Summer 2018

New Methods to Improve Protein Structure Modeling

Maha Abdelrasoul
Old Dominion University

NEW METHODS TO IMPROVE PROTEIN STRUCTURE MODELING

by

Maha Abdelrasoul
B.S. June 2006, Arab Academy for Science and Technology, Egypt
M.S. December 2011, Arab Academy for Science and Technology, Egypt

A Dissertation Submitted to the Faculty of
Old Dominion University in Partial Fulfillment of the
Requirements for the Degree of

DOCTOR OF PHILOSOPHY

COMPUTER SCIENCE

OLD DOMINION UNIVERSITY
August 2018

Approved by:

Yaohang Li (Director)

Ravi Mukkamala (Member)

Michele C. Weigle (Member)

Steven Pascal (Member)

ABSTRACT

NEW METHODS TO IMPROVE PROTEIN STRUCTURE MODELING

Maha Abdelrasoul
Old Dominion University, 2018
Director: Dr. Yaohang Li

Proteins are considered the central compound necessary for life, as they play a crucial role in governing several life processes by performing the most essential biological and chemical functions in every living cell. Understanding protein structures and functions will lead to a significant advance in life science and biology. Such knowledge is vital for various fields such as drug development and synthetic biofuels production.

Most proteins have definite shapes that they fold into, which are the most stable state they can adopt. Due to the fact that the protein structure information provides important insight into its functions, many research efforts have been conducted to determine the protein 3-dimensional structure from its sequence.

The experimental methods for protein 3-dimensional structure determination are often time-consuming, costly, and even not feasible for some proteins. Accordingly, recent research efforts focus more and more on computational approaches to predict protein 3-dimensional structures. Template-based modeling is considered one of the most accurate protein structure prediction methods. The success of template-based modeling relies on correctly identifying one or a few experimentally determined protein structures as structural templates that are likely to resemble the structure of the target sequence as well as accurately producing a sequence alignment that maps the residues in the target sequence to those in the template.

In this work, we aim at improving the template-based protein structure modeling by enhancing the correctness of identifying the most appropriate templates and precisely aligning the target and template sequences. Firstly, we investigate employing inter-residue contact score to

measure the favorability of a target sequence fitting in the folding topology of a certain template. Secondly, we design a multi-objective alignment algorithm extending the famous Needleman-Wunsch algorithm to obtain a complete set of alignments yielding Pareto optimality. Then, we use protein sequence and structural information as objectives and generate the complete Pareto optimal front of alignments between target sequence and template. The alignments obtained enable one to analyze the trade-offs between the potentially conflicting objectives. These approaches lead to accuracy enhancement in template-based protein structure modeling.

This dissertation is dedicated to my family,

for their endless love, support, and encouragement.

ACKNOWLEDGMENTS

Foremost, I would like to express my sincere gratitude to my advisor Dr. Yaohang Li for the continuous support of my Ph.D study and research, for his patience, motivation, enthusiasm, and immense knowledge. His guidance helped me through the time of my research and writing of this dissertation. I could not have imagined having a better advisor and mentor for my Ph.D study.

Beside my advisor, I would like to thank my committee members: Dr. Ravi Mukkamala, Dr. Michele C. Weigle, and Dr. Steven Pascal, for their encouragement, insightful comments, and hard questions. I gratefully acknowledge the generous support of Old Dominion University Dominion scholarship on this research

Also I want to thank my friends in Old Dominion University, I am grateful for their companionship. I want to express gratitude to Dr. Kurt Maly from Old Dominion University, and Dr. Tamer Nadeem from Virginia Commonwealth University, for their constant support and encouragement throughout my Ph.D. years.

I would like to offer special gratitude to Dr. Hussein Abdel-Wahab, who, although no longer with us, continues to inspire by his example and dedication to the students he served over the course of his career.

Finally, I am especially grateful to my family. Regardless of my successes or failures, they always stand by me, support me, and love me.

TABLE OF CONTENTS

Page

LIST OF TABLES xi

LIST OF FIGURES xii

Chapter

1 INTRODUCTION 1

1.1 STATEMENT OF THE PROBLEM 1

1.2 CONTRIBUTIONS OF THIS DISSERTATION 5

1.3 DISSERTATION ORGANIZATION 8

2 BACKGROUND 9

2.1 PROTEINS 9

2.1.1 Amino Acid 9

2.2 PROTEIN STRUCTURE 12

2.2.1 Primary Structure 13

2.2.2 Secondary Structure 14

2.2.2.1 α-Helix 14

2.2.2.2 β-Sheet 15

2.2.3 Tertiary Structure 16

2.2.4 Quaternary Structure 17

2.3 PROTEIN STRUCTURE MODELING 18

2.3.1 X-Ray Crystallography 18

Page

2.3.2 Nuclear Magnetic Resonance (NMR) 19
2.3.3 Cryo-Electron Microscopy 19
2.4 PROTEIN STRUCTURE PREDICTION 19
2.4.1 Ab-initio 20
2.4.2 Comparative Modeling 21
3 LITERATURE REVIEW 23
3.1 TEMPLATE-BASED PROTEIN STRUCTURE MODELING 23
3.1.1 Threading 26
3.1.2 Template Selection 29
3.1.2.1 Sequence-based Methods 30
3.1.2.2 Evolutionary Methods 31
3.1.2.3 Structural Methods 31
3.1.2.4 Knowledge-based Methods 32
3.2 PROTEIN SEQUENCE ALIGNMENT 32
3.2.1 Pairwise Alignment 34
3.2.1.1 Dot Matrix Method 35
3.2.1.2 Dynamic Programming 38
3.2.1.3 Word Methods 45
3.3 MULTI-OBJECTIVE ALIGNMENT 46
3.3.1 Multi-Objective Optimization 46
3.3.1.1 Pareto Optimality 47
3.3.2 Multi-objective Protein Sequence Alignment 48

Page

3.3.2.1 Protein Sequence Alignment objectives 49
3.4 SUMMARY 54
4 TEMPLATE SELECTION APPROACHES 55
4.1 INCORPORATING ICOSA SCORE IN TEMPLATE SELECTION 55
4.1.1 Methodology 56
4.1.1.1 MUSTER Scores 56
4.1.1.2 ICOSA Score of a Structural Template 56
4.1.2 Results 59
4.2 INCORPORATING ICOSA IN SEQUENCE ALIGNMENT 63
4.2.1 Methodology 64
4.2.1.1 Scoring Function 64
4.2.1.2 Alignment Generation 66
4.2.1.3 Parameter Training 66
4.2.2 Results 67
4.3 SUMMARY 74
5 MULTI-OBJECTIVE PROTEIN SEQUENCE ALIGNMENT 76
5.1 MULTI-OBJECTIVE ALIGNMENT (MOA) ALGORITHM 76
5.1.1 Score Matrices Generation 77
5.1.2 Backtracking the Objective Matrices 77
5.1.3 Results 82
5.2 A MULTI-OBJECTIVE NEEDLEMAN-WUNSCH ALGORITHM (MON) 94
5.2.1 Generation of Multi-Objective Score Matrix 95

Page

5.2.2 Backtracking the Pareto-optimal Alignments 100

5.2.3 Time and Space Complexity of Multi-Objective Alignment 106

5.2.4 Multi-Objective Alignment vs. Alignment by Optimizing a Weighted-Sum Consensus Function 106

5.2.5 Multi-Objective Needleman-Wunsch Alignment vs. Multi-Objective Genetic Algorithms 108

5.2.6 Multi-Objective Needleman-Wunsch with Affine Gap 109

5.2.7 Results 112

5.3 SUMMARY 125

6 CONCLUSION AND FUTURE WORK 127

6.1 SUMMARY 127

6.2 FUTURE WORK 129

7 REFERENCES 131

VITA 155

LIST OF TABLES

Table Page

1. Twenty standard amino acids and their abbreviation 11

2. The Correspondence of PAM Numbers with the observed percent of amino acid evolutionary distance 50

3. Overall performance of MUSTER, ICOSA, and MUSTER+ICOSAon the CASP11 targets .. 60

4. Overall performance of MUSTER, MUSTER+ICOSA, and SAICOSA on the CASP11 targets 68

5. All alignments generated using MOA 80

6. Non-dominated alignments 81

7. Overall performance of MUSTER and MOA on the top-ranked template specified by Muster for the CASP11 targets. 85

8. Overall performance of GenTHREADER and MOA on the top-ranked template specified by GenTHREADER for the CASP11 targets 88

9. Overall performance of MUSTER , linear combination objectives and MOA on the top-ranked template specified by CASP for the CASP11 targets. 88

10. Overall performance of MUSTER and MON on the top-ranked template specified by Muster for the CASP11 targets. 113

11. Overall performance of GenTHREADER and MON on the top-ranked template specified by GenTHREADER for the CASP11 targets 113

12. Overall performance of MUSTER , linear combination objectives and MON on the top-ranked template specified by CASP for the CASP11 targets. 116

LIST OF FIGURES

Figure Page

1. Protein Structure Modelling is the determination of the protein three-dimensional structure from its sequence information (the sequence and structure information are for 3BB5 [2]) 2

2. Number of protein sequences and structures available each year. Blue bar denotes the number of protein structures in PDB, orange bar is the number of protein sequences in SWISS-PROT [6] 3

3. Block diagram of the protein template-based modeling steps 4

4. The General Structure of an amino acid. 10

5. Peptide bond Formation 10

6. The four levels of protein structure (source [17]) 13

7. Primary structure of chain A of human insulin protein (1MSO) 14

8. α-Helix from 1amb protein where the hydrogen bond is shown as blue lines and the side chain atoms stem out of the helix 15

9. β-Sheet from chain A of 4erh protein where the hydrogen bond is shown as blue lines and the side chain atoms extend above and below the sheet plane 16

10. Tertiary Structure of 4erh protein chain A. 17

11. Quaternary structure of protein 4erh, and it involves of two polypeptide chains. 18

12. A sequence alignment, between 1F4I (chain A) and 1IFY(chain A). (a) Alignment produced by Chimera program [111], where the highlight represents matching regions. (b) The superimposition of the two structures based on the generated alignment, where 1F4I is blue and its matching regions are cyan, and 1IFY is red and its matched regions are orange. ... 33

Figure Page

13. The three zones of protein sequence alignments. A safe zone where homologous relationship is confident. Sequence identity values below the safe zone boundary, but above 20%, are considered to be in the twilight zone, where homologous relationships are less certain. The region below 20% is the midnight zone, where homologous relationships cannot be reliably determined. (Source: Modified from [112]). 34

14. An example of pairwise sequence comparison showing the distinction between global and local alignment. The global alignment (a) includes all residues of both sequences. The local alignment (b) only includes portions of the two sequences that have the highest regional similarity. 35

15. An example of dot plot method for aligning two sequences, where the dots in diagonal line indicate sequence alignment. The diagonal line below the main diagonal represent internal repeats of either sequence. 36

16. Generating the optimal alignment between sequences A= CTTAACT and B=CGGATCAT using Needleman-Wunsch algorithm; (a) Initializing the scoring matrix;(b) and (c) Computing the scoring matrix; (d) Back-tracing the scoring matrix to generate the optimal alignment. 43

17. Generating the optimal local alignment between sequences A= CTTAACT and B=CGGATCAT using Smith-Waterman algorithm; (a) Computing the scoring matrix; (b) Back-tracing the scoring matrix to generate the local alignment 45

18. Hypothetical trade-off solutions for a car buying decision-making problem (modified from [150]) 47

Figure Page

19. Pareto-optimal front solutions for four combinations of two types of objectives (a) the task is to maximize f1 and minimize f2,(b) the task is to minimize f1 and maximize f2, (c)the task is to minimize both f1 and f1, and (d) the task is to maximize both f1 and f2(modified from [151]). 48

20. PAM250 amino acid substitution matrix 51

21. BLOSUM62 amino acid substitution matrix 52

22. Icosahedral local coordinates with CA at the origin [14]. 57

23. Estimation of ICOSA score for a template alignment, (a) Structural Template of 1r43A, (b) Alignment between 1r43A and target sequences based on structural profile, (c) Substitute template residues (blue) with target residues (orange), (d) Calculating template ICOSA score of substituted 58

24. The GDT-TS score of the top-ranked models selected by MUSTER, ICOSA, and MUSTER+ICOSA in CASP11 targets. 61

25. Top-ranked templates selected by MUSTER, ICOSA, and MUSTER+ICOSA (red) in CASP11 target T0769(green),(a) top-rank template by MUSTER score,(b) top-rank template by ICOSA score, and (c) top-rank template by MUSTER+ICOSA score. 62

26. Top-ranked templates selected by MUSTER, ICOSA, and MUSTER+ICOSA (red) in CASP11 target T0773 (green), (a) top-ranked template by MUSTER score, (b) top-ranked template by ICOSA score, and (c) top-ranked template by MUSTER+ICOSA scores 62

27. The GDT-TS score of the top-ranked models selected by MUSTER, and MUSTER+ICOSA compared to the GDT-TS score of the top-ranked models generated using SAICOSA in CASP11 targets. 69

Figure Page

28. Top-ranked templates selected by MUSTER, MUSTER+ICOSA, and SAICOSA (red) in CASP11 target T0790 (green), (a) top-ranked template by MUSTER score, (b) top-ranked template MUSTER+ICOSA scores, and (c) top-ranked template by SAICOSA. 71

29. Top-ranked templates selected by MUSTER, MUSTER+ICOSA, and SAICOSA (red) in CASP11 target T0766 (green), (a) top-ranked template by MUSTER score, (b) top-ranked template MUSTER+ICOSA scores, and (c) top-ranked template by SAICOSA. 72

30. Top-ranked templates selected by MUSTER, MUSTER+ICOSA, and SAICOSA (red) in CASP11 target T0821 (green), (a) top-ranked template by MUSTER score, (b) top-ranked template MUSTER+ICOSA scores, and (c) top-ranked template by SAICOSA. 73

31. (a) The Needleman-Wunsch alignment matrix based on the profile with the maximum-match path traced to generate the optimal alignment. (b) The Needleman-Wunsch alignment matrix based on the secondary structure with the maximum-match path traced to generate the optimal alignment. (c), & (d) The optimal profile alignment and the optimal secondary structure alignment respectively. (e), & (f) The Needleman-Wunsch alignment matrix based on the profile and the secondary structure respectively with the maximum-match path traced along with the splits due to disagreement of the other matrix, where the decisions taken based on the profile are marked on black and the ones based on the secondary structure are marked on red. 79

32. Scores of the alignments generated by MOA where the red ones represent the dominated alignments and the blue ones represent the non-dominated alignments 81

Figure Page

33. Generation of an alignment matrix for two sequences according to the combination between secondary structure score and solvent accessibility score, (a)the two sequences (b)Secondary structure substitution matrix, (c) Solvent accessibility substitution matrix, (d) the combined substitution matrix, (e) the Needleman-Wunsch alignment matrix based on the combined substitution matrix..... 84

34. The GDT-TS score of Muster alignment and MOA alignment to CASP 11 targets with the top-ranked template selected by Muster. MOA achieved a higher or equal GDT-TS score for 102 targets and most of the time MOA seven of them the difference is more than 10 i.e. T0773-D1..... 86

35. The GDT-TS score of pGenTHREADER alignment and MOA alignment to CASP 11 targets with the top-ranked template selected by pGenTHREADER. In 83 targets MOA GDT-TS score is higher or equal pGenTHREADER, 17 of them MOA GDT-TS score was 10 points higher than pGenTHREADER. i.e. T0840 87

36. The GDT-TS score of Muster alignment and MOA alignment to CASP 11 targets with the top-ranked template selected by CASP. In 93 targets MOA GDT-TS score is higher or equal Muster, 4 of them MOA GDT-TS score was 10 points higher than Muster. i.e. T0769-D1. Muster achieved highly in 3 targets i.e T0782-D1..... 89

Figure Page

37. The GDT-TS score of linear combination of objectives algorithm using same sequence and structure information and MOA for CASP 11 targets with the top-ranked template selected by CASP. In 113 targets MOA achieved higher or equal GDT-TS, most of them MOA GDT-TS score was 10 points higher. i.e. T0759-D1. Only at T0776-D1 MOA was lower and by a very small difference. 90

38. Results for T0766-D1 alignment with 4orlA 92

39. The best scoring alignments generated by MOA and that generated by linear combination for T0766-D1 and 4orlA. The model generated from MOA alignment scores 93.75 GDT-TS while linear combination scores only 88.889 92

40. Results for T0769-D1 alignment with 3ramD 93

41. The best scoring alignment generated by MOA and that generated by linear combination algorithm for T0769-D1 and 3ramD. The model generated from MOA alignment scores 54.639 GDT-TS while linear combination scores only 10.052. 94

42. Generation of Fm, n from three neighboring cells $Fm-1, n$, $Fm-1, n-1$, and $Fm, n-1$ 97

43. The alignment scoring matrix in nucleotide-nucleotide BLAST [185] (blastn). 101

44. The alignment scoring matrix in K80 model 102

45. The four alignments at the Pareto-optimal front. 103

46. The multi-objective score matrix F for the two DNA sequences X=GGCCTACCAT, and Y=AAAGAGATT, where the objectives are the blastn and K80 model. 104

Figure Page

47. Backtracking the Pareto-optimal alignments for the two DNA sequences X=GGCCTACCAT, and Y=AAAGAGATT, where the objectives are the Blastn and K80 model4. Discussion 105

48. Linear weight combinations of objectives fails to find some Pareto optimal solutions .. 108

49. The GDT-TS score of Muster alignment and MON alignment to CASP 11 targets with the top-ranked template selected by Muster. MON achieved a higher or equal GDT-TS score for 104 targets and most of the time MON eight of them the difference is more than 10 i.e. T0773-D1 114

50. The GDT-TS score of pGenTHREADER alignment and MON alignment to CASP 11 targets with the top-ranked template selected by pGenTHREADER. In 84 targets MON GDT-TS score is higher or equal pGenTHREADER, 16 of them MON GDT-TS score was 10 points higher than pGenTHREADER. 115

51. The GDT-TS score of Muster alignment and MON alignment to CASP 11 targets with the top-ranked template selected by CASP. In 95 targets MON GDT-TS score is higher or equal Muster, 10 of them MON GDT-TS score was 10 points higher than Muster. i.e. T0769-D1. Muster achieved highly in 4 targets i.e T0782-D1 117

52. The GDT-TS score of linear combination of objectives algorithm using same sequence and structure information and MON for CASP 11 targets with the top-ranked template selected by CASP. In 113 targets MON achieved higher or equal GDT-TS, most of them MON GDT-TS score was 10 points higher. i.e. T0759-D1. Only at T0776-D1 MON was lower and by a very small difference. 118

Figure Page

53. The best scoring alignments generated from MON and that generated by Muster for T0769-D1 and 3ramD. The model generated from MON alignment scores 40.0 GDT-TS while Muster scores only 20.9 120

54. The best scoring alignments generated from MON and that generated by Muster for T0796-D1 and 2d42A. The model generated from MON alignment scores 52.3 GDT-TS while Muster scores only 36.9 121

55. Results for T0759-D1 alignment with 1lm5B 123

56. The best scoring alignments generated from MON and that generated by linear combination of objectives for T0759-D1 and 1lm5B1. The model generated from MON alignment scores 97.79 GDT-TS while linear combination of objectives scores only 27.21 123

57. Results for T0773-D1 alignment with 3opkA 124

58. The best scoring alignments generated from MON and that generated by linear combination of objectives for T0773-D1 and 3opkA. The model generated from MON alignment scores 69.4 GDT-TS while linear combination of objectives scores only 11.94 125

CHAPTER I

INTRODUCTION

1.1 Statement of the problem

One of the most important biological substances, which is considered the central compound necessary for life, is protein. Proteins play a crucial role in governing several life processes by performing the most essential biological and chemical functions in every living cell. Proteins form skin, muscles, antibodies, and enzymes. Even some hormones are proteins. They play the main role in digestion, respiration, and vision. As a matter of fact, the word “protein” is translated from the Greek root word meaning “primary.”

Proteins are made from amino acids bonded together in long chains. Proteins vary based on the number and type of amino acids in the protein chain. There are 20 different amino acids, each with a different chemical structure and characteristics. The protein structure relies on the amino acids that construct it. Consequently, the protein function is determined by the protein structure. Understanding protein structure and function leads to a significant advance in life sciences and biology. Such knowledge is vital for various fields such as the development of drugs and synthetic biofuels production.

In nature, the protein amino acid chain does not stretch out in a straight line; rather it folds into a unique three-dimensional structure [1]. This structure is critical to the protein biological function. Due to the fact that protein structure information provides insights to its function, many research efforts have been conducted to determine the protein 3-dimensional structure from its sequence information.

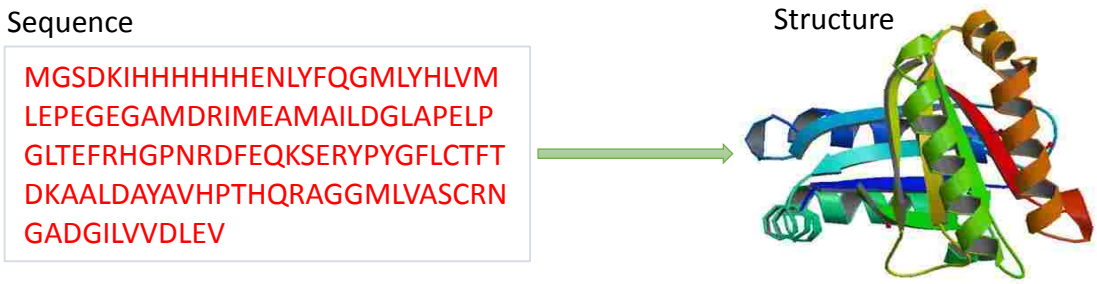


Fig. 1. Protein Structure Modelling is the determination of the protein three-dimensional structure from its sequence information (the sequence and structure information are for 3BB5 [2])

The determination of a protein 3-dimensional structure from its amino acid sequence is known as protein structure modeling (Fig. 1). There are three experimental methods for determining protein structures: X-ray crystallography, NMR, and Cryo-electron microscopy. These methods are often time-consuming, costly, and not feasible for some proteins. Also, these techniques are low-throughput in nature because of the huge experimental and human efforts that are needed to study a single protein [3] [4] [5]. For these reasons, the capacity to produce sequence information is extremely higher than that of producing structural information. Accordingly, computational approaches to accurately predict protein 3-dimensional structures are highly desired. Fig. 2, shows the number of protein sequences and structures available each year.

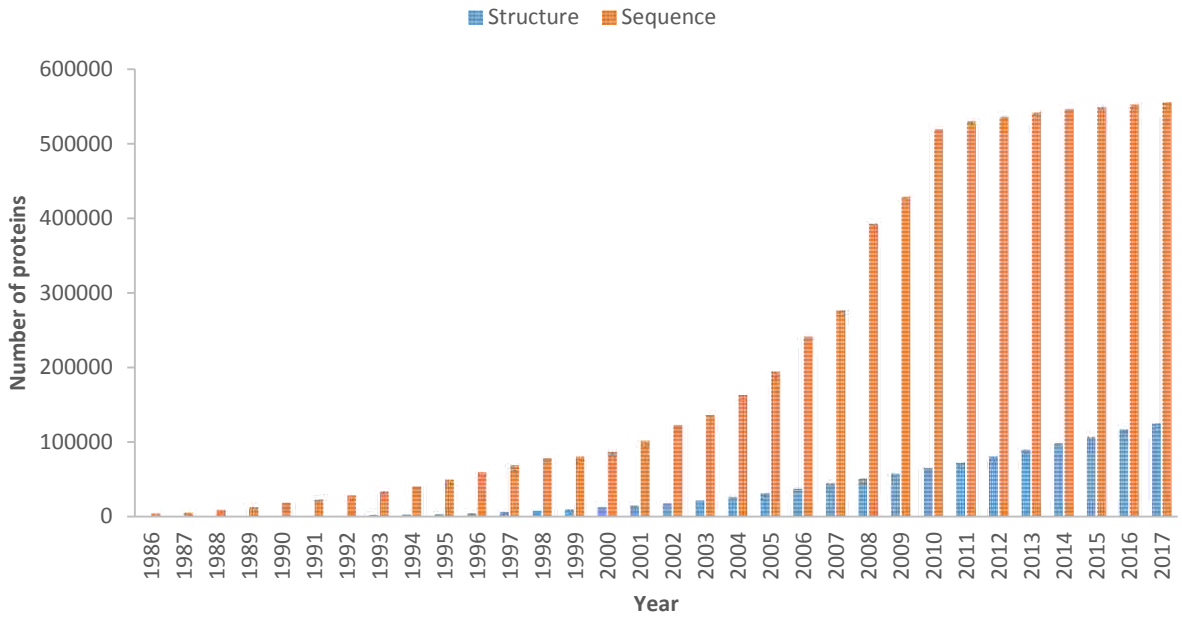


Fig. 2. Number of protein sequences and structures available each year. Blue bar denotes the number of protein structures in PDB, orange bar is the number of protein sequences in SWISS-PROT [6]

Today, one of the most accurate and consistent methodologies for computational protein structure modeling is template-based modeling [7] [8] [9]. The idea behind template-based modeling is simple: when given a protein with unknown structure (target) that is similar in sequence to a known protein, then we can deduce that both proteins share structural similarities. Hence, the first step in template-based modeling is to find a protein with known structure (template) that potentially resembles the target protein sequence. Then, in order to discover the shared similarity between the target and template sequences, the two sequences are aligned together. The matching parts in the alignment will reveal the similar regions in the two sequences, while the dissimilar regions will appear as gaps along the alignment. Subsequently, a framework for the target structure can be constructed by copying the aligned regions from the template structure. Additionally, the unaligned regions are built up, usually as loops. Finally, the complete predicted target structure model is assembled by filling up the gaps in the structural framework

with the constructed unaligned regions. In summary, template-based modeling consists of four main steps: 1) finding a template protein structure with a similar sequence to the target (template selection); 2) aligning the template and target sequences; 3) constructing a framework for the target; and 4) building a complete structural model for the target sequence [10]. The first two steps combined are known as the threading procedure [11] [12]. Fig. 3 shows a block diagram for the template-based modeling steps.

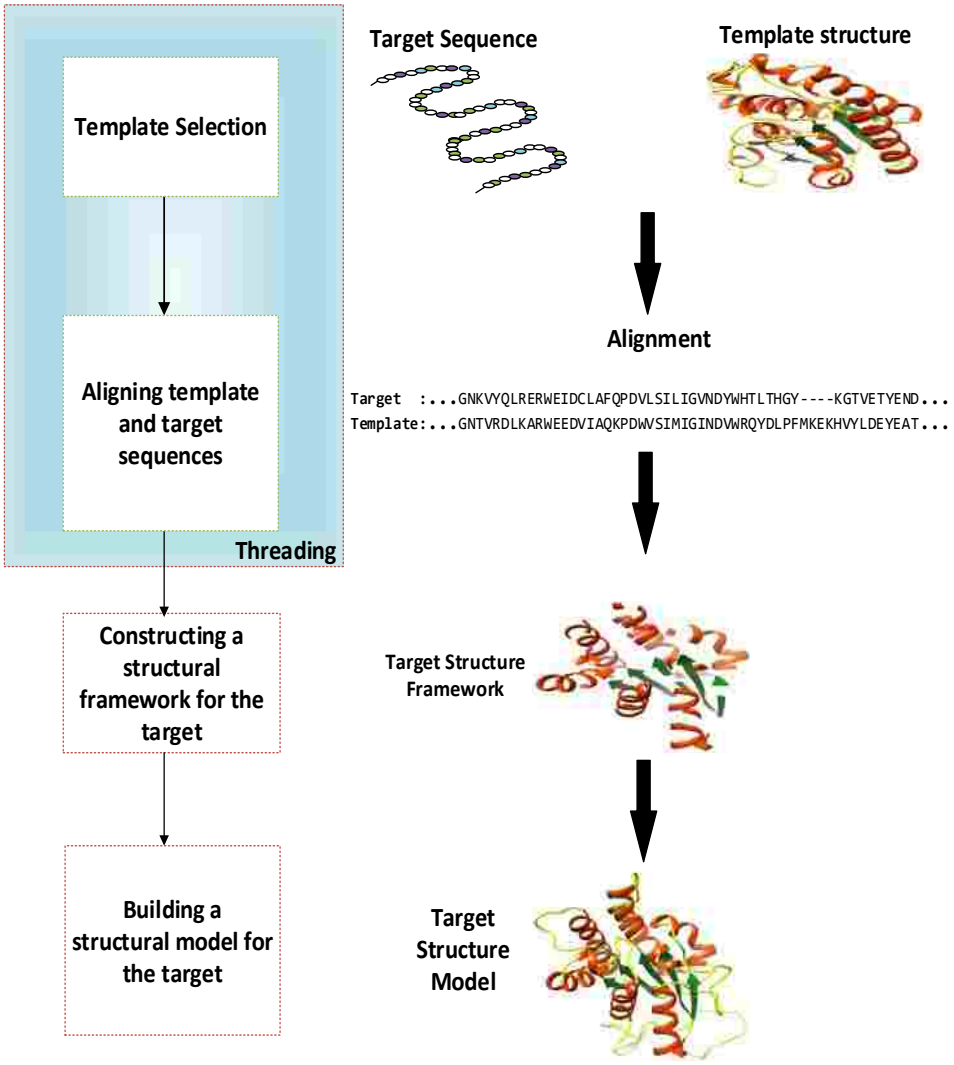


Fig. 3. Block diagram of the protein template-based modeling steps

The success of template-based modeling relies on correctly identifying one or a few experimentally determined protein structures as templates that are likely to resemble the structure of the target sequence, as well as accurately producing a sequence alignment that maps the residues of the target sequence to those of the template. Hence, identifying the most appropriate template protein structures (template selection) to align with the target is a vital process in this methodology. However, the continuously increasing protein sequence and structure data provide a challenge to differentiate the most appropriate templates from the hundreds of thousands of possibilities. Therefore, more sensitive and accurate template selection methods are of a great need to identify the most likely structural templates.

After selecting the most appropriate template for a target sequence, a target-template sequence alignment is generated. The created alignment specifies which residues of the target are to be modeled based on which residues of the template. A correct alignment is essential for successful modeling, while a misalignment of a single residue may result in massive errors in the generated model.

1.2 Contributions of This Dissertation

Mostly the template selection and target-template sequence alignment are combined in the threading procedure. Threading comprises aligning the target protein sequence with all protein structures in a PDB library, and ranks the templates based on the alignment to identify the most compatible template. Accordingly, the target structural model is built using the same alignment that is generated for template selection. However, this alignment may not be the most suitable one to build the target model. As the alignment algorithm implemented in the threading procedure has a strict time constraint, in order to generate alignment for the target with all the template structures

in a given PDB library in a reasonable time. Hence, after selecting the template, implementing a more in-depth target-template sequence alignment shall enhance the protein structure modeling.

In this dissertation, we aim at improving the template-based protein structure modeling by enhancing the correctness of identifying the most appropriate templates and precisely aligning the target and template sequences. The major contributions of this work include:

- **Incorporate inter-residue contacts to enhance template selection**: Most of the template selection methods try to take advantage of multiple structural information sources, such as sequence profiles, secondary structures, solvent accessibility, backbone dihedral angles, etc., to help find the optimal match between the target and the structural templates. In protein structure modeling literature [13], it is well-known that the inter-residue contacts play an important role in forming and stabilizing a protein fold. In this dissertation, we present two template selection approaches that incorporate inter-residue contacts to enhance template selection sensitivity:
    1. Our first template selection approach combines the inter-residue contact score with the sequence profile score, which is a representation of protein structural features. More specifically, we incorporate ICOSA [14], a coarse-grained contact potential correlating inter-residue interaction distance and orientation, into MUSTER [15], one of the most successful template alignment and selection methods in template-based protein structure modeling. Similar to most template selection methods, MUSTER performs alignment for target sequence with all the protein structural templates in its database. The performed alignment is done using dynamic programming

that exploits the protein structural features. These structural feature scores are summed along with balancing weights to give the final MUSTER score. Afterwards, ICOSA is applied to all structural templates found by MUSTER. Since ICOSA is a contact potential measuring global inter-residue interactions while the sequence profile alignment score in MUSTER estimates local interactions, adding the two scores has the potential to enhance template selection sensitivity [16].

2. Our second template selection approach is a further improvement to the template based protein structure modeling. In this approach, instead of evaluating the ICOSA score of a target adopting a potential structural template after an alignment is generated, we use ICOSA score to build the alignment along with other structural features scores. A substitution matrix is built to score the replacement of each amino acid in the template three-dimensional conformation with every amino acid in the target. Then, this substitution matrix is used in building the alignment along with the structural features. The alignment is generated by dynamic programming that exploits the protein features including (1) sequence profiles; (2) predicted secondary structures; (3) fragment profiles; (4) predicted solvent accessibility; and (5) ICOSA score for substituting each target amino acid in the template folding topology. These protein features are summed together using weights that are determined based on Grid search technique. The resulting alignment score is a ranking score that measures the favorability of each potential template.

- **Designing a multi-objective alignment algorithm:** We propose a multi-objective protein sequence alignment method. As a correct alignment is critical for protein modeling, given a set of potentially conflicting objective functions, we develop a novel multi-objective sequence alignment algorithm to obtain a set of diversified alignments yielding Pareto optimality. The multi-objective alignment algorithm guarantees not only Pareto optimality of the alignments, but also completeness of the solutions. In theory, the multi-objective sequence alignment algorithms can be considered as a super consensus method [37] whose goal is to derive all possible alignments with diversified consensus over all positive weight combinations of the given objectives. As a result, compared to finding a single alignment by optimizing a certain combination of individual objective terms, the alignments obtained by the multi-objective alignment algorithm enable one to analyze the trade-offs among potentially conflicting objective functions, which allows us to pick more suitable alignments for protein modeling.

1.3 Dissertation Organization

The rest of the dissertation is organized as follows. Chapter II presents background about proteins, protein structure, protein structure modeling, and protein structure prediction. Chapter III presents a review of the relevant literature to template-based protein structure modeling, template selection, and pairwise sequence alignment. We present our template selection approaches, and sequence alignment algorithms in Chapters IV and V, respectively. Finally, Chapter VI summarizes the dissertation and discusses our future (post-dissertation) research directions.

CHAPTER II

BACKGROUND

Proteins are complex organic compounds formed by chains of simpler compounds, called amino acids. Usually, a protein's chain composition is denoted as the primary structure. The primary structure determines the protein's three-dimensional structure, which in turn regulates the protein's function. In this chapter, we briefly introduce the protein molecular composition, protein structure and protein structure modeling. The protein background presented here will assist in understanding the problems we are investigating in this dissertation.

2.1 Proteins

Proteins are the main components of living cells and constitute more than quarter the weight of a typical cell. They play a crucial role in governing several life processes by performing the most essential biological and chemical functions in every living cell. The protein structure provides invaluable insights into the molecular basis of their functions. Proteins are composed of small molecules named amino acids. There are 20 different amino acids, each with a distinct chemical structure and characteristics.

2.1.1 Amino Acid

Amino acids are compounds that contain an amino group (NH_2), and a carboxyl group (COOH). Both groups are linked to a central carbon (Cα) that is attached to a hydrogen and a side chain (R) (Fig. 4). The side chain determines the specific properties of the amino acid. A protein is a chain of amino acids joined together by peptide bonds. Each pair of amino acids forms a peptide bond between the amino group of one and the carboxyl group of the other (Fig. 5). The atoms forming the peptide bond are known as the backbone atoms. They are the nitrogen of the amino group, the Cα, and the carbon of the carbonyl group.

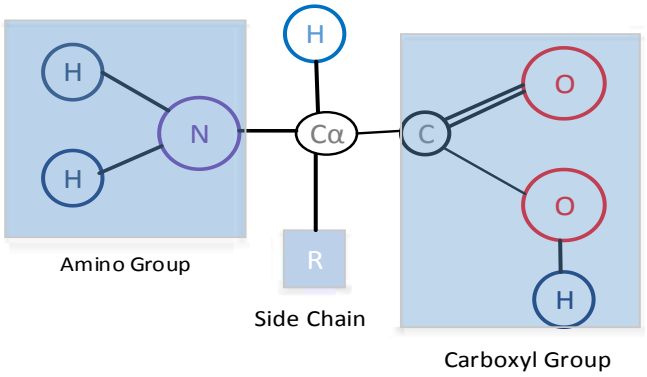


Fig. 4. The General Structure of an amino acid.

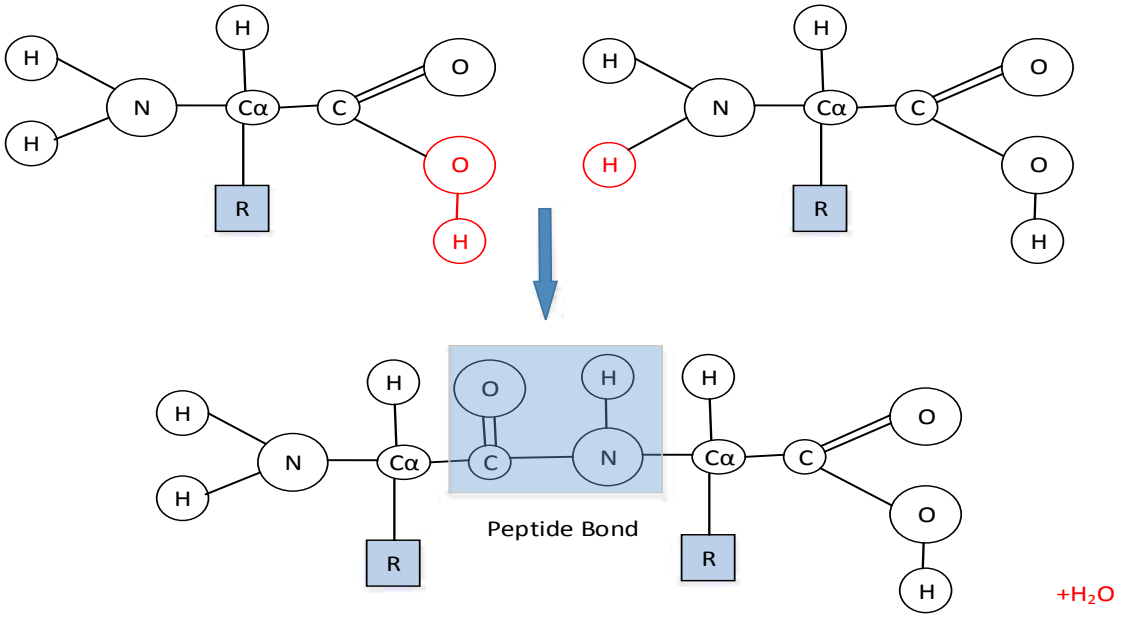


Fig. 5. Peptide bond Formation

Usually, an amino acid is referred to by the first three letters of its name. Such an abbreviation is easy to remember; however it uses up unnecessary memory in computer databases. Hence, the 20 common amino acids can be encoded by 20 letters of the alphabet, and then each

amino acid in a protein sequence uses up only 1 letter rather than 3. Unluckily, we can't simply use the first letter as many amino acids start with the same letter (like Ala, Arg, Asp, and Asn). Table 1 list the standard amino acids and their abbreviations.

Table 1
Twenty standard amino acids and their abbreviation

Amino Acid Name	Three Letter Code	One Letter Code
Alanine	Ala	A
Arginine	Arg	R
Asparagine	Asn	N
Aspartate	Asp	D
Cysteine	Cys	C
Glutamate	Glu	E
Glutamine	Gln	Q
Glycine	Gly	G
Histidine	His	H
Isoleucine	Ile	I
Leucine	Leu	L
Lysine	Lys	K
Methionine	Met	M
Phenylalanine	Phe	F
Proline	Pro	P
Serine	Ser	S
Threonine	Thr	T
Tryptophan	Trp	W
Tyrosine	Tyr	Y

Based on the chemical and physical properties of the side chain amino acids can be grouped into several categories, such as size, charge, and affinity for water. According to these properties, the side chain categories can be represented as: small, large, hydrophobic, and hydrophilic categories. Inside the hydrophobic group of amino acids, they can be subdivided into aliphatic and aromatic. Aliphatic side chains are linear hydrocarbon chains and aromatic side chains are cyclic rings. Inside the hydrophilic group, amino acids can be further divided into polar and charged. Charged amino acids can be either positively charged (basic) or negatively charged (acidic).

2.2 Protein Structure

In nature, the protein amino acid chain doesn't stretch out in a straight line, it rather folds into a unique three-dimensional structure. This structure is critical to the protein biological function. There are four distinct levels of protein structure: primary, secondary, tertiary, and quaternary (Fig. 6).

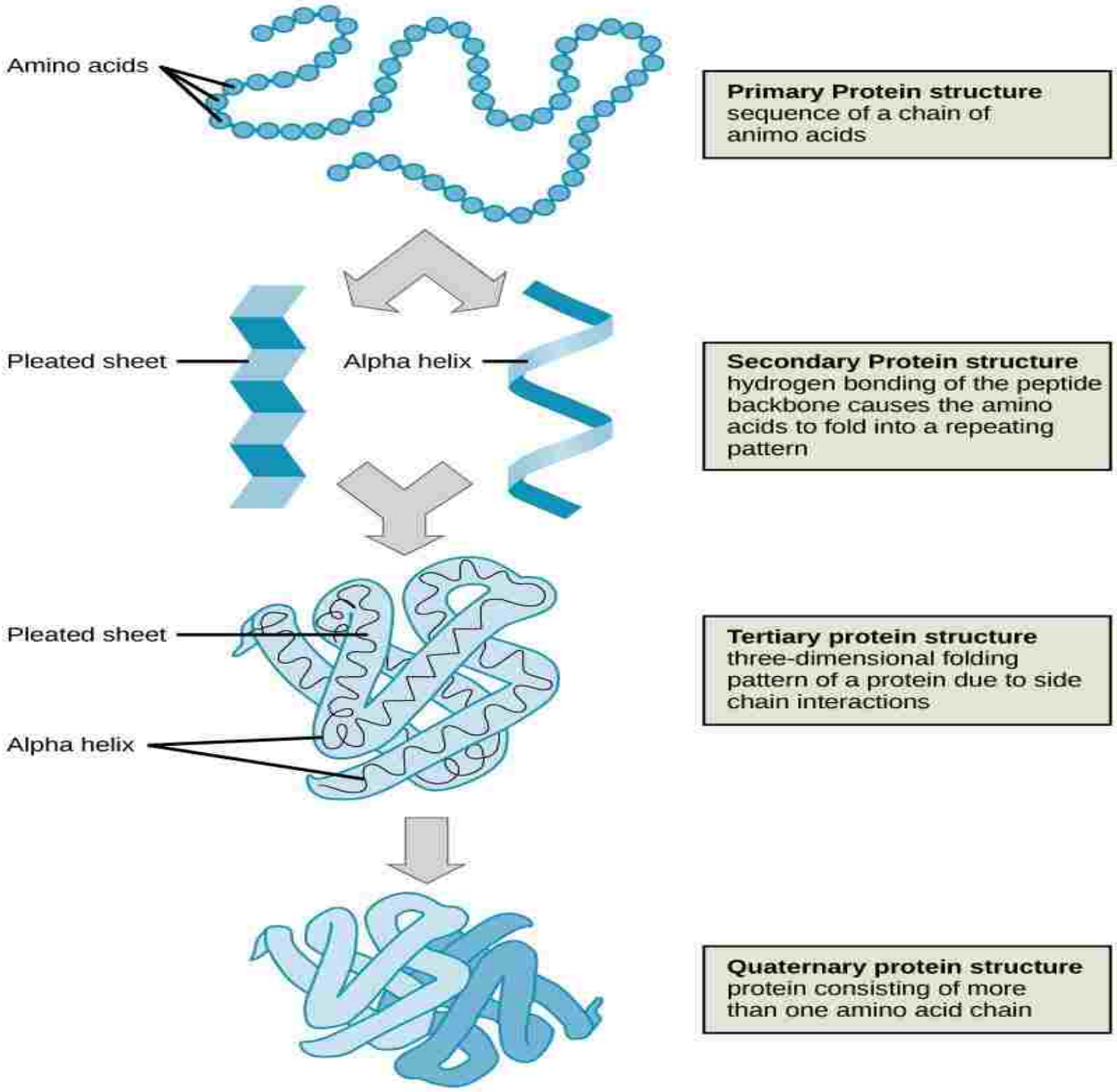


Fig. 6. The four levels of protein structure (source [17])

2.2.1 Primary Structure

The sequence of amino acid residues that form a protein chain is called its primary structure. The primary structure shows the sequence of the amino acids connected together by peptide bonds forming the protein chain. The two ends of the protein chain are: N-terminus at the start and C-terminus at the end.

Fig. 7. Primary structure of chain A of human insulin protein (1MSO)

2.2.2 Secondary Structure

Secondary structure refers to the local conformation of amino acids in the protein chain. They are stabilized by the hydrogen bonds between carbonyl oxygen and amino hydrogen of different amino acids. There are two main types of secondary structure: α helix, and β pleated sheet. Both structures are formed and stabilized by the patterns of hydrogen bonds. Other types of secondary structure have been identified, such as 3_{10}-helix, and π-helix. However, they are less common patterns. Turns or loops are other types of secondary structure that link the more regular secondary structure elements. Finally, the conformations that are not related to a regular secondary structure are named coils or loops.

2.2.2.1 α-Helix

The α-helix main chain conformation resembles a spiral. The α-helix structure is stabilized by hydrogen bonds between amino hydrogen (N-H) group and carbonyl oxygen (C=O) of four amino acids further along the chain. The hydrogen bond is almost parallel to the helix axis, while the side chain groups stem out of the helix perpendicular to its axis. Each turn in the helix spiral holds 3.6 amino acids, and it is about 5.4 Å long. The helix turns can be clockwise or counter-clockwise.

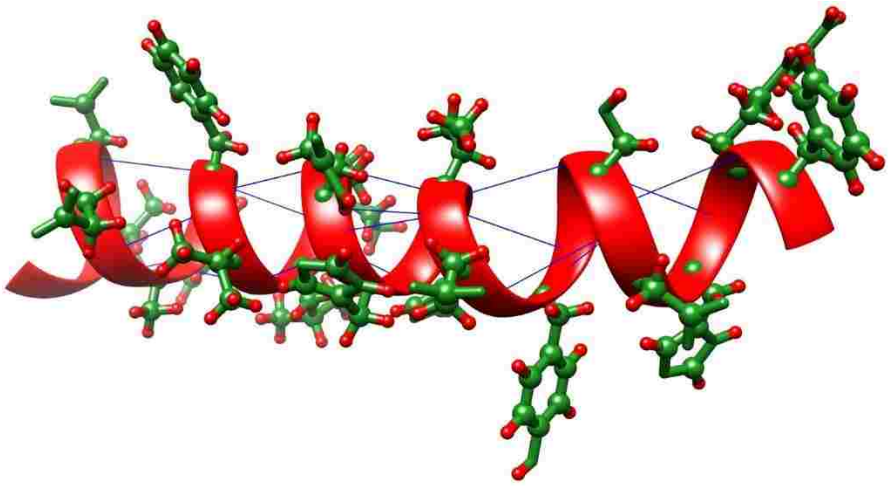


Fig. 8. α-Helix from 1amb protein where the hydrogen bond is shown as blue lines and the side chain atoms stem out of the helix

2.2.2.2 β-Sheet

A β-sheet is a stretched configuration built up from two or more adjacent segments of a polypeptide chain. Each segment involved in forming the β-sheet is a β-strand. The β-sheet structure is held together by hydrogen bonds formed between residues of adjacent strands, while the side chain extends above and below the sheet plane. The β-strands may be parallel (extending in the same direction), or antiparallel (extending in opposite directions).

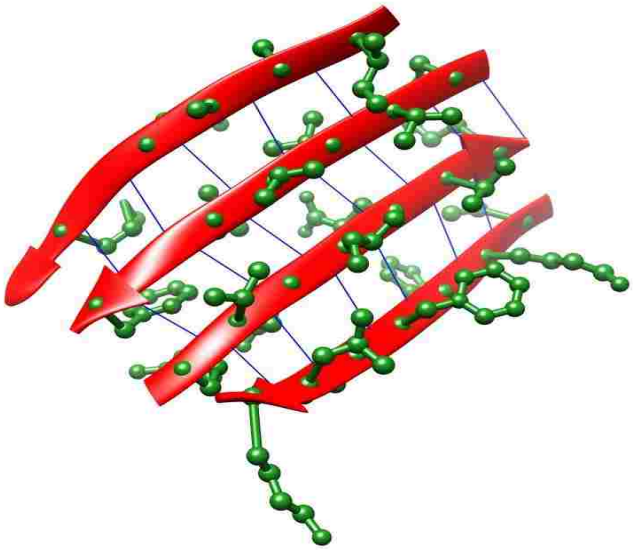


Fig. 9. β-Sheet from chain A of 4erh protein where the hydrogen bond is shown as blue lines and the side chain atoms extend above and below the sheet plane

2.2.3 Tertiary Structure

Tertiary structure refers to the global three-dimensional conformation of a protein. In other words, the tertiary structure is the packing and arrangement of the secondary structure elements. The tertiary structure of a protein is determined by the interactions between long distances amino acids that are brought close together in space by the way the protein folds. These interactions can be electrostatic interactions, hydrophobic interactions, hydrogen bonding, van der Waals bonds, and others. The protein tertiary structure is represented by 3D coordinates for each atom. Fig. 10 shows the tertiary structure of a protein.

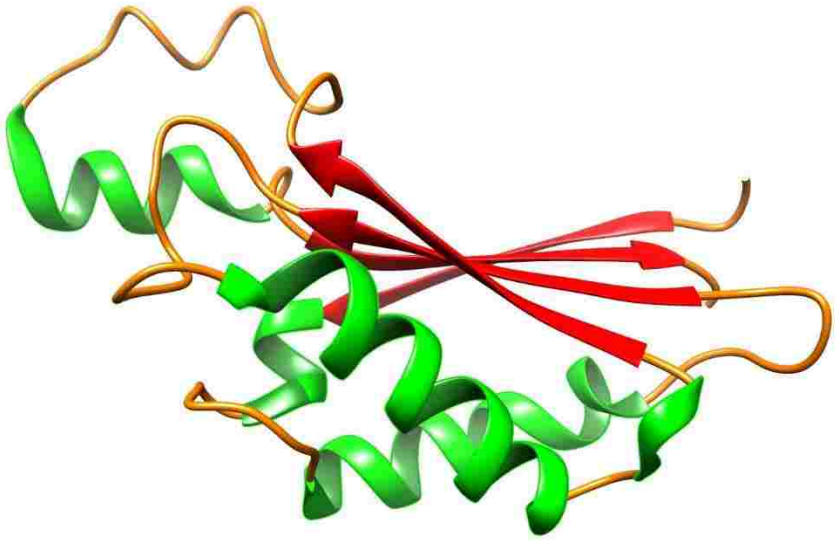


Fig. 10. Tertiary Structure of 4erh protein chain A.

2.2.4 Quaternary Structure

Quaternary structure represents the multiple polypeptide chains interactions. It is the three-dimensional structure of several polypeptide chains that function as a single unit. The quaternary structure is stabilized by non-covalent interactions between the atoms of different chains.

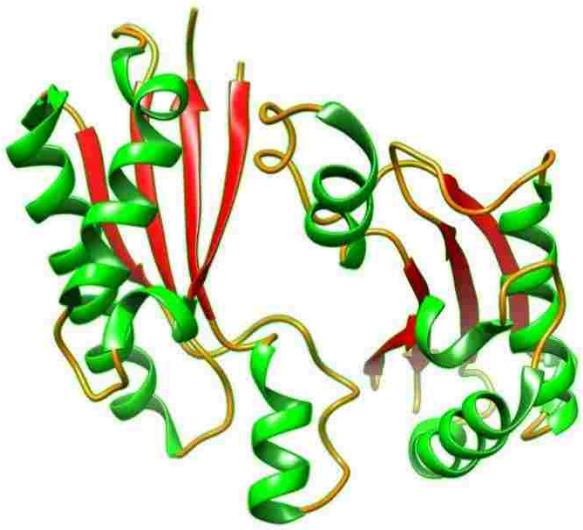


Fig. 11. Quaternary structure of protein 4erh, and it involves of two polypeptide chains.

2.3 Protein Structure Modeling

There are three experimental methods for protein structure modeling:

- X-ray crystallography
- Nuclear Magnetic Resonance (NMR)
- Cryo-electron microscopy

2.3.1 X-Ray Crystallography

The first research implemented to study protein structure and function was in the 1950s. This work mainly aimed to discover the relationship between protein sequence and protein chemical characteristics. One of the earliest research of protein structure modeling is the work conducted by F. Sanger to identify the structure of insulin in 1955 [18]. In 1958 and 1960, John Kendrew published two research papers [19] [20] that are marked as the first protein three-dimensional structure determination solution. In his research, John Kendrew used X-ray crystallography to determine the three-dimensional structure of the myoglobin protein. Today X-ray crystallography is the most common technique to determine the three-dimensional structure of

a protein in the Protein Data Bank (PDB) by measuring the 3D density distribution of electrons for the protein in the crystallized state [21].

2.3.2 Nuclear Magnetic Resonance (NMR)

In 1967 Kurt Wüthrich used NMR techniques to study protein three-dimensional structure [22]. Subsequently, in 1982 and 1984, his research group published several papers that outlined a framework for NMR structure determination of proteins. The NMR technique works by placing a protein molecule in a magnetic field that irradiates the protein molecule with radio-frequency pulses. Afterward, the position of the atoms is determined by the energy radiated back [23]. Around 9% of the protein structures in the PDB are determined by NMR techniques.

2.3.3 Cryo-Electron Microscopy

Recently, Cryo-electron microscopy has become an important means of determining protein structures. Cryo-electron microscopy was first introduced in 1984 [24] by Marc Adrian. Cryo-electron microscopy is a valuable resource for working with very large protein complexes, as it identifies protein structures at a high resolution. Cryo-electron microscopy is a microscopy technique in which a beam of electrons is transmitted through a protein sample to form an image. Its efficiency is allowing specimens to remain in their native state without the need for dyes or fixatives to study the fine cellular structures [25].

2.4 Protein Structure Prediction

The experimental protein structure modeling methods are often time-consuming, costly, and not feasible for some proteins. Therefore, the capacity of producing sequence information is extremely higher than that of producing structural information. Accordingly, researches focus more and more on computational approaches to accurately predict protein 3-dimensional structure.

Protein structure prediction techniques can be categorized into two main approaches: *ab initio* and comparative protein modeling. The *ab initio* approach attempts to build protein three-dimensional structure from scratch, whereas the comparative protein modeling approach uses templates from previously solved structures as the starting points to build the three-dimensional structure.

2.4.1 Ab-initio

The *ab-initio* protein structure modeling method relies on physical principles to search the protein conformation space for a possible solution and identify local structure building blocks. This is done by modeling an atomic interaction force field or a knowledge-based energy potential to locate the conformation yielding the lowest energy. This conformation corresponds to the most stable protein structure, according to Anfinsen's thermodynamics hypothesis. The difficulty in these *ab-initio* approaches lays in the validity of the available molecular models and the complexity of the search space [26] [27].

The most well-known *ab-initio* algorithm is the assembly of the three-dimensional structure of a protein using small fragments, introduced by Bowie and Eisenberg [28]. A similar algorithm is that presented in ROSETTA by Baker's research team [29], which has demonstrated success in the Critical Assessment of protein Structure Prediction (CASP) experiments [30] [31]. An additional *ab-initio* algorithm was introduced in [26], which is based solely on global optimization of a potential energy function. Afterward, Zhang et al. [32] developed the *ab-initio* protein structure prediction approach, called TOUCHSTONE, that combines short-range and long-range knowledge-based potentials to predict the protein structures. Subsequently, the ASTRO-FOLD *ab-initio* protein three-dimensional structure prediction method was designed by Klepeis

using binary patterned combinatorial libraries of *de novo* sequences [33]. ASTRO-FOLD was successfully applied to an α-helical protein of 102 residues [34].

In order to increase the efficiency of *ab-initio* approaches, researchers work on reducing the level of protein structure representation, which accordingly will reduce the size of conformational search space [35] [36] [37]. Despite recent progress in *ab-initio* algorithms, it remains challenging to fold a general protein [36], particularly if it is a long one.

2.4.2 Comparative Modeling

The comparative protein modeling approach is based on the knowledge learned from the previously experimentally-determined protein structures. Comparative modeling is considered the most accurate protein structure prediction method in recent CASP experiments [38] [39] [40] [41]. The fundamental idea behind comparative modeling is to find related proteins with a known structure that we can deduce the unknown protein structure from the shared similarity between the two proteins. These methods are also known as template-based modeling [42].

The idea behind template-based modeling is simple; when given a protein with unknown structure (target) that is similar in sequence to a known protein, then we can deduce that both proteins share structural similarities. Hence, the first step in template-based modeling is to find a protein of known structure (template) that resembles the target protein sequence. Then, in order to discover the shared similarity between the target and template sequences, the two sequences are aligned together. The generated alignment will reveal the similar regions in the two sequences by aligning them together, while the dissimilar regions will appear as gaps along the alignment. Subsequently, a framework for the target structure can be constructed by copying the aligned regions from the template structure. Additionally, a built-up structure is constructed for the unaligned regions. Finally, the complete predicted target structure model is assembled by filling

up the gaps in the structural framework with the constructed unaligned regions. The process of identifying the most compatible templates for a target protein sequence, combined with aligning the template and target sequences, are known as the threading procedure [11] [12].

Threading is one of the most active research areas in protein structure prediction. As the success of the modeling of the protein structure mainly relies on the threading process, the accuracy of template-based modeling mainly depends on the amount of similarity between the target and the template as well as the quality of the alignment performed on the two sequences.

CHAPTER III

LITERATURE REVIEW

This chapter presents a review of the relevant literature to the problems inspected in this dissertation. We provide an overview of the template-based protein structure modeling techniques, template selection methods, and pairwise sequence alignment algorithms.

3.1 Template-based Protein Structure Modeling

The foundation for template-based protein structure prediction is based on three observations: (1) similar sequences embrace similar protein structures [43] [44] [45]; (2) many different sequences fold into similar structures [46] [47]; and (3) the number of unique structural folds is relatively small, when compared to the number of proteins in nature [48] [49] [50] [51] [52] [53]. The first structural model, predicted using a template-based approach, was built in 1969 by Browne and colleagues [44]. Their work was based on the X-ray structure of lysozyme. They started by aligning the target and the template protein sequences, then constructing an initial protein model, and finally finishing by the refinement of the model using energy minimization.

In 1981, Greer developed a computer program to automate the whole procedure of template-based protein structure modeling [54]. Using this program, eight proteins of the mammalian serine proteases family were modeled. The modeling method was based on three experimentally determined structures from the same protease family. In his work, Greer observed that the structure of a protease could be divided into structurally conserved regions, which contain the strong sequence homology and structurally variable regions, including all the additions and deletions. Additionally, Greer found that a variable region that has the same length and residue character in two different known structures usually has the same conformation in both proteins.

Based on these two observations he was able to create the conserved and variable regions of the structurally unknown proteins from the known structures.

This method proved that mammalian serine proteases could be built semi-automatically from the known homologous structures. Hence, both the need for manual examinations and the use of energy force fields were greatly reduced. Greer's procedure was later implemented in a protein molding program, Homology, and integrated into the InsightII molecular graphic package [55].

Despite using multiple protein structures from the same family to define the conserved and variable regions in the target protein, Greer's method only used one protein structure as the template to model the target protein. Blundell and colleagues discovered that an average structure (framework) of multiple protein structures from the same family resembles more the target protein structure than any single protein structure did. Based on this discovery, they developed a program called Composer, which builds a structure framework that serves as a guide for the assembly of fragments of homologous proteins in modeling an unknown protein [56]. The framework-based protein modeling significantly increased the reliability of model construction over the previous semiautomatic methods. Later, Composer was integrated into the protein modeling package Sybyl. Continuous improvements in computer graphics and distance geometry have provided important tools for template-based modeling of protein structures [57]. Subsequently, the structures of many important proteins have been modeled, such as insulin-like growth factors [58], renin [59], and immunoglobulins [60].

Till 1993, protein modeling methods were semiautomatic, including separating modeling procedure for the structure of conserved regions, variable regions, and side chains. Sali and Bundell were the first to create a full-atom protein modeling program (MODELLER) [61].

MODELLER works on finding the most accurate structure for a target sequence given its alignment with known protein structures. The three-dimensional structure of the target protein is obtained by optimally sustaining spatial restraints derived from the alignment and expressed as probability density functions (pdfs) for the features restrained. MODELLER is one of the most popular and widely used modeling programs [62] [63].

In 1996, Manuel Peitsch initiated PROMOD and SWISS-MODEL as a fully automated protein structure modeling server [64]. SWISS-MODEL begins with the identification of suitable template structures. These structures are then aligned with the target, taking into account the similarity between all templates. PROMOD is used to construct models for protein target based on an averaged framework using the generated multiple sequence alignment.

NEST [65] is a model building program that applies an artificial evolution method to construct a model from a given template. NEST performs operations of mutation, insertion, and deletion on the template structures one at a time. After each operation, a torsional energy minimizer is applied and energy is calculated based on a potential function. This process is repeated until the target sequence is completely modeled. "FRankenstein's monster" [66] is a template based protein modeling approach, which was developed by Kosinski et al. It merges the finest fragments of fold-recognition models and iterative model refinement aided by 3D structure evaluation. The originality in "FRankenstein's monster" is that it employs the idea of combining fragments.

In the last decade, the interest in fully automated protein structure modeling methods increased with the growing popularity of CASP [67]. Several fully automated protein structure modeling packages were developed, whereas some were freely available servers, such as $(PS)^2$ [68] and its advanced version $(PS)^2$-v2 [69]. I-TASSER [70] is another freely available protein structure modeling server that was originally developed for participating in CASP7. After being

ranked the best method in the server section of the CASP7 experiment, I-TASSER became freely available [71]. HHblits [72] is a remarkable automated protein structure modeling server, which was the top ranked server for CASP9. Another commonly used freely available server for protein structure modeling is RaptorX [73] [74].

3.1.1 Threading

Originally, the word “threading” was first introduced by Jones, Taylor, and Thornton to describe their novel protein fold recognition approach [12]. The success that was achieved by their method gave it a huge recognition in the 1990s. Owing to the popularity of the method, “threading” became a generic term to describe fold recognition operation. Though “threading” is in fact a special sub-class of fold recognition, the term is often used to distinguish protein three-dimensional prediction structure-based methods from sequence-based methods [75].

Threading is the process of aligning a protein sequence with one or more protein structures, where the protein sequence is threaded onto a given structure to obtain the best sequence-structure compatibility. Obviously, identifying the most compatible templates for a target protein sequence is also part of the threading process. In order to improve the sensitivity of both template identification and target-template alignment, threading introduces the use of evolutionary information.

The development of the threading approach is based on the concept that many unrelated protein sequences fold into similar structures. Moreover, certain structural folds were detected to be popular among proteins without any obvious sequence similarity [76] [77] [78] [79]. Consequently, it is more sensible to relate the template protein structures with the target protein sequence than to match their sequences.

The first threading approach, THREADER [12], uses the technique of double dynamic programming similar to the one used by Taylor and Orengo in [80], in order to perfectly fit a target sequence onto the 3-dimensional structures of known proteins. Then the best models are identified using energy potentials derived from the statistical analysis of known structures. The success of THREADER was publicized by its ability in the first CASP to identify 8 out of 11 target sequences, which have no discernable sequence similarity to known structures [81]. On the years following THREADER, several successful protein structure modeling methods were developed based on the threading approach. One of these approaches is the recursive dynamic programming threading method developed by Ralf Thiele and his colleagues [82].

In 1999 Jones developed another threading algorithm, GenTHREADER [83], which is one of the first methods to combine sequence profile-based searches with energy potentials derived from threading. The GenTHREADER starts by performing a sequence-profile based search against a non-redundant fold library using BLASTP program [84]. This search is performed to generate profiles for each template structure in the fold library. Using the generated profile a sequence to structure alignment is formed for each template. The resulting alignments is then evaluated using the energy potentials from the original THREADER method. Finally, an artificial neural network is trained to recognize targets and templates with matching folds, which is used to evaluate the output alignments based on the alignment scores, pairwise energy scores, solvation energy scores, and length.

Following GenTHREADER, a number of threading methods have been developed that employed a similar hybrid approach. In 2000, the INBGU method [85] was presented by Daniel Fischer. INBGU uses a combination of sequence profiles and comparisons of a predicted secondary structures of the target with the observed secondary structures of each template. By

incorporating secondary structure scoring, INBGU was able to detect distant homologues as the secondary structures are better conserved throughout evolution than sequences. 3D-PSSM [86] is another threading approach that incorporated the predicted secondary structure of the target protein. In 3D-PSSM the target profiles were aligned against 3D position-specific scoring matrices (PSSMs). First, for each template in the fold library, PSI-BLAST [87] was used to generate an initial 1D sequence based PSSM. Then further enhancement to this PSSM is performed, using solvation potentials, secondary structures, and structural alignments, resulting in a 3D-PSSM. Similar to 3D-PSSM, the FUGUE program [88] uses structural alignments, solvent accessibility, and secondary structure information in order to produce environment-specific scoring matrices. Additionally, FUGUE made use of structure-dependent gap to align target sequence profiles against template structural profiles.

Hybrid threading methods have gone through several improvements over the past years in order to integrate new innovations in sequence searching and alignment. For instance, GenTHREADER has been updated to incorporate structural information, which has resulted in the detection of more remote homologues [89]. Later, another version of the method, mGenTHREADER [90], also incorporates profile-profile alignments. Following mGenThreader, pGenTHREADER [91] was presented as another implementation of the GenTHREADER method for structure prediction on a genomic scale. This method combines profile–profile alignments with secondary-structure specific gap-penalties, classic pair- and solvation potentials using a linear combination optimized with a regression Support Vector Machine (SVM) model. Currently, EigenTHREADER [92] is the latest version of GenTHREADER, which implements protein threading by exploiting new developments in residue-residue contact prediction rather than statistical potentials. EigenTHREADER takes a query amino acid sequence, generates a map of

intra-residue contacts, and then searches a library of contact maps of known structures. To allow the contact maps to be compared, EigenTHREADER uses eigenvector decomposition to resolve the principal eigenvectors these can then be aligned using standard dynamic programming algorithms.

Another successful threading method is TASSER [93], which combines the best sequence searching and threading methods along with improvements in the selection of the highest quality models. TASSER was developed by Yang Zhang and Jeffrey Skolnick. After the success TASSER showed in CASP6, Zhang developed I-TASSER [94]. I-TASSER progressively implements the TASSER simulations, where template alignments are generated by four simple variants of the profile–profile alignment method with different combinations of the hidden Markov model and PSI-Blast profiles with dynamic programming alignment algorithms. In CASP7, I-TASSER automated server prediction generates models as good as the human-expert does in all categories, and was ranked the best prediction server. I-TASSER continued its success in the following CASP experiments in both the human-expert and server [95] [96] [97] [98] [9]. Along these experiments, several improvements were made to I-TASSER, such as increasing the coverage of template detections by combining various structural features with profile-to-profile alignments [99]. Also, I-TASSER approach has been extended for annotating the biological function using the predicted protein structures, based on a combination of local and global structural similarities with proteins of known functions [100] [37].

3.1.2 Template Selection

The performance of a threading program largely depends on how close the template structure is to the actual structure of the target protein. Hence, selection of the best template is of fundamental importance for the quality of a generated three-dimensional model. Usually, there

exist several proteins sharing the common structural core with the target, but many of these proteins may still differ in the relative orientation of the secondary structure elements. So, the objective is to select a template from several alternatives that is likely to be most structurally similar to the unknown structure of the target.

Optimally, one template would have a very similar structure to the target and is better than other templates. If such a template exists, it would be the top match and be used as the main template. However, if there is no clearly preferred template, an attentive template selection must be applied. There are mainly four categories for template selection methods: sequence-based, evolutionary, structural, and knowledge-based [101].

Despite the fact that template selection methods work to find a single template, sometimes it is not possible to unequivocally select a single best template from a set of alternatives. In such cases, a model can be built based on multiple templates. This is performed by either averaging the coordinates of superposed templates, or modeling different regions of the target based on different templates. Selecting more than one template proves sometimes to be effective and accurate [102] [103]. Also, this technique was used in several successful protein modeling methods, such as "FRankenstein's monster" [66].

3.1.2.1 Sequence-based Methods

These methods are based on a theory that the template with the highest sequence similarity to the target sequence should also disclose the highest structural similarity. Usually, these methods compare between the target and potential templates by building pairwise alignments or by running PSI-BLAST against the database that includes the templates sequences. The PSI-BLAST is a reliable method for selecting templates if the target template sequence identity is above 40% [104] [105]. Instead, more sensitive sequence-based methods for template selection are needed, when

target-template sequence identity is lower than 40% [105]. Such methods include comparison of profiles or Hidden Markov Models (HHMs) built for sequence families of the target and all alternative templates. An example of a more sensitive sequence-based method is HHPRED [106] [107], which uses target sequence or Multiple Sequence Alignment (MSA) for building a HMM (Hidden Markov Chain). This HMM is aligned with all HMMs representing annotated proteins or domains with known structure.

3.1.2.2 Evolutionary Methods

The Evolutionary methods rely on a hypothesis that the template with the highest structural similarity to the target is the one that is closest to the target on the phylogenetic tree. In this approach, the target and all the templates under considerations are aligned together, and a phylogenetic tree for this group of related protein sequences is calculated [108]. A phylogenetic interpretation, using evolutionary models and maximum likelihood or Bayesian techniques, is a much better estimator of evolutionary distances than similarity scores from pairwise sequence comparison for closely related sequences. Hence, calculating a phylogenetic tree is useful when there is a high sequence similarity between the target and templates. However, for distantly related sequences, phylogenetic tree are unreliable. Consequently, evolutionary approaches are less popular than other template selection methods [101].

3.1.2.3 Structural Methods

The structural template selection methods estimate how a target protein sequence would fit into the structure of each alternative template. Then these fits are judged based on a score from the threading program. Threading methods that use structural methods for template selection, usually build 3D structural profiles [11] [88]. Furthermore, better templates can be obtained after building a model for each template, and scoring them using the Model Quality Assessment Programs

(MQAPs) [109]. The structural template selection methods are mainly used when there is no significant sequence similarity between the target and the templates.

3.1.2.4 Knowledge-based Methods

Each Knowledge-based template selection method has a set of rules that are taken into consideration when selecting a template. Mainly the rules are to discriminate between structures of the same protein solved under different experimental conditions. Hence, an essential rule is that the template structure must be solved under similar conditions to the conditions anticipated for a model. For example, if the target protein to be modeled is in a ligand-bound conformation, then selected templates should be the ones whose structure was solved with a ligand rather than the ligand-free structure templates. Additionally, if the target protein in the biologically active form is an oligomer, then an oligomeric template with the same number of subunits and symmetry as the target should be used. Also, the model should be built and evaluated as an oligomer instead of a monomer. Other rules can be the preference for template structure solved using X-ray crystallography rather than NMR, or structures with higher resolution [101].

3.2 Protein Sequence Alignment

Protein sequence alignment is the basis for structure and function prediction for a target sequence. Sequence alignment reveals the relatedness of two sequences by discovering the shared similarity between them. The evolutionary relationship between two sequences, which means that the two sequences share a common evolutionary origin, is discovered by a correct sequence alignment. For instance, in a sequence alignment, aligned regions that are not identical represent residue substitutions, while regions in one sequence that correspond to nothing in the other sequence represent insertions or deletions in one of the sequences [110]. Fig. 12 shows an alignment between two sequences and the structure matching revealed from the alignment.

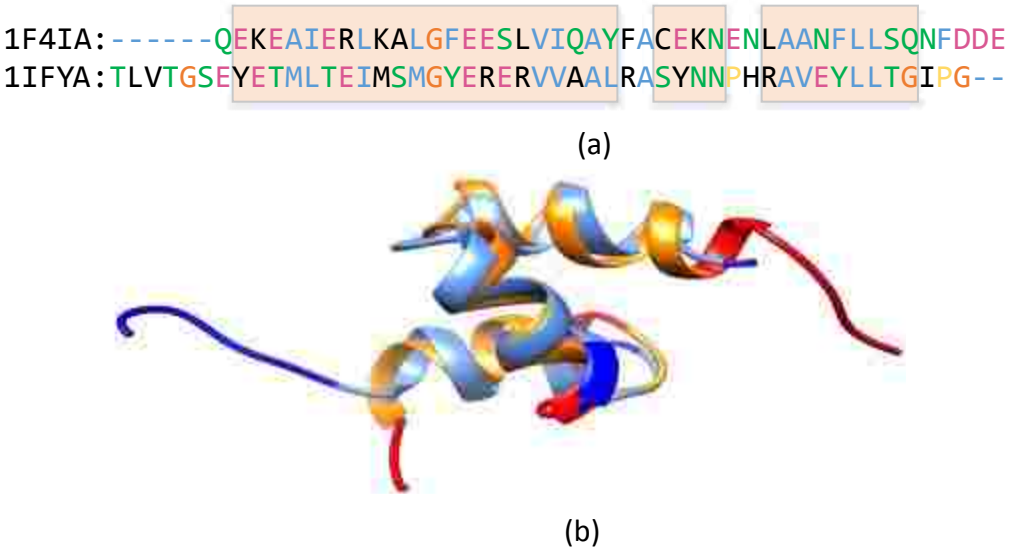


Fig. 12. A sequence alignment, between 1F4I (chain A) and 1IFY(chain A). (a) Alignment produced by Chimera program [111], where the highlight represents matching regions. (b) The superimposition of the two structures based on the generated alignment, where 1F4I is blue and its matching regions are cyan, and 1IFY is red and its matched regions are orange.

Since there are only twenty amino acid residues, accordingly two unrelated protein sequences can match 5% of the residues in a random chance alignment. This percentage can increase to 10-20% when gaps are added. Additionally, sequence length is a factor to determine sequence similarities from an alignment.

For determining a homology relationship of two protein sequences using sequence alignment, there are three regions (classes) based on sequence identity and length. The first region is referred to as being in the “safe zone”, which means that the sequence alignments between a pair of protein sequences unambiguously distinguish between protein pairs of similar and non-similar structure. A sequence alignment lay in the safe zone when the pairwise sequence identity is high (>40% for long alignments). The second region is “twilight zone”. The twilight zone is for sequences with sequence identity between 20% and 30%. The third region is “midnight zone”,

where high proportions of nonrelated sequences are present. This area is for below 20% sequence identity [112]. Fig. 13 shows the three zones of protein sequence alignments.

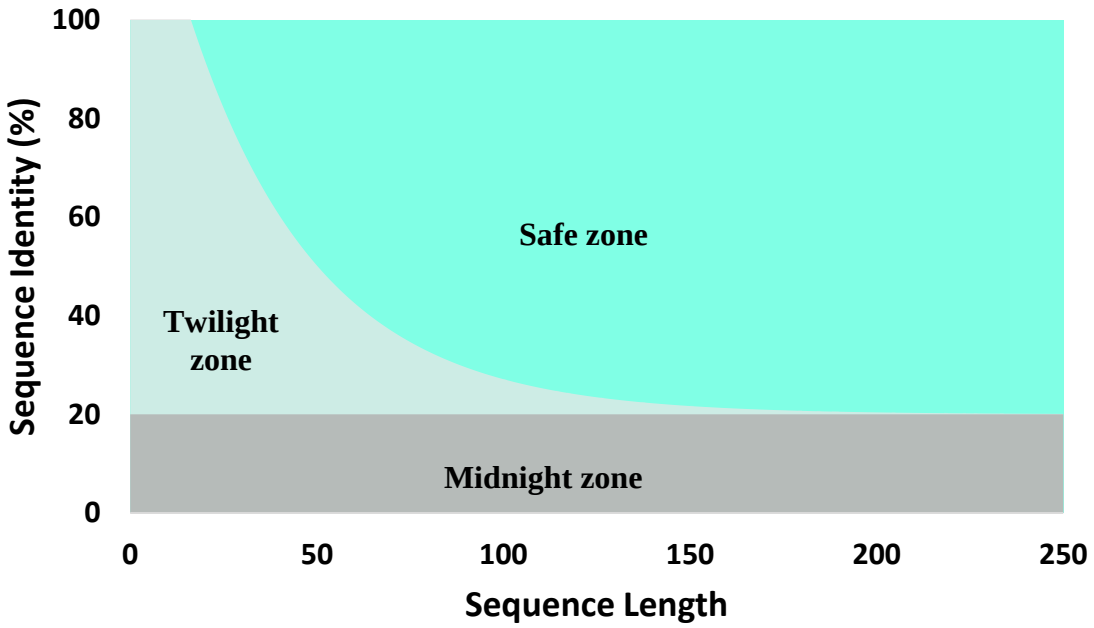


Fig. 13. The three zones of protein sequence alignments. A safe zone where homologous relationship is confident. Sequence identity values below the safe zone boundary, but above 20%, are considered to be in the twilight zone, where homologous relationships are less certain. The region below 20% is the midnight zone, where homologous relationships cannot be reliably determined. (Source: Modified from [112]).

3.2.1 Pairwise Alignment

Pairwise sequence alignment aims to find the best pairing of two sequences, such that there are maximum number of correspondences among residues. There are two alignment strategies that are often used: global alignment and local alignment. In global alignment, the alignment is carried out from beginning to end of both sequences to find the best possible alignment between the two sequences across the entire length. Alternatively, local alignment only finds local regions with the

highest level of similarity between the two sequences and aligns these regions regardless to the rest of the sequence. Fig. 14 shows the differences between global and local pairwise alignment.

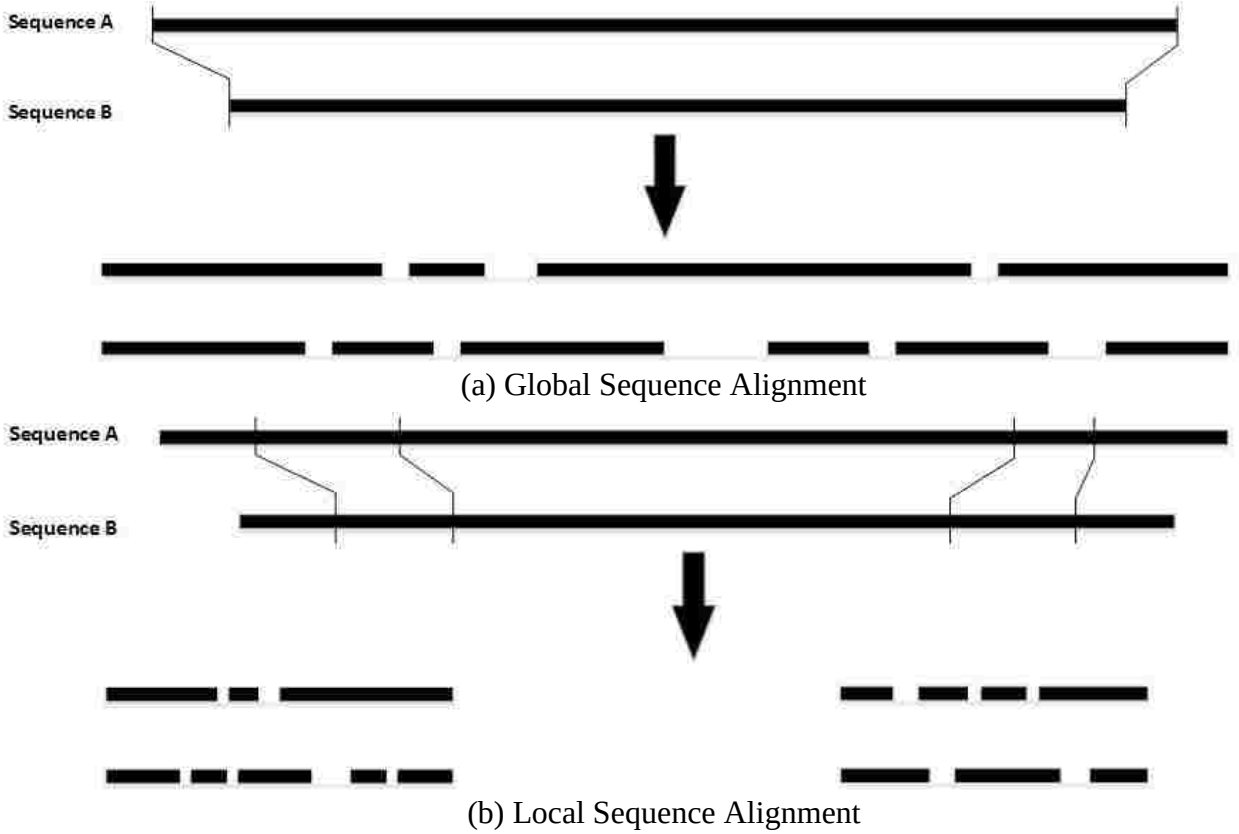


Fig. 14. An example of pairwise sequence comparison showing the distinction between global and local alignment. The global alignment (a) includes all residues of both sequences. The local alignment (b) only includes portions of the two sequences that have the highest regional similarity.

Both global and local alignment algorithms are fundamentally similar, the only difference is the optimization strategy used in aligning similar residues. The two algorithm types can be implemented based on three methods: the dot matrix method, the dynamic programming method, and the word method.

3.2.1.1 Dot Matrix Method

The graphical dot matrix was first introduced in [113] and [114] as a sequence analysis technique. Afterwards, dot-matrix plot was among the most popular methods for analyzing sequence similarity. The dot plot method concept is rather basic, as a graphical way to compare

two sequences. In a dot matrix, one of the sequences is written along the horizontal axis and the other along the vertical axis, then a dot is placed where two residues match. Regions of similarity between two sequences will appear as many dots lining up to form diagonal lines. These diagonal lines reveal the sequence alignment, where interruptions in the middle of the diagonal line indicate insertions or deletions as shown in Fig. 15.

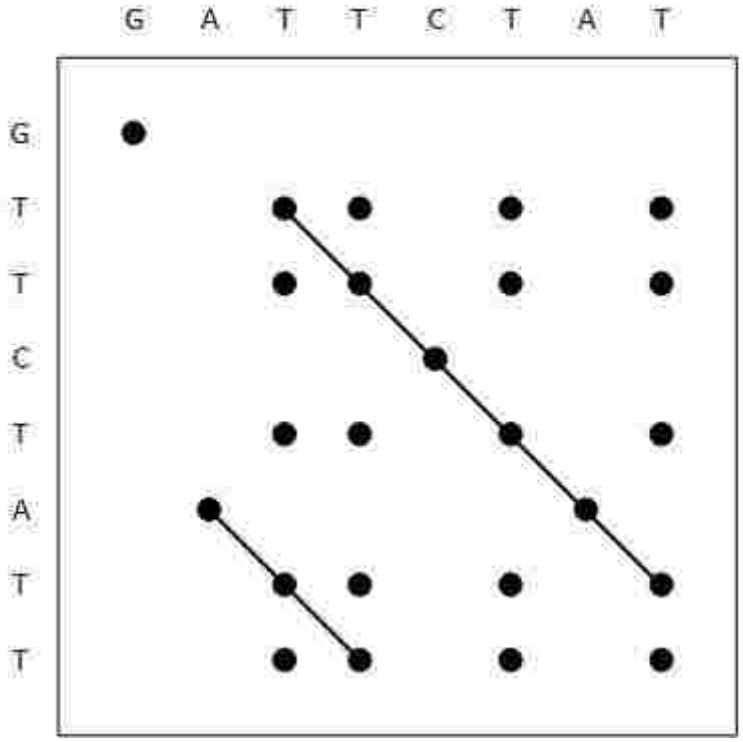


Fig. 15. An example of dot plot method for aligning two sequences, where the dots in diagonal line indicate sequence alignment. The diagonal line below the main diagonal represent internal repeats of either sequence.

The problem with dot matrix method lays when aligning large sequences, spurious matches give rise to a background of single dots. The background dots will obscure the identification of the true alignment. A standard technique to deal with such a problem is by applying a filter window along the diagonals, which keep only dots in the center of the window when their sum exceeds a threshold. The difficulty in the filter window technique is finding the right threshold, as a wrong threshold may result in dot-plots with either too much noise or being lack of the relevant diagonals.

Probabilistic methods have been used to estimate the threshold in several approaches in the 1970s and 80s [115] [116] [117] [118].

Computationally, dot matrix has a problem in the execution time, as it is proportional to the product of the lengths of the sequences. Some algorithms dealt with this problem using heuristic approaches [119] [120], while others combined trees with heuristics [121]. These techniques improve the computation time at the cost of generating a dot-matrix that may not be entirely correct.

An improvement to the initial dot-plot is to encode a score for the dot instead of single-bit dot (either on or off). Two different encoding methods have been used: by color [122] [123] [124], or by lines of varying thickness [125]. Initially, these score encoding techniques were only able to encode 16 different colors or shapes. Later, with the newer graphics hardware allowed the employment of 128 different greyscale colors [126].

The advantage of the dot matrix approaches is that it displays all possible sequence matches, but it does not generate a full alignment. Additionally, it lacks statistical rigor in assessing the quality of the alignment. Hence, dot matrix is considered more as a pairwise sequence comparison tool, rather than an alignment approach. Several sequence alignment visualization programs have been developed based on dot plot. One of these programs is DOTTER [126], which allows segments from the BLAST suite of searching programs to be superimposed on top of the full dot-matrix. VISTA [127] is another sequence alignment visualization program for global DNA sequence alignments. VISTA facilitates the visualization of alignments of various lengths at different levels of resolution using dot plots.

Another dot matrix methods for genome analysis were presented by Huang and Zhang [128]. In their methods, a fast search algorithm is used to identify short similar sequence regions,

then a lookup table containing all possible combinations of a word is employed. The main advantages of these methods are the linear space requirement and the efficient computation speed. Furthermore, a variety of genome sequence analysis and visualization based on dot-plot were introduced, such as GenomeMatcher [129] and MAFFT [130]. GenomeMatcher is a DNA sequence comparison software with graphics user interface that uses two sequence alignment software: BLAST and MUMmer [131]. MAFFT is essentially a multiple sequence alignment software that generates dot plots between the first sequence and the remaining sequences.

3.2.1.2 Dynamic Programming

Dynamic programming methods are the only alignment methods that are capable of determining the optimal alignments. Before dynamic programming methods, the naïve approach to find the optimal alignment of two sequences is to generate all possible alignments, calculate the score for each alignment, then select the alignment with the highest score. For two sequences of 100 residues, there are more than 10^{75} alignments. Hence, generating all these alignments will be both time and space consuming. Fortunately, the development of dynamic programming alignment algorithms allowed the generation of optimal alignments in only mn steps, where m is the length of one of the two sequences and n is length of the other.

Dynamic programming sequence alignment algorithms were first introduced by Needleman and Wunsch [132] for aligning protein sequences, though similar methods independently developed in the late 1960s for the speech processing and computer science fields [133]. Typically, a dynamic programming alignment algorithm uses a substitution matrix to assign scores to amino-acid matches or mismatches, and a gap penalty for matching an amino acid in one sequence to a gap in the other. Once the substitution matrix is completed, the optimal alignment is identified by tracing back through the matrix in reverse order from the lower right-hand corner of

the matrix toward the origin of the matrix. The optimal alignment is the best matching path that holds the maximum total score. The Needleman-Wunsch is explained briefly with an example later in this section

Originally, the Needleman-Wunsch algorithm was developed to find similarities between two protein sequences. It is also applied to statistical tests of relatedness between pairs of sequences by Dayhoff [134]. In 1972, Sankoff introduced another sequence alignment dynamic programming algorithm [135]. Sankoff's algorithm is similar to Needleman's, where the main difference is the introduction of the deletion/insertion (DI) constraint as another indication of similarity between two sequences. Sankoff illustrated that a low match scoring alignment that holds a low DI value may be better than a higher match scoring alignment that suffers a higher DI.

Later, Sellers modified Needleman-Wunsch's algorithm by combining it with Sankoff's to measure the divergence between two sequences [136]. Subsequently, Smith-Waterman extended Sellers' algorithm so that deletion/insertion gaps of any length are allowed [137]. The inclusion of varying length gaps is valuable for comparing protein sequences, since a long gap can be produced from a single deletion/insertion event. Smith-Waterman's algorithm with varying length gap feature is performed by assigning a weight $w_k \leq kw_1$ to a gap of length k, whereas the gap weight is restrained by $w_k = kw_1$ for all k values. Later, Smith-Waterman extended the algorithm to find local alignment between two sequences [138]. In his approach, Smith-Waterman defined local alignment as a pair of segments, one from each of two long sequences, such that there is no other pair of segments with greater similarity.

Despite the effectiveness of Smith-Waterman's algorithm, it has a drawback as its computation requires m^2n steps. Such a drawback is a serious limitation on the algorithm due to the limited computation powers of computers at that time. Accordingly, Gotoh improved Smith-

Waterman's algorithm by computing the divergence of the two sequences in mn steps, then generate the alignment in a second pass, which makes the overall algorithm steps $2mn$ [139]. Another modification in Gotoh's algorithm was in using the affine gap cost, which requires the gap weight function to be $w = uk + v$ where k is the gap length, opening a gap costs u, and each null in the gap costs u. Gotoh further showed that if the gap weights are limited by $w = ul + v$ where $k > l$ for long gaps, the computation could be completed in two passes of $(l + 2)mn$ steps each.

Additionally, Gotoh's algorithm attempts to find only one of the optimal alignments rather than all. However, this single alignment occasionally fails to be optimal. Taylor introduced a modification of Gotoh's algorithm that always finds at least one optimal alignment [140]. The disadvantage of Taylor's algorithm is that its storage requirements depend on the length of the longest gap to be allowed. Another modification of Gotoh's algorithm was presented by Altschul, which finds all the optimal alignments of two sequences in mn steps [141]. After Altschul's algorithm, there have been several other attempts to improve the computation and space requirements for dynamic programming sequence alignment algorithms [142] [143] [144] [145] [146] [147]. However, the recent advancement in computer systems has made these improvements to the original Needleman-Wunsch algorithm pointless. Hence, the most utilized algorithms for pairwise sequence alignments in modern research are the Needleman-Wunsch algorithm for global alignment and the Smith-Waterman algorithm for local alignment.

- **Needleman-Wunsch Algorithm**

Needleman-Wunsch algorithm is a dynamic programming sequence alignment algorithm. The main advantage of Needleman-Wunsch is its capability of computing the optimal sequence alignment for a pair of sequences. Originally, Needleman-Wunsch algorithm was developed to

find similarities between two protein sequences. However, the methodology can be applied to any kind of sequences.

The fundamental idea of the Needleman-Wunsch is to build an alignment scoring matrix (F) for a given pair of sequences, $A = a_1 a_2 \dots a_M$ and $B = b_1 b_2 \dots b_N$, where a_m represent the columns and b_n the rows. The concept behind the Needleman-Wunsch is that the optimal alignment can be determined by incremental extension of the optimal sub-alignments. Each cell $F_{m,n}$ represents the maximum similarity score between subsequence of A of length m, and the subsequence of B of length n. The score for cell $F_{m,n}$ depends on the three corresponding cells ($F_{m-1,n-1}$, $F_{m-1,n}$, and $F_{m-1,n-1}$) and is calculated as follow:

$$F_{m,n} = max \begin{cases} F_{m-1,n-1} + S_{m,n} & Match/Mismatch \\ F_{m-1,n} + g & Deletion \\ F_{m,n-1} + g & Insertion \end{cases} \quad (3.1)$$

where g is a gap penalty and $S_{m,n}$ is the score for matching the two amino acid pairs a_m and b_m. The cells in F are generated one cell at a time starting from one at the up left corner. Once all cells in F are filled, $F_{M,N}$ corresponds to the optimal alignment between sequences A and B. This optimal alignment can be generated by tracing F backward from $F_{M,N}$ to the origin following the pathway that leads to maximum similarity score.

Example:-

Given two sequences:

A = CTTAACT

B = CGGATCAT

Building an alignment using Needleman-Wunsch algorithm based on the following scores:

Match = 1

Mismatch = -1

Gap = -1

Fig. 16 shows the steps of Needleman-Wunsch's algorithm, where the optimal alignment generated according to the given scores is:

C T T A A C - T

C G G A T C A T

First, the score matrix is initialized by assigning zero to the first cell ($F_{0,0}$) (Fig. 16 (a)). Second, the maximum alignment score is calculated for each cell using equation (3.1) as shown in Fig. 16 (a) & (b)), where an arrow is drawn to indicate the origin of the cell score (match/mismatch, insert, or delete). Finally, tracing F backward from the lower left cell to the origin following the pathway indicated by the arrows will lead to generating the alignment (Fig. 16 (d)).

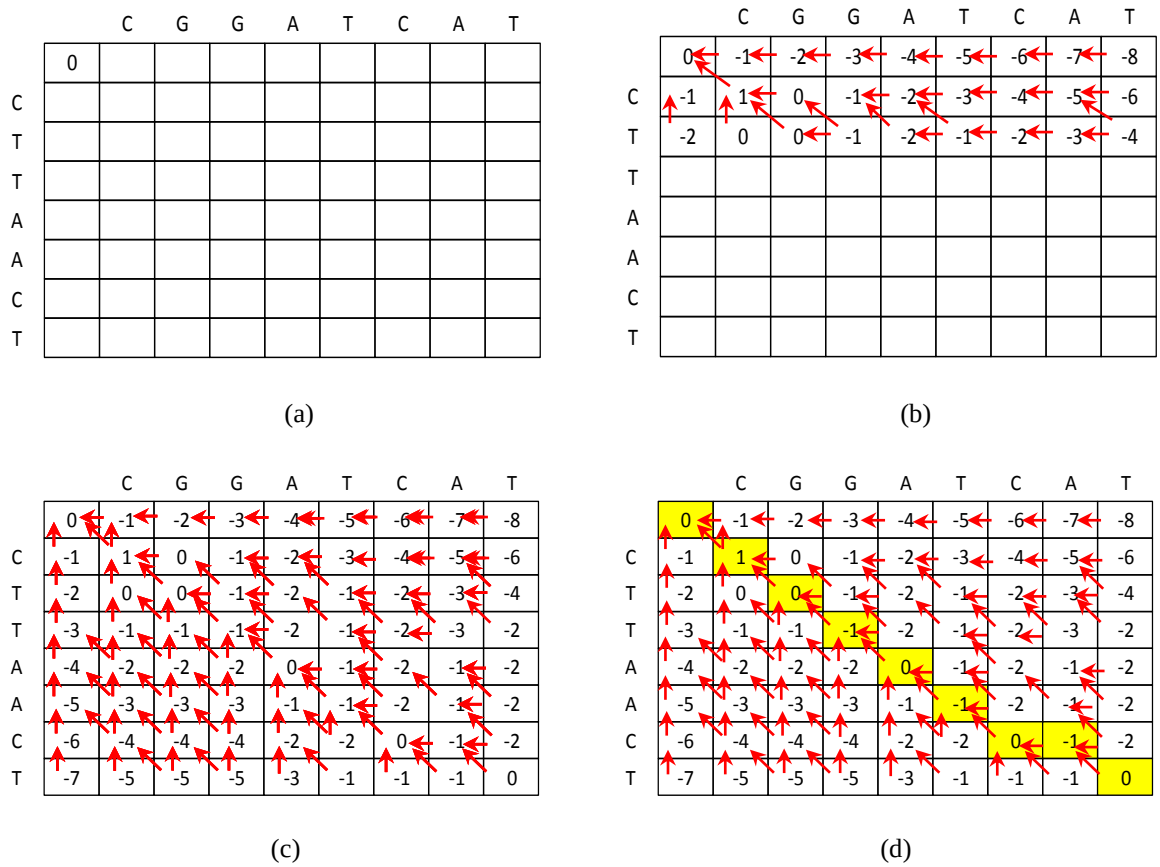


Fig. 16. Generating the optimal alignment between sequences A= CTTAACT and B=CGGATCAT using Needleman-Wunsch algorithm; (a) Initializing the scoring matrix;(b) and (c) Computing the scoring matrix; (d) Back-tracing the scoring matrix to generate the optimal alignment.

- **Smith-Waterman's Algorithm**

Frequently, the divergence level between two sequences to be aligned is not easy to be identified. Additionally, the lengths of the two sequences may be different from one another. In these cases, identification of regional sequence similarity may be of greater importance than finding an alignment that includes all residues. The first dynamic programming algorithm for local alignment is the Smith-Waterman algorithm. In the Smith-Waterman algorithm, the longest segment pair between two sequences that yields the optimal alignment is recognized by comparing all possible segments of all lengths between the two sequences using dynamic programming technique.

The main difference between the Smith-Waterman algorithm and Needleman-Wunsch's is that negative scores are set to zeros. Therefore, the backtracking process starts at the highest positive score cell and proceeds until it encounters a zero score cell.

In Smith-Waterman, the alignment scoring matrix (F) for a given pair of sequences A and B is calculated as follows:

$$F_{m,n} = max \begin{cases} 0 & \\ F_{m-1,n-1} + S_{m,n} & Match/Mismatch \\ F_{m-1,n} + g & Deletion \\ F_{m,n-1} + g & Insertion \end{cases} \quad (3.2)$$

where g is a gap penalty and $S_{m,n}$ is the score for matching the two amino acid pairs a_m and b_m.

Example:-

Given two sequences:

A = CTTAACT

B = CGGATCAT

Building an alignment using Smith-Waterman algorithm based on the following scores:

Match = 2

Mismatch = -1

Gap = -1

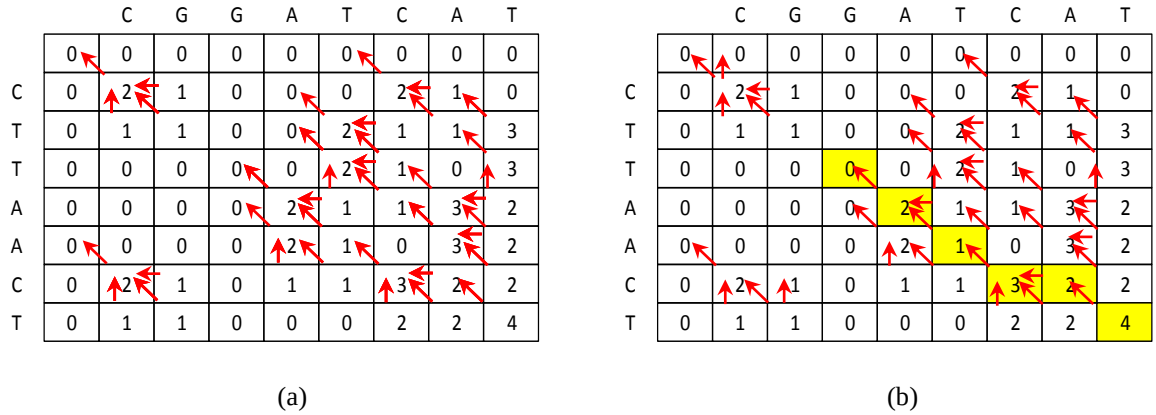
Fig. 17 shows the steps of Smith-Waterman's algorithm, where the optimal local alignment generated according to the given scores is:

A A C - T

A T C A T

As in the Needleman-Wunsch algorithm, the first cell of the matrix is assigned zero. Then, the value for each cell in the matrix is calculated using equation (3.2) as shown in Fig. 17 (a),

where an arrow is drawn to indicate the origin of the cell score (match/mismatch, insert, or delete). Finally, the local alignment is generated by backtracking the highest positive score cell to the first encountered zero score cell (Fig. 17 (b)).



(a) (b)

Fig. 17. Generating the optimal local alignment between sequences A= CTTAACT and B=CGGATCAT using Smith-Waterman algorithm; (a) Computing the scoring matrix; (b) Back-tracing the scoring matrix to generate the local alignment

3.2.1.3 Word Methods

Word methods, also known as the *k*-tuple methods [148], are heuristic methods that work on discovering a series of short and non-overlapping subsequences (word, *k*-tuple). The basic assumption is that two related sequences must have at least one word in common. Since these methods are heuristic methods, they do not guarantee that an optimal alignment can be found. However, they are faster and significantly more efficient than the alignment algorithms based on dynamic programming. Hence, they are useful for large-scale database searches where it is assumed that most of the candidate sequences do not have significant similarity with the target sequence. The most known implementations of word methods are the database search tools FASTA [149] and BLAST [84].

In general, the word methods work as follows: 1) identify a series of short non-overlapping subsequences (words) of size *k* in the target sequence; 2) match these words to candidate database

sequences; and 3) obtain a longer alignment by extending similarity regions from the words after identifying word matches. Once the regions of high sequence similarity are found, adjacent similarity regions can be joined into a full alignment.

In FASTA, k is defined by the user. It is a slower method but more sensitive at lower values of k. Thus it is preferred for searches involving a very short target sequence. In the BLAST family of search methods, a number of algorithms are available for specific types of targets. Unlike FASTA, BLAST uses a fixed default word size that is optimized according to the target and database type. Also, BLAST only evaluates the most significant word matches, rather than every word match as FASTA does. Consequently, BLAST is faster than FASTA but is not as accurate.

3.3 Multi-Objective Alignment

3.3.1 Multi-Objective Optimization

Optimization is the process of finding the most feasible solution which corresponds to the maximum/minimum value of a given objective function. When an optimization problem involves more than one objective function, the task of finding the optimal solutions is known as multi-objective optimization. As most of the real world problems involve multiple objectives, multi-objective optimization has gained lots of popularity in the last decades and has been applied in many fields, including engineering, economics, and logistics [150].

In multi-objective optimization, optimal solutions need to be found in the presence of trade-offs between two or more conflicting objectives. Thus, no single solution exists that optimizes all the objectives. A famous example of multi-objective optimization is decision making involving buying a car, where minimizing car cost and maximizing car comfort are the conflicting objectives involved in the decision. In this problem, the best solution for each objective is totally different from the other one (solution 1 and 2), as shown in

Fig. 18. Between these two extreme solutions, there exist many other solutions, where a tradeoff between cost and comfort exists (solutions A, B, and C). In this problem, all solutions laying on the curve are special in the context of multi-objective optimization and are called Pareto optimal solutions.

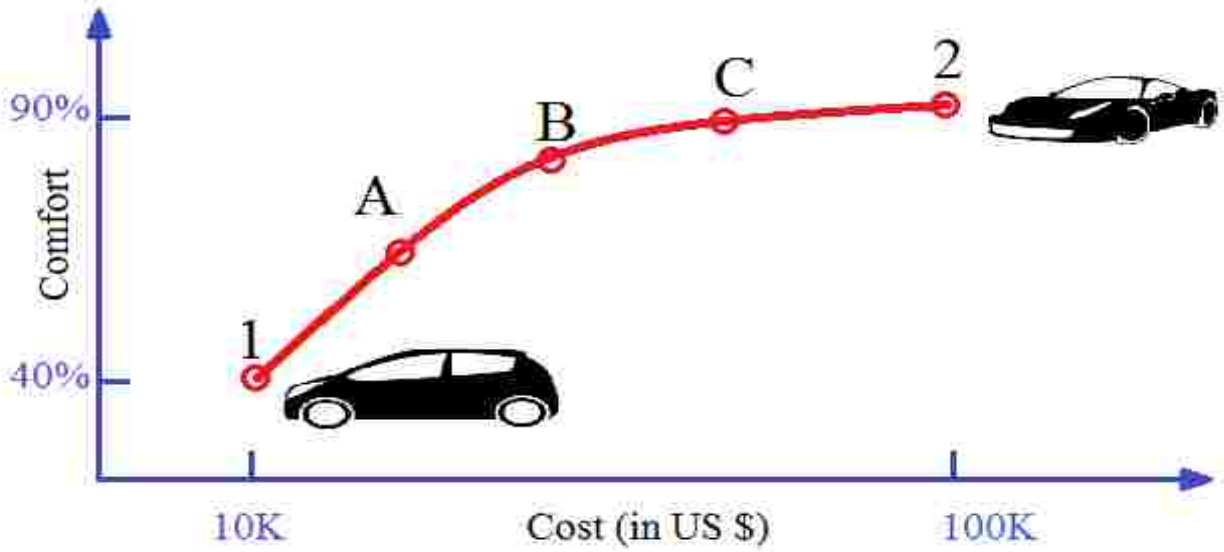


Fig. 18. Hypothetical trade-off solutions for a car buying decision-making problem (modified from [150])

3.3.1.1 Pareto Optimality

In multi-objective optimization problems, no single solution exists that simultaneously optimizes all objectives. A solution is non-dominated if none of the objective functions can be improved in value without deteriorating some of the others. In other words, given a set of objective functions $f_1(.), \ldots, f_s(.)$, without loss of generality, assuming that maximization is the optimization goal for all objective functions, a solution u is considered to dominate another alignment v ($u \prec v$) if both conditions i) and ii) are satisfied:

i) for each objective function $f_i(.)$, $f_i(u) \geq f_i(v)$ holds for all i; and

ii) there is at least one objective function $f_j(.)$ where $f_j(u) > f_j(v)$ is satisfied.

All the non-dominated solutions form the Pareto-optimal set. All Pareto-optimal solutions form the Pareto-optimal front. Fig. 19 shows the Pareto-optimal front for four different combinations of two types of objectives. Each objective can maximized or minimized.

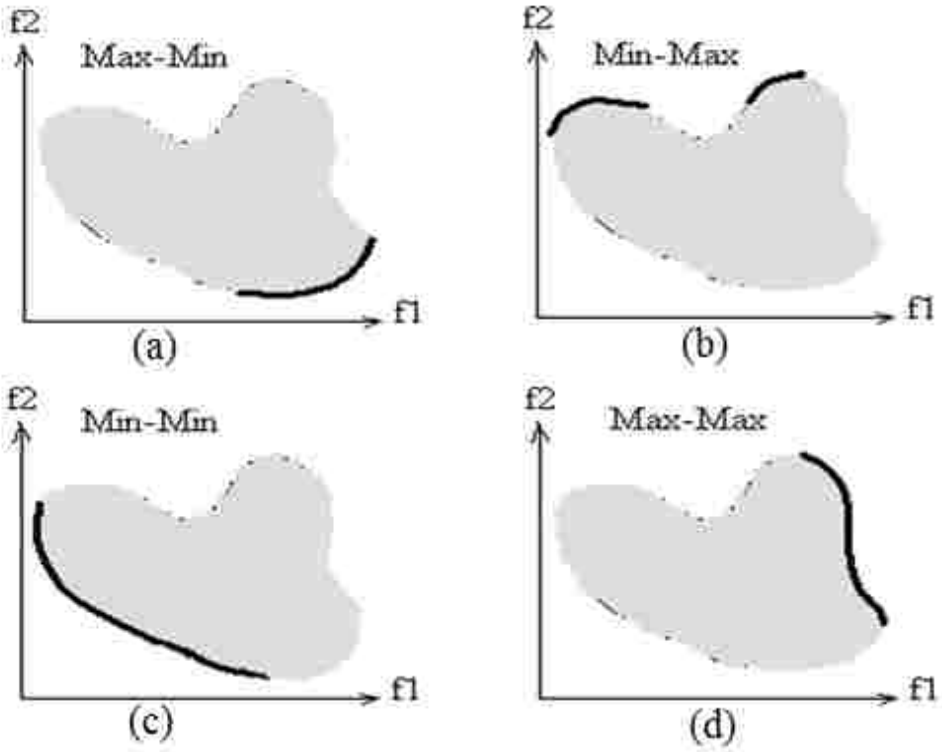


Fig. 19. Pareto-optimal front solutions for four combinations of two types of objectives (a) the task is to maximize f1 and minimize f2,(b) the task is to minimize f1 and maximize f2, (c)the task is to minimize both f1 and f1, and (d) the task is to maximize both f1 and f2(modified from [151]).

3.3.2 Multi-objective Protein Sequence Alignment

The most popular approach for protein sequence alignment is Dynamic Programming [132] [152] [153], which relies on the scheme to score the equivalence of each of the pairs of amino acids. These scoring schemes are calculated based on one or more objectives. The objectives that are most taken into consideration for protein sequence alignment are: sequence profile alignment,

secondary structures, solvent accessibility, backgone torsion angles, and fragments. The most successful protein alignment algorithms integrated all these objectives [11] [83] [91] [154] [15]. These algorithms are acknowledged as multi-objective alignment; however none has been able to generate the entire Pareto-optimal front for a pairwise sequence alignment under these objectives. The previous research of multi-objective pairwise sequence alignment are mainly implemented using one of two techniques: 1) performing linear combinations of more than one objective score to generate one objective function and use it to generate one alignment [15] [154]; 2) Using an evolutionary algorithm which generates an initial population of solutions, modifies those solutions using a set of genetic operators, and evaluates the quality of those solutions using a set of objective functions to keep only the dominant ones and eliminate the others [155] [156]. For the first technique this is not quite a multi-objective way as it treats the multi-objective problem as a single objective function. Besides it will only generate one solution under the combined objective function. For the second technique there is no guarantee that the generated alignments are the optimal and complete.

3.3.2.1 Protein Sequence Alignment objectives

Each of the objectives used in protein sequence alignment relies on a scoring system, which quantifies the likelihood of one amino acid being substituted by another in an alignment. This section will explain the most used objectives for protein sequence alignment and their scoring systems.

- **Sequence Profile**

Initially aligning a pair of protein sequences relies only on a system to score the equivalency of each of the 210 possible pairs of amino acids. The simplest scoring system identifies amino acids as identical and non-identical, where identical pairs are given a positive

score and non-identical pairs are scored zero. Such a scoring system is generally considered inefficient. Such systems represent the 210 pairs as a 20×20 substitution matrix where identical and similar pairs of amino acids are given higher scores that other pairs of amino acids.

In 1978, Dayhoff et al. [157] developed the first amino acid scoring matrix that reflects their physicochemical properties. The developed matrices are known as the PAM (Point Accepted Mutation) matrices, which observe the amino acid mutations that are not expected to significantly change the function of proteins.

The PAM1 matrix is developed from the substitution frequency of proteins 1PAM from each other, where two sequences are within 1PAM distance if they can be converted into each other (very similar). 1PAM distance is defined as 1% of the amino acid positions that have been changed. The PAM2 matrix is calculated by multiplying PAM1 matrix by itself, then the PAM3 by multiplying the PAM2 by PAM1, and so on. The bigger the number of the matrix is, the more suitable it is for the more divergent sequences (Table 2). Fig. 20 shows the PAM 250 substitution matrix.

Table 2
The Correspondence of PAM Numbers with the observed percent of amino acid evolutionary distance

PAM Number	**The observed amino acid Distance (%)**	**Sequence Identity**
0	0	100
1	1	99
30	25	75
80	50	50
110	40	60
200	75	25
250	80	20

C	12																			
S	0	2																		
T	-2	1	3																	
P	-3	1	0	6																
A	-2	1	1	1	2															
G	-3	1	0	-1	1	5														
N	-4	1	0	-1	0	0	2													
D	-5	0	0	-1	0	1	2	4												
E	-5	0	0	-1	0	0	1	3	4											
Q	-5	-1	-1	0	0	-1	1	2	2	4										
H	-3	-1	-1	0	-1	-2	2	1	1	3	6									
R	-4	0	-1	0	-2	-3	0	-1	-1	1	2	6								
K	-5	0	0	-1	-1	-2	1	0	0	1	0	3	5							
M	-5	-2	-1	-2	-1	-3	-2	-3	-2	-1	-2	0	0	6						
I	-2	-1	0	-2	-1	-3	-2	-2	-2	-2	-2	-2	-2	2	5					
L	-6	-3	-2	-3	-2	-4	-3	-4	-3	-2	-2	-3	-2	4	2	6				
V	-2	-1	0	-1	0	-1	-2	-2	-2	-2	-2	-2	-2	2	4	2	4			
F	-4	1	-3	-5	-4	-5	-4	-6	-5	-5	-2	-4	-5	0	1	2	-1	9		
Y	0	-3	-3	-5	-3	-5	-2	-4	-4	-4	0	-4	-4	-2	-1	-1	-2	7	10	
W	-8	-2	-5	-6	-6	-7	-4	-7	-7	-5	-3	-2	-3	-4	-5	-2	-6	0	0	17
	C	**S**	**T**	**P**	**A**	**G**	**N**	**D**	**E**	**Q**	**H**	**R**	**K**	**M**	**I**	**L**	**V**	**F**	**Y**	**W**

Fig. 20. PAM250 amino acid substitution matrix

BLOSUM (blocks substitution matrix) matrices are another amino acid substitution matrix that were developed by Henikoff and Henikoff in 1992 [158]. BLOSUM matrices were derived using local multiple alignments of homologous proteins. Similar to PAM, BLOSUM were constructed as a series of matrices. The BLOSUM matrix index represents the percentage of the identity values of sequences selected to develop the matrix. Hence, the BLOSUM-N matrix is developed from sequences sharing N% identity. Unlike PAM, the greater the matrix index the more suitable it is for the more similar sequences. Fig. 21 shows the BLOSUM62 substitution matrix, which has been popularly used in a lot of bioinformatics applications.

C	9																			
S	-1	4																		
T	-1	1	5																	
P	-3	-1	-1	7																
A	0	1	0	-1	4															
G	-3	0	-2	-2	0	6														
N	-3	1	0	-2	-2	0	6													
D	-3	0	-1	-1	-2	-1	1	6												
E	-4	0	-1	-1	-1	-2	0	2	5											
Q	-3	0	-1	-1	-1	-2	0	0	2	5										
H	-3	-1	-2	-2	-2	-2	1	-1	0	0	8									
R	-3	-1	-1	-2	-1	-2	0	-2	0	1	0	5								
K	-3	0	-1	-1	-1	-2	0	-1	1	1	-1	2	5							
M	-1	-1	-1	-2	-1	-3	-2	-3	-2	0	-2	-1	-1	5						
I	-1	-2	-1	-3	-1	-4	-3	-3	-3	-3	-3	-3	-3	1	4					
L	-1	-2	-1	-3	-1	-4	-3	-4	-3	-2	-3	-2	-2	2	2	4				
V	-1	-2	0	-2	0	-3	-3	-3	-2	-2	-3	-3	-2	1	3	1	4			
F	-2	-2	-2	-4	-2	-3	-3	-3	-3	-3	-1	-3	-3	0	0	0	-1	6		
Y	-2	-2	-2	-3	-2	-3	-2	-3	-2	-1	2	-2	-2	-1	-1	-1	-1	3	7	
W	-2	-3	-2	-4	-3	-2	-4	-4	-3	-2	-2	-3	-3	-1	-3	-2	-3	1	2	11
	C	**S**	**T**	**P**	**A**	**G**	**N**	**D**	**E**	**Q**	**H**	**R**	**K**	**M**	**I**	**L**	**V**	**F**	**Y**	**W**

Fig. 21. BLOSUM62 amino acid substitution matrix

In 1987 Gribskov [159] suggested the use of protein **profiles** as scoring system for alignment instead of substitution matrices. A profile is a Position Specific Scoring Matrix (PSSM) that contains probability information of amino acids. The profile resembles the substitution matrices, but is more complex as it contains position information. In a profile matrix, the rows represent amino acid positions of particular multiple alignments and the columns represent the amino acids. The values in the matrix represent the log odds scores of the amino acids calculated from multiple alignments. The profile scores are significantly important when aligning distantly related protein sequences. PSI-BLAST [87] and Hidden Markov Models (HMM) [160] [161]

represent two popular methods for generating a protein profile based on the multiple sequence alignments among homologous proteins of a target sequence.

- **Secondary Structure**

Secondary structure refers to the general three-dimensional form of the protein local segments, where it is classified into three classes: alpha helix, beta sheet, and coil. To perform sequence alignment for sequences with low profile similarities, research consider using protein structure information [162] [163] [164]. The secondary structure score is a comparison between the secondary structure of the target amino acids and that of the template amino acids. Since the structure of the target sequence is not known, a secondary structure prediction method, such as PSI-PRED [165] and SCORPION [166], is often employed to obtain the predicted secondary structure probability.

- **Solvent Accessibility**

Solvent accessibility of an amino acid refers to the amino acid tendency of exposing to water, where amino acids can be classified to either exposed or buried. The integration of solvent accessibility information in sequence alignment improved the alignment accuracy [83] [154]. Similar to secondary structure, solvent accessibility score is a comparison between the solvent accessibility of the target amino acids and that of the template amino acids. Consequently, a solvent accessibility prediction method is needed to predict the solvent accessibility of the amino acids of the target protein sequence, such as Hopp-Woods method [167], Kyte-Doolittle method [168], and CASA [166].

- **Fragments**

In any short amino acid sequence segment, the molecular interactions constrain the structure into a small number of conformations. These conformations can be modeled as protein

fragments, which are distributed across many protein structures from different families. Several successful protein alignment algorithms have incorporated fragment information [11] [15] and shown noticeable improvement. The constructed fragments are used to build a frequency profile at each position of the template, which is calculated by aligning the fragments sequences at each position on the template. Thus, the fragment score is the frequency of the template amino acid to appear on the fragment sequences corresponding to its position on the template sequence.

3.4 Summary

In this chapter, we presented a review of the relevant literature to the protein template-based modeling. We presented the foundation for the template-based protein structure prediction along with an overview of the template-based protein structure prediction methods (Section 3.1). Additionally, we provided an overview of the threading and template selection techniques in Section 3.1.1 and Section 3.1.2 respectively. We also presented the related research of protein sequence alignment (Section 3.2). Toward the end, we presented an overview of the research that has been established in the multi-objective alignment and more specifically multi-objective protein sequence alignment (Section 3.3). Exploring all these research allowed us to build a knowledge of the problem addressed in this dissertation and previous solutions. Consequently, in the next chapters, we proceed with our proposed methods for template selection and multi-objective protein sequence alignment.

CHAPTER IV

TEMPLATE SELECTION APPROACHES

In this chapter, we aim at improving the template-based protein structure modeling by enhancing the correctness of identifying the most appropriate templates. Most of the template selection methods try to take advantage of multiple structural information sources to help find the optimal match between the target and the structural templates. Here, we present two template selection approaches that incorporate inter-residue contacts to enhance template selection sensitivity. Our first template selection approach combines the inter-residue contact score with the sequence profile score, which is a representation of protein structural features (Section 4.1). Our second template selection approach is a further improvement to the template based protein structure modeling. In this approach, we use the inter-residue contact score to build the alignment along with other structural features scores (Section 4.2). The template selection approaches are tested over CASP 11 targets and have shown a significant improvement compared to the successful template alignment and selection methods.

4.1 Incorporating ICOSA Score in Template Selection

When templates with high sequence identity are not available, most template selection methods try to take advantage of multiple structural information sources, such as sequence profiles, secondary structures, solvent accessibility, backbone dihedral angles, etc., to help find the optimal match between the target and the structural templates. In protein structure modeling literature [148], it is well-known that the inter-residue contacts play an important role in forming and stabilizing a protein fold. We presented an approach to evaluate the favorability of a target sequence fitting in the folding topology of a certain template by placing the target sequence residues into the mapped template residues in their three-dimensional conformation and evaluating

the contact score. Then, we combine the contact score with the sequence profile score to enhance template selection sensitivity. More specifically, we incorporate ICOSA [149], a coarse-grained contact potential correlating inter-residue interaction distance and orientation, into MUSTER [133], one of the most successful template alignment and selection methods in template-based protein structure modeling. We use the CASP11 targets to demonstrate the effectiveness of our method.

4.1.1 Methodology

4.1.1.1 MUSTER Scores

MUSTER is a template-based protein structure modeling method that works by aligning the target sequence with all potential templates in I-TASSER library and then calculating MUSTER scores of all resulted alignments to pick the most appropriate templates. The MUSTER alignment is done by dynamic programming that exploits the protein structural features including (1) sequence profiles; (2) predicted secondary structures; (3) depth-dependent structure profiles; (4) solvent accessibility; (5) backbone dihedral torsion angles; and (6) hydrophobic scoring matrix [15]. Each structural feature gives an independent score. These structural feature scores are summed along with carefully balanced weights derived by various machine learning algorithms [15] and then normalized by alignment length, which gives the final MUSTER score.

4.1.1.2 ICOSA Score of a Structural Template

ICOSA [14] is a knowledge-based, coarse-grained contact potential that correlates pairwise inter-residue interaction distance and orientation using an icosahedral tessellation. ICOSA has accuracy and sensitivity comparable to all-atom fine-grained potentials in discriminating near-natives from misfolds. In addition, ICOSA has been successfully used to fast model protein loops in sub-angstrom accuracy [169]. Due to the fact that ICOSA is a coarse-grained potential that only

Cα-Cα contacts are taken into consideration, it is capable of implicitly estimating side chain interactions via contact orientation and distance while tolerating structural imperfection (Fig. 22). Hereby, ICOSA is used to measure the favorability of a protein target when adopting the folding topology of a potential structural template.

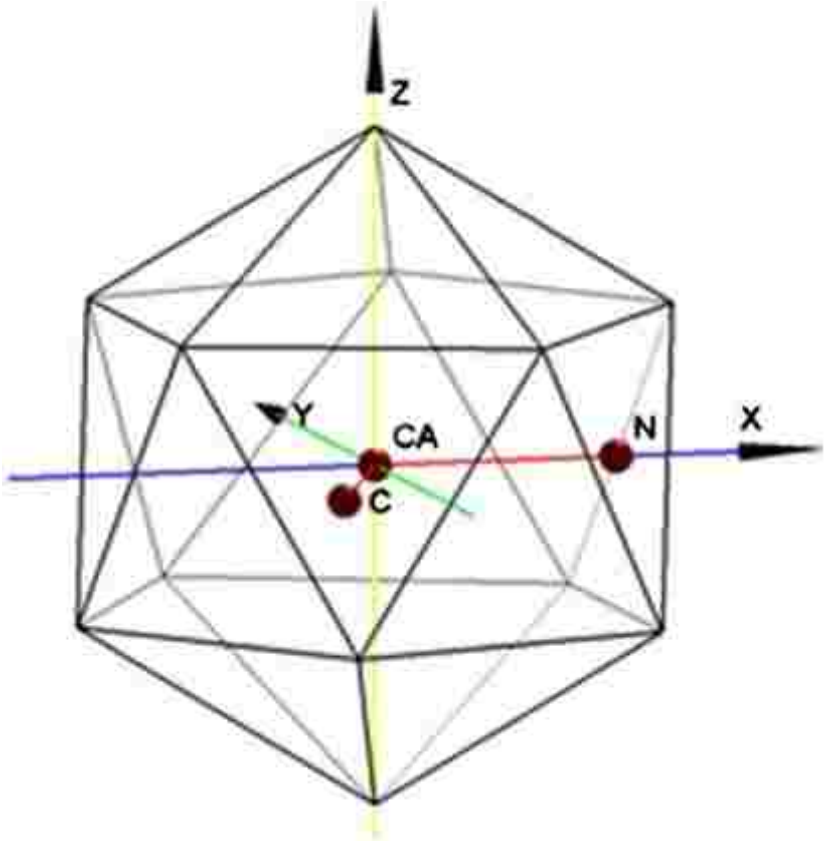
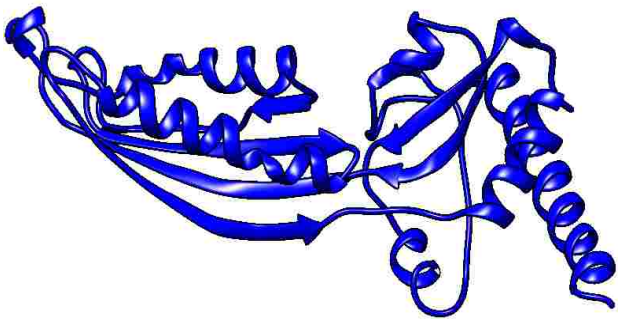


Fig. 22. Icosahedral local coordinates with CA at the origin [14].

Fig. 23 shows an example of evaluating the ICOSA score of a target adopting a potential structural template 1r43A displayed in Fig. 23(a). Fig. 23 (b) shows the MUSTER alignment of the protein sequence target to the sequence of the template. Then, the unmatched residues in the structural template are ignored while the remains are substituted by the corresponding residues in the target highlighted in orange in Fig. 23 (c). Afterwards, as shown in Fig. 23 (d), the ICOSA

score is calculated by summing the pairwise interactions of the target residues adopting the folding conformation of the structural template.



(a)

```
1r43A : ELHIEQGPILEDENKAIGIVTGVQAYNWQKVTVHGVGAHA
Target: ----------------------MLTVEVEVKITADD---

1r43A : GTTPWRLRKDALLSSKIVAASEIAQRHNGLFTCGIIDAKP
Target: -----ENKAEEIVKRVIDEVEREVQKQYPNATITRTLTRD

1r43A : YSVNIIPGEVSFTLDFRHPSDDVLATLKEAAAEFDRLIKI
Target: D------GTVELRIKVKADTEEKAKSIIKLIEERIEEELR

1r43A : NDGGALSYESETLQVSPAVNFHEVCIECVSRSAFAQFKKD
Target: KRDPNATITRTVRTEV---------------------GS

1r43A : QVRQIWSGAGHDSCQTAPHVPTSIFIPSKDGLSHNYYEYS
Target: SWSLEH----H-------------------HHHH-----

1r43A : SPEEIENGFKVLLQAIINYDNYRVIRGH*
Target: ---------------------------*
```

(b)

(c)

Pairwise Residue Interactions in Template			Pairwise Residue Interactions in Target		
V	M	-0.030	A	Q	0.152
Q	L	0.816	H	A	0.146
V	T	0.131	A	Y	0.260
E	V	0.418	I	N	-0.088
V	E	-0.072	A	W	0.212
K	V	0.735	S	Q	0.387
I	E	0.025	I	K	0.028
E	V	0.372	A	V	-0.060
A	K	0.057	K	T	-0.032
K	I	0.018	R	V	0.057
…			…		

(d)

Fig. 23. Estimation of ICOSA score for a template alignment, (a) Structural Template of 1r43A, (b) Alignment between 1r43A and target sequences based on structural profile, (c) Substitute template residues (blue) with target residues (orange), (d) Calculating template ICOSA score of substituted

For each specific target, ICOSA is applied to all structural templates found by MUSTER. A higher ICOSA score typically indicates that the protein target is more favorable in adopting the three-dimensional folding conformation of the structural template.

Since ICOSA is a contact potential measuring global inter-residue interactions, while the sequence profile alignment score in MUSTER estimates local interactions, the ICOSA score and the MUSTER score are deemed to be independent, so they can be directly added up.

4.1.2 Results

We use the Critical Assessment of Protein Structure Prediction (CASP) 11 [67] experiment targets to demonstrate the effectiveness of our method. The MUSTER program is obtained from the I-TASSER Suite [99] Version 5.1 that was released on March, 10th 2017. First of all, we use the MUSTER program to generate structural profile alignments for over 60,000 structural templates extracted from the experiment-determined protein structures available in I-TASSER library. Templates with over 25% sequence identities with the target sequences are removed. Then, we use MUSTER, ICOSA alone, and the combination of MUSTER and ICOSA (MUSTER + ICOSA) to rank the templates. The quality of a structural template is evaluated by the Global Distance Test – Total Score (GDT-TS), which indicates the percentage of the model conformation superimposed correctly onto the native structure, compared to the native structure.

The performance and comparison of MUSTER, ICOSA and MUSTER+ICOSA on the CASP 11 targets are summarized in Table 3. The average GDT-TS score of the top-ranked templates selected by ICOSA is lower than the top-ranked templates selected by MUSTER. This is mainly because ICOSA only takes three-dimensional topology into account while many other important factors such as secondary structures, solvent accessibility, backbone torsion angles conformations, and sequence similarity are not considered. However, when the ICOSA score is

combined with the MUSTER score, the average GDT-TS score of the top-ranked templates increased to 34.31.

Table 3
Overall performance of MUSTER, ICOSA, and MUSTER+ICOSAon the CASP11 targets

Method	MUSTER	ICOSA	MUSTER+ICOSA
Average GDT-TS of the top-ranked model	32.85	22.91	34.31

Fig. 24 compares the GDT-TS scores of the top-ranked templates selected by MUSTER, ICOSA alone, and MUSTER + ICOSA. CASP 11 targets where MUSTER cannot find any templates with over 20.0 GDT-TS score are ignored. One can find that when ICOSA score is combined with MUSTER score, the top-ranked templates in eight targets have enhanced GDT-TS scores than the top ones selected by MUSTER only, while worse in three targets.

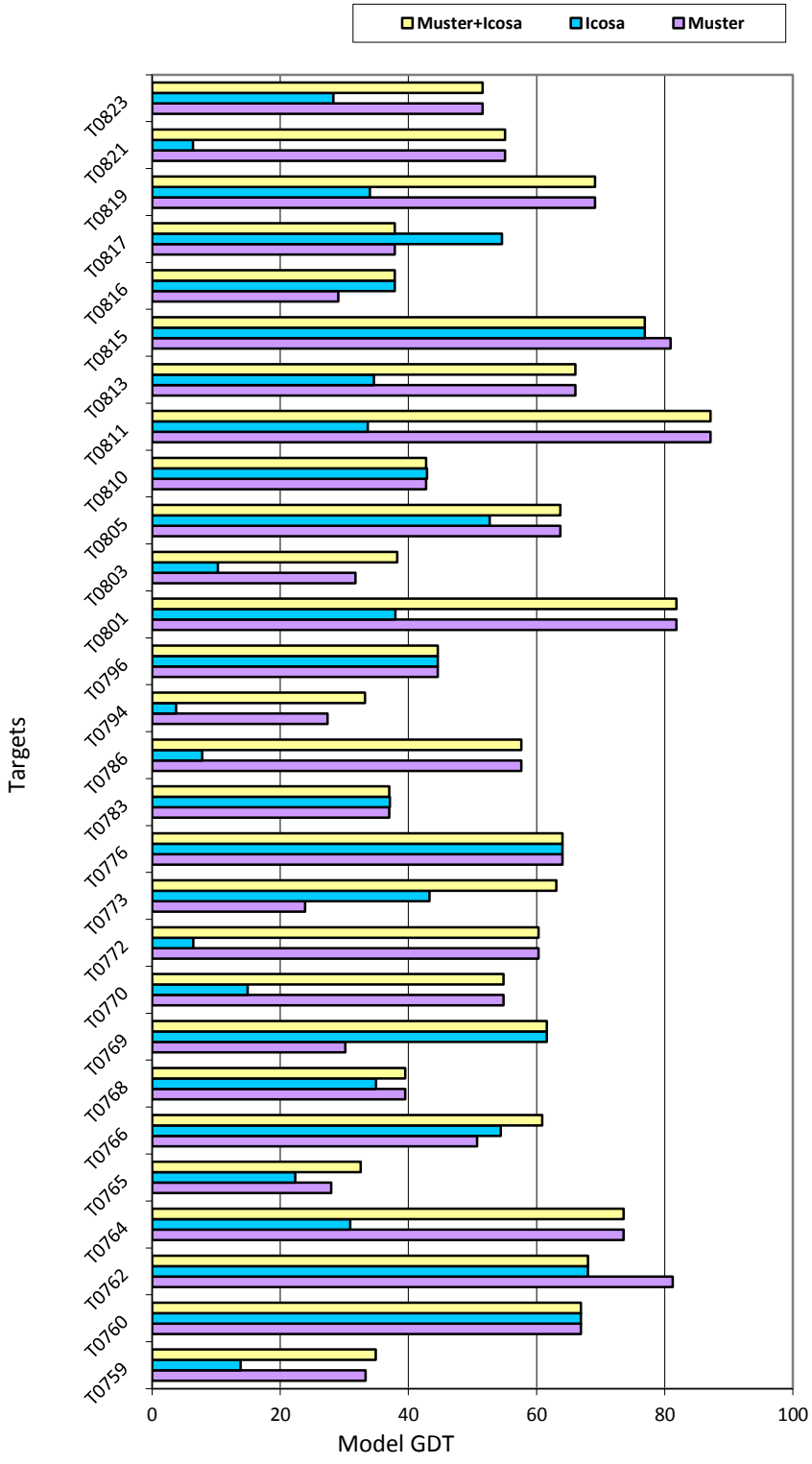
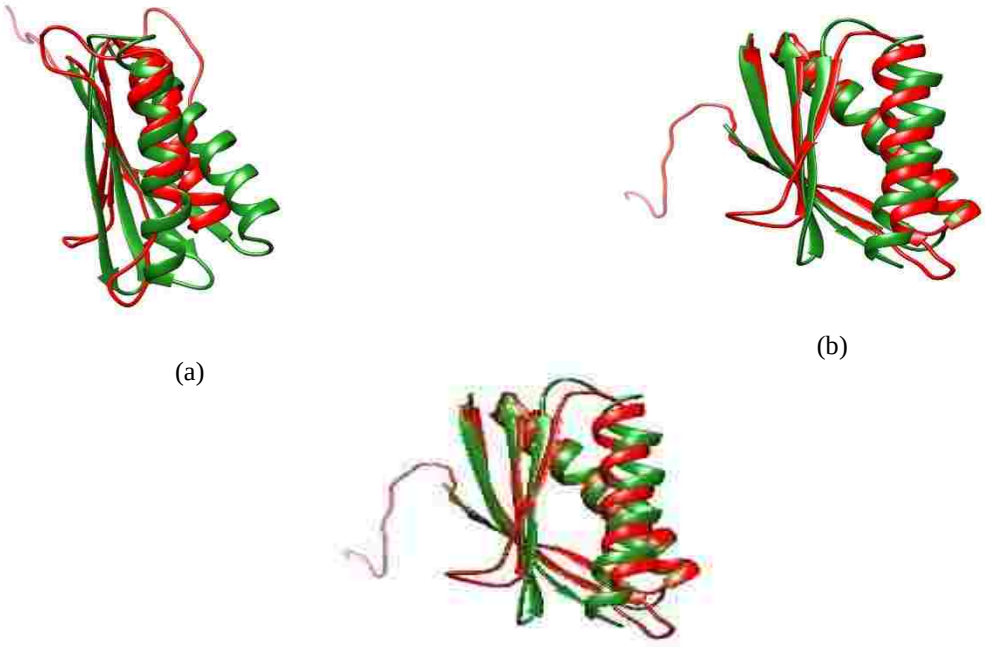
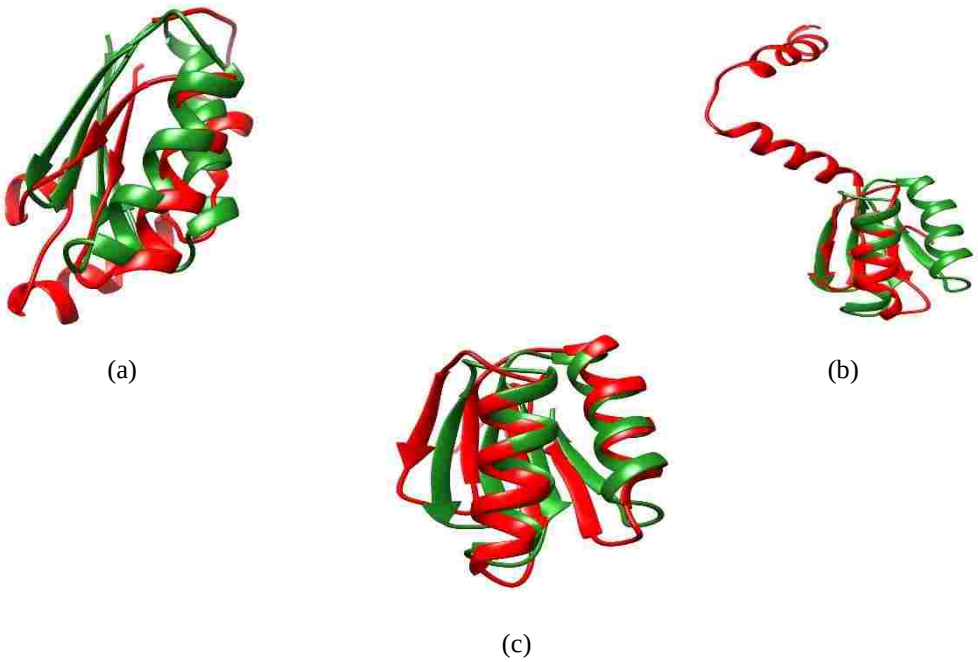


Fig. 24. The GDT-TS score of the top-ranked models selected by MUSTER, ICOSA, and MUSTER+ICOSA in CASP11 targets.



(a) (b) (c)

Fig. 25. Top-ranked templates selected by MUSTER, ICOSA, and MUSTER+ICOSA (red) in CASP11 target T0769(green),(a) top-rank template by MUSTER score,(b) top-rank template by ICOSA score, and (c) top-rank template by MUSTER+ICOSA score.



(a) (b) (c)

Fig. 26. Top-ranked templates selected by MUSTER, ICOSA, and MUSTER+ICOSA (red) in CASP11 target T0773 (green), (a) top-ranked template by MUSTER score, (b) top-ranked template by ICOSA score, and (c) top-ranked template by MUSTER+ICOSA scores

When the ICOSA score is combined with the MUSTER score, the most significant improvement occurs in targets T0769 and T0773, where the GDT-TS scores of the top-ranked templates are improved from 30.1 to 61.6 and from 23.8 to 63.1, respectively. Fig. 25 and Fig. 26 respectively display the models generated from the top-ranked templates by MUSTER, ICOSA, and MUSTER+ICOSA in targets T0769 and T0773. It is interesting to notice that in T0773, ICOSA itself is unable to identify a high-quality template; however, combining the ICOSA score with the MUSTER score leads to the identification of a significantly-improved template.

4.2 Incorporating ICOSA in Sequence Alignment

The results of in the previous section (Section 4.1) have shown the importance of using inter-residue contacts information (ICOSA score) in template selection. However, this is not the only way to integrate inter-residue contacts information in template selection. Instead of evaluating the ICOSA score of a target adopting a potential structural template after an alignment is generated, we use the ICOSA score to build the alignment along with other structural features. The idea is to build a substitution matrix to score the replacement of one amino acid of the template three-dimensional conformation with each amino acid in the target.

Then, this substitution matrix is used in building the alignment along with structural features. Accordingly, we develop a template selection approach that generates sequence alignment incorporating ICOSA (SAICOSA), then uses dynamic programming to search the alignment space for the most appropriate template. The alignment generated by dynamic programming exploits the protein features including (1) sequence profiles [87]; (2) predicted secondary structures [170]; (3) fragment profiles [171]; (4) predicted solvent accessibility [166]; and (5) ICOSA score for substituting each target amino acid in the template folding topology [14]. These protein features are summed together using weights that were carefully balanced using Grid

search technique (will be explained in Section 4.2.1.3). The resulting alignment score is a ranking score that measures the favorability of each potential template.

4.2.1 Methodology

4.2.1.1 Scoring Function

The scoring function of SAICOSA for aligning the *i*th residue on the target sequence and the *j*th residue on the template is:

$$Score(i,j) = S_{profile}(i,j) + w_1 S_{structure}(i,j) + w_2 S_{fragment}(i,j) + w_3 S_{solvent}(i,j) + w_4 S_{Icosa}(i,j) \quad (1)$$

We explain the specific terms as follows.

- **Sequence Profile**

The first term in Eq. (1) is the sequence profiles which is represented as:

$$S_{profile}(i,j) = \sum_{k=1}^{20} Fa_q(i,k) + Fb_q(i,k) L_t(j,k)/2. \quad (2)$$

where ‘‘q’’ stands for the target (query) and ‘‘t’’ for the template protein.

Here $Fa_q(i,k)$ is the frequency of the kth amino acid at the ith position of the multiple sequence alignments (MSA) obtained by PSI-BLAST [87] against the non-redundant (NR) sequence database with an E-value cutoff of 0.001. $Fa_q(i,k)$ is considered as a close alignment frequency. $Fb_q(i,k)$ is a more distant frequency generated using a higher E-value cutoff of 1.0. The idea of combining distant and close sequence profiles comes from [15] [172] [173] [174], which helps increase the alignment sensitivity in different homology areas. $L_t(j,k)$ is the derived log-odds profile of the template sequence for the *k*th amino acid at the *j*th position. The template sequence derived log-odds profile generated from PSI-BLAST search with an E-value cutoff 0.001.

- **Secondary Structure**

The second term in Eq. (1) is the probability that the predicted secondary structure of the *i*th residue of the target sequence matches with that of the *j*th residue of the template sequence, i.e.,

$$S_{structure}(i,j) = Prob\left(ss_q(i) = ss_t(j)\right) \quad (3)$$

where $ss_q(i)$ is the predicted secondary structure of the *i*th residue of the target and $ss_t(j)$ is the secondary structures of the *j*th residue of the template sequence. The secondary structure for the target is predicted by Scorpion [170] and that for the template is assigned by the DSSP program [175].

- **Fragment Profiles**

The third term in Eq. (1) is fragment profiles, which is anticipated as:

$$S_{fragment}(i,j) = \sum_{k=1}^{20} Fb_t(j,k)L_q(i,k) \quad (4)$$

The top hundred ten-residue fragments from the fragment libraries in [171] are collected and used to calculate the frequency profile at each position of the template. $Fb_t(j,k)$ is the frequency of the *k*th amino acid appearing in the 100 sequences corresponding to the *j*th position on the template. $L_q(i,k)$ is the log-odds profile for the *k*th amino acid at the *i*th position of the query sequence from the PSI-BLAST search with an E-value cutoff of 0.001.

- **Solvent Accessibility**

The fourth term in Eq. (1) is the probability that the predicted solvent accessibility of the *i*th residue of the target sequence matches with that of the *j*th residue of the template sequence, i.e.,

$$S_{solvent}(i,j) = Prob\left(sa_q(i) = sa_t(j)\right) \quad (5)$$

where $sa_q(i)$ is the predicted solvent accessibility of the *i*th residue of the target sequence and $sa_t(j)$ is the solvent accessibility of the *j*th residue of the template sequence as indicated by DSSP [175]. To predict solvent accessibility $sa_q(i)$ of the *i*th residue of the target, we use Casa program [166]. Hence, the maximum SA value in an extended tripeptide (Ala-X-Ala) is taken from [176]. The residue exposure status is defined to be either exposed or buried.

- **ICOSA**

The fifth term in Eq. (1) is the contact score calculated by ICOSA for the structural template when replacing the *j*th residue of the template by the *i*th residue of the target. ICOSA is used to score the replacement of one amino acid of the template three-dimensional conformation with each amino acid in the target. Thus, it will incorporate three-dimensional information on building the alignment along with the structural features.

4.2.1.2 Alignment Generation

The Needleman-Wunsch dynamic programming algorithm is used to build the alignment between the target and the template sequences. Gap opening (g_o); and gap extension (g_e) penalties are applied in the alignment generation.

- **Template Ranking**

After the generation of the alignment for all the templates in MUSTER database, the templates are ranked based on the raw alignment score.

4.2.1.3 Parameter Training

There are overall six parameters in SAICOSA algorithm (i.e. w_1 to w_4, g_o and g_e), which need to be carefully tuned. One of the popular alignment benchmarks is SABmark [177], which is often used in tuning methods. SABmark includes alignments that cover the entire known fold space, as classified by SCOP [77]. SABmark is a large database of more than 20,000

nonhomologous protein pairs. Using all these pairs for training will not be feasible for two reasons. First, some pairs in the SABmark database do not share a similar topology, which may mislead the training algorithm. Second, the possibility of over fitting, as the protein pairs are not uniformly distributed over the fold space. Hence, 425 pairs of proteins were selected with a TM-score [178] >0.17, and each pair belongs to one of SCOP super-families. TM-score is used to assess the topological similarity of protein structure pairs with a score in [0,1]. Statistically, a TM-score<0.17 means a randomly selected protein pair with gapless alignment taken from PDB. Accordingly, the selected protein pairs for training are categorized as: 167 pairs with a TM-score>0.5, 163 pairs with TM-score<0.5 and>0.3, and 95 pairs with TM-score<0.3 and >0.17. All the 425 pairs share the same *class*, *fold*, and *super-family* in SCOP but different *family*.

To train the template selection algorithm, one can tune the parameters by maximizing the quality of the structure model created from the generated alignment. Thus, the template selection parameters are optimized based on the overall TM-score of the resulting protein models. We use a grid-search technique, which split the 6-dimensional parameter space into lattices and try all the lattice points. In our grid search implementation a coarse-grained lattice system was used, where a finer tuning near the first selected lattice is performed. Finally, the lattice with the highest average TM-score is selected. The final parameters used are $w_1 = 11.7$, $w_2 = 0.09$, $w_3 = 12.22$, $w_4 = 0.01$, $g_0 = 1.47$, and $g_e = 1$.

4.2.2 Results

In order to determine the competence of SAICOSA in serving as a template selection method, we once again use (CASP) 11 experiment targets. After using SAICOSA to generate structural profile alignments for all structures in I-TASSER library, we rank the templates based

on the raw alignment score. The quality of a structural template is evaluated by the GDT-TS compared to the native structure.

The performance and comparison of MUSTER, MUSTER+ICOSA and SAICOSA on the CASP 11 targets are summarized in Table 4. It can be noticed that SAICOSA selects even better templates with an average GDT-TS score of 45.23, which is higher than both selected by MUSTER, and MUSTER+ICOSA.

Table 4
Overall performance of MUSTER, MUSTER+ICOSA, and SAICOSA on the CASP11 targets

Method	MUSTER	MUSTER+ICOSA	SAICOSA
Average GDT-TS of the top-ranked model	32.85	34.31	45.23

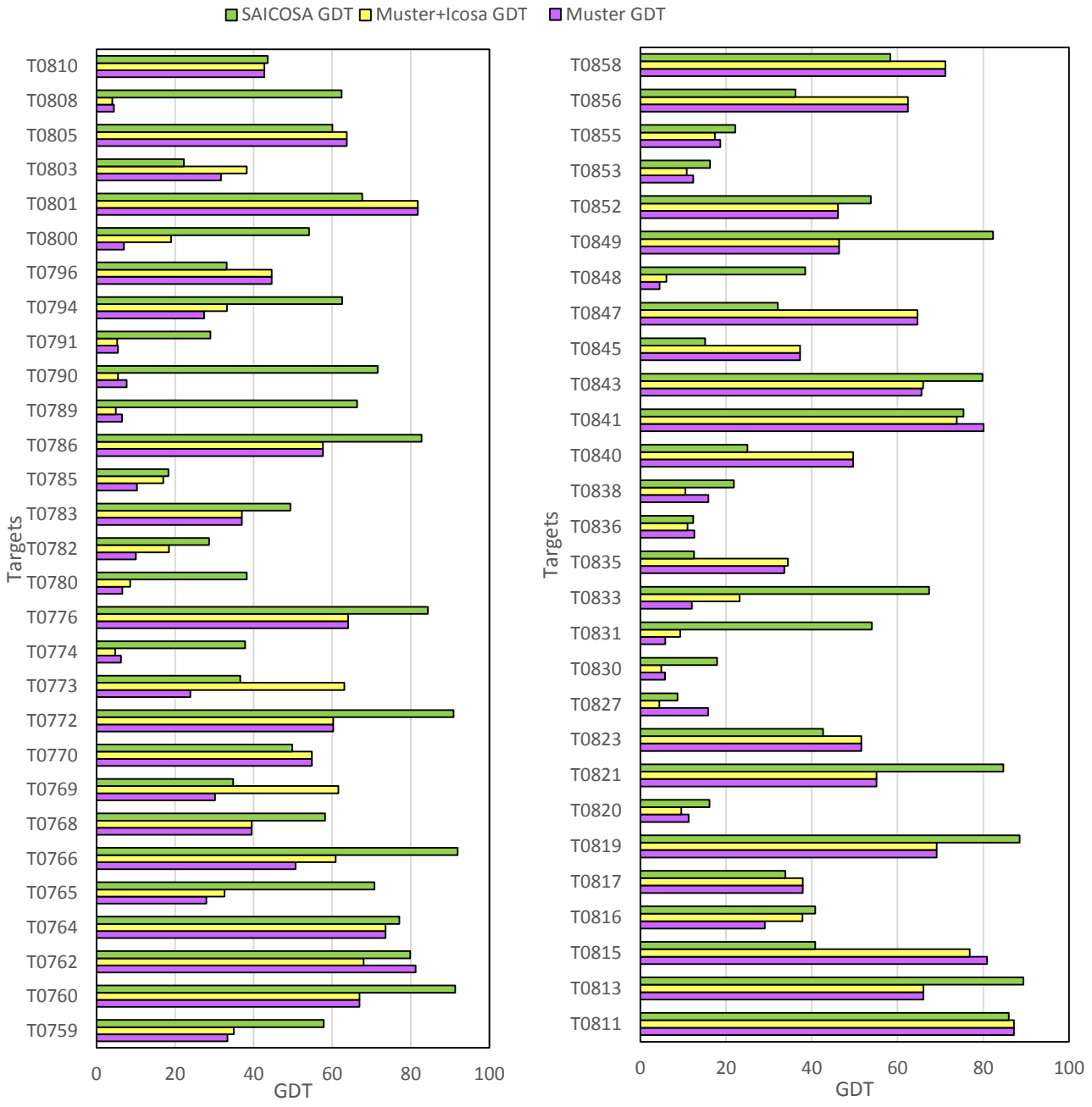


Fig. 27. The GDT-TS score of the top-ranked models selected by MUSTER, and MUSTER+ICOSA compared to the GDT-TS score of the top-ranked models generated using SAICOSA in CASP11 targets.

Fig. 27 compares the GDT-TS scores of the top-ranked templates selected by MUSTER, MUSTER + ICOSA, and SAICOSA. CASP 11 targets where the three techniques cannot find any

templates with over 20.0 GDT-TS score are ignored. One can find that for most of the targets the top-ranked model generated and selected using SAICOSA have a higher GDT-TS score than the ones generated and selected by MUSTER. This demonstrates that not only SAICOSA is better in template selection, but also that SAICOSA is capable of generating highly competitive structural profile alignments. One of the reasons for such an improvement is the highly accurate tools used in generating the protein structural features. For example, the Scorpion [154] secondary structure prediction method outperform the secondary structure prediction method used by MUSTER (PSI-PRED [165]). Also, CASA [166] has proven to predict the solvent accessibility more accurately than other states of art methods. Moreover, the fragment libraries proposed in [171] exhibit better representability across diverse protein structures. Finally, it is clear that using three-dimensional information (ICOSA) in sequence alignment and template selection can highly improve template-based protein structure modeling.

For further analysis targets T0790, T0766, and T0821 are picked. Fig. 28, Fig. 29, and Fig. 30 respectively display the models for the top-ranked templates generated by MUSTER, MUSTER+ICOSA, and SAICOSA in targets T0790, T0766 and T0821. T0790 has the most significant improvement achieved by SAICOSA, where the GDT-TS score of the top-ranked template improved from 7.6 by MUSTER and 5.47 by MUSTER+ICOSA to 71.6. For T0766, despite the improvement in the GDT-TS score of the top-ranked template when ICOSA score is combined with MUSTER score, GDT-TS score improved from 50.69 to 60.88, a significant improvement is achieved by SAICOSA (91.9 GDS-TS score). Finally, in T0821 it is interesting to notice that combining ICOSA score with MUSTER score doesn’t lead to any improvement (55.01 GDT-TS score), however, SAICOSA reaches a significant improvement (84.70 GDT-TS score).

In T0790, there is a great improvement achieved by SAICOSA in modeling both the β-sheet and the α-helix regions (Fig. 28). However, in T0766, it is noticed that the main enhancement in SAICOSA model appears in the α-helix regions (Fig. 29). Additionally, for α-helix protein as T0821, it becomes clearer that SAICOSA models are better aligning α-helix regions (Fig. 30).

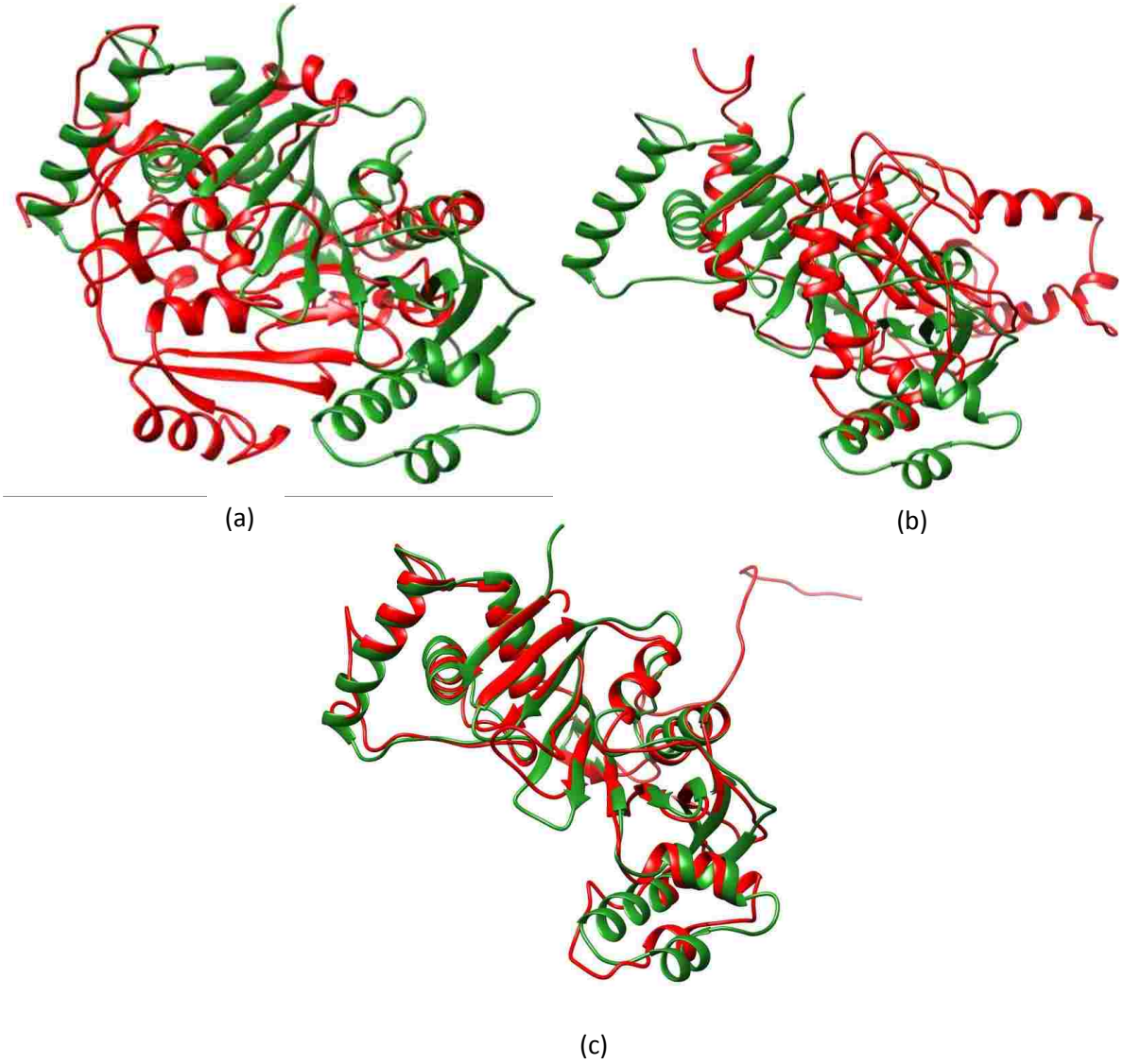


Fig. 28. Top-ranked templates selected by MUSTER, MUSTER+ICOSA, and SAICOSA (red) in CASP11 target T0790 (green), (a) top-ranked template by MUSTER score, (b) top-ranked template MUSTER+ICOSA scores, and (c) top-ranked template by SAICOSA.

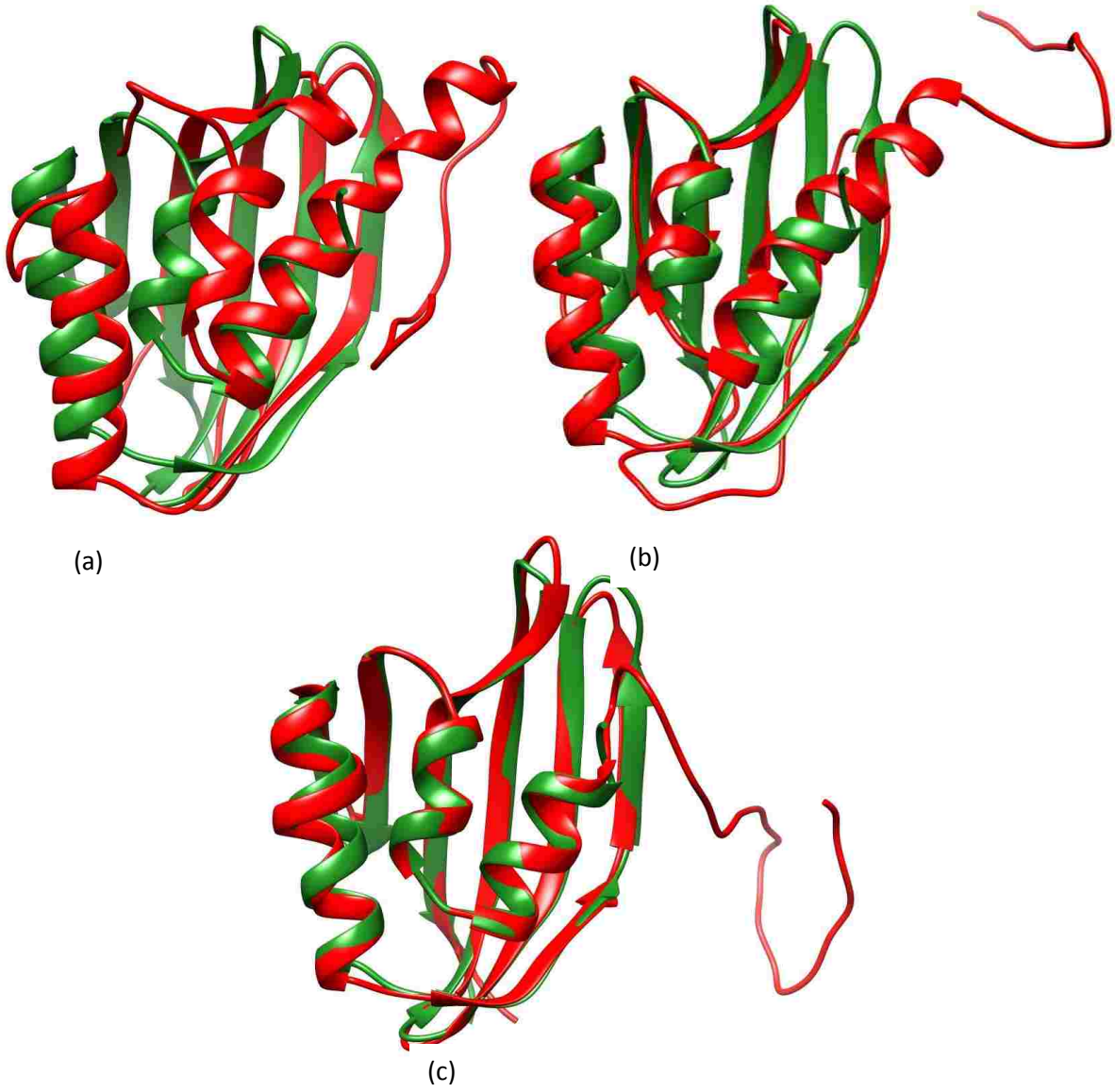


Fig. 29. Top-ranked templates selected by MUSTER, MUSTER+ICOSA, and SAICOSA (red) in CASP11 target T0766 (green), (a) top-ranked template by MUSTER score, (b) top-ranked template MUSTER+ICOSA scores, and (c) top-ranked template by SAICOSA.

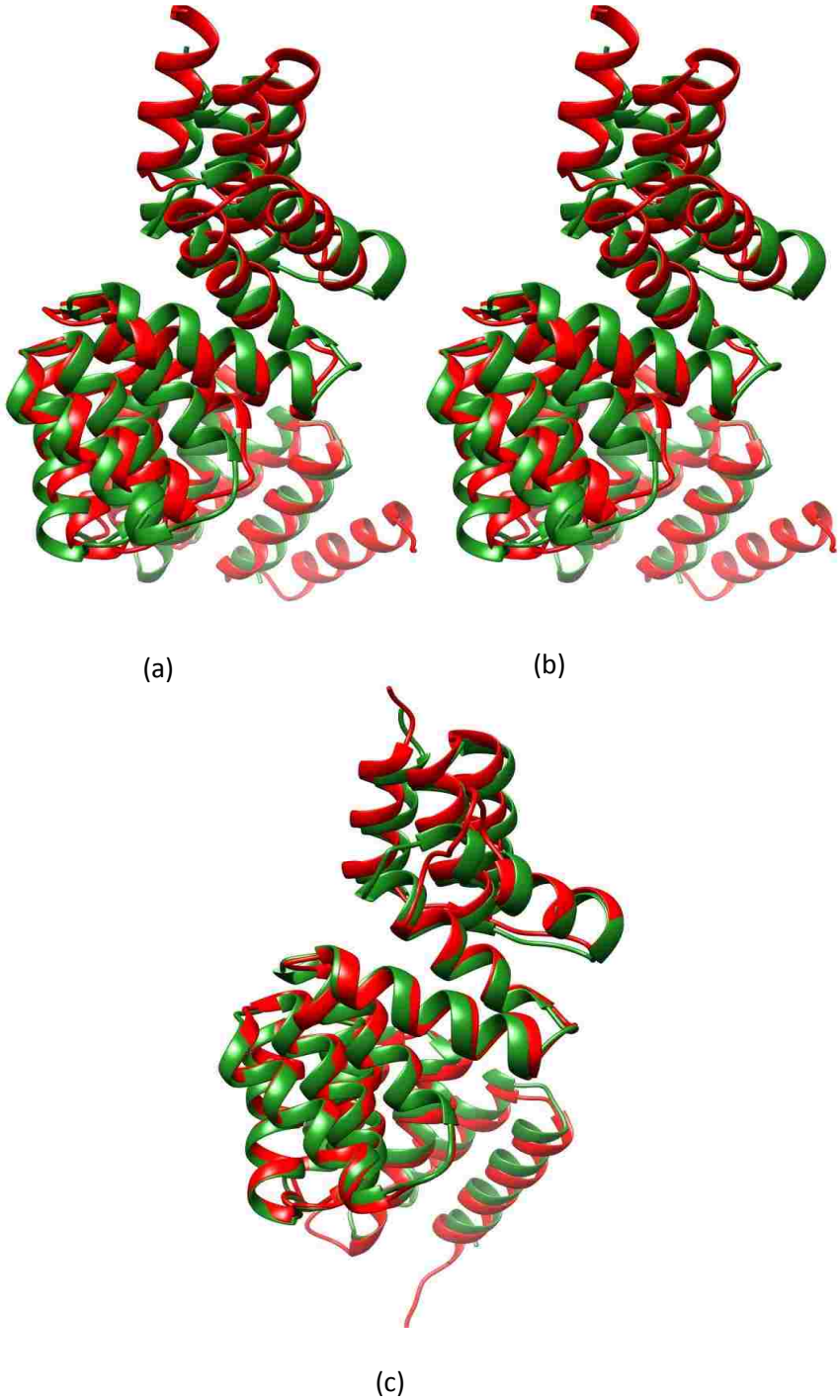


Fig. 30. Top-ranked templates selected by MUSTER, MUSTER+ICOSA, and SAICOSA (red) in CASP11 target T0821 (green), (a) top-ranked template by MUSTER score, (b) top-ranked template MUSTER+ICOSA scores, and (c) top-ranked template by SAICOSA.

4.3 Summary

Today, one of the most accurate and consistent methodologies for computational protein structure modeling is template-based modeling. The success of template-based modeling relies on correctly identifying one or a few experimentally determined protein structures as templates that are likely to resemble the structure of the target sequence. This work takes advantage of an inter residue contact scoring function to measure the favorability of a target sequence fitting in the folding topology of a certain template. This is performed by placing the target sequence residues into the mapped template residues three-dimensional conformation and evaluating the contact score. Then, we combine the contact score with the sequence profile score to enhance template selection sensitivity. This approach has shown a notable improvement in the accuracy and sensitivity of template selection in template-based protein structure modeling [16].

After the recognizable progress that is achieved in template selection using our first approach, we present a second template selection approach that employs three-dimensional information of protein in a more efficient way. In this approach, instead of evaluating the favorability of a target adopting a potential structural template after an alignment is generated, we use the three-dimensional information to build the alignment along with other structural features. The idea is to build a substitution matrix to score the replacement of one amino acid of the template three-dimensional conformation with each amino acid in the target. Then, we can use this substitution matrix to incorporate three-dimensional information in building the alignment along with the structural features. Consequently, the structural profile alignment between the target and templates are totally performed using our own alignment algorithm. The alignment is done by dynamic programming that exploits several protein structural features in addition to the three dimensional features. The template selection approach is tested over CASP 11 targets and has

shown a significant improvement compared to the successful template alignment and selection methods.

CHAPTER V

MULTI-OBJECTIVE PROTEIN SEQUENCE ALIGNMENT

In this chapter, we present two multi-objective alignment algorithms to obtain a set of diversified alignments yielding Pareto optimality. The first algorithm is a preliminary multi-objective alignment algorithm to examine the suitability of multi-objective alignment in protein structure modeling [179] (Section 5.1). Additionally, we develop a multi-objective alignment algorithm based on the Needleman-Wunsch algorithm (Section 5.2). The multi-objective Needleman-Wunsch algorithm guarantees not only Pareto optimality of the alignments, but also completeness. Both algorithms are examined on a set of CASP11 targets.

5.1 Multi-Objective Alignment (MOA) Algorithm

Our idea for MOA is based on the Needleman-Wunsch algorithm, but instead of building only one score matrix, we build a score matrix for each objective function. Tracing the maximum-match pathway in each matrix will end up generating the optimal alignment for the objective used to build this matrix. To get the multi-objective alignments we trace the maximum-match pathway in all the matrices to get each objective's optimal alignment. Whenever these alignment decisions (match, insert, and delete) of the objectives disagree, a new alignment, which has the same starting part as the alignment being traced but will continue by following the alignment decision of the disagreeing matrix, will be added. This procedure is done until all the alignments are discovered while tracing the objective matrices. Finally, the scores of the generated alignments are calculated according to all the objectives, and only the non-dominating alignments are kept.

The implementation of our method is split into two stages: score matrices generation and tracing objective matrices to generate the multi-objective alignments.

5.1.1 Score Matrices Generation

Given a set of objective functions $f_1(.), \dots, f_k(.)$, for two sequences $A = a_1 a_2 \dots a_M$ and $B = b_1 b_2 \dots b_N$, a score $s_{m,n}(f_i)$ is given to an aligned pair of residues a_m and b_n based on objective function $f_i(.)$. Besides, a gap penalty $g(f_i)$ is for aligning a residue from A/B to a gap. For each objective function $f_i(.)$ a score matrix $F(f_i)$ is computed according to Needleman-Wunsch algorithm and based on $f_i(.)$ scores, where $F_{m,n}(f_i)$ is calculated as follows:

$$F_{m,n}(f_i) = max \begin{cases} F_{m-1,n-1}(f_i) + s_{m,n}(f_i) & match/mismatch \\ F_{m-1,n}(f_i) + g(f_i) & insert \\ F_{m,n-1}(f_i) + g(f_i) & delete \end{cases} \quad (6)$$

The cells in $F(f_i)$ are generated one cell at a time starting from one at the top left corner. Once all the objective matrices are generated ($F(f_1), \cdots, F(f_k)$), the multi-objective alignments of sequences A and B with respect to $f_1(.), \dots, f_k(.)$. can be generated by tracing these matrices.

5.1.2 Backtracking the Objective Matrices

Once the score matrices ($F(f_1), \cdots, F(f_k)$) are completely generated, the multi-objective alignments will be generated by backtracking. The difference here is that the backtracking is done in more than one matrix. The backtracking of the multi-objective alignments is performed using the following iterating steps:

1. Initialize a set of alignments U where U initially holds only one alignment U_1. An alignment U_j is represented by two empty strings $A_A \leftarrow$ "" and $A_B \leftarrow$ "" to store the alignment, and two indices $m = M$ and $n = N$ to keep track of the current index in each sequence.
2. For each alignment $U_j \in U$, trace the score at the cell of indices m, n in every score matrix ($F(f_1), \cdots, F(f_k)$), to determine the source of $F_{m,n}(f_1) \cdots F_{m,n}(f_k)$.
    a. If all $F_{m,n}(f_1) \cdots F_{m,n}(f_k)$ came from a match, update U_j accordingly as $A_A \leftarrow a_m + A_A$, $B_B \leftarrow b_n + B_B$, $m = m - 1$, and $n = n - 1$.

b. If all $F_{m,n}(f_1) \cdots F_{m,n}(f_k)$ came from an insert, update U_j accordingly as $A_A \leftarrow a_m + A_A$, $B_B \leftarrow " - " + B_B$, $m = m - 1$, and $n = n$.
c. If all $F_{m,n}(f_1) \cdots F_{m,n}(f_k)$ came from a delete, update U_j accordingly as $A_A \leftarrow " - " + A_A$, $B_B \leftarrow b_n + B_B$, $m = m$, and $n = n - 1$.
d. If $\exists\ F_{mn}(f_i)$ that came from an insert while others $F_{m,n}(.)$ came from match, add a new alignment U_x based on the insert where $A_A \leftarrow a_m + A_A$, $B_B \leftarrow " - " + B_B$, $m = m - 1$, and $n = n$. Also, update U_j according to the match as $A_A \leftarrow a_m + A_A$, $B_B \leftarrow b_n + B_B$, $m = m - 1$, and $n = n - 1$.
e. If $\exists\ F_{mn}(f_i)$ that came from a delete while others $F_{m,n}(.)$ came from match, add a new alignment U_x based on the delete where $A_A \leftarrow " - " + A_A$, $B_B \leftarrow b_n + B_B$, $m = m$, and $n = n - 1$. Also, update U_j according to the match as $A_A \leftarrow a_m + A_A$, $B_B \leftarrow b_n + B_B$, $m = m - 1$, and $n = n - 1$.
f. If $\exists\ F_{mn}(f_i)$ that came from an insert while others $F_{m,n}(.)$ came from delete, add a new alignment U_x based on the insert where $A_A \leftarrow a_m + A_A$, $B_B \leftarrow " - " + B_B$, $m = m - 1$, and $n = n$. Also, update U_j according to the delete as $A_A \leftarrow " - " + A_A$, $B_B \leftarrow b_n + B_B$, $m = m$, and $n = n - 1$.
g. If $\exists\ F_{mn}(f_i)$ that came from an insert and $\exists\ F_{mn}(f_l)$ that came from a delete while others $F_{m,n}(.)$ came from match, add a new alignment U_x based on the insert where $A_A \leftarrow a_m + A_A$, $B_B \leftarrow " - " + B_B$, $m = m - 1$, and $n = n$. Also, add a new alignment U_y based on the delete where $A_A \leftarrow " - " + A_A$, $B_B \leftarrow b_n + B_B$, $m = m$, and $n = n - 1$. Besides, update U_j according to the match as $A_A \leftarrow a_m + A_A$, $B_B \leftarrow b_n + B_B$, $m = m - 1$, and $n = n - 1$.

3. Repeat step 2 till all the alignments in U reach indices 0,0

4. For each alignment $U_j \in U$ calculate its score according to all the objectives

5. Remove the dominated alignments from U.

- **Example**

To demonstrate how the algorithm works a simple alignment example is done over the following sequences.

Sequence A	P	Q	Q	Y	Y	P	Q	
Secondary Structure	C	H	H	B	B	C	C	
Sequence B	P	N	N	Y	Q	P	Y	Q
Secondary Structure	H	C	C	C	H	H	B	B

The objectives here are the profile and the secondary structure. The scoring function for both will be 1 for a match and -1 for mismatch or gap. Fig. 31 shows an illustration of the alignments

generation. Table 5 shows all the alignments generated for our example along with their scores and Table 6 shows the non-dominated ones.

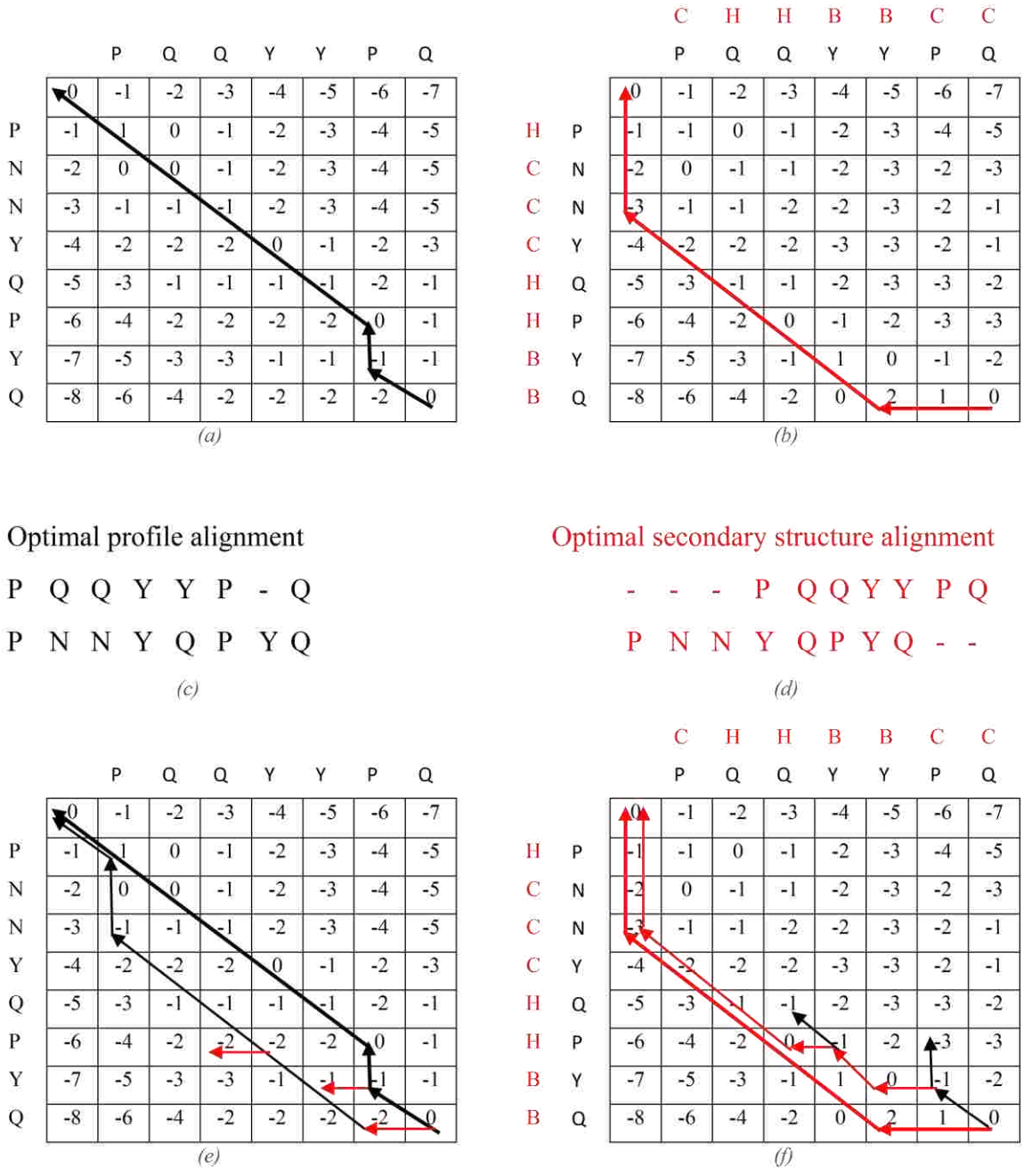


Fig. 31. (a) The Needleman-Wunsch alignment matrix based on the profile with the maximum-match path traced to generate the optimal alignment. (b) The Needleman-Wunsch alignment matrix based on the secondary structure with the maximum-match path traced to generate the optimal alignment. (c), & (d) The optimal profile alignment and the optimal secondary structure alignment respectively. (e), & (f) The Needleman-Wunsch alignment matrix based on the profile and the secondary structure respectively with the maximum-match path traced along with the splits due to disagreement of the other matrix, where the decisions taken based on the profile are marked on black and the ones based on the secondary structure are marked on red.

Table 5
All alignments generated using MOA

Generated alignment	Profile Score	Secondary Structure Score
P Q Q Y Y P - Q P N N Y Q P Y Q	0	-8
- - - P Q Q Y Y P Q P N N Y Q P Y Q - -	-6	0
- - P Q Q Y Y P Q P N N Y Q P Y - Q	-3	-3
- - P Q Q Y Y P Q - P N N Y Q P Y Q -	-5	-3
- - P Q Q Y Y P - Q P N N Y Q - P - Y Q	-6	-6
- - - P Q Q Y Y P Q P N N Y Q P - Y - Q	-4	-2
- - P Q Q Y Y P - Q P N N Y Q - - P Y Q	-4	-6
- - P Q Q Y Y P - Q P N N Y Q P - - Y Q	-6	-6
P Q Q Y - Y P - Q P N N Y Q P - Y Q	-3	-9
P Q Q Y - Y P - Q P N N Y Q - P Y Q	-1	-9
- - - P Q Q Y Y P - Q P N N Y Q P - - - Y Q	-7	-5
- P - Q Q Y Y P Q P N N Y Q P Y - Q	-3	-3
- P - Q Q Y Y P Q - P N N Y Q P Y Q -	-5	-3
P Q - Q Y Y P - Q - P N N Y Q P Y Q	-3	-7
- - P - Q Q Y Y P Q P N N Y Q P Y Q - -	-6	0
- - P - Q Q Y Y P Q P N N Y Q P - Y - Q	-4	-2
- - P - Q Q Y Y P Q P N N Y Q P - Y Q -	-6	-2
- P - Q Q Y Y P - Q P N N Y Q - P - Y Q	-6	-6
- P - Q Q Y Y P - Q P N N Y Q - - P Y Q	-4	-6
- P - Q Q Y Y P - Q P N N Y Q P - - Y Q	-6	-6
P Q - Q Y - Y P - Q - P N N Y Q P - Y Q	-6	-8
P Q - Q Y - Y P - Q - P N N Y Q - P Y Q	-4	-8
- - P - Q Q Y Y P - Q P N N Y Q P - - - Y Q	-7	-5
P - - Q Q Y Y P Q - P N N Y Q P Y Q -	-1	-5
P - - Q Q Y Y P Q - P N N Y Q P Y Q -	-3	-5
P Q - Q Y Y P - Q P - N N Y Q P Y Q	-1	-9
- P - - Q Q Y Y P Q - P N N Y Q P Y Q -	-6	0
- P - - Q Q Y Y P Q P N N Y Q P Y - Q	-4	-2
- P - - Q Q Y Y P Q - P N N Y Q P Y Q -	-6	-2
P - - Q Q Y Y P - Q P N N Y Q - P - Y Q	-4	-8
P - - Q Q Y Y P - Q P N N Y Q - - P Y Q	-2	-8
P - - Q Q Y Y P - Q P N N Y Q P - - Y Q	-4	-8
P Q - Q Y - Y P - Q P - N N Y Q - P Y Q	-4	-10
P Q - Q Y - Y P - Q P - N N Y Q - P Y Q	-2	-10
- P - - Q Q Y Y P - Q P N N Y Q P - - - Y Q	-7	-5
P - - - Q Q Y Y P Q - P N N Y Q P Y Q - -	-4	-2
P - - - Q Q Y Y P Q P N N Y Q P - Y - Q	-2	-4
P - - - Q Q Y Y P Q P N N Y Q P - Y - Q	-4	-4
P - - - Q Q Y Y P - Q P N N Y Q P - - - Y Q	-5	-7

Table 6
Non-dominated alignments

Non-dominated alignment	Profile Score	Secondary Structure Score
P Q Q Y Y P - Q P N N Y Q P Y Q	0	-8
- - - P Q Q Y Y P Q P N N Y Q P Y Q -	-6	0
- - P Q Q Y Y P Q P N N Y Q P Y - Q	-3	-3
- - - P Q Q Y Y P Q P N N Y Q P - Y Q	-4	-2
- P - Q Q Y Y P Q P N N Y Q P Y - Q	-3	-3
- - P - Q Q Y Y P Q P N N Y Q P Y Q -	-6	0
- - P - Q Q Y Y P Q P N N Y Q P - Y Q	-4	-2
P - - Q Q Y Y P Q P N N Y Q P Y Q -	-1	-5
- P - - Q Q Y Y P Q P N N Y Q P Y Q -	-6	0
- P N - Q Q Y Y P Q P N N Y Q P - Y Q	-4	-2
P - - - Q Q Y Y P Q P N N Y Q P Y Q -	-2	-4

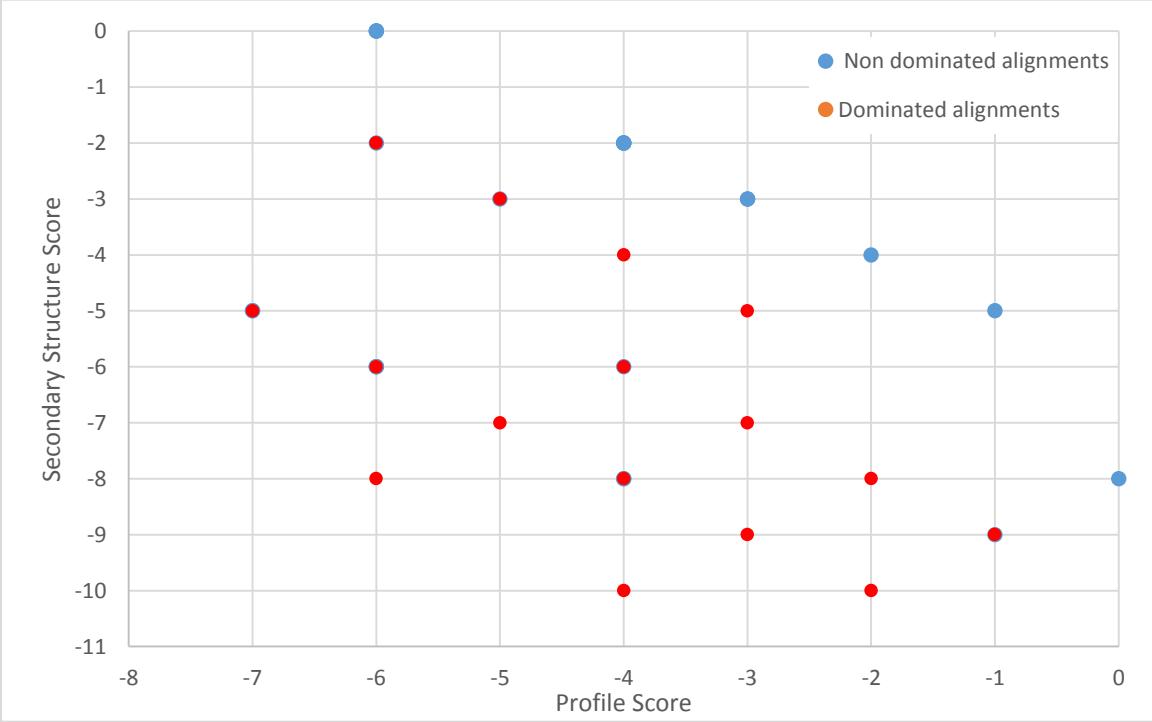


Fig. 32. Scores of the alignments generated by MOA where the red ones represent the dominated alignments and the blue ones represent the non-dominated alignments

5.1.3 Results

The Critical Assessment of Protein Structure Prediction (CASP) 11 experiment targets are used to demonstrate the effectiveness of MOA. Here, we use two scoring functions to measure the alignment between the *i*th residue in the query sequence and the *j*th residue in the template sequence, which result in score matrices of the query and template sequences. The first one is based on the sequence profile, which is $S_{profile}(i,j)$ from Eq. (2) illustrated in Section 4.2.1.1.

The second scoring function is based on structural features including predicted secondary structures and solvent accessibility.

$$SS(i,j) = w_a\, S_{structure}(i,j) + w_b\, S_{solvent}(i,j) \quad (7)$$

Here, $S_{structure}(i,j)$ is the probability that the predicted secondary structure of the *i*th residue of the query sequence matches with that of the *j*th residue in the template sequence (Eq. (3)). Similarly, $S_{solvent}(i,j)$ is the probability that the predicted solvent accessibility of the *i*th residue of the query sequence matches with that of the *j*th residue in the template sequence (Eq. (5)). Finally, w_a and w_b are weights that are carefully balanced using the grid search technique explained in Section 4.2.1.3, where $w_a = 1.17$ and w_b=1.21.

The idea of combining the secondary structure information with the solvent accessibility information of amino acids lies in the fact that environments around the protein residues can affect their tendencies for different structures [180]. Additionally, it has been previously suggested that more accurate secondary structure predictions can be achieved by taking solvent accessibility into account [181] [182]. Remarkably, secondary structure and solvent accessibility have been shown to have a strong influence on amino acid substitution [183]. Accordingly, the amino acid solvent accessibility is an effective factor for increasing the structure alignment accuracy between two protein sequences.

Fig. 33 shows an example of how the secondary structure score and solvent accessibility score are combined to generate one alignment matrix. In the example, the two protein sequences being aligned are 1A34 (chain A) and 1STM (chain A) (Fig. 33 (a)). The matrices in Fig. 33 (b), and Fig. 33 (c) represent the secondary structure and solvent accessibility probabilities respectively. By applying Eq. (7) a combined matrix is built (Fig. 33 (d)). Finally, using the Needleman-Wunsch algorithm an alignment matrix is generated according to the combined matrix (Fig. 33 (e)).

1A34A :

TGDNSNVVTMIRAGSYPKVNPTPTWVRAIPFEVSVQSGIAFKVPVGSLFSANFRTDSFTSVTVMSVRAWTQLTPPVN
EYSFVRLKPLFKTGDSTEEFEGRASNINTRASVGYRIPTNLRQNTVAADNVCEVRSNCRQVALVISCCFN

1STMA :

AAATSLVYDTCYVTLTERATTSFQRQSFPTLKGMGDRAFQVVAFTIQGVSAAPLMYNARLYNPGDTDSVHATGVQLM
GTVPRTVRLTPRVGQNNWFFGNTEEAETILAIDGLVSTKGANAPSNTVIVTGCFRLAPSELQSSTLVTGSEYETMLT
EIMSMGYERERVVAALRASYNNPHRAVEYLLTGIPG

(a)

	T	G	D	N	S	⋯
A	0.9975	0.995	0.9826	0.8712	0.7793	⋯
A	0.9975	0.995	0.9826	0.8712	0.7793	⋯
A	0.9975	0.995	0.9826	0.8712	0.7793	⋯
T	0.9975	0.995	0.9826	0.8712	0.7793	⋯
S	0.0009	0.0039	0.0089	0.0106	0.0325	⋯
⋮	⋮	⋮	⋮	⋮	⋮	⋱

	T	G	D	N	S	⋯
A	0.9858	0.932	0.9886	0.9776	0.8994	⋯
A	0.9858	0.932	0.9886	0.9776	0.8994	⋯
A	0.9858	0.932	0.9886	0.9776	0.8994	⋯
T	0.9858	0.932	0.9886	0.9776	0.8994	⋯
S	0.9858	0.932	0.9886	0.9776	0.8994	⋯
⋮	⋮	⋮	⋮	⋮	⋮	⋱

(b)

(c)

	T	G	D	N	S	⋯
A	2.3599	2.2919	2.3458	2.2022	2.0	⋯
A	2.3599	2.2919	2.3458	2.2022	2.0	⋯
A	2.3599	2.2919	2.3458	2.2022	2.0	⋯
T	2.3599	2.2919	2.3458	2.2022	2.0	⋯
S	1.1939	1.1323	1.2066	1.1953	1.1263	⋯
⋮	⋮	⋮	⋮	⋮	⋮	⋱

(d)

		T	G	D	N	S	⋯
	0	-2.5	-3	-3.5	-4	-4.5	⋯
A	-2.5	2.3599	0.3599	-0.1401	-0.6401	-1.1401	⋯
A	-3	0.3599	4.6518	2.7057	2.0621	1.3599	⋯
A	-3.5	-0.1401	2.6518	6.9976	4.9976	4.4976	⋯
T	-4	-0.6401	2.1518	4.9976	9.1998	7.1998	⋯
S	-4.5	-1.1401	1.6518	4.4976	7.1998	10.3261	⋯
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋱

(e)

Fig. 33. Generation of an alignment matrix for two sequences according to the combination between secondary structure score and solvent accessibility score, (a)the two sequences (b)Secondary structure substitution matrix, (c) Solvent accessibility substitution matrix, (d) the combined substitution matrix, (e) the Needleman-Wunsch alignment matrix based on the combined substitution matrix.

MOA is compared with two popularly used template alignment and selection methods for template-based protein structure modeling (Muster [15] obtained from the I-TASSER Suite [99] Version 5.1and GenTHREADER [83] obtained from the pGenTHREADER Suite [91] Version 8.9). Each target sequence is aligned with the same templates by the structure profile alignment method. Then, tertiary protein structure models are generated by the Modeller program [184] according to the alignments. The GDT-TS is used to measure the quality of these models and the corresponding alignments. Since MOA generates all Pareto-optimal alignments, which is usually more than one, we only show the one with the highest GDT-TS score.

We first compare MOA and Muster on the top-ranked template of each target specified by Muster. The performance of MUSTER and MOA on the CASP 11 targets are summarized in Table 7. It can be noticed that MOA outperformed MUSTER despite using less objectives than MUSTER. Additionally, Fig. 34 shows the GDT-TS score for Muster along with the MOA. As it appears in the figure that MOA achieved a higher or equal GDT-TS score for 102 targets. Also MOA GDT-TS score was greater than Muster by at least 10 points in seven targets. A similar comparison is done between MOA and pGenTHREADER, which is shown in

Table 7
Overall performance of MUSTER and MOA on the top-ranked template specified by Muster for the CASP11 targets.

Method	MUSTER	MOA
Average GDT-TS	33.28	36.46

Table 8 and Fig. 35. In 83 targets, the GDT-TS scores of models generated by MOA are higher than pGenTHREADER, wherein 17 of them, the gain is at least 10 points or higher.

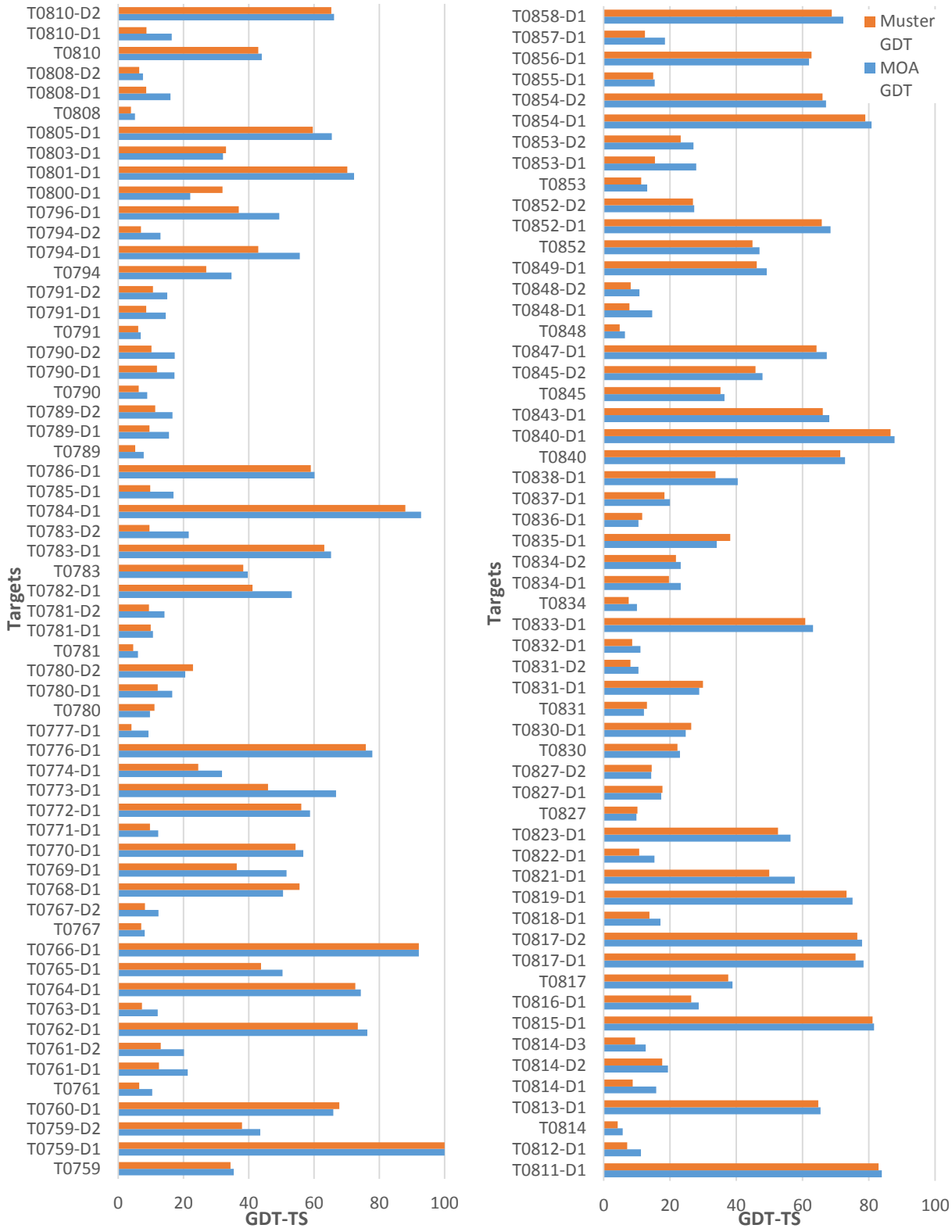


Fig. 34. The GDT-TS score of Muster alignment and MOA alignment to CASP 11 targets with the top-ranked template selected by Muster. MOA achieved a higher or equal GDT-TS score for 102 targets and most of the time MOA seven of them the difference is more than 10 i.e. T0773-D1

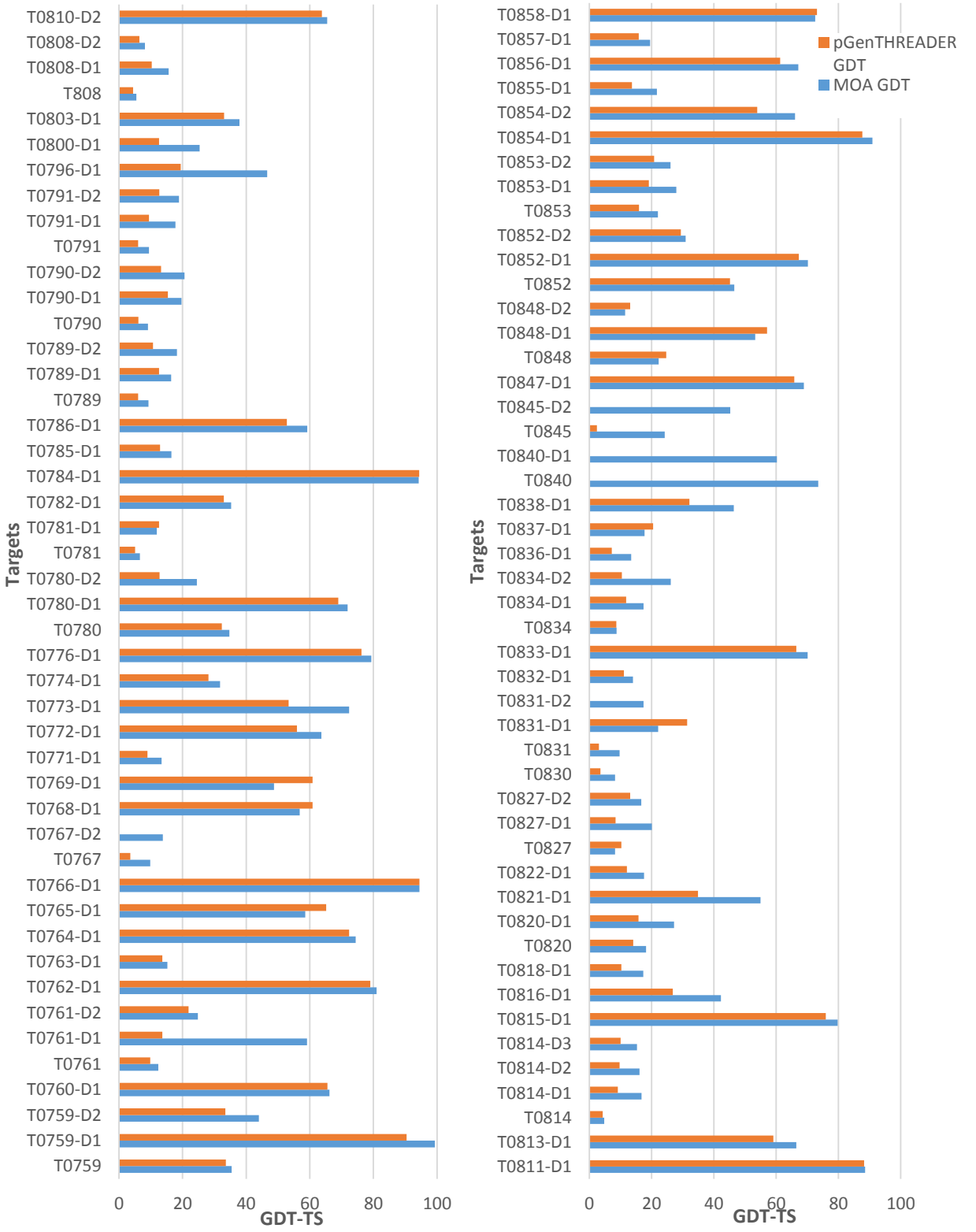


Fig. 35. The GDT-TS score of pGenTHREADER alignment and MOA alignment to CASP 11 targets with the top-ranked template selected by pGenTHREADER. In 83 targets MOA GDT-TS score is higher or equal pGenTHREADER, 17 of them MOA GDT-TS score was 10 points higher than pGenTHREADER. i.e. T0840

Table 8
Overall performance of GenTHREADER and MOA on the top-ranked template specified by GenTHREADER for the CASP11 targets

Method	GenTHREADER	MOA
Average GDT-TS	29.61	36.77

Another comparison is done with Muster and linear combination of objectives using the same sequence and structure information over CASP 11 on the top-ranked template of each target specified by CASP. The performance of MUSTER, linear combination, and MOA on the CASP 11 targets are summarized in Table 9. From the table it is clear that MOA outperformed MUSTER and linear combination. Also, it is noticed that the presence of a better template enhanced the performance of MOA, however, this is not the case for MUSTER. This indicates that MUSTER performance better when the template is present in its PDB library. Fig. 36 shows the GDT-TS scores of the models generated by MOA and Muster, wherein 93 targets MOA was able to generate at least one alignment with a higher GDT-TS score than Muster. Fig. 37 shows the GDT-TS scores of the models generated by MOA and linear combination of objectives. One can find that the GDT-TS scores of the top models generated by MOA are almost always better than those generated by the linear combination of objectives. Particularly, the MOA models exceed those generated by the linear combination of objectives by at least 10 points in 43 targets.

Table 9
Overall performance of MUSTER , linear combination objectives and MOA on the top-ranked template specified by CASP for the CASP11 targets.

Method	MUSTER	Linear Combination	MOA
Average GDT-TS	31.8	28.35	39.29

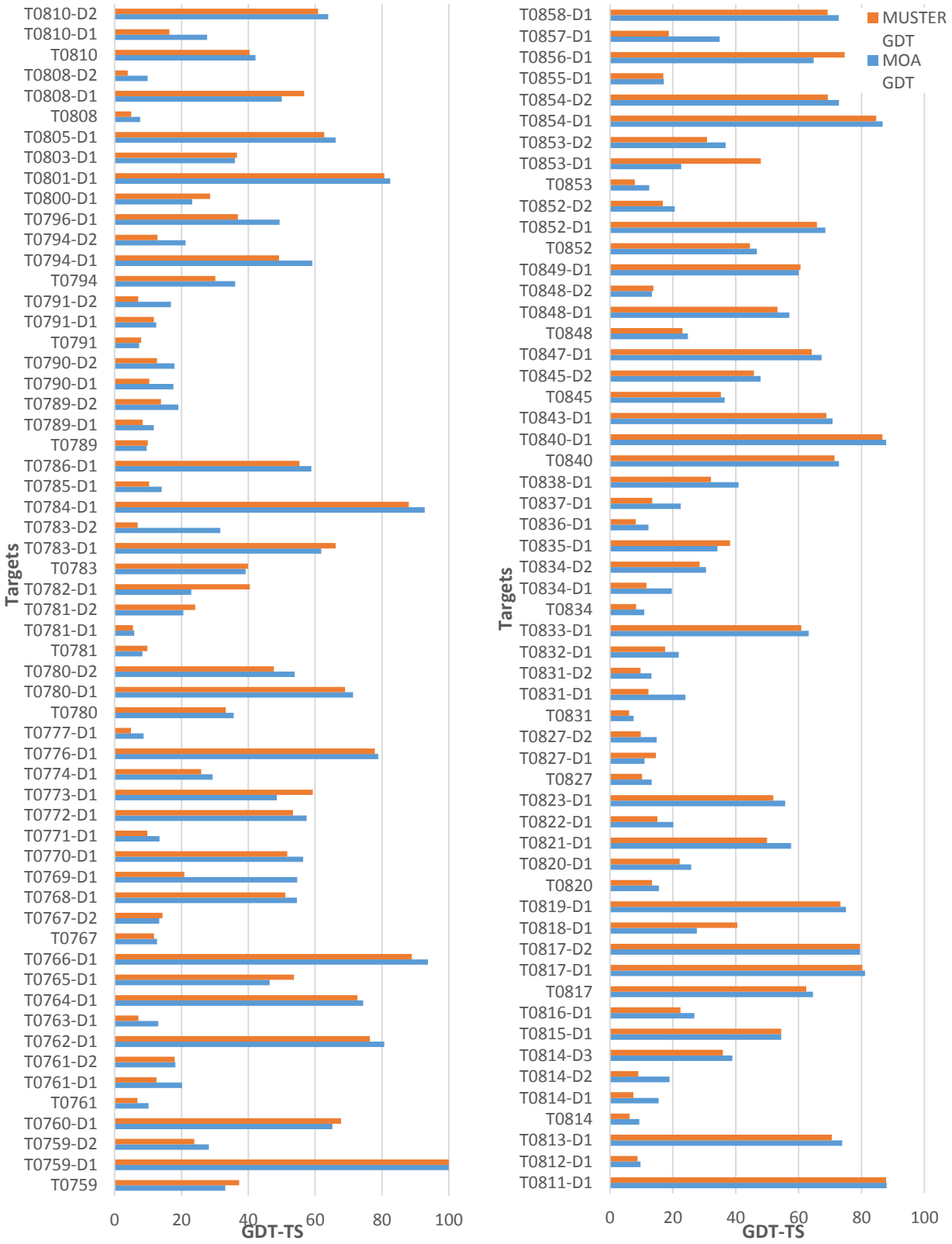


Fig. 36. The GDT-TS score of Muster alignment and MOA alignment to CASP 11 targets with the top-ranked template selected by CASP. In 93 targets MOA GDT-TS score is higher or equal Muster, 4 of them MOA GDT-TS score was 10 points higher than Muster. i.e. T0769-D1. Muster achieved highly in 3 targets i.e T0782-D1

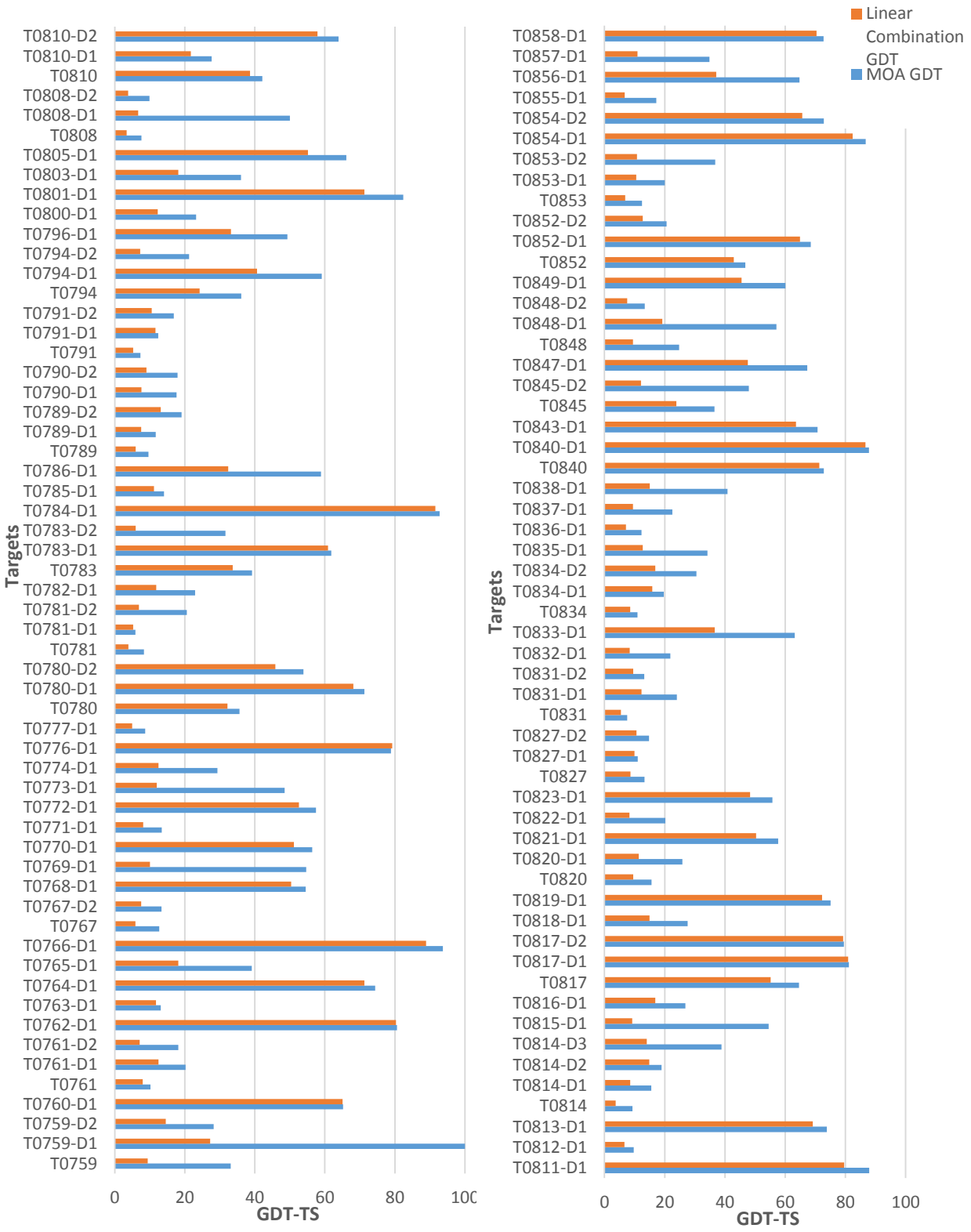


Fig. 37. The GDT-TS score of linear combination of objectives algorithm using same sequence and structure information and MOA for CASP 11 targets with the top-ranked template selected by CASP. In 113 targets MOA achieved higher or equal GDT-TS, most of them MOA GDT-TS score was 10 points higher. i.e. T0759-D1. Only at T0776-D1 MOA was lower and by a very small difference.

For further analysis, targets T0766-D1 and T0769-D1 are picked. According to CASP 11 the best template that matches T0766-D1 is 4orlA. When MOA operates on T0766-D1 and 4orlA, it generates 6 alignments and the best model has 93.75 GDT-TS score, while linear combination model scores only 88.889. Fig. 38 shows the profile and secondary structure/solvent accessibility for T0766-D1 and 4orlA alignments. As it is clear from the figure that linear combination alignment is dominated by all the alignments generated by MOA. Fig. 39 shows the best model generated by MOA alignment and the model generated from Muster alignment along with their alignments.

For T0769-D1 CASP 11 indicates that 3ramD is its best template. When MOA operates on T0769-D1 and 3ramD, it is able to produce 217 alignments where the best model has 54.639 GDT-TS score, while linear combination model scored only 10.052. Fig. 40 shows the profile and secondary structure/solvent accessibility scores for T0769-D1 and 3ramD alignments. As it is clear from the figure that linear combination alignment is dominated by all the alignments generated by MOA. Fig. 41 shows the best model generated by MOA alignment and the model generated from linear combination alignment along with their alignments.

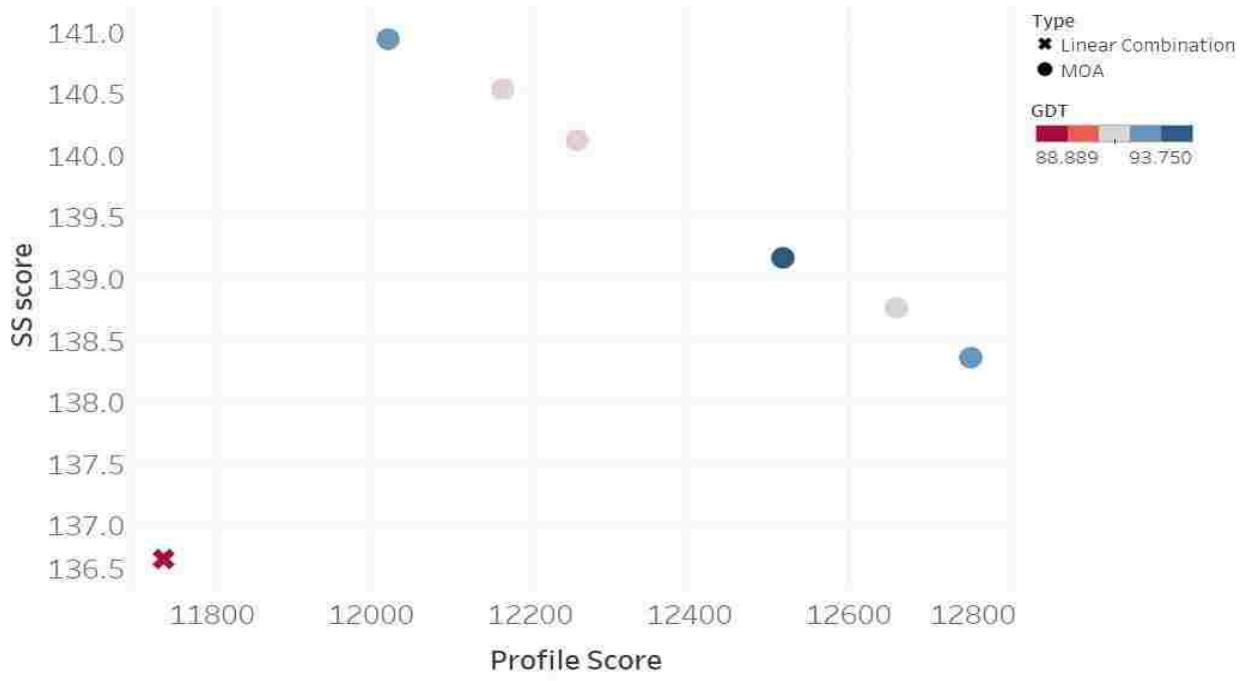
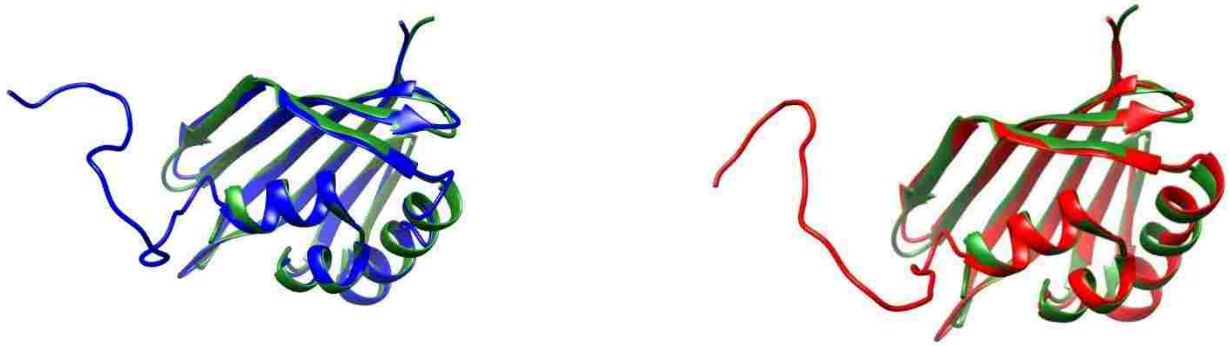


Fig. 38. Results for T0766-D1 alignment with 4orlA



(a) MOA (b) Linear Combination

By MOA

```
T0766-D1: MKKRVIFLLTGLFIWTSVLLAQNVPEGVIGAFKEGNSQELNKYLGDKVDLIIQNKSTHADKRTAEGTMA
4orlA:    --------------------GQEIPAGVITAFKRGSSQELSKYG-DKVNLVFQGRSTNVDKQKATAA-Q
T0766-D1: AFFSNHKVGSFNVNHQGKRDESGFVIGILMTANGNFRVNCFFRKVQNKYVIHQIRIDKTDE*
4orlA:    EFFTKNKVSGFNVNHQGKRDESSFVIGTLATTNGNFRVNCFLKKVQNQYLIHQIRIDKINE*
```

By Linear Combination

```
T0766-D1: MKKRVIFLLTGLFIWTSVLLAQNVPEGVIGAFKEGNSQELNKYLGDKVDLIIQNKSTHADKRTAEGTMA
4orlA:    --------------------GQEIPAGVITAFKRGSSQELSKYG-DKVNLVFQGRSTNVD-KQKATAAQ
T0766-D1: AFFSNHKVGSFNVNHQGKRDESGFVIGILMTANGNFRVNCFFRKVQNKYVIHQIRIDKTDE*
4orlA:    EFFTKNKVSGFNVNHQGKRDESSFVIGTLATTNGNFRVNCFLKKVQNQYLIHQIRIDKINE*
```

(c) Alignment between 4orlA and T0766-D1

Fig. 39. The best scoring alignments generated by MOA and that generated by linear combination for T0766-D1 and 4orlA. The model generated from MOA alignment scores 93.75 GDT-TS while linear combination scores only 88.889

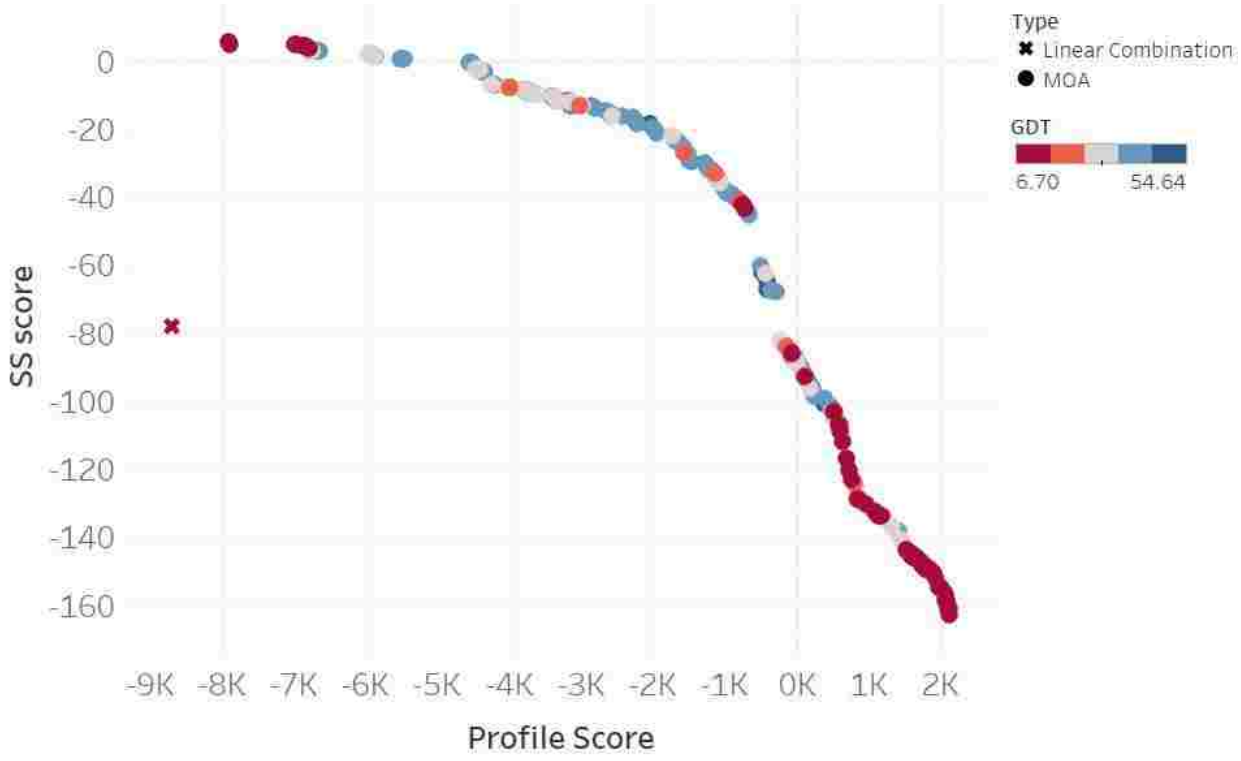
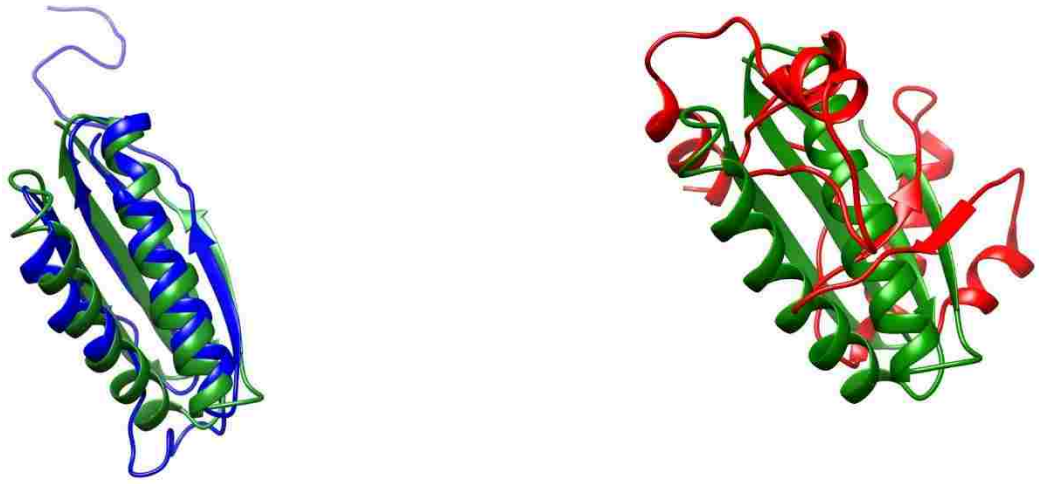


Fig. 40. Results for T0769-D1 alignment with 3ramD



(a) MOA (b) Linear Combination

By MOA

```
T0769-D1:M-------------------------------------------------------------------------------
3ramD:   GEKQQILDYIETNKYSYIEISHRIHERPELGNEEIFASRTLIDRLKEHDFEIETEIAGHATGFIATYDSGLDGPAIGFLA
T0769-D1:-------------------------------------------------------------------------------
3ramD:   EYDALPGLGHACGHNIIGTASVLGAIGLKQVIDQIGGKVVVLGCPAEEGGENGSAKASYVKAGVIDQIDIALIHPGNETY
T0769-D1:------LTVEVEVKITAD--DENKAEEIVKRV--------IDEVEREVQKQY-PNATITRTLTR----DD---GTVELRI
3ramD:   KTIDTLAVDVLDVKFYGKSAHASEN-A--DEALNALDAISYFNGVAQLRQHIKKDQRVHGVILDGGKAANIIPDYTHARF
T0769-D1:KVKADTEEKAKSIIKLIEERIEEELRKRDPNATITR-------------------------TVR--------TEVG--
3ramD:   YTRATRKE-LDILTEKVNQIARGAAIQTGCDYEFGPIQNGVNEFIKTPKLDDLFAKYAEEVGEAVIDDDFGYGSTDTGNV
T0769-D1:---------------SSWSLEHH---------------------------HHH------------H*
3ramD:   SHVVPTIHPHIKIGSRNLVGHTHRFREAAASVHGDEALIKGAKIALGLELITNQDVYQDIIEEHAHLKG*
```

By Linear Combination

```
T0769-D1:-------------------------------------------------------------------------------
3ramD:   GEKQQILDYIETNKYSYIEISHRIHERPELGNEEIFASRTLIDRLKEHDFEIETEIAGHATGFIATYDSGLDGPAIGFLA
T0769-D1:-------------------------------------------------------------------------------
3ramD:   EYDALPGLGHACGHNIIGTASVLGAIGLKQVIDQIGGKVVVLGCPAEEGGENGSAKASYVKAGVIDQIDIALIHPGNETY
T0769-D1:-----------------------------------MLTVEVEVKIT------ADDENKA------E--EIVKRVIDEVE
3ramD:   KTIDTLAVDVLDVKFYGKSAHASENADEALNALDAISYFNGVAQLRQHIKKDQRVHGVILDGGKAANIIPD-YTHARFYT
T0769-D1:---REVQKQYPNATITRTLTRDDGT--VELRI-----------KVKADTEEKAKS----------------IIK-L--
3ramD:   RATRKELDILTEKVNQIARGAAIQTGCDYEFGPIQNGVNEFIKTPKLDDLFAKYAEEVGEAVIDDDFGYGSTDTGNVSHV
T0769-D1:---I--EE-----------RIEEE----LRKRDPNATITR--T-VRTE--VGSSWSLEHHHHHH--*
3ramD:   VPTIHPHIKIGSRNLVGHTHRFREAAASVHGDEALIKGAKIALGLELITNQDVYQDIIEEHAHLKG*
```

(c) Alignment between 3ramD and T0769-D1

Fig. 41. The best scoring alignment generated by MOA and that generated by linear combination algorithm for T0769-D1 and 3ramD. The model generated from MOA alignment scores 54.639 GDT-TS while linear combination scores only 10.052.

5.2 A Multi-Objective Needleman-Wunsch Algorithm (MON)

Despite the competitive results shown from MOA, it suffers from two main deficiencies. First, the MOA algorithm generates a new alignment whenever the objectives disagree with each

other, which may lead an exponential growth of the number of the traces and then end up with a large number of alignments. This is very computationally costly, particularly when aligning long protein sequences. In fact, we are only interested in the non-dominated alignments. Second, MOA does not guarantee the generation of the entire Pareto-optimal front. Hence, we develop the Multi-Objective Needleman-Wunsch (MON) algorithm. Given a set of potentially conflicting scoring functions, MON pursues a multi-objective optimization strategy and report a novel multi-objective sequence alignment algorithm based on the Needleman-Wunsch algorithm to obtain a set of diversified alignments yielding Pareto optimality. MON guarantees not only Pareto optimality of the alignments, but also completeness.

By definition, the alignments which are not dominated by any other alignments form the Pareto-optimal front. Our algorithm is designed to extend the Needleman-Wunsch algorithm to generate the complete set of Pareto-optimal alignments given a set of objective functions.

5.2.1 Generation of Multi-Objective Score Matrix

Given a set of objective functions $f_1(.), \dots, f_s(.)$, for two sequences $A = a_1 a_2 \dots a_M$ and $B = b_1 b_2 \dots b_N$, a vector $\vec{S}_{m,n}$ is given to an aligned pair of residues a_m and b_n as follows:

$$\vec{S}_{m,n} = \begin{bmatrix} s_{m,n,1} \\ s_{m,n,2} \\ \vdots \\ s_{m,n,s} \end{bmatrix} \tag{8}$$

where s is the number of objective functions and each score $s_{m,n,i}$ is the score to align a_m and b_n based on objective i. Besides the score vector, there is a gap score vector $\vec{G}$ for aligning a residue from A/B to a gap such that

$$\vec{G} = \begin{bmatrix} g_1 \\ g_2 \\ \vdots \\ g_s \end{bmatrix} \tag{9}$$

Similar to the Needleman-Wunsch algorithm, we firstly construct a $(M+1)\times(N+1)$ matrix F whose axes are the two sequences, A and B, to be aligned. Instead of keeping the best score for the sub-alignment in the Needleman-Wunsch algorithm, here each cell $F_{m,n}$ is designed to hold a complete set of Pareto-optimal score vectors with respect to the sub-alignment generated so far between the two sub-sequences $\dot{A}=a_1a_2\cdots a_m$ and $\dot{B}=b_1b_2\ldots b_n$ ending up at this cell. Assuming that there are k sub-alignments, $u_1,\ldots,u_k$, ending up in $F_{m,n}$, then

$$F_{m,n}=\left\{\vec{D}_{m,n}(u_1)_,,\vec{D}_{m,n}(u_2)_,,\cdots,\vec{D}_{m,n}(u_k)\right\} \tag{10}$$

where $\vec{D}_{m,n}(u_i)$ is the score vector for the non-dominating sub-alignment u_i ending in cell $F_{m,n}$

At the beginning, the cells in the first row and first column are calculated. As these cells represent aligning one of the two sequences to nothing, so there could only be one alignment passing by any of these cells, which is aligning one sequence to a gap such that

$$F_{0,n}=\left\{n\vec{G}\right\},\ and$$
$$F_{m,0}=\left\{m\vec{G}\right\}. \tag{11}$$

Then, starting from three neighboring cells $F_{m-1,n}$, $F_{m-1,n-1}$, and $F_{m,n-1}$, three sets of score vectors $P_{m,n}$, $U_{\mathrm{m,n}}$, and $Q_{\mathrm{m,n}}$ are generated by match, insert, and delete, respectively, such that

$$P_{m,n}=\left\{\vec{D}_{m-1,n-1}(u_1)+\vec{S}_{m,n},\cdots,\vec{D}_{m-1,n-1}(u_x)+\vec{S}_{m,n}\right\},$$
$$U_{m,n}=\left\{\vec{D}_{m-1,n}(u_1)+\vec{G},\cdots,\vec{D}_{m-1,n}(u_y)+\vec{G}\right\},$$
$$Q_{m,n}=\left\{\vec{D}_{m,n-1}(u_1)+\vec{G},\cdots,\vec{D}_{m,n-1}(u_z)+\vec{G}\right\}. \tag{12}$$

where x, y, and z are the number of dominating score vectors corresponding to sub-alignments ending at cells $F_{m-1,n-1}$, $F_{m-1,n}$, and $F_{m,n-1}$, respectively.

Denoting $F^*_{m,n}$ as the union of score vectors generated from three neighboring cells, i.e.,

$$F^*_{m,n}=P_{m,n}\cup U_{m,n}\cup Q_{m,n}, \tag{13}$$

a domination function $Dom(.)$ is carried out on $F^*_{m,n}$ to eliminate the dominated score vectors and generate the score vector set for $F_{m,n}$ such that

$$F_{m,n} = Dom\left(F^*_{m,n}\right). \quad (14)$$

Here, $Dom(.)$ is a domination function returning the Pareto optimal score vectors from a score vector set. The pseudo code implementation of the domination function $Dom(.)$ is described in Algorithm 1.

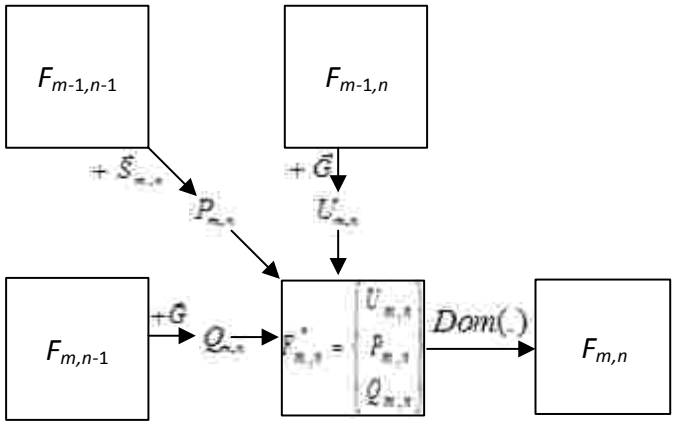


Fig. 42. Generation of $F_{m,n}$ from three neighboring cells $F_{m-1,n}$, $F_{m-1,n-1}$, and $F_{m,n-1}$

Algorithm 1 (Dom(.) function): Finding the Pareto optimal (non-dominating) score vectors from a score vector set

Input:

Set of score vectors D^*

Output:

Set of Pareto-optimal scores vectors D

Procedure:

$D \leftarrow \{\}$ //initialize an empty set for D
for each $\vec{D}_i \in D^*$ **do**
dominated ← False
for each $\vec{D}_j \in D^*$ **do**
if $i \neq j$ and $\vec{D}_j \prec \vec{D}_i$ **then**
dominated ← True
break
end if
end for
if dominated = False **then**
$D \leftarrow D \cup \vec{D}_i$
end if
end for
return D

The cells in F are generated one row at a time and one cell at a time starting from one at the up left corner, exactly the same as the Needleman-Wunsch algorithm. Once all cells in F are filled, the set of score vectors in cell at the down right corner correspond to the complete Pareto-optimal front of alignments of sequences A and B with respect to $f_1(.), \dots, f_s(.)$. By putting all pieces together, Algorithm 2 depicts the procedure of generating the multi-objective score matrix F. Furthermore, Theorem 1 shows the Pareto-optimality and solution completeness of the multi-objective Needleman-Wunsch algorithm.

Algorithm 2: Generation of multi-objective score matrix *F*

Input:

Two sequences $A = a_1 a_2 \dots a_M$ and $B = b_1 b_2 \dots b_N$, score vectors $\vec{S}_{m,n}$ for matching a_m and b_n, and gap penalty vector $\vec{G}$

Output: *F* matrix

Procedure:

// initialize first column
for $m \leftarrow 0$ to $length(A)$ **do**
    $F(m, 0) \leftarrow \{m * \vec{G}\}$
end for
// initialize first row
for $n \leftarrow 0$ to $length(B)$**do**
    $F(0, n) \leftarrow \{n * \vec{G}\}$
end for
// fill out the rest of the elements
for $m = 1\ to\ length(A)$ **do**
    for $n = 1\ to\ length(B)$ **do**
        $P_{m,n} \leftarrow \{\}$
        for $k = 1\ to\ |\mathrm{F}_{m-1,n-1}|$ **do**
            $P_{m,n} \leftarrow P_{m,n} \cup \{\vec{D}_{m-1,n-1,k} + \vec{S}_{m,n}\}$
        end for
        $U_{m,n} \leftarrow \{\}$
        for $k = 1\ to\ |\mathrm{F}_{m-1,n}|$ **do**
            $U_{m,n} \leftarrow U_{m,n} \cup \{\vec{D}_{m-1,n,k} + \vec{G}\}$
        end for
        $Q_{m,n} \leftarrow \{\}$
        for $k = 1\ to\ |\mathrm{F}_{m,n-1}|$ **do**
            $Q_{m,n} \leftarrow Q_{m,n} \cup \{\vec{D}_{m,n-1,k} + \vec{G}\}$
        end for
        $F^* \leftarrow P_{m,n} \cup U_{m,n} \cup Q_{m,n}$
        $F_{m,n} = Dom(F^*_{m,n})$
    end for
end for

Theorem 1: (Pareto-optimality and Completeness of Multi-Objective Needleman-Wunsch Algorithm) The score vectors kept in each cell $F_{m,n}$ are Pareto optimal and complete for all alignments end up at $F_{m,n}$.

Proof: Theorem 1 is proved by induction.

Base case: Assume aligning two one character sub-sequences $A = a_1$ and $B = b_1$. Initially, there is only one alignment ends up in $F_{0,1}$ and $F_{1,0}$. Clearly, $F_{0,1} = F_{1,0} = \{\vec{G}\}$ are Pareto optimal and are complete. Then, for $F_{1,1}$, there exist only three alignments, the *Dom*(.) function carried out on $F^*_{1,1}$ guarantees that the score vectors in $F_{1,1}$ are also Pareto optimal and complete.

Induction step: Suppose that $F_{m-1,n-1}$, $F_{m-1,n}$, and $F_{m,n-1}$ all contain complete Pareto optimal score vectors with respect to all sub-alignments ending up in these cells. We need to show that the score vectors in $F_{m,n}$ are also Pareto optimal and complete. Here we consider the following three sub-alignments terminating in $F_{m,n}$.

1) Given two sub-alignments u and v ending in $F_{m-1,n-1}$ with score vectors $\vec{D}_{m-1,n-1}(u)$ and $\vec{D}_{m-1,n-1}(v)$, if $\vec{D}_{m-1,n-1}(u) \prec \vec{D}_{m-1,n-1}(v)$, then, $\vec{D}_{m-1,n-1}(u) + \vec{S}_{m,n} \prec \vec{D}_{m-1,n-1}(v) + \vec{S}_{m,n}$. That is, for the correspondent incrementally built sub-alignment u' and v' from u and v by adding a match (a_m, b_n), respectively, u' dominates v'.

2) Given two sub-alignments u and v ending in $F_{m,n-1}$ with score vectors $\vec{D}_{m,n-1}(u)$ and $\vec{D}_{m,n-1}(v)$, if $\vec{D}_{m,n-1}(u) \prec \vec{D}_{m,n-1}(v)$, then, $\vec{D}_{m,n-1}(u) + \vec{\mathrm{G}} \prec \vec{D}_{m,n-1}(v) + \vec{\mathrm{G}}$. That is, for the correspondent incrementally built sub-alignment u' and v' from u and v by adding a gap, respectively, u' dominates v'.

3) Given two sub-alignments u and v ending in $F_{m-1,n}$ with score vectors $\vec{D}_{m-1,n}(u)$ and $\vec{D}_{m-1,n}(v)$, if $\vec{D}_{m-1,n}(u) \prec \vec{D}_{m-1,n}(v)$, then, $\vec{D}_{m-1,n}(u) + \vec{\mathrm{G}} \prec \vec{D}_{m-1,n}(v) + \vec{\mathrm{G}}$. That is, for the correspondent incrementally built sub-alignment u' and v' from u and v by deleting a gap, respectively, u' dominates v'.

All sub-alignments ending up in $F_{m,n}$ have to pass through either $F_{m-1,n-1}$, $F_{m-1,n}$, or $F_{m,n-1}$. This indicates that any sub-alignments terminating in $F_{m,n}$ built on top of a non-dominating sub-alignments from $F_{m-1,n-1}$, $F_{m-1,n}$, or $F_{m,n-1}$ will remain non-dominating. As a result, $F^*_{m,n} = P_{m,n} \cup U_{m,n} \cup Q_{m,n}$ contains all potentially dominant sub-alignments. Again, the Dom(.) function eliminates the non-dominating score vectors in $F^*_{m,n}$, which results in a complete and Pareto-optimal set in $F_{m,n}$.

5.2.2 Backtracking the Pareto-optimal Alignments

Similar to the Needleman-Wunsch algorithm, once the score matrix F is completely generated, the Pareto-optimal alignments will be generated by backtracking. The only difference is that here backtracking is done by matching the score vectors instead of a single score. For each score vector $\vec{D}_{M,N,c} \in F_{M,N}$, the generation of $\vec{D}_{M,N,c}$ will be traced through the matrix from $F_{M,N}$ back to $F_{0,0}$ to generate the alignment. The backtracking of $\vec{D}_{M,N,c}$ is performed using the following iterating steps:

1. Initialize two empty strings $A_A \leftarrow$ "" and $A_B \leftarrow$ "" to hold store the alignment.
2. Initialize $m = M$ and $n = N$ to keep track of the current index in each sequence.
3. For each $\vec{D}_{m,n,i} \in F_{m,n}$, check the three possible sources of $\vec{D}_{m,n,i}$ in $F_{m-1,n-1}$, $F_{m-1,n}$, and $F_{m,n-1}$ to determine the source of $\vec{D}_{m,n,i}$.

a. If $\exists\vec{D}_{m-1,n-1,t}$ such that $\vec{D}_{m,n,i} = \vec{D}_{m-1,n-1,t} + \vec{S}_{m,n}$, then $\vec{D}_{m,n,i}$ came from a match. Update A_A, B_B, m, n, and w accordingly as $A_A \leftarrow a_m + A_A$, $B_B \leftarrow b_n + B_B$, $m = m - 1$, $n = n - 1$, and $i = t$.

b. If $\exists\vec{D}_{m-1,n,r}$ such that $\vec{D}_{m,n,i} = \vec{D}_{m-1,n,r} + \vec{G}$, then $\vec{D}_{m,n,i}$ came from an insert. Update A_A, B_B, m, n, and r accordingly as $A_A \leftarrow a_m + A_A$, $B_B \leftarrow " - " + B_B$, $m = m - 1$, $n = n$, and $i = r$.

c. If $\exists\vec{D}_{m,n-1,v}$where $\vec{D}_{m,n,i} = \vec{D}_{m,n-1,v} + \vec{G}$, then $\vec{D}_{m,n,i}$ came from a delete. Update A_A, B_B, m, n, and w accordingly as $A_A \leftarrow " - " + A_A$, $B_B \leftarrow b_n + B_B$, $m = m$, $n = n - 1$, and $i = v$

4. Repeat step 3 till $F_{0,0}$ is reached.

- **Example**

Here we use two short DNA sequences X = GGCCTACCAT and Y = AAAGAGATT to demonstrate the alignment procedure of the multi-objective Needleman-Wunsch algorithm. The alignment is performed under two objectives using the following two alignment scoring matrices:

1) The default alignment scoring matrix in nucleotide-nucleotide BLAST [185] (blastn) (Fig. 43), which is one of the most widely used bioinformatics programs for sequence alignment searching:

	A	C	G	T
A	2	-3	-3	-3
C	-3	2	-3	-3
G	-3	-3	2	-3
T	-3	-3	-3	2

Fig. 43. The alignment scoring matrix in nucleotide-nucleotide BLAST [185] (blastn).

The scores represent the most sensitive mode in blastn (-task blastn) targeting sequences at 90% sequence identity [186].

2) The alignment scoring matrix in K80 model (also known as Kimura 2-parameter) [187] (Fig. 44), which distinguishes between transitions ($A \leftrightarrow G$ or $C \leftrightarrow T$) and transversions ($A \leftrightarrow C$ or $G \leftrightarrow T$).

	A	C	G	T
A	6	1	2	1
C	1	6	1	2
G	2	1	6	1
T	1	2	1	6

Fig. 44. The alignment scoring matrix in K80 model

Same gap penalty = -1 is used in both objectives.

Fig. 46 shows the resulting multi-objective score matrix F. Each cell containing a set of non-dominating score vectors, which represent the Pareto-optimal sub-alignments ending at this cell. The origin of each score vector is represented by an arrow initiated from one of the three neighbor cells. The yellow, green, and blue arrows represent score vectors generated from the match score set P, the insert score set U, and the delete score set Q. The four score vectors $\begin{bmatrix}-12\\22\end{bmatrix}$, $\begin{bmatrix}-5\\20\end{bmatrix}$, $\begin{bmatrix}-4\\17\end{bmatrix}$, and $\begin{bmatrix}-3\\13\end{bmatrix}$ in the cell at the down right corner correspond to the four complete Pareto-optimal front of alignments of sequences A and B with respect to the blastn and K80 model scores. Tracing these four score vectors through the matrix back to the origin, as shown in Fig. 47, generates the following four alignments at the Pareto-optimal front:

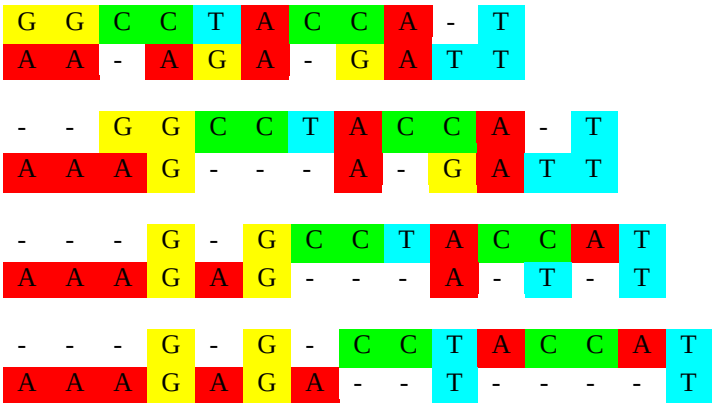


Fig. 45. The four alignments at the Pareto-optimal front.

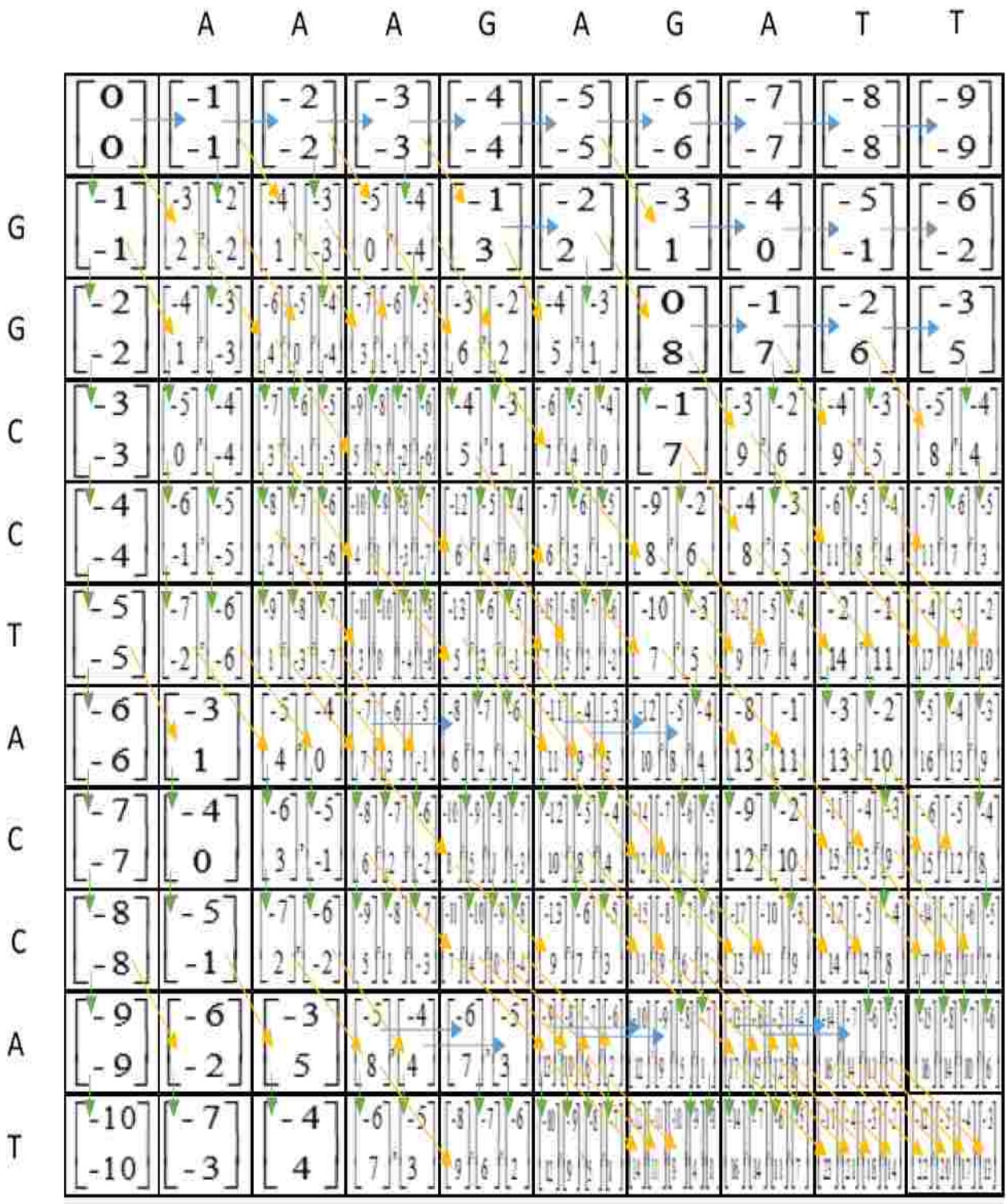


Fig. 46. The multi-objective score matrix F for the two DNA sequences X=GGCCTACCAT, and Y=AAAGAGATT, where the objectives are the blastn and K80 model.

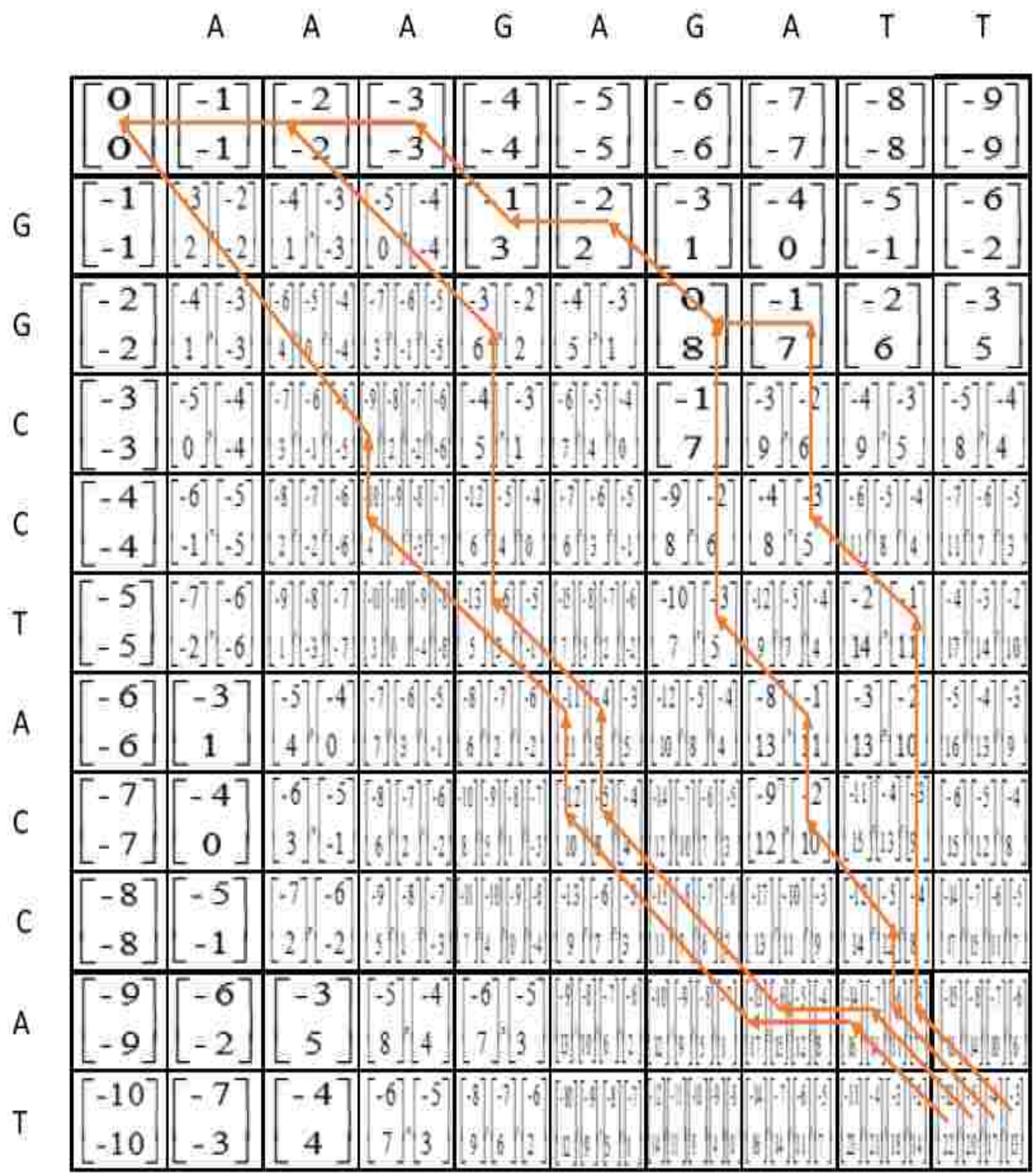


Fig. 47. Backtracking the Pareto-optimal alignments for the two DNA sequences X=GGCCTACCAT, and Y=AAAGAGATT, where the objectives are the Blastn and K80 model4. Discussion

5.2.3 Time and Space Complexity of Multi-Objective Alignment

Similar to the Needleman-Wunsch algorithm for pairwise sequence alignment under a single objective, the number of steps to filling the score matrix and that of backtracking are both $O(MN)$. However, unlike single-objective Needleman-Wunsch, the number of operations to fill out each cell as well as each trace back step in the score matrix are no longer constant, which instead is related to the number of Pareto-optimal alignments at each cell. Therefore, the overall time complexity of the multi-objective Needleman-Wunsch algorithm is $O(kMN)$, where k denotes the maximum number of Pareto-optimal alignments in each cell. Similarly, the space complexity of multi-objective Needleman-Wunsch is also $O(kMN)$. The value of k depends on the nature of the multiple objectives. If the multiple objectives are strongly positively correlated, k is usually small. In the extreme case, if all objectives are consistent, the multi-objective Needleman-Wunsch algorithm is equivalent to the Needleman-Wunsch algorithm under a single objective function. Whereas, if the multiple objectives are strongly conflicting, k can increase dramatically along the multi-objective optimization process. In the worst case, if every alignment generated from three neighboring cells are non-dominated, the total number of Pareto-optimal alignments can reach

$$\sum_{i=0}^{min(M,N)} \binom{M-i}{M+N-2i} \binom{i}{M+N-1}, \tag{15}$$

whose mathematical derivation based on three-dimensional Pascal triangle*.

5.2.4 Multi-Objective Alignment vs. Alignment by Optimizing a Weighted-Sum Consensus Function

A popular approach to combine multiple objectives is the weighted-sum method, where weights are assigned to various objective terms and a single consensus scoring function is built by linearly combining the weighted score terms. Here, the weights are typically determined by

* The formula was developed by Andrew Fu and Yanyu Jiang from Princess Anne High School, Virginia Beach, Virginia.

machine learning methods. Optimizing a weighted-sum function by combining multiple terms representing different objectives has been widely used in many sequence-alignment applications. Nevertheless, there is a fundamental difference between optimizing a consensus weighted-sum function as a single objective function and optimizing multiple objective functions in sequence alignment. That is, there is one optimal alignment (or a few optimal alignments if they yield the same objective function values) in optimizing a weighted-sum consensus function, whereas in multi-objective alignment, many Pareto-optimal solutions may exist due to the trade-offs between the conflicting objectives.

Although sequence alignment using a weighted-sum consensus function has been popularly used, it encompasses several issues. First, the weighted-sum function assumes the existence of a certain preference factor among the objectives that can be applied to deduce fixed weights to combine the objectives. However, the optimal weights may vary in aligning different pairwise sequences and thus there is unlikely a single set of weights that can satisfy all alignment situations, particularly when the objectives are strongly conflicting. In contrast, multi-objective alignment algorithms attempt to enumerate all Pareto-optimal alignments and thus is unnecessary to determine weights and thus is not sensitive to weights. Second, an optimal alignment with respect to a weighted-sum consensus function is Pareto-optimal; however, conversely, certain Pareto-optimal alignment may be unreachable when the Pareto optimal front is concave [188] [189] [190]. Fig. 48 illustrates a concave function space composed of two objective functions F_1 and F_2 and the multi-objective optimization problem is to maximize F_1 and F_2. When a set of weights are selected, a contour line such as “a” or “b” shown in Fig. 48 is formed. Maximizing the weighted-sum consensus function leads to a Pareto-optimal alignment, which is showed as the tangent point of the contour line and the feasible solution space. However, no contour line can

make a tangent point at the solutions on the concave Pareto-optimal front, regardless the weight selection, due to the fact that the contour line becomes a tangent with another point in the solution space before touching the concave Pareto-optimal front. In conclusion, some alignments yielding optimized compromise among objective terms are not reachable by maximizing any weighted-sum consensus functions, regardless the weights selection. Actually, even non-linear combinations used to integrate the objective functions may still leave certain Pareto-optimal alignments unreachable.

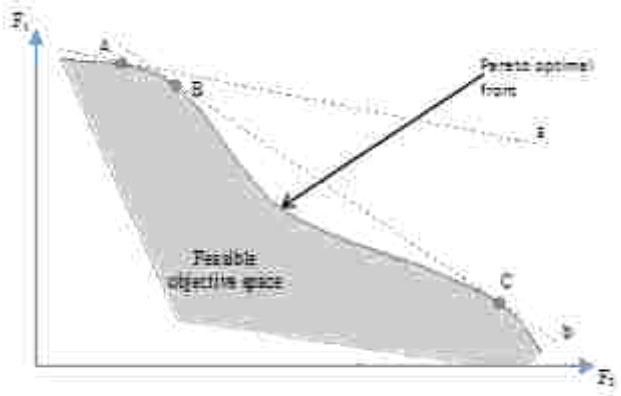


Fig. 48. Linear weight combinations of objectives fails to find some Pareto optimal solutions

5.2.5 Multi-Objective Needleman-Wunsch Alignment vs. Multi-Objective Genetic Algorithms

The multi-objective genetic algorithms (MOGA) [188] have been applied to pairwise sequence alignment. MOGAs explore the Pareto-optimal front and obtain diversified Pareto-optimal solutions by iteratively applying genetic operators such as mutations and crossovers to a population of candidate alignments. Similar to evolutionary algorithms in many other applications, MOGA mostly has difficulty to guarantee Pareto-optimality for the obtained alignments as well as ensuring all Pareto-optimal alignments are generated. In contrast, the multi-objective Needleman-

Wunsch algorithm not only guarantees Pareto-optimality, but also guarantees solution completeness.

5.2.6 Multi-Objective Needleman-Wunsch with Affine Gap

Algorithm 2 presented in Section 5.2 employs a linear gap penalty function, which treats open gaps and extending gaps the same. In many sequence alignment applications, it is more desirable to use an affine gap penalty function with g_o for opening a gap and g_e for extending a gap. Generally, g_o is a greater penalty than g_e. Accordingly, the penalty for a gap of length l becomes $g = g_o + lg_e$. DeRonne and Karypis [191] developed a dynamic programming approach extending Gotoh's sequence alignment algorithm with affine gap [139] to find pairwise sequence alignments at the Pareto optimal front, which requires calculation and maintenance of four score matrices. Here, we present a multi-objective Needleman-Wunsch algorithm with affine gap that requires only one score matrix and guarantees Pareto optimal completeness.

Different from the multi-objective Needleman-Wunsch algorithm with linear gap penalty, to handle affine gap, one needs to keep track of if each sub-alignment is generated from match/mismatch, insert, or delete of the previous step. A more challenging problem is, due to the fact that g_o is a bigger penalty than g_e, a dominated sub-alignment may actually lead to a non-dominated sub-alignment in future steps. Consequently, such a sub-alignment should not be eliminated in the domination function until it is clear that it has no possibility to generate non-dominated sub-alignments.

In the multi-objective Needleman-Wunsch algorithm with affine gap, there are two gap score vectors $\overrightarrow{Go}$ and $\overrightarrow{Ge}$, corresponding to the open gap and the extending gap penalties, respectively. For each sub-alignment $u_{m,n}$ terminating at cell $F_{m,n}$, we use a field $u_{m,n}.\vec{s}$ to keep track of the score vector, $u_{m,n}.i$ to indicate how u_i is generated (by match, insert, or delete) from

the previous step, and $u_{m,n}.\vec{g}$ to hold the accumulated gap penalties of of sub-alignment u_i under different objective functions. Then, for all sub-alignments terminating at cell $F_{m,n}$, we classify them into two sets: $F^*_{m,n}$, which contains all sub-alignments that can be safely eliminated if dominated and $V_{m,n}$, which contains sub-alignments that may be dominated in cell $F_{m,n}$ but have the potential to lead to non-dominated sub-alignments in future steps. The score vectors of the sub-alignments generated from the diagonal cell $F_{m-1,n-1}$ by match are updated by $\vec{S}_{m,n}$ only and are not affected by opening and extending gap penalies. Therefore, they can be eliminated safely if dominated and are deposited in $F^*_{m,n}$, as illustrated in the following pseudocode.

foreach $u_{m-1,n-1}$ in $F_{m-1,n-1}$

$u_{m,n}.\vec{s} \leftarrow u_{m-1,n-1}.\vec{s} + \vec{S}_{m,n}$

$u_{m,n}.i \leftarrow$ "$match$"

$u_{m,n}.\vec{g} \leftarrow \vec{0}$

$F^*_{m,n} \leftarrow F^*_{m,n} \cup \{u_{m,n}\}$

end

For the sub-alignments generated from insert or delete, only the dominated sub-alignments resulted from a new opening gap or from an extending gap having an accumulated extending gap score not less than the opening gap penalty can be safely eliminated. The following pseudocodes describe the generations of sub-alignments $u_{m,n}$ from $F_{m-1,n}$ by insert and from $F_{m,n-1}$ by delete.

// handling sub-alignments from $F_{m-1,n}$ by insert

foreach $u_{m-1,n}$ in $F_{m-1,n}$

$u_{m,n}.i \leftarrow$ "$insert$"

if $(u_{m-1,n}.i == "match" || u_{m-1,n}.i == "delete")$

$u_{m,n}.\vec{s} \leftarrow u_{m-1,n}.\vec{s} + \overrightarrow{Go} + \overrightarrow{Ge}$

$u_{m,n}.\vec{g} \leftarrow \vec{0}$

$F^*_{m,n} \leftarrow F^*_{m,n} \cup \{u_{m,n}\}$

elseif $(u_{m-1,n}.\vec{g} \geq \overrightarrow{Go})$

$u_{m,n}.\vec{s} \leftarrow u_{m-1,n}.\vec{s} + \overrightarrow{Ge}$

$u_{m,n}.\vec{g} \leftarrow u_{m-1,n}.\vec{g} + \overrightarrow{Ge}$

$F^*_{m,n} \leftarrow F^*_{m,n} \cup \{u_{m,n}\}$

else

$u_{m,n}.\vec{s} \leftarrow u_{m-1,n}.\vec{s} + \overrightarrow{Ge}$

$u_{m,n}.\vec{g} \leftarrow u_{m-1,n}.\vec{g} + \overrightarrow{Ge}$

$V_{m,n} \leftarrow V_{m,n} \cup \{u_{m,n}\}$

end

end

// handling sub-alignments from $F_{m,n-1}$ by delete

foreach $u_{m,n-1}$ in $F_{m,n-1}$

$u_{m,n}.i \leftarrow "delete"$

if $(u_{m,n-1}.i == "match" || u_{m,n-1}.i == "insert")$

$u_{m,n}.\vec{s} \leftarrow u_{m,n-1}.\vec{s} + \overrightarrow{Go} + \overrightarrow{Ge}$

$u_{m,n}.\vec{g} \leftarrow \vec{0}$

$F^*_{m,n} \leftarrow F^*_{m,n} \cup \{u_{m,n}\}$

elseif ($u_{m,n-1}.\vec{g} \geq \overrightarrow{Go}$)

$u_{m,n}.\vec{s} \leftarrow u_{m,n-1}.\vec{s} + \overrightarrow{Ge}$

$u_{m,n}.\vec{g} \leftarrow u_{m,n-1}.\vec{g} + \overrightarrow{Ge}$

$F^*_{m,n} \leftarrow F^*_{m,n} \cup \{u_{m,n}\}$

else

$u_{m,n}.\vec{s} \leftarrow u_{m,n-1}.\vec{s} + \overrightarrow{Ge}$

$u_{m,n}.\vec{g} \leftarrow u_{m,n-1}.\vec{g} + \overrightarrow{Ge}$

$V_{m,n} \leftarrow V_{m,n} \cup \{u_{m,n}\}$

end

end

The sub-alignments in $F_{m,n}$ are $V_{m,n} \cup F^*_{m,n}$. Then, the Dom(.) function is applied to $F_{m,n}$ to identify the non-dominated sub-alignments. However, only those in $V_{m,n}$ will be eliminated to guarantee the completeness of the Pareto optimal alignments.

5.2.7 Results

Similar to MOA, the CASP 11 experiment targets are used to demonstrate the effectiveness of MON. Here also, we used the same two scoring functions in the alignment generation (sequence profile $S_{seq}(i,j)$, and structural features including predicted secondary structures and solvent accessibility $S_{structure}(i,j)$) to measure the alignment between the *i*th residue in the query sequence and the *j*th residue in the template sequence.

MON is also compared with Muster [15] and GenTHREADER [83], two popularly used template alignment and selection methods for template-based protein structure modeling. Each target sequence is aligned with the same templates by the structure profile alignment method. Then, tertiary protein structure models are generated by the Modeller program [184] according to the alignments. The GDT-TS is used to measure the quality of these models and the corresponding alignments. Since MON generates all Pareto-optimal alignments, which is usually more than one, we only show the one that leads to the highest GDT-TS score.

We first compare MON and MUSTER on the top-ranked template of each target specified by MUSTER. The performance of MUSTER and MON on the CASP 11 targets are summarized in Table 10. It can be noticed that MON outperformed MUSTER despite using less objectives than MUSTER. Additionally, Fig. 49 shows the GDT-TS score for MUSTER along with the MON. As it appears in the figure that MON achieved a GDT-TS score higher or equal to that of MUSTER in 104 targets out of 115 total targets. Also MON GDT-TS score was greater than Muster by at least 10 in eight targets. A similar comparison is done between MON and pGenTHREADER, which is shown in Table 11 and Fig. 50. In 84 targets, the GDT-TS scores of models generated by MOA are higher than pGenTHREADER, where in 16 of them, the gain is at least 10.

Table 10
Overall performance of MUSTER and MON on the top-ranked template specified by Muster for the CASP11 targets.

Method	MUSTER	MON
Average GDT-TS	33.28	36.65

Table 11
Overall performance of GenTHREADER and MON on the top-ranked template specified by GenTHREADER for the CASP11 targets

Method	GenTHREADER	MON
Average GDT-TS	29.61	36.03

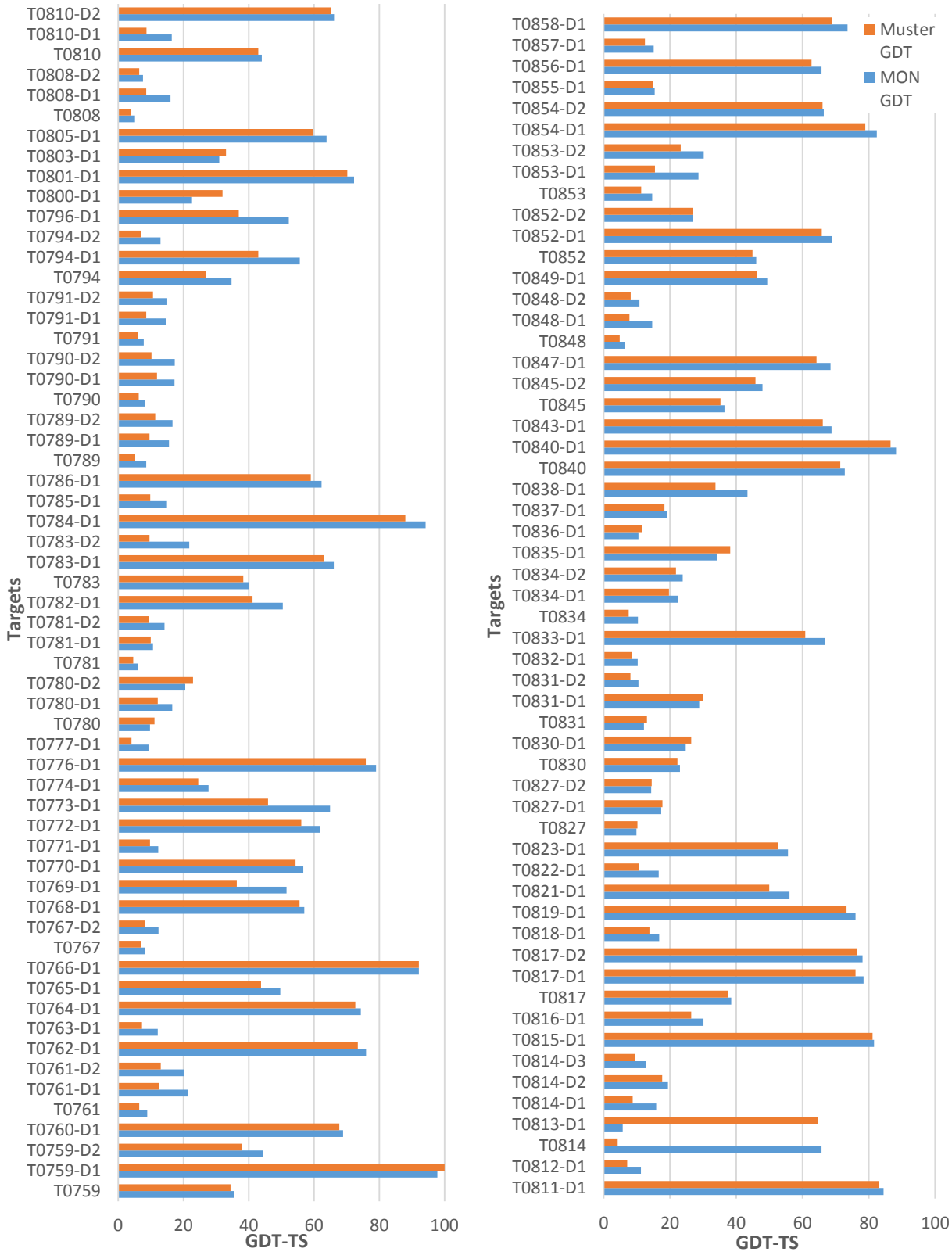


Fig. 49. The GDT-TS score of Muster alignment and MON alignment to CASP 11 targets with the top-ranked template selected by Muster. MON achieved a higher or equal GDT-TS score for 104 targets and most of the time MON eight of them the difference is more than 10 i.e. T0773-D1

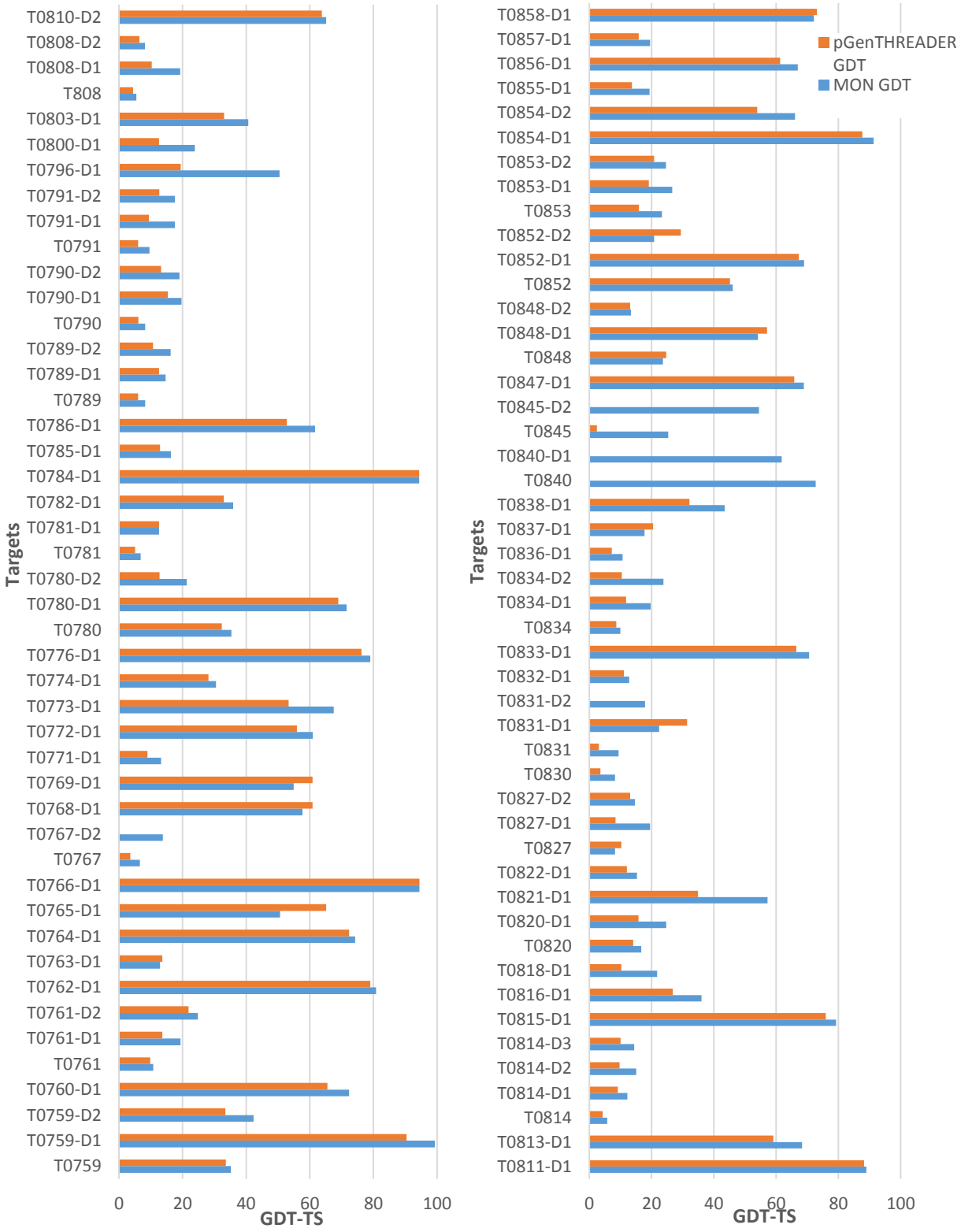


Fig. 50. The GDT-TS score of pGenTHREADER alignment and MON alignment to CASP 11 targets with the top-ranked template selected by pGenTHREADER. In 84 targets MON GDT-TS score is higher or equal pGenTHREADER, 16 of them MON GDT-TS score was 10 points higher than pGenTHREADER.

Another comparison is done with Muster and linear combination of objectives over CASP 11 on the top-ranked template of each target specified by CASP. The performance of MUSTER, linear combination, and MON on the CASP 11 targets are summarized in Table 12. From the table it is clear that MON outperformed MUSTER and linear combination. Fig. 51 shows the GDT-TS scores of the models generated by MON and Muster, where MON is able to generate at least one alignment with a higher GDT-TS score than MUSTER in 95 targets. Fig. 52 compares the GDT-TS scores of the models generated by MON and linear combination of objectives. One can find that the GDT-TS scores of the top models generated by MON are almost always better than those generated by linear combination of objectives. Particularly, the MON models exceed those generated by linear combination of objectives by at least 10 in GDT-TS score in 45 targets.

Table 12
Overall performance of MUSTER , linear combination objectives and MON on the top-ranked template specified by CASP for the CASP11 targets.

Method	MUSTER	Linear Combination	MON
Average GDT-TS	31.8	28.35	39.34

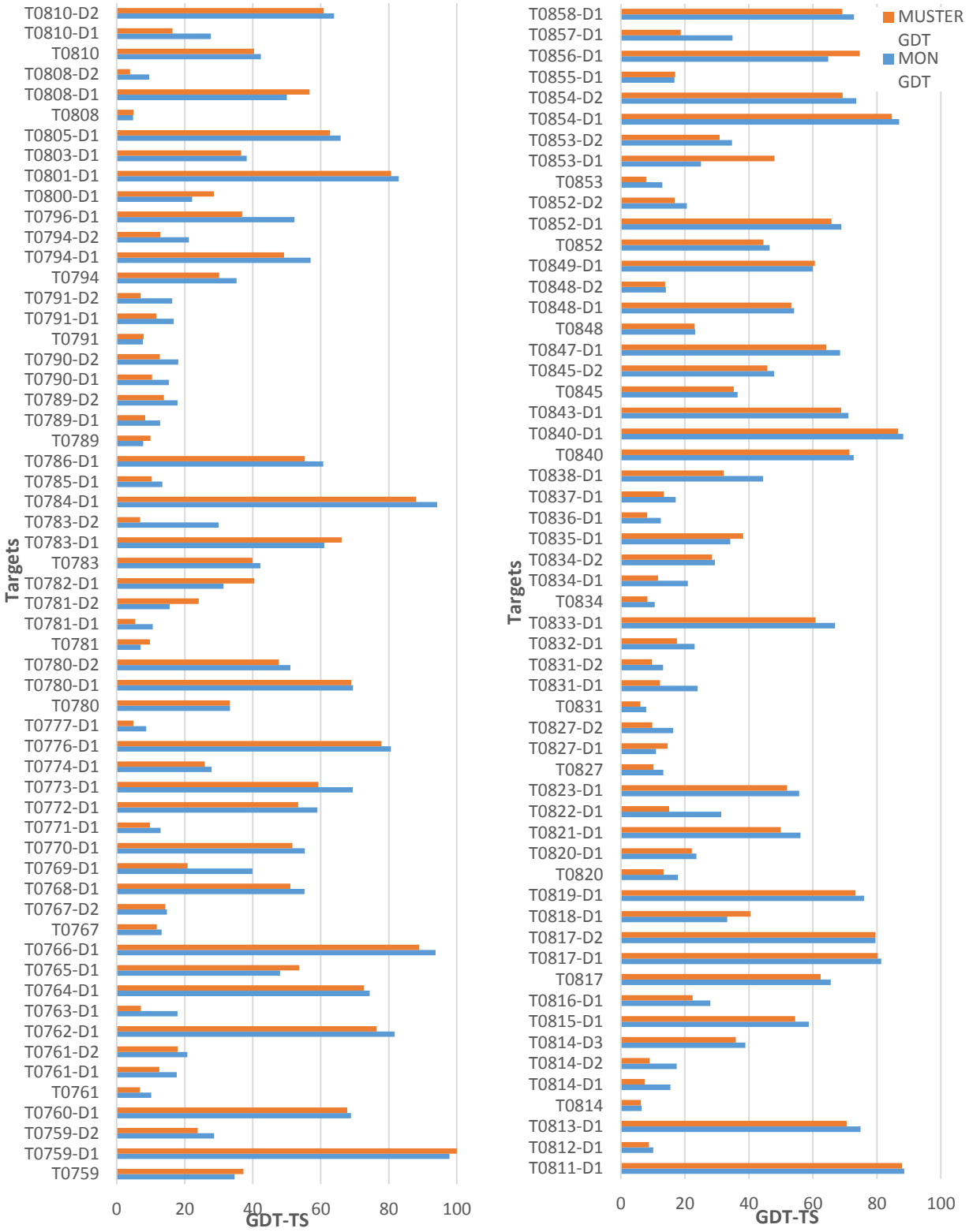


Fig. 51. The GDT-TS score of Muster alignment and MON alignment to CASP 11 targets with the top-ranked template selected by CASP. In 95 targets MON GDT-TS score is higher or equal Muster, 10 of them MON GDT-TS score was 10 points higher than Muster. i.e. T0769-D1. Muster achieved highly in 4 targets i.e T0782-D1

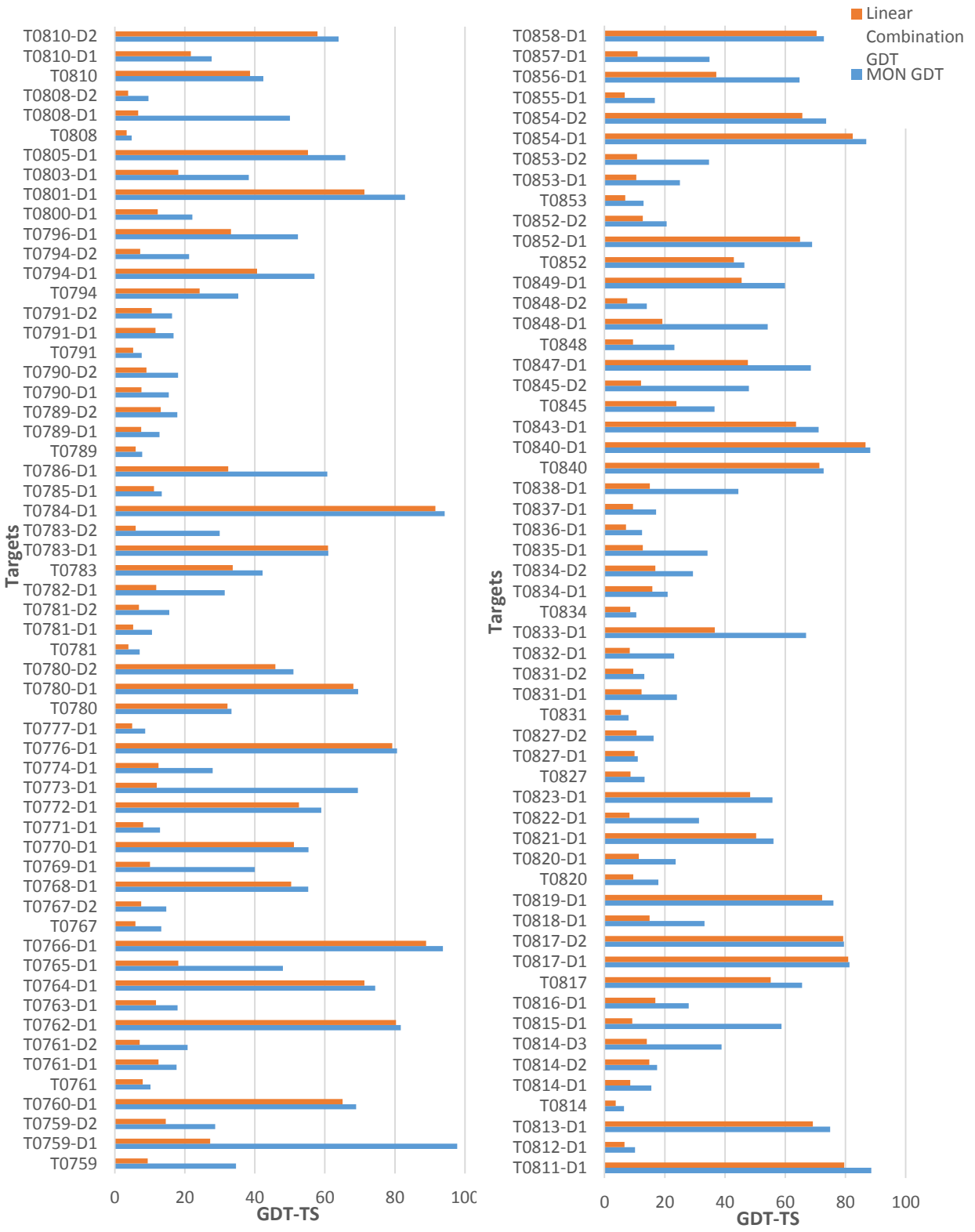
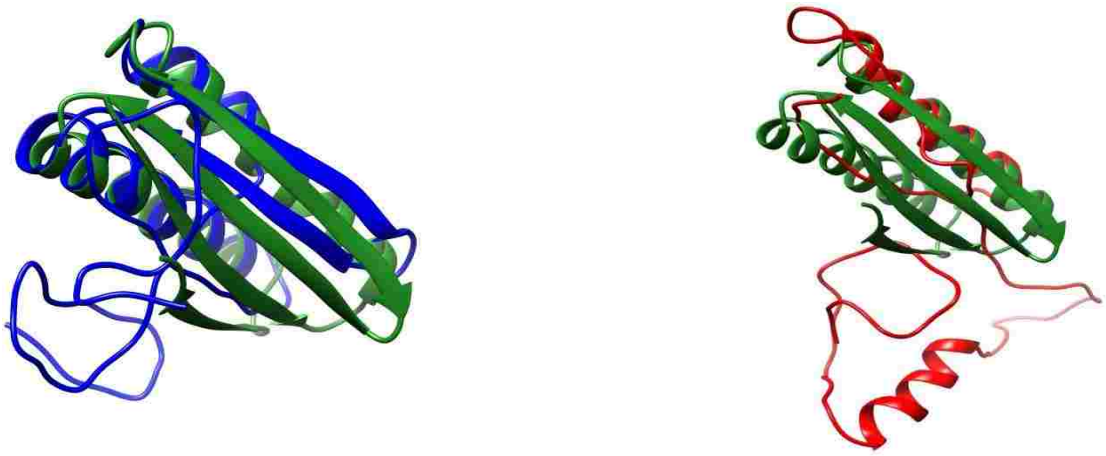


Fig. 52. The GDT-TS score of linear combination of objectives algorithm using same sequence and structure information and MON for CASP 11 targets with the top-ranked template selected by CASP. In 113 targets MON achieved higher or equal GDT-TS, most of them MON GDT-TS score was 10 points higher. i.e. T0759-D1. Only at T0776-D1 MON was lower and by a very small difference.

When comparing MUSTER to MON over CASP 11 on the top-ranked template of each target specified by CASP, the most significant enhancement occurs in targets T0769-D1 and T0796-D1, where the GDT-TS scores of the models generated from the alignments improved from 20.9 to 40.0 and from 36.9 to 52.3, respectively. Fig. 53 and Fig. 54 respectively display the alignments and models generated by MUSTER and MON in targets T0769-D1 and T0796-D1. It is interesting to notice that in T0769-D1, MON alignment improves the modeling of both the α-helix and the β-sheet regions. Hence, the main improvement is in the β-sheet regions, where MUSTER model (Fig. 53 (b)) does not include any β-strand while MON model (Fig. 53(a)) successfully identifies two β-strands. The improvement in the β-sheets alignment can also be found in T0796-D1. For example, the MON model (Fig. 54(a)) appears to have a more accurate alignment of the β-strands than the MUSTER model (Fig. 54(b)).



(a) MON (RMSD=10.39)

(b) MUSTER (RMSD=19.61)

By MON

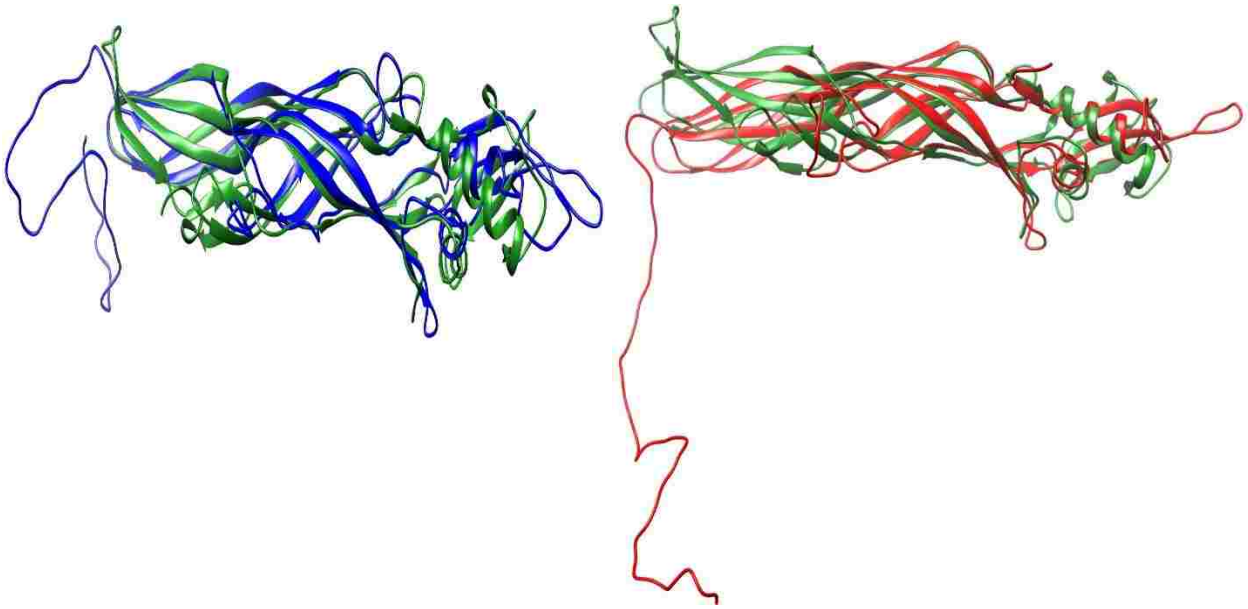
```
T0769-D1: -------------------------ML---------------------TVEVEV---------KITADDE
3ramD:    GEKQQILDYIETNKYSYIEISHRIHERPELGNEEIFASRTLIDRLKEHDFEIETEIAGHATGFIATYDSGLD
T0769-D1: N-------------------KAE--------------EIVKR----------------------------V
3ramD:    GPAIGFLAEYDALPGLGHACGHNIIGTASVLGAIGLKQVIDQIGGKVVVLGCPAEEGGENGSAKASYVKAGV
T0769-D1: ID-------------------------------------------------------EVEREVQKQYPNATITR
3ramD:    IDQIDIALIHPGNETYKTIDTLAVDVLDVKFYGKSAHASENADEALNALDAISYFNGVAQLRQHIKKDQRVH
T0769-D1: TLTRDDG--------TVELRIKVKADTEEKAKSIIKLIEERIEEELRKRDPNATITR---------------
3ramD:    GVILDGGKAANIIPDYTHARFYTRATRKE-LDILTEKVNQIARGAAIQTGCDYEFGPIQNGVNEFIKTPKLD
T0769-D1: ------------TVRTEVGSSWS----------------------LEHHHHH------------------
3ramD:    DLFAKYAEEVGEAVIDDDFGYGSTDTGNVSHVVPTIHPHIKIGSRNLVGHTHRFREAAASVHGDEALIKGAK
T0769-D1: -------------------------H*
3ramD:    IALGLELITNQDVYQDIIEEHAHLKG*
```

By MUSTER

```
T0769-D1:------------------------------------------------------------------------
3ramD:   GEKQQILDYIETNKYSYIEISHRIHERPELGNEEIFASRTLIDRLKEHDFEIETEIAGHATGFIATYDSGLDG
T0769-D1:------------------------------------------------------------------------
3ramD:   PAIGFLAEYDALPGLGHACGHNIIGTASVLGAIGLKQVIDQIGGKVVVLGCPAEEGGENGSAKASYVKAGVID
T0769-D1:------------------------------------------------------------------------
3ramD:   QIDIALIHPGNETYKTIDTLAVDVLDVKFYGKSAHASENADEALNALDAISYFNGVAQLRQHIKKDQRVHGVI
T0769-D1:---------ML-TVEVEVKITADDENKAEEIVKRVIDEVEREVQKQYP----NATITRTLTRDDGTVELRIKV
3ramD:   LDGGKAANIIPDYTHARFYTRAT-----RKELDILTEKVNQIARGA--AIQTGCDYEFGPIQN-GVNEFI---
T0769-D1:KADTEEKAKSII--KLIEERIEEELRKRDPNATITRTV--RTEVGSS------------------WSLEHHHH
3ramD:   -----------KTPKLDDLFAKYAEEVGEAVI-----DDDFGYGSTDTGNVSHVVPTIHPHIKIGSRNLVGH
T0769-D1:HH-----------------------------------------------*
3ramD:   THRFREAAASVHGDEALIKGAKIALGLELITNQDVYQDIIEEHAHLKG*
```

(c) Alignment between 3ramD and T0769-D1

Fig. 53. The best scoring alignments generated from MON and that generated by Muster for T0769-D1 and 3ramD. The model generated from MON alignment scores 40.0 GDT-TS while Muster scores only 20.9



(a) MON (RMSD=8.8600) (b) MUSTER (RMSD=15.7166)

By MON

```
T0796-D1: MIFLAILDL–KSLVLNAINYWGPKNNNGIQGGDFGYPISEKQIDTSIITSTHPRLIPHDLTIPQNLETI
2d42A:    A----IINLLRELEIY-GMQY---ANSHQ-----YTYGSSYSDDTNPIRIAGLDARI-PDPIVTDPVNH
T0796-D1: FTTTQVLTNNTDLQQSQTVSFAKKTTTTTSTSTTNGWTEGGKISDTLEEKVSVSIPFIGEGGGKNSTTI
2d42A:    IVLDRRIITNTTSNSLEGVFSFSNAYTSRTSSQTRDGVTAGTN--ITGKYFANLF-----FEQVGLSGR
T0796-D1: EANFAHNSSTTTFQQASTDIEWNISQPVLVPPRKQVVATLVIMGGNFTIPMDLMTTIDSTEHYSGYPIL
2d42A:    IAFEG–AVTNENKYTLDATQDFRDSQTIRVPPFHRATGVYTLEQGAFEKMTVLECVVSGNGIIRYYRTL
T0796-D1: TWISSPDNSYNGPFMSWYFANWPNLPSGFGPLNSDNTVTYTGSVVSQVSAGVYATVRFDQYDIHNLRTI
2d42A:    PDNSYTEIVQR--VNIIDVLQANGTPG-FTISKEQNRAYFTGEGTISGQIGLQTFIDVVIEPLPGH---
T0796-D1: EKTWYARHATLHNGKKISINNVTEMAPTSPIKTN*
2d42A:    ---------------------------------A*
```

By MUSTER

```
T0796-D1: MIFLAILDLKSLVLNAINYWGPKNNNGIQGGDFGYPISEKQIDTSIITSTHPRLIPHDLTIPQNLETIF
2d42A:    ----AIINLLRELEIYGMQYA---NSHQYTYGSSYSDDTNPIRIAGLDARIPDP-----IVTDPVNHIV
T0796-D1: TTTQVLTNNTDLQQSQTVSFAKKTTTTTSTSTTNGWTEGGKISDTLEEKVSVSIPFIGEGGGKNSTTIE
2d42A:    LDRRIITNTTSNSLEGVFSFSNAYTSRTSSQTRDGVTAGTNITGKYFANLFFE---------QVGLSGR
T0796-D1: ANFAHNSSTTTFQQASTDIEWNISQPVLVPPRKQVVATLVIMGGNFTIPMDLMTTIDSTEHYSGYPILT
2d42A:    IAFEGAVTNENKYTLDATQDFRDSQTIRVPPFHRATGVYTLEQGAFEKMTVLECVVSGNGIIRYYRTLP
T0796-D1: WISSPDNSYNGPFMSWYFANWPNLPSGFGPLNSDNTVTYTGSVVSQVSAGVYATVRFDQYDIHNLRTIE
2d42A:    DNSYTEIVQ--RVNIIDVLQANGTPGFTIS-KEQNRAYFTGEGTISGQIGLQTFIDVVIEPLPGHA---
T0796-D1: KTWYARHATLHNGKKISINNVTEMAPTSPIKTN*
2d42A:    ---------------------------------*
```

(c) Alignment between 2d42A and T0796-D1

Fig. 54. The best scoring alignments generated from MON and that generated by Muster for T0796-D1 and 2d42A. The model generated from MON alignment scores 52.3 GDT-TS while Muster scores only 36.9

Targets T0759-D1 and T0773-D1 are picked for further analysis of the comparison between the MON algorithm and the linear combination of objectives algorithm. T0759-D1 and T0773-D1 are aligned with the corresponding top-ranked templates 1lm5B1 and 3opkA, according to CASP11, resulting in 116 and 225 Pareto-optimal alignments, respectively. The best models generated by MON in T0759-D1 and T0773-D1 yield GDT-TS scores of 97.79 and 69.4, respectively, which are significantly higher than those generated by the linear combination of objectives (27.21 and 11.94). Fig. 55 and Fig. 57 show the profile and secondary structure/solvent accessibility scores of the generated alignments for T0759-D1 and T0773-D1, respectively. It is observed that for the two targets the scores of the linear combination of objectives alignments are dominated by all the Pareto-optimal alignments generated by MON.

In Fig. 56, we show the alignments and their corresponding models, produced by MON and linear combination of objectives for T0759-D1 with 1lm5B1. The target is an α-helix protein, thus an accurate alignment should correctly align the target and template α-helix regions. It appears that the MON alignment of the α-helices is highly accurate and is capable of generating an almost perfect model for T0759-D1 (Fig. 56(a)), while in the linear combination of objectives model, the α-helices are shifted from the correct ones (Fig. 56(b)). Also, in Fig. 58 it is shown that the main improvement of the MON alignment is in the α-helix residues. For instance, the MON model accurately aligns the two α-helices (Fig. 58(a)) while the linear combination of objectives model (Fig. 58(b)) fails to align one of the α-helix and shifts the other.

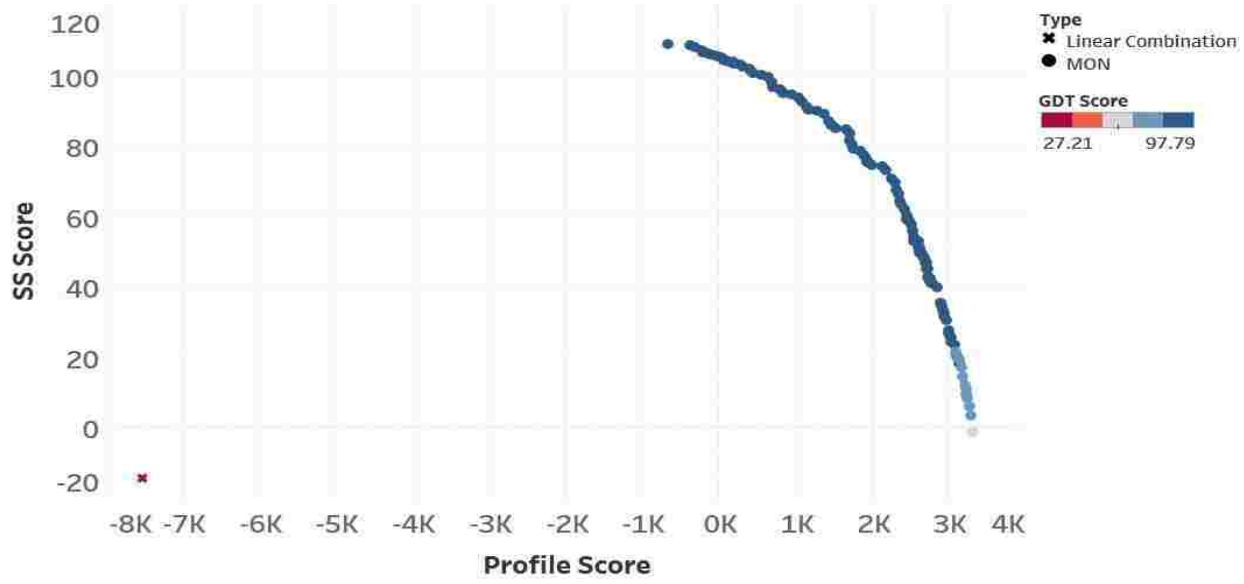
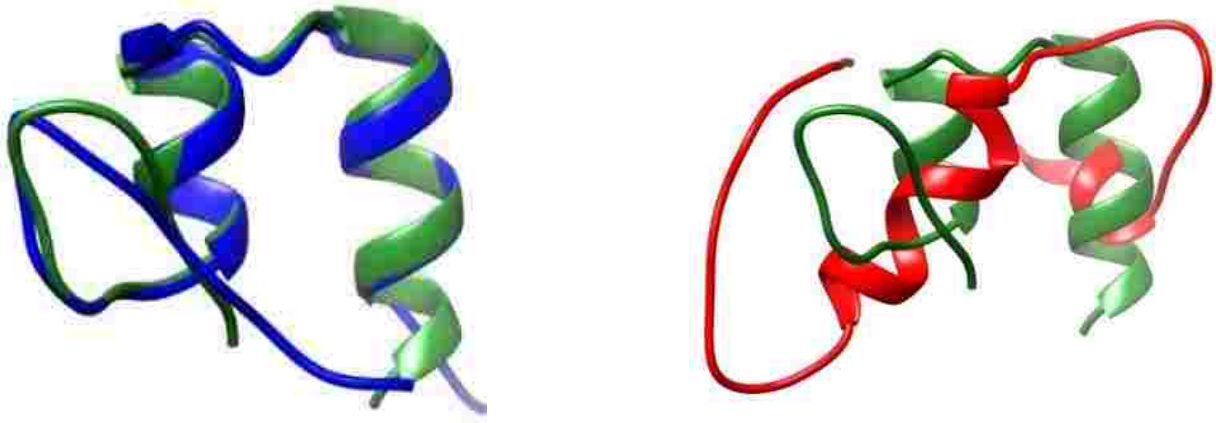


Fig. 55. Results for T0759-D1 alignment with 1lm5B



(a) MON (RMSD= 1.5237) (b) Linear Combination (RMSD=14.2181)

By MON

```
T0759-D1: MGHHHHHHSHMV-VIHPDPGRELSPEEAHRAGLIDWNMFVKLRS-Q-ECD------------------
1lm5B1:   ----------IAAIFDTENLEKISITEGIERGIVDSITGQRLLEAQA---CTGGIIHPTTGQKLSLQDA
T0759-D1: ------------------------------------------------*
1lm5B1:   VSQGVIDQDMATRLKPAQKAFIGFEGVKKMSAAEAVKEKWLPYEAGQRFLE*
```

By Linear Combination

```
T0759-D1: --------------------------MGHHHHHHSH-----MV------V--IHPDP--GR---ELSP
1lm5B1:   IAAIFDTENLEKISITEGIERGIVDSITGQRLLEAQACTGGIIHPTTGQKLSLQDAVSQG-VIDQDMAT
T0759-D1: EEAHRAGLID----------WNMFV------KLRSQECD*
1lm5B1:   RLKPAQKAFIGFEGVKKMSAAEAVKEKWLPYEAGQRFLE*
```

(c) Alignment between 1lm5B1 and T0759-D1

Fig. 56. The best scoring alignments generated from MON and that generated by linear combination of objectives for T0759-D1 and 1lm5B1. The model generated from MON alignment scores 97.79 GDT-TS while linear combination of objectives scores only 27.21

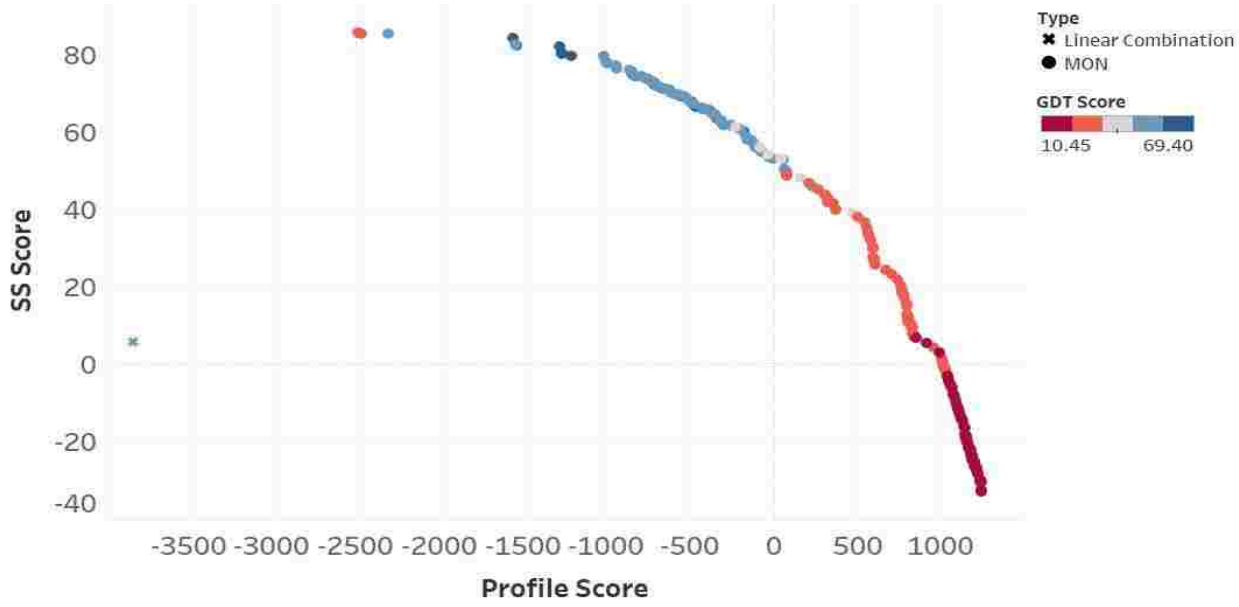
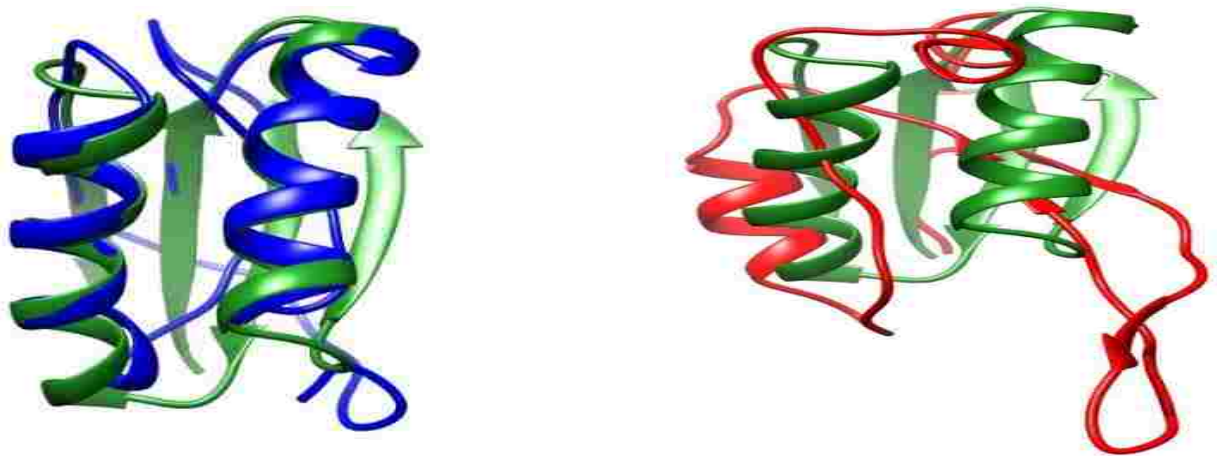


Fig. 57. Results for T0773-D1 alignment with 3opkA



(a) MON (RMSD= 3.6513) (b) Linear Combination (RMSD= 13.4021)

BY MON

```
T0773-D1: --MVDLKIDVSDDEEAEKIIREIREQWPKATVT---------RTNGDIKLDA-------QTEKEAEK
3opkA:    PEAVVVLCTAPDEATAQDLAAKVLAEKLAACATLLPGATSLYYWEGKLEQEYEVQMILKTTVSHQQA
T0773-D1: MEKAVKKVKP--NATI---RKTGGS---LEH-HHHHH*
3opkA:    LIDCLKSHHPYQTPELLVLPVTHGDTDYLSWLNASLR*
```

BY Linear Combination

```
T0773-D1: --------------------MVDL--K---IDVSDDEEAEKII-R-EIREQWPKATVTRT-NGDIKL
3opkA:    PEAVVVLCTAPDEATAQDLAAKVLAEKLAACATLLPGATSLYYWEGKLEQEYEVQMILKTTVSHQQA
T0773-D1: DAQTEKE--A-EKMEKAVKKVKP----NATIRKTGGSLEHH-HHHH*
3opkA:    LIDCLKSHHPYQ-TP--ELLVLPVTHGD--TDYLSWLNA-SLR---*
```

(c) Alignment between 3opkA and T0773-D1

Fig. 58. The best scoring alignments generated from MON and that generated by linear combination of objectives for T0773-D1 and 3opkA. The model generated from MON alignment scores 69.4 GDT-TS while linear combination of objectives scores only 11.94

5.3 Summary

Protein sequence alignment is fundamental to many problems in biology, such as protein structure modeling, protein design, and functional annotation of proteins. In template based protein structure modeling, protein sequence alignments discovers the shared similarity between the target and template sequences. The success of template-based modeling highly relies accurately producing a sequence alignment that maps the residues of the target sequence to those of the template. In this work, we present two multi-objective alignment algorithms to obtain a set of diversified alignments yielding Pareto optimality. The first algorithm is a preliminary multi-

objective alignment algorithm to examine the suitability of multi-objective alignment in protein structure modeling. The preliminary multi-objective alignment algorithm shows competitive results compared to other state-of-the-art algorithms [179]. Accordingly, we develop a multi-objective alignment algorithm based on the Needleman-Wunsch algorithm. The multi-objective Needleman-Wunsch algorithm guarantees not only Pareto optimality of the alignments, but also completeness. The proposed algorithm has been used to generate potentially more accurate protein sequence alignments that shall improve the performance of protein structure modeling. The multi-objective Needleman-Wunsch algorithm is examined on a set of CASP11 targets using the following objectives: (1) sequence profile, (2) secondary structure, and solvent accessibility objective functions. The multi-objective Needleman-Wunsch algorithm has demonstrated competitive results compared to other state of art methods.

CHAPTER VI

CONCLUSION AND FUTURE WORK

In this chapter, I summarize the contribution of this dissertation and discuss future research directions.

6.1 Summary

Computationally modeling protein structure from its sequence is a grand challenge with broad scientific and economic impacts. Today, one of the most accurate and consistent methodologies for computational protein structure modeling is template-based modeling. The success of template-based modeling relies on correctly identifying one or a few experimentally determined protein structures as templates that are likely to resemble the structure of the target sequence as well as accurately producing a sequence alignment that maps the residues of the target sequence to those of the template. Therefore, addressing these tasks is the key to improving the accuracy of template-based protein structure modeling.

This work takes advantage of an inter residue contact scoring function to measure the favorability of a target sequence fitting in the folding topology of a certain template. This is performed by placing the target sequence residues into the mapped template residues three-dimensional conformation and evaluating the contact score. Then, we combine the contact score with the sequence profile score to enhance template selection sensitivity. This approach has shown a notable improvement in the accuracy and sensitivity of template selection in template-based protein structure modeling [16].

After the recognizable progress that is achieved in template selection using our first approach, we present a second template selection approach that employs three-dimensional information of protein in a more efficient way. In this approach, instead of evaluating the

favorability of a target adopting a potential structural template after an alignment is generated, we use the three-dimensional information to build the alignment along with other structural features. The idea is to build a substitution matrix to score the replacement of one amino acid of the template three-dimensional conformation with each amino acid in the target. Then, we can use this substitution matrix to incorporate three-dimensional information in building the alignment along with the structural features. Consequently, the structural profile alignment between the target and templates are totally performed using our own alignment algorithm. The alignment is done by dynamic programming that exploits several protein structural features in addition to the three dimensional features. The template selection approach is tested over CASP 11 targets and has shown a significant improvement compared to the successful template alignment and selection methods.

Furthermore, we present two multi-objective alignment algorithms to obtain a set of diversified alignments yielding Pareto optimality. The first algorithm is a preliminary multi-objective alignment algorithm to examine the suitability of multi-objective alignment in protein structure modeling. The preliminary multi-objective alignment algorithm shows competitive results compared to other state-of-the-art algorithms [179]. Accordingly, we develop a multi-objective alignment algorithm based on the Needleman-Wunsch algorithm. The multi-objective Needleman-Wunsch algorithm guarantees not only Pareto optimality of the alignments, but also completeness. The proposed algorithm has been used to generate potentially more accurate protein sequence alignments that shall improve the performance of protein structure modeling. The multi-objective Needleman-Wunsch algorithm is examined on a set of CASP11 targets using the following objectives: (1) sequence profile, (2) secondary structure, and solvent accessibility

objective functions. The multi-objective Needleman-Wunsch algorithm has demonstrated competitive results compared to other state of art methods.

6.2 Future Work

In this dissertation we aim at improving the template-based protein structure modeling by correctly identifying the most appropriate template protein structures and precisely align the target and template sequences. However, there are several interesting aspects that we would like to explore in order to further enhance the template-based protein structure modeling.

Firstly, in the presented template selection approach, we only consider the row alignment score to rank the templates. However, it is noticed that the model selected by the row alignment score is not always the best model generated by the alignment. Accordingly, the selected structure template is not the most appropriate template for a given target sequence. Consequently, it would be interesting to explore employing a machine learning technique that makes use of all of protein structural features employed in the alignment along with the alignment score to select a better template. Such a technique will act as a recommendation system that makes use of the similarity between the available protein structures in selecting the most appropriate template for a target sequence

Secondly, the multi-objective Needleman-Wunsch algorithm is examined using only two objectives: (1) sequence profile, (2) secondary structure + solvent accessibility. Our future work direction will employ more objective functions in the protein sequence alignments, such as fragment profiles. Additionally, the number of Pareto-optimal sub-alignments may grow dramatically along the multi-objective optimization iterations. However, in practice, one is often only able to process a small number of representative alignments. Therefore, it is important that these limited number of alignments are selected to evenly distribute on the Pareto-optimal front so

that the maximum diversity about the Pareto-optimal front is maintained. Consequently, in addition to optimality, another important criteria to assess the quality of the multi-objective optimization algorithms are solution diversity and uniformity. We will explore developing an empirical, greedy approach to limit the number of generated alignments while maintaining solution diversity.

REFERENCES

[1] K. A. Dill, S. Bromberg, K. Yue, K. M. Fiebig, D. P. Yee, P. D. Thomas and H. S. Chan, "Principles of protein folding -A perspective from simple exact models," *Protein Science,* vol. 4, no. 4, pp. 561-602, 1995.

[2] "RCSB Protein Data Bank, Joint Center for Structural Genomics (JCSG)," 20 11 2007. [Online]. Available: http://www.rcsb.org/pdb/explore/remediatedSequence.do?structureId=3BB5. [Accessed 11 5 2018].

[3] H. Rangwala and G. Karypis, Introduction to Protein Structure Prediction: Methods and Algorithms, Hoboken, New Jersey: Wiley, 2010.

[4] G. Pandey, V. Kumar and M. Steinbach, "Computational Approaches for Protein Function Prediction: A Survey. TR 06-028," Department of Computer Science and Engineering, University of Minnesota, Minneapolis, MN, 2006.

[5] D. Lee, O. Redfern and C. Orengo, "Predicting protein function from sequence and structure," *Nature Reviews. Molecular Cell Biology,* vol. 8, no. 12, pp. 995-1005, 2007.

[6] UniProt Consortium, "UniProt: the universal protein knowledgebase," *Nucleic Acids Research,* vol. 45, no. D1, p. D158–D169, 2017.

[7] J. Moult, K. Fidelis, A. Kryshtafovych, T. Schwede and A. Tramontano, "Critical assessment of methods of protein structure prediction (CASP)—round x," *Proteins: Structure, Function, and Bioinformatics,* vol. 82, no. S2, pp. 1-6, 2014.

[8] J. Moult, K. Fidelis, K. Andriy, T. Schwede and A. Tramontano, "Critical assessment of methods of protein structure prediction: Progress and new directions in round XI," *Proteins: Structure, Function, and Bioinformatics,* vol. 84, no. S1, pp. 4-14, 2016.

[9] J. Moult, K. Fidelis, A. Kryshtafovych, T. Schwede and A. Tramontano, "Critical assessment of methods of protein structure prediction (CASP)—Round XII," *Proteins: Structure, Function, and Bioinformatics,* vol. 86, pp. 7-15, 2018.

[10] Y. Zhang, "Progress and challenges in protein structure prediction," *Current Opinion in Structural Biology,* vol. 18, no. 3, pp. 342-348, 2008.

[11] J. Bowie, R. Luthy and D. Esenberg, "A method to identify protein sequences that fold into a known three-dimensional structure," *Science,* vol. 253, pp. 164-170, 1991.

[12] D. Jones, W. Taylor and J. Thornton, "A new approach to protein fold recognition," *Nature,* vol. 358, pp. 86-89, 1992.

[13] R. B. Best, G. Hummer and W. A. Eaton, "Native Contacts Determine Protein Folding Mechanisms in Atomic Simulations," *Proceeding of the National Academy of Science of the United States of America,* vol. 110, no. 44, pp. 17874-17879, 2013.

[14] W. Elhefnawy, L. Chen, Y. Han and Y. Li, "ICOSA: A Distance-Dependent, Orientation-Specific Coarse-Grained Contact Potential for Protein Structure Modeling," *Journal of Molecular Biology,* vol. 427, no. 15, pp. 2562-2576, 2015.

[15] S. Wu and Y. Zhang, "MUSTER: Improving protein sequence profile-profile alignments by using multiple sources of structure information," *Proteins,* p. 547–556, 2008.

[16] M. Abdelrasoul and Y. Li, "Coarse-Grained Contact Potential Helps Improve Fold Recognition Sensitivity in Template-Based Protein Structure Modeling," in *Big Data and*

Cloud Computing (BDCloud), Social Computing and Networking (SocialCom), Sustainable Computing and Communications (SustainCom)(BDCloud-SocialCom-SustainCom), 2016 IEEE International Conferences, 2016.

[17] C. Rye, R. Wise, O. V. Jurukovski, J. Desaix, J. Choi and Y. Avissar, Biology, Houston, Texas: OpenStax, 2013.

[18] F. Sanger, E. P. Thompson and R. Kitai, "The Amide Groups of Insulin," *Biochemistry Journal,* vol. 59, no. 3, pp. 509-518, 1955.

[19] J. Kendrew, G. Bodo, H. Dintzis, R. Parrish and H. Wyckoff, " A Three-Dimensional Model of the Myoglobin Molecule Obtained by X-ray Analysis," *Nature,* vol. 181, pp. 662-666, 1958.

[20] J. Kendrew, R. E. Dickerson, B. E. Strandbreg, R. G. Hart , D. R. Davis, D. C. Phillips and V. C. Shore, "Structure of Myoglobin: A Three-Dimensional Fourier synthesis at 2 Å Resolution," *Nature,* vol. 185, pp. 422-427, 1960.

[21] H. M. Berman, J. Westbrook, Z. Feng, G. Gilliland, T. N. Bhat, H. Weissig, L. N. Shindyalov and P. E. Bourne, "The Protein Data Bank," *Nucleic Acids Research,* vol. 28, no. 1, pp. 235-242, 2000.

[22] K. Wüthrich, NMR in Biological Research: Peptides and Proteins, North Holland, Amsterdam, 1976.

[23] K. Wüthrich, "The way to NMR structures of proteins," *Nature Structural Biology,* vol. 8, pp. 923 - 925, 2001.

[24] M. Adrian, J. Dubochet, J. Lepault and A. W. Mcdowall, "Cryo-electron microscopy of viruses," *Nature,* vol. 308, pp. 32 - 36, 1984.

[25] C. Dellisanti, "A barrier-breaking resolution," *Nature Structural & Molecular Biology,* vol. 22, no. 5, p. 361, 2015.

[26] A. Liwo, J. Lee, D. R. Ripoll, J. Pillardy and H. A. Scheraga, "Protein structure prediction by global optimization of a potential energy function," *Proceedings of the National Academy of Sciences,* vol. 96, no. 10, pp. 5482-5485, 1999.

[27] J. Pillardy, C. Czaplewski, A. Liwo, W. J. Wedemeyer, J. Lee, D. R. Ripoll, S. Oldziej, Y. A. Arnautova and H. A. Scheraga, "Development of Physics-Based Energy Functions that Predict Medium-Resolution Structures for Proteins of the α, β, and α/β Structural Classes," *The Journal of Physical Chemistry,* vol. 105, no. 30, pp. 7299-7311, 2001.

[28] J. Bowie and D. Eisenberg, "An Evolutionary Approach to Folding Small Alpha-Helical Proteins that Uses Sequence Information and an Empirical Guiding Fitness Function," *Proceedings of the National Academy of Sciences of the United States of America,* vol. 91, no. 10, pp. 4436-4440, 1994.

[29] K. Simons, C. Kooperberg, E. Huang and D. Baker, "Assembly of protein tertiary structures from fragments with similar local sequences using simulated annealing and Bayesian scoring functions," *Journal of Molecular Biology,* vol. 268, no. 1, pp. 209-225, 1997.

[30] S. Ovchinnikov, H. Park, D. E. Kim, F. DiMaio and D. Baker, "Protein structure prediction using Rosetta in CASP12," *Proteins: Structure, Function, and Bioinformatics,* vol. 86, pp. 113-121, 2018.

[31] S. Ovchinnikov, . H. Park, D. E. Kim, F. DiMaio and D. Baker, "Protein structure prediction using Rosetta in CASP12," *Proteins: Structure, Function, and Bioinformatics,* vol. 86, pp. 113-121, 2018.

[32] Y. Zhang, A. Kolinski and J. Skolnick, "TOUCHSTONE II: A New Approach to Ab Initio Protein Structure Prediction," *Biophysical Journal,* vol. 85, no. 2, pp. 1145-1164, 2003.

[33] J. Klepeis and C. Floudas, "ASTRO-FOLD: A Combinatorial and Global Optimization Framework for Ab Initio Prediction of Three-Dimensional Structures of Proteins from the Amino Acid Sequence," *Biophysical Journal,* vol. 85, no. 4, pp. 2119-2146, 2003.

[34] J. L. Klepeis, Y. Wei, M. H. Hecht and C. A. Floudas, "Ab Initio Prediction of the Three-Dimensional Structure of a De Novo Designed Protein: A Double-Blind Case Study," *Proteins: Structure, Function, and Bioinformatics,* vol. 58, pp. 560-570, 2005.

[35] A. Liwo, M. Khalili, C. Czaplewski, S. Kalinowski, S. Oldziej, K. Wachucik and H. A. Scheraga, "Modification and optimization of the united-residue (UNRES) potential-energy function for canonical simulations. I. Temperature dependence of the effective energy function and tests of the optimization method with single training proteins," *Journal of Physical Chemistry ,* vol. 111, no. 1, pp. 260-285, 2007.

[36] D. Xu and Y. Zhang, "Ab initio protein structure assembly using continuous structure fragments and optimized knowledge-based force field," *Protein: Structure, Function, and Bioinformatics,* vol. 80, no. 7, pp. 1717-1735, 2012.

[37] W. Zhang, J. Yang, B. He, S. E. Walker, H. Zhang, B. Govindarajoo, J. Virtanen, Z. Xue, H.-B. Shen and Y. Zhang, "Integration of QUARK and I-TASSER for Ab Initio Protein

Structure Prediction in CASP11," *Proteins: Structure, Function, and Bioinformatics,* vol. 84, no. S1, pp. 76-86, 2016.

[38] M. Tress, I. Ezkurdia, O. Grana, G. Lopez and A. Valencia, "Assessment of predictions submitted for the CASP6 Comparative Modeling Category," *Proteins,* vol. 61, no. suppl 7, pp. 27-45, 2005.

[39] G. Wang, Y. Jin and R. Dunbrack Jr, "Assessment of Fold Recognition Predictions in CASP6," *Proteins,* vol. 61, no. Suppl 7, pp. 46-66, 2005.

[40] J. kopp, L. Bordoli, J. Battey, F. Kiefer and T. Schwedek, "Assessment of CASP7 Predictions for Template-Based Modeling Targets," *Proteins,* vol. 69, no. Suppl 8, pp. 38-56, 2007.

[41] V. Mariani, F. Kiefer, T. Schmidt, J. Hass and T. Schwede, "Assessment of Template Based Protein Structure Predictions in CASP9," *Proteins,* vol. 79, no. Suppl 10, pp. 37-58, 2011.

[42] Y. Zhang and J. Skolnick, "The protein structure prediction problem could be solved using the current PDB library," *Proceedings of the National Academy of Sciences of the United States of America,* vol. 102, no. 4, pp. 1029-1034, 2005.

[43] C. Chothia and A. M. Lesk, "The relation between the divergence of sequence and structure in proteins," *The EMBO Journal,* vol. 5, no. 4, pp. 823-826, 1986.

[44] W. J. Browne, A. C. North, D. C. Phillips, K. Brew, T. C. Vanaman and R. L. Hill, "A possible three-dimensional structure of bovine alpha-lactalbumin based on that of hen's egg-white lysozyme," *Journal of Molecular Biology,* vol. 42, pp. 65-86, 1969.

[45] S. Ovchinnikov, H. Park, N. Varghese, P.-S. Huang, G. Pavlopoulos, D. Kim, H. Kamisetty, N. Kyrpides and D. Baker, "Protein structure determination using metagenome sequence data," *Science,* vol. 355, no. 6322, pp. 294-298, 2017.

[46] M. J. Sippl and H. Flockner, "Threading thrills and threats," *Structure,* vol. 4, pp. 15-19, 1996.

[47] D. Fischer, D. Rice, J. U. Bowie and D. Eisenberg, "Assigning amino acid sequences to 3-dimensional protein folds," *The FASEB journal,* vol. 10, pp. 126-136, 1996.

[48] C. Zhang and C. Delisi, "Estimating the number of protein folds," *1998,* vol. 284, no. 5, pp. 1301-1305, Journal of Molecular Biology.

[49] Z. X. Wang, "A re-estimation for the total numbers of protein folds and superfamilies," *Protein Engineering,* vol. 11, no. 8, pp. 621-626, 1998.

[50] Z. X. Wang, "How many fold types of protein are there in nature?," *Proteins,* vol. 26, pp. 186-191, 1996.

[51] C. Chothia, "Proteins. One thousand families for the molecular biologist," *Nature,* vol. 357, pp. 543-544, 1992.

[52] S. Govindaanjan, R. Recabarren and D. Eisenberg, "Estimating the total number of protein folds," *Proteins,* vol. 35, pp. 408-414, 1999.

[53] C. T. Zhang, "Relations of the numbers of protein sequences, families and folds," *Protein Engineering,* vol. 10, pp. 757-761, 1997.

[54] J. Greer, "Comparative model-building of the mammalian serine proteases," *Journal of Molecular Biology,* vol. 153, pp. 1027-1042, 1981.

[55] "Insight II, Reference Guide, Version 1.1.0 and Reference Guide 2.0.0," Biosym Technologies, San Diego, CA, 1991.

[56] M. J. Sutcliffe, I. Haneef, D. Carney and T. L. Blundell, "Knowledge based modelling of homologous proteins, Part I: Three-dimensional frameworks derived from the simultaneous superposition of multiple structures," *Protein Engineering,* vol. 1, pp. 377-384, 1987.

[57] W. C. Ripka, "Computer-assisted model building," *Nature,* vol. 321, pp. 93-94, 1986.

[58] T. Blundell, S. Bedarkar, E. Rinderknecht and R. E. Humbel, "Insulin- like growth factor: a model for tertiary structure accounting for immunoreactivity and receptor binding," *Proceedings of the National Academy of Sciences of the United States of America,* vol. 75, pp. 180-184, 1978.

[59] T. Blundell, B. Sibanda and L. Pearl, "Three-dimensional structure, specificity and catalytic mechanism of renin," *Nature,* vol. 304, pp. 273-275, 1983.

[60] C. Chothia, A. M. Lesk, M. Levitt, A. G. Amit, R. A. Mariuzza, S. E. Phillips and R. J. Poljak, "The predicted structure of immunoglobulin D1.3 and its comparison with the crystal structure," *Science,* vol. 233, pp. 755-758, 1986.

[61] A. Sali and T. Blundell, "Comparative protein modelling by satisfaction of spatial restraints," *Journal of Molecular Biology,* vol. 234, no. 3, pp. 779-815, 1993.

[62] J. Wooley and Y. Ye, "A Historical Perspective and Overview of Protein," in *Computational methods for protein structure prediction and modeling*, New York, Springer, 2007, pp. 1-43.

[63] B. Webb and A. Sali, "Protein Structure Modeling with MODELLER," *In Functional Genomics,* pp. 39-54, 2017.

[64] M. C. Peitsch, "ProMod and Swiss-Model: Internet-based tools for automated comparative protein modelling," *Chemical Design Automation News,* vol. 11, pp. 13-14, 1996.

[65] D. Petrey, Z. Xiang, C. Tang, L. Xie, M. Gimpelev, T. Mitros, C. Soto, S. Goldsmith-Fischman, A. Kernytsky, A. Schlessinger, I. Koh, E. Alexov and B. Honig, "Using multiple structure alignments, fast model building, and energetic analysis in fold recognition and homology modeling," *Proteins ,* vol. 53, no. S6, pp. 430-435, 2003.

[66] J. Kosinski, I. Cymerman, M. Feder, M. Kurowski, J. Sasin and J. Bujnicki, "A “FRankenstein's monster” approach to comparative modeling: Merging the finest fragments of Fold-Recognition models and iterative model refinement aided by 3D structure evaluation," *Proteins,* vol. 53, no. S6, pp. 369-379, 2003.

[67] J. Moult, K. Fidelis, A. Kryshtafovych, T. Schwede and A. Tramontano, "Critical assessment of methods of protein structure prediction: Progress and new directions in round XI," *Proteins,* vol. 84, no. S1, pp. 4-14, 2016.

[68] C. Chen, J. Hawng and J. Yang, "(PS) 2 : protein structure prediction server," *Nucleic Acids Research,* vol. 34, no. S2, pp. W152-W157, 2006.

[69] C.-C. Chen, J.-K. Hwang and J.-M. Yang, "(PS)2-v2: template-based protein structure prediction server," *Bioinformatics,* vol. 366, no. 10, 2009.

[70] Y. Zhang, "I-TASSER server for protein 3D structure prediction," *Bioinformatics,* vol. 40, no. 9, 2008.

[71] Z. Chengxin, S. M. Mortuza, B. He, Y. Wang and Y. Zhang, "Template-based and free modeling of I-TASSER and QUARK pipelines using predicted contact maps in CASP12," *Proteins: Structure, Function, and Bioinformatics,* vol. 86, pp. 136-151, 2018.

[72] M. Remmert, A. Biegert, A. Hauser and J. Söding, "HHblits: lightning-fast iterative protein sequence searching by HMM-HMM alignment," *Nature Methods,* vol. 9, pp. 173-175, 2011.

[73] M. Källberg, H. Wang, S. Wang, J. Peng, Z. Wang, H. Lu and J. Xu, "Template-based protein structure modeling using the RaptorX web server," *Nature Protocols,* vol. 7, no. 8, pp. 1511-1522, 2012.

[74] Y. Gao, S. Wang, M. Deng and J. Xu, "RaptorX-Angle: real-value prediction of protein backbone dihedral angles through a hybrid method of clustering and deep learning.," *BMC bioinformatics,* vol. 19, no. 4, p. 100, 2018.

[75] L. J. McGuffin, "Protein Fold Recognition and Threading," in *Computational Structural Biology: Methods and Applications*, World Scientific, 2008, pp. 37-60.

[76] C. A. Orengo, D. T. Jones and J. M. Thornton, "Protein superfamilies and domain superfolds," *Nature,* vol. 372, pp. 631-634, 1994.

[77] A. G. Murzin, S. E. Brenner, T. Hubbard and C. Chothia, "SCOP: a structural classification of proteins database for the investigation of sequences and structures," *Journal of Molecular Biology,* vol. 247, pp. 536-540, 1995.

[78] S. Govindarajan and R. A. Goldstein, "Why are some proteins structures so common?," *Proceedings of the National Academy of Sciences of the United States of America,* vol. 93, pp. 3341-3345, 1996.

[79] C. A. Orengo, A. D. Michie, S. Jones, D. T. Jones, M. B. Swindells and J. M. Thornton, "CATH–a hierarchic classification of protein domain structures," *Structure,* vol. 5, pp. 1093-1108, 1997.

[80] W. R. Taylor and C. A. Orengo, "Protein structure alignment," *Journal of Molecular Biology,* vol. 208, pp. 1-22, 1989.

[81] C. M. Lemer , M. J. Rooman and S. J. Wodak, "Protein structure prediction by threading methods: evaluation of current techniques," *Proteins,* vol. 23, pp. 337-355, 1995.

[82] R. Thiele, R. Zimmer and T. Lengauer, "Protein threading by recursive dynamic programming," *Journal of Molecular Biology*, vol. 290, no. 3, pp. 757-779, 1999.

[83] D. T. Jones, "GenTHREADER: an efficient and reliable protein fold recog- nition method for genomic sequences," *Journal of Molecular Biology,* vol. 287, pp. 797-815, 1999.

[84] S. E. Altschul, W. Gish, W. Miller, E. Myers and D. J. Lipman, "Basic local alignment search tool," *Journal of Molecular Biology,* vol. 215, pp. 403-410, 1990.

[85] D. Fischer, "Hybrid Fold Recognition: Combining Sequence Derived Properties with Evolutionary Information," *Pacific Symposium on Biocomputing,* vol. 5, pp. 116-127, 2000.

[86] L. Kelley, R. MacCallum and M. Sternberg, "Enhanced genome annotation using structural profiles in the program 3D-PSSM," *Journal of Molecular Biology,* vol. 299, no. 2, pp. 501-522, 2000.

[87] S. F. Altschul, T. L. Madden, A. A. Schaffer, J. Zhang, Z. Zhang, W. Miller and D. J. Lipman, "Gapped BLAST and PSI-BLAST: a new generation of protein data base search programs," *Nucleic Acids Research,* vol. 25, no. 17, pp. 3389-3402, 1997.

[88] J. Shi, T. Blundell and K. Mizuguchi, "FUGUE: sequence-structure homology recognition using environment-specific substitution tables and structure-dependent gap penalties," *Journal of Molecular Biology,* vol. 310, pp. 243-257, 2001.

[89] L. J. McGuffin and D. T. Jones, "Improvement of the GenTHREADER method for genomic fold recognition," *Bioinformatics,* vol. 19, pp. 874-881, 2003.

[90] D. T. Jones, K. Bryson, A. Coleman, L. J. McGuffin, M. I. Sadowski, J. S. Sodhi and J. J. Ward, "Prediction of novel and analo- gous folds using fragment assembly and fold recognition," *Proteins,* vol. 61, no. S7, pp. 143-151, 2005.

[91] A. Lobley, M. I. Sadowski and D. T. Jones, "pGenTHREADER and pDomTHREADER: new methods for improved protein fold recognition and superfamily discrimination," *Bioinformatics,* vol. 25, no. 14, p. 1761–1767, 2009.

[92] D. W. Buchan and D. T. Jones, "EigenTHREADER: analogous protein fold recognition by efficient contact map threading," *Bioinformatics,* vol. btx217, 2017.

[93] Y. Zhang, A. K. Arakaki and J. Skolnick, "TASSER: An automated method for the prediction of protein tertiary structures in CASP6," *Proteins,* vol. 61, no. S7, pp. 91-98, 2005.

[94] Y. Zhang, "Template-based modeling and free modeling by I-TASSER in CASP7," *Proteins,* vol. 69, no. S8, pp. 108-117, 2007.

[95] Y. Zhang, "I-TASSER: Fully automated protein structure prediction in CASP8," *Proteins,* vol. 77, no. S9, pp. 100-113, 2009.

[96] D. Xu, J. Zhang, A. Roy and Y. Zhang, "Automated protein structure modeling in CASP9 by I-TASSER pipeline combined with QUARK-based ab initio folding and FG-MD-based structure refinement," *Proteins,* vol. 79, no. S10, pp. 147-160, 2011.

[97] Y. Zhang, "Interplay of I-TASSER and QUARK for template-based and ab initio protein structure prediction in CASP10," *Protein,* vol. 82, no. S2, pp. 175-187, 2013.

[98] J. Yang, W. Zhang, B. He, S. E. Walker, H. Zhang, B. Govindarajoo, J. Virtanen, Z. Xue, H.-B. Shen and Y. Zhang, "Template-based protein structure prediction in CASP11 and retrospect of I-TASSER in the last decade," *Proteins,* vol. 84, no. S1, pp. 233-246, 2015.

[99] J. Yang, R. Yan, A. Roy, D. Xu, J. Poisson and Y. Zhang, "The I-TASSER Suite: protein structure and function prediction," *Nature Methods,* vol. 12, pp. 7-8, 2015.

[100] A. Roy, A. Kucukural and Y. Zhang, "I-TASSER: a unified platform for automated protein structure and function prediction," *Natural Protocols,* vol. 5, no. 4, pp. 725-738, 2010.

[101] J. Kosinski, K. L. Tkaczuk, J. M. Kasprzak and J. M. Bujnicki, "Template Based Prediction of Three-dimensional Protein Structures: Fold Recognition and Comparative Modeling," in *Prediction of Protein Structures, Functions, and Interactions*, West Sussex, Wiley, 2009, pp. 87-116.

[102] A. Heger and L. Holm, "Picasso: generating a covering set of protein family profiles," *Bioinformatics,* vol. 20, no. 4, pp. 272-279, 2001.

[103] D. Mittelman, R. Sadreyev and N. Grishin, "Probabilistic scoring measures for profi le - profile comparison yield more accurate short seed alignments," *Bioinformatics,* vol. 19, no. 12, pp. 1531-1539, 2003.

[104] B. Contreras-Moreira, P. W. Fitzjohn and P. A. Bates, "In silico proteinrecombination: enhancing template and sequence alignment selection for comparative protein modelling," *Journal of Molecular Biology,* vol. 328, no. 3, pp. 593-608, 2003.

[105] M. I. Sadowski and D. T. Jones, "Benchmarking template selection andmodel qualityassessment for high-resolution comparative modeling," *Proteins,* vol. 69, no. 3, pp. 476-485, 2007.

[106] J. Soding, A. Biegert and A. N. Lupas, "The HHpred interactive server for protein homology detection and structure prediction," *Nucleic acids research,* vol. 33, no. S2, pp. W244-W248, 2005.

[107] L. Zimmermann, A. Stephens, S.-Z. Nam, D. Rau, J. Kübler, M. Lozajic, F. Gabler, J. Söding, A. Lupas and V. Alva, "A Completely Reimplemented MPI Bioinformatics Toolkit with a New HHpred Server at its Core," *Journal of molecular biology,* 2017.

[108] M. A. Martí-Renom, A. C. Stuart, A. Fiser, R. Sánchez, F. Melo and A. Šali, "Comparative Protein Structure Modeling of Genes and Genomes," *Annual Review of Biophysics and Biomolecular Structure,* vol. 29, pp. 291-325, 2000.

[109] B. Wallner and A. Elofsson, "Quality Assessment of Protein Models," in *Prediction of protein structures, functions, and interactions*, Chichester, Wiley, 2009, pp. 143-157.

[110] J. Xiong, Essential bioinformatics, Cambridge University Press, 2006.

[111] E. F. Pettersen, T. D. Goddard, C. C. Huang, G. S. Couch, D. M. Greenblatt, E. C. Meng and T. Ferrin, "UCSF Chimera--a visualization system for exploratory research and analysis," *Journal of computational chemistry,* vol. 25, no. 13, pp. 1605-1612, 2004.

[112] B. Rost, "Twilight zone of protein sequence alignments," *Protein engineering design and selection,* vol. 12, no. 2, pp. 85-94, 1999.

[113] W. M. Fitch, "Locating gaps in amino acid sequences to optimize the homology between two proteins," *Biochemical genetics,* vol. 3, no. 2, pp. 99-108, 1969.

[114] A. J. Gibbs and G. A. Mcintyre, "The Diagram, a Method for Comparing Sequences," *European Journal of Biochemistry ,* vol. 16, pp. 1-11, 1970.

[115] A. D. McLachlan, "Tests for comparing related amino-acid sequences. Cytochrome c and cytochrome c551," *Journal of Molecular Biology,* vol. 61, no. 2, pp. 409-424, 1971.

[116] A. D. McLachlan, "Analysis of gene duplication repeats in the myosin rod," *Journal of Molecular Biology,* vol. 169, pp. 15-30, 1983.

[117] A. D. McLachlan and D. R. Boswell, "Confidence limits for homology in protein or gene sequences: The c-myc oncogene and adenovirus E1a protein," *Journal of Molecular Biology,* vol. 185, pp. 39-49, 1985.

[118] J. G. Reich and W. Meiske, "A simple statistical significance test of window scores in large dot matrices obtained from protein or nucleic acid sequences," *Bioinformatics,* vol. 3, pp. 25-30, 1987.

[119] W. R. Pearson and D. J. Lipman, "Improved tools for biological sequence comparison," *Proceedings of the National Academy of Sciences,* vol. 85, no. 8, pp. 2444-2448, 1988.

[120] S. Schwartz, W. Miller, C. M. Yang and R. C. Hardison, "Software tools for analyzing pairwise alignments of long sequences," *Nucleic Acids Research,* vol. 19, no. 17, pp. 4663-4667, 1991.

[121] C. Lefévre and J.-E. Ikeda, "A fast word search algorithm for the representation of sequence similarity in genomic DNA," *Nucliec Acids Research,* vol. 22, no. 3, pp. 404-411, 1994.

[122] J. V. Maizel and R. P. Lenk, "Enhanced graphic matrix analysis of nucleic acid and protein sequences," *Proceedings of the National Academy,* vol. 78, no. 12, pp. 7665-7669, 1981.

[123] A. H. Reisner and C. A. Bucholtz, "The use of various properties of amino acids in color and monochrome dot-matrix analyses for protein homologies," *Bioinformatics,* vol. 4, no. 3, pp. 395-402, 1988.

[124] M. Zuker, "Suboptimal sequence alignment in molecular biology: Alignment with error analysis," *Journal of Molecular Biology,* vol. 221, no. 2, pp. 403-420, 1991.

[125] P. Argos, "A sensitive procedure to compare amino acid sequences," *Journal of Molecular Biology,* vol. 193, no. 2, pp. 385-396, 1987.

[126] E. L. Sonnhammer and R. Durbin, "A dot-matrix program with dynamic threshold control suited for genomic," *Gene,* vol. 167, pp. GC1-GC10, 1995.

[127] C. Mayor, M. Brudno, J. R. Schwartz, A. Poliakov, E. M. Rubin, K. A. Frazer, L. S. Pachter and I. Dubchak, "VISTA : visualizing global DNA sequence alignments of arbitrary length," *Bioinformatics,* vol. 16, no. 11, p. 1046–1047, 2000.

[128] Y. Huang and L. Zhang, "Rapid and sensitive dot-matrix methods for genome analysis," *Bioinformatics,* vol. 20, no. 4, pp. 460-466, 2004.

[129] Y. Ohtsubo, W. Ikeda-Ohtsubo, Y. Nagata and M. Tsuda, "GenomeMatcher: A graphical user interface for DNA sequence comparison," *BMC Bioinformatics,* vol. 9, p. 376, 2008.

[130] K. Katoh and D. Standley, "MAFFT Multiple Sequence Alignment Software Version 7: Improvements in Performance and Usability," *Molecular Biology and Evolution,* vol. 30, no. 4, pp. 772-780, 2013.

[131] S. Kurtz, A. Phillippy, A. Delcher, M. Smoot, M. Shumway, C. Antonescu and S. Salzberg, "Versatile and open software for comparing large genomes," *Genome Biology,* vol. 5, no. 2, p. R12, 2004.

[132] S. B. Needleman and C. D. Wunsch, "A General Method Applicable to the Search for Similarities in the Amino Acid Sequence of Two Proteins," *Journal of Molecular Biology,* vol. 48, pp. 443-453, 1970.

[133] D. Sankoff and J. B. Kruskal, Time warps, string edits, and macromolecules: the theory and practice of sequence comparison., D. Sankoff and J. B. Kruskal, Eds., Reading: Addison-Wesley Publication, 1983.

[134] M. O. Dayhoff, Atlas of Protein Sequence and Structure, National Biomedical Research Foundation, 1972.

[135] D. Sankoff, "Matching Sequences under Deletion/Insertion Constraints," *Proceedings of the National Academy of Sciences of the United Stated of America,* vol. 69, pp. 4-6, 1972.

[136] P. H. Sellers, "On the Theory and Computation of Evolutionary Distances," *Journal of Applied Mathematics* , vol. 26, no. 4, pp. 787-793, 1974.

[137] M. S. Waterman, T. F. Smith and W. A. Beyer, "Some biological sequence metrics," *Advances in Mathematics,* vol. 20, no. 3, pp. 367-387, 1976.

[138] M. S. Waterman and T. F. Smith, "Identification of common molecular subsequences," *Journal of Molecular Biology,* vol. 147, pp. 195-197, 1981.

[139] O. Gotoh, "An Improved Algorithm for Matching Biological Sequences," *Journal of Molecular Biology,* vol. 162, pp. 705-708, 1982.

[140] P. Taylor, "A Fast Homology for Aligning Biological Sequences," *Nucleic Acids Research,* vol. 12, pp. 447-455, 1984.

[141] S. F. Altschul and B. W. Erickson, "Optimal Sequence Alignment Using Affine Gap Costs," *Bulletin of Mathematical Biology,* vol. 5, no. 6, pp. 603-616, 1986.

[142] E. W. Myers and W. Miller, "Optimal alignments in linear space," *bioinformatics,* vol. 4, pp. 11-17, 1988.

[143] X. Huang and K. M. Chao, "A generalized global alignment algorithm," *Bioinformatics,* vol. 19, no. 2, pp. 228-233, 2003.

[144] A. Chakraborty and S. Bandyopadhyay, "FOGSAA: Fast Optimal Global Sequence Alignment Algorithm," *Scientific reports,* vol. 3, p. 1746, 2013.

[145] W. Ye, Y. Chen, Y. Zhang and Y. Xu, "H-BLAST: a fast protein sequence alignment toolkit on heterogeneous computers with GPUs.," *Bioinformatics,* vol. 33, no. 8, pp. 1130-1138, 2017.

[146] B. Balech, A. Monaco, M. Perniola, M. Santamaria, G. Donvito, S. Vicario, G. Maggi and G. Pesole, "DNA Multiple Sequence Alignment Guided by Protein Domains: The MSA-PAD 2.0 Method.," *In Viral Metagenomics,* pp. 173-180, 2018.

[147] M. Šošić and M. Šikić, "Edlib: a C/C++ library for fast, exact sequence alignment using edit distance," *Bioinformatics,* vol. 33, no. 9, pp. 1394-1395, 2017.

[148] M. DM., Bioinformatics: Sequence and Genome Analysis, Cold Spring Harbor, NY: Cold Spring Harbor Laboratory Press, 2004.

[149] D. Lipman and W. Pearson, "Rapid and Sensitive Protein Similarity Searches," *Science,* vol. 227, no. 4693, pp. 1435-1441, 1985.

[150] K. Deb, Multi-Objective Optimization using Evolutionary Algorithms, Chichester, West Sussex, England: John Wiley & Sons, LTD, 2001.

[151] A. K. Nandi and K. Deb, "Multi-objective evolutionary algorithms: Application in Designing Particle Reinforced Mould Materials," in *Material Science and Engineering: Concept, Methodologies, Tools, and Applications*, Hershey PA, USA, IGI Global, 2017, pp. 185-229.

[152] M. Gerstein and M. Levitt, "Using Iterative Dynamic Programming to Obtain Accurate Pairwise and Multiple Alignments of Protein Structures," in *ISMB-96*, 1996.

[153] S. F. Altschul, "Generalized Affine Gap Costs for Protein Sequence Alignment," *PROTEINS: Structure, Function, and Genetics,* vol. 32, pp. 88-96, 1998.

[154] Y. Yang, E. Faraggi, H. Zhao and Y. Zhou, "Improving protein fold recognition and template-based modeling by employing probabilistic-based matching between predicted one-dimensional structural properties of query and corresponding native properties of templates," *Bioinformatics,* vol. 27, no. 15, pp. 2076-2082, 2011.

[155] C. Notredame and D. G. Higgins, "SAGA: sequence alignment by genetic algorithm," *Nucleic Acids Research,* vol. 24, no. 8, pp. 1515-1524, 1996.

[156] A. Taneda, "Multi-objective pairwise RNA sequence alignment," *Bioinformatics,* vol. 26, no. 19, pp. 2383-2390, 2010.

[157] M. O. Dayhoff, R. M. Schwartz and B. C. Orcutt, "In Atlas of Protein Sequence and Structure," *National Biomedical Research Foundation,* vol. 5, pp. 345-358, 1978.

[158] S. Henikoff and J. Henikoff, "Amino Acid Substitution matrices from Protein blocks," *Proceedings of the National Academy of Science of the United States of America,* vol. 89, pp. 10915-10919, 1992.

[159] M. Gribskov, A. D. Mclachlan and D. Eisenberg, "Profile Analysis: Detection of Distantly related Proteins," *Proceedings of the National Academy of Sciences of the United States of America,* vol. 84, no. 13, pp. 4355-4358, 1987.

[160] S. R. Eddy, "Profile Hidden Markov Models," *Bioinformatics,* pp. 755-763, 1998.

[161] K. Karplus, C. Barrett and R. Hughey, "Hidden Markov Models for Detecting Remote Protein Homologies," *Bioinformatics,* vol. 14, pp. 846-856, 1998.

[162] M. S. Johnson and J. Overington, "A Structural Basis for Sequence Comparisons:: An Evaluation of Scoring Methodologies," *Molecular Biology,* vol. 233, no. 4, pp. 716-738, 1993.

[163] A. Prlic, F. Domingues and M. Sippl, "Structure-derived substitution matrices for alignment of distantly related sequences," *Protein Engineering Design and Selection,* vol. 13, no. 8, pp. 545-550, 2000.

[164] Q. Le, F. Sievers and D. G. Higgins, "Protein multiple sequence alignment benchmarking through secondary structure prediction," *Bioinformatics,* vol. 33, no. 9, pp. 1331-1337, 2017.

[165] D. Jones, "Protein secondary Structure Prediction Based on Position Specific Scoring Matrices," *Journal of Molecular Biology,* vol. 292, pp. 195-202, 1999.

[166] A. Yassen and Y. Li, "CASA: A Protein Solvent Accessibility Prediction Server using Context-based Features to Enhance Prediction Accuracy," *BMC Bioinformatics,* 2014.

[167] T. P. Hopp and K. R. Woods, "Prediction of protein antigenic determinants from amino acid sequences," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 78, no. 6, pp. 3824-3828, 1981.

[168] J. Kyte and R. Doolittle, "A Simple Method for Displaying the Hydropathic Character of a Protein," *Journal of Molecular Biology,* vol. 157, pp. 105-132, 1982.

[169] J. López-Blanco, A. Canosa-Valls, Y. Li and P. Chacón, "RCD+: Fast Loop Modeling Server," *Nucleic Acids Research,* vol. 44, no. W1, pp. W395-W4000, 2016.

[170] A. Yaseen and Y. Li, "Context-based features enhance protein secondary structure prediction accuracy," *Journal of chemical information and modeling,* vol. 54, no. 3, pp. 992-1002, 2014.

[171] W. Elhefnawy, M. Li, J. Wang and Y. Li, "Construction of Protein Backbone Fragments Libraries on Large Protein Sets Using a Randomized Spectral Clustering Algorithm," in *Bioinformatics Research and Applications. ISBRA 2017. Lecture Notes in Computer Science*, Springer, 2017, pp. 108-119.

[172] J. Skolnick and D. Kihara, "Defrosting the frozen approximation: PROSPECTOR--a new approach to threading," *Proteins,* vol. 42, no. 3, pp. 319-331, 2001.

[173] J. Skolnick, D. Kihara and Y. Zhang, "Development and large scale benchmark testing of the PROSPECTOR_3 threading algorithm," *Proteins,* vol. 56, no. 3, pp. 502-518, 2004.

[174] H. Zhou and J. Skolnick, "Ab Initio Protein Structure Prediction Using Chunk-TASSER," *Biophysical,* vol. 93, no. 5, pp. 1510-1518, 2007.

[175] K. Wolfgang and C. Sander, "Dictionary of protein secondary structure: pattern recognition of hydrogen-bonded and geometrical features," *Biopolymers,* vol. 22, no. 12, pp. 2577-2637, 1983.

[176] S. Ahmad, M. M. Gromiha and A. Sarai, "Real value prediction of solvent accessibility from amino acid sequence," *Proteins: Structure, Function, and Bioinformatics,* vol. 50, no. 4, pp. 629-635, 2003.

[177] I. Van Walle, I. Lasters and L. Wyns, "SABmark—a benchmark for sequence alignment that covers the entire known fold space," *Bioinformatics,* vol. 21, no. 7, pp. 1267-1268, 2004.

[178] Y. Zhang and J. Skolnick, "Scoring function for automated assessment of protein structure template quality," *Proteins: Structure, Function, and Bioinformatics,* vol. 57, no. 4, pp. 702-710, 2004.

[179] M. Abdelrasoul and Y. Li, "Exploring Multi-Objective with Protein Sequence Alignment," in *International Conference on Bioinformatics and Computational Biology (BICOB),* Las Vegas, NV, USA, 2018.

[180] L. Zhong and W. C. Johnson, "Environment Affects Amino Acid Preference for Secondary Structure," *Proceedings of the National Academy of Sciences,* vol. 89, no. 10, pp. 4462-4465, 1992.

[181] R. Adamczak, A. Porollo and J. Meller, "Combining prediction of secondary structure and solvent accessibility in proteins," *Proteins: Structure, Function, and Bioinformatics,* vol. 59, no. 3, pp. 467-475, 2005.

[182] J. R. Macdonald and W. C. Johnson JR, "Environmental features are important in determining protein secondary structure," *Protein Science,* vol. 10, no. 6, pp. 1172-1177, 2001.

[183] N. Goldman, T. L. Jeffery and D. T. Jones, "Assessing the impact of secondary structure and solvent accessibility on protein evolution," *Genetics,* vol. 149, no. 1, pp. 445-458, 1998.

[184] Š. Andrej and T. L. Blundell, "Comparative protein modelling by satisfaction of spatial restraints.," *J. Mol. Biol.,* vol. 234, pp. 779-815, 1993.

[185] S. Altschul , W. Gish, W. Miller, E. Myers and D. Lipman, "A basic Local Alignment Search Tool," *Journal of Molecular Biology,* vol. 215, pp. 403-410, 1990.

[186] D. States, W. Gish and S. Altschul, "Improved Sensitivity of Nucleic Acid Database Searches Using Application-Specific Scoring Matrices," *METHODS: A Companion to Methods in Enzymology.,* vol. 3, pp. 66-70, 1991.

[187] M. Kimura, "A Simple Method for Estimating Evolutionary Rates of Base Substitutions Through Comparative Studies of Nucleotide Sequences," *Journal of Molecular Evolution,* vol. 16, no. 2, pp. 111-120, 1980.

[188] K. Deb, Multi-objective optimization using evolutionary algorithms, John Wiley & Sons, 2001.

[189] Y. Li, "MOMCMC: An Efficient Monte Carlo Method for Multi-Objective Sampling over Real Parameter Space," *Computers and Mathematics with Applications,* vol. 64, pp. 3542-3556, 2012.

[190] W. Zhu, A. Yaseen and Y. Li, "DEMCMC-GPU: An Efficient Multi-Objective Optimization Method with GPU Acceleration on the Fermi Architecture," *New Generation Computing,* vol. 29, no. 2, pp. 163-184, 2011.

[191] K. W. DeRonne and G. Karypis, "Pareto Optimal Pairwise Sequence Alignment," *IEEE/ACM Transactions on Computational Biology and Bioinformatics,* vol. 10, no. 2, pp. 481-493, 2013.

[192] A. Auger, J. Bader, D. Brockhoff and E. Zitzler, "Hypervolume-based multiobjective optimization: theoretical foundations and practical implications," *Theoretical Computer Science,* vol. 425, p. 75–103, 2012.

VITA

Maha Mahmoud Abdelaal Abdelrasoul

Department of Computer Science

Old Dominion University

Norfolk, VA 23529

Maha received her Bachelor and Master degrees in Computer Engineering from the Arab Academy for Science and Technology, Cairo, Egypt, in 2006 and 2011, respectively. In Fall 2014, she joined the Computer Science Department of Old Dominion University and started her research in computational biology and machine learning. Maha's research objectives are directed toward studying and implementing novel computational biology and machine learning algorithms to accommodate biological and chemical experiments on proteins. She developed several computational methods for a set of fundamental and universal bioinformatics challenges, such as identifying conformational clusters of phosphorylated tyrosine, developing a multi-objective alignment algorithm to align biological sequences, and designing and implementing a template selection approach for protein template-based modeling.