Differential Evolution Explores a Multiobjective Knowledge-based Energy Function for Protein Structure Prediction

by

Xingqian Chen

A dissertation

submitted to the Graduate School of Science and Engineering

in Partial Fulfillment of the Requirements

for the Degree of

Doctor of Engineering

University of Toyama

Gofuku 3190, Toyama-shi, Toyama 930-8555 Japan

2021

(Submitted October 12, 2021)

Acknowledgements

I would like to express my sincere thanks to the people who gave me with useful and helpful assistance during my study and research. Without their care, consideration, and constructive suggestions, This article will most likely not be completed.

First, I would like to express my sincere thanks to Professor Zheng Tang for his support and continuous encouragement, who introduced me to the important and fascinating world of computational intelligence and artificial intelligence. Without his helpful assistance and continuous encouragement, I will never complete this degree. I would also like to express my sincere thanks to Associate Professor Shangce Gao, who introduced me to the research field of bioinformatics, for his great support and extensive experience.

Last but not the least, I would like to express my sincere thanks to all the members of the Intelligent Information Systems Research Lab in University of Toyama, for all their help and friendship that made this time much more interesting. In addition, I would like to thank all the members of my family, for their unconditional love, support, and encouragement through this study process.

Abstract

How to predict the 3D structure of a given protein starting only from its amino acid sequenceis called the protein structure prediction (PSP) problem. Despite the rapid development of computer techniques and the unremitting efforts of researchers, the PSP problem remains challenging in bioinformatics and computational biology. The two pivotal factors of a successful free modeling prediction approach are an efficient search strategy and an effective energy function.

In my research of defending PhD, I try to model the PSP problem as a multiobjective optimization problem and use an differential evolution search strategy to solve the problem. In details, the PSP problem is modeled as a multiobjective optimization problem, and a free modeling approach called MODE-K is proposed to solve this problem. my efforts center on two aspects. First, a knowledge-based energy function called RWplus is used as the evaluation criterion. This function is decomposed into two terms: an orientation-dependent energy term and a distance-dependent energy term. Second, a multiobjective differential evolution coupled with an external archive employed to perform conformation space searching. After conformation space searching, we introduce a cluster method to select the final predicted structure from series of decoy structures. The performance of the method was verified with eighteen test proteins. The experimental results demonstrate the effectiveness of the proposed method and indicate that incorporating knowledge-based energy functions into multiobjective approaches to solve the PSP problem is promising.

The contribution of this thesis is fourfold: first, the PSP problem is modeled as a multiobjective optimization problem and two knowledge-based energy terms are used to construct the energy function. Second, a new MODE algorithm that interacts

with an external archive is proposed. Third, an integral work flow is provided. The clustering method which called MUFOLD-CL is used to identify the final predicted structure from a set of decoy structures that are stored in the archive. Fourth, eighteen test proteins categorized into three structural classes are used to evaluate the proposed method. More investigation of the experimental results provides evidence of the superior performance of the proposed approach.

The remainder of this thesis is organized as follows. Chapter 1 presents the introduction of the PSP problem. Chapter 2 presents the related works that are based on evolutionary algorithms for solving the PSP problem. Chapter 3 presents three important concepts, i.e., protein structure, canonical differential evolution, and multiobjective optimization. Chapter 4 presents the details of the proposed approach. Chapter 5 presents the experimental results and discussion. Finally, the conclusion of this study is drawn in Chapter 6.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Proteins are complex large organic macromolecules that are composed of one or more chains of 20 different amino acids in specific orders. Many fundamental biological functions in organisms are performed by proteins, such as structural support, material transport, and regulation functions. Since the structure of a protein determines its biological functions [1], knowledge of its native structures is essential for understanding its role in life activities [2]. Three experimental methods, i.e., X-ray crystallography, NMR spectroscopy, and cryo-electron microscopy, are commonly used to perform protein structure determination. However, all of these experimental methods are costly and waste much of time [3, 4]. On the other hand, since the three-dimensional (3D) structure of a protein is determined by its amino acid sequence, it is very meaningful for researchers to obtain the three-dimensional structure of the protein from its sequence by calculation methods [5].

The problem, i.e., predicting the 3D structure of a given protein starting only from its amino acid sequence, is called the protein structure prediction problem. This problem was first proposed by Nobel Laureate Anfinsen in the 1970s [6]. The high theoretical value and practical significance make the research of this problem necessary and promising. However, despite the unremitting efforts of researchers over the past half century, this problem remains challenging in computational biology and bioinformatics [5].

Numerous approaches have been proposed to solve the PSP problem [7]. These approaches can be roughly grouped into two categories: template-based modeling

(TBM) and free modeling (FM). TBM is based on the assumption that similar sequences lead to similar structures. If the similar structures of a target protein are identified in the PDB [8], the target model can be constructed according to these "templates". At present, I-TASSER [9] and SwissModel [10] are considered the most successful methods among the TBM approaches. Although TBMs can yield prediction models that have a higher accuracy than FMs, they seem to give us little insight into the principle of protein folding. In contrast, if there is no similar structure in the PDB, then use FM. In fact, successful FM approaches can help us investigate the intrinsic mechanism of proteins. Moreover, they rely little on a priori knowledge. The most successful FM approaches are those based on fragment assembly, such as Rosetta [11] and QUARK [12]. In recent years, the incorporation of deep learning technology into FM has attracted the interest of many researchers [13, 14].

All FM approaches follow the thermodynamic hypothesis that a native protein structure remains stable at the lowest Gibbs free energy [6]. This hypothesis indicates the basic paradigm for solving the PSP problem: computing the free energy of every possible conformation and identifying the one with the lowest free energy as the final result. Therefore, as suggested in [15], the two key factors of a successful FM approach are an effective energy function and an efficient search strategy.

Protein energy functions are used to select more native-like conformations during the process of protein folding. The existing protein energy functions can be roughly classified into two groups [16]: physics-based energy functions (PBEFs) and knowledge-based energy functions (KBEFs). The PBEFs, such as CHARMM [17] and OPLS [18], are based on the analysis of the forces between the particles and are derived from the laws of physics. The KBEFs, e.g., dDFIRE [19], GOAP [20], and OPUS-DOSP [16], refer to the statistical energy, which are derived from the statistics of known 3D protein structures. The KBEFs can be further divided into three categories: contact potentials [21], distance-dependent potentials [22], and orientation-dependent potentials [23]. In general, it is well accepted that KBEFs are more effective and more accurate than PBEFs in the field of PSP [24, 19, 25, 16].

Since the conformation search space is very large, an exhaustive search strategy

is infeasible under normal circumstances. A successful FM must employee efficient search strategies to find the global minimum of a given energy function. The most common conformation search method employed in FMs is the Monte Carlo (MC) algorithm or its variations [26, 12]. However, MC sampling seems to fall easily into a local minimum of a rugged energy function [27, 28]. Recently, employing evolutionary computation techniques as the search strategy in FMs has attracted researchers' interest [27, 29, 30, 31, 4], and considerable success has been achieved. These works suggest that the strong search capacity of evolutionary algorithms (EAs) [32, 33, 34, 35] for solving a large set of real-world problems [36, 37, 38, 39, 40, 41, 42]. Specifically, the differential evolution (DE) algorithm [43] has emerged as one of the most powerful algorithms in the field of evolutionary computation [44]. The variants of DE have outperformed other optimization algorithms in almost all the CEC competitions [45]. Recent research has shown its efficiency for diverse optimization problems [46, 47, 48, 49, 50, 51, 52]. These works drive us to use the most advanced optimization technique to address the PSP problem.

In recent years, modeling the PSP problem as a multiobjective optimization problem (MOOP) provides a new direction for solving it [53, 54, 55, 56, 57, 58, 59]. Not only the energy function but also the search scheme in multiobjective approaches is different from that of the single-objective approaches. In multiobjective approaches, the solution path is not fixed, and more adaptability is provided. More importantly, the multiobjective approaches seem to be capable of finding more fruitful solutions around the bottom of the funnel-shaped energy landscape [44, 60], owing to the Pareto dominance concept. These works based on multiobjective optimization have intensifiled the advantage of modeling the PSP problem as a MOOP. However, too much attention is given to the PBEFs in these works although KBEFs are considered more accurate and more effective than PBEFs [24, 25]. As far as the author knows, adopting a pure KBEF in a multiobjective approach to address the PSP problem has not been well explored.

In this study, an FM approach called MODE-K is proposed to solve the PSP problem. This approach follows the framework of common FM approaches. We use

the KBEF RWplus [25] as the energy function. Considering the distance-dependent potentials and the orientation-dependent potentials describe the different interactions of a protein conformation, we decompose RWplus into two terms as the multiobjective function: a distance-dependent energy term and an orientation-dependent term. Moreover, a multiobjective differential evolution (MODE) algorithm is employed as the search strategy to explore the conformation space. The MODE algorithm is coupled with an archive to maintain the optimal solutions. The external archive is based on nondominated sorting and is capable of properly addressing slightly worse solutions. The contribution of this paper is fourfold: first, we model the PSP problem as a MOOP and use two knowledge-based energy terms to construct the energy function. Second, a new MODE algorithm that interacts with an external archive is proposed. Third, an integral work flow is provided. We use the clustering method MUFOLD-CL to distinguish the final predicted structure from a series of decoy structures that are stored in the archive. Fourth, eighteen test proteins categorized into three structural classes are used to evaluate the proposed approach. More investigation of the experimental results provides evidence of the superior performance of the proposed approach.

Chapter 2

Related works

Most FM approaches treat the PSP problem as a single-objective optimization problem (SOOP), in which a conformational search is executed under the guidance of a single-objective energy function [61, 30, 62, 4]. For example, Borguesan et al. proposed an angle probability list knowledge-based prediction method called GA-APL to solve the PSP problem [61], where the search strategy is a single-objective genetic algorithm (GA) and the energy function is the Rosetta scoring function. Correa et al. employed the Rosetta scoring function as the energy function and presented a memetic algorithm to solve the PSP problem [30]. In additon, Zhang et al. proposed an improved single-objective DE called SCDE for solving the PSP problem [62]. The contact of amino acid residue and secondary structure prediction message are used to construct the energy function in their work. In [4], Zhou et al. proposed a cooperative DE and applied it to PSP. The Rosetta score3 energy function was employed to evaluate conformation in their work.

Since the multiobjective optimization approaches give new perspectives for solving the PSP problem, numerous approaches based on multiobjective EA have been proposed in recent years. For example, Cutello et al. proposed a multiobjective evolutionary strategy called I-PAES [63] to solve the PSP problem. The CHARMM27 energy function was splited into bond and nonbond energy terms as the fitness function in I-PAES. A similar idea was adopted in ADEMO/D [54] where the search strategy is an adaptive DE algorithm. In addition, Gao et al. proposed a multiobjective EA called MO3 [56] and its variant AIMOES [57] for solving the PSP problem.

In their works, coupled with CHARMM22 energy function, solvent terms were incorporated as the third objective to implicitly reflect the effect of solvent. Moreover, Song et al. combined CHARMM22 and dDFIRE as the compound multiobjective energy function and used a multiobjective particle swarm optimization (MOPSO) to solve the PSP problem [58]. In [55], Rocha et al. proposed a multiobjective GA, which insert co-evolution information from contact maps to solve the PSP problem. In [59], Zaman and Shehu proposed a memetic EA called Evo-Diverse to control decoy diversity in conformation sampling. This approach served as a complementary approach of Rosetta, and the Rosetta *score*4 energy function was decomposed into three terms as the evaluation criterion. These works have shown the advantage of modeling the PSP problem as a MOOP. However, adopting a pure KBEF in a multiobjective approach to address the PSP problem has not been well explored in these works, though KBEFs are considered more accurate and more effective than PBEFs [24, 25].

Chapter 3

Materials

3.1 Protein structures and their representations

The structures of proteins are very complex. Briefly, protein structures are distinguished into four levels: primary structure, secondary structure, tertiary structure, and quaternary structure. The protein primary structure is the linear sequence of amino acids, which are held together by peptide bonds. Twenty categories of amino acids compose natural proteins. Usually, the sequence of a protein is reported as a character string. The secondary structures of a protein are the local segments held together by hydrogen bonds [64]. Two common and important types of secondary structures are α-helices and β-sheets [65]. Tertiary structure is also called the 3D structure of a protein. Normally, a native protein has a unique 3D structure in a cellular environment. In addition, the function of a protein is determined by its 3D structure. Several tertiary structures consist of a quaternary structure, and they work as a single functional unit.

The digitization to construct a protein is the first step to solve the PSP problem. The representation of full-atom torsion angles is used in this study, and all kinds of atoms in a protein are considered. The torsion angles of a protein consist of back-bond torsion angles ϕ (N-C_α bond), ψ (C_α-C bond), and ω (C-N bond), and side-chain torsion angles χ_i ($i \in \{0, 1, 2, 3, 4\}$). We use Table 3.1 to represent the numbers of the side-chain torsion angles. These angles are all limited from -180° to 180°. In addition, the bond length and bond angles of the conformation of a protein have been

Table 3.1: Number of χ_i angles in each amino acid.

Amino acid	number of χ_i angles
ALA, GLY, PRO	0
CYS, SER, THR, VAL	1
ASN, ASP, HIS, ILE, LEU, PHE, TRP, TYR	2
MET, GLN, GLU	3
ARG, LYS	4

set to ideal value. In this way, the number of degrees of freedom is reduced greatly when compared with the representation of Cartesian space. Sequentially, all torsion angles constitute a vector, which can uniquely determine a protein conformation, as shown in Fig. 3.1. Since the energy calculation of a protein is based on its 3D structure, the representation of the torsion angles of the protein is decoded as its representation of Cartesian space in the evaluation process. It is worth noting that all changes in a protein conformation occur in the torsion angle space during the process of a conformation space search.

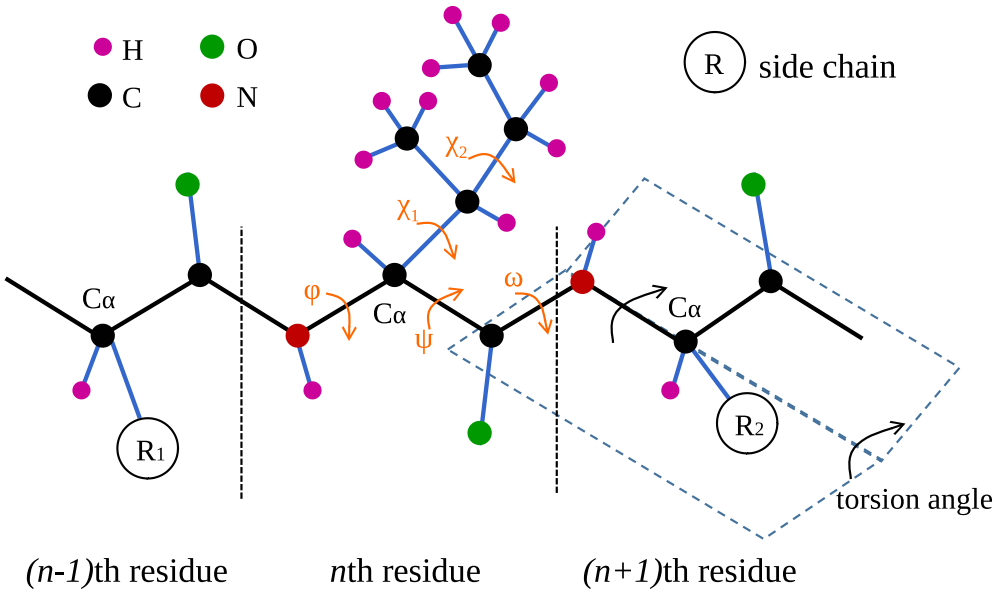


Figure 3.1: The torsion angles in nth residue of a protein. In addition, this example shows that a torsion angle is the angle between two planes and is determined by four atoms.

3.2 Canonical DE

Inspired by the natural behavior [66, 67], computational intelligence has shown its powerful capabilities for solving many practical problems [68, 69, 70, 71, 72, 73, 74]. The evolutionary computation is a promising technique in computational intelligence. The differential evolution (DE) algorithm [43] is a very powerful algorithm in the field of evolutionary computation [44]. Intrinsically, it is a stochastic optimization algorithms [44]. The canonical DE algorithm was first proposed by Storn and Price in the 1990s [43]. DE solves a problem by mutating a set of candidate solutions with scaled differences, which are extracted from the current population. Given a practical problem, a candidate solution is encoded like a vector in DE. DE maintains a population of NP solutions (individuals) and evolves them in the search space. A candidate individual in DE is represented as follows:

$$\mathbf{X}^{(i)(t)} = (x_1^{(i)(t)}, x_2^{(i)(t)}, ..., x_d^{(i)(t)}), \tag{3.1}$$

where d is the number of dimensions of the search space, and $\mathbf{X}^{(i)(t)}$ showes the ith solution in the NP-size population at iteration t. Usually, the search space of a practical problem is restricted within a hypercube. The prescribed minimum and maximum bounds of these individuals are described as $\mathbf{X}_{min} = (x_{min,1}, x_{min,2}, ..., x_{min,d})$ and $\mathbf{X}_{max} = (x_{max,1}, x_{max,2}, ..., x_{max,d})$, respectively.

The regular DE algorithm follows the normal constitution of EAs and makes up of four mainly constitution: initialization, mutation, crossover, and selection. We use Algorithm 1 to show the pseudo-code of the regular DE, and the four main components are described as follows.

Initialization: the population is initialized to cover the search space uniformly as:

$$x_j^{(i)(0)} = x_{min,j} + r_u(x_{max,j} - x_{min,j}), \tag{3.2}$$

where $x_j^{(i)(0)}$ is the jth element of ith individual $\mathbf{X}^{(i)(0)}$. r_u is a unified distributed random number in $[0, 1]$ and has the same meaning in the following context.

Algorithm 1: The pseudo-code of the canonical DE.

```
begin
    /* Initialization.                                              */
    Initialize the population {X^(i)(t) | i ∈ 1, 2, ..., NP}.
    Evaluate all X^(i)(t)s.
    while Stopping criterion is not met. do
        for i in {1, 2, ..., NP} do
            /* Mutation.                                            */
            Create the donor vector V^(i)(t) .
            /* Crossover.                                           */
            Create the trial vector U^(i)(t) .
            Evaluate U^(i)(t).
            /* Selection.                                           */
            X^(i)(t+1) ← Select(X^(i)(t), U^(i)(t)).
    Output result.
```

Mutation: For each individual $\mathbf{X}^{(i)(t)}$ (also called the target vector) in the current population, a corresponding donor vector $\mathbf{V}^{(i)(t)} = (v_1^{(i)(t)}, v_2^{(i)(t)}, ..., v_d^{(i)(t)})$ is produced by a mutation strategy [44]. The most common strategy, "DE/rand/1", is shown as follows:

$$\mathbf{V}^{(i)(t)} = \mathbf{X}^{(r_1)(t)} + F(\mathbf{X}^{(r_2)(t)} - \mathbf{X}^{(r_3)(t)}), \tag{3.3}$$

where F is the scale factor, which commandes the scaled difference. r_1, r_2, and r_3 are three different random integers, ranging in $[1, NP]$. Thus, $\mathbf{X}^{(r_1)(t)}$, $\mathbf{X}^{(r_2)(t)}$, and $\mathbf{X}^{(r_3)(t)}$ are three different vectors randomly selected from the current population.

Crossover: A trial vector $\mathbf{U}^{(i)(t)} = (u_1^{(i)(t)}, u_2^{(i)(t)}, ..., u_d^{(i)(t)})$ is generated from its corresponding target vector $\mathbf{X}^{(i)(t)}$ and donor vector $\mathbf{V}^{(i)(t)}$. Binomial crossover is commonly employed in DE to exchange the components between the target vector and the donor vector. It is expressed as follows:

$$u_j^{(i)(t)} = \begin{cases} v_j^{(i)(t)}, & if\ (r_u \leq Cr\ or\ j = j_r), \\ x_j^{(i)(t)}, & otherwise, \end{cases} \tag{3.4}$$

where $u_j^{(i)(t)}$ is the jth element of $\mathbf{U}^{(i)(t)}$ and r_u is a random number as mentioned

above. Cr is called the crossover velocity. j_r is a random integer in $[1, d]$ and ensures the difference between $\mathbf{U}^{(i)(t)}$ and $\mathbf{X}^{(i)(t)}$.

Selection: The target vector $\mathbf{X}^{(i)(t)}$ competes with the trial vector $\mathbf{U}^{(i)(t)}$ for surviving to the next generation. An individual is updated as follows:

$$\mathbf{X}^{(i)(t+1)} = \begin{cases} \mathbf{U}^{(i)(t)}, & if\ (f(\mathbf{U}^{(i)(t)}) \leq f(\mathbf{X}^{(i)(t)})), \\ \mathbf{X}^{(i)(t)}, & otherwise. \end{cases} \tag{3.5}$$

3.3 Multiobjective optimization

Multiobjective optimization problems arise frequently when solving real-world problems. In contrast to the more general SOOPs, in MOOPs, there exists more than one objective function that need optimization at the same time. These objectives are in conflict in normal cases. Mathematically, a MOOP (for minimization) can be formulated as follows:

$$\begin{aligned} &minimize\ \mathbf{f}(\mathbf{x}) = (f_1(\mathbf{x}), f_2(\mathbf{x}), ..., f_m(\mathbf{x})), \\ &subject\ to\ \mathbf{x} = (x_1, x_2, ..., x_n) \in \Omega, \end{aligned} \tag{3.6}$$

where $\mathbf{x}$ is called the decision vector with n dimensions and Ω is the decision space. $\mathbf{f} : \Omega \rightarrow R^m$ is the objective vector, consisting of m ($m \geq 2$) objective functions: $f_1, f_2, ..., f_m$.

Due to the conflicts among these objectives in MOOPs, the comparison between two feasible solutions refers to the concept of Pareto dominance. Use $\mathbf{a}$ and $\mathbf{b}$ be two feasible solutions in Ω, $\mathbf{a}$ is better than $\mathbf{b}$; that is, $\mathbf{a}$ dominates $\mathbf{b}$ (denoted by $\mathbf{a} \prec \mathbf{b}$ [75]) if

$$\begin{aligned} &1)\ \ \forall i \in \{1, 2, ..., m\},\ f_i(\mathbf{a}) \leq f_i(\mathbf{b})\ \ and \\ &2)\ \ \exists j \in \{1, 2, ..., m\},\ f_j(\mathbf{a}) < f_j(\mathbf{b}). \end{aligned} \tag{3.7}$$

A solution $\mathbf{x}^* \in \Omega$ is the Pareto optimal solution if there exists no solution $\mathbf{x}' \in \Omega$ that dominates $\mathbf{x}^*$. All Pareto optimal solutions make up a Pareto optimal set P, and the image of the Pareto optimal set is called a Pareto front PF [76]. Fig. 3.2

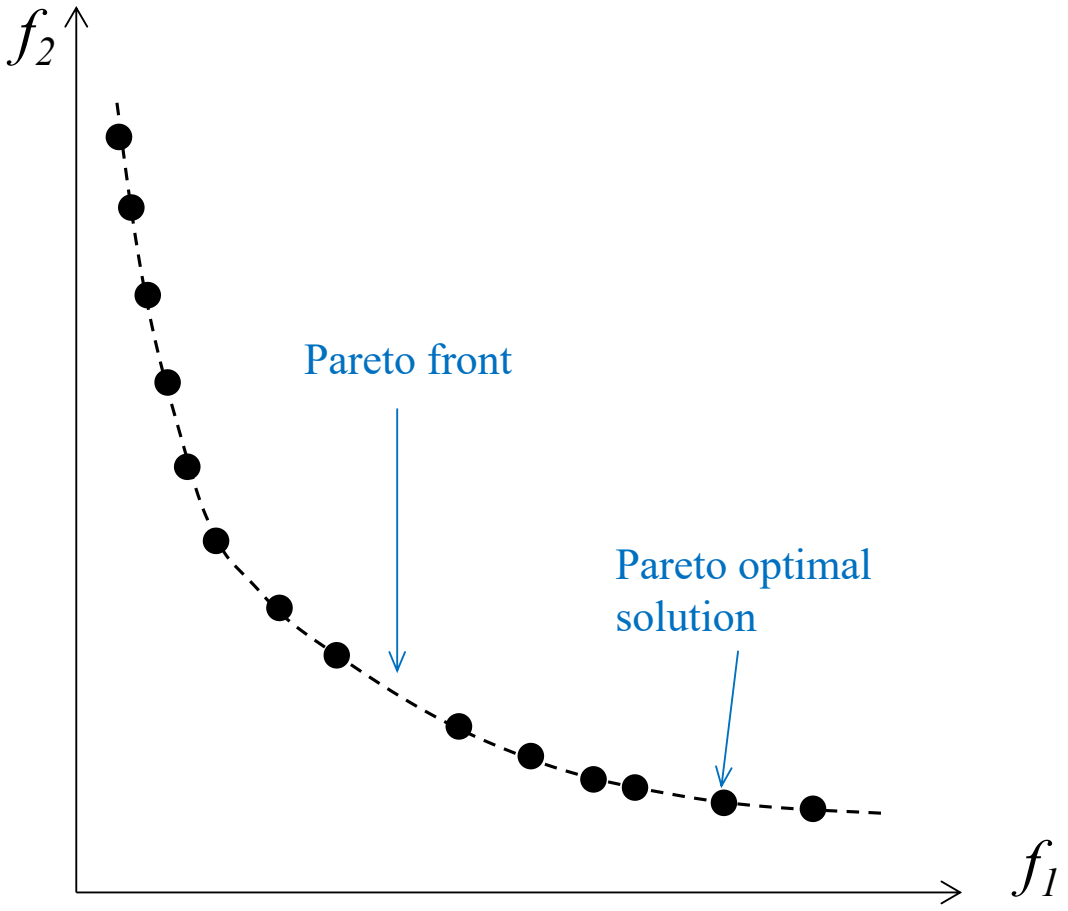


Figure 3.2: An example of the Pareto Front of a two-objective optimization problem.

shows an example of the Pareto Front of a two-objective optimization problem. P and PF are defined as follows:

$$\begin{aligned} P &= \{\mathbf{x}^* \in \Omega \mid \neg\exists\, \mathbf{x}' \in \Omega, \mathbf{x}' \prec \mathbf{x}^*\}, \\ PF &= \{\mathbf{f}(\mathbf{x}) \mid \mathbf{x} \in P\}. \end{aligned} \tag{3.8}$$

In reality, there exist many or infinite Pareto optimal solutions for a practical MOOP, which delegate different trade-offs among the objectives. However, only one or a small number of preferred solutions are eventually selected by the decision maker. Thus, the key issue of an optimization method is how to obtain a perfect approximation of the real Pareto front of a MOOP. Usually, the quality of a approximated

Pareto front to a real one is mainly assessed according to its convergence and diversity [77, 78].

Recently, the adoption of EAs to solve MOOPs has sparked the interest of researchers. These methods based on EAs are classified as multiobjective evolutionary algorithms (MOEAs) [76, 77, 78, 79, 80, 81, 82]. One of the significant differences between MOEAs and the EAs is how to compare two candidate solutions. It is easy to differentiate two candidate solutions in a SOOP because a complete order between them can be obtained by the single-objective fitness function. However, in MOOPs, the Pareto dominance relationship only delimites a partial order because two feasible solutions may be nondominated. Thus, designing an additional strategy to assist in Pareto dominance is a crucial issue in MOEAs [78, 83, 84, 85].

This study follows the widely used two-stage strategy [86] to assign the complete order to series of feasible solutions. Considering the condition of continuous optimization and given a set of feasible solutions, first, every solution will be distributed a rank value by counting the number of solutions that dominates it in the solution set. It is easily concluded that the solutions with smaller ranks are preferred, and the solutions of rank 0 constitute the Pareto optimal set. Then, the solutions with the same rank are further assigned a density value. The common-used density estimation methods are crowding distance [77] and gridding [78]. The solutions with a lower density value (in the lower crowded region) are preferred. Therefore, a feasible solution $\mathbf{a}$ is said to be better than $\mathbf{b}$ if

$$\begin{aligned} &1) rank_{\mathbf{a}} < rank_{\mathbf{b}} \quad or \\ &2) rank_{\mathbf{a}} = rank_{\mathbf{b}} \; and \; density_{\mathbf{a}} < density_{\mathbf{b}}. \end{aligned} \tag{3.9}$$

In addition, the solutions with the same rank are nondominated. This inference is vital for the scheme of nondominated sorting in MOEAs, as shown later in Section 4.3. This inference can be proven briefly through a proof by contradiction as follows.

proof *Step 1*: it is easily concluded that Pareto dominance is a transitive relation from Eq. 3.7, and is expressed as follows:

$$\mathbf{a} \prec \mathbf{b} \; and \; \mathbf{b} \prec \mathbf{c} \quad \Rightarrow \quad \mathbf{a} \prec \mathbf{c}, \tag{3.10}$$

where **a**, **b**, and **c** are three feasible solutions in Ω.

Step 2: considering two feasible solutions $\mathbf{p}, \mathbf{q} \in \Omega$ with the same rank r, if they are not nondominated, one will dominate the other. Without loss of generality, we suppose that **p** dominates **q**. Since $rank_{\mathbf{p}} = \mathrm{r}$, there are r solutions in Ω that dominate **p**. Because Pareto dominance is transitive and $\mathbf{p} \prec \mathbf{q}$, there are at least $(r+1)$ solutions in Ω that dominate **q**. Hence, $rank_{\mathbf{q}} >= r+1$, which leads to a contradiction because $rank_{\mathbf{q}} = r$ in the proposition. Therefore, the solutions with the same rank are nondominated.

Chapter 4

Method

This chapter presents the details of the proposed MODE-K approach. MODE-K follows the general procedure of FM approaches and includes of three main steps. First, some preprocessing is performed after inputing a query protein sequence. Then, the MODE algorithm is used to perform conformational space searching to obtain series of decoy structures. Finally, the decoy selection method MUFOLD-CL is executed to choose the final predicted structure from the set of decoy structures. Fig. 4.1 shows the flowchart of MODE-K. As followed, more details of MODE-K are explained.

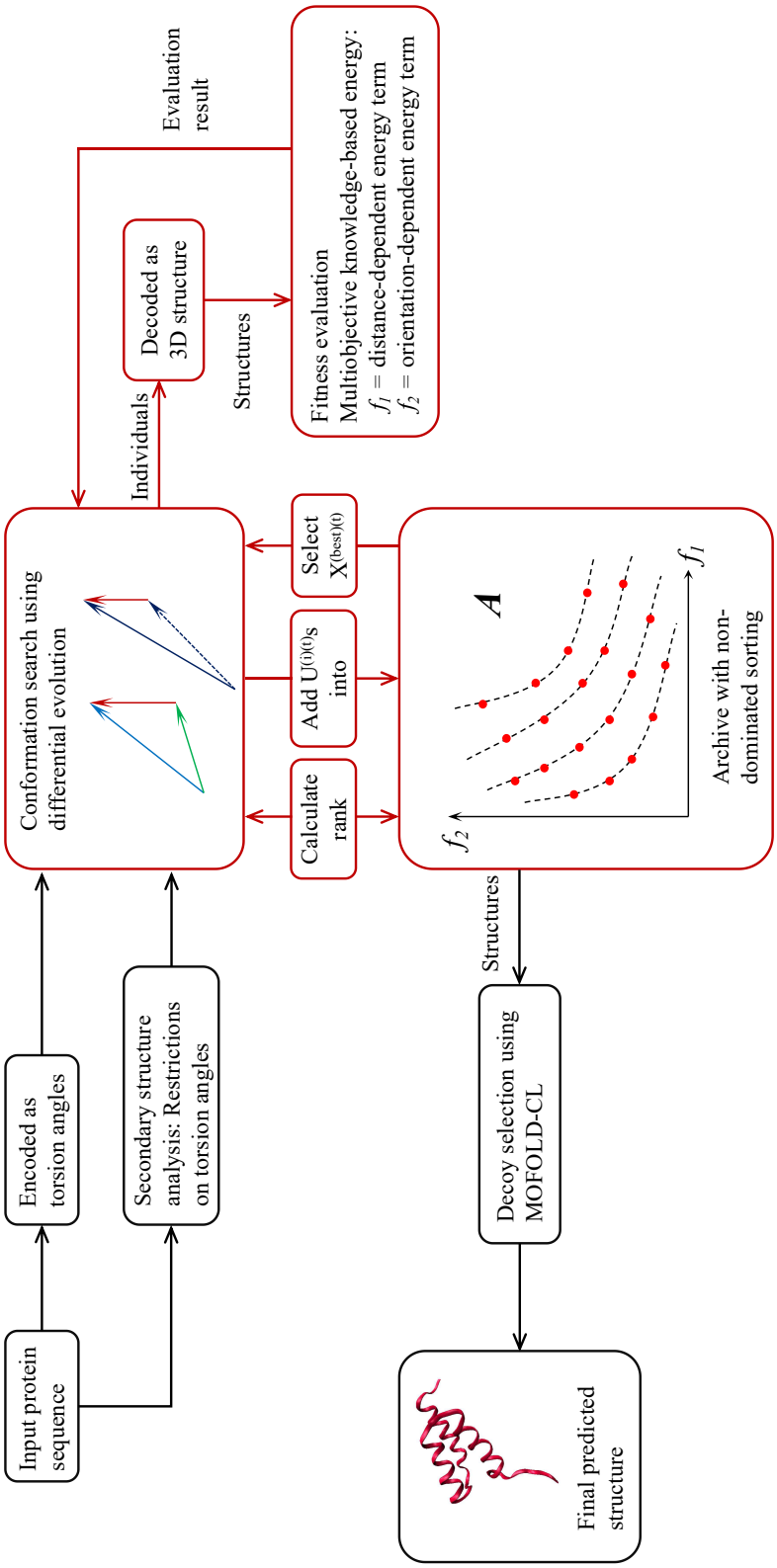


Figure 4.1: The flow of the proposed MODE-K approach for protein structure prediction.

4.1 Protein energy function

In this study, the KBEF RWplus potential [25] is used to assess the conformation of a protein during simulation. This protein energy function consists of two energy terms and has the following form:

$$E_{RWplus} = E_{RW} + wE_{ori}, \tag{4.1}$$

where E_{RW} is a distance-dependent energy term, E_{ori} is an orientation-dependent term. w is a weight constant and is set to 0.1 [25].

The distance-dependent energy E_{RW} is a pair-wise energy term, and the construction of this energy term follows the inverse of Boltzmann's law:

$$\begin{aligned} E_{RW} = \sum_{\alpha,\beta} \overline{u}(\alpha,\beta,R) &= \sum_{\alpha,\beta} -kTln\frac{P_{obs}(\alpha,\beta,R)}{P_{exp}(\alpha,\beta,R)} \\ &\approx \sum_{\alpha,\beta} -kTln\frac{N_{obs}(\alpha,\beta,R)}{N_{exp}(\alpha,\beta,R)}, \end{aligned} \tag{4.2}$$

where k is the Boltzmann constant and T is the temperature. R is the distance between two atoms α and β that are with specific atom types. $P_{obs}(\alpha,\beta,R)$ and $P_{exp}(\alpha,\beta,R)$ are the observed probability and the expected probability, respectively, within a distance shell $R+\Delta R$. $N_{obs}(\alpha,\beta,R)$ and $N_{exp}(\alpha,\beta,R)$ are the observed and the excepted number of atom pairs (α and β) in the same distance shell. The calculation of $N_{obs}(\alpha,\beta,R)$ is similar among most KBEFs [19], and the major difference is the calculation of $N_{exp}(\alpha,\beta,R)$. To calculate $N_{obs}(\alpha,\beta,R)$, the classic KBEFs are based on a noninteracting ideal gas reference state, such as DOPE [87] and DFIRE [88]. In contrast, the RW potential uses a random-walk reference state constructed by a freely jointed chain model [89]. This state reflects and counteracts the inherent chain connectivity effect. As a result, the effectiveness of the RW is improved.

The orientation-dependent energy term E_{ori} is used to specify the side-chain pack-

ing orientation and is expressed as follows:

$$\begin{aligned} E_{ori} &= \sum_{A,B} \delta(A,B)\overline{u}(A,B,O_{AB}) \\ &= \sum_{A,B} -\delta(A,B)kTln\frac{P_{obs}(A,B,O_{AB})}{P_{exp}(A,B,O_{AB})} \\ &\approx \sum_{A,B} -\delta(A,B)kTln\frac{N_{obs}(A,B,O_{AB})}{N_{exp}(A,B,O_{AB})}, \end{aligned} \tag{4.3}$$

where A is a vector pair used to describe the side-chain packing orientation of a residue and B is another vector pair. $\delta(A,B)$ is 1 when the distance between vector pairs A and B is within a given value; otherwise, it is 0. The definitions of $P_{obs}(A,B,O_{AB})$, $P_{exp}(A,B,O_{AB})$, $N_{obs}(A,B,O_{AB})$, and $N_{exp}(A,B,O_{AB})$ are similar to those for the terms in Eq. 4.2.

Since the distance-dependent potentials [22, 88] and the orientation-dependent potentials [23] describe the different types of interactions of a protein conformation and act collectively, combining them into a mixed potential to improve the performance of the potential has peaked researchers' interest [25, 20, 16]. In fact, the orientation-dependent and distance-dependent contributions vary in different ranges of the contact distance [20, 16]. The distance-dependent potentials can capture the feature of local interactions more effectively than the orientation-dependent potentials. The orientation-dependent potentials can reflect the effect of no-bonded interactions, such as hydrogen bonding and salt bridges [20]. The different characteristics suggest that concurrently taking these two contributions into account is reasonable and promising. Although a linear combination of different energy terms is widely adopted in most energy functions [27, 90], optimizing the weight parameter of these energy function terms is not easy [16, 91]. On the other hand, multiobjectization provides another perspective from which to address this problem. The methods of splitting an energy function into a short-range term and a long-range term have aroused the interest of researchers [60, 54, 59]. More importantly, the shape of multiobjective energy functions is steady and is not sensitive to these weights. These characteristics benefit the

conformation search of a prediction system.

In this work, the RWplus potential is decomposed into two energy terms: the distance-dependent energy term and the orientation-dependent energy term. The abovementioned reasons provide evidence for allowing us to separate them. A more detailed discussion about the conflict between the two energy terms is summarized later in Section 5.4. The two objective functions for optimization are defined as follows:

$$\begin{cases} f_1 = E_{RW}, \\ f_2 = wE_{ori}, \end{cases} \tag{4.4}$$

where f_1 is the first objective function and f_2 is the second objective function. E_{RW}, w, and E_{ori} have the same definitions as mentioned above.

4.2 Multiobjective differential evolution

Briefly, comparing the algorithm components in the EAs and the components in MOEAs, the biggest difference between them is the scheme in the selection procedure [86]. It is difficult to compare two nondominated solutions in MOEAs. In this study, we modify the selection procedure of the canonical DE and extend it as MODE. The abovementioned two-stage strategy is incorporated into the selection procedure to assign complete order to two possible solutions. On the other hand, the goal of MOEAs is to obtain series of nondominated solutions. Maintaining a second population, also called an external archive, storing these solutions is common and wise [78, 92]. We follow this idea and maintain an external archive to interact with the current population.

The proposed MODE algorithm follows the framework of the canonical DE. However, the two main differences are the selection procedure and the external archive. To explain the details of the proposed MODE algorithm, the pseudo-code is shown in Algorithm 2.

Initially, the iteration counter t is set to 0. Within the constrained search space, all the target vectors $\mathbf{X}^{(i)(t)}$s in the current population are initialized, as described

in Section 3.2. Subsequently, the multiobjective function of each target vectors is evaluated. Later, all the target vectors are directly added to the empty external archive A. From this point, the main loop starts. For each target vector $\mathbf{X}^{(i)(t)}$, the mutation operator, crossover operator, and selection operator are executed in

Algorithm 2: The main procedure of the proposed MODE algorithm.

```
begin
    /* Initialization.                                                    */
    t ← 0.
    Initialize the population {X^(i)(t) | i ∈ {1, 2, ..., NP}}.
    Evaluate all target vectors X^(i)(t)s by multiobjective energy function.
    Add all X^(i)(t)s into the empty archive A directly.
    while t < T_max do
        for i in {1, 2, ..., NP} do
            /* Mutation.                                                  */
            Generate a new scale factor F as described in Section 4.4.
            Select a optimal solution from A as a best solution X^(best)(t).
            Create the donor vector V^(i)(t) according to Eq. 4.5.
            /* Crossover.                                                 */
            Generate a new crossover rate Cr according to Eq. 4.6.
            Create the trial vector U^(i)(t) according to Eq. 3.4.
            Evaluate U^(i)(t) by multiobjective energy function.
            /* Selection.                                                 */
            if U^(i)(t) ≺ X^(i)(t) then
                X^(i)(t+1) ← U^(i)(t)
            else if X^(i)(t) ≺ U^(i)(t) then
                X^(i)(t+1) ← X^(i)(t)
            else
                /* X^(i)(t), U^(i)(t) are non-dominated.                  */
                Calculate the rank of X^(i)(t) and U^(i)(t) referring to archive A.
                if rank_U ≤ rank_X then
                    X^(i)(t+1) ← U^(i)(t)
                else
                    X^(i)(t+1) ← X^(i)(t)
        Fetch all U^(i)(t)s and add them into archive A.
        Update archive A as described in Section 4.3.
        t ← t + 1
    Output result.
```

turn to generate a new offspring. Compared with the canonical DE, the proposed MODE algorithm has several improvements. First, the parameters F and Cr are not fixed but change dynamically. This scheme is explained in the next section. Second, the "DE/best/1" strategy is employed in the mutation procedure to generate donor vectors as follows:

$$\mathbf{V}^{(i)(t)} = \mathbf{X}^{(r_{best})(t)} + F(\mathbf{X}^{(r_2)(t)} - \mathbf{X}^{(r_3)(t)}), \tag{4.5}$$

where $\mathbf{X}^{(r_{best})(t)}$ is an optimal solution selected randomly from the solutions of rank 0 in archive A.

The selection procedure of the MODE is modified greatly from the canonical DE because the dominance relationship between $\mathbf{X}^{(i)(t)}$ and $\mathbf{U}^{(i)(t)}$ has two conditions. If one of them administrates the other, the better one is choosen as the new offspring. If they are nondominated, we refer to archive A to determine which is better. The rank of $\mathbf{X}^{(i)(t)}$ or $\mathbf{U}^{(i)(t)}$ is calculated by counting the number of solutions that dominate it in archive A. The solution with lower rank is preferred. The trail vector $\mathbf{U}^{(i)(t)}$ substitutes the target vector $\mathbf{X}^{(i)(t)}$ when it is of lower rank. Next, all the trail vectors are added to archive A if the addition criterion is met. The updating strategy for archive A is shown in Section 4.3. Finally, the algorithm terminates and outputs the result.

4.3 The archive based on nondominated sorting

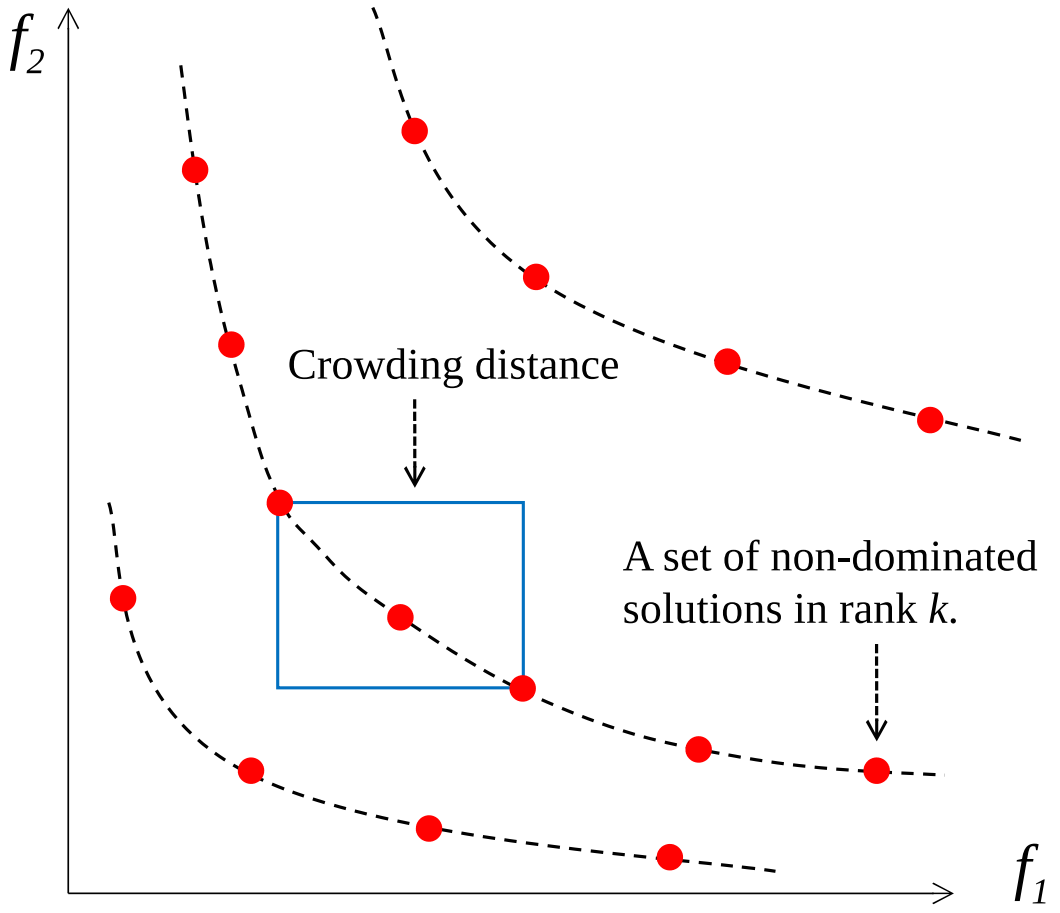


Figure 4.2: The crowding distance of a solution is defined as the perimeter of the rectangle determined by its two neighbors in the same rank.

It is an effective way to maintain an external archive to store nondominated solutions along with evolution in many MOEAs [78, 92]. An external archive with a bound size N_A is also maintained in the proposed MODE algorithm. Inspired by previous works [11, 12, 58], this archive A not only stores the nondominated solutions in the Pareto front but also other suboptimal solutions with slightly worse energy. The method of assigning complete order over all solutions is as discussed in Section 3.3. The updating strategy for archive A is shown in Fig. 4.3. At each iteration, if the archive is not full, all the trail vectors $\mathbf{U}^{(i)(t)}$s are directly added into A. On the other

hand, if A is full, nondominated sorting is used. First, all the trail vectors $\mathbf{U}^{(i)(t)}$s are appended to A. A new set $A \cup U$ of size $(N_A + NP)$ is generated. Next, each solution is assigned a rank value, and the solutions with the same rank value are collected into a subset. Then, these solution subsets are sorted in ascending order by the rank value. Next, the triming operation is implemented backward on $A \cup U$ to delete the worse solutions (with higher rank values). To keep the size of A equal to N_A, the last deleted set, "Rank k", is sorted according to density. In fact, these solutions in "Rank k" are nondominated. This inference has been proven in Section 3.3. The density of each solution in "Rank k" is estimated by the crowding distance, as calculated in the NSGA-II algorithm [77]. Fig. 4.2 shows an example of how to calculate the crowding distance of a solution. The solutions with a larger crowding distance are considered in sparse regions and are preferred. According to the crowding distance values, All of these solutions in "Rank k" are sorted in descending order. The triming operation is then implemented backward on "Rank k" to delete the worse solutions. Finally, archive A is updated.

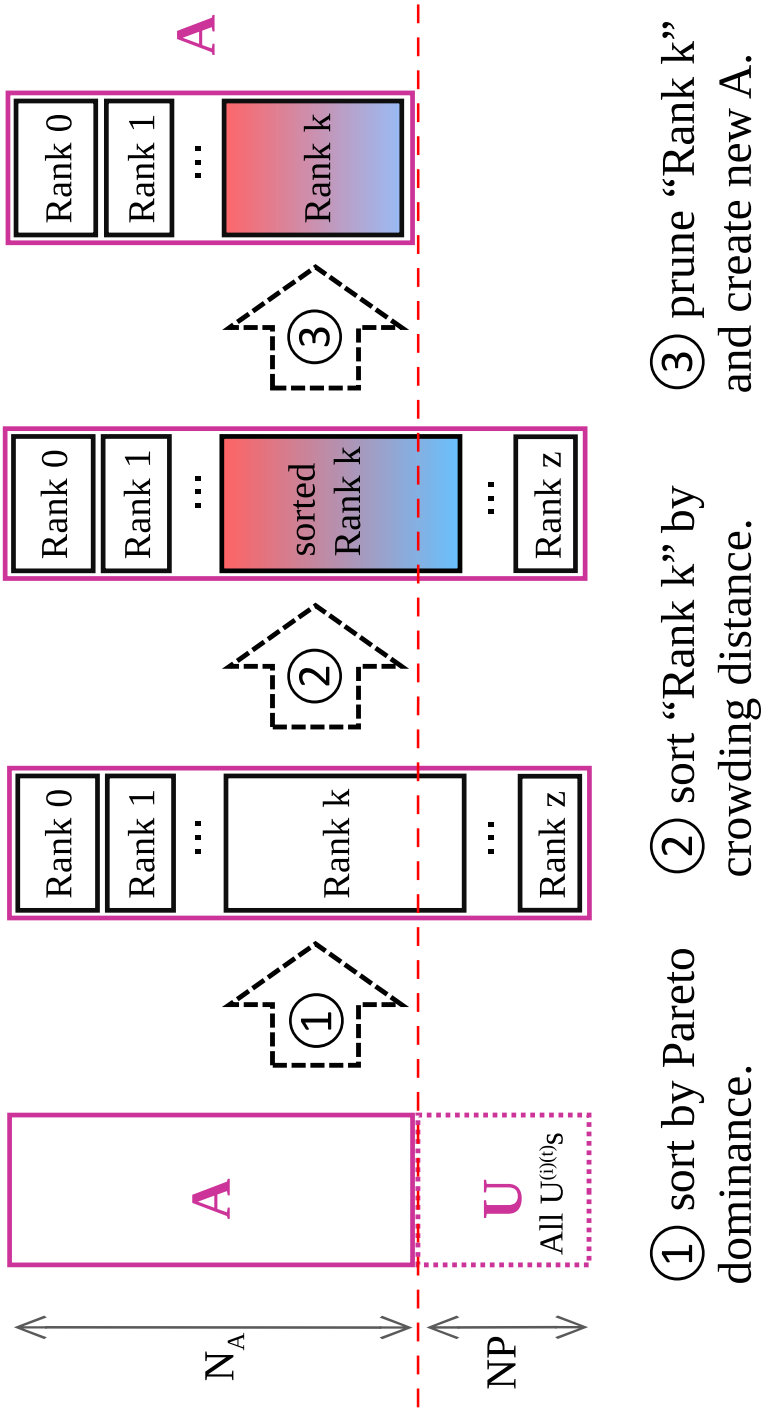


Figure 4.3: The updating strategy for archive A.

It is worth noting that the solutions of rank 0 in archive A consist of the Pareto optimal set, which are so far reached by MODE. In extreme cases, archive A is full of nondominated solutions, which are all assigned a rank of 0. The diversity of the Pareto front is ensured by the utilization of the crowding distance. In addition, since updating archive A is time consuming, we update it after updating all of the current population. Only at the beginning of updating archive A, all newly generated trail vectors $\mathbf{U}^{(i)(t)}$s are added into archive A. Thus, the proposed MODE is considered a synchronous DE [45].

4.4 Parameter controlling

There are two main control parameters in DE: the mutation scale factor F, and the crossover rate Cr. The performance of DE is highly associated with these parameters [45]. Inspired by the work of saDE [93], the mutation scale factor F in MODE fits a Gaussian distribution with mean $\mu = 0.5$ and standard deviation $\sigma = 0.3$. A new F is generated and applied to each target vector in the mutation, as shown in Algorithm 2. In this way, both exploitation (with small values) and exploration (with large values) are balanced throughout evolution. This setting satisfies the characteristics of the PSP problem. On the other hand, in the classic Rosetta approach [26, 11] the protein conformation is modified by inserting a 9-residue fragment in the beginning stage and a 3-residue fragment in the ending stage. It suggests that the crossover rate Cr in the proposed MODE changes throughout the evolution process as follows:

$$Cr = \frac{40}{d} exp(-\frac{t}{T_{max}}), \tag{4.6}$$

where d is the number of dimensions, i.e., the number of torsion angles of a protein. t is the iteration counter, and T_{max} is the maximum number of iterations. In this way, the number of modified torsion angles changes from approximately 40 to 14 throughout the evolution procedure. In other words, the equivalent number of modified residues changes from approximately 9 to 3 throughout evolution.

4.5 Complexity analysis

Compared with the canonical DE, the additional complexity of the proposed MODE mainly depends on updating the archive A. We make an analysis of the time complexity of updating the archive A as follows. For two solutions with m objective values, $O(m)$ comparisons are required to determine the dominance relationship. Then, to allocate a rank value to each solution in $A \cup U$, $O((N_A + NP)(N_A + NP - 1)/2)$ Pareto dominance comparisons are required. Later, the crowding distance of each solution in "Rank k" is calculated, and according to the crowding distance valuse, these solutions are sorted. The complexity of these two processes is far less than $O(N_A + NP) + O((N_A + NP)(N_A + NP - 1)/2)$. Finally, the complexity of the pruning operation is less than $O(N_A + NP)$. Thus, the complexity of updating the archive A can be calculated as follows:

$$\begin{aligned} &\leq O(m)O((N_A + NP)(N_A + NP - 1)/2) + O(N_A + NP) \\ &\quad + O((N_A + NP)(N_A + NP - 1)/2) + O(N_A + NP) \\ &= O((m + 1)(N_A + NP)(N_A + NP - 1)/2) + 2O(N_A + NP) \\ &\approx O(m(N_A + NP)^2), \end{aligned} \tag{4.7}$$

Since the archive A is updated at every iteration, the additional complexity is $O(m(N_A + NP)^2 T_{max})$. Considering the complexity of the canonical DE [44] is $O(NP \cdot dT_{max})$, the overall complexity of the proposed MODE is $O((NP \cdot d + m(N_A + NP)^2)T_{max})$.

4.6 Implementation of MODE in PSP

We model the PSP problem as a MOOP and use the MODE algorithm to solve it. As shown in Fig. 4.1, the proposed MODE algorithm plays a central role in MODE-K. To accomplish the MODE algorithm to solve the PSP problem, the knowledge-based energy function RWplus is employed as the fitness function to assess a solution in MODE. Specifically, the representation of the Cartesian space of a protein is generated from the representation of torsion angles for the purpose of evaluating energy.

Table 4.1: Constraints of the secondary structure for torsion angles ϕ and ψ.

	ϕ	ψ
α-helix	$[-67°, -47°]$	$[-57°, -37°]$
β-sheet	$[-130°, -110°]$	$[110°, 130°]$
coil	$[-180°, 180°]$	$[-180°, 180°]$

The variation range of each torsion angle is $[-180°, 180°]$. Additional restrictions are set on these torsion angles to narrow the search space. In fact, the search space sampled by FM approaches is usually limited in some way [94]. A secondary structure prediction method PSIPRED [64] is used to predict the secondary structure of a query protein. Each residue of the query sequence is distributed one type: α-helix, β-sheet, or coil. Then, the torsion angles ϕ and ψ of each type are restricted, as shown in Table 4.1. Moreover, all ω are set to 180°, which fits the observation that ω is very close to 180° in most natural proteins [95]. Furthermore, all side-chain torsion angles are limited in the constraints that are extracted from the Rotamer library [96]. Finally, the minimum and maximum bounds of MODE, i.e., $\mathbf{X}_{min}$ and $\mathbf{X}_{max}$, are set according to these constraints.

4.7 Decoy selection

Usually, a large-size set of decoy structures is generated by a typical FM approach before the final decision is made. The selection of the final predicted structure from these decoys is an important issue in the PSP field [97]. The process of decoy selection is essential for an integrated FM approach because it is not meaningful if we can not pick out one or a few correct structures from series of decoy structures after conformation space searching. The methods based on clustering [98] are commonly used in decoy selection [99, 97]. The clustering methods are effective because they can make use of the consensus information extracted from the set of decoy structures. In this study, we use the fast model clustering method, called MUFOLD-CL [100], to

select the best near-native model from series of decoy structures. After clustering all of the solutions in archive A, the cluster centroid with a large cluster size is selected as the final predicted structure.

Chapter 5

Experimental studies

In this chapter, we present the experiments to evaluate the performance of the proposed MODE-K approach. We report the results obtained by MODE-K and compare them to the results of other works.

5.1 Experimental setup

We use the most common metric, the root-mean-square deviation (RMSD), to measure the similarity between two structures. The RMSD is calculated as follows:

$$RMSD_{(s_1,s_2)} = \sqrt{\frac{\sum_{i=1}^{n} d_i^2}{n}}, \tag{5.1}$$

where two structures s_1 and s_2 have been superimposed optimally by using the Kabsch rotation matrix. n is the number of matching atoms, and d_i is the distance between the ith atom in s_1 and the matched ith atom in s_2. When a predicted structure is compared with the native structure, the smaller the value of RMSD is, the more accurate the predicted structure. Usually, the C_α atoms in the backbone are only considered for calculating the RMSD in real-world applications.

Eighteen proteins are used as the test instances to verify the performance of MODE-K. The sequences the test proteins are listed in Table 5.1. Detailed information about them is listed in Table 5.2. The structural classification of the test proteins contain α, β, and α/β. Table 5.2 also shows that the degrees of freedom of

Table 5.1: The sequences of test proteins.

PDB ID	Sequence
1AB1	TTCCPSIVARSNFNVCRLPGTSEAICATYTGCIIIPGATC PGDYAN
1BDD	TADNKFNKEQQNAFYEILHLPNLNEEQRNGFIQSLKDDPS QSANLLAEAKKLNDAQAPKA
1DFN	DCYCRIPACIAGERRYGTCIYQGRLWAFCC
1E0G	DSITYRVRKGDSLSSIAKRHGVNIKDVMRWNSDTANLQPG DKLTLFVK
1E0M	SMGLPPGWDEYKTHNGKTYYYNHNTKTSTWTDPRMSS
1ENH	RPRTAFSSEQLARLKREFNENRYLTERRRQQLSSELGLNE AQIKIWFQNKRAKI
1I6C	KLPPGWEKRMSRSSGRVYYFNHITNASQWERPSGNSSSG
1K36	VSITKCSSDMNGYCLHGQCIYLVDMSQNYCRCEVGYTGVR CEHFFL
1ROP	MTKQEKTALNMARFIRSQTLTLLEKLNELDADEQADICES LHDHADELYRSCLARF
1SXD	GSHMAALEGYRKEQERLGIPYDPIHWSTDQVLHWVVWVMK EFSMTDIDLTTLNISGRELCSLNQEDFFQRVPRGEILWSH LELLRKYVLAS
1ZDD	FNMQCQRRFYEALHDPNLNEEQRNAKIKSIRDDC
2GB1	MTYKLILNGKTLKGETTTEAVDAATAEKVFKQYANDNGVD GEWTYDDATKTFTVTE
2KDL	TTYKLILNLKQAKEEAIKELVDAGTAEKYIKLIANAKTVE GVWTLKDEIKTFTVTE
2M7T	GCPQGRGDWAPTSCSQDSDCLAGCVCGPNGFCG
2P6J	MKQWSENVEEKLKEFVKRHQRITQEELHQYAQRLGLNEEA IRQFFEEFEQRK
2P81	AKREFNENRYLTERRRQQLSSELGLNEAQIKIWFQNKRAK IKKS
3DF8	SNAMLRYGDTEICIDPSESVLHLLGKKYTMLIISVLGNGS TRQNFNDIRSSIPGISSTILSRRIKDLIDSGLVERRSGQI TTYALTEKGMNVRNSLMPLLQYISVLDRN
3NRW	RPSLSPREARDRYLAHRQTDAADASIKSFRYRLKHFVEWA EERDITAMRELTGWKLDEYETFRRGSDVSPATLNGEMQTL KNWLEYLARIDVVDEDLPEKVHVP

the prediction system is reduced significantly by using the representation of torsion angles.

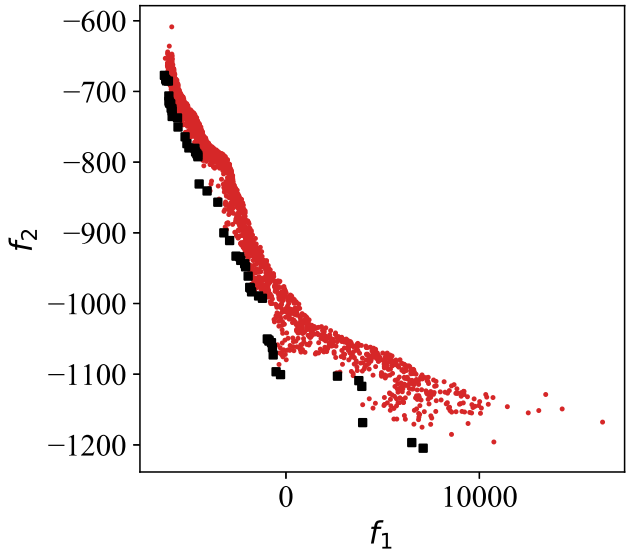
All algorithms are carried out in C and Python. They are executed on a Linux 64-bit system with Core-i5 CPU, 3.4 GHz, and 8 GB memory. As followed, the

Table 5.2: Details of the test proteins.

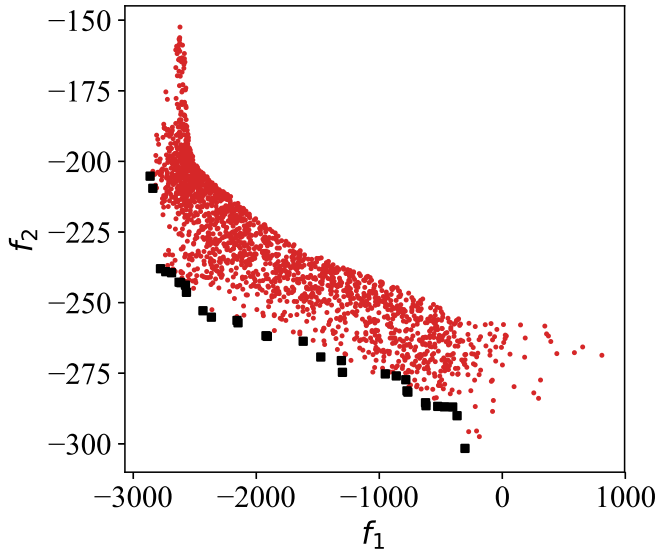
PDB ID	Length (Number of amino acids)	Structural class	Number of torsion angles	Number of atoms
1AB1	46	α/β	192	645
1BDD	60	α	297	942
1DFN	30	β	139	471
1E0G	48	α/β	236	777
1E0M	37	β	174	586
1ENH	54	α	289	947
1I6C	39	β	185	613
1K36	46	β	218	708
1ROP	56	α	284	905
1SXD	91	α	448	1501
1ZDD	34	α	180	570
2GB1	56	α/β	264	855
2KDL	56	α	278	920
2M7T	33	α/β	134	414
2P6J	52	α	289	921
2P81	44	α	241	780
3DF8	109	α/β	529	1733
3NRW	104	α	517	1714

parameters of the MODE are set. The population size NP is set to 50, the archive size N_A is set to 2000, and the maximum iteration T_{max} is set to 2000. As a result, the cost budget for one prediction at a time is $100,000$ evaluations. Normally, it takes about 20 hours to run a structure prediction.

5.2 Optimization results



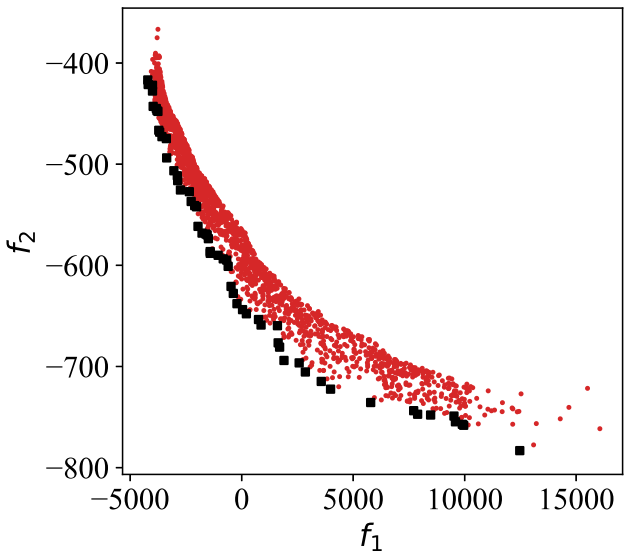
(a) 1BDD



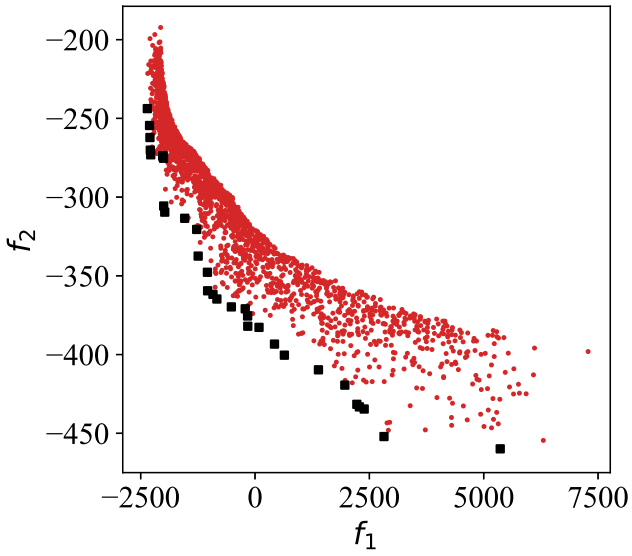
(b) 1DFN

Figure 5.1: The optimization results for 1BDD and 1DFN. The solutions of rank 0 (Pareto front) in A are marked in black, and other solutions are marked in red.

We applied the proposed MODE-K approach to these test proteins. For each test protein, series of optimal solutions stored in archive A are generated after optimiza-



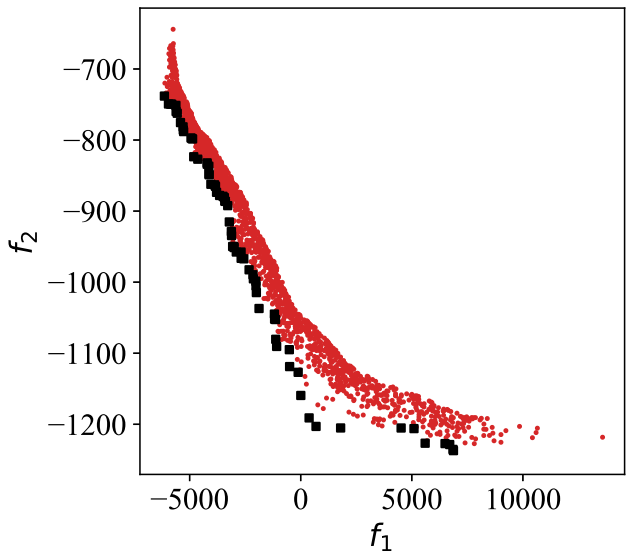
(a) 1E0G



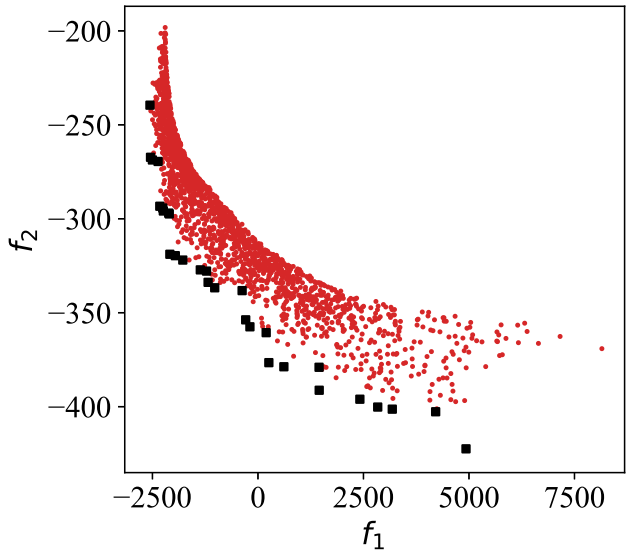
(b) 1E0M

Figure 5.2: The optimization results for 1E0G and 1E0M. The solutions of rank 0 (Pareto front) in A are marked in black, and other solutions are marked in red.

tion. The solutions of each test protein in the objective space are plotted in Fig. 5.1 $\sim$ Fig. 5.8, and Fig. 5.15 $\sim$ Fig. 5.17 (a). Moreover, the Pareto optimal solutions (i.e., the solutions of rank 0 in archive A) are marked in black points. It is clear that the



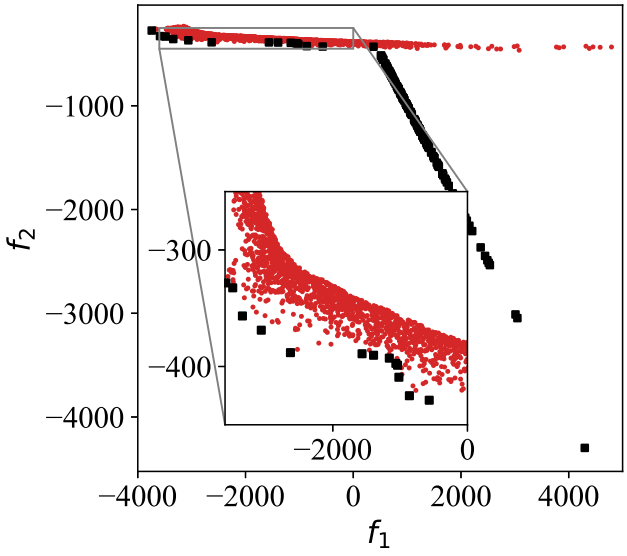
(a) 1ENH



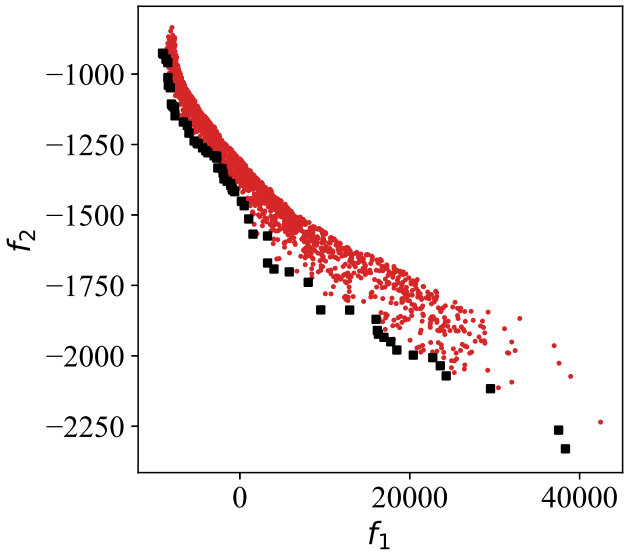
(b) 1I6C

Figure 5.3: The optimization results for 1ENH and 1I6C. The solutions of rank 0 (Pareto front) in A are marked in black, and other solutions are marked in red.

solutions of different test proteins form different images in the objective space. This finding indicates that the landscapes of energy functions of these test proteins have different characteristics. Moreover, these solutions in archive A are considered diverse



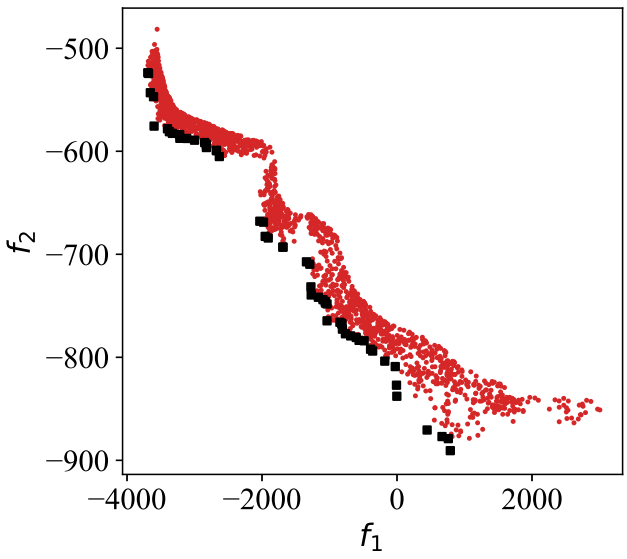
(a) 1K36



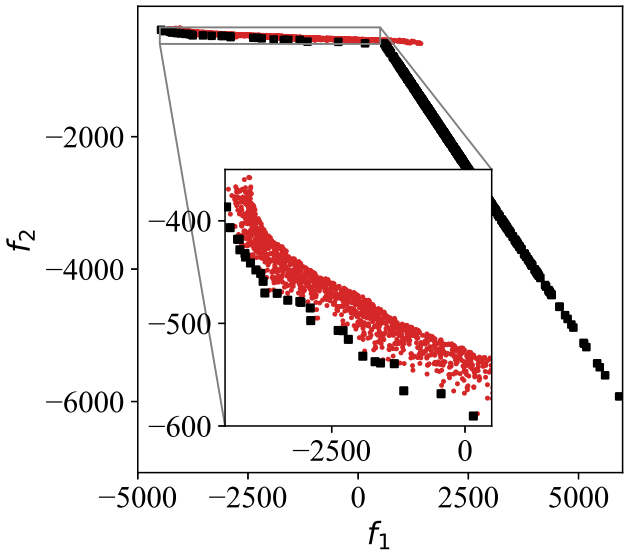
(b) 1SXD

Figure 5.4: The optimization results for 1K36 and 1SXD. The solutions of rank 0 (Pareto front) in A are marked in black, and other solutions are marked in red.

because they are distributed along the Pareto front in each figure. It strengthens the advantages of the proposed MODE algorithm, where the scheme of nondominated sorting to ensure diversity in the solutions is employed.



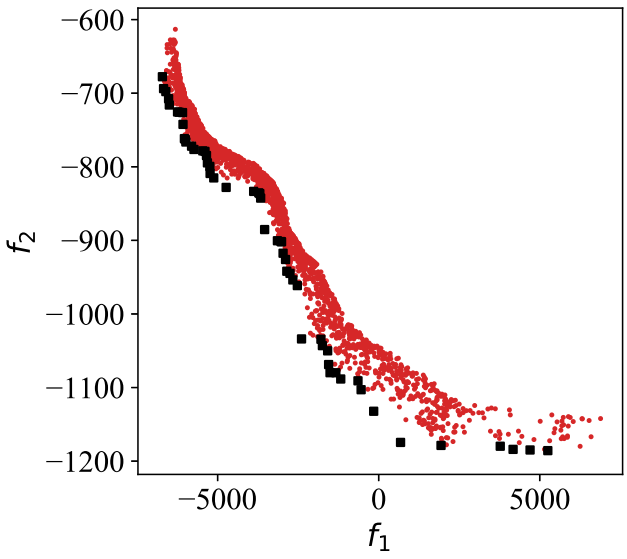
(a) 1ZDD



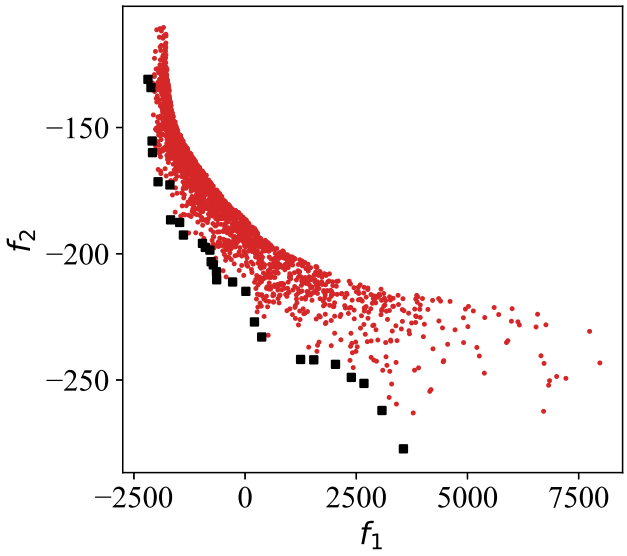
(b) 2GB1

Figure 5.5: The optimization results for 1ZDD and 2GB1. The solutions of rank 0 (Pareto front) in A are marked in black, and other solutions are marked in red.

The solutions of protein 1K36 and 2GB1 exhibit uncustomary images in the objective space, where the Pareto fronts form long tails. The second objective of these solutions in the long tail is optimized and sufficiently small. We investigate the



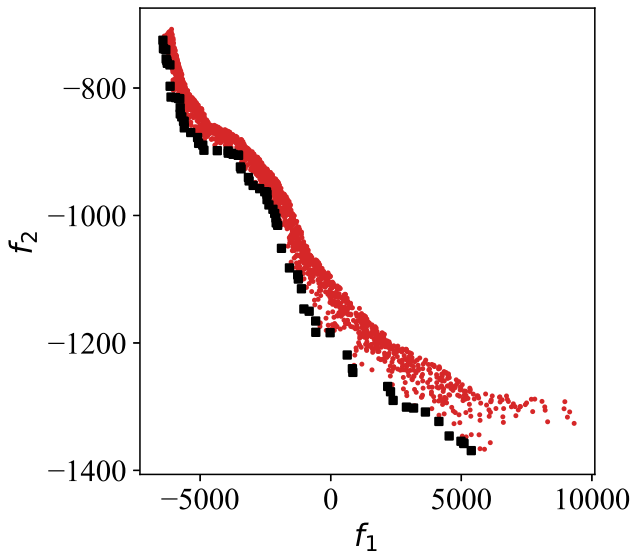
(a) 2KDL



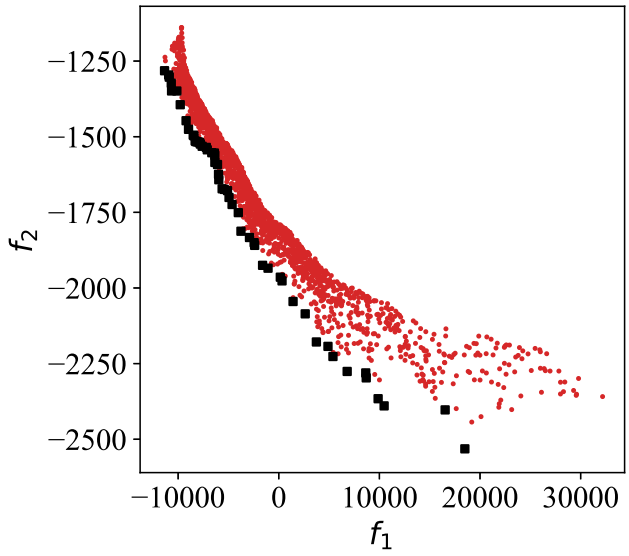
(b) 2M7T

Figure 5.6: The optimization results for 2KDL and 2M7T. The solutions of rank 0 (Pareto front) in A are marked in black, and other solutions are marked in red.

RMSDs of these solutions. They have poor accuracies, and their RMSD values are near 30 Å. In brief, the search algorithm falls into the local optima of the rugged energy functions. However, there are still many solutions with high accuracy main-



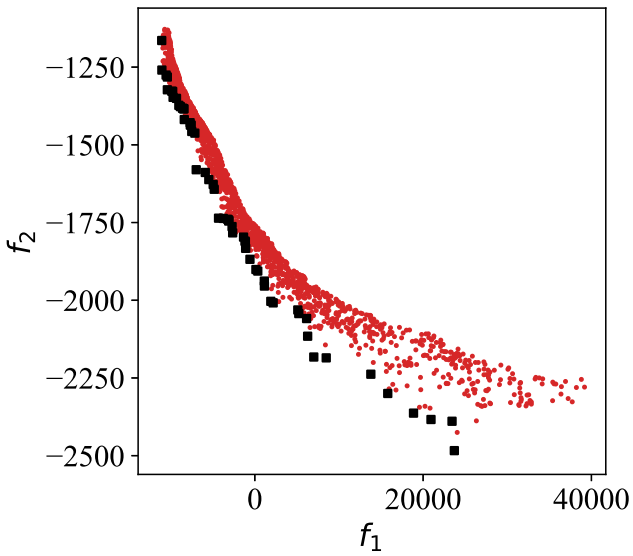
(a) 2P6J



(b) 3DF8

Figure 5.7: The optimization results for 2P6J and 3DF8. The solutions of rank 0 (Pareto front) in A are marked in black, and other solutions are marked in red.

tained in archive A. These solutions are more likely to cluster in the basins of the energy functions. Hence, the decoy selection method cannot be easily misled by the worse solutions. These findings strengthen the necessity of retaining the solutions

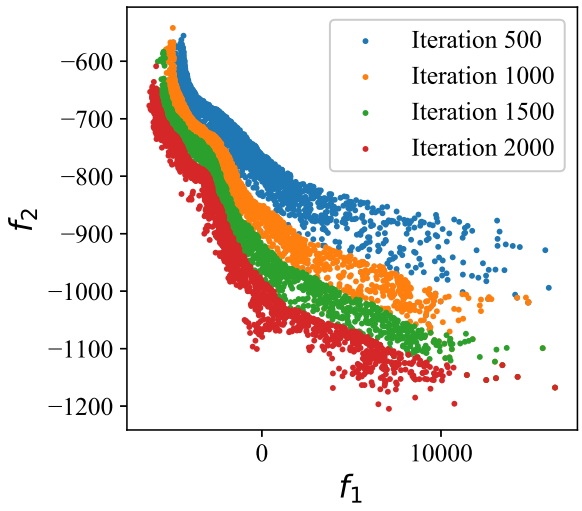


(a) 3NRW

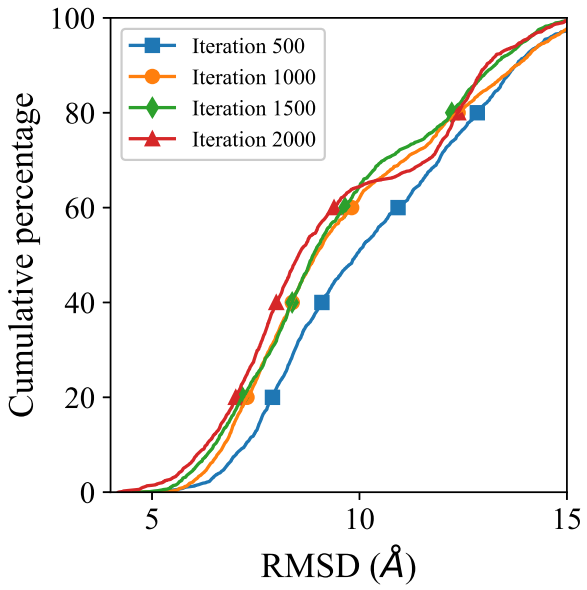
Figure 5.8: The optimization results for 3NRW. The solutions of rank 0 (Pareto front) in A are marked in black, and other solutions are marked in red.

with slightly worse energy, and the updating strategy for archive A is considered appropriate.

5.3 The evolution of archive A



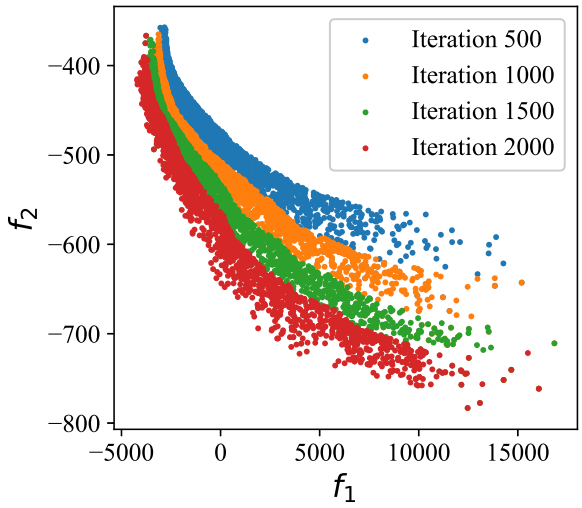
(a) 1BDD



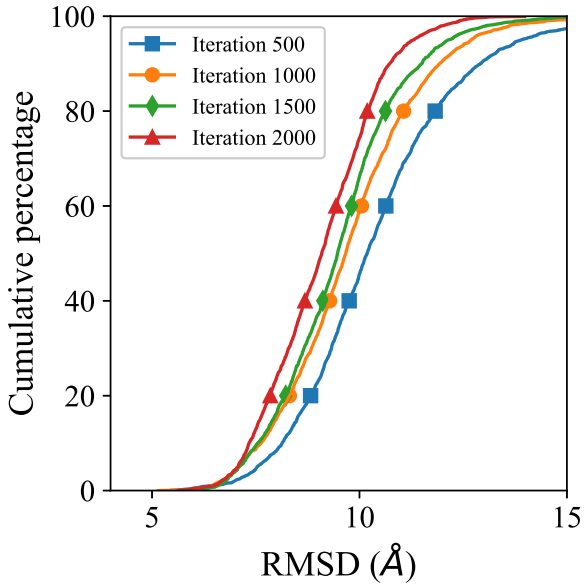
(b) 1BDD

Figure 5.9: For 1BDD (α), the dynamics of the solutions in archive A at iterations 500, 1000, 1500, and 2000 are exhibited in subfigures (a). The corresponding cumulative distribution of the RMSD values for all solutions in archive A are plotted in subfigures (b).

To investigate the relationship between the multiobjective energy and the con-



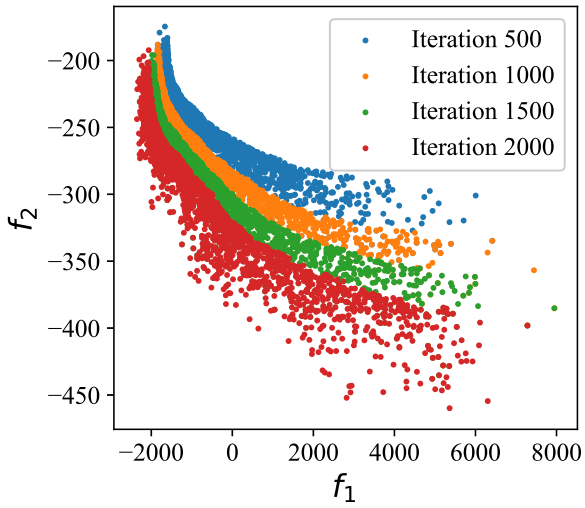
(a) 1E0G



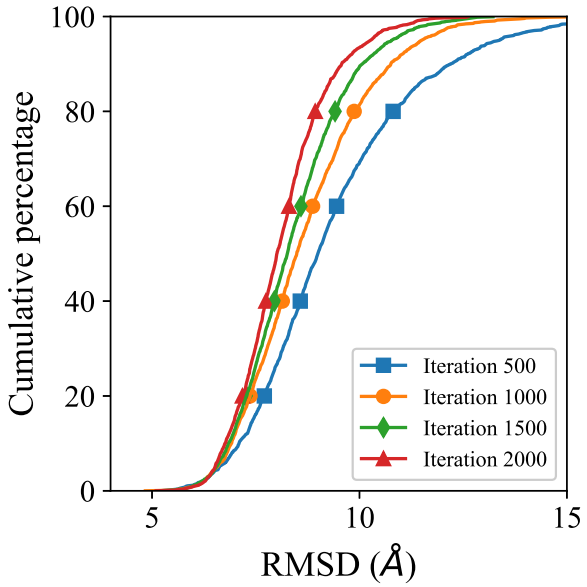
(b) 1E0G

Figure 5.10: For 1E0G (α/β), the dynamics of the solutions in archive A at iterations 500, 1000, 1500, and 2000 are exhibited in subfigures (a). The corresponding cumulative distribution of the RMSD values for all solutions in archive A are plotted in subfigures (b).

formation accuracy, we consider all the solutions in archive A during evolution. For three typical proteins, 1BDD (α), 1E0G (α/β), and 1E0M (β), Fig. 5.9 $\sim$ Fig. 5.11 shows the dynamics of these solutions in archive A at iterations of 500, 1000, 1500,



(a) 1E0M



(b) 1E0M

Figure 5.11: For 1E0M (β), the dynamics of the solutions in archive A at iterations 500, 1000, 1500, and 2000 are exhibited in subfigures (a). The corresponding cumulative distribution of the RMSD values for all solutions in archive A are plotted in subfigures (b).

and 2000. The two objective energy functions of them are shown in Fig. 5.9 $\sim$ Fig. 5.11 (a), respectively. The corresponding cumulative distribution functions of RMSD values are shown in Fig. 5.9 $\sim$ Fig. 5.11 (b), respectively. From Fig. 5.9 $\sim$ Fig. 5.11

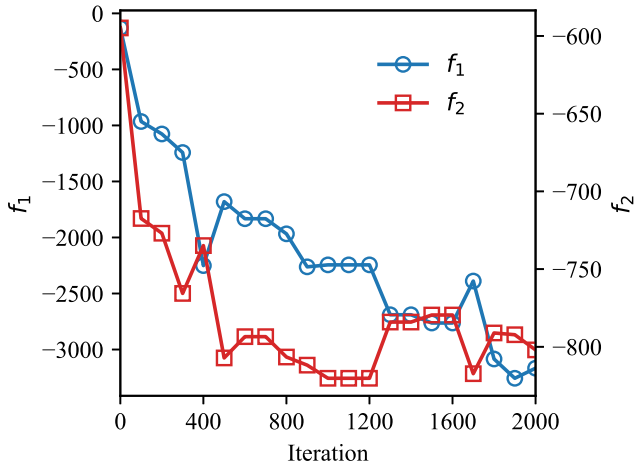
(a), it is clear that the solutions in archive A shift entirely to the lower region of both objective energy functions. Moreover, their quality improved gradually because the percentage of solutions with a high accuracy increased, as shown in Fig. 5.9 $\sim$ Fig. 5.11 (b). For the rest of the test proteins, they also got similar results. These results show that the solutions with lower energies correspond to a roughly more accurate structure, even for the multiobjective energy function.

5.4 Investigating the conflicts

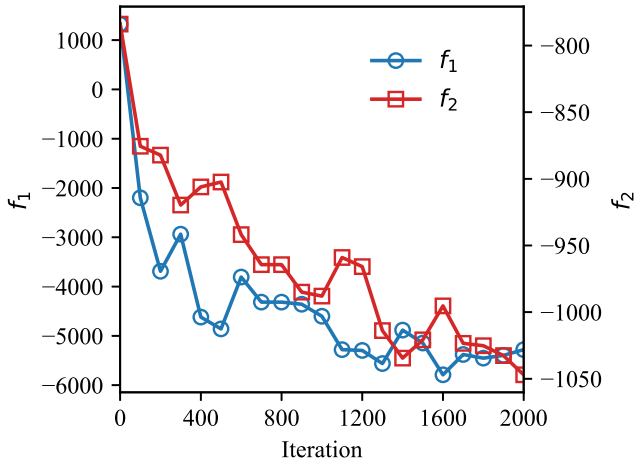
The conflicts among different objectives is a typical characteristic of a MOOP [101]. In this study, we constitute the bi-objective energy function by decomposing the knowledge-based energy function RWplus into two energy terms: the distance-dependent energy term and the orientation-dependent energy term. Since there is no ceremonial definition of conflicts among objectives in the field of MOOPs, we provide some insights about the conflict between the two objectives experimentally.

To investigate the conflict between the two objectives, we trace the optimization process of the test proteins during the iterations of the MODE algorithm. Six typical test proteins with three different structural classes are investigated. The simple intuitions for six test proteins are displayed in Fig. 5.12 $\sim$ 5.14. For each test protein, the two objective functions of a common individual during the iterations of MODE are plotted. The remaining test proteins have similar intuitions. In Fig. 5.12 $\sim$ Fig. 5.14, we add to the number of iterations, the two objective functions gradually decrease. This phenomenon is due to the powerful optimization performance of the proposed MODE algorithm. From another aspect, two functions compete with each other for minimization. It's obviouslly that the two objective functions are in conflict because in general, one function increases as the other function decreases.

These findings can also be explained from the viewpoint of structural biology. Given two blocks in a protein conformation, the physical contact distance between them plays a leading role in close distances; thus, the distance-dependent contributions are more sensitive than the orientation-dependent contributions in this case. When the contact distance between the two squares is large, the orientation-dependent contributions are more sensitive because nonbonded interactions depend on specific angles. On the other hand, a protein conformation is considered a collection of blocks. The optimization process of the protein energy function is considered as a protein folding process. A perturbation of the protein conformation can locally decrease the distance-dependent contributions and globally increase the orientation-dependent contributions, and vice versa. As a result, two objective functions are in



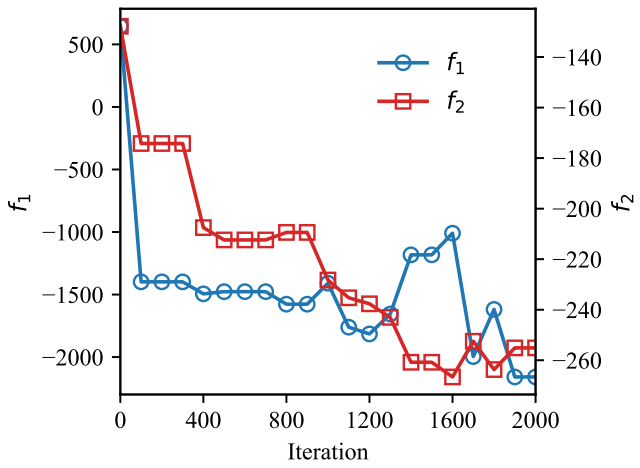
(a) 1BDD (α)



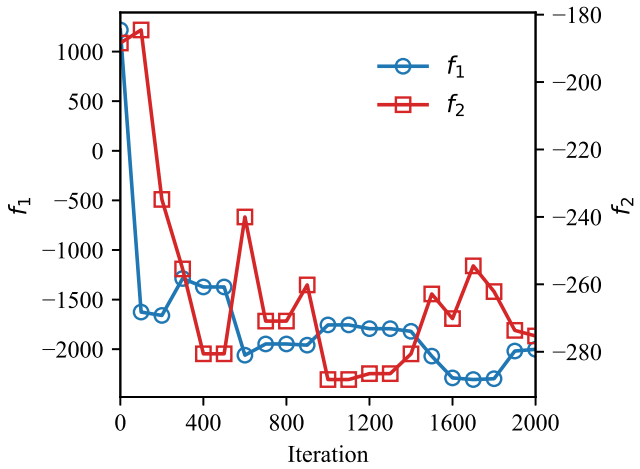
(b) 1ROP (α)

Figure 5.12: For two typical α proteins, the conflict between the two objective functions of common individuals during the iterations of MODE are investigated. Generally, one function increases as the other function decreases.

conflict during the optimization process.

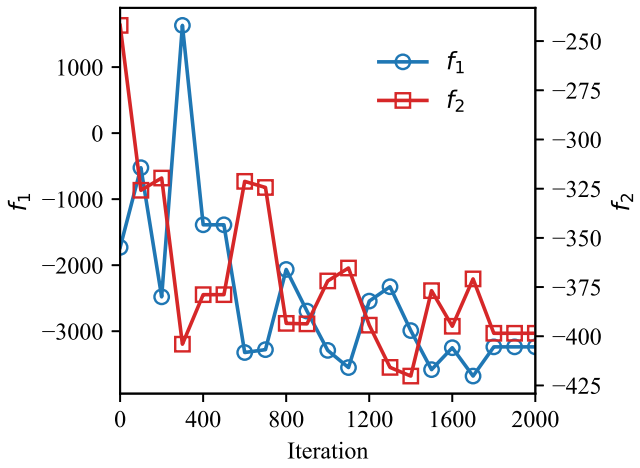


(a) 1DFN (β)

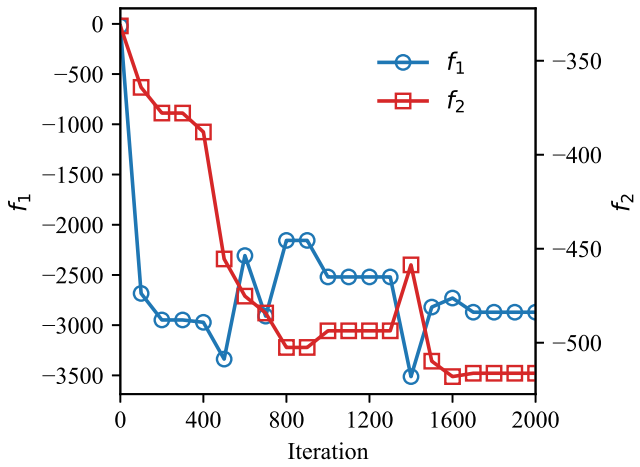


(b) 1E0M (β)

Figure 5.13: For two typical β proteins, the conflict between the two objective functions of common individuals during the iterations of MODE are investigated. Generally, one function increases as the other function decreases.



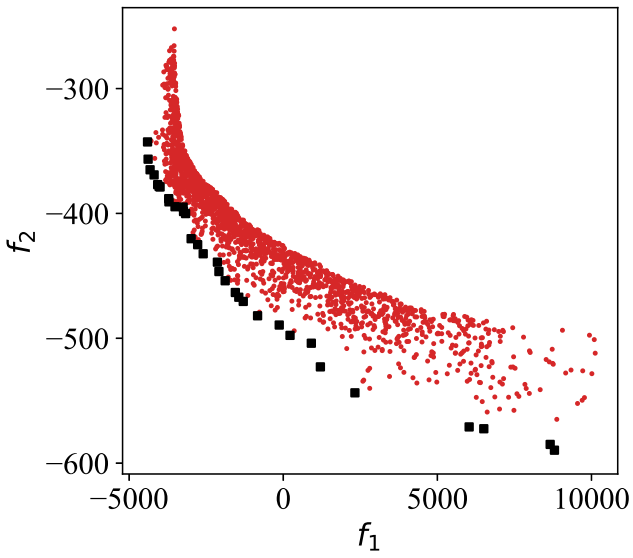
(a) 1AB1 (α/β)



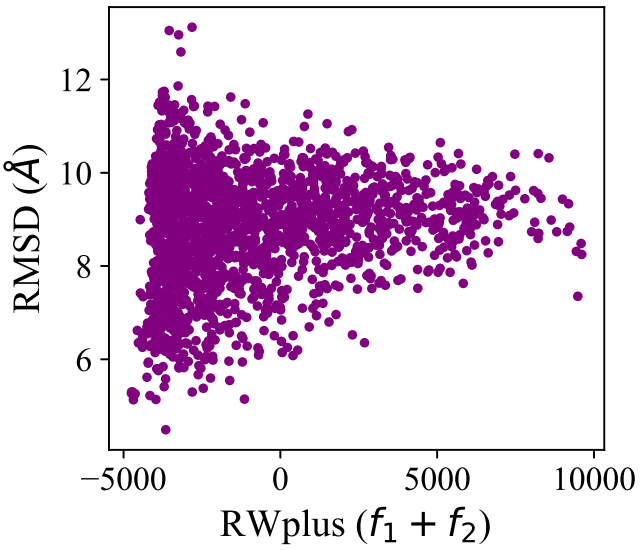
(b) 1E0G (α/β)

Figure 5.14: For two typical α/β proteins, the conflict between the two objective functions of common individuals during the iterations of MODE are investigated. Generally, one function increases as the other function decreases.

5.5 Energy versus accuracy



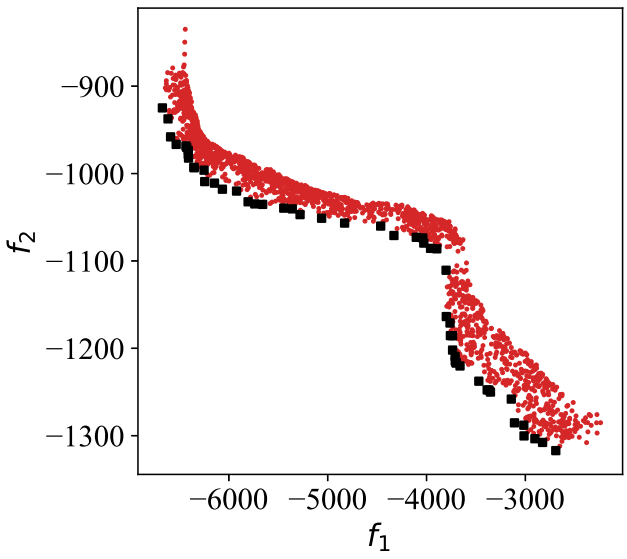
(a) 1AB1



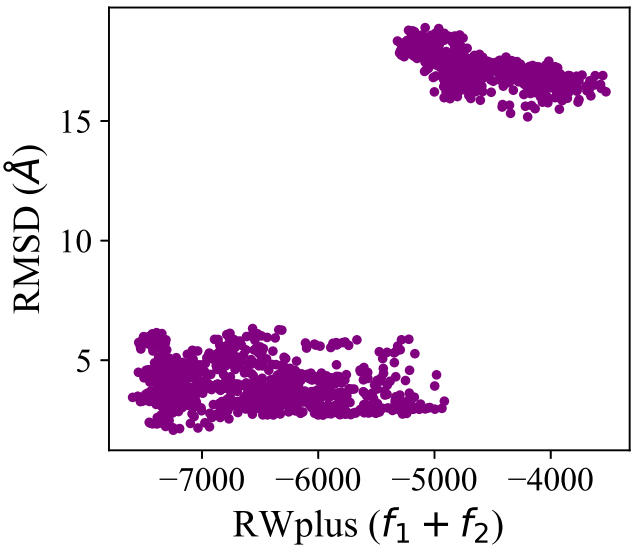
(b) 1AB1, PCC=0.15

Figure 5.15: For protein 1AB1, the image of the solutions of archive A in the objective space are shown in subfigure (a). The correlation of the energy versus the RMSD are shown in subfigure (b).

The correlation between the accuracy and the original energy RWplus ($f_1 + f_2$)



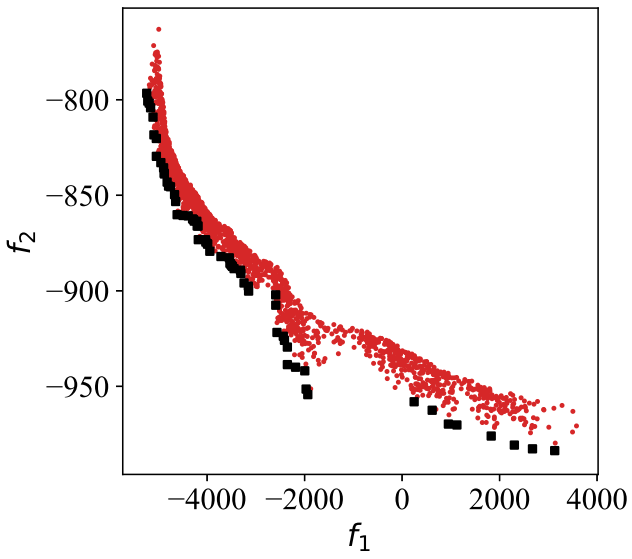
(a) 1ROP



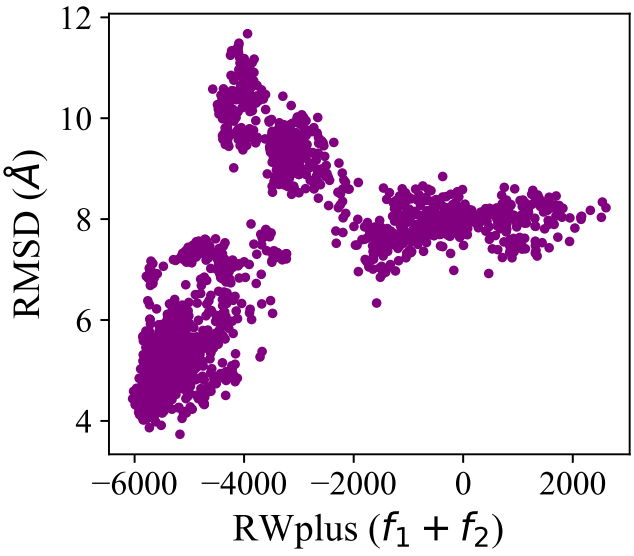
(b) 1ROP, PCC=0.86

Figure 5.16: For protein 1ROP, the image of the solutions of archive A in the objective space are shown in subfigure (a). The correlation of the energy versus the RMSD are shown in subfigure (b).

of the predicted structures is also investigated. Three typical energy landscapes are displayed in Fig. 5.15 $\sim$ Fig. 5.15 (b), where the RMSD values of the solutions in archive A are plotted along with their RWplus energy. Overall, the RWplus energy



(a) 2P81



(b) 2P81, PCC=0.54

Figure 5.17: For protein 2P81, the image of the solutions of archive A in the objective space are shown in subfigure (a). The correlation of the energy versus the RMSD are shown in subfigure (b).

has an obvious positive correlation with the RMSD because the values of the Pearson correlation coefficient (PCC) for the three proteins are all much larger than zero. This finding means that the solutions (structures) with lower energy have roughly

a more native-like structure. This result also suggests that minimizing the RWplus energy of a conformation drives it toward the true structure. However, these solutions are grouped into different numbers of clusters. This finding seems to have a deep connection with the image formed by these solutions in the objective space. In the case of 1AB1, the image of the solutions in the objective space are smooth, as shown in Fig. 5.15 (a). The solutions cluster into one group, as shown in Fig. 5.15 (b). However, there exists an inflexion point in the image for 1ROP, as shown in Fig. 5.16 (a). This inflexion point divides the solutions into two groups with different accuracies, as shown in Fig. 5.16 (b). Moreover, there are at least three inflexion points in Fig. 5.17 (a). As a result, more than three clusters are shown in Fig. 5.17 (b). This phenomenon indicates that the proposed MODE algorithm can achieve a high sampling rate on different regions of the conformation space. Moreover, this phenomenon also indicates that the proposed MODE algorithm does not easily fall into a local optimum.

5.6 Prediction results

Table 5.3: The summary of the final prediction results (1).

PDB ID		f_1	f_2	RMSD(Å)	TM-score	GDT_TS
1AB1	Native [a]	-5.43E+03	-3.57E+02	-	-	-
	Predicted [b]	-3.56E+03	-2.80E+02	7.38	0.2352	38.04
	Average [c]	-5.34E+02	-4.16E+02	8.81	0.2397	36.22
1BDD	Native	-6.66E+03	-5.72E+02	-	-	-
	Predicted	-5.84E+03	-6.97E+02	4.98	0.4325	52.08
	Average	-1.47E+03	-8.86E+02	9.37	0.2903	38.28
1DFN	Native	-6.66E+03	-5.72E+02	-	-	-
	Predicted	-2.34E+03	-2.33E+02	7.00	0.2432	46.67
	Average	-1.82E+03	-2.32E+02	6.83	0.2447	47.86
1E0G	Native	-5.54E+03	-5.11E+02	-	-	-
	Predicted	-2.84E+03	-4.77E+02	8.10	0.2583	38.54
	Average	4.25E+02	-5.68E+02	9.08	0.2336	35.17
1E0M	Native	-3.45E+03	-3.31E+02	-	-	-
	Predicted	-7.30E+02	-3.41E+02	6.49	0.3076	45.95
	Average	-2.70E+02	-3.09E+02	8.12	0.2239	39.13
1ENH	Native	-8.79E+03	-4.89E+02	-	-	-
	Predicted	-7.90E+02	-1.01E+03	7.80	0.2890	42.13
	Average	-1.67E+03	-9.37E+02	8.62	0.3180	42.04
1I6C	Native	-3.24E+03	-2.62E+02	-	-	-
	Predicted	7.82E+02	-3.32E+02	7.76	0.1979	37.18
	Average	-4.82E+02	-3.01E+02	8.77	0.1974	35.84
1K36	Native	-4.69E+03	-3.65E+02	-	-	-
	Predicted	-2.47E+03	-3.24E+02	8.34	0.2415	40.22
	Average	-1.37E+03	-4.21E+02	12.22	0.1970	30.95
1ROP	Native	-8.13E+03	-5.83E+02	-	-	-
	Predicted	-6.00E+03	-1.01E+03	3.01	0.4902	66.07
	Average	-4.81E+03	-1.06E+03	9.14	0.4287	54.11

[a] Native structure.
[b] Final predicted solution, which is selected by MUFOLD-CL.
[c] The average value of the solutions stored in archive A.

After generating series of solutions (decoy structures) in archive A, we use MUFOLD-CL to select a solution as the predicted structure finally for each test protein. Table 5.3 and Table 5.4 report the prediction results for the eighteen test proteins, where the RMSD values and the two objective energy functions of these predicted structures are summarized. In addition, the values of two widely-used metrics TM-score [102]

Table 5.4: The summary of the final prediction results (2).

PDB ID		f_1	f_2	RMSD(Å)	TM-score	GDT_TS
1AB1	Native [a]	-5.43E+03	-3.57E+02	-	-	-
	Predicted [b]	-3.56E+03	-2.80E+02	7.38	0.2352	38.04
	Average [c]	-5.34E+02	-4.16E+02	8.81	0.2397	36.22
1BDD	Native	-6.66E+03	-5.72E+02	-	-	-
	Predicted	-5.84E+03	-6.97E+02	4.98	0.4325	52.08
	Average	-1.47E+03	-8.86E+02	9.37	0.2903	38.28
1DFN	Native	-6.66E+03	-5.72E+02	-	-	-
	Predicted	-2.34E+03	-2.33E+02	7.00	0.2432	46.67
	Average	-1.82E+03	-2.32E+02	6.83	0.2447	47.86
1E0G	Native	-5.54E+03	-5.11E+02	-	-	-
	Predicted	-2.84E+03	-4.77E+02	8.10	0.2583	38.54
	Average	4.25E+02	-5.68E+02	9.08	0.2336	35.17
1E0M	Native	-3.45E+03	-3.31E+02	-	-	-
	Predicted	-7.30E+02	-3.41E+02	6.49	0.3076	45.95
	Average	-2.70E+02	-3.09E+02	8.12	0.2239	39.13
1ENH	Native	-8.79E+03	-4.89E+02	-	-	-
	Predicted	-7.90E+02	-1.01E+03	7.80	0.2890	42.13
	Average	-1.67E+03	-9.37E+02	8.62	0.3180	42.04
1I6C	Native	-3.24E+03	-2.62E+02	-	-	-
	Predicted	7.82E+02	-3.32E+02	7.76	0.1979	37.18
	Average	-4.82E+02	-3.01E+02	8.77	0.1974	35.84
1K36	Native	-4.69E+03	-3.65E+02	-	-	-
	Predicted	-2.47E+03	-3.24E+02	8.34	0.2415	40.22
	Average	-1.37E+03	-4.21E+02	12.22	0.1970	30.95
1ROP	Native	-8.13E+03	-5.83E+02	-	-	-
	Predicted	-6.00E+03	-1.01E+03	3.01	0.4902	66.07
	Average	-4.81E+03	-1.06E+03	9.14	0.4287	54.11

[a] Native structure.
[b] Final predicted solution, which is selected by MUFOLD-CL.
[c] The average value of the solutions stored in archive A.

and GDT_TS [103] on the prediction results are also shown in Table 5.3 and Table 5.4. It can be proved that the RMSDs of the predicted structures are all less than 10 Å, except for three long proteins (i.e., 1SXD, 3DF8, and 3NRW). It indicates that longer sequences remains challenging for our proposed approach. Moreover, the superposition of the native structure and the predicted structure (selected by MUFOLD-CL) is exhibited in Fig. 5.18. These figures show the high performance of the proposed MODE-K approach because the structures with considerable accuracy are reached.

In addition, from Fig. 5.18, we can see that the accuracy of a predicted structure is dependent on its sequence length and structural class.

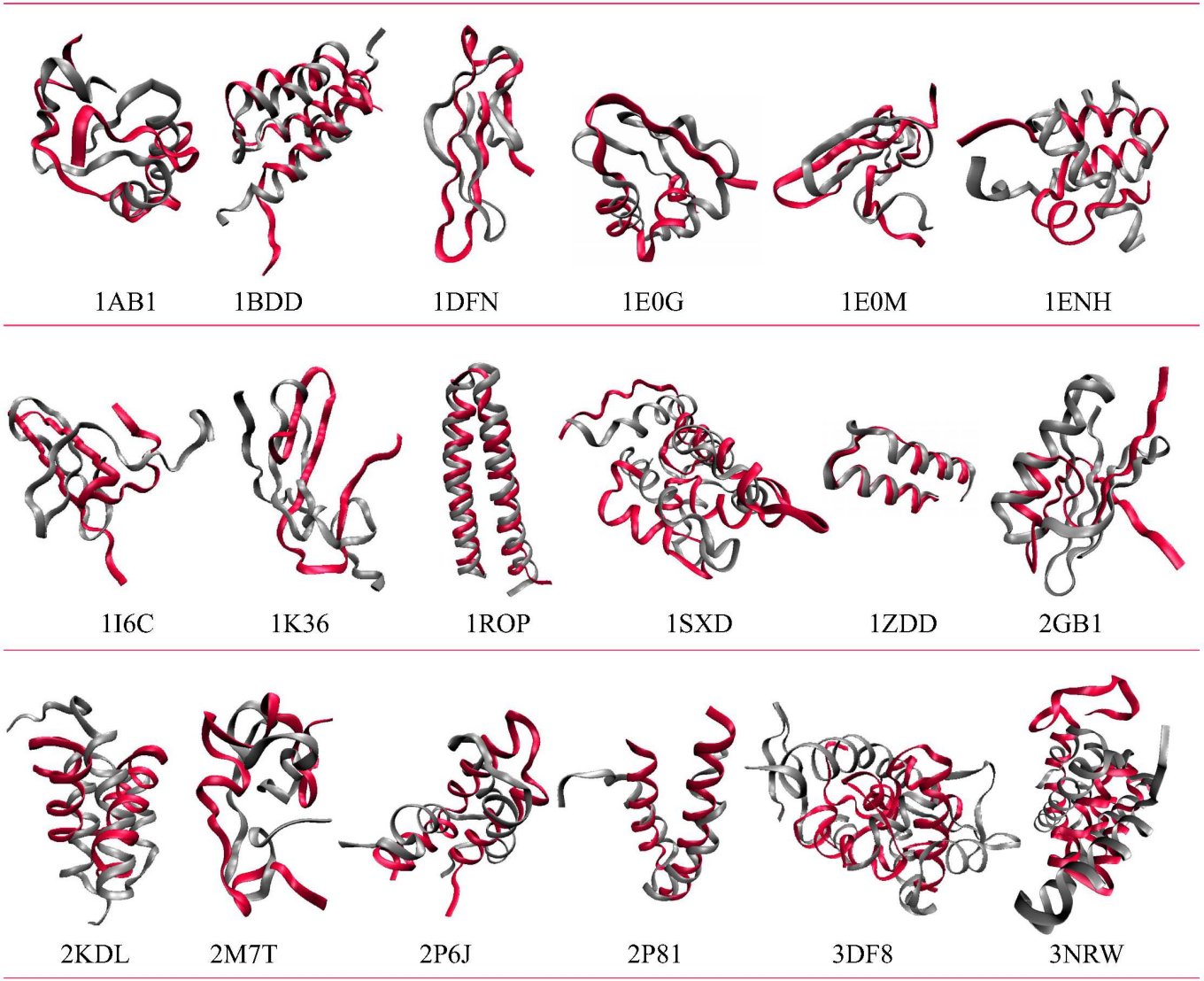


Figure 5.18: Superposition of the native structure (in gray) and the predicted structures (selected by MUFOLD-CL). The corresponding values of the RMSD refer to Table. 5.3 and Table. 5.4.

5.7 Comparison with other works based on EAs

By comparing the results obtained from the proposed MODE-K approach with seven works based on EAs in the literature. These works have been introduced in Section 2, including I-PAES [63], GA-APL [61], ADEMO/D [54], MO3 [56], AIMOES [57], SCDE [62] and MOPSO [58]. The impressive works [53, 104, 30, 105, 59], that pay more attention to decoy generation and do not contain final decoy selection methods, are not compared in this section. In these works, only the best predicted structures with the lowest RMSD value among series of generated decoy structures are reported.

The results of the seven compared approaches come from the corresponding published papers. Table 5.5 summarizes the RMSD values of the structures (after decoy selection) predicted by each approach. For each test protein, the best results of all methods are marked in bold. From Table 5.5, we can see that MODE-K achieves the best performance on 8 out of 16 test proteins in comparison with the other approaches. It suggests that the proposed MODE-K approach can provide a better or very competitive result compared with these approaches based on EAs. Moreover, MODE-K is the only approach that uses a pure multiobjective knowledge-based energy function. The comparison result indicates that incorporating KBEFs into a multiobjective approach contributes to solving the PSP problem.

Table 5.5: Comparison of the prediction results among eight approaches. Each cell contains the RMSD value (Å) of the predicted protein structure.

PDB ID	MODE-K [b]	I-PAES [b]	GA-APL [a]	ADEMO/D [b]	MO3 [b]	AIMOES [b]	SCDE [a]	MOPSO [b]
1AB1(46)	7.38	9.09	10.10	-	7.52	**6.77**	-	9.80
1BDD(60)	**4.98**	-	-	-	-	6.95	-	5.64
1DFN(30)	**7.00**	10.06	10.21	-	7.45	7.65	-	-
1E0G(48)	8.10	-	-	-	-	**7.28**	-	-
1E0M(37)	6.49	7.27	-	-	8.00	**5.94**	-	-
1ENH(54)	7.80	11.13	14.99	-	11.99	6.67	**2.52**	8.92
1I6C(39)	**7.76**	-	-	-	-	8.02	9.09	8.47
1K36(46)	**8.34**	-	-	-	-	10.15	-	-
1ROP(56)	**3.01**	3.70	9.80	4.48	3.22	-	-	3.51
1SXD(91)	**10.82**	-	-	-	-	12.12	-	-
1ZDD(34)	2.50	2.27	4.60	**2.14**	3.26	4.45	-	2.15
2GB1(56)	8.26	-	-	-	-	**6.48**	-	-
2KDL(56)	**7.72**	-	10.30	-	-	-	-	10.29
2M7T(33)	**6.83**	-	8.00	-	-	-	-	8.46
2P6J(52)	6.29	10.26	15.18	-	**5.96**	10.82	-	9.44
2P81(44)	4.76	6.81	8.53	-	**4.30**	6.43	-	6.28

[a] Single-objective approach.

[b] Multiobjective approach.

5.8 Qualitative comparison with other works based on EAs

We also contrast the results inferred from the proposed MODE-K approach with other recent works based on EAs. Specifically, these works are considered integral because the final decoy selection methods are used to select the final predicted structures. In this section, it is worth emphasizing that the accuracies of the final predicted structures are compared rather than the best structure in the generated decoy structures. The characteristics of all of these methods are summarized in Table 5.6. Note that the test proteins with lengths smaller than 30 have been removed from the comparisons for a realistic comparison. From Table 5.6, we can see that MODE-K is the only method to use a pure multiobjective knowledge-based energy function.

Since the test proteins are different in these methods, a direct comparison among them is not easy to carry out. Inspired by a previous study [106], we execute a qualitative comparison based on a logarithmic regression analysis, as shown in Fig. 5.19. The fitted function is defined as

$$y = alnx + b \tag{5.2}$$

where a and b are the parameters to fit. A point represents a prediction result of a test protein and is located by the length and the RMSD. Then, the logarithmic regression lines for all methods are plotted according to these points. According to the tendency of the regression lines, it is clear that MODE-K achieves the highest performance and provides competitive results compared with the other methods. These results strengthen the idea that using KBEFs as a multiobjective function contributes to solving the PSP problem.

To make a direct comparison among these methods, we calculated the average backbone RMSD_{100} [106], rather than an ordinary RMSD, of the prediction results obtained by these methods. The RMSD_{100} is a protein-size normalized RMSD of the backbone coordinates. This metric can approximatively compare the accuracy of the

predicted structure with different lengths. We summarize the comparison results in Table 5.7. From Table 5.7, it is easy to show that the proposed MODE-K approach achieves the second-best performance in terms of the average RMSD_{100}. Although ADEMO/D obtained the smallest average RMSD_{100} value, there were only 4 test proteins, and the test proteins did not contain the α/β structural class. Overall, the proposed MODE-K approach can provide a better or very competitive result compared with other methods based on EAs.

Table 5.6: The features of different methods based on EAs.

Method	Search strategy	Energy function	Function type	Number of objectives	Decoy selection
GA-APL [61]	genetic algorithm (GA)	Rosetta	Mixed	1	Minimum energy
ADEMO/D[54]	DE	CHARMM27	PBEF	2	Empirical point
AIMOES [57]	evolution strategy (ES)	CHARMM22, Solvent effect	PBEF, PBEF	3	Hierarchical clustering
MOPSO [58]	PSO	CHARMM22, dDFIRE	PBEF, KBEF	3	MUFOOLD-CL
MODE-K	DE	RWplus	KBEF	2	MUFOOLD-CL

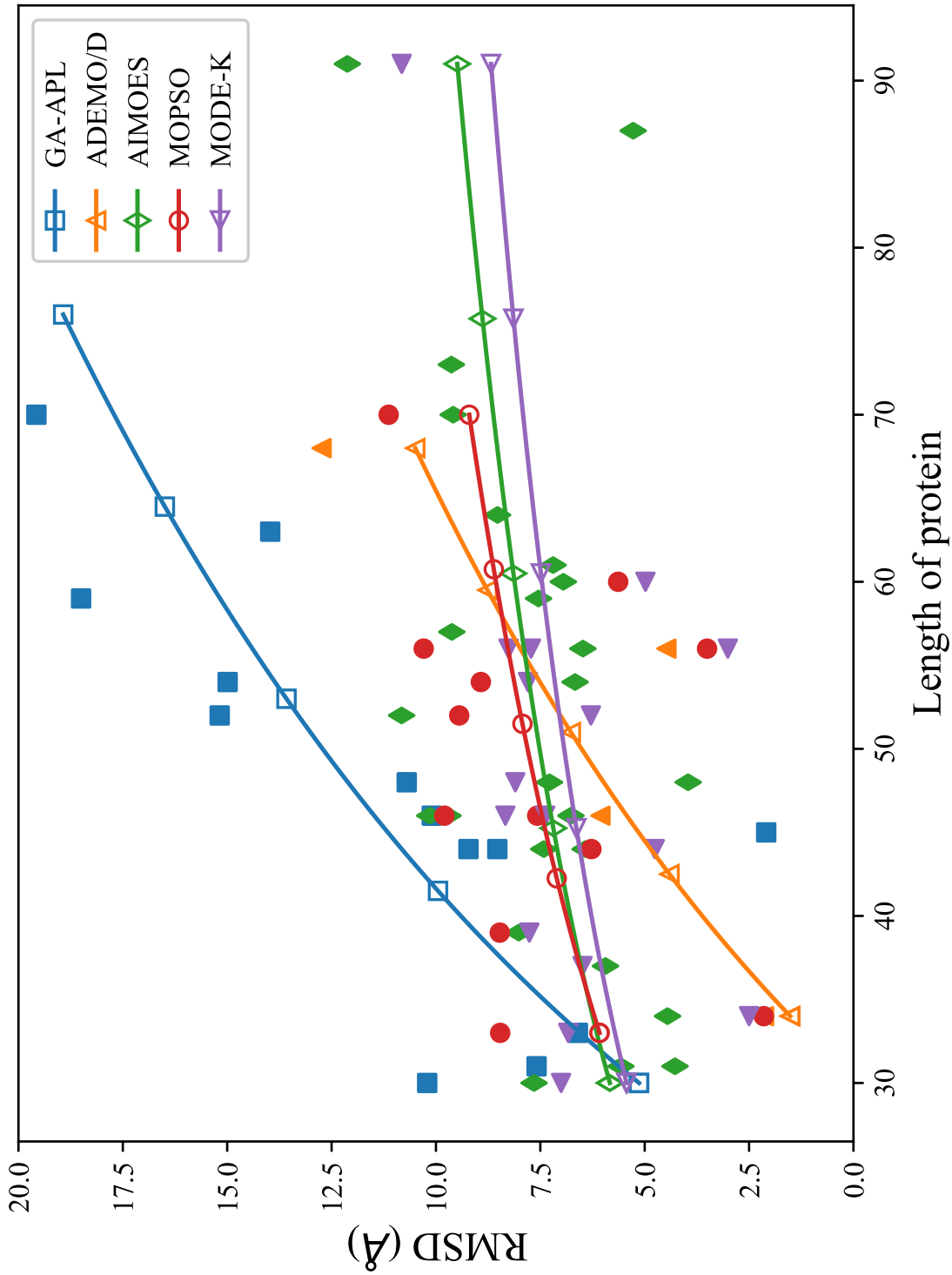


Figure 5.19: The qualitative comparison among different works. A solid point represents the predicted results reported in these works. The logarithmic regression lines indicated by hollow points are plotted according to these solid points.

Table 5.7: The comparison of different methods according to the prediction accuracy.

Method	Number of test proteins	Length range	Structural class	Average $RMSD_{100}$ (Å)
GA-APL	14	30∼76	α,β,α/β	18.72
ADEMO/D	4	34∼68	α,β	9.16
AIMOES	25	30∼91	α,β,α/β	11.76
MOPSO	12	33∼70	α,β,α/β	12.18
MODE-K	16	30∼91	α,β,α/β	11.12

5.9 Comparison with two state-of-the-art approaches

For further testing and verifying the performance of the proposed MODE-K approach, we also compare the prediction results obtained by MODE-K and two state-of-the-art approaches, i.e., QUARK and Rosetta, that are recognized as ones of the most top-performing FM approaches in the PSP area [107]. Table 5.8 summarizes the RMSD values of the predicted results gotten from the three approaches for all test proteins. From Table 5.8, we can see that there is no one approach which is always superior to the others on all test proteins. However, we should admit that the performance of MODE-K is worse than QUARK and Rosetta, especially for the larger test proteins (i.e., 1SXD, 3DF8, and 3NRW). MODE-K achieves better results than QUARK only on three test proteins (i.e., 2KDL, 2P6J, and 2P81), and achieves better results than Rosetta only on two proteins (i.e., 1K36 and 2P81). In addition, it can also be said that MODE-K can usually provide satisfactory solutions for several proteins (e.g., 1BDD, 1DFN, 1E0M, and 1ROP). Finally, we note that QUARK and Rosetta are two FM approaches based on the fragment-assembly technique [26, 12], where a predicted structure is built by integrating the fragments derived from existing protein structures. These fragment-assembly approaches is successful because of the sophisticated fragment generation and the well-designed folding strategy. Since the fragment-assembly technique is not incorporated into MODE-K, the proposed MODE-K approach is considered more straightforward and nearer to the *ab initio* prediction [94]. Taking advantage of this technique to enhance our approaches deserves our future investigation.

Table 5.8: Comparing MODE-K with two state-of-the-art approaches QUARK and Rosetta. Each cell contains the RMSD value (Å) of the predicted protein structure.

PDB ID	MODE-K	QUARK	Rosetta	PDB ID	MODE-K	QUARK	Rosetta
1AB1(46)	7.38	4.95	3.28	1SXD(91)	10.82	3.05	6.24
1BDD(60)	4.98	4.87	4.14	1ZDD(34)	2.50	0.93	1.18
1DFN(30)	7.00	6.65	6.56	2GB1(56)	8.26	2.31	1.12
1E0G(48)	8.10	2.29	2.48	2KDL(56)	7.72	11.51	5.70
1E0M(37)	6.49	4.80	5.26	2M7T(33)	6.83	3.89	4.45
1ENH(54)	7.80	1.78	3.24	2P6J(52)	6.29	14.46	3.92
1I6C(39)	7.76	4.97	5.21	2P81(44)	4.76	7.91	7.71
1K36(46)	8.34	3.15	8.82	3DF8(109)	16.71	4.89	9.18
1ROP(56)	3.01	1.60	2.23	3NRW(104)	11.85	1.79	1.36

Chapter 6

Conclusion

Despite the rapid development of computer techniques and the unremitting efforts of researchers, the protein structure prediction (PSP) problem remains challenging in bioinformatics and computational biology. An efficient search strategy and an effective energy function are the two pivotal factors for solving the protein structure prediction problem. In this study, we modeled the PSP problem as a MOOP and proposed an integrated FM approach called MODE-K to solve this problem. Considering the reality that KBEFs are usually more effective than PBEFs, we used the KBEF RWplus as the energy function. We decomposed RWplus into an orientation-dependent term and a distance-dependent energy term to develop the multiobjective energy function. Since DE algorithm is can be said one of the most powerful stochastic optimization techniques, we adopted DE algorithm as the search strategy and proposed a MODE algorithm to sample the conformation space. In addition, an external archive based on nondominated sorting was maintained to store the optimal solutions during evaluation. Finally,we introduce the clustering method MUFOLD-CL to select the final predicted structure from series of decoy structures.

The performance of the proposed MODE-K approach was verified by testing eighteen proteins. The experimental results and the comparison results demonstrated that MODE-K is effective in solving the PSP problem. Specifically, the energy function of the MODE-K approach is an two-objective KBEF. This handing way is considered novel because limited efforts are made to this aspect. In addtion, since the search strategy and the energy function were both improved in this study, a new point for

solving the PSP problem was given in this paper.

In future studies, we will continue to improve our multiobjective approaches to address the PSP problem. We insist that the adoption of multiobjective approaches is a high-performing alternative for solving this problem. In addition, the contact prediction technique [1, 14] has become more compelling in the field of PSP in recent years. Combining this technique with multiobjective approaches is worth future studies.

Bibliography

[1] Johannes Söding. Big-data approaches to protein structure prediction. *Science*, 355(6322):248–249, 2017.

[2] Fred E Cohen and Jeffery W Kelly. Therapeutic approaches to protein-misfolding diseases. *Nature*, 426(6968):905, 2003.

[3] Yang Zhang. Protein structure prediction: when is it useful? *Current opinion in structural biology*, 19(2):145–155, 2009.

[4] Xiaogen Zhou, Chun-Xiang Peng, Jun Liu, Yang Zhang, and Gui-jun Zhang. Underestimation-assisted global-local cooperative differential evolution and the application to protein structure prediction. *IEEE Transactions on Evolutionary Computation*, 2019.

[5] Ken A Dill and Justin L MacCallum. The protein-folding problem, 50 years on. *science*, 338(6110):1042–1046, 2012.

[6] Christian B. Anfinsen. Principles that govern the folding of protein chains. *Science*, 181(4096):223–230, 1973.

[7] John Moult, Krzysztof Fidelis, Andriy Kryshtafovych, Torsten Schwede, and Anna Tramontano. Critical assessment of methods of protein structure prediction (casp)—round xii. *Proteins: Structure, Function, and Bioinformatics*, 86:7–15, 2018.

[8] Stephen K Burley, Helen M Berman, Cole Christie, Jose M Duarte, Zukang Feng, John Westbrook, Jasmine Young, and Christine Zardecki. Rcsb protein

data bank: Sustaining a living digital data resource that enables breakthroughs in scientific research and biomedical education. *Protein Science*, 27(1):316–330, 2018.

[9] Ambrish Roy, Alper Kucukural, and Yang Zhang. I-tasser: a unified platform for automated protein structure and function prediction. *Nature protocols*, 5(4):725, 2010.

[10] Marco Biasini, Stefan Bienert, Andrew Waterhouse, Konstantin Arnold, Gabriel Studer, Tobias Schmidt, Florian Kiefer, Tiziano Gallo Cassarino, Martino Bertoni, Lorenza Bordoli, et al. Swiss-model: modelling protein tertiary and quaternary structure using evolutionary information. *Nucleic acids research*, 42(W1):W252–W258, 2014.

[11] Kristian W Kaufmann, Gordon H Lemmon, Samuel L DeLuca, Jonathan H Sheehan, and Jens Meiler. Practically useful: what the rosetta protein modeling suite can do for you. *Biochemistry*, 49(14):2987–2998, 2010.

[12] Dong Xu and Yang Zhang. Ab initio protein structure assembly using continuous structure fragments and optimized knowledge-based force field. *Proteins: Structure, Function, and Bioinformatics*, 80(7):1715–1735, 2012.

[13] R Evans, J Jumper, J Kirkpatrick, L Sifre, TFG Green, C Qin, A Zidek, A Nelson, A Bridgland, H Penedones, et al. De novo structure prediction with deeplearning based scoring. *Annu Rev Biochem*, 77:363–382, 2018.

[14] Joerg Schaarschmidt, Bohdan Monastyrskyy, Andriy Kryshtafovych, and Alexandre MJJ Bonvin. Assessment of contact predictions in casp12: Coevolution and deep learning coming of age. *Proteins: Structure, Function, and Bioinformatics*, 86:51–66, 2018.

[15] David Baker and Andrej Sali. Protein structure prediction and structural genomics. *Science*, 294(5540):93–96, 2001.

[16] Gang Xu, Tianqi Ma, Tianwu Zang, Weitao Sun, Qinghua Wang, and Jianpeng Ma. Opus-dosp: A distance-and orientation-dependent all-atom potential derived from side-chain packing. *Journal of molecular biology*, 429(20):3113–3120, 2017.

[17] Robert B Best, Jeetain Mittal, Michael Feig, and Alexander D MacKerell Jr. Inclusion of many-body effects in the additive charmm protein cmap potential results in enhanced cooperativity of α-helix and β-hairpin formation. *Biophysical journal*, 103(5):1045–1051, 2012.

[18] Devleena Shivakumar, Joshua Williams, Yujie Wu, Wolfgang Damm, John Shelley, and Woody Sherman. Prediction of absolute solvation free energies using molecular dynamics free energy perturbation and the opls force field. *Journal of chemical theory and computation*, 6(5):1509–1519, 2010.

[19] Yuedong Yang and Yaoqi Zhou. Specific interactions for ab initio folding of protein terminal regions with secondary structures. *Proteins: Structure, Function, and Bioinformatics*, 72(2):793–803, 2008.

[20] Hongyi Zhou and Jeffrey Skolnick. Goap: a generalized orientation-dependent, all-atom statistical potential for protein structure prediction. *Biophysical journal*, 101(8):2043–2052, 2011.

[21] Chao Zhang, George Vasmatzis, James L Cornette, and Charles DeLisi. Determination of atomic desolvation energies from the structures of crystallized proteins. *Journal of molecular biology*, 267(3):707–726, 1997.

[22] Hongyi Zhou and Yaoqi Zhou. Distance-scaled, finite ideal-gas reference state improves structure-derived potentials of mean force for structure selection and stability prediction. *Protein science*, 11(11):2714–2726, 2002.

[23] Mingyang Lu, Athanasios D Dousis, and Jianpeng Ma. Opus-psp: an orientation-dependent statistical all-atom potential derived from side-chain packing. *Journal of molecular biology*, 376(1):288–301, 2008.

[24] Alan M Poole and Rama Ranganathan. Knowledge-based potentials in protein design. *Current opinion in structural biology*, 16(4):508–513, 2006.

[25] Jian Zhang and Yang Zhang. A novel side-chain orientation dependent potential derived from random-walk reference state for protein fold selection and structure prediction. *PloS one*, 5(10):e15386, 2010.

[26] Carol A Rohl, Charlie EM Strauss, Kira MS Misura, and David Baker. Protein structure prediction using rosetta. In *Methods in enzymology*, volume 383, pages 66–93. Elsevier, 2004.

[27] Juyong Lee, Jinhyuk Lee, Takeshi N Sasaki, Masaki Sasai, Chaok Seok, and Jooyoung Lee. De novo protein structure prediction by dynamic fragment assembly and conformational space annealing. *Proteins: Structure, Function, and Bioinformatics*, 79(8):2403–2417, 2011.

[28] Jooyoung Lee, Peter L Freddolino, and Yang Zhang. Ab initio protein structure prediction. In *From protein structure to function with bioinformatics*, pages 3–35. Springer, 2017.

[29] Mario Garza-Fabre, Shaun M Kandathil, Julia Handl, Joshua Knowles, and Simon C Lovell. Generating, maintaining, and exploiting diversity in a memetic algorithm for protein structure prediction. *Evolutionary computation*, 24(4):577–607, 2016.

[30] Leonardo Correa, Bruno Borguesan, Camilo Farfán, Mario Inostroza-Ponta, and Márcio Dorn. A memetic algorithm for 3d protein structure prediction problem. *IEEE/ACM transactions on computational biology and bioinformatics*, 15(3):690–704, 2018.

[31] Seung Hwan Hong, InSuk Joung, Jose C Flores-Canales, Balachandran Manavalan, Qianyi Cheng, Seungryong Heo, Jong Yun Kim, Sun Young Lee, Mikyung Nam, Keehyoung Joo, et al. Protein structure modeling and refine-

ment by global optimization in casp12. *Proteins: Structure, Function, and Bioinformatics*, 86:122–135, 2018.

[32] Shangce Gao, Catherine Vairappan, Yan Wang, Qiping Cao, and Zheng Tang. Gravitational search algorithm combined with chaos for unconstrained numerical optimization. *Applied Mathematics and Computation*, 231:48–62, 2014.

[33] Shangce Gao, Yirui Wang, Jiahai Wang, and JiuJun Cheng. Understanding differential evolution: A poisson law derived from population interaction network. *Journal of computational science*, 21:140–149, 2017.

[34] Junkai Ji, Shuangbao Song, Cheng Tang, Shangce Gao, Zheng Tang, and Yuki Todo. An artificial bee colony algorithm search guided by scale-free networks. *Information Sciences*, 473:142–165, 2019.

[35] Shangce Gao, Yang Yu, Yirui Wang, Jiahai Wang, Jiujun Cheng, and MengChu Zhou. Chaotic local search-based differential evolution algorithms for optimization. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 2019.

[36] Shangce Gao, Yirui Wang, Jiujun Cheng, Yasuhiro Inazumi, and Zheng Tang. Ant colony optimization with clustering for solving the dynamic location routing problem. *Applied Mathematics and Computation*, 285:149–173, 2016.

[37] Zahid Halim and Tufail Muhammad. Quantifying and optimizing visualization: An evolutionary computing-based approach. *Information Sciences*, 385:284–313, 2017.

[38] Zhenyu Song, Yajiao Tang, Xingqian Chen, Shuangbao Song, Shuangyu Song, and Shangce Gao. A preference-based multi-objective evolutionary strategy for ab initio prediction of proteins. In *2017 International Conference on Progress in Informatics and Computing (PIC)*, pages 7–12. IEEE, 2017.

[39] Shangce Gao, Mengchu Zhou, Yirui Wang, Jiujun Cheng, Hanaki Yachi, and Jiahai Wang. Dendritic neuron model with effective learning algorithms for

classification, approximation, and prediction. *IEEE transactions on neural networks and learning systems*, 30(2):601–614, 2018.

[40] Junkai Ji, Shuangbao Song, Yajiao Tang, Shangce Gao, Zheng Tang, and Yuki Todo. Approximate logic neuron model trained by states of matter search algorithm. *Knowledge-Based Systems*, 163:120–130, 2019.

[41] Shuangyu Song, Xingqian Chen, Cheng Tang, Shuangbao Song, Zheng Tang, and Yuki Todo. Training an approximate logic dendritic neuron model using social learning particle swarm optimization algorithm. *IEEE Access*, 7:141947–141959, 2019.

[42] Frederico T Silva, Mateus X Silva, and Jadson C Belchior. A new genetic algorithm approach applied to atomic and molecular cluster studies. *Frontiers in chemistry*, 7, 2019.

[43] Rainer Storn and Kenneth Price. Differential evolution–a simple and efficient heuristic for global optimization over continuous spaces. *Journal of global optimization*, 11(4):341–359, 1997.

[44] Swagatam Das and Ponnuthurai Nagaratnam Suganthan. Differential evolution: A survey of the state-of-the-art. *IEEE transactions on evolutionary computation*, 15(1):4–31, 2011.

[45] Swagatam Das, Sankha Subhra Mullick, and Ponnuthurai N Suganthan. Recent advances in differential evolution–an updated survey. *Swarm and Evolutionary Computation*, 27:1–30, 2016.

[46] Rohan Mukherjee, Gyana Ranjan Patra, Rupam Kundu, and Swagatam Das. Cluster-based differential evolution with crowding archive for niching in dynamic environments. *Information Sciences*, 267:58–82, 2014.

[47] Xiao-gen Zhou, Gui-jun Zhang, Xiao-hu Hao, and Li Yu. A novel differential evolution algorithm using local abstract convex underestimate strategy for global optimization. *Computers & Operations Research*, 75:132–149, 2016.

[48] Mostafa Z Ali, Noor H Awad, Ponnuthurai Nagaratnam Suganthan, and Robert G Reynolds. An adaptive multipopulation differential evolution with dynamic population reduction. *IEEE transactions on cybernetics*, 47(9):2768–2779, 2016.

[49] Xiao-Gen Zhou and Gui-Jun Zhang. Differential evolution with underestimation-based multimutation strategy. *IEEE transactions on cybernetics*, 49(4):1353–1364, 2018.

[50] Pinar Civicioglu and Erkan Besdok. Bernstain-search differential evolution algorithm for numerical function optimization. *Expert Systems with Applications*, 138:112831, 2019.

[51] Yong Zhang, Dun-wei Gong, Xiao-zhi Gao, Tian Tian, and Xiao-yan Sun. Binary differential evolution with self-learning for multi-objective feature selection. *Information Sciences*, 507:67–85, 2020.

[52] Seyed Mohammad Seyedpoor and Mohammad Hossein Nopour. A two-step method for damage identification in moment frame connections using support vector machine and differential evolution algorithm. *Applied Soft Computing*, 88:106008, 2020.

[53] Christiane Regina Soares Brasil, Alexandre Claudio Botazzo Delbem, and Fernando Luís Barroso da Silva. Multiobjective evolutionary algorithm with many tables for purely ab initio protein structure prediction. *Journal of computational chemistry*, 34(20):1719–1734, 2013.

[54] Sandra M Venske, Richard A Gonçalves, Elaine M Benelli, and Myriam R Delgado. Ademo/d: an adaptive differential evolution for protein structure prediction problem. *Expert Systems with Applications*, 56:209–226, 2016.

[55] Gregório K Rocha, Karina B Dos Santos, Jaqueline S Angelo, Fabio L Custodio, Helio JC Barbosa, and Laurent E Dardenne. Inserting co-evolution information from contact maps into a multiobjective genetic algorithm for protein structure

prediction. In *2018 IEEE Congress on Evolutionary Computation (CEC)*, pages 1–8. IEEE, 2018.

[56] Shangce Gao, Shuangbao Song, Jiujun Cheng, Yuki Todo, and Mengchu Zhou. Incorporation of solvent effect into multi-objective evolutionary algorithm for improved protein structure prediction. *IEEE/ACM transactions on computational biology and bioinformatics*, 15(4):1365–1378, 2018.

[57] Shuangbao Song, Shangce Gao, Xingqian Chen, Dongbao Jia, Xiaoxiao Qian, and Yuki Todo. Aimoes: Archive information assisted multi-objective evolutionary strategy for ab initio protein structure prediction. *Knowledge-Based Systems*, 146:58–72, 2018.

[58] Shuangbao Song, Junkai Ji, Xingqian Chen, Shangce Gao, Zheng Tang, and Yuki Todo. Adoption of an improved pso to explore a compound multi-objective energy function in protein structure prediction. *Applied Soft Computing*, 72:539–551, 2018.

[59] Ahmed Bin Zaman and Amarda Shehu. Balancing multiple objectives in conformation sampling to control decoy diversity in template-free protein structure prediction. *BMC bioinformatics*, 20(1):211, 2019.

[60] Vincenzo Cutello, Giuseppe Narzisi, and Giuseppe Nicosia. A multi-objective evolutionary approach to the protein structure prediction problem. *Journal of The Royal Society Interface*, 3(6):139–151, 2006.

[61] Bruno Borguesan, Mariel Barbachan e Silva, Bruno Grisci, Mario Inostroza-Ponta, and Márcio Dorn. Apl: An angle probability list to improve knowledge-based metaheuristics for the three-dimensional protein structure prediction. *Computational biology and chemistry*, 59:142–157, 2015.

[62] Zhang Guijun, Ma Laifa, Wang Xiaoqi, and Zhou Xiaogen. Secondary structure and contact guided differential evolution for protein structure prediction.

IEEE/ACM Transactions on Computational Biology and Bioinformatics, pages 1–1, 2018.

[63] Vincenzo Cutello, Giuseppe Narzisi, and Giuseppe Nicosia. Computational studies of peptide and protein structure prediction problems via multiobjective evolutionary algorithms. In *Multiobjective problem solving from nature*, pages 93–114. Springer, 2008.

[64] Daniel WA Buchan, Federico Minneci, Tim CO Nugent, Kevin Bryson, and David T Jones. Scalable web services for the psipred protein analysis workbench. *Nucleic acids research*, 41(W1):W349–W357, 2013.

[65] Cyrus Ahmadi Toussi and Javad Haddadnia. Improving protein secondary structure prediction: the evolutionary optimized classification algorithms. *Structural Chemistry*, 30(4):1257–1266, Aug 2019.

[66] Grzegorz Rozenberg, Thomas Bäck, and Joost N Kok. *Handbook of natural computing*. Springer, 2012.

[67] Javier [Del Ser], Eneko Osaba, Daniel Molina, Xin-She Yang, Sancho Salcedo-Sanz, David Camacho, Swagatam Das, Ponnuthurai N. Suganthan, Carlos A. [Coello Coello], and Francisco Herrera. Bio-inspired computation: Where we stand and what's next. *Swarm and Evolutionary Computation*, 48:220 – 250, 2019.

[68] Rudolf Kruse, Christian Borgelt, Christian Braune, Sanaz Mostaghim, and Matthias Steinbrecher. *Computational intelligence: a methodological introduction*. Springer, 2016.

[69] Tianle Zhou, Chaoyi Chu, Shuangbao Song, Yirui Wang, and Shangce Gao. A dendritic neuron model for exchange rate prediction. In *2015 IEEE International Conference on Progress in Informatics and Computing (PIC)*, pages 10–14, Dec 2015.

[70] Ying Yu, Shuangbao Song, Tianle Zhou, Hanaki Yachi, and Shangce Gao. Forecasting house price index of china using dendritic neuron model. In *2016 International Conference on Progress in Informatics and Computing (PIC)*, pages 37–41, 2016.

[71] Junkai Ji, Shangce Gao, Jiujun Cheng, Zheng Tang, and Yuki Todo. An approximate logic neuron model with a dendritic structure. *Neurocomputing*, 173:1775 – 1783, 2016.

[72] Seyed Mohammad Mirjalili, Jin Song Dong, Ali Safa Sadiq, and Hossam Faris. Genetic algorithm: Theory, literature review, and application in image reconstruction. In *Nature-Inspired Optimizers*, 2019.

[73] Zhida Deng, Mihai D. Rotaru, and Jan K. Sykulski. Kriging assisted surrogate evolutionary computation to solve optimal power flow problems. *IEEE Transactions on Power Systems*, 35(2):831–839, 2020.

[74] Farid Ghareh Mohammadi, M. Hadi Amini, and Hamid R. Arabnia. *Evolutionary Computation, Optimization, and Learning Algorithms for Data Science*, pages 37–65. Springer International Publishing, Cham, 2020.

[75] Eckart Zitzler, Lothar Thiele, Marco Laumanns, Carlos M Fonseca, and Viviane Grunert Da Fonseca. Performance assessment of multiobjective optimizers: An analysis and review. *IEEE Transactions on evolutionary computation*, 7(2):117–132, 2003.

[76] Qingfu Zhang and Hui Li. Moea/d: A multiobjective evolutionary algorithm based on decomposition. *IEEE Transactions on evolutionary computation*, 11(6):712–731, 2007.

[77] Kalyanmoy Deb, Amrit Pratap, Sameer Agarwal, and TAMT Meyarivan. A fast and elitist multiobjective genetic algorithm: Nsga-ii. *IEEE transactions on evolutionary computation*, 6(2):182–197, 2002.

[78] Carlos A Coello Coello, Gregorio Toscano Pulido, and M Salazar Lechuga. Handling multiple objectives with particle swarm optimization. *IEEE Transactions on evolutionary computation*, 8(3):256–279, 2004.

[79] Yousef Abdi and Mohammad-Reza Feizi-Derakhshi. Hybrid multi-objective evolutionary algorithm based on search manager framework for big data optimization problems. *Applied Soft Computing*, 87:105991, 2020.

[80] Ruochen Liu, Runan Zhou, Rui Ren, Jiangdi Liu, and Licheng Jiao. Multi-layer interaction preference based multi-objective evolutionary algorithm through decomposition. *Information Sciences*, 509:420–436, 2020.

[81] Fei Zou, Gary G Yen, and Lixin Tang. A knee-guided prediction approach for dynamic multi-objective optimization. *Information Sciences*, 509:193–209, 2020.

[82] Dunwei Gong, Yiping Liu, and Gary G Yen. A meta-objective approach for many-objective evolutionary optimization. *Evolutionary computation*, 28(1):1–25, 2020.

[83] Seyedali Mirjalili, Shahrzad Saremi, Seyed Mohammad Mirjalili, and Leandro dos S Coelho. Multi-objective grey wolf optimizer: a novel algorithm for multi-criterion optimization. *Expert Systems with Applications*, 47:106–119, 2016.

[84] Anupam Trivedi, Dipti Srinivasan, Krishnendu Sanyal, and Abhiroop Ghosh. A survey of multiobjective evolutionary algorithms based on decomposition. *IEEE Transactions on Evolutionary Computation*, 21(3):440–462, 2016.

[85] Ye Tian, Ran Cheng, Xingyi Zhang, Fan Cheng, and Yaochu Jin. An indicator-based multiobjective evolutionary algorithm with reference point adaptation for better versatility. *IEEE Transactions on Evolutionary Computation*, 22(4):609–622, 2017.

[86] Aimin Zhou, Bo-Yang Qu, Hui Li, Shi-Zheng Zhao, Ponnuthurai Nagaratnam Suganthan, and Qingfu Zhang. Multiobjective evolutionary algorithms: A sur-

vey of the state of the art. *Swarm and Evolutionary Computation*, 1(1):32–49, 2011.

[87] Min-yi Shen and Andrej Sali. Statistical potential for assessment and prediction of protein structures. *Protein science*, 15(11):2507–2524, 2006.

[88] Yuedong Yang and Yaoqi Zhou. Ab initio folding of terminal segments with secondary structures reveals the fine difference between two closely related all-atom statistical energy functions. *Protein science*, 17(7):1212–1219, 2008.

[89] Yang Zhang, Haijun Zhou, and Zhong-Can Ou-Yang. Stretching single-stranded dna: interplay of electrostatic, base-pairing, and base-pair stacking interactions. *Biophysical journal*, 81(2):1133–1143, 2001.

[90] Matthew J O'Meara, Andrew Leaver-Fay, Michael D Tyka, Amelie Stein, Kevin Houlihan, Frank DiMaio, Philip Bradley, Tanja Kortemme, David Baker, Jack Snoeyink, et al. Combined covalent-electrostatic model of hydrogen bonding improves structure prediction with rosetta. *Journal of chemical theory and computation*, 11(2):609–622, 2015.

[91] Xiaogen Zhou, Jun Hu, Chengxin Zhang, Guijun Zhang, and Yang Zhang. Assembling multidomain protein structures through analogous global structural alignments. *Proceedings of the National Academy of Sciences*, 116(32):15930–15938, 2019.

[92] Jingqiao Zhang and Arthur C Sanderson. Jade: adaptive differential evolution with optional external archive. *IEEE Transactions on evolutionary computation*, 13(5):945–958, 2009.

[93] A Kai Qin, Vicky Ling Huang, and Ponnuthurai N Suganthan. Differential evolution algorithm with strategy adaptation for global numerical optimization. *IEEE transactions on Evolutionary Computation*, 13(2):398–417, 2009.

[94] Glennie Helles. A comparative study of the reported performance of ab initio protein structure prediction algorithms. *Journal of the royal society interface*, 5(21):387–396, 2007.

[95] Pierrick Craveur, Agnel Praveen Joseph, Pierre Poulain, Alexandre G de Brevern, and Joseph Rebehmed. Cis–trans isomerization of omega dihedrals in proteins. *Amino acids*, 45(2):279–289, 2013.

[96] Roland L Dunbrack Jr. Rotamer libraries in the 21st century. *Current opinion in structural biology*, 12(4):431–440, 2002.

[97] Andriy Kryshtafovych, Bohdan Monastyrskyy, Krzysztof Fidelis, Torsten Schwede, and Anna Tramontano. Assessment of model accuracy estimations in casp12. *Proteins: Structure, Function, and Bioinformatics*, 86:345–360, 2018.

[98] Zahid Halim et al. Optimizing the minimum spanning tree-based extracted clusters using evolution strategy. *Cluster Computing*, 21(1):377–391, 2018.

[99] Yang Zhang and Jeffrey Skolnick. Spicker: a clustering approach to identify near-native protein folds. *Journal of computational chemistry*, 25(6):865–871, 2004.

[100] Jingfen Zhang and Dong Xu. Fast algorithm for population-based protein structural model analysis. *Proteomics*, 13(2):221–229, 2013.

[101] C. A. Coello Coello. Evolutionary multi-objective optimization: a historical view of the field. *IEEE Computational Intelligence Magazine*, 1(1):28–36, Feb 2006.

[102] Yang Zhang and Jeffrey Skolnick. Scoring function for automated assessment of protein structure template quality. *Proteins: Structure, Function, and Bioinformatics*, 57(4):702–710, 2004.

[103] Adam Zemla. Lga: a method for finding 3d similarities in protein structures. *Nucleic acids research*, 31(13):3370–3374, 2003.

[104] Kevin Molloy, Sameh Saleh, and Amarda Shehu. Probabilistic search and energy guidance for biased decoy sampling in ab initio protein structure prediction. *IEEE/ACM Transactions on Computational Biology and Bioinformatics (TCBB)*, 10(5):1162–1175, 2013.

[105] Leonardo de Lima Corrêa, Bruno Borguesan, Mathias J Krause, and Márcio Dorn. Three-dimensional protein structure prediction based on memetic algorithms. *Computers & Operations Research*, 91:160–177, 2018.

[106] Oliviero Carugo and Sándor Pongor. A normalized root-mean-spuare distance for comparing protein three-dimensional structures. *Protein science*, 10(7):1470–1473, 2001.

[107] Luciano A Abriata, Giorgio E Tamò, Bohdan Monastyrskyy, Andriy Kryshtafovych, and Matteo Dal Peraro. Assessment of hard target modeling in casp12 reveals an emerging role of alignment-based contact prediction methods. *Proteins: Structure, Function, and Bioinformatics*, 86:97–112, 2018.