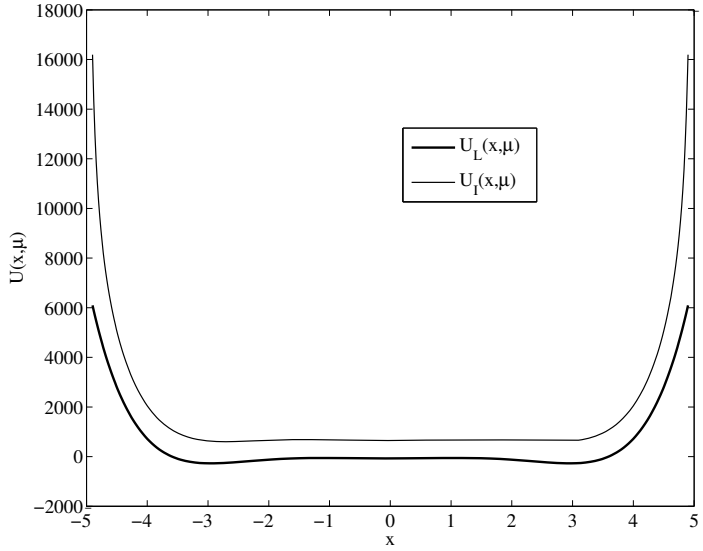


Figure 4.1: Interior point unconstrained functions

The function U_I in (4.6) is referred to as the *inverse barrier function*. Note that $I(x)$ and $s(\mu)$ in both logarithmic and inverse barrier functions satisfy the properties stated in Definitions 4.1 and 4.2.

For example, consider the following problem from Floudas *et al.* (1999)

$$\begin{aligned} &\text{minimize } x^6 - 15x^4 + 27x^2 + 250 \\ &\text{subject to } -5 \leq x \leq 5. \end{aligned} \tag{4.7}$$

The interior point unconstrained function utilizing either the logarithmic barrier function or inverse barrier function for the problem (4.7) can be obtained as

$$U_L(x, \mu_k) = x^6 - 15x^4 + 27x^2 + 250 - \mu_k(\ln(x+5) + \ln(5-x)), \tag{4.8}$$

$$U_I(x, \mu_k) = x^6 - 15x^4 + 27x^2 + 250 + \mu_k^2 \left(\frac{1}{x+5} + \frac{1}{5-x} \right). \tag{4.9}$$

Figure 4.1 shows a plot of the interior point unconstrained function shown in (4.8) and (4.9) for $\mu_k = 10$.

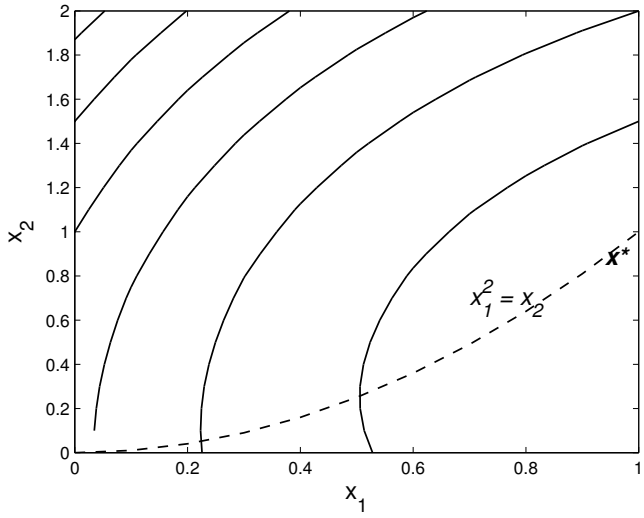


Figure 4.2: Contours of problem (4.10)

Thus by varying the barrier parameter μ_k, the interior point function in (4.8) or (4.9) provides a sequence of unconstrained minimization function such that when $\mu_k \to 0$, the sequence of solution obtained approaches the local minimizer of the original problem. The success of barrier function method also depends on the initialization of barrier parameter μ. The initial value of μ and its subsequently updated value can largely influence the quality of the solution obtained. Generally, initializing μ to a large value and then reducing it gradually results in obtaining a good quality solution.

In order to illustrate how the logarithmic barrier function converges to a solution, consider the following problem from Bazaraa *et al.* (1993)

$$\begin{aligned} &\text{minimize } (x_1 - 2)^4 + (x_1 - 2x_2)^2 \\ &\text{subject to } x_1^2 - x_2 \leq 0. \end{aligned} \tag{4.10}$$

Figure 4.2 shows the contours of the objective function and the boundary of the feasible region, as marked by the equality constraint $x_1^2 - x_2 = 0$. The solution

Table 4.1: Summary of computations for the barrier function method

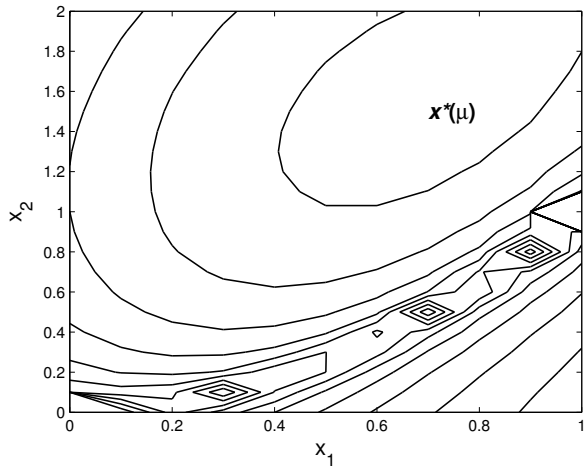
k	μ_k	$x_1(\mu_k)$	$x_2(\mu_k)$	$f(x)$	$U_L(x, \mu_k)$
1	10	0.7051	1.5452	8.5012	5.3990
2	1	0.8798	0.9980	2.8205	2.5720
3	0.1	0.8813	0.9132	2.4594	2.4366

to the problem (4.10) is known to be $x^* = (0.9456, 0.8941)$. The logarithmic barrier reformulation of the problem is obtained as shown below:

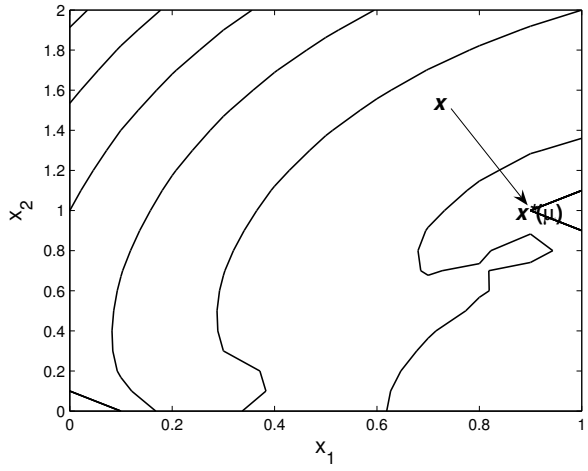
$$\text{minimize } U_L(x, \mu_k) = (x_1 - 2)^4 + (x_1 - 2x_2)^2 - \mu_k \ln(x_2 - x_1^2). \tag{4.11}$$

Thus the above unconstrained minimization problem, can be solved for a single local minimum for each value of μ_k. The values of $x_1(\mu_k)$ and $x_2(\mu_k)$ for various values of μ_k are given in the Table 4.1. Figure 4.2 shows the contour plot of problem (4.11) along with the local minima and the path traced by the barrier trajectory. The figure geometrically shows the values of points corresponding to the values of μ_k as provided in Table 4.1. As $\mu_k \to 0$, the sequence of minimizing points approaches the solution $(0.9456, 0.8941)$. From the table, as μ_k decreases, it can be observed that the objective function $(f(x))$ and the auxiliary function $(U_L(x, \mu_k))$ are nondecreasing functions of μ_k.

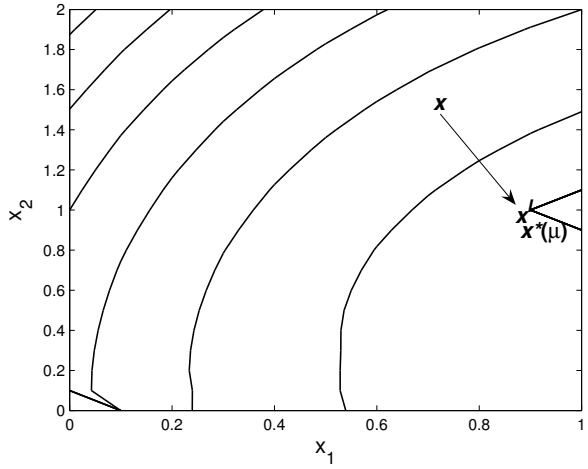
The barrier function method can be used to solve a constrained nonlinear programming problem only when the feasible region has a nonempty interior. Finding an initial point for some problems may be challenging and often heuristics have been used to overcome this difficulty. Moreover, due to the structure of the barrier function, for small values of the parameter μ_k, the search procedure may face difficulty due to ill-conditioning and round-off errors.This effect is more pronounced as the solution approaches the boundary of the feasible region.



(a) $\mu = 10$



(b) $\mu = 1$



(c) $\mu = 0.1$

Figure 4.3: Barrier trajectory path

4.3 Logarithmic Barrier Function

As discussed in Section 4.2, the barrier methods transform a constrained problem into an unconstrained problem or into a sequence of unconstrained problems. In order to achieve this, the inequality constraints of a problem are often integrated with its objective function by a barrier term. The barrier function, $\Omega(x)$, that we intend to use is defined to be

$$\Omega(x) = -\sum_{i=1}^{n} \frac{1}{(x_i - l_i)\ln(x_i - l_i) + (u_i - x_i)\ln(u_i - x_i)} \tag{4.12}$$

The barrier function above is well-defined for values of $l_i \leq x_i \leq u_i, i = 1, 2, \ldots, n$, and can be used to reformulate problem (4.23) into an unconstrained problem as shown below:

$$\text{Minimize } f(x) - \mu \sum_{i=1}^{n} \frac{1}{(x_i - l_i)\ln(x_i - l_i) + (u_i - x_i)\ln(u_i - x_i)}, \tag{4.13}$$

where $\mu > 0$ is a barrier parameter. For a specific value of μ, the unconstrained problem (4.13) can be solved using a variety techniques that exist today. The solution of the unconstrained problem, for a specific value of μ can be used as the initial point for solving the subsequent unconstrained functions with a reduced value of μ. This procedure is repeated until μ reaches zero, at which point the subproblem will resemble the original problem to be solved. The key benefits of this method are as follows:

- Elimination of inequality constraints totally.
- Reduction in objective function value and the non-violation of constraints are simultaneously achieved.

- Transforming the original problem into a sequence of unconstrained problems facilitate the use of a number of known methods for minimizing an unconstrained function.

- Irrespective of the search method, the transformed problem eliminates motion along the boundary completely. Moving along the boundary of the feasible region is a cumbersome process, more so if the surface is nonlinear.

The convexity of the barrier term, $\Omega(x)$ as shown in (4.12) is essential for the solution methodology and is one of the important properties of the barrier function. Given a convex barrier function, then for a large μ, the function $f(x)+\mu\Omega(x)$ will also be convex. Thus the barrier parameter, μ, acts as a smoothing parameter to render the nonconvexity of $f(x)$ ineffective by avoiding the possibility of multiple local minimum solutions.

4.4 Properties of Barrier Function

In this section, we describe the properties of barrier function, $\Omega(x)$ and that of the transformed objective function, (4.13). Firstly, the following lemmas are presented, which are later required to prove Theorem 4.2.

Lemma 4.1. *If the range of bounds on the variable x_i, $u_i - l_i \leq 1$, then the function, $q_i(x) = (x_i - l_i)\log(x_i - l_i) + (u_i - x_i)\log(u_i - x_i)$ is negative for all $x_i \in X^0$, where $X^0 := \{x_i \mid l_i < x_i < u_i, i = 1, 2, \ldots, n\}$.*

Proof. Suppose $x \in X^0$ be any feasible point, then

$$0 < u - x < 1.$$

Taking log on both sides of the above inequality,

$$\log(u - x) < 0. \tag{4.14}$$

Similarly,

$$0 < x - l < 1.$$

Taking log on both sides of the above inequality,

$$\log(x - l) < 0. \tag{4.15}$$

Dividing (4.14) by (4.15) gives,

$$\frac{\log(u - x)}{\log(x - l)} > 0. \tag{4.16}$$

Also, note that

$$\frac{x - l}{u - x} > 0 \text{ or } -\frac{x - l}{u - x} < 0. \tag{4.17}$$

From (4.16) and (4.17)it follows that

$$\frac{\log(u - x)}{\log(x - l)} > -\frac{x - l}{u - x}.$$

Since $\log(x - l) < 0$,

$$(u - x)\log(u - x) < -(x - l)\log(x - l).$$

Therefore,

$$(x - l)\log(x - l) + (u - x)\log(u - x) < 0.$$

□

Lemma 4.2. *If the range of bounds on the variable x_i, $u_i - l_i \geq 2$, then the function, $q_i(x) = (x - l)\log(x - l) + (u - x)\log(u - x)$ is positive for all $x_i \in X^0$, where $X^0 := \{x_i \mid l_i < x_i < u_i, i = 1, 2, \ldots, n\}$.*

Proof. Let $x_i \in X^0$ be any feasible point. Let $u_i - l_i = \delta_i \geq 2$.

Taking the limits on each of the terms in $q_i(x)$, as $x_i \to u_i^-$, we have

$$\lim_{x_i \to u_i^-} (u_i - x_i) \log(u_i - x_i) = 0, \tag{4.18}$$

$$\lim_{x_i \to u_i^-} (x_i - l_i) \log(x_i - l_i) = \delta_i \log \delta_i > 0, \ (\because \delta_i \geq 2). \tag{4.19}$$

Adding (4.18) and (4.19), we have

$$\lim_{x_i \to u_i^-} (u_i - x_i) \log(u_i - x_i) + (x_i - l_i) \log(x_i - l_i) > 0.$$

Similarly, it can be proved that,

$$\lim_{x_i \to l_i^+} (u_i - x_i) \log(u_i - x_i) + (x_i - l_i) \log(x_i - l_i) > 0.$$

□

Lemma 4.3. *If the range of bounds on the variable x_i, $1 < u_i - l_i < 2$, then the function, $q_i(x) = (x_i - l_i) \log(x_i - l_i) + (u_i - x_i) \log(u_i - x_i)$ is either positive or negative depending on the position of $x_i \in X^0$, where $X^0 := \{x_i \mid l_i < x_i < u_i, i = 1, 2, \ldots, n\}$.*

Proof. Let $u_i - l_i = \delta_i$. Taking the limits on $q_i(x)$, we have

$$\lim_{x_i \to l_i^+} (x_i - l_i) \log(x_i - l_i) + (u_i - x_i) \log(u_i - x_i) = \delta_i \log \delta_i \Rightarrow q_i(x) > 0.$$

$$\lim_{x_i \to \frac{l_i + u_i}{2}} (x_i - l_i) \log(x_i - l_i) + (u_i - x_i) \log(u_i - x_i) = \delta_i \log \left(\frac{\delta_i}{2} \right) \Rightarrow q_i(x) < 0.$$

$$\lim_{x_i \to u_i^-} (x_i - l_i) \log(x_i - l_i) + (u_i - x_i) \log(u_i - x_i) = \delta_i \log \delta_i \Rightarrow q_i(x) > 0.$$

As x_i varies from l_i to u_i, the sign of $(q_i(x)$ varies from positive to negative to positive, when $1 < u_i - l_i < 2$. □

Theorem 4.2. *Suppose $\Omega : X \to \Re$ is a C^2 function, where $X \subset [l,u]^n$. Then for all $x \in X \setminus D$, $\Omega(x)$ is a strictly convex function, where $D := \{x \mid x \in X, 1 < u_{x_i} - l_{x_i} < 2, i = 1,2,...,n\}$, u_{x_i} and l_{x_i} are the upper and lower bounds on x_i, respectively.*

Proof. From the expression of $\Omega(x)$ as defined in (4.12) and its derivatives given in (4.21) and (4.22), the Hessian matrix of $\Omega(x)$ at any $x \in X \setminus D$ is given by

$$\nabla^2_{xx}\Omega(x) = Diag\left(\frac{\left(\frac{1}{u_i-x_i}+\frac{1}{x_i-l_i}\right)}{q_i(x)^2} - \frac{2\ln\left(\frac{x_i-l_i}{u_i-x_i}\right)}{q_i(x)^3}\right), i = 1,...,n,$$

where $q_i(x) = (x_i - l)\ln(x_i - l_i) + (u_i - x_i)\ln(u_i - x_i)$ and Diag(x) denotes a diagonal matrix with the components of x as its diagonal elements. Let

$$t_1 = \frac{\left(\frac{1}{u_i-x_i}+\frac{1}{x_i-l_i}\right)}{q_i(x)^2} \text{ and } t_2 = \frac{2\ln\left(\frac{x_i-l_i}{u_i-x_i}\right)}{q_i(x)^3}.$$

In order for the diagonal elements of the Hessian matrix to be nonnegative, t_1 should be greater than or equal to t_2. Consider the following three cases:

Case 1: $u_{x_i} - l_{x_i} \leq 1$

From Lemma 4.1, it follows that $q_i(x) < 0$ for $u_{x_i} - l_{x_i} \leq 1$.

Suppose that $t_2 > t_1$. Then,

$$\frac{2\ln\left(\frac{x_i-l_i}{u_i-x_i}\right)}{q_i(x)^3} > \frac{\left(\frac{1}{u_i-x_i}+\frac{1}{x_i-l_i}\right)}{q_i(x)^2}.$$

Since $q_i(x) < 0$, rearranging the terms in above inequality, we have

$$\ln\left(\frac{x_i-l_i}{u_i-x_i}\right) < \frac{q_i(x)^3\left(\frac{1}{u_i-x_i}+\frac{1}{x_i-l_i}\right)}{2q_i(x)^2}.$$

Since the RHS of the above inequality is negative,

$$0 < \frac{x_i - l_i}{u_i - x_i} < 1 \Rightarrow l_i < x_i < \frac{u_i + l_i}{2},$$

which contradicts that $x \in [l, u]^n$. Hence, $t_1 > t_2$ and the Hessian of $\Omega(x)$ is positive definite.

Case 2: $u_{x_i} - l_{x_i} \geq 2$

From Lemma 4.2, it follows that $q_i(x) > 0$ for $u_{x_i} - l_{x_i} \geq 2$.

Suppose that $t_2 > t_1$. Then,

$$\frac{2 \ln \left(\frac{x_i - l_i}{u_i - x_i} \right)}{q_i(x)^3} > \frac{\left(\frac{1}{u_i - x_i} + \frac{1}{x_i - l_i} \right)}{q_i(x)^2}$$

Since $q_i(x) > 0$, rearranging the terms in above inequality, we have

$$\ln \left(\frac{x_i - l_i}{u_i - x_i} \right) > \frac{q_i(x)^3 \left(\frac{1}{u_i - x_i} + \frac{1}{x_i - l_i} \right)}{2 q_i(x)^2}$$

Since the RHS of the above inequality is positive,

$$\frac{x_i - l_i}{u_i - x_i} > 1 \Rightarrow x_i > \frac{u_i + l_i}{2},$$

which contradicts that $x \in [l, u]^n$. Hence, $t_1 > t_2$ and the Hessian matrix of $\Omega(x)$ is positive definite.

Case 3: $1 < u_{x_i} - l_{x_i} < 2$

From Lemma 4.3, we see that the sign of $q_i(x)$ varies and hence the sign of diagonal elements of the Hessian matrix could either be positive or negative depending on the range of bounds on the variable x_i.

Since the Hessian of $\Omega(x)$ is positive definite for all $x \in X \setminus D$, $\Omega(x)$ is strictly convex on $X \setminus D$. □

The figure 4.4 illustrates the behavior of the barrier function for different values of the range of bounds, as detailed in three cases above. Both the figure 4.4 and Theorem 4.2, show that the convexity of the barrier function and that of the transformed function highly depends on the range of bounds of the variable involved. In the interest of Lemma 4.4 to be proved later, we present below the

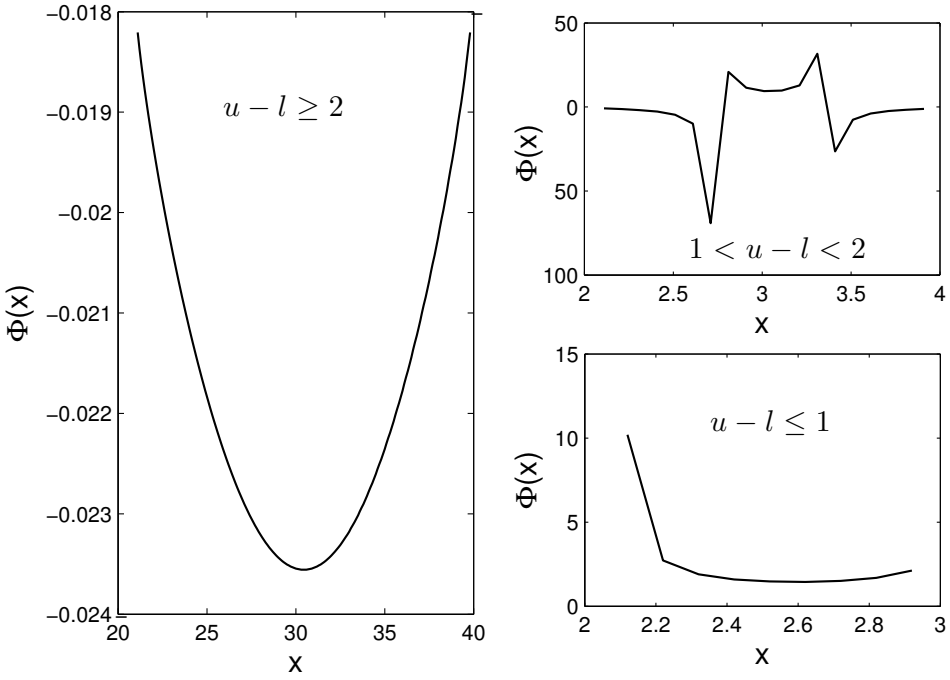


Figure 4.4: Effect of range of bounds on barrier function, $\Omega(x)$

transformed problem and its derivatives: The transformed problem is

$$F(x, \mu) = f(x) + \mu\Omega(x), \tag{4.20}$$

where $\mu > 0$ is the barrier parameter.

The first derivative of $F(x,\mu)$ is

$$\frac{\partial F(x,\mu)}{\partial x_i} = \frac{\partial f(x)}{\partial x_i} + \mu \frac{\ln\left(\frac{x_i - l_i}{u_i - x_i}\right)}{q_i(x)^2}, \tag{4.21}$$

where $q_i(x) = (x_i - l_i)\ln(x_i - l_i) + (u_i - x_i)\ln(u_i - x_i)$, $i = 1, \ldots, n$. The second derivative of $F(x,\mu)$ is given by

$$\frac{\partial^2 F(x,\mu)}{\partial {x_i}^2} = \frac{\partial^2 f(x)}{\partial {x_i}^2} + \mu \left\{ \frac{\left(\frac{1}{u_i - x_i} + \frac{1}{x_i - l_i}\right)}{q_i(x)^2} - \frac{2\ln\left(\frac{x_i - l_i}{u_i - x_i}\right)}{q_i(x)^3} \right\}. \tag{4.22}$$

Lemma 4.4. *If $f : X \setminus D \to \Re$ is a C^2 function and Ω is as defined in (4.12), then there exists a real $M > 0$ such that if $\mu \geq M$, then $f + \mu\Omega$ is a strictly convex function on $(l,u)^n$.*

Proof. Let $x \in X \setminus D$. Then, the Hessian of $\Omega(x)$ is a diagonal matrix with the i^{th} diagonal entry as

$$\frac{\left(\frac{1}{u_i - x_i} + \frac{1}{x_i - l_i}\right)}{q_i(x)^2} - \frac{2\ln\left(\frac{x_i - l_i}{u_i - x_i}\right)}{q_i(x)^3}.$$

The above function has a minimum at

$$x_i = \frac{u_i + l_i}{2},$$

which implies that every diagonal entry of $\nabla^2\Omega(x)$ is at least

$$\frac{4}{((u_i - l_i)\ln(\frac{u_i - l_i}{2}))^2}.$$

Thus the minimum eigenvalue of the Hessian of $\Omega(x)$,

$$\lambda_{\min}(\nabla^2\Omega(x)) \geq \frac{4}{((u_i - l_i)\ln(\frac{u_i - l_i}{2}))^2}$$

and hence from Theorem 4.3, we conclude that $f + \mu\Omega$ is a strictly convex function on $X \setminus D$. The result of this lemma follows from Theorem 4.3. □

To close this section, Theorem 4.3 is presented below. Since its proof can be found in Murray & Ng (2008), it is omitted here.

Theorem 4.3. *(**Murray & Ng (2008)**) Suppose that $f : [l, u]^n \to \mathbb{R}$ is a C^2 function and $\Omega : X \to \Re$ is a C^2 function such that the minimum eigenvalue of its Hessian matrix $\nabla^2\Omega(x)$ is greater than $\xi(> 0)$ for all $x \in X$, where $X \subset [l, u]^n$. Then there exists a constant $M > 0$ such that, when $\mu > M$, $f + \mu\Omega$ is a strictly convex function on X.*

Since the transformed problem $f + \mu\Omega$ is convex, for a sufficiently large value of μ, there exists a unique solution $x^*(\mu)$ for problem (4.13). Based on Theorem 8 in Fiacco & McCormick (1968), if $x^*(\mu)$ is a solution of problem (4.13), then there exists a sequence of points $\{x(\mu)\}$, such that $\lim_{\mu\to 0} x^*(\mu) = x^*$, where x^* is the solution to the original problem (4.23).

Thus the original nonconvex problem with box constraints, (4.23) has been converted to a smooth unconstrained nonlinear program. For a sufficiently large value of μ, each and every unconstrained problem will have a unique (global) minimizer, $x^*(\mu)$. By using an appropriate method to solve the transformed problem, we hope to obtain a global or at least a good local minimum of the original problem by solving a sequence of unconstrained problems.

4.5 Barrier Function Algorithm

The solution methods that we propose to solve the energy minimization problem belongs to a class of interior point methods, which are often employed to solve linear and nonlinear optimization problems. A variety of solution techniques for solving the nonconvex energy function have been proposed and were discussed

above. Specialized algorithms with nice convergence properties for a particular class of problems (Klepeis *et al.*, 1997) or application oriented heuristics which gives approximate solutions have always been developed. A book series that runs for more than 80 volumes have been published by Springer on the title "Nonconvex Optimization and its Applications". Pardalos *et al.* (1994) discusses different optimization methods that are used in the minimization of nonconvex potential energy functions.

Here, we will discuss our proposed solution approach to solve nonlinear non-convex optimization problems with bound constraints as shown below in problem (4.23).

$$\begin{aligned} &\text{Minimize } f(x) \\ &\text{subject to } l_i \leq x_i \leq u_i, \quad i = 1, \ldots, n, \end{aligned} \tag{4.23}$$

where $f(x)$ is a twice-continuously differentiable function, $x \in \Re^n, l_i$ and u_i are the lower and upper bounds on the variable x_i, respectively. It is also assumed that l_i and u_i, $i = 1, 2, \ldots, n$, are finite, which results in a bounded feasible region. The reason for our interest in such problems is its relevance to the optimization problems in the area of computational biology. Specifically, these type of problem structures are very common in the area of minimum energy determination of molecules. Hence, the solution methodologies that we propose is built around solving problems of type (4.23) for a sequence of decreasing μ. As Doyle (2003) observes, the difference between different barrier function methods lies in their choice of algorithms to solve the problem, how μ is adjusted, and the choice of termination conditions. Based on a particular descent direction, similar to the one in Dang & Xu (2000), the search method that we propose finds a solution to the problem (4.23). We derive the direction of search based on the first-order

necessary conditions and later, prove that it is a descent direction of the function $F(x, \mu)$. The following section illustrates how the search direction is obtained and proves it to be the descent direction of the function $F(x, \mu)$.

4.5.1 Determining the Descent Direction

For any positive μ and $x \in X \setminus D$, the first-order necessary optimality conditions for problem 4.20 is

$$\frac{\partial F(x,\mu)}{\partial x_i} = 0, \qquad i = 1, 2, \ldots, n.$$

Then from (4.20), it implies that

$$\frac{\partial f(x)}{\partial x_i} + \mu \frac{\ln\left(\frac{x_i - l_i}{u_i - x_i}\right)}{q_i(x)^2} = 0, \qquad i = 1, 2, \ldots, n, \tag{4.24}$$

where $q_i(x) = (x_i - l_i)\ln(x_i - l_i) + (u_i - x_i)\ln(u_i - x_i)$. From (4.24), we obtain

$$x_i = \frac{u_i + l_i \exp\left(\frac{q_i(x)^2}{\mu}\frac{\partial f(x)}{\partial x_i}\right)}{1 + \exp\left(\frac{q_i(x)^2}{\mu}\frac{\partial f(x)}{\partial x_i}\right)}, \qquad i = 1, 2, \ldots n. \tag{4.25}$$

Let

$$\eta_i(x) = \exp\left(\frac{q_i(x)^2}{\mu}\frac{\partial f(x)}{\partial x_i}\right), \qquad i = 1, 2, \ldots, n,$$

and rearranging (4.25), we let

$$\gamma_i(x) = \frac{u_i + l_i\eta_i(x)}{1 + \eta_i(x)} - x_i, \qquad i = 1, 2, \ldots, n.$$

Thus, for any x in the interior of the feasible region of problem (4.20) and for any $\mu > 0$, the following lemma shows that $\gamma_i(x)$ is a descent direction of $F(x, \mu)$.

Lemma 4.5. *For any $\mu > 0$, and $x \in X \setminus D$, $\gamma_i(x)$ is a descent direction of $F(x, \mu)$ when $\gamma_i(x) \neq 0$.*

Proof. In order to prove $\gamma_i(x)$ to be the descent direction of $F(x,\mu)$, it would suffice to prove that $\nabla_x F(x,\mu)^\top \gamma_i(x) < 0$.

Case 1: When $\gamma_i(x) > 0$, we have

$$\frac{u_i + l_i\eta_i(x)}{1+\eta_i(x)} - x_i > 0. \tag{4.26}$$

Rearranging the terms in (4.26), we get

$$\eta_i(x)\frac{x_i - l_i}{u_i - x_i} < 1.$$

Substituting the value of $\eta_i(x)$,

$$\frac{x_i - l_i}{u_i - x_i}\exp\left(\frac{q_i(x)^2}{\mu}\frac{\partial f(x)}{\partial x_i}\right) < 1.$$

Taking the logarithm on both sides of the above inequality,

$$\log\left(\frac{x_i - l_i}{u_i - x_i}\right) + \frac{q_i(x)^2}{\mu}\frac{\partial f(x)}{\partial x_i} < 0. \tag{4.27}$$

Multiplying $\frac{\mu}{q_i(x)^2} > 0$ on both sides of (4.27), we get

$$\frac{\partial F(x,\mu)}{\partial x_i} = \frac{\partial f(x)}{\partial x_i} + \frac{\mu}{q_i(x)^2}\ln\left(\frac{x_i - l_i}{u_i - x_i}\right) < 0.$$

Thus, when $\gamma_i(x) > 0$, $\dfrac{\partial F(x,\mu)}{\partial x_i} < 0$.

Case 2: When $\gamma_i(x) < 0$, we have

$$\frac{u_i + l_i\eta_i(x)}{1+\eta_i(x)} - x_i < 0. \tag{4.28}$$

Rearranging the terms in (4.28), we get

$$\eta_i(x)\frac{x_i - l_i}{u_i - x_i} > 1.$$

Substituting the value of $\eta_i(x)$,

$$\frac{x_i - l_i}{u_i - x_i} \exp\left(\frac{q_i(x)^2}{\mu}\frac{\partial f(x)}{\partial x_i}\right) > 1.$$

Taking the logarithm on both sides of the above inequality,

$$\log\left(\frac{x_i - l_i}{u_i - x_i}\right) + \frac{q_i(x)^2}{\mu}\frac{\partial f(x)}{\partial x_i} > 0. \tag{4.29}$$

Multiplying $\frac{\mu}{q_i(x)^2} > 0$ on both sides of (4.29), we get

$$\frac{\partial F(x,\mu)}{\partial x_i} = \frac{\partial f(x)}{\partial x_i} + \frac{\mu}{q_i(x)^2}\ln\left(\frac{x_i - l_i}{u_i - x_i}\right) > 0.$$

Thus, when $\gamma_i(x) < 0$, $\dfrac{\partial F(x,\mu)}{\partial x_i} > 0$.

Case 3: When $\gamma_i(x) = 0$, we have

$$\frac{u_i + l_i\eta_i(x)}{1 + \eta_i(x)} - x_i = 0. \tag{4.30}$$

Rearranging the terms in (4.30), we get

$$\eta_i(x)\frac{x_i - l_i}{u_i - x_i} = 1.$$

Substituting the value of $\eta_i(x)$,

$$\frac{x_i - l_i}{u_i - x_i} \exp\left(\frac{q_i(x)^2}{\mu}\frac{\partial f(x)}{\partial x_i}\right) = 1.$$

Taking the logarithm on both sides of the above equation,

$$\log\left(\frac{x_i - l_i}{u_i - x_i}\right) + \frac{q_i(x)^2}{\mu}\frac{\partial f(x)}{\partial x_i} = 0. \tag{4.31}$$

Multiplying $\frac{\mu}{q_i(x)^2} > 0$ on both sides of (4.31), we get

$$\frac{\partial F(x,\mu)}{\partial x_i} = \frac{\partial f(x)}{\partial x_i} + \frac{\mu}{q_i(x)^2}\ln\left(\frac{x_i - l_i}{u_i - x_i}\right) = 0.$$

Thus, when $\dfrac{\partial F(x,\mu)}{\partial x_i} = 0$, $\gamma_i(x) = 0,$.

Hence, we conclude that $\gamma_i(x)$ is the descent direction of $F(x,\mu)$ and $\gamma_i(x) = 0$ if and only if $\nabla_x F(x,\mu) = 0$. □

4.5.2 Proposed Algorithm

Based on the descent direction obtained above, we develop an interior point based algorithm, which could find a solution for problems of type (4.23). The framework of the proposed Barrier Function Algorithm (BFA) is shown in Algorithm 1. The iterative scheme that we propose is based on the barrier parameter μ, which is reduced in every iteration of the algorithm. The barrier function, $\Omega(x)$, added to the objective function, $f(x)$, ensures that the minimum of the function is achieved in the interior of the feasible region.

From Section 4.5.1, we know that $\gamma(x)$ is the direction of descent of $F(x)$, where $F(x) = f(x)+\mu\Omega(x)$. Once the direction of search is found, it is imperative to find the steplength, α for determining the next iterate $x + \alpha\gamma(x)$. While there are plenty of line search methods available, we use the Golden Section Search (GSS) method, the framework of which is provided in Algorithm 2. The reasons for using the GSS are three-fold,

- It does not use any derivative information
- It is computationally inexpensive
- It is efficient and easy to implement

The GSS works well with the BFA, and since we are interested only in the performance of BFA, we have not proposed any enhancements to the GSS method. The GSS method is implemented as it is described in Bazaraa *et al.* (1993). The interval of uncertainty for the steplength is taken to be [0,1]. As we are dealing with interior point methods, care must be taken to ensure that the subsequent iterates also lie in the interior of the feasible region.

Algorithm 1 Barrier Function Algorithm

Set μ_0 = initial barrier parameter,
ϵ_D = tolerance for the magnitude of direction,
ϵ_μ = tolerance for barrier parameter,
θ_μ = reduction factor,
n = total number of variables,
K = maximum number of iterations,
r = any feasible starting point.
Set μ = μ_0.

while $\mu > \epsilon_\mu$
  Set $x_0 = r$.
    for $k = 0, 1, \ldots, K$
      Compute $\gamma_i(x^k), \forall i = 1, 2, \ldots, n$.
      if $\| \gamma(x^k) \| < \epsilon_D$
        Set $x^K = x^k$, $k = K$.
      else
        Compute λ such that it is optimal to
          $\min_{\lambda \in [0,1]} F\left(x^k + \lambda\gamma(x^k), \mu\right)$.
        Set $x^{k+1} = x^k + \lambda_k \gamma_k(x)$.
      end if
    end for
  Set $\mu = \theta_\mu \mu$,
    $r = x_K$.
end while

Algorithm 2 Golden Section Search for determining steplength

Let $[a_k, b_k]$ = interval of uncertainty
$F(\cdot)$ = function to be minimized
l = allowable length of uncertainty
γ = reduction factor
λ = steplength
k = iteration counter
Set $[a_1, b_1] = [0, 1]$
$\gamma = 0.618$
$\alpha_1 = a_1 + (1 - \gamma)(b_1 - a_1)$
$\beta_1 = a_1 + \gamma(b_1 - a_1)$
$k = 1$
$flag = 0$
Compute $F(\alpha_1)$ and $F(\beta_1)$

while $flag = 0$
  if $b_k - a_k > l$
    if $F(\alpha_k) > F(\beta_k)$
      $a_{k+1} = \alpha_k$
      $b_{k+1} = b_k$
      $\alpha_{k+1} = \beta_k$
      $\beta_{k+1} = a_{k+1} + \gamma(b_{k+1} - a_{k+1})$
      Compute $F(\beta_{k+1})$
      $k = k + 1$
    else
      $a_{k+1} = a_k$
      $b_{k+1} = \beta_k$
      $\beta_{k+1} = \alpha_k$
      $\alpha_{k+1} = a_{k+1} + (1 - \gamma)(b_{k+1} - a_{k+1})$
      Compute $F(\alpha_{k+1})$
      $k = k + 1$
    end if
  else
    $\alpha = \frac{a(k)+b(k)}{2}$
    $flag = 1$
  end if
end while

In barrier function methods, it is imperative to choose an interior feasible point as the initial iterate. This is why a nonempty feasible region forms a important part of the requirements of a barrier function. The initial iterate for the problem, x_0 belonging to the interior of the feasible region X, is generally preferred to be away from the boundary of the feasible region. To begin the search process starting from a point close to the boundary will render the search method inefficient. However, for a large value of initial barrier parameter, there are no inherent risks in picking any point in the interior of the feasible region. Thus an unbiased initial iterate, compatible with the barrier parameter and located in the interior of the feasible region is highly important and is commonly referred to as the "neutral point" in the literature. One such point is the analytic center of the feasible region, which is often used as the starting point for the interior point algorithms. For more about analytic center, the reader is referred to Ye (1997).

Apart from the initial starting point, it is also important to carefully choose the parameters associated with the proposed algorithm. As discussed in Lemma 4.4, a large value of barrier parameter is required to maintain the convexity of the objective function. Thus a large initial barrier parameter value is important for a trajectory of iterates converging to either a global minimum or a good local minimum. Similarly, care should be taken while choosing the value for updating the reduction parameter after every iteration. A large value of reduction parameter could cause the path of iterates to change from one trajectory to another. Hence it is always better to initialize the parameters conservatively. Though this might translate to an increased computational time, the chances of obtaining a good quality solution are very high. Based on computational experience, the range of parameters used in the BFA are shown in Table 4.2.

Table 4.2: Range of parameters used

Parameter	Range
Initial barrier parameter, μ_0	100 to 1000
Reduction factor, θ_μ	0.85 to 0.99
Tolerance for μ, ϵ_μ	0.01 to 0.0001
Tolerance for direction, ϵ_D	0.05 to 0.1

4.6 Computational Experience

In order to evaluate the proposed algorithm, we use some of the standard test problems from the literature. Floudas *et al.* (1999) provides a collection of test problems and their global optimal solutions, obtained from various sources. These test problems are widely used as the benchmark test problems in the area of global optimization and we utilize the same problems to test our proposed algorithm. The list of test problems that we use are listed below:

Test Problem 1

The following problem is a minimization of a 50^{th} degree polynomial of single variable.

$$\text{Minimize} \quad \sum_{i=1}^{50} a_i x^i$$
$$\text{subject to } 1 \leq x \leq 2,$$

where $a = (-500, 2.5, 1.666666666, 1.25, 1, 0.8333333, 0.714285714, 0.625,$
$0.555555555, 1, -43.6363636, 0.41666666, 0.384615384, 0.357142857,$
$0.3333333, 0.3125, 0.294117647, 0.277777777, 0.263157894, 0.25,$
$0.238095238, 0.227272727, 0.217391304, 0.208333333, 0.2, 0.192307692,$
$0.185185185, 0.178571428, 0.344827586, 0.6666666, -15.48387097, 0.15625,$
$0.1515151, 0.14705882, 0.14285712, 0.138888888, 0.135135135, 0.131578947,$
$0.128205128, 0.125, 0.121951219, 0.119047619, 0.116279069, 0.113636363,$
$0.1111111, 0.108695652, 0.106382978, 0.208333333, 0.408163265, 0.8).$

Test Problem 2

Minimize $0.000089248x - 0.0218343x^2 + 0.998266x^3 - 1.6995x^4 + 0.2x^5$

subject to $0 \leq x \leq 10$.

Test Problem 3

Minimize $4x^2 - 4x^3 + x^4$

subject to $-5 \leq x \leq 5$.

Test Problem 4

Minimize $x^6 - 15x^4 + 27x^2 + 250$

subject to $-5 \leq x \leq 5$.

Test Problem 5

Minimize $x^4 - 3x^3 - 1.5x^2 + 10x$

subject to $-5 \leq x \leq 5$.

Test Problem 6

$$\text{Minimize} \quad x^6 - \frac{52}{25}x^5 + \frac{39}{80}x^4 + \frac{71}{10}x^3 - \frac{79}{20}x^2 - x + \frac{1}{10}$$
$$\text{subject to} \quad -2 \leq x \leq 11.$$

Test Problem 7

$$\text{Minimize} \quad \cos x_1 \sin x_2 - \frac{x_1}{x_2^2 + 1}$$
$$\text{subject to} \quad -1 \leq x_1 \leq 2$$
$$-1 \leq x_2 \leq 1.$$

Test Problem 8

The following problem is known in the literature as the Goldstein and Price function.

$$\text{Minimize} \quad \left[1 + (x_1 + x_2 + 1)^2(19 - 14x_1 + 3x_1^2 - 14x_2 + 6x_1x_2 + 3x_2^2)\right] \times$$
$$\left[30 + (2x_1 - 3x_2)^2(18 - 32x_1 + 12x_1^2 + 48x_2 - 36x_1x_2 + 27x_2^2)\right]$$
$$\text{subject to} \quad -2 \leq x_1 \leq 2$$
$$-2 \leq x_2 \leq 2.$$

Test Problem 9

The following problem is popularly known in the literature as the three-hump camel-back function.

$$\text{Minimize} \quad 2x_1^2 - 1.05x_1^4 + \frac{1}{6}x_1^6 - x_1x_2 + x_2^2$$
$$\text{subject to} \quad -5 \leq x_1, x_2 \leq 5.$$

Test Problem 10

The following problem is popularly known in the literature as the six-hump camel-back function.

$$\begin{aligned} \text{Minimize} \quad & 4x_1^2 - 2.1x_1^4 + \frac{1}{3}x_1^6 + x_1x_2 - 4x_2^2 + 4x_2^4 \\ \text{subject to} \quad & -3 \leq x_1 \leq 3 \\ & -2 \leq x_2 \leq 2. \end{aligned}$$

The ten above-mentioned problems were solved using our proposed algorithm and the results are shown in Table 4.3. The Source column in the table cites the paper from which that particular test problem was taken. Under the Reported column, the table also shows the global optimal objective value and the corresponding variable values at optimality. The column Found displays the values found by our method. The last two columns show the time taken and the number of iterations involved. All the computations were carried out on a PC with Intel Core 2 Duo processor running at 1.83 GHz and 1 GB of memory. The algorithms were implemented in MATLAB Version 7.2.

The initial value of barrier parameter (μ) in our Algorithm 1 is set to 100 and is reduced by a factor of $0.95(\theta_\mu)$ when $\epsilon_D \leq 0.01$. The method terminates when $\epsilon_\mu < 0.01$. The solution found by the proposed method almost always matches with that of the reported solution except for Problem No. 6. Results reported for Problem No.6 shows that the objective function value of -29763.2330 is achieved when $x = 10$. A mere substitution of the value, $x = 10$ into the corresponding objective function does not yield the reported value. Under the Found column for the corresponding problem, we report the results that we have obtained for that problem. For Problem No. 4, irrespective of the starting point, the algorithm always found the local optimum solution of 250 when $x = 0$. In order to get out

Table 4.3: Computational results for test problems

Prob No.	Source	Optimal Objective Value		Variable Values		Time (sec)	Iterations
		Reported	Found	Reported	Found		
1	Moore (1979)	-663.5	-663.5001	1.0911	1.0912	7	199
2	Wilkinson (1963)	-443.67	-442.8717	6.3250	6.3231	38	1232
3	Dixon & Szegö (1975)	0	0	0 or 2	0	8	2
4	Goldstein & Price (1971)	7	7	3 or -3	3	182	3938
5	Dixon (1990)	-7.5	-7.5	-1.0000	-1.0000	44	1013
6	Wingo (1985)	-29763.2330	-7.4873	10	0.4869	264	5385
7	Adjiman *et al.* (1998)	-2.0218	-1.9970	(2, 0.10578)	(1.9970, 0)	10	143
8	Goldstein & Price (1971)	3	3.0010	(0,-1)	(0.0018,-0.9987)	147	1606
9	Dixon & Szegö (1975)	0	0.0276	(0,0)	(-0.0962,-0.1555)	59	1525
10	Dixon & Szegö (1975)	-1.0316	-1.0316	(0.0898,-0.7126)	(0.0899, -0.7122)	26	694

of the local minima, we set the initial value of barrier parameter to 1000 and θ_μ to 0.99 and ran the algorithm again to find the reported global optimal solution of 7 at $x = 3$. An alternate solution is also known to exist for the problem at $x = -3$.

The test problems used above are very effective in determining the efficiency of the search method when polynomials of higher degree are encountered. It does not test the capacity of the method when the number of variables involved are larger. Hence, we use the following problem from Pardalos (1991) to determine the effectiveness of the proposed algorithm for larger problems.

Test Problem 12

$$\begin{aligned} \text{Minimize} \quad & -(n-1)\sum_{i=1}^{n} x_i - \frac{1}{n}\sum_{i=1}^{n/2} x_i + 2\sum_{i<j} x_i x_j \\ \text{subject to} \quad & x_i \in \{0,1\},\ i = 1, 2, \cdots, n, \end{aligned} \tag{4.32}$$

where n is an even positive integer.

This problem has an exponential number of discrete local minima. For a problem of size n, the unique global minimum point of (4.32) is $x^* = (1, \cdots, 1, 0, \cdots, 0)$, which has $n/2$ ones followed by $n/2$ zeros, with an optimal objective value of $-(n^2+2)/4$. We have used our proposed Algorithm 1 to solve the relaxed version of (4.32) up to 500 variables. For all the problems tested here, the analytic centre of the feasible region, $\frac{1}{2}e$ is taken to be the initial iterate for the algorithm. The other parameters are set at their default values as before and the results obtained are shown in Table 4.4. The objective value, Z^* shown in Table 4.4 gives the global optimum objective function value, which can be verified analytically. The values given under the column Z are the ones found by our Algorithm. It may be observed from the table that $Z \neq Z^*$ and this is due to the fact that the value Z

Table 4.4: Numerical results for problem (4.32)

Variables	Time (min)	Obj Value (Z)	Obj Value (Z^*)	$Z - Z^*$	Iterations
50	0.45	-623.31	-625.5	2.19	233
100	0.94	-2496.13	-2500.5	4.37	251
150	2.14	-5618.95	-5625.5	6.55	255
200	4.62	-9991.77	-10000.5	8.73	258
250	9.18	-15614.59	-15625.5	10.91	430
300	26.63	-22486.74	-22500.5	13.76	536
350	52.35	-30608.67	-30625.5	16.83	743
400	75.76	-39982.50	-40000.5	18.00	760
450	106.27	-50594.37	-50625.5	31.13	690
500	137.47	-62478.52	-62500.5	21.98	700

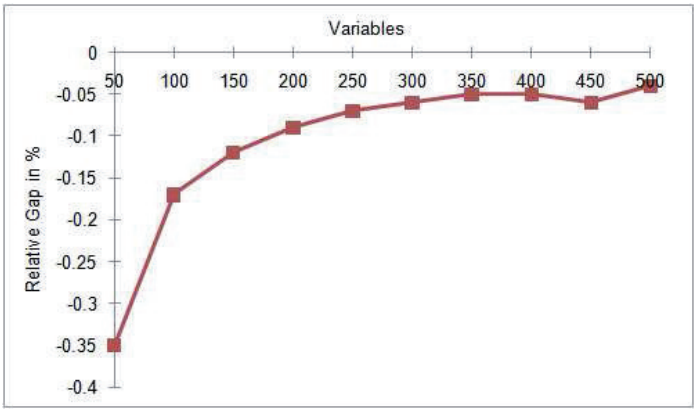


Figure 4.5: Effect of variables on % Gap

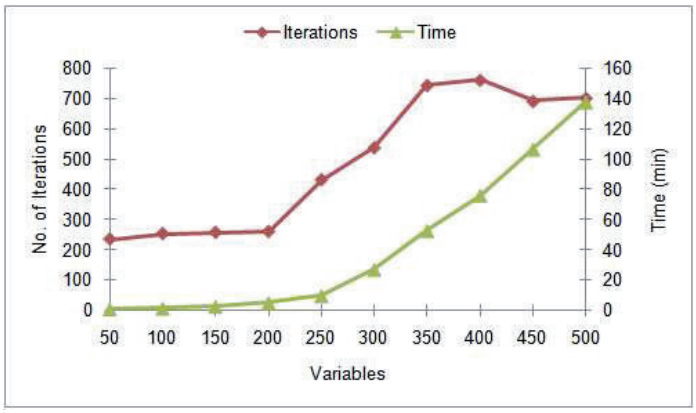


Figure 4.6: No. of iterations and time taken by BFA

is calculated at the non-integral values of the variables (before rounding). If the variables are rounded to its nearest integer values, it has been verified that the objective value found by our method is globally optimal. The effectiveness of an algorithm can be gauged by its ability to produce results as close as possible to the global optimum value. The absolute difference, $Z - Z^*$ shown in the table helps in this regard. Thus, the relative gap in % measure is calculated as $100(\frac{Z-Z^*}{Z^*})$ and is plotted against the number of variables in Figure 4.5. As expected, the % gap increases with increasing number of variables. Similar trend can be observed with time and number of iterations against the number of variables (see Figure 4.6). Thus the algorithm has been tested using polynomials of varying degrees and bounds. Based on the results obtained, it can be seen that the algorithm is able to find good quality solutions within reasonable time.

Chapter 5

Intrinsic Barrier Function Algorithm

The BFA algorithm discussed in Chapter 4 utilizes an external logarithmic barrier function, which conforms to the properties required of it. Given the complexity of the potential energy equation of polypeptides, adding an external function might complicate an already complex objective function. Hence, in this chapter, we explore the possibility of using a particular term in the energy function as a barrier function. We also propose an algorithm, called Intrinsic Barrier Function Algorithm (IBFA), which utilizes the intrinsic barrier function and solves the problem in question. Part of the contents and results of this chapter was published in Ng *et al.* (2011).

5.1 Proposed Solution Method

Though a plethora of methods are available to solve nonconvex optimization problems that are similar to the one that we encounter in the protein structure prediction, interior point methods are quite uncommon in the area of *ab initio* methods. Hence, we propose a solution technique based on inherent barrier func-

tion to solve the formulation shown in (3.26). This involves using the steepest descent method for minimizing the transformed objective function.

5.1.1 Description of the Algorithm

From the potential energy equation of peptide systems given in (3.12),we can hypothetically treat the energy function as a combination of just the dihedral and electrostatic interactions and formulate the problem as given in (5.1).

Hypothetical Primal Problem

$$\begin{aligned} \text{Minimize } f(\Phi) = & \sum_{dihedrals} k_\phi(1 + cos(n\phi - \delta)) \; + \sum_{\substack{nonbonded \\ pairs}} \frac{q_i q_j}{\epsilon_1 r_{ij}} \\ \text{Subject to} & \\ & r_{ij}(\Phi) \geq 0, \\ & -\pi \leq \Phi \leq \pi, \end{aligned} \tag{5.1}$$

Here, r_{ij} is a function of the dihedral angle Φ. To handle the constraints in (5.1), a barrier function method is used. When added to the objective function, barrier functions prevent the generated points from leaving the feasible region. They generate a sequence of feasible points whose limit is a solution to the original problem. The requirement of a barrier function is that it should be continuous in the interior of feasible region and it takes a value of ∞ on its boundary. This would make sure that successive feasible points that are generated stay within the feasible region (Bazaraa *et al.*, 1993). In our problem, the term for van der Waals interaction turns out to be a good candidate for such a function and is given below:

$$\text{vdW}(\Phi) = \sum_{\substack{nonbonded \\ pairs}} \epsilon_{ij} \left[\left(\frac{R_{min_{ij}}}{r_{ij}(\Phi)} \right)^{12} - \left(\frac{R_{min_{ij}}}{r_{ij}(\Phi)} \right)^{6} \right]. \tag{5.2}$$

The van der Waals interaction term, vdW(Φ), is continuous over the region, $\{\Phi : \mathbf{r}(\Phi) > \mathbf{0}\}$, and approaches ∞ as the boundary of the region $\{\Phi : \mathbf{r}(\Phi) \geq \mathbf{0}\}$ is reached. If μ is the barrier parameter and the van der Waals interaction term is used as the barrier function, $B(\Phi)$, then the barrier problem can be formulated as follows:

Hypothetical Barrier Problem

$$\begin{aligned} &\min_{\Phi} \theta(\Phi, \mu) = \inf\{f(\Phi) + \mu B(\Phi) : r_{ij}(\Phi) \geq 0, -\pi \leq \Phi \leq \pi\} \\ &\text{where } B(\Phi) = \sum_{\substack{nonbonded \\ pairs}} \epsilon_{ij} \left[\left(\frac{R_{min_{ij}}}{r_{ij}(\Phi)} \right)^{12} - \left(\frac{R_{min_{ij}}}{r_{ij}(\Phi)} \right)^{6} \right]. \end{aligned} \tag{5.3}$$

Note that the constraints present in the original formulation (3.26) have been included in the objective function using the barrier function. Thus a series of problems are solved by decreasing the value of barrier parameter μ from a large initial value at every iteration and the optimal solution of the i^{th} iteration is used as an initial solution for the $(i+1)^{th}$ iteration. Algorithm 3 shows the Intrinsic Barrier Function Algorithm (IBFA) that we propose. For a given value of the barrier parameter, the method searches for a minimum point of the barrier function along the descent direction.

5.1.2 Method of Steepest Descent

The method of steepest descent, also called gradient descent method, proposed by Cauchy continues to be the basis of several gradient based solution procedures. The method uses first order approximation of the function being minimized. The method starts at an initial point, say, x_k and moves to the next point x_{k+1} by minimizing along the line extending from x_k in the descent direction, $-\nabla f(x_k)$.

Let $f : \Re^n \to \Re^1$ be a differentiable function in x. Given an initial point x_k,

Algorithm 3 Intrinsic Barrier Function Algorithm

Initialization Step
Let $\epsilon > 0$ be a termination scalar. Let $\mu_1 > 1,\ \beta \in (0,1)$ and $k = 1$. Let the randomly generated torsion angle Φ_1 be the starting solution.

Step 1:
Starting with $\Phi_k,\ \mu_k$, solve the following problem using the method of steepest descent:

$$\min_{\Phi} \quad \theta(\Phi, \mu)$$

Let Φ_{k+1} be a solution to the barrier problem; Go to *Step 2*.

Step 2:
If $\mu_k \leq 1$, solve the barrier problem using Φ_{k+1} and $\mu_{k+1} = 1$ as the initial points and stop. Otherwise let $\mu_{k+1} = \beta\mu_k,\ k \leftarrow k+1$ and go to *Step 1*.

the method of steepest descent iteratively finds the next point x_{k+1} such that $f(x_{k+1}) < f(x_k)$, where x_{k+1} is given by $x_{k+1} = x_k + \lambda d$. Here d is the direction of steepest descent of f at x_k, given by $d = -\nabla f(x_k)$ and λ is the step length satisfying the following:

$$\begin{gathered} \text{Minimize } f(x_k + \lambda d) \\ \text{Subject to } \lambda > 0 \end{gathered} \tag{5.4}$$

The method of steepest descent, though locates the local optima, has a very slow convergence rate when functions with long and narrow valleys are encountered. It also poorly performs as it reaches the optimum (Bazaraa *et al.*, 1993). Moreover, the method is highly dependent on the quality of the initial solution provided.

5.2 Generating Initial Solution

In order to generate a good quality initial solution for the IBFA algorithm, we propose a Heuristic for Initial Solution (HIS), based on a guided search through

the domain of the feasible region. The objective is to find a suitable set of dihedral angles that would minimize the energy function. The problem formulation is the same as in Section 3.4, where the variables are allowed to take on any values from -180° to 180°. The search procedure proposed here utilizes some problem specific ideas and is shown in Algorithm 5.2. From the energy function to be minimized shown in (3.12), it is obvious that in order for the functional value to be minimum, the variable, r_{ij} should be as big as possible. However, r_{ij}, the distance between the atoms i and j, cannot be infinitely big as it is constrained by the size of the molecule. Since atoms i and j are non-bonded atoms, they are not constrained by the fixed bond length. An increase in the value of r_{ij} could be obtained by increasing the bond angles. Since the bond angles are constants, the required effect could be achieved by varying the dihedral angle. This is achieved using the variables α and β, set at 0.5 and 0.25 respectively. The values of α and β used here have been found after trying out various combinations of α and β. Thus, a fraction of the bond angle is used to perturb the current set of dihedrals in a view to obtain new values that would minimize the energy function.

Consider atoms 1, 2, 3, and 4 connected in that order to form a dihedral in a protein. Then r_{ij} (r_{14}) is the distance between the atoms i (1) and j (4). Now, in order to increase the distance between the atoms 1 and 4, we increase the current torsion around 2 and 3 by a fraction of bond angles, $\angle$ 1-2-3 and $\angle$ 2-3-4. The variable *ichange* in the algorithm makes sure that after every fixed number of iterations, there is a sufficient change in the objective function value recorded. Failing which, the fraction of bond angle added to the torsion is increased to help break out of the situation which causes it. By no means, we are proposing this algorithm to obtain an optimal solution to the original problem. Our intention

Algorithm 4 Heuristic for Initial Solution

Let ϵ = objective function tolerance,
n = multiplication factor,
f_{old} = arbitrarily large value,
i_{max} = maximum number of iterations,
i_{chg} = no. of iterations for which the change in objective is less than ϵ,
Set i = 1,
n =1,
α =0.5,
β =0.25,
ϕ_{ct} $\in \Phi$ be initial set of torsion angles.

Repeat until $i < i_{max}$
Compute $f(i) \leftarrow V(\phi)$
If $f(i) < f_{old}$ **Then**
$f_{old} \leftarrow f(i),\ \phi_{new} \leftarrow \phi_{ct}$
Endif

If $i > i_{chg} * n$ **Then**
If $f(i) - f(i - i_{chg} * n) < \epsilon$ **Then**
$\phi_{new} = \phi_{ct} + \alpha \times bond\ angle$
Endif
$n \leftarrow n + 1$
Else
$\phi_{new} = \phi_{ct} + \beta \times bond\ angle$
Endif

If $\phi_{new} > 180$ **Then**
$$\phi_{new} = \phi_{new} - \left\lfloor \frac{\phi_{new}}{180} \right\rfloor \times 180$$
Endif

If $\phi_{new} < -180$ **Then**
$$\phi_{new} = \phi_{new} + \left\lfloor \left| \frac{\phi_{new}}{180} \right| \right\rfloor \times 180$$
Endif
$i \leftarrow i + 1$
End Repeat

is to rapidly generate a good solution which can be used as an initial solution to the IBFA algorithm. Algorithm 5.2 presents the pseudo code of the proposed method.

5.3 Computational Experience

In general, initial tests on performance of an algorithm are done on a standard set of problems for which the solution is known. Performing tests on such problems will help us to determine the ability of the proposed algorithm based on the quality of solutions obtained. Similar tests were done in Section 4.6 for BFA algorithm to gauge its performance. However, for IBFA algorithm we are using problem specific characteristics in the proposed method and this will render the standard test problems ineffective in this case.

In order to circumvent this, we use the widely studied model problem for molecular conformation, which is minimizing the Lennard-Jones potential. The objective is to find the minimum energy configuration of Lennard-Jones clusters. The scaled Lennard-Jones potential which is used in the computation is

$$\upsilon(r) = \frac{1}{r^{12}} - \frac{2}{r^6}, \tag{5.5}$$

where r is the distance of separation. The function in (5.5) is similar to the barrier function used in IBFA algorithm. Therefore using this function to generate test problems for IBFA would help to gauge the true potential of the proposed algorithm. Thus the following problem statement follows from Maranas & Floudas (1992):

> *Given N interacting particles, find their configuration(s) in three-dimensional Euclidean space involving the global minimum potential*

energy.

The mathematical formulation of the above-mentioned problem statement in (x_i, y_i, z_i) coordinate space can be written as follows:

$$\begin{aligned} \min V =& \sum_{i=1}^{N-1} \sum_{j=i+1}^{N} \upsilon_{ij} \\ \text{where } \upsilon_{ij} =& \frac{1}{[(x_i - x_j)^2 + (y_i - y_j)^2 + (z_i - z_j)^2]^6} - \\ & \frac{2}{[(x_i - x_j)^2 + (y_i - y_j)^2 + (z_i - z_j)^2]^3} \end{aligned} \tag{5.6}$$

The formulation in (5.6) is an unconstrained nonconvex optimization problem with large number of variables. Difficulties associated with solving the problem in (5.6) mainly involves dealing with the numerous local minima. Often, bounds on the interatomic distance and the energy function value are employed to constrain the feasible region of the problem. However, developing bounds and solution procedures applicable to the above-mentioned problem is not in the scope of our work. Our sole purpose of using (5.6) as test problem is to compare our solution with those already reported in the literature.

For this purpose we adapt the approach used in Gockenbach *et al.* (1997) to compare numerical results. Since the putative global minimum is known, the values of coordinates are perturbed so as to obtain a completely new coordinate, which will be used as a starting point. If p_i is the coordinate of the i^{th} atom, then the new starting point is obtained as follows

$$p_i = p_i + \rho u p_i, \tag{5.7}$$

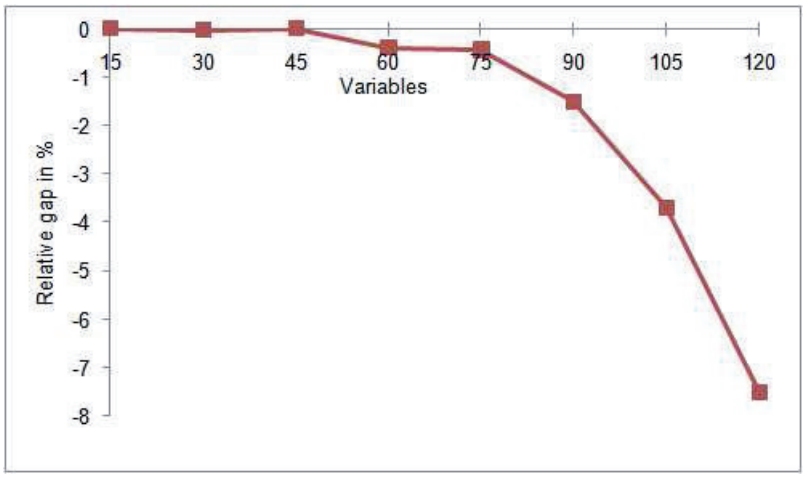
where ρ is the perturbation factor and u is a value from (pseudo-)random uniform distribution on $[-0.5, 0.5]$. The formulation in (5.6) has $3N$ variables for a total

Table 5.1: Numerical results for Lennard-Jones clusters

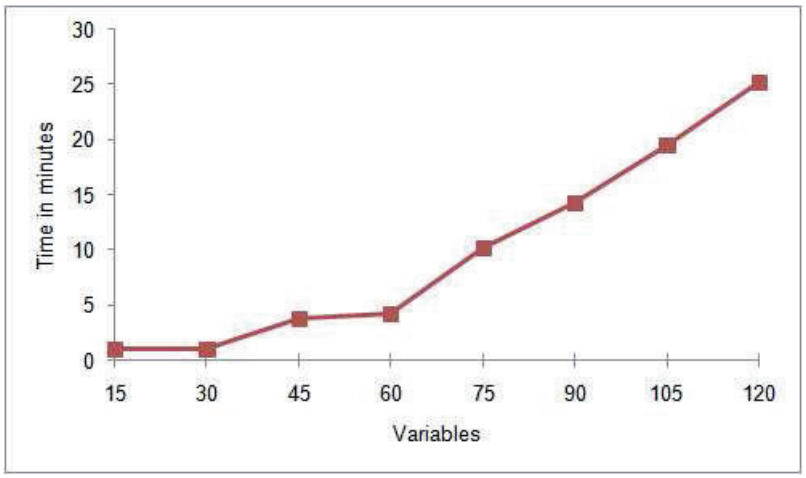
N	Variables	Putative Min (kcal/mol)	Energy Found (kcal/mol)	Time (min)	Relative Gap (%)
5	15	-9.1038	-9.1036	1.04	-0.0022
10	30	-28.4225	-28.4164	1.02	-0.0215
15	45	-52.3226	-52.3226	3.81	0.0000
20	60	-77.1770	-76.8713	4.18	-0.3961
25	75	-102.3726	-101.9281	10.19	-0.4342
30	90	-128.2865	-126.3547	14.21	-1.5058
35	105	-155.7566	-150.0031	19.54	-3.6939
40	120	-185.2498	-171.3761	25.16	-7.4892

of N participating atoms. In order to remove the translational and rotational degrees of freedom, we set $x_1, y_1, z_1, y_2, z_2, z_3$ to 0, i.e., we fix the first atom at the origin, second atom on the x-axis and the third atom on the xy-plane. Thus for a N-atom problem we have $3N - 6$ variables to describe the coordinates of N atoms.

The formulation (5.6) was solved using the IBFA algorithm for values of N ranging from 5 to 40 (15 to 120 variables). Setting the value of $\rho = 0.75$, the initial point is obtained as in (5.7). Hence, we do not use the HIS algorithm and directly employ the IBFA algorithm to solve the problem and the results obtained are shown in Table 5.1. The columns titled N and Variables list the number of atoms considered and the number of variables associated with the problem, respectively. The energy value found (V) by IBFA algorithm and the time taken to solve the problem are also reported. The table also lists the putative minimum (V^*) obtained from Gockenbach *et al.* (1997). All the computations were carried out on a PC with Intel Core 2 Duo processor running at 1.83 GHz and 1 GB of memory. The algorithms were implemented in MATLAB Version 7.2.



(a)



(b)

Figure 5.1: Effect of variables on (a) % Gap (b) Time

The barrier parameter, μ, in the algorithm is reduced from 100 to 1 by 5% at every iteration and the algorithm is terminated when $\mu \leq 1$. Then μ is set to 1 and the problem is solved again to obtain the final solution. The energy value found by IBFA very closely matches the putative minimum value. The relative gap in % measure is calculated as $100\left(\frac{V-V^*}{V^*}\right)$ and is plotted against the number of variables in Figure 5.1(a). As the number of variables increases, so does the

difference between energy value found and the putative minimum. For problems with variables less than 75, the relative gap is negligible and it reaches up to 7.5% for problems with 120 variables. From Figure 5.1(b), we can see a similar trend in the effect of variables on computational time. Based on the results obtained, we conclude that the performance of IBFA algorithm is very competitive.

Chapter 6

Application to Peptides

The main objective of this Chapter is to test the efficiency and the applicability of the proposed algorithms in finding the minimum energy conformation of peptides. While the ability of BFA and IBFA algorithms was demonstrated by solving the standard test problems in OR literature (see Section 4.6), its applicability to peptide systems is yet to be tested. Hence, the algorithms are used to solve a number of polypeptides to determine its minimum energy conformation. The results thus obtained are also compared with the solution found by other methods. All the computations were carried out on the same PC with Intel Core 2 Duo processor running at 1.83 GHz and 1 GB of memory. Both the algorithms were implemented using Matlab version 7.2. In order to generate the values for constants of the energy function and other interaction energy values, Tinker v4.2, a software suite developed by Ponder (2004) is used.

6.1 Computational Details

There are a variety of factors to be considered before actually solving the problem of minimum energy conformation. The type of peptide to be modeled, its corresponding data set for the parameters involved and the means to implement

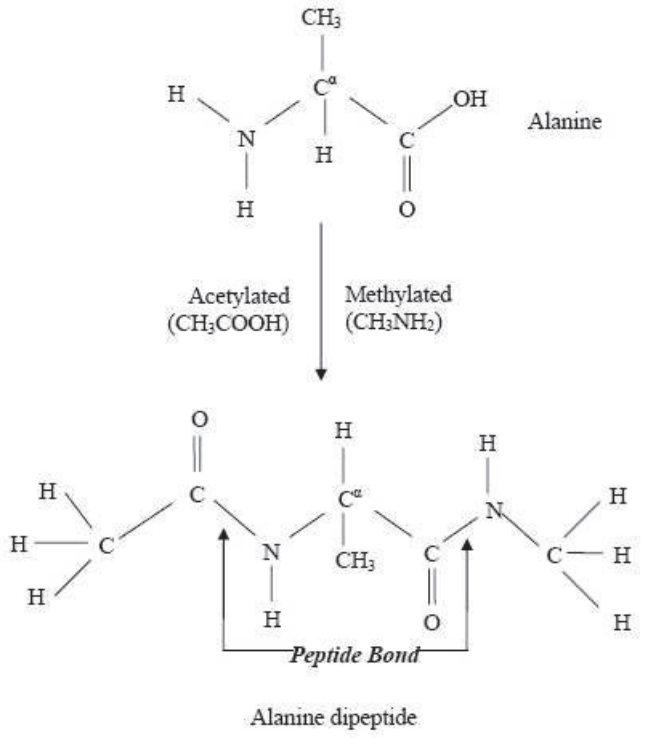


Figure 6.1: Blocking of alanine dipeptide

the coordinate conversions should be taken care of. In the following section, we explain the various factors and implementation details required for setting up the problem.

6.1.1 Dipeptide Structures

Dipeptides are nothing but a continuous chain of amino acids, which are frequently used to test the performance and robustness of newly developed algorithms. Hence, in order to test the efficiency of the proposed methods we adapt the dipeptide structures. Due to blocking of amino and carboxyl end groups, different forms of dipeptides of the same amino acid are available. Both the amino

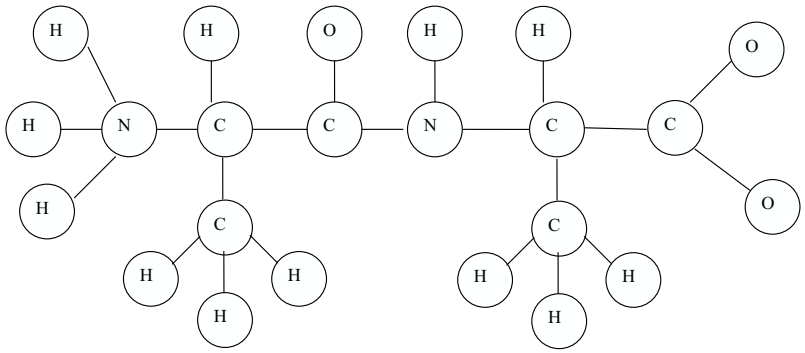


Figure 6.2: Schematic structure of di-alanine

and the carboxyl end group of the chain is replaced with the methyl group by the process of acetylation and methylation respectively. This creates two peptide bonds with a single amino acid. The process of converting the naturally occurring amino acid, alanine, into its dipeptide form is shown in Figure 6.1. In order to reduce the computational cost, sometimes the analogues of dipeptides are also used. For our research, we consider the di-alanine formed when two alanine amino acids are joined together by a peptide bond. Figure 6.2 shows the schematic structure of di-alanine, which has 23 atoms connected by 22 bonds. It has 39 triples (bond angles) and 49 dihedrals.

6.1.2 Parameters

The equation for the energy function involves a lot of constants that are specific to the type of atoms that are involved in a particular interaction. Moreover,

bond lengths and bond angles of atoms are also required to model and solve the problem. Values for these constants and other parameters are determined via experimental techniques or *ab initio* methods and is a complex process by itself. Such parametrization is available for different energy functions and we used the one that is consistent with the CHARMM force field. In order to generate the required values, Tinker v4.2, a publicly available software suite developed by Ponder (Ponder, 2004) is used. We use the CHARMM27 parametrization data that is provided by the software for our calculations.

6.1.3 Coordinate Conversions

The term r_{ij}, which appears in the objective function represents the Euclidean distance between the atoms i and j and is a function of internal coordinates (bond lengths, angles and dihedrals). Unfortunately, computing distances using the internal coordinates is extremely difficult and is not advocated in case of optimization problems where it has to be executed repeatedly. Hence, conversion to a cartesian system of coordinates is imperative. One of the efficient algorithms for this has been proposed in Thompson (1967), and is often used for performing the conversions (Byrd *et al.*, 1996; Floudas, 2000; Lim, Beliakov & Batten, 2003).

Consider four atoms, 1,2,3 and 4 that are connected to form a chain. A base coordinate system is defined by the positions of atoms 1, 2 and 3 by fixing atom 1 at the origin and atom 2 on the negative x-axis at a distance of r_{12} (bond length). Now, the 3^{rd} atom could be placed anywhere on the x-y plane with the bond length and bond angle information. Now, subsequent atoms could be fixed in the sequence if we know the bond length, bond angle and dihedral of the corresponding atom. A series of equations have been derived in Thompson

(1967) and we have adapted those to perform the coordinate conversions for our problem.

For example,let the position of first three atoms in a sequence be fixed, i.e., the first one is fixed at the origin, $(0,0,0)$, the second one is positioned at $(-l_2,0,0)$ and the third one at $(l_3 cos\theta_3 - l_2, l_3 sin\theta_3, 0)$, where the variable l_k denotes the bond length between the atoms k and $k-1$. A conversion scheme for m atom sequence, with bond angle, θ and dihedral angle, ϕ is detailed below:

$$\begin{bmatrix} x_m \\ y_m \\ z_m \\ 1 \end{bmatrix} = B_1 \; B_2 \ldots B_m \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} \quad \forall m = 1, ..., n, \tag{6.1}$$

where $x_m,\ y_m,\ z_m$ represents the three-dimensional cartesian coordinates of the m^{th} atom and the matrices $B_1,\ B_2, ..., B_m$ are given as in (6.2) and (6.3).

$$B_1 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad B_2 = \begin{bmatrix} -1 & 0 & 0 & -l_2 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \tag{6.2}$$

$$B_i = \begin{bmatrix} -cos\theta_i & -sin\theta_i & 0 & -l_i cos\theta_i \\ sin\theta_i cos\phi_i & -cos\theta_i cos\phi_i & -sin\phi_i & l_i sin\theta_i cos\phi_i \\ sin\theta_i sin\phi_i & -cos\theta_i sin\phi_i & cos\phi_i & l_i sin\theta_i sin\phi_i \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad \forall i = 3, ..., m. \tag{6.3}$$

Thus with the explicit expressions for the cartesian coordinates, $x_m,\ y_m,\ z_m$, the Euclidean distance, r_{1m}, can be found as $\sqrt{x_m^2 + y_m^2 + z_m^2}$.

6.2 Computational Results

6.2.1 Problem Background

We intend to test the proposed algorithms with the di-alanine structure discussed in Section 6.1.1. There are a total of 49 dihedral angles present in alanine dipeptide, including the backbone dihedral angles. We consider different number of

dihedral angles as variables to test the computational efficiency of the algorithm developed. Such an experiment also helps to identify several minimal energy conformations of the peptide that is considered. The minimum energy conformations that were identified by our method, can be used as initial conformers for other programs and would hence reduce the overall computational cost in other applications, such as protein structure prediction, peptide docking and drug design. The work in this paper also illustrates the possibility of exploiting the structure of physical functions encountered so that suitable computational methods can be used to solve the underlying optimization problem effectively.

It is common to consider only 2 to 5 variables for determining the minimum energy conformation of di-alanine. This is done to reduce the computational load and the accurate empirical value of energy function is derived by interfacing the solution method developed with other force field programs available. We vary the number of dihedrals (variables) considered for each experiment and do not interface with any of the force field programs available. The energy value reported is completely calculated using the solution method developed. The dihedrals, van der Waals and electrostatic interaction energy are calculated only for the number of participating dihedral angles and it is due to this that the energy values are different in all the four cases. Moreover, we allow the torsional angles to take on any value between $-\pi$ and π to determine the minimum energy configuration. All the computations were carried out on a PC with Intel Core 2 Duo processor running at 1.83 GHz and 1 GB of memory. The algorithms were implemented in MATLAB Version 7.2.

6.2.2 Computational Experience of BFA

The BFA algorithm was used to solve the energy conformation problem of di-alanine and the results are reported in Table 6.1. The analytic center of the feasible region was chosen to be the initial iterate for the algorithm. The initial value of barrier parameter (μ) in our Algorithm 1 is set to 100 and is reduced by a factor of $0.95(\theta_\mu)$ when $\epsilon_D \leq 0.01$. The method terminates when $\epsilon_\mu < 0.01$. We also ran the algorithm repeatedly from different set of starting points and each run always converged to the same minimum solution which is reported.

The Var column in Table 6.1 refers to the number of dihedral angles considered for that experiment, while V_{start} & V_{end} refer to the energy values in kcal/mol of the starting and ending conformation, respectively. The number of atomic interactions that were considered for each experiment are listed under the column heading Interactions. The value of dihedral angles ϕ and ψ are also reported for the minimum energy conformation found. The last column, Itns refers to the total number iterations required to determine the reported minimum energy value. The number of atomic interactions reported here is important because it forms a core component of the total energy function. Moreover, for each interaction considered, the distance between the end atoms (r_{ij}) has to be calculated, thereby increasing the computational cost.

For the 2-variable problem, we consider only the backbone atoms, excluding the side chain atoms, and fix the torsion around the peptide bond, ω, to 180 °. In the case of 5 variables, we include the two side chain carbon atoms and also allow ω to vary between $-\pi$ and π. For the 15-variable problem, we include the end group hydrogen atoms and oxygen atoms along with the hydrogen and

Table 6.1: Minimum energy values of di-alanine computed via BFA

Var	V_{start} (kcal/mol)	V_{end} (kcal/mol)	Time (sec)	Interactions	ϕ (deg)	ψ (deg)	Itns
2	64.48	27.78	14	6	-0.17	-2.38	17
5	83.72	25.64	16	13	0	180	43
25	286.72	-147.61	528	73	76.24	107.13	156
49	48.39	-231.56	3947	192	-83.26	-47.64	258

oxygen atoms that form the peptide plane. The complete structure of di-alanine is considered for the 49-variable case. Generally the hydrogen bond interactions are not included and a cut-off distance is also used to reduce the computational load. However, we do not consider such assumptions so that we could study the structure in its entirety.

6.2.3 Computational Experience of HIS and IBFA

In this Section, we discuss our computational experience of using HIS and IBFA algorithms to determine the minimum energy conformation of di-alanine. Before invoking the IBFA algorithm, the HIS algorithm is utilized to find a good initial point for the IBFA algorithm. The underlying premise of HIS is that, by increasing the distance between end atoms, the energy function value would decrease. This is done by adding a fraction of the bond angle to the dihedral under consideration which was detailed in Section 5.2. The number of variables in the peptides considered is varied and the minimum energy conformation found for each of them is shown in Table 6.2. In all the cases where ω is fixed at 180^o, understandably, the energy value obtained has been better than the other cases, which is due to the extended planar structure of the peptide at that dihedral val-

Table 6.2: Minimum energy values of di-alanine computed via HIS

Var	V_{start} (kcal/mol)	V_{end} (kcal/mol)	Time (sec)	Interactions	ϕ (deg)	ψ (deg)	Itns
2	42.23	27.88	1.98	6	174.00	177.00	692
5	1.4×10^4	27.05	4.31	13	-113.25	-177.37	242
25	5.3×10^6	-32.75	25.74	73	-120.00	52.00	537
49	23.28	-56.05	71.75	192	89.00	179.00	916

Table 6.3: Minimum energy values of di-alanine computed via IBFA

Var	V_{start} (kcal/mol)	V_{end} (kcal/mol)	Time (sec)	Interactions	ϕ (deg)	ψ (deg)	Itns
2	27.88	27.86	8	6	174.73	176.90	90
5	27.05	25.11	12	13	-179.52	-176.98	90
25	-32.75	-149.54	354	73	112.00	68.00	90
49	-56.05	-229.89	3667	192	-85.33	-53.40	90

ues. For each instance, 1000 iterations were run in order to perform an exhaustive search. The lowest energy value found is recorded and the iteration in which it was obtained is also reported.

The difference in the energy between the starting conformation and the ending conformation, as presented in Table 6.1, shows the efficiency of the IBFA algorithm. The reason for the difference being less in the first two cases is the ability of HIS algorithm to identify the minimum energy configuration. The barrier parameter, μ, in the IBFA algorithm is reduced from 100 to 1 by 5% at every iteration. In a general barrier function method, the barrier parameter is usually reduced to close to zero, at which point, the augmented objective function becomes close to the original objective function and the solution obtained at that instance is considered to be an approximate solution for the original problem.

In our case, since we use the van der Waals function which is inherently present in the objective function as the barrier function, allowing the barrier parameter to converge to zero would not solve the original problem. Hence, the framework of the algorithm is altered to suit the barrier function that we are using. The augmented objective function will resemble the original objective function when $\mu = 1$. Therefore, while reducing the value of μ at every iteration, the algorithm is terminated when $\mu \leq 1$. At this point, we set $\mu = 1$ and use the optimum solution obtained in the preceding iteration as the initial point to solve the problem again.

Generally, a barrier algorithm is terminated when μ approaches 0. However, in the proposed BFA algorithm, we intend to terminate the algorithm when $\mu \leq 1$ due to the aforementioned reasons. In order to confirm if this affects the quality of solution obtained, we performed some experiments in which we allowed μ to approach 0, and the solution obtained was used as an initial solution to solve the original problem. These experiments showed that the quality of solutions obtained in such settings were much inferior to what was obtained earlier. Hence, based on this inference we terminate the algorithm when the barrier parameter, $\mu \leq 1$. Such an early termination also has an advantage of avoiding ill-conditioning issues encountered in barrier function methods when the barrier parameter approaches 0. Moreover, it also helps to avoid getting trapped at a local solution.

6.2.4 Computational Experience of Genetic Algorithm

While seeking to compare the performance of our method with other methods in the literature, we do not find much work that solves the problem under the same assumptions or conditions adopted in our work. As an example, even though

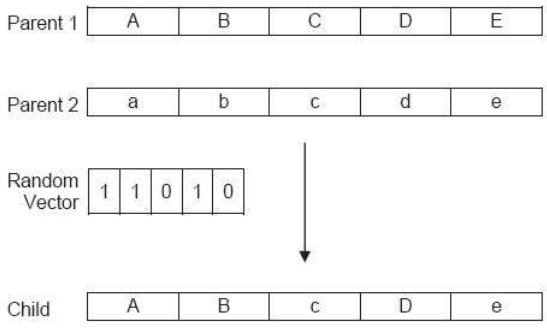


Figure 6.3: Example of crossover operation

the αBB approach in Maranas *et al.* (1996) belongs to the *ab inito* methods, the results reported are for blocked dipeptide structures by interfacing the algorithm with other energy programs and holding the dihedral angles at known constant values. Moreover, the αBB approach uses the ECEPP energy function. Due to the difference in assumptions, parameter values and even the different energy functions used, it is difficult to find a benchmark against which we can compare. Hence, we have instead used a genetic algorithm approach to compare with the performance of the proposed methods. The CHARMM energy function (3.25) was used as the fitness function with the variables taking on values between -180° to 180°. The genetic algorithm was implemented with a scattered crossover function which generates a random binary vector and selects the genes from parent 1 if the component of a random vector is 1, and the genes from parent 2 if the component of that random vector is 0. This crossover operation is illustrated in Figure 6.3. The mutation operation was achieved using a crossover fraction, which determines the percentage of crossover children in the next generation without including the elite children. The crossover fraction is varied from 0 to 1, by a factor of 0.05 at every run of the algorithm. Starting from an initial population of 20,

Table 6.4: Comparison of results from BFA, IBFA and GA

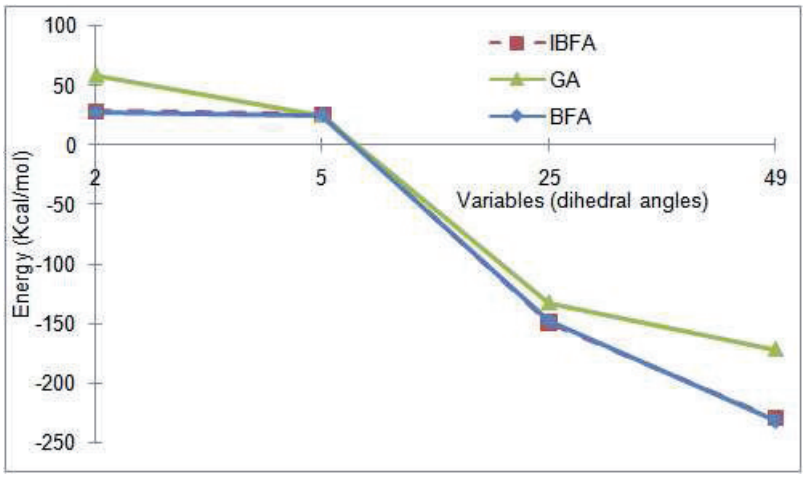
Variables	Energy (kcal/mol)			Time (sec)		
	BFA	IBFA	GA	BFA	IBFA	GA
2	27.78	27.86	58.52	14	8	144
5	25.64	25.11	25.13	16	12	131
25	-147.61	-149.54	-132.54	528	354	582
49	-231.56	-229.89	-171.69	3947	3667	1530

the algorithm is terminated when the population size reaches 500. This genetic algorithm was also implemented in MATLAB.

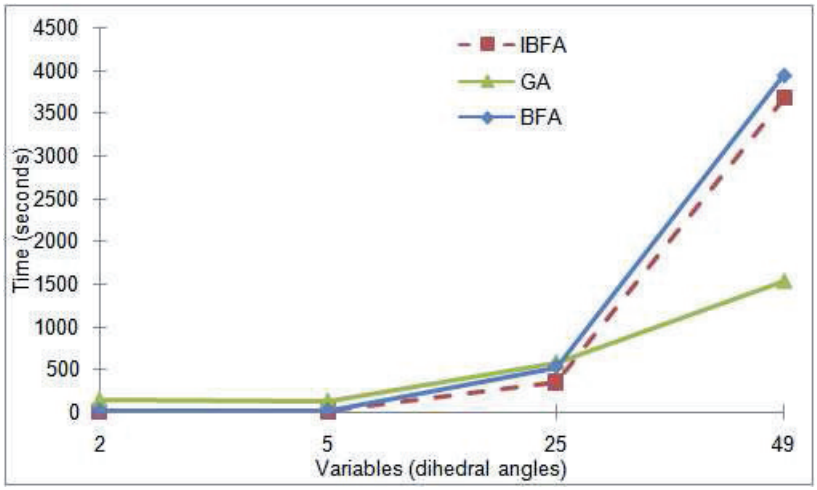
The results obtained by the genetic algorithm are presented in Table 6.4 and compared against the results of BFA and IBFA. It can be inferred from the table that both the BFA and the IBFA method locates a minimum conformation which is better than the one found by the genetic algorithm method. A comparison of energy value found and the computation time required by BFA, IBFA and GA is shown in Figure 6.4. From the figure, we also infer that GA is computationally more expensive than BFA and IBFA. Though, the time taken by BFA and IBFA methods is more than that of GA for the 49 variables case, it is compensated by the significant improvement in the energy values identified.

6.2.5 Application to Polyalanines

In this section, we discuss the computational experience of applying the proposed solution approaches to larger peptide systems. For this purpose, we adapt the structure of polyalanines, AcNH-$(\text{Ala})_n$-CONHCH_3, where n is the number of alanine residues considered in the study. The minimum energy conformation is determined by considering two dihedral angles (ϕ/ψ) as variables for each of the



(a)

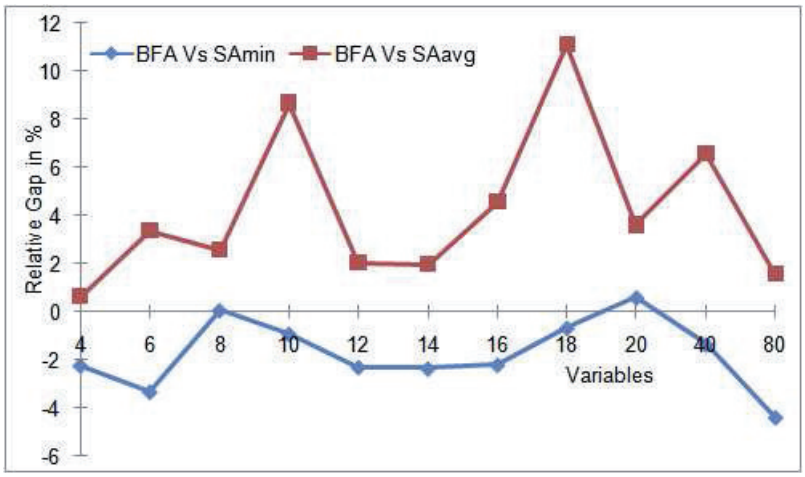


(b)

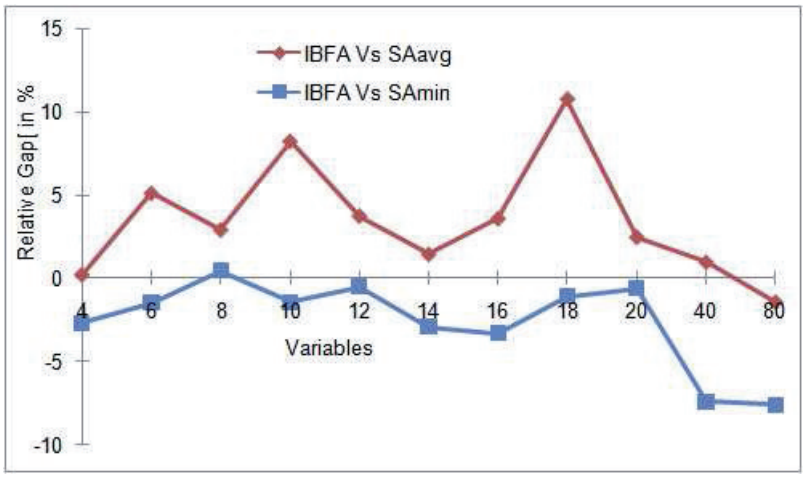
Figure 6.4: Comparison of results from BFA, IBFA and GA for (a) Energy value determined (b) Computational time

alanine residue in a given polyalanine. This particular structure has been studied using simulated annealing (SA) in Wilson & Cui (1990). The energy values found by the SA approach is compared with that of the BFA and IBFA methods. Table 6.5 provides a detailed comparison of the energy values and the time taken to solve the problem by the aforementioned methods. Energy values in Wilson &

Cui (1990) are reported in KJ/mol, whereas the energy values calculated by our algorithm are in kcal/mol. In order to facilitate ease of comparison, the energy values in KJ/mol are converted to kcal/mol using 1 KJ/mol = 4.2 kcal/mol.



(a)



(b)

Figure 6.5: Comparison of energy values obtained (a) BFA Vs SA (b) IBFA Vs SA

In the SA approach, each problem is solved 10 times and the results are reported for each run. In Table 6.5, the columns Min Energy and Avg Energy

Table 6.5: Comparison of results for polyalanines

	No. of	SA Approach (Wilson & Cui, 1990)			BFA Approach		IBFA Approach	
	Variables	Min Energy	Avg Energy	Time	Energy	Time	Energy	Time
n	(dihedrals)	(kcal/mol)	(kcal/mol)	(min)	(kcal/mol)	(min)	(kcal/mol)	(min)
2	4	-24.55	-23.86	2.42	-24.01	0.23	-23.91	0.20
3	6	-36.15	-33.81	3.93	-34.98	0.35	-35.62	0.37
4	8	-50.20	-48.96	4.35	-50.23	0.57	-50.42	0.60
5	10	-64.16	-58.07	6.00	-63.57	0.80	-63.25	0.97
6	12	-79.05	-75.71	15.41	-77.26	0.98	-78.64	1.20
7	14	-94.04	-90.06	9.98	-91.86	1.52	-91.37	1.63
8	16	-109.15	-101.90	12.34	-106.78	2.45	-105.67	2.43
9	18	-124.22	-109.70	14.03	-123.38	4.27	-122.87	4.02
10	20	-139.43	-135.22	14.51	-140.26	8.30	-138.63	5.07
20	40	-291.45	-268.73	144.00	-287.52	80.58	-271.31	67.25
40	80	-528.58	-498.40	296.10	-506.26	212.08	-491.37	189.37

correspond to the minimum value and the average value of the energy found in 10 runs, respectively. The time taken per run in minutes is also reported for the SA approach. The energy value found and time taken for both the BFA and IBFA approach are also reported.

From Table 6.5, we see that the energy values determined by BFA and IBFA are consistently lower than the average energy value determined by the SA method. While comparing the results obtained with the minimum energy determined by the SA method, the results are mixed. In order to understand the results of comparison better, we calculate the relative gap (in %) between the energy values reported as follows:

$$\begin{aligned} \xi_1^B &= 100 \times \left(\frac{E_{BFA} - SA_{\min}}{E_{BFA}} \right), \\ \xi_2^B &= 100 \times \left(\frac{E_{BFA} - SA_{avg}}{E_{BFA}} \right), \end{aligned} \tag{6.4}$$

where $E_{BFA}, SA_{\min}$ and SA_{avg} denote the energy values reported by the BFA method, minimum energy reported by SA method and the average energy reported by SA method, respectively. ξ_1^B & ξ_2^B denote the corresponding relative gap in % measure. The values of ξ_1^B & ξ_2^B are plotted against the number of variables involved in that problem in Figure 6.5(a). Similar graph is also plotted in Figure 6.5(b) to study the performance of IBFA algorithm against the SA approach.

The IBFA's results are better when compared to that of the average energy values reported by the SA approach. While the IBFA matches the minimum energy found by SA in some cases, the difference is more pronounced as the variable size increases. The BFA method also compares with the SA method in a fashion similar to that of IBFA. While the trend is similar, the % deviation is

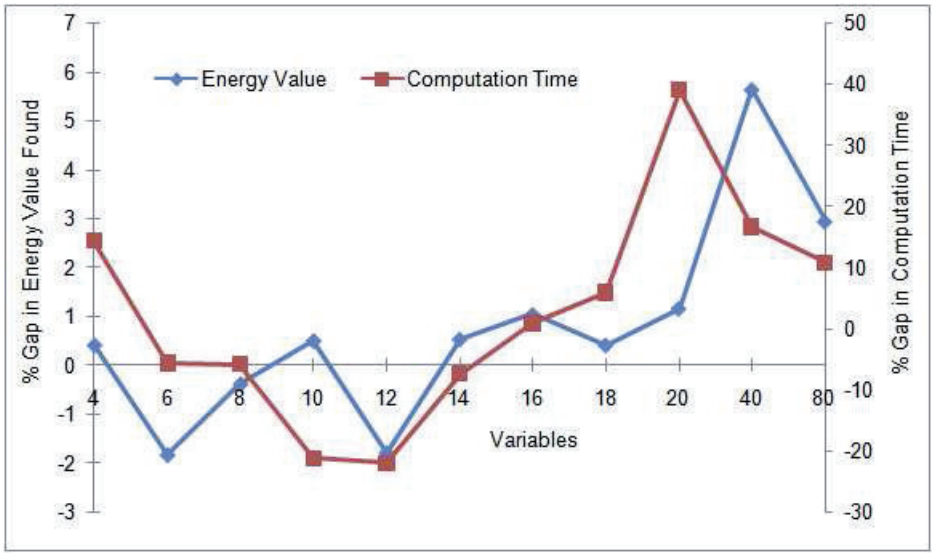


Figure 6.6: Performance comparison of BFA and IBFA

much lesser in BFA. It should be noted that the SA approach utilizes an energy function which is different from what we have used. From the results, we can also see that the time taken by each of the BFA and IBFA approach is much lesser than that required by the SA approach. Although both approaches use different energy functions, the results indicate that both BFA and IBFA approaches are able to obtain comparable energy values in lesser time.

In order to study the performance comparison between BFA and IBFA Figure 6.6 is plotted. Since the energy values and computation time of both BFA and IBFA are very close to each other, plotting the absolute value will be of no avail. Hence, we plot the % deviation of BFA's solution from that of IBFA's. Similar to (6.4), the relative gap (in %) between the BFA's solution and IBFA's solution

is calculated as given in (6.5) and is plotted in Figure 6.6.

$$\begin{aligned} \kappa &= 100 \times \left(\frac{E_{BFA} - E_{IBFA}}{E_{BFA}} \right), \\ \tau &= 100 \times \left(\frac{T_{BFA} - T_{IBFA}}{T_{BFA}} \right), \end{aligned} \tag{6.5}$$

where E_{IBFA}, T_{IBFA} and T_{BFA} denote the energy value reported by the IBFA method, computational time required for the IBFA method and the BFA method, respectively. κ and τ denote the corresponding relative gap in % measure.

Figure 6.6 shows that BFA finds the minimum energy configuration in most of the cases and in particular, as the variable size increases, BFA's solution is much better than that of IBFA. With respect to computational time, BFA takes lesser time than that of IBFA initially and as the variable size increases, the time taken by BFA is more than that of IBFA. However, the increase in computational time is compensated by the quality of solution found.

6.3 Application to Lennard-Jones Clusters

In order to gauge the performance of the BFA and IBFA algorithms for bigger-sized problems, the Lennard-Jones cluster problem discussed in Section 5.3 is utilized. Both the BFA and IBFA algorithms are used to solve the problem with variables ranging from 60 to 510. In order to compare our results with that of other methods, we refer to the hybrid approach proposed by Zhang (2011). The hybrid method uses the combination of discrete gradient method for the local search phase and simulated annealing for the global search phase. Results obtained from BFA and IBFA method are presented in Table 6.6 along with that of the hybrid approach.

In Table 6.6, N represents the number of atoms in the Lennard-Jones cluster

Table 6.6: Comparison of results for Lennard-Jones clusters

		Energy Values (kcal/mol)			
N	No. of Variables	Putative Minimum	Hybrid Method	BFA	IBFA
20	60	-77.177043	-77.177043	-77.177038	-76.871300
23	69	-92.844472	-92.844461	-92.844232	-92.695193
25	75	-102.372663	-102.372663	-102.372631	-101.928100
27	81	-112.873584	-112.825517	-112.867814	-112.685649
30	90	-128.286571	-128.09696	-128.089248	-126.354700
34	102	-150.044528	-150.044528	-150.044437	-148.953821
44	132	-207.688728	-207.631655	-207.644635	-207.229583
49	147	-239.091864	-239.091863	-239.090741	-238.693910
56	168	-283.643105	-283.324945	-283.378529	-282.195297
65	195	-334.971532	-334.014007	-333.984813	-332.847311
84	252	-452.657214	-452.26721	-452.463515	-451.869512
93	279	-510.877688	-510.653123	-509.647385	-508.775928
148	444	-881.072971	-881.072948	-879.758314	-876.489319
170	510	-1024.791797	-1024.791771	-1022.649288	-1015.739136

and the second column denotes the number of variables considered in the problem. The column Putative Minimum gives the best known global optimum value. The remaining columns give the energy values obtained from the respective methods. Based on the results, we see that the BFA algorithm is able to provide results close to the putative minimum. The results of BFA algorithm are generally close to that of hybrid algorithm for variables up to 279. As the variable size increases, the quality of the solution obtained by BFA slightly decreases when compared to the hybrid method. IBFA's performance when compared to that of BFA and hybrid method is on the lower side. Even though IBFA finds solutions in the vicinity of putative minimum, the quality of the solution is lower when compared to the other methods. Thus it can be seen that both the proposed methods are competitive and has the ability to find good solution(s).

Chapter 7

Conclusions and Future Work

The primary focus of this thesis is to develop solution methods to determine the minimum energy conformation of polypeptides. The solution methods developed here could be extended to other areas of computational biology as well. Conclusions and further work to be done are discussed in this chapter.

7.1 Conclusions

In summary, we have developed interior-point methods to solve nonlinear nonconvex optimization problems with box constraints. Interior-point methods, seldom used in the area of computational biology was effectively utilized to solve the problem of minimum energy conformation of polypeptides.

It is particularly important to have a set of low energy conformations if a number of populated states are present (Wilson & Cui, 1990). First pass optimization methods play a vital role in identifying a set of low energy conformations. These low energy conformations can be used to approximate the entropic contributions associated with the stability of the molecule. Once a sufficient ensemble of low energy minima has been identified, a statistical analysis can be used to estimate the relative entropic contributions (Klepeis & Floudas, 1999). Methods such as

the one proposed in this paper help to identify both the stable three-dimensional structure (global minimum), as well as a set of low energy conformations (local minimum). The advantages of *ab initio* methods as proposed by McAllister & Floudas (2010) lies in its ability to

- predict structures when a related structural homologue is not available
- extend the predictions to different environments
- provide insight into the mechanism, thermodynamics, and kinetics of protein folding

Moreover, new structures continue to be discovered, which would not be possible by methods that rely on comparison to known structures (Floudas *et al.*, 2006).

Two approaches, namely BFA and IBFA have been proposed. Both the methods utilize a barrier function to transform a constrained problem into an unconstrained problem or into a sequence of unconstrained problems. The difference lies in the type of barrier function that was utilized. While BFA employs an external barrier function, IBFA utilizes the vdW term in the energy function as the barrier function. This illustrates the possibility of exploiting the structure of physical functions encountered so that suitable computational methods can be used to solve the underlying optimization problem effectively. Both the methods have been tested with standard problems in the literature before applying them to solve polypeptide structures. BFA in particular was tested with polynomials of higher degrees. The performance of both, BFA and IBFA was found to be encouraging. The results were also compared with that of a genetic algorithm implementation.

Interior-point methods are highly dependent on the initial solution provided. Hence, for both the methods it is imperative to have a good quality initial solution. The starting solution provided might influence the quality of final solution obtained. While BFA utilizes the analytic centre of the feasible region as an initial solution, IBFA uses the HIS algorithm to find a good starting solution. Barrier parameters are set to a constant value for each subproblem that is being solved. It would be helpful to dynamically update the barrier parameter value based on the variable it is associated with. Such an approach would help us to have more control on the behavior of variables involved. One could also consider using other types of barrier functions to solve the problem of minimum energy conformation. Improvement in terms of performance could also be achieved by considering other search directions and line search procedures.

7.2 Future Work

The problem of protein structure prediction, is nothing but minimizing a non-convex potential energy equation which possess a plethora of local minima points in the multivariable potential energy hyperspace. Though the focus of this thesis is on interior-point algorithms for determining minimum energy conformation of polypeptides, it is possible to extend and adapt the algorithm to solve optimization problems arising from other areas. The following section elaborates the possible future work.

7.2.1 Molecular Structure Prediction

Atoms, the building blocks of molecules remain the same in every molecule. It is only the orientation of the atom that changes with different molecules calling

for methods to predict the molecular structure. Similar to proteins, there are several force fields that are developed for determining the total potential energy of the molecule. The assumption that the most energetically stable conformation of the molecule is the one that corresponds to the global minimum potential energy holds good here as well. The difference between protein and molecular structure prediction is in the potential energy equation and the interaction terms that are involved in it. Since the problem structure is so similar an extension into this area should only be natural. Maranas & Floudas (1994a) and Maranas & Floudas (1994b) gives an in-depth information regarding the energy functions and implementation aspects pertaining to molecular structure prediction methods.

7.2.2 Peptide Docking

The problem of peptide docking comes as a natural extension of the protein folding problem. It requires identification of equilibrium structures for a macromolecule-ligand complex which highlights the complexity of the problem. The free energy equation which accounts for solvation terms is used as the objective function for this problem. The most obvious and most difficult approach would be to optimize the entire system of two interacting peptides.

Generally, the first step in solving the problem is the identification of a "pocket" or the binding site. A mathematical model accounting for all the interactions of the specific pocket and a naturally occurring amino acid is developed. Any of the protein force fields along with solvation terms could be used to model the energy function. The difference between the global minimum energy of the complex and that of the naturally occurring amino acid is calculated and used as a measure to gauge the binding affinity between the pocket and the given amino

acid. Androulakis *et al.* (1997) details the prediction of peptide docking to a particular protein using the αBB algorithm.

7.2.3 Incorporating Sequence-Structure Relations

It is of our interest to predict only the tertiary structure as it is only at this native structure the protein performs the function it is intended to. The other forms, such as the primary and secondary structure are extremely short-lived and do not have any impact directly on the end function. But, the information of the secondary structures such as $\alpha-$helix, $\beta-$sheets and coils could be used in the prediction of the tertiary structure. When a particular sequence of amino acids occur, based on the data available, it is possible to say what kind of secondary structure it would adapt. From this information, angle and distance restraints could be derived and used. However, resorting to information other than the sequence of amino acids contradicts with the idea of *ab initio* prediction methods, which does not use any external information. With the rapid improvement in the prediction methods the boundaries between different classes of prediction methods have been blurred (Floudas *et al.*, 2006) and is generally accepted to include some external information which could aid the prediction process.

Moreover, biological data are available in plenty at several databases that are maintained around the globe and is publicly available. Available data for a particular protein under study could be used to infer details which can be included in the problem formulation as constraints. Sometimes partial data from failed NMR experiments is also available which can be used to tighten the feasible space. Information pertaining to distance between atoms and bond angles of atoms involved can also be deduced and used accordingly.

Bibliography

Adjiman, C.S., Dallwig, S., Floudas, C.A. & Neumaier, A. (1998). A global optimization method, αbb for general twice-differentiable NLPs - I. Theoretical Advances. *Computers & Chemical Engineering*, **22**, 1137–1158.

Al-Mekhnaqi *et al.*, A.M. (2009). Prediction of protein conformation in water and on surfaces by monte carlo simulations using united-atom method. *Molecular Simulation*, **35**, 292–300.

Altschul, S.F., Gish, W., Miller, W., Myers, E.W. & Lipman, D.J. (1990). Basic local alignment search tool. *Journal of Molecular Biology*, **215**, 403–410.

Altschul, S.F., Madden, T., Schaffer, A.A., Zhang, J., Zhang, Z., Miller, W. & Lipman, D. (1997). Gapped BLAST and PSI-BLAST: A new generation of protein database search programs. *Nucleic Acids Research*, **25**, 3389–3402.

Andonov, R., Balev, S. & Yanev, N. (2004). Protein threading: From mathematical models to parallel implementations. *INFORMS Journal on Computing*, **16**, 393–405.

Androulakis, I.P., Maranas, C.D. & Floudas, C.A. (1995). *αbb*: A global optimization method for general constrained nonconvex problems. *Journal of Global Optimization*, 337–363.

Androulakis, P., Nayak, N.N., Ierapetritou, M.G., Monos, D.S. & Floudas, C.A. (1997). A predictive method for the evaluation of peptide binding in pocket 1 of hla-drb1 via global minimization of energy interactions. *Proteins*, **29**, 87–102.

Anfinsen, C.B. (1973). Principles that govern the folding of protein chains. *Science*, **181**, 223–238.

Bazaraa, M.S., Sherali, H.D. & Shetty, C.M. (1993). *Nonlinear Programming: Theory and Algorithms*. John Wiley and Sons, New York, 2nd edn.

Bhattacharya, D. & Cheng, J. (2013). 3drefine: Consistent protein structure refinement by optimizing hydrogen bonding network and atomic-level energy minimization. *Proteins: Structure, Function, and Bioinformatics*, **81**, 119–131.

Blommers, M.J.J., Lucasius, C.B., Kateman, G. & Kaptein, R. (1992). Conformational analysis of a dinucleotide photodimer with the aid of the genetic algorithm. *Biopolymers*, **32**, 42–52.

Brain, Z. & Addicoat, M. (2011). Optimization of a genetic algorithm for searching molecular conformer space. *Journal of Chemical Physics*, **135**.

Branden, C. & Tooze, J. (1991). *Introduction to Protein Structure*. Garland Publishing, Inc.

Brooks, C., M.Karplus & B.M.Pettitt (1988). *Proteins: A theoretical Perspective of Dynamics, Structure and Thermodynamics*. John Wiley & Sons, New York.

Byrd, R.H., Eskow, E., van der Hoek, A., Schnabel, R.B., Shao, C.S. & Zou, Z. (1996). Global optimization methods for protein folding problems. In P.M. Pardalos, D. Shalloway & G. Xue, eds., *Global Minimization of Nonconvex Energy Functions: Molecular Conformation and Protein Folding*, vol. 23, 29–39, American Mathemetical Society.

Byrd *et al.*, R.H. (1996). Global optimization methods for protein folding problems. In P.M. Pardalos, D. Shalloway & G. Xue, eds., *Global Minimization of Nonconvex Energy Functions: Molecular Conformation and Protein Folding*, vol. 23, 29–39, American Mathemetical Society.

Chothia, C. & Lesk, A. (1986). The relation between the divergence of sequence and structure in proteins. *The EMBO Journal*, **5**, 823–836.

Cornell, W.D., Cieplak, P., Bayly, C., Gould, I., Merz Jr, K.M., Ferguson, D., Spellmeyer, D., Fox, T., Caldwell, J. & Kollman, P. (1995). A second generation force field for the simulation of proteins, nucleic acids, and organic molecules. *JOurnal of the American Chemical Society*, **117**, 5179–5197.

Dang, C. & Xu, L. (2000). Barrier function method for the nonconvex quadratic programming problem with box constraints. *Journal of Global Optimization*, **18**, 165–188.

Das, B., Meirovitch, H. & Navon, I.M. (2003). Performance of hybrid methods for large-scale unconstrained optimization as applied to models of proteins. *Das, B., Meirovitch, H., Navon, I. M.,*, **24**, 1222–1231.

de Sancho, D. & Rey, A. (2008). Energy minimizations with a combination of two knowledge-based potentials for protein folding. *Journal of Computational Chemistry*, **29**, 1684–1692.

Derrida, B. (1980). Random energy model: Limit of a family of disordered models. *Physical Review Letters*, **45**, 79–82.

Dixon, L.C.W. (1990). On finding the global minimum of a function of one variable. Technical Report No. 236, Numerical Optimisation Centre, Hatfield Polytechnic, UK.

Dixon, L.C.W. & Szegö, G.P. (1975). *Towards Global Optimization*. Elsevier Science, North Holland, Amsterdam.

Doyle, M. (2003). *A Barrier Algorithm for Large Nonlinear Optimization Problems*. Ph.D. thesis, Stanford University, Stanford, CA, USA.

Duan, Y. & Kollman, P. (1998). Pathways to a protein folding intermediate observed in a 1-microsecond simulation in aqueous solution. *Science*, **282**, 740–744.

Eyrich, V.A., Standley, D.M., Anthony K, F. & Friesner, R.A. (1999). Protein tertiary structure prediction using a branch and bound algorithm. *Proteins: Structure, Function, and Genetics*, **35**, 41–57.

Fiacco, A.V. & McCormick, G.P. (1968). *Nonlinear Programming: Sequential Unconstrained Minimization Techniques*. John Wiley & Sons, New York.

Floudas, C., Fung, H.K., McAllister, S.R., Mönnigmann, M. & Rajgaria, R. (2006). Advances in protein structure prediction and de novo protein design: A review. *Chemical Engineering Science*, **61**, 966–988.

Floudas, C.A. (2000). *Deterministic Global Optimization: Theory, Methods and Applications*, vol. 37 of *Nonconvex Optimization and its Applications*. Kluwer Academic Publishers, The Netherlands.

Floudas, C.A. (2007). Computational methods in protein structure prediction. *Biotechnology and Bioengineering*, **97**, 207–213.

Floudas, C.A., Pardalos, P.M., Adjiman, C.S., Esposito, W.R., Gümüs, Z.H., Harding, S.T., Klepeis, J.L., Meyer, C.A. & Schweiger, C.A. (1999). *Handbook of test problems in local and global optimization*. Kluwer Academic Publishers.

Gockenbach, M.S., Kearsley, A.J. & Symes, W.W. (1997). An infeasible point method for minimizing the lennard-jones potential. *Computational Optimization and Applications*, **8**, 273–286.

Goldberg, D. (1989). *Genetic Algorithms in Search, Optimization and Machine Learning*. Addison-Wesley.

Goldstein, A.A. & Price, J.F. (1971). On descent from local minima. *Mathematics of Computation*, **25**, 569–574.

Greer, J. (1981). Comparitive model-building of the mammallian serine proteases. *Journal of Molecular Biology*, **153**, 1027–1042.

Guex, N. & Peitsch, M.C. (1997). Swiss-model and the swiss-pdbviewer: An environment for comparitive protein modelling. *Electrophoresis*, **18**, 2714–2723.

Guvench, O. & MacKerell, A.D. (2008). Automated conformational energy fitting for force-field development. *Journal of Molecular Modeling*, **14**, 667–679.

Havel, T.F. & Snow, M.E. (1991). A new method for building protein conformations from sequence alignments with homologues of known structure. *Journal of Molecular Biology*, **217**, 1–7.

Hoffmann, F. & Strodel, B. (2013). Protein structure prediction using global optimization by basin-hopping with nmr shift restraints. *JOURNAL OF CHEMICAL PHYSICS*, **138**.

Holland, J. (1973). Genetic algorithm and the optimal allocation of trials. *SIAM Journal of Computing*, **2**, 88–105.

Huber, G.A. & McCammon, J.A. (1997). Weighted-ensemble simulated annealing: Faster optimization on hierarchical energy surfaces. *Physical Review E*, **55**, 4822–4825.

John, B. & Sali, A. (2003). Comparative protein structure modeling by iterative alignment, model building and model assessment. *Nucleic Acids Research*, **31**, 3982–3992.

Jones, D.T. (1999). Genthreader: An efficient and reliable protein fold recognition method for genomic sequences. *Journal of Molecular Biology*, **287**, 797–815.

Jones, D.T., Taylort, W.R. & Thornton, J.M. (1992). A new approach to protein fold recognition. *Nature*, **358**, 86–89.

Jurasek, L., Olafson, R.W., Jhonson, P. & L.B.Smillie (1976). Proteolysis and physiological regulation. In D. Ribbons & K. Brew, eds., *Proceedings of the Miami Winter Symposia*, vol. 11, 93–123, Academic Press, New York.

Karplus, K., Barret, C. & Hughey, R. (1998). Hidden markov models for detecting remote protein homologies. *Bioinformatics*, **14**, 846–856.

Kelley, L., MacCallum, R. & Sternberg, M. (2000). Enhanced genome annotation using structural profiles in the program 3D-PSSM. *Journal of Molecular Biology*, **299**, 499–520.

Khimasia, M.M. & Coveney, P.V. (1997). Protein structure prediction as a hard optimization problem: The genetic algorithm approach. *Molecular Simulation*, **19**, 205–226.

Kim, D., Xu, D., Guo, J., Ellrott, K. & Xu, Y. (2003). PROSPECT II: protein structure prediction program for genome-scale applications. *Protein Engineering*, **16**, 641–650.

Kirkpatrick, S., Gelatt, C. & Vecchi, M. (1983). Optimization by simulated annealing. *Science*, **220**, 671–680.

Klepeis, J.L. & Floudas, C.A. (1999). Free energy calculations for peptides via deterministic global optimization. *Journal of Chemical Physics*, **110**, 7491–7512.

Klepeis, J.L., Nguyen, X. & Floudas, C.A. (1997). *GLO-FOLD: A package for global optimization using alphaBB in protein folding*. Ph.D. thesis, Princeton University, Princeton, NJ.

Kolinski, A. & Skolnick, J. (1994). Monte carlo simulations of protein folding. I. lattice model and interaction scheme. *Proteins: Structure, Functions, and Genetics*, **18**, 338–352.

Kondov, I. (2013). Protein structure prediction using distributed parallel particle swarm optimization. *Natural Computing*, **12**, 29–41.

Lathrop, R. & Smith, T. (1994). A branch-and boundalgorithm for optimal protein threading with pairwise (contact potential) amino acid interactions. In L. Hunter & B. Shriver, eds., *Proceedings of 27th Hawaii International Conference on System Sciences*, vol. 5, 365–374, IEEE Computer Society Press.

Lathrop, R., Rogers, R., Smith, T. & White, J. (1998). A bayes-optimal probability theory that unifies protein sequence-structure recognition and alignment. *Bulletin of Mathematical Biology*, **60**, 1039–1071.

Lathrop, R.H. (1994). The protein threading problem with sequence amino acid interaction preferences is NP-complete. *Protein Engineering*, **7**, 1059–1068.

Lathrop, R.H. & Smith, T.F. (1996). Global optimum protein threading with gapped alignment and empirical pair score functions. *Journal of Molecular Biology*, **255**, 641–665.

Laughton, C.A. (1994). Prediction of protein side-chain conformations from local three-dimensional homology relationships. *Journal of Molecular Biology*, **235**, 1088–1097.

Lim, K.F., Beliakov, G. & Batten, L.M. (2003). Predicting molecular structures: An application of the cutting angle method. *Physical Chemistry Chemical Physics*, **5**, 3884–3890.

Lin, G., Xu, D., Chen, Z.Z., Jiang, T., Wen, J. & Xu, Y. (2002). An efficient branch-and-bound algorithm for the assignment of protein backbone nmr peaks. In *Bioinformatics Conference, 2002. Proceedings*, 165–174, IEEE Computer Society.

Liu, Y. & Beveridge, D.L. (2002). Exploratory studies of ab initio protein structure prediction: Multiple copy simulated annealing, amber energy functions, and a generalized born/solvent accessibility solvation model. *PROTEINS: Structure, Function, and Genetics*, **46**, 128–146.

Liu, Y.L. & Tao, L. (2006). An improved parallel simulated annealing algorithm used for protein structure prediction. In *Proceedings of 2006 International Conference on Machine Learning and Cybernetics*, 2335–2338, Dalian, China.

Mackerell, A.D., Bashford, D., M.Bellott, Dunbrack, R.L., Evanseck, J.D., Yin, D. & Karpulus, M. (1998). All-atom empirical

potential for molecular modeling and dynamics studies of proteins. *Journal of Physical Chemistry B*, **102**, 3586–3616.

Mackerell *et al.*, A.D. (1998). All-atom empirical potential for molecular modeling and dynamics studies of proteins. *Journal of Physical Chemistry B*, **102**, 3586–3616.

Maiorov, V. & Crippen, G. (1992). Contact potential that recognizes the correct folding of globular proteis. *Jornal of Molecular Biology*, **227**, 876–888.

Maranas, C.D. & Floudas, C.A. (1992). A global optimization approach for lennard-jones microclusters. *Journal of Chemical Physics*, **97**, 7667–7678.

Maranas, C.D. & Floudas, C.A. (1994a). A deterministic global optimization approach for molecular structure determination. *Journal of Chemical Physics*, **100**, 1247–1261.

Maranas, C.D. & Floudas, C.A. (1994b). Global minimum potential energy conformations of small molecules. *Journal of Global Optimization*, **4**, 135–170.

Maranas, C.D., Androulakis, I.P. & Floudas, C.A. (1996). A deterministic global optimization approach for the protein folding problem. In P.M. Pardalos, D. Shalloway & G. Xue, eds., *Global Minimization of Nonconvex Energy Functions: Molecular Conformation and Protein Folding*, vol. 23, 133–150, American Mathemetical Society.

McAllister, S.R. & Floudas, C.A. (2010). An improved hybrid global optimization method for protein tertiary structure prediction. *Computational Optimization and Applications*, **45**, 377–413.

MELLER, J., WAGNER, M. & ELBER, R. (2002). Maximum feasibility guideline in the design and analysis of protein folding potentials. *Journal of Computational Chemistry*, **23**, 1–8.

Melo, M.C.R., Bernardi, R.C., Fernandes, T.V.A. & Pascutti, P.G. (2012). Gsafold: A new application of gsa to protein structure prediction. *Proteins: Structure, Function, and Bioinformatics*, **80**, 2305–2310.

Meza, J. & Martinez, M. (1994). Direct search methods for the molecular conformation problem. *Journal of Computational Chemistry*, **15**, 627–632.

Mirzaei, H., Beglov, D., Paschalidis, I.C., Vajda, S., Vakili, P. & Kozakov, D. (2012). Rigid body energy minimization on manifolds for molecular docking. *Journal of Chemical Theory and Computation*, **8**, 4374 – 4380.

Moloi, N.P. & Ali, M.M. (2005). An iterative global optimization algorithm for potential energy minimization. *Computational Optimization and Applications*, **30**, 119–132.

Momany, F., McGuire, R., Burgess, A. & Scheraga, H. (1975). Energy parameters in polypeptides vii. geometric parameters, partial atomic charges, nonbonded interactions, hydrogen bond interactions, and intrinsic torsional potentials for the naturally occurring amino acids. *The Journal of Physical Chemistry*, **79**, 2361–2381.

Moore, R.E. (1979). *Methods and applications of interval analysis*. SIAM, Philadelphia.

Murray, W. & Ng, K.M. (2008). An algorithm for nonlinear optimization problems with binary variables. *Computational Optimization and Applications*, **47**, 257–288.

National Institute of Health (1999). Uniform Resource Locators (URL). Tech. rep., National Institute of General Medical Sciences, `http://www.nigms.nih.gov/`; accessed July, 2006.

Needleman, S.B. & Wunsch, C.D. (1970). A general method applicable to the search for similarities in the amino acid sequence of two proteins. *Journal of Molecular Biology*, **48**, 443–453.

Nemethy, G., Gibson, K.D., Palmer, K.A., Yoon, C.N., Paterlini, G., Zagari, A., Rumsey, S. & Scheraga, H.A. (1992). Energy parameters in polypeptldes. 10. improved geometrical parameters and nonbonded interactions for use in the ecepp/3 algorithm, with appllcatlon to proline-containing peptides. *The Journal of Physical Chemistry*, **96**, 6472–6484.

Neumaier, A. (1997). Molecular modeling of proteins and mathematical prediction of protein structure. *SIAM Review*, **39**, 407–460.

Ng, K.M., Solayappan, M. & Poh, K.L. (2011). Global energy minimization of alanine dipeptide via barrier function methods. *Computational Biology and Chemistry*, **35**, 19–23.

Notredame, C. (2002). Recent progress in multiple sequence alignment: A survey. *Pharmacogenomics*, **3**, 131–144.

Pardalos, P., Shalloway, D. & G.Xue (1994). Optimization methods for computing global minima of nonconvex potential energy functions. *Jornal of Global Optimization*, **4**, 117–133.

Pardalos, P.M. (1991). Construction of test problems in quadratic bivalent programming. *ACM Transactions on Mathematical Software*, **17**, 74–87.

Pedersen, J.T. & Moult, J. (1995). Ab initio structure prediction for small polypeptides and protein fragments using genetic algorithms. *PROTEINS: Structure, Function, and Genetics*, **23**, 454–460.

Peng, J. & Xu, J. (2010). Low-homology protein threading. *Bioinformatics*, **26**, i294–i300.

Peng, J. & Xu, J. (2011). A multiple-template approach to protein threading. *Proteins-Structure Function and Bioinformatics*, **79**, 1930–1939.

Pétrowski, D. & Taillard, S. (2006). *Metaheuristics for Hard Optimization*. Springer-Verlag, Germany.

Pintér, J.D. (1996). *Global Optimization in Action*, vol. 6 of *Nonconvex Optimization and its Applications*. Kluwer Academic Publishers, The Netherlands.

Ponder, J.W. (2004). Uniform Resource Locators (URL). Tinker version 4.2, Washington University School of Medicine, `http://dasher.wustl.edu/tinker/`; accessed July, 2006.

Ponder, J.W. & Case, D.A. (2003). Force fields for protein simulations. *Advances in Protein Chemistry*, **66**, 27–85.

Rodrigues, J.P.G.L.M., Levitt, M. & Chopra, G. (2012). Kobamin: a knowledge-based minimization web server for protein structure refinement. *Nucleic Acids Research*, **40**, W323–W328.

Rohl, C.A., Strauss, C.E.M., Chivian, D. & Baker, D. (2004). Modeling structurally variable regions in homologous proteins with rosetta. *Proteins: Structure, Function, and Bioinformatics*, **55**, 656–677.

Rohl *et al.*, C.A. (2004). Modeling structurally variable regions in homologous proteins with rosetta. *Proteins: Structure, Function, and Bioinformatics*, **55**, 656–677.

Schneider, T.R. (2002). A genetic algorithm for the identification of conformationally invariant regions in protein molecules. *Acta Crystallographica*, **D58**, 195–208.

Scott, W., Hunenberger, P., Trioni, I., Mark, A.E., Billeter, S., Fennen, J., Torda, A., Huber, T., Kruger, P. & VanGunsteren, W. (1997). The GROMOS biomolecular simulation program package. *Journal of Physical Chemistry A*, **103**, 3596–3607.

Son *et al.*, W.J. (2012). Simulated q-annealing: conformational search with an effective potential. *Journal of Molecular Modeling*, **18**, 213–220.

Sun, S. (1993). Reduced representation model of protein structure prediction:statistical potential and genetic algorithms. *Protein Science*, **2**, 762–785.

Swindells, M.B. & Thornton, J.M. (1991). Modelling by homology. *Current Opinion in Structural Biology*, **1**, 219–223.

Tanka, S. & Scheraga, H. (1976). Medium- and long-range interaction parameters between amino acids for predicting three-dimensional structures of proteins. *Macromolecules*, **9**, 945–950.

Thompson, H.B. (1967). Calculation of cartesian coordinates and their derivatives froom internal molecular coordinates. *The Journal of Chemical Physics*, **47**, 3407–3410.

Tramontano, A. & Morea, V. (2003). Assessment of homology based predictions in CASP5. *Proteins: Structure, Function, and Bioinformatics*, **53**, 352–368.

Tuffrey, P., Etchebest, S., Hazout, S. & Levery, R. (1991). A new approach to the rapid determination of protein side chain conformations. *Journal of Biomolecular Structure and Dynamics*, **8**, 1267–1289.

Tyka, M.D., Jung, K. & Baker, D.D. (2012). Efficient sampling of protein conformational space using fast loop building and batch minimization on highly parallel computers. *Journal of Computational Chemistry*, **33**, 2483–2491.

Šali, A. & Blundell, T.L. (1993). Comparative protein modelling by satisfaction of spatial restraints. *Journal of Molecular Biology*, **234**, 779–815.

Wagner, M., Meller, J. & Elber, R. (2004). Large-scale linear programming techniques for the design of protein folding potentials. *Mathemetical Programming*, **101**, 301–318.

Wider, G. (2000). Structure determination of biological macromolecules in solution using NMR spectroscopy. *BioTechniques*, **29**, 1278–1294.

Wilkinson, J.H. (1963). *Rounding errors in algebraic processes*. Prentice Hall, Engelwood Cliffs, NJ.

Wilson, S.R. & Cui, W. (1988). Conformational analysis of flexible molecules: Location of the global minimum energy conformation by the simulated annealing method. *Tetrahedron Letters*, **29**, 4373–4376.

Wilson, S.R. & Cui, W. (1990). Applications of simulated annealing to peptides. *Biopolymers*, **29**, 225–235.

Wingo, D. (1985). Globally minimizing polynomials without evaluating derivatives. *International Journal of Computer Mathematics*, **17**, 287–294.

Xu, J., Li, M., Kim, D. & Xu, Y. (2003). RAPTOR: Optimal protein threading by linear programming. *Journal of Bioinformatics and Computational Biology*, **1**, 95–117.

Xu, J., Li, M. & Xu, Y. (2004). Protein threading by linear programming: Theoretical analysis and computational results. *Journal of Combinatorial Optimization*, **8**, 403–418.

Xu, Y. & Xu, D. (2000). Protein threading using PROSPECT: design and evolution. *Proteins: Structure, Function and Bioinformatics*, **40**, 343–354.

Xu, Y., Xu, D. & Uberbacher, E.C. (1998). An efficient computational method for globally optimal threadings. *Journal of Computational Biology*, **5**, 597–614.

Ye, Y. (1997). *Interior Point Algorithms: Theory and Analysis*. Wiley-Interscience Series in Discrete Mathematics Optimization, John Wiley and Sons, New York.

Zhang, J. (2011). A brief review on results and computational algorithms for minimizing the lennard-jones potential. In *CoRR*, arXiv:1101.0039.

Zhang, Y. (2008). Progress and challenges in protein structure prediction. current opinion in structural biology. **18**, 342–348.