Universidad Pablo de
Olavide de Sevilla

New Evolutionary Approaches to Protein Structure Prediction

MEMORIA QUE PRESENTA
Alfonso Eduardo Márquez Chamorro

PARA OPTAR AL GRADO DE DOCTOR POR LA
UNIVERSIDAD PABLO DE OLAVIDE DE SEVILLA

DIRECTORES
Jesús S. Aguilar Ruiz
Federico Divina

Área de Lenguajes y Sistemas Informáticos
Escuela Politécnica Superior

Marzo de 2013

Universidad Pablo de
Olavide de Sevilla

Área de Lenguajes y Sistemas Informáticos
Escuela Politécnica Superior

New Evolutionary Approaches to Protein Structure Prediction

Tesis Doctoral

Alfonso Eduardo Márquez Chamorro

Sevilla, Marzo de 2013

D. Jesús S. Aguilar Ruiz y D. Federico Divina, profesores Titulares de Universidad adscritos al Área de Lenguajes y Sistemas Informáticos de la Universidad Pablo de Olavide de Sevilla,

CERTIFICAN QUE:

D. Alfonso E. Márquez Chamorro, Ingeniero Informático por la Universidad Autónoma de Madrid, ha realizado bajo su supervisión el trabajo de investigación titulado:

NEW EVOLUTIONARY APPROACHES TO PROTEIN STRUCTURE PREDICTION

Una vez revisado, autorizan la presentación del mismo como tesis doctoral en la Universidad Pablo de Olavide de Sevilla y estiman oportuna su presentación al tribunal que habrá de valorarlo. Dicha tesis ha sido realizada dentro del programa de doctorado *Biotecnología y Tecnología Química,* de la Universidad Pablo de Olavide de Sevilla.

Sevilla, Marzo de 2013.

D. Jesús S. Aguilar Ruiz y D. Federico Divina, profesores adscritos al área de Lenguajes y Sistemas Informáticos de la Universidad Pablo de Olavide de Sevilla, como directores de la tesis titulada

NEW EVOLUTIONARY APPROACHES TO PROTEIN STRUCTURE PREDICTION,

proponen la siguiente composición del tribunal titular, a fin de que la Comisión de Doctorado designe al tribunal encargado de juzgar la tesis doctoral.

PRESIDENTE:	Dr. D. José C. Riquelme Santos
VOCAL:	Dra. Dña. Cristina Rubio Escudero
SECRETARIO:	Dr. D. Domingo S. Rodriguez Baena
SUPLENTES:	Dra. Dña. Alicia Troncoso Lora
	Dr. D. Raúl Giráldez Rojo
	Dr. D. Roberto Ruiz Sánchez

A mis padres

Agradecimientos

Esta tesis no habría sido posible sin la guía y consejo de varias personas que de una u otra manera han contribuido con su valiosa ayuda en la preparación y realización de este trabajo.

A mis directores de tesis, Jesús y Federico. A Jesús por darme la oportunidad de entrar en su grupo de investigación, por su brillante trayectoria y por la genialidad de sus ideas y aportaciones. A Federico, por su cercanía personal y por su valía profesional.

A todos los compañeros del grupo de investigación de Minería de datos y Bioinformática, que de una forma u otra han aportado su granito de arena en estos cuatro años de trabajo. En especial, me gustaría dar las gracias a Gualberto, como amigo y compañero de fatigas, sus ideas y comentarios fueron de gran valor a lo largo de todo este tiempo.

A mi familia, por ser el soporte de mi vida. A mis padres por su entrega, amor, paciencia, trabajo y sacrificio en todos estos años. A mis tíos y padrinos, por sus atenciones, motivaciones y consejos. A mi primo y colega Juan Antonio, por su entusiasmo motivador y ejercer de primo y hermano mayor. A mi prima Isabel por su calidez personal, su dedicación e interés y a mi primo Manolo por su cercanía. A mis tías y querida abuela y al resto de familiares y amigos, gracias por estar siempre ahí.

Abstract

The problem of Protein Structure Prediction (PSP) is one of the principal topics in Bioinformatics. Multiple approaches have been developed in order to predict the protein structure of a protein. Determining the three dimensional structure of proteins is necessary to understand the functions of molecular protein level. An useful, and commonly used, representation for protein 3D structure is the protein contact map, which represents binary proximities (contact or non-contact) between each pair of amino acids of a protein. This thesis work, includes a compilation of the soft computing techniques for the protein structure prediction problem (secondary and tertiary structures). A novel evolutionary secondary structure predictor is also widely described in this work. Results obtained confirm the validity of our proposal. Furthermore, we also propose a multi-objective evolutionary approach for contact map prediction based on physico-chemical properties of amino acids. The evolutionary algorithm produces a set of decision rules that identifies contacts between amino acids. The rules obtained by the algorithm impose a set of conditions based on amino acid properties in order to predict contacts. Results obtained by our approach on four different protein data sets are also presented. Finally, a statistical study was performed to extract valid conclusions from the set of prediction rules generated by our algorithm.

Contents

I Introduction **15**

1 Introduction **17**
1.1 Motivation . . . 18
1.2 Objectives . . . 19
1.3 Overview . . . 19
1.4 Contribution . . . 20
1.5 Summary . . . 23

2 Biological background **25**
2.1 Proteins . . . 25
2.2 Protein structure . . . 26
2.2.1 Primary structure . . . 27
2.2.2 Secondary structure . . . 28
2.2.3 Tertiary structure . . . 29
2.2.4 Quaternary structure . . . 29
2.3 Protein synthesis . . . 30
2.4 Protein Folding . . . 31
2.5 Classification of Protein Structures . . . 32
2.5.1 SCOP . . . 32
2.5.2 CATH . . . 33
2.6 Protein Structure Prediction Problem . . . 33
2.7 The PDB Protein Structure Data Archive . . . 36
2.8 Contact maps . . . 37
2.8.1 Binary contact map . . . 37
2.8.2 Distance matrix . . . 39
2.8.3 Fuzzy contact map . . . 39
2.9 Summary . . . 40

3 Evolutionary Computation **41**
3.1 Introduction . . . 41
3.2 Components of EC . . . 43
3.2.1 Encoding . . . 43
3.2.2 Evaluation . . . 44

3.2.3 Selection 44
3.2.4 Crossover 45
3.2.5 Mutation 46
3.3 Multi-objective Optimization 47
3.4 Multi-objective Evolutionary Algorithms 48
3.5 Summary 50

II State of the art **51**

4 State of the art **53**
4.1 Secondary structure prediction methods 53
4.1.1 Statistical approaches 54
4.1.2 Neural networks methods 59
4.1.3 Support vector machines methods 63
4.1.4 Nearest neighbors-based methods 64
4.1.5 Hybrid methods 67
4.2 Tertiary structure prediction methods 70
4.2.1 Statistical approaches 71
4.2.2 Neural networks methods 72
4.2.3 Support vector machines methods 76
4.2.4 Evolutionary algorithm methods 78
4.2.5 Case-based reasoning methods 83
4.2.6 Other predictive methods 84
4.3 Summary 87

III Proposals **89**

5 Evolutionary approaches for the protein structure prediction **91**
5.1 Multi-objective Evolutionary Contact Map Predictor (MECoMaP) 91
5.1.1 Methodology 91
5.1.2 Physico-chemical properties of the amino acids 92
5.1.3 Structural features of protein residues 93
5.1.4 Data preparation 95
5.1.5 Encoding 98
5.1.6 Fitness Function 100
5.1.7 Genetic Operators 101
5.1.8 Algorithm 103
5.1.9 Efficient evaluation of the individuals 104
5.1.10 MECoMaP application 105

5.2 Protein Secondary Structure Predictor based on Evolutionary Computation . . . 107
5.2.1 Methodology . . . 108
5.2.2 Encoding . . . 109
5.2.3 Fitness Function . . . 109
5.2.4 Genetic Operators . . . 110
5.2.5 Algorithm . . . 110
5.3 Summary and conclusions . . . 111

IV Results **113**

6 Tertiary structure prediction experiments **115**
6.1 MECoMaP results . . . 115
6.1.1 Data sets . . . 115
6.1.2 Preliminary studies . . . 116
6.1.3 First experimentation . . . 120
6.1.4 Second experimentation . . . 121
6.1.5 Third experimentation . . . 121
6.1.6 Fourth experimentation . . . 122
6.2 Analysis of MECoMaP predicting rules . . . 125
6.3 Summary and conclusions . . . 127

7 Secondary structure prediction experiments **131**
7.1 Preliminary statistical analysis . . . 131
7.2 Experiments and discussion . . . 134
7.2.1 Data sets . . . 134
7.2.2 Alpha experimentation . . . 135
7.2.3 Beta experimentation . . . 137
7.3 Summary and conclusions . . . 138

V Conclusions **141**

8 Conclusions and Future works **143**

VI Appendix **147**

A Analysis of Infobiotics predicting rules **149**

B Tables of experiments **159**

C List of proteins **161**

D Glossary **167**

E Acronyms **169**

VII Bibliography **171**

List of Figures

2.1 Amino acid. Chemical composition. 27
2.2 A peptide unit. 28
2.3 Four levels of protein structure. 30
2.4 Quaternary structure of the Hemoglobin 31
2.5 The number of known protein sequences (Swissprot) versus the number of known structures (PDB). 34
2.6 Example of a binary contact map. 38
2.7 Example of a binary contact map with the graphical representation of secondary structure elements. 39
2.8 Example of a distance map for 1E79I protein. 40

3.1 One-point crossover. 45
3.2 Two-point crossover. 46
3.3 Uniform crossover. 46

4.1 GOR method representation. 55
4.2 Neural network schema detailed in [Qian *et al.*, 1988]. 60

5.1 Experimental procedure scheme. 92
5.2 Preprocessing procedure scheme. 98
5.3 Example of a complete individual. 99
5.4 Example of encoding for the element Q_i of an individual $R_{i,j}$. H_1, H_2, P_1 and P_2 are lower and upper bounds for the hydrophobicity and polarity and volume values. C represents the charge value of the residue. SS represents secondary structure and SA indicates the solvent accessibility value. . . 100
5.5 Example of a decision rule. 100
5.6 An example of one-point crossover for the element Q_i of two parent individuals $Par.1$ and $Par.2$ and the offspring individual $Off.1$. The random cut is established between H_2 and P_1. 101
5.7 Example of Gaussian mutation for the element Q_i of an individual $R_{i,j}$ with an increment value of $+0.2$ for the H_2 property. 102

5.8 Example of enlarge mutation operator for the element Q_i of an individual $R_{i,j}$ for P_1 and P_2 properties. 102
5.9 Example of AVL tree. Each leaf node represents a list with the training examples that fulfill the conditions imposed by its predecessor nodes, in this case they are referred to the hydrophobicity (H) of amino acid i and to the polarity (P) of amino acid j. 105
5.10 Screenshot of MECoMaP application. 106
5.11 Example of a generated distance map for 1A7GE protein by MECoMaP application. Distances in angstroms. 107
5.12 Relevant positions in an α-helix. 108
5.13 Experimental and prediction procedure. 108
5.14 Example of encoded chromosome for a beginning of an α-helix or a β-strand. 109

6.1 Sequence separation vs. number of contacts. 116
6.2 Pareto fronts for an execution in different generations. 119
6.3 Generated contact map of protein 5PTI. 124
6.4 Relative frequency of hydrophobicity values for amino acid i in our predicted rules. 125
6.5 Relative frequency of polarity values for amino acid i in our predicted rules. 126
6.6 Relative frequency of solvent accessibility values for amino acid i in our predicted rules. 127
6.7 Relative frequency of secondary structures for amino acid i in our predicted rules. We consider H for α-helix, E for β-sheets and C for coil. 127
6.8 Hydrophobicity regions for amino acids i and j covered by our predicted rules. 128
6.9 Polarity regions for amino acids i and j covered by our predicted rules. 128
6.10 Relative frequency (%) of charge values for amino acids i and j in our predicted rules. 129
6.11 An example of the best two rules obtained by MECoMaP on DS1. 129

7.1 Propensity matrix for N-cap,N_1 helix positions. 132
7.2 Propensity matrix for C_1, C-cap helix positions. 133
7.3 Propensity matrix for N-cap, N_1 strand positions. 133
7.4 Propensity matrix for C_1-Ccap strand positions. 134
7.5 Maximum Fitness vs. Average Fitness. 138

A.1 An example of rule obtained by Infobiotics. 150

A.2 Relative frequency of SA values for amino acid $r1$ in BioHEL rules. 152
A.3 Relative frequency of SS values for amino acid $r1$ in BioHEL rules. 154
A.4 Relative frequency of CN values for amino acid $r1$ in BioHEL rules. 154
A.5 Relative frequency of propensity values for amino acid $r1$ in BioHEL rules. 155
A.6 Relative frequency of RCH values for amino acid $r1$ in BioHEL rules. 155
A.7 Relative frequency of separation between residues for amino acid $r1$ in BioHEL rules. 156
A.8 Relative frequency of PSSM of amino acid D at position $r1$ in BioHEL rules. 156

List of Tables

2.1 A selective list of functional roles for proteins within cells. . . 26

4.1 Resume of statistical methods for secondary structure prediction. 58
4.2 Resume of neural networks methods for secondary structure prediction. 62
4.3 Resume of SVM methods for secondary structure prediction. 66
4.4 Resume of NN approaches for secondary structure prediction. 66
4.5 Resume of hybrid methods for secondary structure prediction. 69
4.6 Resume of neural network methods for tertiary structure prediction. 75
4.7 Resume of statistical methods for tertiary structure prediction. 77
4.8 Resume of SVM methods for tertiary structure prediction. . 77
4.9 Resume of evolutionary computation methods for tertiary structure prediction. 82
4.10 Resume of CBR methods for tertiary structure prediction. . 86
4.11 Resume of other methods for tertiary structure prediction. . 86
4.12 Resume of latest CASP competitions. 87

5.1 Values of different properties according to the cited scales for each amino acid. H represents the hydrophobicity, P the polarity and C the charge. 94
5.2 Parameter setting used in the experiments. 103

6.1 Number of total, positive (class=1) and negative (class=0) examples in the four data sets for the WEKA experimentation. 117
6.2 Average accuracy, coverage and standard deviation values obtained for different Weka classification algorithms for the DS1, DS2, DS3 and DS4 protein data sets with the same experimental settings. 118
6.3 Efficiency of our method predicting DS1 protein data set. . . 121
6.4 Efficiency of our method predicting DS2 protein data set. . . 122
6.5 Efficiency of our method predicting DS3 protein data set. . . 122

6.6 Efficiency of Ib1 WEKA classifier predicting DS1 protein dataset with different sets of attributes. 123
6.7 Efficiency of our method predicting DS1 protein data set including PSSM attributtes. 124

7.1 Average results for the prediction of the beginning of α-helices obtained for different number of iterations. Standard deviation is reported between brackets. 136
7.2 Average results for the prediction of the end of α-helices obtained for different number of iterations. Standard deviation is reported between brackets. 136
7.3 Average results for the prediction of the beginning of β-strands obtained for different number of iterations. Standard deviation is reported between brackets. 137
7.4 Average results for the prediction of the end of β-strands obtained for different number of iterations. Standard deviation is reported between brackets. 138

A.1 Top 20 most frequent attributes used in BioHEL's rules, where Ratio is the percentage of rules where the attribute appears. 151
A.2 Rank of the evolutionary information attributes aggregated by their AA type. 152
A.3 Top 20 most frequent pairs of attributes used in BioHEL's rules, where Ratio is the percentage of rules containing the attributes. 153
A.4 Average and best rank of the information sources in BioHEL's rules grouped by amino acid window and sorted by average rank. 157
A.5 Definition of the attributes in the BioHEL's rules, where $r1$ = first residue in the pair of residues that are tested whether they are in contact or not, $r2$ = second residue in the pair and central = middle point in the chain between the two residues in the pair. 158

B.1 Alpha helix, beta strand and coil amino acid propensities, in percentage terms, from 12,860 non-redundant PDB proteins sharing less than 30% sequence identity, using the DSSP program. Amino acids L, A and E (10.6, 9.74 and 8.63 % respectively) show a high propensity for alpha helices. Amino acids V, L and I (10.26, 8.91 and 7.88 % respectively) indicates high propensities for beta sheets. On the other hand, Amino acids S, G, A and K (9.28, 9.11, 7.50 and 7.05 % respectively) show high propensities for coils. 160

C.1 List of proteins for the third experiment of MECoMaP. . . . 162
C.2 List of proteins for the first experiment of MECoMaP. 163
C.3 List of proteins for the second experiment of MECoMaP. . . 164
C.4 List of proteins for the WEKA experiment. 165

Part I

Introduction

Chapter 1

Introduction

Bioinformatics has been described as the science of managing, mining, and interpreting information from biological sequences and structures [Gu and Bourne, 2003]. Two of the most important fields in bioinformatics are: genomics and proteomics. Genomics is the study and analysis of the genomes of organisms, while proteomics is defined as the characterization and identification of the proteins encoded in a genome.

This work thesis is framed within proteomics, and in particular, the field of structural bioinformatics. Structural bioinformatics is the branch related to the analysis and prediction of the three-dimensional structure of biological macromolecules such as proteins, ribonucleic acid (RNA), and deoxyribonucleic acid (DNA) [Gu and Bourne, 2003]. The prediction of the three-dimensional structure of a protein from its sequence of amino acids is one of the main open problems in structural bioinformatics. This problem is known as Protein Structure Prediction (PSP). The specific biochemical function of a protein is determined by its structure complexity, which, in turn, is determined by the specific sequence of amino acids. Therefore, solving this problem would allow to know the protein function directly from its amino acids sequence. Knowledge of protein structure has also great importance in different medical areas, *e.g.*, the treatment of some diseases like Alzheimer and Cystic fibrosis, or the development of new drugs.

With the success of the genome sequence projects, the amount of available protein sequences has increased dramatically. However, the number of protein structures available is relatively small. This is due to the difficulty of obtaining experimentally such structures. In fact, these structures can be experimentally determined using techniques such as X-ray crystallography or nuclear magnetic resonance (NMR). However, these techniques are expensive and time consuming.

This implies that it is crucial to develop computational methods for the automatic prediction of the 3D protein structures, as they would provide a cheaper and faster way to solve the PSP problem. Different approaches

have been developed to solve the PSP problem. These methods can be classified into two categories: statistical approaches and soft computing approaches. Statistical approaches are methods based on mathematical concepts, models (*e.g.*, bayesian models), and techniques (*e.g.*, Monte Carlo method), which are used in statistical analysis of data. Soft Computing is a branch of Artificial Intelligence focused on solving real problems dealing with incomplete, uncertain and inaccurate information. Therefore, Soft Computing provides capacity to deal with PSP problem. Among the main soft computing paradigms, there are artificial neural networks (ANNs), evolutionary computation (EC), and support vector machines (SVMs).

PSP can be divided into secondary and tertiary structure prediction [Gu and Bourne, 2003]. The problem of protein secondary structure prediction consists in predicting the location of α-helices, β-sheets and turns into a sequence of amino acids without any knowledge of the tertiary structure of the protein. The location of the elements in secondary structure can be used for approximation algorithms to obtain the tertiary structure of the protein.

On the other hand, methods for tertiary structure prediction are focused on determining contact or distance maps between amino acid residues of a protein sequence. When a contact map is defined, proteins can be folded and 3D structure can be obtained. Several contact map prediction methods have been applied to the PSP problem (*e.g.*, ANNs [Tegge *et al.*, 2009], SVMs [Cheng and Baldi, 2007], EC [Gupta *et al.*, 2005] and template-based modelling [Zhang, 2009]).

In this thesis, we proposed two methods, one for the prediction of secondary structure and one for the prediction of tertiary structures. Both methods are based on the EC paradigm. In particular, the proposed methods will provide a prediction model based on decision rules. The prediction is based on some physical-chemical properties and some structural features of protein residues.

1.1 Motivation

More and more amino acid sequences of proteins are available by the day, but their three-dimensional structures remain often unknown, and so their functions cannot be determined. Consequently the gap between protein sequence information and protein structural information is rapidly increasing. It follows that computational methods are needed in order to reduce this gap, as they would provide an inexpensive and fast way to solve the PSP problem.

This motivates us to propose two computational predictive methods, in this case evolutionary algorithms (EAs), for PSP. We believe that EAs well suited for solving the PSP problem, since PSP can be seen as a search

problem through the space determined by all the possible foldings. Such a space is highly complex and has a huge size. Finding the optimal solution in such space is very hard. In these cases, EAs have proven to be effective methods that can provide sub-optimal solutions. Our approaches will produce a set of rules that express conditions on the particular biochemical properties of the amino acids. Such a model can then be used in order to determine whether or not there is a contact between two amino acids or to determine a secondary structure conformation. An advantage of such approaches is that the generated rules can be easily interpreted by experts in the field in order to extract further insight of the folding process of proteins.

The main novelty of our proposal is that the prediction is based on a set of amino acid properties which are very important in the folding process [Gu and Bourne, 2003]. The reason for basing the prediction on such properties, is that it has been shown that amino acids that are in contact, are characterized by similar properties [Gupta *et al.*, 2005]. To the best of our knowledge, no other EA considers amino acid properties for the prediction.

1.2 Objectives

The three main objectives of this thesis are:

- To develop an approach based on evolutionary computation for the prediction of secondary protein structure motifs. The prediction model will consist of a set of rules that predict both the beginning and the end of the regions corresponding to a secondary structure state conformation (α-helix or β-strand). The prediction will be based on a set of specific amino acid physical-chemical properties.

- To develop a multi-objective evolutionary approach to predict protein contact maps. The algorithm will provide a set of decision rules, determining whether or not there is a contact between a pair of amino acids. Such rules will be based on a set of specific amino acid properties. These properties determine the particular features of each amino acid represented in the rules.

- To perform a deep analysis of the resulting rules in order to extract useful insights of the protein folding problem.

1.3 Overview

In chapter 2 we provide comprehensive background notions and definitions on proteomics, such as proteins, amino acids, peptide bonds, etc.

Chapter 3 introduces the main concepts of evolutionary computation and multi-objective optimization.

Chapter 4 presents a state of the art of different existing techniques which pretend to solve the PSP problem. Statistical and Soft Computing method descriptions are included.

Chapter 5 describes our proposals for secondary structure prediction as well as the algorithm for contact map prediction.

Chapter 6 presents the results achieved with the contact map predictor algorithm as well as a several analysis of the data sets and generated decision rules.

Chapter 7 propose a discussion of results achieved with the secondary structure predictor algorithm as well as several studies of data sets and obtained results.

Conclusions and future works are summarized in chapter 9.

Furthermore, at the end of this dissertation, we provide the appendix, the glossary, acronyms and the bibliography.

1.4 Contribution

The main contributions obtained as results of this thesis, are classified according to the related proposal:

1. Our proposal for the protein tertiary structure prediction, known as Multi-objective Evolutionary Contact Map Predictor (MECoMaP) is presented in [Márquez *et al.*, 2012a]. Two previous works also based in multi-objective approaches are shown in [Márquez *et al.*, 2012b, Márquez *et al.*, 2011a].

    - [Márquez *et al.*, 2012a] Evolutionary Decision Rules for Predicting Protein Contact Maps. Márquez, A.E., Asencio, G., Divina, F., Aguilar-Ruiz, J.S. Pattern Analysis and Applications (PAAA), Springer, (IF: 1.097), September 2012, pp. 1-13.
    - [Márquez *et al.*, 2012b] A NSGA-II Algorithm for the Residue-Residue Contact Prediction. Márquez, A.E., Divina, F., Aguilar-Ruiz, J.S., Bacardit, J., Asencio, G. In: 10th European Conference on Evolutionary Computation, Machine Learning and

Data Mining in Bioinformatics, (EvoBio 2012), Lecture Notes in Computer Science, 7246, pp. 234-244.

- [Márquez *et al.*, 2011a] A Multi-objective Genetic Algorithm for the Protein Structure Prediction. Márquez, A.E., Divina, F., Aguilar-Ruiz, J.S.. In: Proceedings of the 11th Annual ACM on Intelligent Systems Design and Applications (ISDA 2011), pp. 1086-1090.

2. Our proposal for the protein secondary structure predictor based on evolutionary computation is introduced in [Márquez *et al.*, 2011b, Márquez *et al.*, 2011c]. A previous work for the α-helix prediction is shown in [Márquez *et al.*, 2010a].

    - [Márquez *et al.*, 2011b] Protein Secondary Structures Prediction based on Evolutionary Computation. Márquez, A.E., Divina, F., JS Aguilar Ruiz. Applied computing Review ACM SIGAPP, 11(4), pp. 17-25.
    - [Márquez *et al.*, 2010a] Alpha helix prediction based on evolutionary computation. Márquez, A.E., Divina, F., Aguilar-Ruiz, J.S., Asencio, G.. Proceedings of the 5th IAPR international conference on Pattern recognition in bioinformatics (PRIB 2010), Lecture Notes in Computer Science, 6282, pp. 358-367.
    - [Márquez *et al.*, 2011c] Evolutionary Computation for the Prediction of Secondary Protein Structures. Márquez, A.E., Divina, F., Aguilar-Ruiz, J.S. Proceedings of the 26th Annual ACM Symposium on Applied Computing (SAC-2011), pp. 1087-1092.

3. An analysis of resulting folding rules generated by a protein contact map predictor is presented in [Bacardit *et al.*, 2012].

    - [Bacardit *et al.*, 2012] Contact map prediction using a large-scale ensemble of rule sets and the fusion of multiple predicted structural features. Bacardit, J., Widera, P., Márquez-Chamorro, A.E., Divina, F., Aguilar-Ruiz, J.S., Krasnogor, N. Bioinformatics (IF: 5.468), 28(19), pp. 2441-2448.

4. Other related contributions based in different soft computing schemes for the protein structure prediction are shown in the following publications:

    - [Márquez *et al.*, 2011d] Evolutionary Protein Contact Maps Prediction based on Amino Acid Properties. Márquez, A.E., Divina, F., Aguilar-Ruiz, J.S. 6th International Conference on Hybrid Artificial Intelligent Systems (HAIS 2011), Lecture Notes in Computer Science, 6678, pp. 303-310.

- [Márquez *et al.*, 2011e] Residue-residue Contact Prediction based on Evolutionary Computation. Márquez, A.E., Divina, F., Aguilar-Ruiz, J.S., Asencio, G. 5th International Conference on Practical Applications of Computational Biology and Bioinformatics (PACBB 2011), Advances in Intelligent and Soft Computing, 93/2011, pp. 279-283.
- [Márquez *et al.*, 2011f] An Evolutionary Approach for Protein Contact Map Prediction. Márquez, A.E., Divina, F., Aguilar-Ruiz, J.S., Asencio, G. 9th European Conference on Evolutionary Computation, Machine Learning and Data Mining in Bioinformatics (EvoBio 2011), Lecture Notes in Computer Science, 6623, pp. 101-110.
- [Asencio *et al.*, 2012] Prediction of Mitochondrial Matrix Protein Structures Based on Feature Selection and Fragment Assembly. Asencio, G., Aguilar-Ruiz, J.S., Márquez, A.E., R Ruiz, C E Santiesteban. 10th European Conference on Evolutionary Computation, Machine Learning and Data Mining in Bioinformatics (EvoBio 2012), Lecture Notes in Computer Science, 7246, pp. 156-167.
- [Santiesteban *et al.*, 2012] Short-Range Interactions and Decision Tree-Based Protein Contact Map Predictor. Santiesteban, C.E., Asencio, G., Márquez, A.E., Aguilar-Ruiz, J.S. 10th European Conference on Evolutionary Computation, Machine Learning and Data Mining in Bioinformatics (EvoBio 2012), Lecture Notes in Computer Science, 7246, pp. 224-233.
- [Asencio *et al.*, 2011a] Prediction of protein distance maps by assembling fragments according to physicochemical similarities. Asencio, G., Aguilar-Ruiz, J.S., Márquez, A.E. 5th International Conference on Practical Applications of Computational Biology and Bioinformatics (PACBB 2011), Advances in Intelligent and Soft Computing, 93, pp. 271-278.
- [Asencio *et al.*, 2011b] A nearest neighbor-based approach for viral protein structure prediction. Asencio, G., Aguilar-Ruiz, J.S., Márquez, A.E.. 9th European Conference on Evolutionary Computation, Machine Learning and Data Mining in Bioinformatics (EvoBio 2011), Lecture Notes in Computer Science, 6623, pp. 69-76.
- [Santiesteban *et al.*, 2011] A Decision Tree-Based Method for Protein Contact Map Prediction. Santiesteban, C.E., Márquez, A.E., Asencio, G., Aguilar-Ruiz, J.S. Evolutionary Computation, Machine Learning and Data Mining in Bioinformatics - 9th European Conference (EvoBio 2011), Lecture Notes in Computer Science, 6623, pp. 153-158.

- [Márquez *et al.*, 2011g] Un Algoritmo Genético para la Predicción de Mapas de Contacto Basado en Propiedades de Aminoácidos. Márquez, A.E., Divina, F., Aguilar-Ruiz, J.S., Asencio, G. Actas de la XIV Conferencia de la Asociación Española para la Inteligencia Artificial (CAEPIA 2011).
- [Asencio *et al.*, 2011c] Predicción de Mapas de Distancia de Proteínas basados en Vecinos más Cercanos, Asencio, G., Aguilar-Ruiz, J.S., Márquez, A.E.. Actas de la XIV Conferencia de la Asociación Española para la Inteligencia Artificial (CAEPIA 2011).
- [Márquez *et al.*, 2010b] Definición de umbral mínimo para la predicción de estructura secundaria de proteínas. Márquez, A.E., JS Aguilar Ruiz, Anguiano, E. Actas del XV Congreso Español sobre Tecnologías y Lógica Fuzzy (ESTYLF 2010), pp. 465-470.
- [Márquez *et al.*, 2009] Marco de Referencia en la Calidad de la Predicción de Mapas de Contacto de Proteínas. Márquez, A.E., Aguilar-Ruiz, J.S., Anguiano, E. In: Actas de la XIII Conferencia de la Asociación Española para la Inteligencia Artificial (CAEPIA 2009), pp. 11-19.

1.5 Summary

In this chapter we have introduced the main characteristics and the context of the thesis. This thesis is framed within the field of proteomics and more specifically within the protein structure prediction problem. Throughout the document, two proposals for the resolution of the problem will be described. We have also presented our motivations and main objectives. An overview of this thesis document with a brief description of the chapters is also provided. Finally, the contributions obtained as result of this thesis are listed.

Chapter 2

Biological background

In this chapter, we will provide the reader with some basic notions of proteomics, such as a description of proteins or protein structure. These aspects will be needed in the rest of this thesis. We will also describe an approach that is commonly used in the protein structure prediction problem.

2.1 Proteins

Proteins are an important class of biological macromolecules present in all biological organisms. They form the basis of cellular and molecular life and significantly affect the structural and functional characteristics of cells and genes. Numerous functions, as structural support, mobility, protection, regulation or transport, are developed by proteins in the cells (table 2.1). For instance, hemoglobin protein transports oxygen molecules in the red blood cells of all vertebrates, and myosin proteins are known for their role in the muscle contraction, as cardiac myosin which regulates the heart's pumping.

A protein can be seen on four different levels depending on which structures of the protein are considered. Essentially, primary structure of proteins consist of linear sequences of twenty natural amino acids joined together by peptide bonds. Change in a single amino acid in a critical area of a protein can alter its biological function. The secondary structure of a protein is the folding or coiling of the peptide chain. The tertiary structure is the three dimensional shape of the polypeptide chain. The quaternary structure is the final dimensional structure formed by all the polypeptide chains making up a protein. We will return on these concepts in the next section.

The process of synthesis of a protein, or production of proteins, in the cell is divided into several steps; DNA sequences are first transcribed into messenger RNA (mRNA) sequences, which are then translated into protein sequences. These protein sequences fold into three-dimensional structures with a determined function [Gu and Bourne, 2003]. More details on the

protein synthesis will be provided in section 2.3.

Table 2.1: A selective list of functional roles for proteins within cells.

Funtion	Examples
Transport proteins	Hemoglobin, myoglobin, ceruloplasmin
Effector proteins	Insuline, thyroid hormone
Structural proteins	Keratin, collagen, elastin
Contractil proteins	Actin, myosin, tubulin
Defence proteins	Ricin, inmunoglobins, toxins
Enzymes or catalytic proteins	Trypsin, DNa polymerases
Receptors	CD4, acetycholine receptor
Repressor proteins	Methionine repressor MetJ, lac repressor
Chaperones	GroEL, DnaK
Storage proteins	Ferritin, gliadin

The knowledge of the amino acid sequence of a protein is a very important issue [Berg and Stryer, 2008]. Amino acid sequence determines the structure of proteins and is the link between the genetic message in DNA and the three-dimensional structure which is associated to a biological function. Therefore, the knowledge of the sequence is essential to discover the protein functionality. On the other hand, the knowledge of the structure of the protein provides a great advantage for the development of new drugs and the design of new proteins.

2.2 Protein structure

Proteins are formed by union of simpler substances called amino acids. All amino acids contain carbon, hydrogen, nitrogen and oxygen with two of them also containing sulfur. There are twenty types of different amino acids: Alanine (Ala), Arginine (Arg), Asparagine (Asn), Aspartic acid (Asp), Cysteine (Cys), Glutamine (Gln), Glutamic acid (Glu), Glycine (Gly), Histidine (His) Isoleucine (Ile), Leucine (Leu), Lysine (Lys), Methionine (Met), Phenylalanine (Phe), Proline (Pro), Serine (Ser), Threonine (Thr), Tryptophan (Trp), Tyrosine (Tyr) and Valine (Val). An amino acid is a molecule formed by an amino group and a carboxyl group and a variable R group. This chemical group, also known as side chain, determines the identity and properties of the different amino acids (figure 2.1) and is attached to the central carbon atom. The different R groups provides to amino acids individual characteristics (*e.g.*, -SH, -OH, -NH_2). The sequence of different amino acids will then give each protein unique characteristics.

A protein consists of polymers of amino acids. Polymers, also known

as polypeptides, consist of a sequence of amino acids, linked together by hydrogen bonds called peptide bonds. A peptide bond is a covalent chemical bond[1] between the amino group (a functional group that consists of one nitrogen atom and two hydrogen atoms, -NH_2) of an amino acid and the carboxyl group (a functional group consisting of a carbonyl (RR'C=O) and a hydroxyl (R-O-H)-COOH, usually written as -COOH) of the next amino acid. This reaction is described as a condensation resulting in the elimination of a molecule of water (H_2O) and the formation of a dipeptide. An amino acid unit in the polypeptide chain is called a residue.

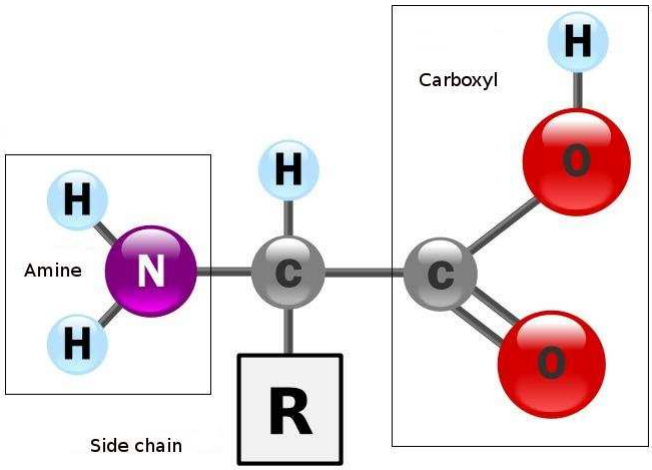


Figure 2.1: Amino acid. Chemical composition.

The main chain or backbone of the polypeptide chain is established by the formation of peptide bonds between amino acids. Figure 2.2 shows a peptide unit, where ϕ is the rotation angle around the N–C_α bond while ψ is the rotation angle around the C_α–C bond. These rotations determine each protein structure.

According to their structure, proteins have a structural hierarchy containing primary, secondary, tertiary and quaternary levels.

2.2.1 Primary structure

The primary structure defines the linear sequence of assembled amino acids. At each boundary of the sequence, there is a free amino or carboxyl group, these are referred as N and C terminus which represents the unbounded N and C atom in the amino or carboxyl group respectively (Figure 2.3.a).

[1] In a covalent chemical bond the atoms are bounded by shared electrons, in contrast to ionic bonds where the atoms are bounded by the attraction between oppositely-charge ions. Such bonds lead to stable molecules.

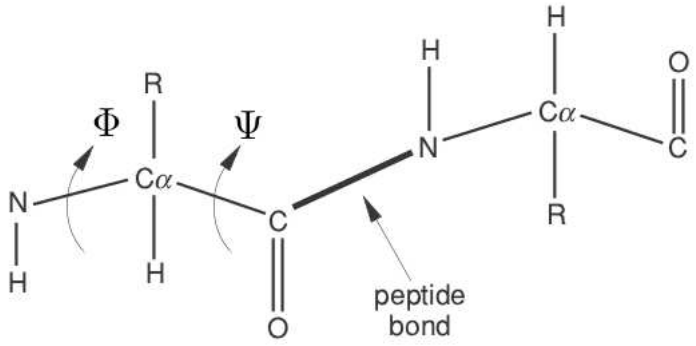


Figure 2.2: A peptide unit.

2.2.2 Secondary structure

The secondary structure of the protein refers to the interactions due to a regular arrangement of hydrogen bonds between CO and NH groups (carboxyl and amino) of its amino acids, forming different motifs. These motifs are α-helix, β-sheet, loops and turns. Repetitive motifs appear in a secondary structure, and the most common kind of motif is α-helix (Figure 2.3.b). An α-helix is a dextro-helical structure, with about 3.6 amino acids per turn. Such structure is held by hydrogen bonds, where the amino group of amino acid n provides a hydrogen bond with the carbonyl group (a functional group composed of a carbon atom double-bonded to an oxygen atom, C=O) of the amino acid $n + 4$. On the other hand, β-sheets are characterized by theirs flattened and extended shape. They have a maximum number of hydrogen bonds between peptides that provides stability to the structure. Several peptide chains (β-strands), that are held together with hydrogen bonds in a zig-zag, constitute a β-sheet motif. This lamellar structure proportionates flexibility but no elasticity. The adjacent chains of a β-sheet can be targeted in the same direction (parallel β-sheet) or opposite direction (antiparallel β-sheet).

The α-helix and the β-sheet are areas of repetitive conformation, repeating the values for the torsion angles of the polypeptide chain, which describe the rotations of the polypeptide backbone around the bonds between N-C_α (called Phi, ϕ) and C_α-C (called Psi, ψ). On the other hand, loops and turns are other secondary structure motifs which are non-repetitive regions. These two conformations cause changes in the direction of the polypeptide chain. Many of these changes are caused by a common structural unit called β-turn. In this structure, the carbonyl group of a residue n provides a hydrogen bond with the amino group of the residue

$n+3$. There are several types of β-turn: type I, type II and type III[2], among others. A typical protein contains about 32% α-helices, 21% β-sheets and 47% loops or non-regular structures.

Sometimes, it is observed that certain structural components comprising a number of secondary structures, are frequently repeated within proteins, *e.g.*, two α-helices joined by a loop region. These are termed supersecondary structures. Some of these structures are associated with certain biological functions, while others are part of larger structural or functional units.

2.2.3 Tertiary structure

The tertiary structure is a description of the complex and irregular folding of the peptide chain in three dimensions (figure 2.3.c). These complex structures are held together by a combination of several molecular interactions that involve the R-groups of each amino acid in the chain. These interactions are determined and stabilized by chemical bonds and forces, including weak bonds[3], such as hydrogen bonds[4], ionic bonds[5], Van der Waals bonds[6], hydrophobic interactions[7] and covalent bonds, such as disulfide bond (S-S).

The tertiary structure can be altered by a number of factors that will interfere with the molecular processes that hold the structure together. These include changes in temperature, pH[8] and ionic strength[9]. Since the function of a protein is dependent on its structure, any factor that affects the structure will also affect the activity of the protein.

2.2.4 Quaternary structure

The quaternary structure is the final level of structural hierarchy. Although some proteins are monomeric, *i.e.*, consist of only one polypeptide chain, others are multimeric, and consist of several chains. These subunits may work cooperatively, and the functional state of one subunit can depend on the state of the other units. An example of a protein with quaternary structure is hemoglobin. Hemoglobin, an oxygen transport protein, is a tetramer (four units) consisting of two subunits of α-proteins and two

[2]These classes were defined according to ϕ and ψ angles.

[3]Weak bonds are those forces of attraction that do not take a large amount of energy to break.

[4]A type of attractive interaction between an electronegative atom and a hydrogen atom bonded to another electronegative atom.

[5]A type of chemical bond formed through an electrostatic attraction between two oppositely charged ions.

[6]Weak forces which contribute to intermolecular bonding.

[7]Describe the relations between water and hydrophobes (low water-soluble molecules).

[8]A measure of the acidity or basicity of the solution.

[9]A measure of the concentration of ions in the solution.

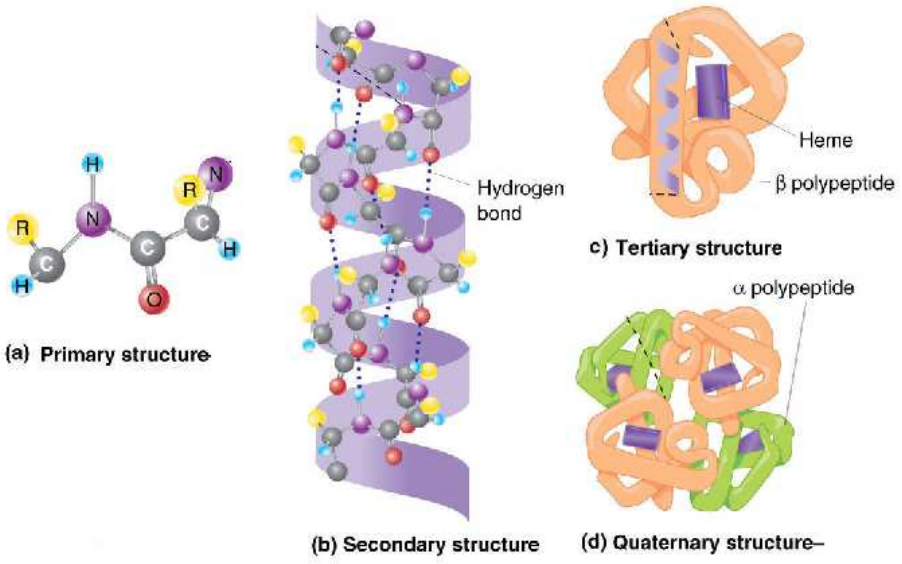


Figure 2.3: Four levels of protein structure.

subunits of β-proteins. Figure 2.4 shows the quaternary structure of hemoglobin.

2.3 Protein synthesis

Proteins are synthesized in the ribosomes. The processes of formation of a protein are known as transcription and translation, concepts that we address in the following.

The instructions to manufacture proteins are contained in the deoxyribonucleic acid (DNA) of an organism. A DNA strand is formed by two molecule strands which are linked together through hydrogen bonds. Each strand is made up of a long sequence of nucleotides. There are four types of nucleotides or bases: adenine (A), cytosine (C), guanine (G) and thymine (T). Between the two strands of DNA, different bases pair up with each other: A with T and C with G. Therefore, a single strand contains all the information of the whole DNA molecule. DNA strands are tightly pack and this packaging is known as a chromosome. Human DNA is packed into 46 chromosomes, two sets of 23. The term genome refers to all the hereditary information contained in the DNA (both genes and non-coding regions). A gene is a section of a DNA strand that has the code for a specific protein.

On the other hand, RNA is a nucleic acid whose main task is the transfer of the genetic code, from the core to the ribosomes, for the creation of

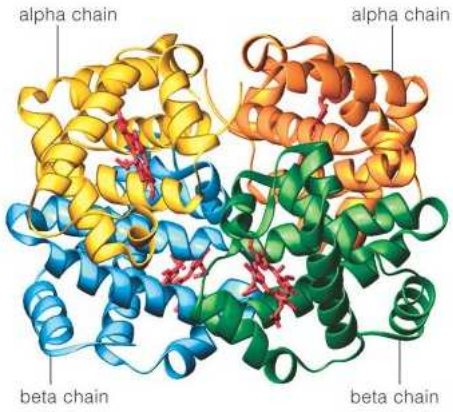


Figure 2.4: Quaternary structure of the Hemoglobin

proteins. RNA is similar to DNA: it consists of strands of four different bases attached to a backbone. RNA remains as a single strand. Like DNA, RNA has the bases A, C and G but instead of thymine, it uses another base, called uracil (U). A type of RNA is the messenger ribonucleic acid (mRNA) which is made by the process called transcription, where the genetic information on a strand of DNA is used to synthesize a strand of complementary RNA. The created mRNA molecule then travels from the core where the DNA is contained to the ribosomes in the cytoplasm.

In the ribosome, assisted by transfer ribonucleic acid (tRNA), each sequence of three nucleotides in the mRNA is interpreted as an instruction (known as a codon) to manufacture a specific type of amino acid. The process by which this takes place is known as translation. The pairing between codons and the 20 amino acids is known as the genetic code [Gu and Bourne, 2003].

The construction of a protein is started when the codon AUG appears in the mRNA sequence. AUG codes for the amino acid methionine. This construction is stopped when one of the codons, UAA, UAG or UGA is found in the mRNA sequence. The flow of information that constitutes the transcription of DNA into mRNA, and the translation of mRNA into proteins is the central dogma of molecular biology. After this processes, the folding of the protein is carried out.

2.4 Protein Folding

A protein spontaneously folds into a 3-dimensional structure after having been manufactured in the ribosomes. The 3D structure of a protein determines its function. A specific protein will fold in the same way and

will end up with the same 3D structure. This phenomenon is called the native state of the protein.

Protein folding represents the process whereby higher structures are formed from the primary structure. A folded protein can have more than one stable folded state or conformation. Each conformation has its own biological activity. At any stage, only one conformation is active. The transitions between different conformations are called conformational changes.

Sometimes, a protein can fold into a wrong shape. We know the folding rules lie in the amino acid sequence, however, in some cases, a type of protein called chaperones, is used to keep their target proteins from folding incorrectly. Some factors can influence in the misfolding of a protein, such as temperature, solvent viscosity and acidity.

A single missing or incorrect amino acid could cause such a misfold. As already stated, protein function is determined by its structure, therefore a misfold implies that a protein can not fulfill its function correctly. Alzheimer's disease, Cystic fibrosis, Bovine spongiform encephalopathy (mad cow disease) and its human variant are now all attributed to protein misfolding. The knowledge of the misfolding factors and understanding the protein folding process, would help in developing cures for these diseases.

Since Anfinsen's experiment discovered that the amino acid sequence determines the shape of a protein [Anfinsen, 1972], a huge number of computational experiments were performed with the aim of obtaining the rules of the protein folding.

2.5 Classification of Protein Structures

Protein tertiary structures are organized into domains. A domain is a region of the protein which is able to fold by itself into a stable three-dimensional structure (native structure). A protein may be constituted by one or more domains. Different domains are divided into groups or subgroups.

The most widely used protein structure classifications are CATH and SCOP, which are based on a hierarchical classification of the different known domains.

2.5.1 SCOP

Practically all proteins have structural similarities with other proteins and in some cases this similarity is accompanied by a common evolutionary origin. The Structural Classification of Proteins database (SCOP) [Murzin *et al.*, 1995] provides a detailed and comprehensive description of structural and evolutionary relationships among known protein structures. Proteins are classified according to their SCOP class described as follows: Alpha proteins contain only alpha helical secondary structure. Beta

proteins contain only beta-sheet secondary structure. Alpha/beta proteins contain alternating α-helical and β-sheet secondary structure elements. This structure is known as a TIM barrel. In alpha/beta proteins, the α-helical and β-sheet regions occur in independent regions of the molecule. Small proteins are usually dominated by metal ligand or disulphide bridges. Finally, Coiled-coil proteins refers to a structural motif in which α-helices are coiled together.

2.5.2 CATH

CATH [Orengo *et al.*, 1997] is a manually curated classification of protein domain structures. CATH is an acronym of the four main levels in the classification (class, architecture, topology and homology). These four levels are defined as follows: Class classification, considers the secondary structure of the protein. Proteins are divided into: Mainly α, Mainly β, α-β and few secondary structures. Architecture: this level generally describes the shape of the domain based on the orientations of secondary structure elements, but omits connectivities of these elements (e.g. Barrel, *beta*-propellor). Topology: at this level, the connection of the various elements within the various architectures is described. These are called folding families (e.g. α-bundle, β-barrel, β-sandwich). Homology: this level takes into account the homology of primary structure sequences. Different categories are referred to groups of proteins (superfamilies) with a sequence homology such that refers to a common phylogenetic ancestor[10] (evolutionary relationship).

2.6 Protein Structure Prediction Problem

The protein structure prediction problem (PSP) consists in determining the structure of a protein, that is, the prediction of its secondary, tertiary, and quaternary structures, using only information contained in its amino acid sequence. As already stated the knowledge of the 3D structure of a protein would be of enormous help for designing new drugs for diseases such as cancer or Alzheimer. Although there exist experimental methods for determining protein structures, e.g., X-ray crystallography and nuclear magnetic resonance, such techniques are very expensive and present limitations with the structures of certain proteins [Jaravine *et al.*, 2006, Lattman, 2004, Service, 2005].

The primary structure, or amino acid sequence, of a protein is much easier to determine than its tertiary structure. Moreover, the gap between the number of proteins with known sequence and the number of proteins with known tertiary structure is rapidly increasing (Figure 2.5). This is caused by the limitations of methods mentioned above. In order to reduce this

[10] Phylogenies describe the evolution of domain structures and proteomes.

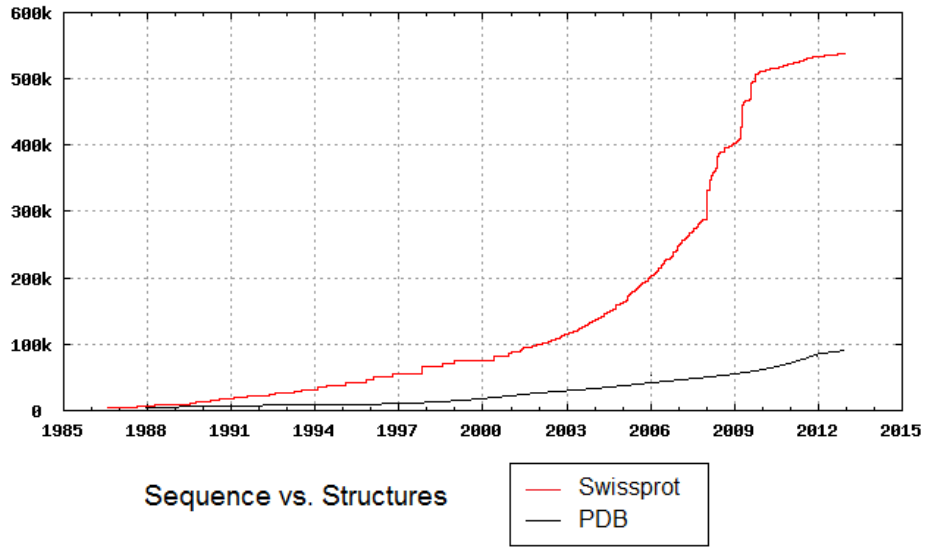


Figure 2.5: The number of known protein sequences (Swissprot) versus the number of known structures (PDB).

gap, there have been many researches focused on determining the tertiary structure of a protein from its sequence. The high number of protein sequences whose three-dimensional structures must be determined, make computational methods for protein structure prediction an essential tool.

Unfortunately, there is still no method that predicts tertiary structure accurately enough. The accuracy achieved by the most recent and relevant proposals in the literature is up to 30% approximately [Monastyrskyy *et al.*, 2011], and clearly must be improved. In order to promote research in this field, the performance of current methods is assessed in the CASP experiment (Critical Assessment of Techniques for Protein Structure Prediction) every two years.

Different methods may be classified in secondary, tertiary and quaternary structure prediction methods, according to the type of structure to be predicted.

The problem of protein secondary structure (SS) prediction consists in predicting the location of α-helices, β-sheets and turns from a sequence of amino acids without any knowledge of the tertiary structure of the protein. A first generation of SS prediction methods were based on single residue statistics. The second generation of prediction techniques was a combination of a larger database of protein structures and the use of statistics based on segments. These algorithms combined statistical informations, physico-chemical properties, sequence patterns, multi-layered

neural networks, graph-theory, multivariate statistics, expert rules and nearest-neighbor algorithms. Methods in the third generation includes evolutionary information from homologous sequences. Secondary structure prediction represents an important topic in bioinformatics. It can be considered the previous step for the 3D image reconstruction of the proteins. By knowing the position of secondary structure motifs such as α-helices, β-sheets or turns we can obtain an approximate model of the tertiary structure of the protein.

Protein tertiary structure prediction consists in predicting the three-dimensional model of a protein. In the PSP literature, there are three main data structures to represent a protein 3D structure: torsion angles, distance maps and contact maps. Torsion angles represent the values of the flexible angles of a protein molecule. Torsion angles are based on the assumption of constant bond lengths and some constant bond angles between atoms. This representation is based on three torsion angles in the protein backbone plus the angles in protein side chains. This is a simplification of the real situation, where the supposed constant bond lengths and angles depend on the environment of atoms. Examples of recent proposals that predict protein torsion angles are [Faraggi *et al.*, 2009] and [Furuta *et al.*, 2009]. On the other hand, distance maps represent the distances between reference atoms of each pair of protein residues. Examples of methods that predict protein distance maps are [Cortes *et al.*, 2011, Kloczkowski *et al.*, 2009]. Contact maps are the most commonly used structure in the PSP literature. In a nutshell, a contact map represents binary proximities between each pair of protein residues, which are predicted by residue-residue contact predictors. In addition to this, some proposals discretize the distances between atoms, providing an intermediate representation between contact and distance maps. For instance, Walsh et al. [Walsh *et al.*, 2009] use 4-class distance maps. More details of contact and distance maps will be provided in section 2.8.

As already mentioned, in PSP, the prediction of the 3D structure of a protein must be based on characteristics of the amino acids forming its sequence. Some commonly used features are the physico-chemical properties of residues. Usually the properties that are used are hydrophobicity, polarity, charge and residue size, as well as the properties of the AAindex repository [Kawashima *et al.*, 2008], which contains currently 544 amino acid properties. On the other hand, predictors often use secondary structures (commonly from DSSP [Kabsch and Sander, 1983] or PSIPRED [Jones, 1999]), solvent accessibility [Lippi and Frasconi, 2009], evolutionary information (commonly the Position Specific Scoring Matrix (PSSM) from PSIBLAST [Altschul *et al.*, 1997]) and contact orders (usually CO [Plaxco *et al.*, 1998], RCO [Kihara, 2005], CN [Kinjo *et al.*, 2005] or the most recent RWCO [Song and Burrage, 2006]). Some authors also used topological measures of the protein molecule like the recursive convex hull

[Stout *et al.*, 2008]. Several types of approaches have been proposed in the literature with the aim of computationally solving the PSP problem. These methods will be detailed in chapter 4. Within such proposals, evolutionary algorithms (EAs) have proven to achieve excellent results (see, for instance [Calvo *et al.*, 2011]). EAs have become popular as robust and effective methods for solving optimization problems. In particular, EAs have shown the capacity of finding suboptimal solutions in search spaces when the search space is characterized by high dimensionality. This is the case of the protein folding problem, where the set of possible folding rules of a protein determine the search space.

Quaternary structure prediction aims to predict the interactions between proteins. Protein interactions can be transitory (signaling networks) or relatively permanent (*e.g.* multi-protein complexes). Many important cellular processes are carried out by protein complexes. Methods belonging to this area can assist in the prediction of interaction sites on protein surface and in the prediction of the structure of the intermolecular complex formed between two or more molecules (docking). Computational techniques have been developed for docking [Janin, 2010, Vajda and Kozakov, 2009], and also can help to infer 3D structure of a protein from the knowledge of the protein interactions [Fornes *et al.*, 2009]. There are two strategies for modeling the interaction between two proteins from sequence data. The first one is to model the unbound sequence proteins and to dock them into the final complex (*i.e.*, solving first the tertiary structure of the proteins and afterwards the quaternary). The second is to model the interacting pair or complex using as template the structural knowledge of an available homologous interacting pair.

2.7 The PDB Protein Structure Data Archive

The Worldwide Protein Data Bank (PDB) [Berman *et al.*, 2003], is an international collaboration which organizes the processing and distribution of the PDB file. The online PDB file [Berman *et al.*, 2000], is a repository that coordinates and relates information of about 90,000 structures (88,325 structures in February 19, 2013), including proteins, nucleic acids, and complex macromolecules that have been obtained through techniques of X-ray crystallography, nuclear magnetic resonance (NMR) and electron microscopy (see Glossary for more information). All specific information determined by these experiments is stored in a file for each protein.

These files are divided into several sections. Each section has various records which specify the different characteristics of the protein structure.

The Primary structure section contains SEQRES records, which contain a list of consecutive chemical components covalently linked to form a linear polymer. The chemical compounds included in this list may be of standard

amino acids or nucleic acid residues.

The Secondary structure section contains relevant information about secondary structure of the protein. HELIX records are used to identify the position of helices in the molecule. Helices are named, numbered, and classified by type. The residues where the helix begins and ends are noted, as well as the total length. SHEET records are used to identify the position of sheets in the molecule. As with helices, sheets are also named and numbered, and the residues where the sheet begins and ends are noted. Coordinate section stores the atomic coordinates of each atom in the protein. The MODEL record specifies the model serial number when multiple models of the same structure are presented in a single coordinate entry, as is often the case with structures determined by NMR. The ATOM records present the atomic coordinates for standard amino acids and nucleotides. They also present the occupancy and temperature factor for each atom. Non-polymer chemical coordinates are maintained by the HETATM records. ATOM records contain the element symbol. Optionally this record can also hold the charge. The ENDMDL records are paired with MODEL records to group individual structures found in a coordinate entry.

2.8 Contact maps

The native structure of a protein is approximated by the set of the coordinates listed in its PDB file. If a protein contains N atoms, the corresponding representation requires $3N$ coordinates. An alternative view of the protein makes use of a distance matrix, a symmetric square $N \times N$ matrix whose elements are the distances among the atoms in the protein. This representation is obviously redundant. However, it is still very important, since it has been demonstrated that the redundancy can help in the reconstruction of the 3D structure of the protein only when some elements of the distance matrix are available.

In order to establish the distance between two residues, the distance between the C_α atoms can be used [Duarte *et al.*, 2010]. Alternatively, C_β atoms can be considered or again the minimal distance between atoms belonging to the side chain or to the backbone of the two residues.

2.8.1 Binary contact map

A contact map is a binary version of the distance matrix defined as a square symmetric matrix of order L, where L is the number of amino acids in the sequence. The contact map is divided into two parts: the observed part (upper triangular) and the predicted part (lower triangular). An element (i, j) of the contact map is 1 if amino acids i and j are in contact, or 0 otherwise. In this context, we consider two amino acids to be in contact if the

distance between them is less than or equal to a given threshold. To this aim, a commonly used threshold is 8 angstroms (Å) [Monastyrskyy *et al.*, 2011].

Usually contacts between amino acids are divided and predicted by groups according to their sequence separation. Sequence separation between amino acids a_i and a_j, where i and j represent the positions of the residues in the sequence, is $|i - j|$. Based on the separations, contacts are classified into three classes: short, medium and long range. In short range, a minimum separation of six residues is used in order to consider a contact, whereas in medium and long range, the minimum separations are 12 and 24, respectively. An example of binary contact map is shown in figure 2.6. A cell with black color represents a contact for a determined pair of amino acids, while a white cell represents a non-contact. In this case, predicted contacts are located in the lower triangle. The upper triangle stores the real contacts of the protein.

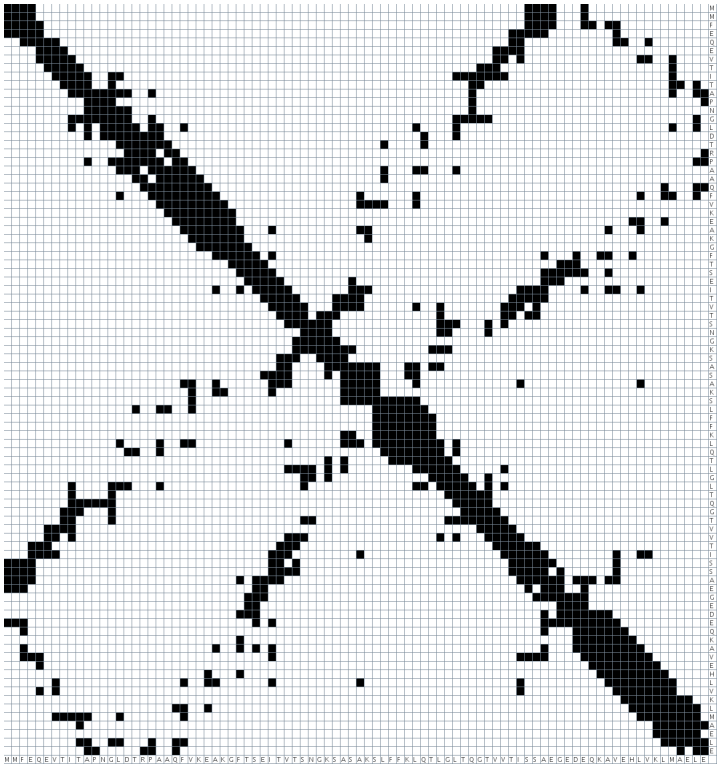


Figure 2.6: Example of a binary contact map.

Figure 2.7 represents a binary contact map in which we can graphically appreciate the different secondary structure elements (α-helices and β-sheet). This example denotes the potential of this representation which stores tertiary and secondary structure information at the same time.

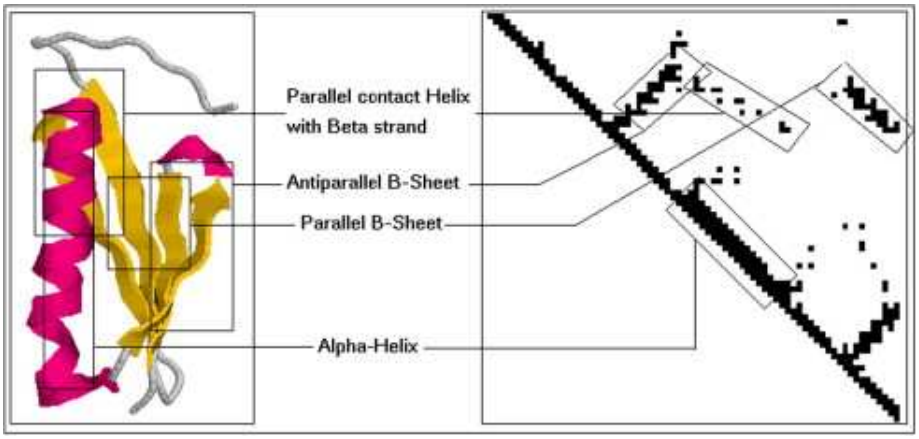


Figure 2.7: Example of a binary contact map with the graphical representation of secondary structure elements.

2.8.2 Distance matrix

Assuming that the main drawback of binaries contact maps is the data loss that occurs in the discretization, an alternative view of the protein is a distance matrix. A distance matrix is a symmetric square $L \times L$ matrix whose elements are the distances among the atoms in the protein. The calculation of the distances between the residues is determined by the Euclidean distance. An example of distance map is represented in figure 2.8. Estimated distances are located in the lower triangle and real distances are located in the upper triangle. A cell with red color represents a contact or proximity of contact for a determined pair of amino acids. On the other hand, a blue cell represents a high distance in angstroms and consequently a non-contact.

2.8.3 Fuzzy contact map

Fuzzy contact maps were introduced with two aims: to take into account potential measurement errors in atom coordinates, and to allow highlighting features that occurs at different thresholds. Formally, a fuzzy contact is defined by:

$$F_{i,j} = \mu(\overline{[i,j]}, \Re) \tag{2.1}$$

where $\mu()$ is a particular definition of (fuzzy) contact, [i, j] stands for the Euclidean distance between residues i, j, and $\Re$ is the threshold as for the crisp contacts. The standard, i.e. binary, contact map is just a special case of the fuzzy contact map. Fuzzy contact maps are further generalized by removing the constraint (in the original model) of having only one threshold $\Re$ as a reference distance. The formal definition of a General Fuzzy Contact

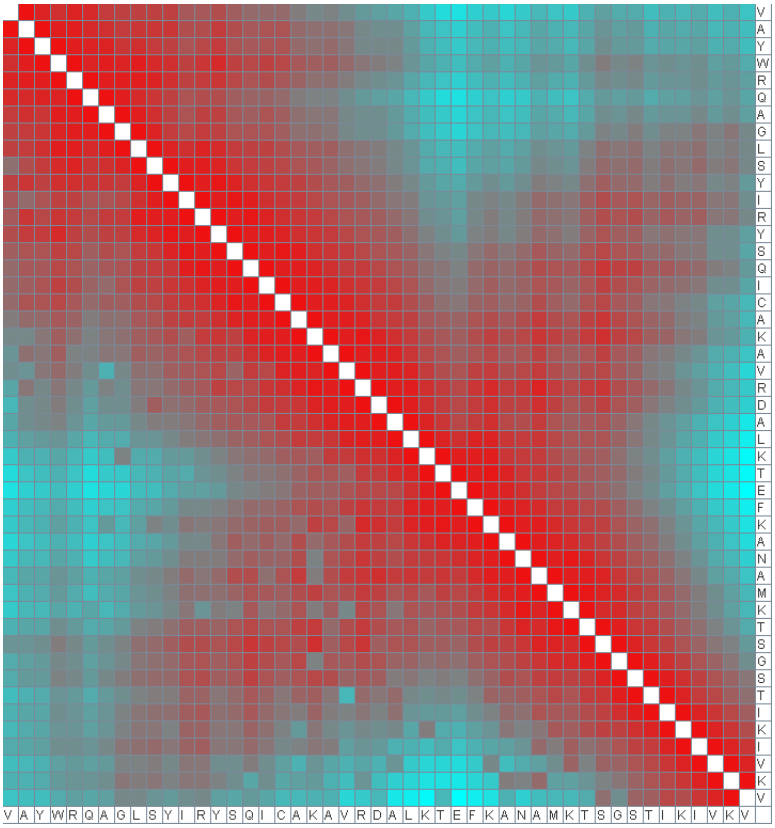


Figure 2.8: Example of a distance map for 1E79I protein.

is given by:

$$F_{i,j} = max\{\mu_1(\overline{[i,j]}, \Re_1), ..., \mu_n(\overline{[i,j]}, \Re_n)\} \tag{2.2}$$

That is, up to n different thresholds and up to n different semantic interpretations of contact are used to define the $L \times L$ contact map.

2.9 Summary

In this chapter we have summarized the basic concepts in proteomics. We have provided a definition of proteins according to their structure level (primary, secondary, tertiary and quaternary levels). An explanation about the protein formation is also provided. We have analyzed the protein folding and the consequences of a bad folding. Furthermore, we have defined the problem of protein structure prediction and have been exposed the different structural classifications of proteins. Finally, we have described an useful concept for the PSP, named contact map.

Chapter 3

Evolutionary Computation

In this chapter, we will introduce the main concepts of evolutionary computation and multi-objective optimization.

3.1 Introduction

Evolutionary computation (EC) is a population-based stochastic iterative optimization technique based on the Darwinian concepts of evolution. EC tackles difficult problems by evolving approximate solutions of an optimization problem. EC have been applied to find solutions of problems in a variety of domains, *e.g.*, finance and economics, optimization, design, classification or biological modelling among others.

EC can be divided into two main areas; evolutionary algorithms (EAs) and swarm intelligence. The proposed approaches in this thesis are based on EAs. EAs combine the notions of survival of the fittest individual with a structured and random exchange of features between individuals in a population of possible solutions. Imitating the mechanics of biological evolution in nature, genetic algorithms operate on a population of possible solutions of the problem. Each element of the population is called individual. An individual represents a possible solution of the problem. EAs use a population of candidate solutions, that are evolved through a number of generations. At each generation, a set of candidates are selected for reproduction and mutation. These candidates are evaluated and the best candidate is selected at the end of the evolutionary process.

These algorithms follow the method of selection of individuals within a specie summarized in the next rules: individuals with better genetic information are more likely to reproduce. Evolution is caused by the combination of the parental chromosomes.

The main advantages of the use of evolutionary algorithms in solving optimization problems are among others: these methods operate on a population (or set of solutions). They do not require previous knowledge

of the problem to be solved. These algorithms can be combined with other search techniques to improve their performance and are easily parallelizable. Furthermore, these methods are conceptually easy to implement and use.

Within evolutionary algorithms, we can find four basic paradigms:

Evolution Strategy (ES): ES uses a natural problem-dependent representations consisting of real-valued vectors. ES uses mutation operator as main exploratory search operator, but nowadays ES use also crossover. ES was introduced in [Rechenberg, 1973].

Evolutionary Programming (EP): EP was originally introduced for developing finite state automata for solving specific problems. One representation commonly used is a fixed-length real-valued vector. EP does not rely on any kind of recombination. EP was presented in [Fogel *et al.*, 1966].

Genetic Algorithm (GA): GAs are optimization algorithms which try to find the best solution to a given problem from a set of possible solutions. The mechanisms used by GAs to carry out this search can be seen as a metaphor of the processes of biological evolution (reproduction and mutation). This kind of optimization and search algorithms can be applied to solve optimization problems in various fields [Goldberg, 1989]. GAs were developed by John Holland, with his research team at the University of Michigan in the 1970's [Holland, 1975].

Genetic Programming (GP): GP is a methodology inspired by biological evolution to build computer programs that perform a user-defined task. It is a specialization of genetic algorithms where each individual is a computer program. Individuals typically are tree structures. GP was first described in [Koza, 1992].

Algorithm 1 General scheme of GAs

```
begin
  Initialize population
  Evaluate each individual in population
  while not (Stopping criteria) do
    Select parents
    Recombine pairs of parents
    Mutate the resulting offspring
    Evaluate offspring
    Create new population
  end while
  Extract solution from population
end
```

A GA general scheme is presented in algorithm 1. First step of the scheme is the initialization. In this step, an initial population is randomly generated. This population consists of a set of chromosomes which represent possible solutions of the problem. It is important to ensure that we have a structural diversity of the solutions within the initial population in order to prevent a premature convergence. The second step is the evaluation of the individuals that will apply a fitness function to know how "good" is the encoded solution. After this, the algorithm performs a number of generations. The GA should stop when the optimal solution is reached, or other stopping criteria is fulfilled. Typically two criterias are established: running a maximum number of generations or the algorithm stops when there are not changes in the population. Until the stop condition is not fulfilled, the algorithm performs the following steps. A selection operator selects a number of individuals in order to generate offsprings. Selection is usually based on the fitness of the individuals, where fitter individuals have more chances of being selected, simulating the concept of survival of the fittest. Offsprings are generated with the application of crossover and mutation. The crossover and mutation operator are applied according to a given probability. Offsprings are then evaluated and a new population is created. Finally, the best solution is extracted from the population.

3.2 Components of EC

In the following we address various aspects of EC, such as encoding and evaluation of the individuals and genetic operators (selection, crossover and mutation).

3.2.1 Encoding

Two basic concepts in genomics must be described: genotype and phenotype, as they are also suitable in the EC context. The genotype represents all information contained in the chromosome. Such information may manifest in the individual or not. The phenotype refers to the expression of the genotype in addition to the environmental influence. In ECs, the genotype has the same meaning, it is an ensemble of genes which constitute a chromosome. The phenotype would represent the decoding (translation and expression) of the information contained in the genotype. This information could be 1's and 0's chains or another type of symbols. In a biological sense, expressed phenotype in living beings is the result of the interaction between the genotype and the environment, however in ECs, such interaction is not considered, and phenotype is only a simple process of decoding of the genotype.

EAs require that the set of variables of the problem are encoded into a chromosome. Each chromosome has several genes that correspond to the

respective parameters of the problem. These genes must be encoded, *e.g.*, in a string of symbols (numbers or letters), to be computationally treated.

The representation scheme defines how the chromosomes correspond to the solutions of the problem. To design the scheme of representation, it is necessary to establish the parameters that identify the solutions, and then encode these parameters into a chromosome. For example, in GA, the three most commonly used representation are binary, integer and real representation where each gene corresponds to a binary, integer or real number respectively.

3.2.2 Evaluation

In order to evaluate individuals, a fitness function is used. This function provides a numerical value that can be used in order to assess the quality of an individual.

The evaluation of the individuals quantifies the aptitude of each individual as a solution of the problem, and determines the probability of selection. A good definition of this function is essential for a correct functioning of the algorithm, because it provides the mechanism by which the population evolves toward fitter chromosomes. The fitness value assigned to an individual must reflect the quality of the solution represented by the individual, where better fitness values are assigned to better solutions.

3.2.3 Selection

At each generation, individuals are selected, using a selection operator in order to generate offsprings. The selection operator selects a determined individual based on its fitness values. The selection operator is the responsible for transmitting and preserving those features of the solutions that are considered valuable throughout the generations. For this, fitter individuals are more likely to be selected and thus to reproduce. However, it is also necessary to include a random factor that allows the reproduction of not very well adapted individuals. This is due to the fact that this type of individuals could contain useful information for future generations. Moreover, this can help to maintain a certain diversity in the population. Different selection operators have been proposed:

- Ranking selection: the chromosomes are sorted according to their values of adaptation. Then, the first m individuals are selected for reproduction. Thus, the chances of an individual of being selected, depends only on its relative position to other individuals and not on the absolute value of fitness.

- Roulette-wheel or Fitness proportionate selection: a probability is assigned to each individual depending on its score according to

equation 3.1.

$$pr(h_j) = \frac{fitness(h_j)}{\sum_{i=1}^{n} fitness(h_i)} \tag{3.1}$$

where $pr(h_j)$ represents the probability of individual h_j to be selected, $fitness(h_j)$ indicates the fitness value of this individual, and n represents the size of the population. The higher the fitness, the more chances of being selected.

- Tournament selection N/K: we randomly select N individuals of the population. From these N individuals we choice K individuals with the best scores.

3.2.4 Crossover

The crossover operator allows an exploration of all information stored in the population and combines it to create better individuals. This operator is responsible for transferring genetic material from one generation to the next. Within the usual methods, we emphasize the following ones:

- One-point crossover: this is the simplest method of crossover. A position of the chromosome of the parent individuals is selected (random cut). Then, the genes are exchanged on both sides of this position. Two new descendants are generated (Figure 3.1).

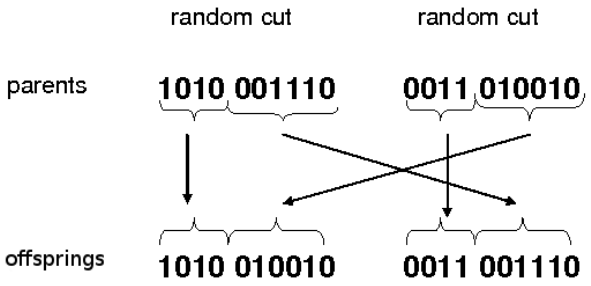


Figure 3.1: One-point crossover.

- N-point crossover: this type of method is a generalization of the previous one. Various positions (N) are randomly selected in the chromosomes of the parents and the genes are exchanged on both sides of these positions. Figure 3.2 represents a N-point crossover operator where $N = 2$.

- Uniform crossover: this operator evaluates each gene in the parents for exchange with a probability of 0.5. Thus genes are randomly copied from the first or from the second parent.

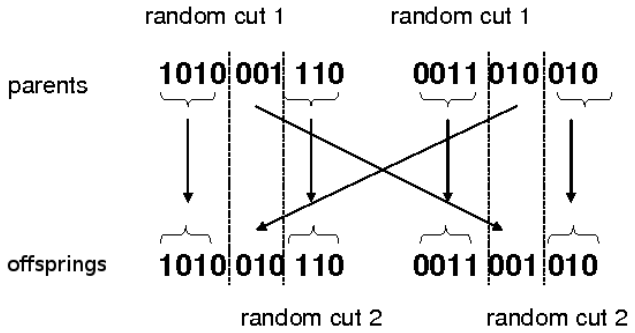


Figure 3.2: Two-point crossover.

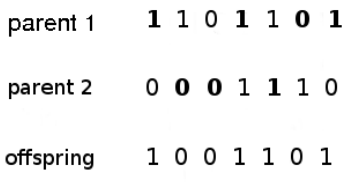


Figure 3.3: Uniform crossover.

- Arithmetic crossover: some arithmetic operation (sum, arithmetic mean...) is performed to generate a new offspring.
- BLX-α crossover: this operator creates a new offspring, where the values of the genes are mutated within an interval delimited by the maximum and minimum values of the two parent individuals for the same gene. A α value is also selected to calculate this interval. This parameter must be higher than or equal to 0. This crossover operator can be seen as a linear combination of the two parents.

3.2.5 Mutation

The mutation of chromosomes (along with the generation of the initial population) is responsible for maintaining genetic diversity of population. The mutation is implemented by a mutation operator. This operator allows the exploration of the search space. The mutation operator works at the block level within the chromosomes, making random changes. First the mutation operator could randomly select a gene, and then mutate it. The

nature of the change depends on the composition of the blocks of the chromosomes. If each block is a bit (binary encoding), the only possible change is to invert its value. If the blocks are real numbers, the modification could be the addition or subtraction of a small random value. Mutation of the population guarantees that the search does not get stuck into local optima.

3.3 Multi-objective Optimization

Generally, EAs try to solve single-objective optimization problem. This means that all the objectives that have to be optimized are combined into a single fitness function, that guides the evolutionary search performed by the EA. The single objective approach works well when there is only one objective to optimize, or all the objectives are not in conflict with each other. Often, however, a problem requires to optimize different objectives at the same time, and it can be difficult to combine them in a single function. Moreover, when the search space is highly complex, it is often impossible to find a single optimal solution. Instead, one is usually more interested in finding a set of solutions that presents a good compromise among all the objectives. Rather than combining the multiple objectives into a single fitness function, a better approach to find this optimal set is to optimize the objectives separately, i.e., treat the problem as a multi-objective problem. EAs are particularly suited for tackling multi-objective optimization problems, mainly due to the population-based nature of EAs.

A Multi-objective optimization problem requires the optimization of a set of objectives, usually in conflict with each other. The existence of multiple objectives poses a fundamental difference with the single objective problems: typically there will not be a single solution, but a set of solutions that can present different clashes between the values of the objectives to optimize. We can define a multi-objective optimization problem in this way: let $(f_1(x), f_2(x)...f_n(x))$ be a set of functions to be optimized, where $x = (x_1, ..., x_p)$ is a vector of decision variables belonging to a universe X and $f_i(x)$ is an arbitrary linear or non-linear function, $1 \leq i \leq n$. Therefore, the problem consists of finding the x that provides the best compromise value for all $f_i(x)$.

To solve the above problem, we should define some criteria to determine which solutions are considered of good quality and which are not. To this aim, the concept of dominance is generally used. A solution x is said to be not dominated *iff* there is not another solution y such that: $f_i(y) \leq f_i(x)$ for all i and $f_i(y) < f_i(x)$ for some i, where $1 \leq i \leq n$. From this, it follows that the best solutions are those that are not dominated. Such solutions form a set called Pareto front.

In this thesis, the PSP problem is addressed as a multi-objective problem,

as will be described in chapter 5.

3.4 Multi-objective Evolutionary Algorithms

A Multi-objective Evolutionary Algorithm (MOEA) should be designed to achieve two purposes simultaneously: to achieve good approximations to the Pareto front and to maintain the diversity of solutions, in order to adequately sample the solution space and not converge to a unique solution.

The evolutionary mechanisms of EAs can achieve the first purpose. For preserving the diversity, MOEAs use techniques like niches, sharing, crowding or similar, traditionally used by EAs in multimodal function optimization.

In the following we list the most popular MOEAs:

MOGA (Multi-objective Optimization GA): in MOGA [Fonseca *et al.*, 1993], a range is assigned to each individual of the population. This range determines the order criterion for the selection. The range is assigned according to a non-dominance criterion: if x_i is a non-dominated individual then range(x_i)=1. Otherwise, range(x_i) = 1 + (number of individuals that dominates x_i)

NPGA (Niched Pareto Genetic Algorithm): NPGA [Horn and Nafpliotis, 1993] is based on combining the tournament selection operator and the concept of Pareto Dominance. Given two selected individuals, a random subset of size T of the population is selected. If one of the selected individual is dominated by any member of the set and the other not, then the latter is considered the winner of tournament. If both individuals are dominated, the result of tournament is decided by the method of proportion. The individual with fewer chromosomes in the niche is selected. In order to determine a niche, sharing is used. Sharing establishes the same fitness value of points belonging to a same niche. This strategy is used in order to maintain diversity in the population.

NSGA (Non-dominated Sorting GA): NSGA [Srinivas *et al.*, 1995] sorts the population according to levels of non-dominance (ranking Pareto fronts). The method does not work with the fitness value, but with a constant dummy fitness which is established from the dominance ranking position. To maintain the diversity, the use of count of niches is adopted. If there are many individuals sharing the same niche (or neighborhood), the fitness is proportionally decreased according to the number of individuals sharing the same niche.

A new evolutionary model which uses an external population has been designed. This external population stores the non-dominated solutions encountered during the search.

SPEA (Strength Pareto Evolutionary Algorithms):
SPEA [Ziztler and Thiele, 1998] uses a external population with non-dominated solutions, which is obtained at the end of every generation. The algorithm is based on the strength concept. The strength of an individual x is given by the number of individuals that x dominates. The fitness of an individual is proportional to its strength. The use of binary tournament is another feature of this method.

PAES (Pareto Achieved Evolution Strategy):
PAES [Corne *et al.*, 2000] consists in an evolutionary strategy (1 + 1), *i.e*, a single parent produces one child, in combination with an external file that stores some of the non-dominated solutions found previously. Each mutated individual x is compared with individuals in the external file. If x is not dominated by the individuals contained in the external file, then algorithm selects which dominated individuals leave the external file and insert x in the file. This strategy helps to maintain the diversity of population and uniformly distributes non-dominated produced solutions.

SPEA-II: SPEA-II [Ziztler *et al.*, 2001] has three main differences to original SPEA: it incorporates a new strategy to assign the fitness to an individual x, which takes into account the number of individuals that dominate x, and the number of individuals that are dominated by x. This method also uses an estimating technique of neighboring density, which guides the search in a more efficiently way. Finally, it uses a truncation schema of the external non-dominated population to ensure the preservation of boundary solutions.

NSGA-II: NSGA-II [Deb *et al.*, 2002] initially creates a population of parents. The population is sorted according to levels of non-dominance (ranking Pareto fronts). Each solution is then assigned a fitness value according to their level of non-dominance (1 is the best level). Tournament selection, crossover and mutation are used to create the offspring population. NSGA-II includes the use of elitism, is much more computationally efficient than NSGA.

OMOEA (Orthogonal Multi-objective Evolutionary Algorithm):
OMOEA [Zeng *et al.*, 2004] is proposed for multi-objective optimization problems (MOPs) with constraints. Firstly, these constraints are taken into account determining the Pareto dominance. As a result, a strict partial-ordered relation is obtained, and feasibility is not considered later in the selection process. An original niche evolves first, and splits into a group of sub-niches. Then every sub-niche repeats the above process This algorithm is superior to other MOEAs, such as

NSGAII or SPEA2, in terms of the precision, quantity and distribution of solutions.

AMGA (Archive-based Micro Genetic Algorithm):
AMGA [Tiwari *et al.*, 2008] employs a new kind of selection procedure which benefits from the search history of the algorithm and attempts to minimize the number of function evaluations required to achieve the desired convergence. The proposed algorithm works with a very small population size and maintains an archive of best and diverse solutions obtained so as to report a large number of non-dominated solutions at the end of the simulation.

3.5 Summary

In this chapter we provide a brief introduction of evolutionary computation. We have summarized the basic components of evolutionary computation, such as encoding and evaluation, as well as the basic evolutionary operators, such as selection, crossover and mutation. On the other hand, we have also exposed the concept of multi-objective optimization. An explanation of the use of evolutionary algorithms for solving multi-objective problems is also provided, as well as a brief description of the most representative multi-objective evolutionary algorithms.

Part II

State of the art

Chapter 4

State of the art

In this chapter, we present the state of the art of protein structure prediction methods based on soft computing techniques, lazy methods and statistical approaches.

Furthermore, protein structure prediction methods can be further classified in: homology-based methods, threading methods and *ab initio* methods. As its name suggests, homology-based methods predict protein structures based on sequence homology with known structures. The principle behind this is that if two proteins share a high degree of similarity in their sequences, then they should have similar 3D structures. Threading, or sequence-structure alignment methods or fold recognition methods, try to determine the structure of a new protein sequence by finding its best "fit" to some fold in a library of structures. Fold recognition methods are motivated by the notion that evolution conserves structure rather than the sequence. *Ab initio* methods attempt to generate models of proteins solely based on sequence information and without the aid of known protein structures. The goal is to predict the structure of a protein based entirely on the laws of physics and chemistry. Our proposal lies in this last category.

In the following, we first address techniques for the prediction of secondary structure, and then we will focus our attention on tertiary structure prediction methods.

4.1 Secondary structure prediction methods

The problem of protein secondary structure (SS) prediction consists in predicting the location of α-helices, β-sheets and turns within a sequence of amino acids without any knowledge of the tertiary structure of the protein. Despite the fact that SS prediction is a far less active area than a decade ago, accurate SS prediction is very useful to biologists. Moreover, it is also an essential component of tertiary structure prediction, which is far from being solved and continues to be a highly active area of research. Accurate

prediction of protein SS is also essential for accurate sequence alignment, three-dimensional structure modeling, and function prediction. Hence improving the accuracy of SS prediction is essential for future developments throughout the field of proteomics.

Several standard quality measures are used to evaluate the accuracy of secondary structure methods, being Q3, Q8 and SOV [Rost *et al.*, 1994, Venclovas *et al.*, 1999] the most commonly used. Q3 represents the overall three-state accuracy, which calculates the average predictive accuracy of the system for each conformational state: helix, strand and loops (H, E and C). On the other hand, Q8 evaluates the accuracy for the eight secondary conformational states defined by the Dictionary of Protein Secondary Structure (DSSP) method [Kabsch and Sander, 1983]. These conformational states are 3_{10}-helix, α-helix, π-helix, hydrogen bonded turn, extended β-strand, β-bridge, bend and coil which are represented by G, H, I, T, E, B, S and C, respectively (see Glossary for more information). These eight states are commonly grouped into the three already seen larger classes: helix (G, H and I), strand (E and B) and loop (T, S and C). Another important measure is SOV (segment overlap score). This measure is based on the average overlap between the observed and predicted segment of a determined secondary structure.

4.1.1 Statistical approaches

We first present the state of the art of statistical approaches to SS prediction.

These methods calculate amino acids propensities and determine if the amino acid belongs to a given type of secondary structure (α-helix, β-sheets, and turns). Chou-Fasman [Chou *et al.*, 1974] proposed a method where, if the calculated propensity is higher than one, it means that the residue is likely to be found in the corresponding secondary structure. These propensities (scores) are calculated using the amino acid appearing frequencies. A set of heuristic rules are also used in order to predict the structure type. This method uses a table of conformational propensities of the amino acid. On the other hand, GOR [Garnier *et al.*, 1978] also uses a table that contains information content about 17 amino acids positions ($i-8, i+8$, where i is the target amino acid). This fixed residue window was later used in most of prediction algorithms (Figure 4.1). An amino acid will be part of an α-helix if it is surrounded by residues that are predisposed to bend in the form of the α-helix. Another early work in SS prediction is detailed in [Lim, 1974]. This algorithm predicts α-helical and β-structural regions using a system of complex rules. These rules impose a set of conditions on the hydrophobicity property of residues and formation of secondary structures. This method is based on chemical side-chain properties.

These pioneering algorithms constituted the basis of some later works.

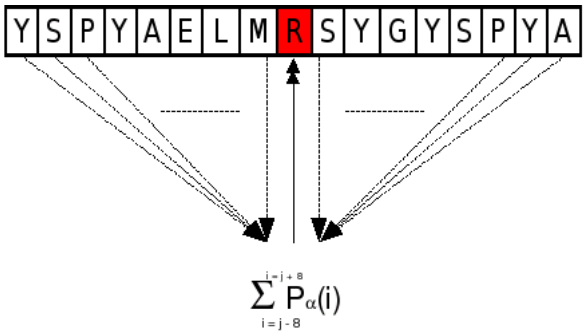


Figure 4.1: GOR method representation.

Chou-Fasman parameters are used in [Deleage and Roux, 1987]. This algorithm also uses the prediction of the class of the proteins from their amino acid composition. The two predictions are then optimized over a SS database to provide the final prediction. GOR is improved in several works. [Kloczkowski *et al.*, 2002] improves the GOR algorithm by using the evolutionary information provided by MSA and introducing a variable size window. [Sen *et al.*, 2005] develop the GOR V algorithm. The method combines information theory, Bayesian statistics and evolutionary information. In [Biou *et al.*, 1988], three SS prediction methods are combined: GOR method, the homologue method and the bit pattern method based on hydrophilic/hydrophobic residue patterns. The method is tested with 67 protein sequences and achieved an accuracy of 75% for helices and sheets.

Bayesian models/classifiers calculate the probability that an example belongs to a class depending on a set of variables. In this case, an amino acid represents an example, and the class corresponds to a SS conformation. The classifier learns from the training set the conditional probability of each variable given a class. Naïve Bayes classifier is the simplest Bayesian classifier and it assumes conditional independence of predictive variables for a given class. Two proposals convert SS prediction into a general Bayesian inference problem. [Schmidler *et al.*, 2000] presents a statistical method to predict the sequence-structure relationships in terms of structural segments. Several structure features, such as helical capping signals, side chain correlations, and segment length distributions, are taken into account. The method calculates probability distributions over all possible segmentations of the sequence. [Robles *et al.*, 2004] uses several multi-classifiers based on Bayesian networks, validated on 9 different datasets.

Other methods use multiple sequence alignment (MSA) profiles to improve SS prediction. DSC method, described in [King *et al.*, 1996], is based on residue conformational propensities, mean hydrophobicity

moments[1] and position of insertions and deletions in aligned homologous sequence. From a single protein sequence or a MSA, Jpred3 method [Cole *et al.*, 2008] provides alignment profiles from which predictions of SS and solvent accessibility (SA)[2] are made. The system synchronizes with major updates to SCOP [Murzin *et al.*, 1995] and UniProt (a protein sequence and functional information database) [Bairoch *et al.*, 2005] and so ensures that Jpred 3 will maintain high-accuracy predictions. The technique proposed in [Francesco *et al.*, 1996] analyzes the relationship between the location of secondary structural elements, gaps, and variable residue positions in MSA to improve the SS prediction, using the Quadratic-Logistic method (described in [Munson *et al.*, 1994]) with profiles. [Mehta *et al.*, 1995] uses MSAs of substituted but structurally related proteins. The algorithm calculates residue exchange weight matrices for the three structures (helix, sheet and coil) for a total of 2,500 protein sequences from 70 protein families. [Zvelebil *et al.*, 1987] approach is based on SS propensities for aligned residues and on the observation that insertions and high sequence variability tend to occur in loop regions between SS. Accordingly, the algorithm first aligns a family of sequences and obtains a value for the extent of sequence conservation at each position. This value modifies a prediction on the averaged sequence to yield the improved prediction. [Levin *et al.*, 1986] method follows the hypothesis that short homologous sequences of amino acids have the same SS trends. The method uses a similarity matrix which assigns a sequence similarity score between any two sequences.

Hidden Markov models (HMM) [Baum and Petrie, 1966] have also been used to predict secondary structure. Once a multiple sequence alignment profile is built using short segments of similar sequences with known structure, HMMs are generated in a structure context that is then used to predict the structure of the protein. Several methods employs HMM for the SS prediction. HMMSTR [Bystroff *et al.*, 2000], is based on a library of sequence-structure motifs. [Asai *et al.*, 1993] uses output probabilities from HMMs to predict the SS of the sequences. The authors test this prediction system on 100 sequences from a public database (Brookhaven PDB). A hidden semi-Markov model (HSMM) is also defined in [Aydin *et al.*, 2006]. This method considers statistically significant amino acid correlation at structural segment borders and tries to refine estimations of HSMM parameters using an iterative training method. PASSML [Livs *et al.*, 1998] provides a reconstruction of phylogenies and prediction of secondary structure from aligned amino acid sequences. This approach is based on a Markov process with discrete states in continuous time, and the organization of structure along protein sequences is described by a HMM. Eight categories

[1]The vectorial sum of all the hydrophobicity indices, divided by the number of residues.
[2]Represents the solvent exposed surface area of a residue in a protein.

of structural environment are distinguished according to solvent accessibility.

Another method which uses statistical approach is proposed in [Geourjon and Deleage, 1994]. This method, called SOPM, consists of three phases: first the database is divided in sub-databases of protein sequences and their known SS. This is done by making binary comparisons of all protein sequences and taking into account the prediction of structural classes of proteins. Then, each protein of the sub-database is submitted to a SS prediction algorithm based on sequence similarity. Finally, SOPM determines the predictive parameters that optimize the prediction quality on the whole sub-database.

A Monte Carlo simulated annealing procedure is described in [Simons *et al.*, 1999]. The scoring function of the method consists of sequence-dependent terms representing hydrophobic burial and specific pair interactions such as electrostatics and disulfide bonding.

Finally, a protein machine induction system, called PROMIS, is developed by [Wood *et al.*, 2005]. This method generates rules to predict secondary structure from a known primary structure. Such rules are based on chemical properties of the residues, *e.g.*, amphipathic[3] nature of α-helices.

A summary, in chronological order, of the statistical methods for SS prediction is shown in table 4.1. The first and second column indicate the name of the method and its reference respectively. The third column represents the available accuracy achieved by the method. The forth column indicates the size of the data set of proteins, while the fifth column shows main characteristics of the different algorithms.

[3]Pertains to a molecule containing both hydrophobic and hydrophilic regions in its structure.

Method	Reference	Q3 Acc.(%)	Dataset size	Description
	[Chou *et al.*, 1974]	50	15	Propensities
	[Lim, 1974]	60	25	Complex rules
GOR	[Garnier *et al.*, 1978]	55	26	Sliding window of 17 residues
	[Levin *et al.*, 1986]	62.2	61	Similarity matrix
	[Deleage and Roux, 1987]	61.3	59	Prediction of protein class
	[Zvelebil *et al.*, 1987]	66.1	11	Sequence alignments
	[Biou *et al.*, 1988]	75.0	67	Combination of methods (GOR, Homologue and HP)
	[Asai *et al.*, 1993]	66.0	100	HMMs
SOPM	[Geourjon and Deleage, 1994]	69.0	239	Sequence similarities
	[Mehta *et al.*, 1995]	72.2	2500	MSA and residue exchange weight matrices
	[King *et al.*, 1996]	70.1	126	Multiple alignments
	[Francesco *et al.*, 1996]	62.4	95	MSA, Quadratic Logistic method
PASSML	[Livs *et al.*, 1998]	63.0	207	Markov model, phylogenies and SS prediction
ROSETTA	[Simons *et al.*, 1999]	6.4 (RMSD)	CASP3	Monte Carlo simulated annealing
	[Schmidler *et al.*, 2000]	68.8	452	Bayesian inference, structure features
GOR V	[Kloczkowski *et al.*, 2002]	73.5	513	MSA and variable size window
HMMSTR	[Bystroff *et al.*, 2000]	74.3	61	HMM
	[Robles *et al.*, 2004]	81.65	126	Bayesian networks
GOR V	[Sen *et al.*, 2005]	73.5	513	Bayesian statistics and evolutionary information
PROMIS	[Wood *et al.*, 2005]	60.0	43	Induction system, AA properties, predictive rules
BSPSS	[Aydin *et al.*, 2006]	72.0	2720	HSMM, AA correlation
JPRED 3	[Cole *et al.*, 2008]	81.5	239	MSA, SS and SA predictions

Table 4.1: Resume of statistical methods for secondary structure prediction.

4.1.2 Neural networks methods

The basic principle behind an artificial neural network (ANN), is that the ANN can be trained to recognize patterns of known amino acid structures and use them to differentiate between different types of protein secondary structures or predict protein contact maps.

In [Qian *et al.*, 1988], an ANN receives as input a 17-dimensional vector which represents a segment of a protein sequence of 17 amino acids. The output layer corresponds to a 3D vector which represents the prediction of different types of protein secondary structure (alpha, beta and coil), as shown in figure 4.2. [Kneller *et al.*, 1990] enhance this method with the addition of periodic sequence information to the neural network. They also divide their database into all-alpha, all-beta and other (alpha/beta). This method also introduces neural network units that detect periodicities in the input sequence and tertiary structural class. To validate the predicted structure, a scheme for employing neural networks unit is proposed. The proposal predicts the mapping between primary sequence and SS. A pioneering method is also described in [Holley *et al.*, 1989]. This method trains the network to recognize the relation between SS and sequences, and calculates a numerical measure of helix and sheet tendency for each residue.

The employment of evolutionary information is one of the most recurrent feature of ANN methods. Information that is often used is the MSAs or the position specific scoring matrices (PSSMs) generated by PSIBLAST. [Rost and Sander, 1993] use a MSA as the input of the network. The method stores an evolutionary profile for each residue. If a multiple alignment is not given, program will generate it. [Petersen *et al.*, 2000] use evolutionary profiles from PSSMs. A balloting procedure estimates probabilities corresponding to each SS class (H, E, C). [Jones, 1999] develop a two-stage neural network based on the PSSM generated by PSI-BLAST. This method, called PSIPRED, is evaluated by blind testing in CASP3. [Rost and Sander, 1994] method uses evolutionary information as MSA as input of the ANN. The position-specific conservation weight is used as part of the input. This method also takes into account the number of insertions and deletions which reduces the tendency for overprediction and increases overall accuracy. [Riis *et al.*, 1996] method uses neural networks and MSAs. Amino acid properties and propensities are also used as inputs of ANN.

The combination of ANNs and HMM have also been effective in various methods. JNET [Cuff *et al.*, 1998], uses MSA as input data as well as HMM profiles. [Lin *et al.*, 2005] employs this combination to optimize output data.

Two methods in the bibliography use Bidirectional Recurrent Neural Networks (BRNNs). SSPro-SSPro8 [Pollastri *et al.*, 2002], uses Position-Specific Iterated Basic Local Alignment Search Tool (PSIBLAST) [Altschul *et al.*, 1997] to determine the input profiles. Input data is classified in three (SSPro) or eight different classes (SSPro8) by DSSP program

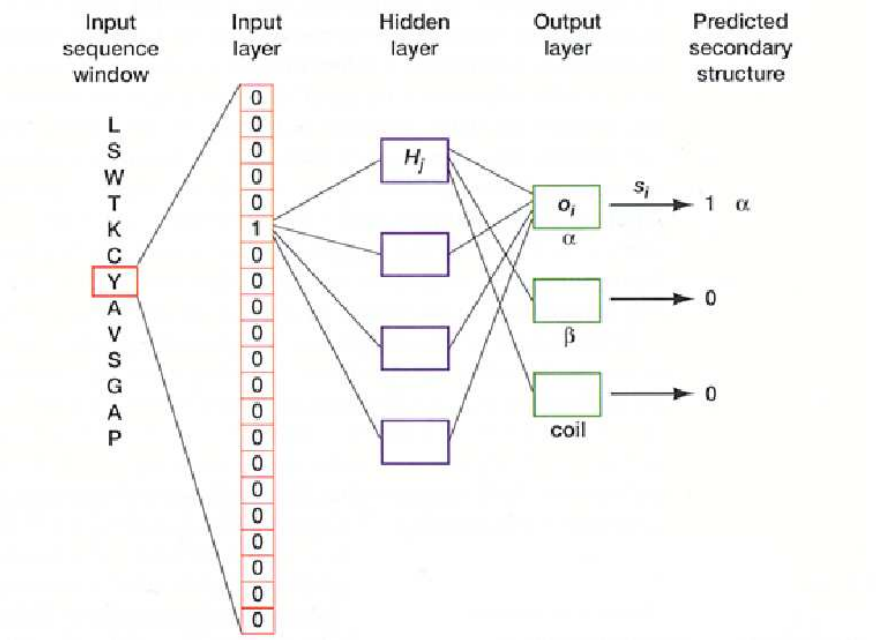


Figure 4.2: Neural network schema detailed in [Qian *et al.*, 1988].

to improve prediction accuracy. BRNNs improve some limitations of simple feed-forward networks like small fixed-length input window. This architecture is a 2D generalization of the BIOHMMs (bi-directional input-output HMMs) and BRNNs. Porter [Pollastri and Mclysaght, 2005], is a three-classes SS predictor. The main features of the method are: an accurate coding of input profiles obtained from MSA generated by PSIBLAST, a second stage filtering by recurrent neural networks, the incorporation of long range information and large-scale ensembles of predictors.

[Chandonia *et al.*, 1999, Muskal *et al.*, 1992] develop methods based on two neural networks. [Chandonia *et al.*, 1999] predicts SS and structural protein class. The method produces an estimate of the probability of finding each type of secondary structure at every position in the sequence. [Muskal *et al.*, 1992] also uses information about the protein amino acid composition, molecular weight and heme presence.

Torsion or dihedral angles represent the position of the atoms of an amino acid chain. This representation is employed in [Wood *et al.*, 2005, Faraggi *et al.*, 2012]. DESTRUCT method, detailed in [Wood *et al.*, 2005], is based on an iterative set of cascade–correlation neural networks which predict both SS and dihedral angles. [Faraggi *et al.*, 2012] develop a multi-step neural network algorithm, called SPINE X, by coupling SS and SA and backbone torsion angles in an iterative way.

The method proposed in [Katzman *et al.*, 2008], allows an arbitrary

number of hidden layers, units per layer, window sizes and local structure alphabets.

The method detailed in [Ouali *et al.*, 2000] is constituted by different types of classifiers using neural networks and linear discrimination. Some features of this method are the use of a local window and resampling techniques.

Finally, an ANN based multi level classifier is designed for predicting protein SS in [Bordoloi *et al.*, 2012]. ANNs are trained to make them capable of recognizing SS in a sequence of amino acids.

Neural networks provide a high degree of flexibility. Besides encoded input vectors of pair of amino acids, we may include neurons with additional information, *e.g.* sequence length, hydrophobicity values of the environment or evolutionary information. On the other hand, neural networks have certain limitations, *e.g.*, restriction in the encoding of input data, the use of appropriate parameters of the ANN and overfitting.

A resume, in chronological order, of the neural network methods for SS prediction is shown in table 4.2.

Method	Reference	Q3 Acc.(%)	Dataset size	Description
	[Qian *et al.*, 1988]	63	15	ANN, 17-dimensional method vector
	[Holley *et al.*, 1989]	79	48	Measure of SS tendency
	[Kneller *et al.*, 1990]	71	129	Periodic sequence information
	[Muskal *et al.*, 1992]	5.6 (RMSD)	14	AA properties
	[Rost and Sander, 1993]	72	26	Multiple sequence alignment as input
	[Rost and Sander, 1994]	71.6	126	MSA, PSSM
	[Riis *et al.*, 1996]	71.3	126	AA properties and propensities
JNET	[Cuff *et al.*, 1998]	73	61	MSA and HMM profiles as inputs
	[Chandonia *et al.*, 1999]	74.6	681	Probabilities
PSIPRED	[Jones, 1999]	78.3	187	2-stage ANN, PSSM
	[McGuffin *et al.*, 2000]	60	25	Profiles generated with PSI-Blast
	[Ouali *et al.*, 2000]	76.7	496	ANN and Linear discrimination
	[Petersen *et al.*, 2000]	74.3	61	PSSM
BRNN-PRED	[Pollastri *et al.*, 2002]	45-65	1520	BRNNs, PSIBLAST and DSSP
YASPIN	[Lin *et al.*, 2005]	50	15	HMM
Porter	[Pollastri and Mclysaght, 2005]	79 (Q8)	Rost	BRNN, long range information
DESTRUCT	[Wood *et al.*, 2005]	80.7	CASP4	SS and dihedral angles prediction
PREDICT-2ND	[Katzman *et al.*, 2008]	55	26	Local structure alphabet
	[Bordoloi *et al.*, 2012]	-	4	ANN-based multi level classifier
SPINE X	[Faraggi *et al.*, 2012]	82.0	2,640	Multi-step ANN, SA and torsion angles

Table 4.2: Resume of neural networks methods for secondary structure prediction.

4.1.3 Support vector machines methods

Support Vector Machines (SVMs) are based on the transformation of the input space into a feature space of higher dimensionality. SVMs techniques then build a hyperplane, or a set of hyperplanes, in this space trying to maximize the margin between each different classes. The function that performs the transformation of the space is called kernel function. SVMs are used as a machine learning tool to predict secondary structure contacts from the primary sequence. The first approach based on SVMs was introduced in [Hua and Sun, 2001].

Evolutionary information (PSSMs) is usually employed by SVM methods. For instance, [Kim and Park, 2003] introduce a technique called SVMPsi tool. They incorporate a new tertiary classifier system and an optimization strategy for maximizing the Q3 measure. The strategy proposed in [Ward *et al.*, 2003] presents the following main features: the use of PSSM from three iterations of a PSI-BLAST search, a reduction of the eight states provided by the DSSP program, and a "One-versus-all" method that combines outputs from binary classifiers into a multiple class prediction. [Karypis, 2006] introduce an algorithm called YASSPP. Results obtained were improved with a better kernel and combining position-specific and non-position-specific information. Finally, [Guo *et al.*, 2004] combines PSSM profiles with the SVM analysis. Prediction results are provided from the second SVM layer output.

Three methods propose novel techniques to predict SS based on physico-chemical properties of amino acids. The algorithm proposed in [Chatterjee *et al.*, 2011], includes multiclass SVMs as classifiers for three different structural conformations (helix, sheet and coil). PSSMs obtained from PSI-BLAST and five physico-chemical properties of amino acids are fed into SVMs as features for sequence-to-structure prediction. The obtained confidence values are then used for performing structure-to-structure prediction. Training and test set are formed by RS126 dataset and CASP 9 target proteins, respectively. [Yang *et al.*, 2011] and [Qu *et al.*, 2012] proposals consists of three parts: a mixed-modal SVM (MMS) module, a modified Knowledge Discovery in Databases (KDD) process, and a mixed-modal back propagation neural network (MMBP) module.

[Chen *et al.*, 2009] proposes an approach based on Chou-Fasman parameters for SS prediction. This method employs a regressing system and adopts a different pseudo amino acid composition called PseAAC.

The methodology detailed in [Hu *et al.*, 2004], is based on SVM and several encoding schemes, such as orthogonal matrix, hydrophobicity matrix, and BLOSUM 62 substitution matrix. A varying window length for the six SVM binary classifiers is another feature of the algorithm.

A summary of the SVM methods for SS prediction is shown in table 4.3.

4.1.4 Nearest neighbors-based methods

The basic idea of the nearest-neighbor (NN) approach is to use the labels of examples closely related to a test instance to determine its label. In PSP problem, NN methods operate by matching segments of the protein sequence with segments within a database of known protein structures, and making a prediction based on the observed structures of the best matches, according to a distance measure.

Several proposals combine NN approaches with other methods. [Yi and Lander, 1993] is one of the earliest methods based on the NN scheme and is called NN Secondary Structure Prediction (NNSSP). This method combines a NN and a neural network approach. Protein sequences are represented as multiple alignments. Segment similarity score is calculated based on a scoring table derived from local structural environment [Bowie *et al.*, 1991]. [Frishman and Argos, 1996] proposes a technique, called PREDATOR, that combines a NN and a statistical approach. This method uses additional propensities scores of hydrogen bonds peptides. The likelihood for hydrogen bonds in α-helix and β-sheets is calculated for each residue. Another prediction factor is the similarity of the sequence segment to the aligned segment. All values are calculated over a four residue window. A set of rules are used to predict conformational state of each residue, taking into account previous propensity values.

The alignment of sequences generated by PSIBLAST also constitutes a valuable tool to obtain feature vectors for NN methods. Four methods use MSAs. SSPAL method, proposed in [Salamov *et al.*, 1997], analyzes aligned sequence segments of variable length. SSPAL calculates K-best non overlapping local alignments of a query sequences with N sequences of known structure. For a given query, $N \times K$ local alignments are produced. A score based on conformation state is computed for each position in the query sentence from these alignments. [Joo *et al.*, 2004] introduces a method that applies PSIBLAST to protein sequences with known secondary structures to construct pattern databases. For each protein sequence, PSIBLAST generates a profile that defines patterns for amino acid residues and their local sequence environments. The approach proposed in [Kim *et al.*, 2006], develops a parallel algorithm based on the fuzzy k-NN method, that uses evolutionary profile obtained from PSIBLAST as the feature vectors. Finally, [Zhou *et al.*, 2010] introduces a strategy called Frag1D, which is based on fragment matching. The basic idea of the method is the same as the NN approach. Both approaches predict the secondary-structure state of the central residue of a test segment based on the secondary structure of high-scoring candidate segments from proteins with known structures. In this method, candidate segments are selected by a profile-profile score derived from PICASSO score [Mittleman *et al.*, 2003] and the profile is created by taking the advantage of PSIBLAST.

[Kim, 2004] proposes a protein β-turn predictor which incorporates a filter that uses predicted protein SS information from PSI-PRED. This work modifies the traditional β-turn prediction from k-NN in order to take into account the unbalanced ratio of the natural occurrence of β-turns and non-β-turns. The proposed method has three β-turn prediction schemes, all of which consist of two stages of prediction. The first stage is a k-NN method while the second stage is a filter, which refines the prediction taking into account correlations amongst residues.

[Leng *et al.*, 1993] presents a case-based reasoning (CBR) system. In this approach, they considered several different measures from the biology literature for determining similarity between proteins. Once these proteins are found, their approach involves decomposing the novel sequence into smaller segments. Each amino acid in the sequence is assigned a class (α-helix, β-strand, or coil) by applying a weighted sum calculated from known protein structures.

Finally, SIMPA96 scheme, [Levin *et al.*, 1997], is based on an updated version of NN method. This method includes a large protein dataset and BLOSUM 62 substitution matrix.

Table 4.4 summarize the NN methods for SS prediction and their respective results and data sets employed.

Method	Reference	Q3 Acc.(%)	Dataset size	Description
	[Hua and Sun, 2001]	73.5	513	First SVM approach
SVMPsi	[Kim and Park, 2003]	78.0	480	PSSM
	[Ward *et al.*, 2003]	77.0	121	PSSM, eight-class prediction
PMSVM	[Guo *et al.*, 2004]	74.0	396	PSSM, PSIBLAST profiles
	[Hu *et al.*, 2004]	78.8	126	Encoding schemes (BLOSUM62)
YASSPP	[Karypis, 2006]	79.0	129	PSSM
PSP-MCSVM	[Chatterjee *et al.*, 2011]	71.0	126,CASP9	PSSM and AA properties
	[Yang *et al.*, 2011, Qu *et al.*, 2012]	85.6	CASP8	Mixed-modal SVM, KDD and MMBP module

Table 4.3: Resume of SVM methods for secondary structure prediction.

Method	Reference	Q3 Acc.(%)	Dataset size	Description
	[Leng *et al.*, 1993]	68.2	106	CBR system
NNSSP	[Yi and Lander, 1993]	68.0	126	Combination of NN and ANN approach
PREDATOR	[Frishman and Argos, 1996]	68.0	125	Combines NN and statistical approach
SIMPA96	[Levin *et al.*, 1997]	72.8	111	BLOSUM62 matrix
SSPAL	[Salamov *et al.*, 1997]	71.2	124	Multiple sequence alignment as input
PREDICT	[Joo *et al.*, 2004]	78.8, 77.4 (SOV)	513	Profiles generated with PSIBLAST
	[Kim, 2004]	46.5	426	Beta-turn prediction, PSIPRED
	[Kim *et al.*, 2006]	71.8	60	Fuzzy k-nearest neighbor method, PSIBLAST
Frag1D	[Zhou *et al.*, 2010]	82.9	2,241	Fragment matching

Table 4.4: Resume of NN approaches for secondary structure prediction.

4.1.5 Hybrid methods

The combination of several methodologies can improve the efficiency of a predictive system. It is thus interesting to develop ensemble methods which allow the combination of existing classifiers so as to improve their recognition rate.

ANNs have proven to be a reliable tool. Three methods propose a combination of an ANN with other predictive strategies. A first method proposed in [Geourjon and Deleage, 1995], is called SOPM, and predicts sequences of a set of aligned proteins belonging to the same family. A second method [Zhang *et al.*, 1992] is based on three subsystems: a neural network module, a statistical module and a memory-based reasoning module. Finally, a third proposal [Adamczak *et al.*, 2005] includes relative solvent accessibility (RSA) of an amino acid in addition to attributes derived from evolutionary profiles. This approach combines the 2-stage protocol described in [Rost and Sander, 1993], with a number recurrent neural networks (RNNs) into a consensus predictor.

Three existing methods use a consensus algorithm to determine the best predictive results. [Cuff *et al.*, 1999] describes a combination of methods DSC, PHD, NNSSP, and PREDATOR described in previous sections. The method uses a simple consensus prediction using generated multiple sequence alignments. [Selbig *et al.*, 1999] proposes an ensemble of different SS prediction methods. A consensus algorithm selects the best result and uses a machine learning technique to build decision trees from existing data. [Subramani *et al.*, 2012] describes a protein structure predictor called ASTRO-FOLD 2.0. The key features referred to SS prediction are: SS prediction using a novel optimization-based consensus approach and β-sheet topology prediction using mixed-integer linear optimization (MILP)[4].

The algorithm described in [Muggleton *et al.*, 1992] allows relational descriptions for the SS prediction. This method, named the Inductive Logic Programming computer program, Golem, is applied to learning SS prediction rules for alpha-alpha domain type proteins. Golem learns a set of rules that predict which residues are part of the α-helices—based on their position relationships and chemical and physical properties.

[Montgomerie *et al.*, 2006] performed a structure-based sequence alignments predictor. A consensus result is obtained by integrating the conventional sequence-based methods.

The approach described in [Lin *et al.*, 2005], combines a knowledge-based prediction algorithm, called PROSP. The proposal uses small peptides with structural information. Authors introduce a measured named local match rate, which indicates the amount of structural information that each amino acid can extract from the knowledge base.

[4]MILP problems are optimization problems involving only linear functions and finitely many variables, some of which are constrained to attain only integer values.

[Pollastri *et al.*, 2007] and [Wang *et al.*, 2011] predict solvent accessibility in addition to secondary structure. The former method uses homology proteins of known structure in the form of simple structural frequency profiles extracted from sets of PDB templates. SS and SA are extracted directly from the templates. Structural information from templates improves SS and SA prediction quality. The latter method presents a probabilistic strategy for 8-class SS prediction using conditional neural fields (CNFs), a recently proposed probabilistic graphical model. This CNF method not only models the complex relationship between sequence features and SS, but also exploits the interdependency among SS types of adjacent residues. The method also uses evolutionary information for SS prediction.

A summary, in chronological order, of the hybrid methods for SS prediction and their accuracy results is shown in table 4.5.

Method	Reference	Q3 Acc.(%)	Dataset size	Description
	[Zhang *et al.*, 1992]	66.4	107	Combination of ANN, statistical and MBR modules
Golem	[Muggleton *et al.*, 1992]	81.0	12	Inductive logic programming, predictive rules
SOPM	[Geourjon and Deleage, 1995]	74.0	126	Alignment sequences predictive method and ANN
	[Cuff *et al.*, 1999]	72.9	396	Consensus system combining DSC, PHD and NNSSP
CoDe	[Selbig *et al.*, 1999]	72.8	396	Decision trees
	[Adamczak *et al.*, 2005]	78.4	603	Consensus predictor, RNNs
PROSP	[Lin *et al.*, 2005]	80.7	EVADS	Knowledge-based prediction algorithm, PSIPRED
	[Montgomerie *et al.*, 2006]	81.3	1644	Structure-based sequence alignments
Porter	[Pollastri *et al.*, 2007]	79.0	2171	Structure homology, SS and SA predictions
	[Wang *et al.*, 2011]	64.9 (Q8)	513	Probabilistic graphical model CNF
ASTROFOLD	[Subramani *et al.*, 2012]	80.2	CASP9	Mixed-integer linear optimization (MILP)

Table 4.5: Resume of hybrid methods for secondary structure prediction.

4.2 Tertiary structure prediction methods

Tertiary structure prediction methods are focused on determining the three-dimensional shape of a protein. Techniques for the prediction of contact or distance map between amino acids residues of a protein sequence are also included in this category.

The most relevant data structures used to represent the tertiary structure of a protein are the 3D models, *e.g.* torsion angle models and lattice models, distance maps (DM) and contact maps (CM).

The quality measures used to evaluate the accuracy of the 3D models are: RMSD or Root-mean-square deviation, that represents the absolute deviation (in angstroms) of individual C_α atoms between the model and the known true structure. GDT-TS or Global distance test-Total score, described in [Zemla, 2003], is used as major assessment criteria in CASP. GDT-TS is the number of α-carbons of a prediction not deviating from more than an established cutoff (in $\mathring{A}$) from the α-carbons of the targets, after optimal superimposition. TM-score or Template modeling score, detailed in [Zhang and Skolnick, 2004], measures the global structural similarity between the model and template proteins, according to the distances of each pair of residues.

For the accuracy assessment of contact maps, other three measures are also employed. Coverage indicates what percentage of contacts have been correctly identified. Accuracy reflects the number of correctly predicted contacts. X_d represents the distribution accuracy of the predicted contacts. X_d can differentiate between the distribution of predicted distances and a random distribution.

Contact maps present several advantages with respect to other representations. For instance, unlike 3D models of proteins, contact maps, as well as distance maps, have the desirable property of being insensitive to rotation or translation of the protein molecule. Also, given a contact map of a protein, it is possible to reconstruct a 3D model of the protein backbone, solving the Molecular Distance Geometry Problem (MDGP) [Lavor *et al.*, 2011]. This can be done in different ways, e.g., using quadratic potential GO model [Toona, 2012] or using tools like FT-COMAR [Vassura *et al.*, 2011, Vassura *et al.*, 2008]. It is also possible to obtain the coordinates of all protein atoms from the protein backbone using tools like SCWRL, IRECS, SCAP, SCATD or SCCOMP [Faure *et al.*, 2008], or using the recent tool SIDEpro [Nagata *et al.*, 2012]. Contact maps, as protein structure representation, are also useful to compare protein structures, using the maximum contact map overlap [Di Lena *et al.*, 2010].

Many different approaches for contact map prediction have been proposed in the literature, being the three mostly used approaches based on ANNs, EAs and SVMs.

4.2.1 Statistical approaches

Typically, the molecular shape of a protein determines the biological mechanism of the protein. Therefore, proteins with similar 3D structures are expected to have similar functions. This allows to predict the function of a protein based on its structural resemblance to proteins with known functions.

Many techniques based on the comparison of 3D structures have been proposed. [Zhang, 2009] develops a prediction server called I-TASSER which is a homology-based protein structures predictor based on sequence homology with known structures. This method was ranked as the No 1 server for protein structure prediction in recent CASP7, CASP8 and CASP9 competitions. Profile-Profile threading Alignment (PPA) and the Threading ASSEmbly Refinement program are the main components of this server application. HMM and Monte Carlo simulation are also used during the prediction stage. In [Karplus, 2009], a method named SAM-T08, uses HMMs and provides in addition to the 3D model and other information such as multiple sequence alignments (MSAs), prediction of local structure features, lists of potential templates of known structure, alignments to templates and residue–residue contact predictions.

Two protein folding recognition methods are described in [Olmea *et al.*, 1999] and [Raval *et al.*, 2002]. The first method introduces a statistical approach for folding recognition, which demonstrates that protein families are a rich source of information: sequence conservation and sequence correlation are two of the main properties which can be derived from the analysis of multiple sequence alignments. Sequence conservation is related to the direct evolutionary pressure to retain the chemical characteristics of some positions in order to maintain a given function. Sequence correlation is attributed to the small sequence adjustment needed to maintain protein stability against constant mutational drift. It is showed that sequence conservation and correlation provide enough information to detect incorrectly folded proteins. The second method is based on Bayesian networks. This approach is focused on protein fold and superfamily recognition. This Bayesian network also includes HMM's.

[Zhou *et al.*, 2008] proposes several approaches to predict contact order from the amino acid sequence only. A first approach is based on a weighted linear combination of predicted secondary structure content and amino acid composition. A second approach is based on sequence similarity to known three-dimensional structures.

A summary, in chronological order, of the methods described in this section is shown in table 4.7.

4.2.2 Neural networks methods

As stated before, ANNs are one of the most popular methods for the tertiary structure prediction. Some ANN proposals base the prediction of contact maps (CMs) on chemico-physical properties and structural information of the amino acids. [Bohr *et al.*, 1990] presents a feed-forward neural network which is trained with matching sets of amino acid sequences and two different types of structural information: the corresponding secondary structure and the contact map of known proteins. The input of the network consists of the amino acid sequence (using a windows of 61 amino acids). The hidden layer has 300 neurons and the output layer 30 neurons used to predict contacts between the amino acid that occupies the central position in the window and the rest of the residues of the window. Three additional neurons are used to predict the tertiary motifs. The method proposed in [Fariselli and Casadio, 1999] is based on two input neurons which use input vectors with information about the pair of amino acids in contact and their environment, the length of the protein sequence and the sequence separation between amino acids. In addition, several variables are added, such as the hydrophobicity of the environment, as well as evolutionary information. This work was enhanced with the addition of correlated mutations in [Fariselli *et al.*, 2001]. This method uses as input evolutionary information, sequence conservation, correlated mutations and predicted secondary structures. Also a filtering procedure is added to the predictor, to avoid contact overprediction, taking into account the amount of contacts that each residue type can establish. The filtering procedure is based on the occupancy data (or residue-coordination numbers) of each residue. [Gorodkin *et al.*, 1999] adopts two layer feed-forward ANN. The ANN is trained using the results of a study about correlation between sequence separation and distance of each amino acid pair. [Chen *et al.*, 2005] presents a probabilistic neural network (PNN) with conformational energy function (CEF) based on chemico-physical knowledge of amino acids. In this method, the principal components are first extracted from selected protein structures with lower sequence identity, and an initial matrix of contact map is constructed by K-L expansion[5]. Then, the PNN is used for predicting the long-range interaction of amino acids. In particular, this method uses the CEF and chemico-physical characteristics of amino acids for the prediction. [Liu *et al.*, 2005] uses a recurrent neural network with bias units for contact maps prediction. The architecture consists of three layers of neurons: one output neuron representing the contact propensity, one hidden layer containing 10 neurons and one input layer with 40 neurons for 5 residue pairwise, 4 neurons for residue classification according to hydrophobicity, polarity, acidity and basicity and 3 neurons for secondary

[5]Karhunen-Loeve expansion is a representation of a stochastic process as an infinite linear combination of orthogonal functions.

structure information. This topology also has 10 conjunction units and two bias units. A Gaussian function is used as the activation transfer function of the network. A transiently chaotic neural network (TCNN) is developed in [Liu *et al.*, 2006]. This topology consists of three layers of neurons: one output neuron representing the contact propensity, one hidden layer containing ten neurons and one input layer with different number of neurons depending on the amount of information encoded; the method uses 1050 neurons for 5 residue pairwise, 10 neurons for residue classification according to hydrophobicity, polarity, acidity and basicity, 6 neurons for secondary structure information.

Other methods are based on radial basis function neural network (RBFNN). In particular, [Zhang and Huang, 2004] introduces a genetic algorithm which is used to optimize the radial basis function widths and hidden centers of the RBFNN. Then a novel binary encoding scheme is employed to train the network for the purpose of learning and predicting the inter-residue contacts patterns of protein sequences. This model generates a multivariate nonlinear mapping, useful for solve multi-parameters and multi-model type of nonlinear classification problem, such as protein inter-residue contact maps prediction. [Zhang *et al.*, 2005] improves the previous method including a binary encoding scheme for learning the inter-residue contact patterns.

Another popular type of ANNs for contact map prediction is represented by recursive neural networks (RNNs). [Vullo *et al.*, 2006] introduces a predictor based on ensembles of two-layered BRNNs. The method classifies the components of the principal eigenvector (PE) and uses predicted secondary structure information and hydrophobicity interaction scales. [Tegge *et al.*, 2009] was ranked as one of the most accurate methods from CASP8. This method performs two steps. First, a 2D-RNN predicts a residue-residue contact map. After that, an ANN predicts the special β-sheet conformation. [Walsh *et al.*, 2009] introduces a new class of distance restraints for protein structures: multi-class distance maps. Two predictors of 4-class maps based on RNNs were developed: one *ab initio*, or relying on the sequence and on evolutionary information; one template-based, or in which homology information to known structures is provided as a further input.

Evolutionary, as well as structure information are employed in various ANNs. SPINE-2D [Xue *et al.*, 2009] consists of two neural networks using one and two layers, respectively. These networks use 34 features as input, including PSSM from PSIBLAST [Altschul *et al.*, 1997], seven physico-chemical properties of amino acids, including hydrophobicity, volume and polarizability, and secondary structure from the DSSP secondary structure assignment program [Kabsch and Sander, 1983]. [Lippi and Frasconi, 2009] proposes a novel hybrid architecture based on neural and Markov logic networks with grounding-specific weights, in order to predict beta contacts.

Multiple alignment profiles, secondary structure and solvent accessibility in two states are used as input. [Punta and Rost, 2005] proposes a method that combines different sources of information as protein properties, biophysical features, evolutionary profiles, secondary structure prediction and alignment information. This method, called PROFcon, achieves good accuracy levels for long-range contacts (inter-residue separation of 24). [Di Lena *et al.*, 2012] develops a machine learning approach for contact map prediction, named SCRATCH. This approach consists of three steps. First, two neural networks predict contacts and secondary structure elements. Second, an energy-based method predicts contact probabilities between residues in secondary structure elements. Finally, a deep neural network architecture organizes and refines the prediction of contacts.

All the cited ANN methods for tertiary structure prediction and their achieved results are summarize, in chronological order, in table 4.6.

Method	Reference	Acc.(%)	Dataset size	Description
	[Fariselli and Casadio, 1999]	16.0	408	Evolutionary information, AA features
distanceP	[Gorodkin *et al.*, 1999]	70.3	9	Correlation between distances and sequence separation
	[Fariselli *et al.*, 2001]	21.0	173	Correlated mutations
	[Zhang and Huang, 2004]	32.0	173	RBFNN optimized by GA
	[Chen *et al.*, 2005]	31.0	100	Probabilistic neural network (PNN)
	[Liu *et al.*, 2005]	8.0	105	RNN, AA properties
	[Zhang *et al.*, 2005]	32.0	173	RBFNN, binary encoding scheme
PROFcon	[Punta and Rost, 2005]	<20.0	748	SS prediction, evolutionary profiles, AA properties
	[Liu *et al.*, 2006]	8.0	2,095	Transiently chaotic neural network (TCNN)
	[Vullo *et al.*, 2006]	36.5	327	Ensembles of two-layered BRNN
NNCON	[Tegge *et al.*, 2009]	31.0	48	2D-Recursive Neural Network
	[Walsh *et al.*, 2009]	16.0	CASP7	Multi-class distance map
	[Lippi and Frasconi, 2009]	47.3	80	ANN, Markov logic networks
SPINE-2D	[Xue *et al.*, 2009]	23.0-26.0	500,CASP7	ANN, PSSM
SCRATCH	[Di Lena *et al.*, 2012]	30.0	CASP8,CASP9	ANN, SS

Table 4.6: Resume of neural network methods for tertiary structure prediction.

4.2.3 Support vector machines methods

Several contact map predictors are based on SVM approaches. [Zhao and Karypis, 2002] incorporates various features such as sequence profiles and their conservation, correlated mutation analysis based on various amino acid physico-chemical properties and secondary structure. The method proposed in [Cheng and Baldi, 2007] uses a large set of informative features as pairwise information features, secondary structure, relative solvent accessibility, contact potentials or local window feature. [Wu *et al.*, 2011] and [Lo *et al.*, 2009] methods use evolutionary information for the contact map prediction. [Wu *et al.*, 2011] develops a composite set of nine SVM-based contact predictors that are used in ITASSER [Roy *et al.*, 2010] simulation in combination with sparse template contact restraints. They use the original energy function of ITASSER and contact predictions generated by extended versions of SVMSEQ [Wu and Zhang, 2008]. [Lo *et al.*, 2009] proposes a hierarchical scheme for contact prediction, with an application in membrane proteins. This approach consists of two levels: in the first level, contact residues are predicted from sequences; while in the second one, their pairing relationships are further predicted. The statistical analyses on contact propensities are combined with evolutionary profile, relative solvent accessibility and helical features.

On the other hand, [Han *et al.*, 2005] develops a fold recognition method based on SVM and PSIBLAST. The alignment, between a query protein and a template, is transformed into a feature vector of length n+1 (where n is the length of the template), which is then evaluated by the SVM. The otput of the SVM is a probability which a query sequence is related to a template.

SVM methods for tertiary structure prediction are summarized in table 4.8.

Method	Reference	Acc.(%)	Dataset size	Description
	[Olmea *et al.*, 1999]	76.2	71	MSA, sequence conservation and sequence correlation
	[Raval *et al.*, 2002]	77.0	25	Protein fold and superfamily recognition
	[Zhou *et al.*, 2008]	74.2	499	Weighted linear combination of predicted SS
I-TASSER	[Zhang, 2009]	34.0	CASP8	Protein structure alignment
SAM-T08	[Karplus, 2009]	61.4 (GDT-TS)	CASP8	HMMs and MSAs

Table 4.7: Resume of statistical methods for tertiary structure prediction.

Method	Reference	Acc.(%)	Dataset size	Description
	[Zhao and Karypis, 2002]	22.4	177	Sequence profiles, correlated mutations, SS, AA features
SVMcon	[Cheng and Baldi, 2007]	21.0	48	SA, SS, pairwise information
	[Han *et al.*, 2005]	46.0	16	Estimating the significance of the alignments
	[Lo *et al.*, 2009]	56.0	52	Propensities, evolutionary profile, SA
	[Wu *et al.*, 2011]	31.0	273	I-TASSER, sparse template contact restraints

Table 4.8: Resume of SVM methods for tertiary structure prediction.

4.2.4 Evolutionary algorithm methods

Methods based on EAs may use various possible representations of a protein structure. A first possibility is represented by torsion angles. Torsion or dihedral angles (Φ, Ψ) represent the position of the atoms of an amino acid chain. A possible representation could be $[(\Phi_1, \Psi_1)...(\Phi_n, \Psi_n)]$ where n represents the total number of residues of a protein. Collisions among atoms must be avoided according to the Ramachandran plot[6]. [Ramachandran *et al.*, 1965].

A second representation is based on lattice models. For the lattice models, each element location can be represented as a vector $(x_1, y_1)...(x_n, y_n)$ where x and y are the coordinates of each amino acid in a 2-dimensional lattice (or three coordinates in a 3-dimensional lattice).

Considering the possible number of movements to the next point, another representation could be direction vectors, $(L_1, L_2..L_n)$ where $L_i \in$ UP, DOWN, LEFT, RIGHT are the locations of each amino acid with respect to the previous one.

HP (hyprophobic-polar) model is detailed in [Dill, 1985] where a sequence is represented as a string $s \in (H, P)^+$, where H represents a hydrophobic amino acid and P a hydrophilic one.

We grouped the different evolutionary approaches according to the representation models described. We start with methods that use dihedral or torsion angles. [Judson *et al.*, 1993] uses the Chemistry at HARvard Macromolecular Mechanics (CHARMM) force fields [Brooks *et al.*, 1983] as fitness function. [Dandekar and Argos, 1994] employs a fitness function which takes into account some parameters like hydrophobic interactions, local forces, hydrogen bonds, clashes and secondary and tertiary structure. [Cui *et al.*, 1998] develops a 3-torsion angles representation (Φ, Ψ, ω). No mutation operator is used. The fitness function consists of hydrophobic interactions and van der Waals contacts measures. [Schulze-Kremer *et al.*, 2000] develops a GA for a force field model, that represents chemical reactions and physical forces that occurs in a protein. [Kehyayan and Mansour, 2008] uses a fitness function based on CHARMM, to evaluate the potential energy values, and a scatter search algorithm. For the representation, authors use some amino acid features,*e.g.*, partial charge and van der Waals bond. In [Pedersen and Moult, 1997] a global free energy function of an unfolded conformation is calculated as the sum of threes types of energy: local backbone electrostatic energy, which represents the sum of the interactions between N-H and C=O groups among amino acids, intra-molecular electrostatic energy, that reflects the rest of intra-molecular interactions, and solvation free energy, that takes into account different interactions as hydrogen bonding, ion-dipole, and dipole-dipole attractions

[6] A way to visualize backbone dihedral angles of amino acid residues in protein structure. This plot shows the allowed (Φ, Ψ) backbone conformational regions.

or van der Waals forces.

Direction vector representation is employed in several methods: [Dandekar and Argos, 1992], [Unger and Moult, 1993] and [Braden, 2002]. [Dandekar and Argos, 1992] used a tetrahedral lattice as structure conformation with direction vectors, while [Unger and Moult, 1993] used a three-dimensional square lattice model. Movements required in the representation are $\{U, D, L, R, F, B\}^n$ (up, down, left, right, forward, backward), where n is the sequence length. The method proposed in [Braden, 2002] improves the model detailed in [Unger and Moult, 1993]. It used three-dimensional protein representation with 32 possible movements for each protein residue. The fitness function analyzes some protein characteristics like hydrophobicity, charge, and side-chain size.

HP models are used in several evolutionary proposals. For instance, [Unger, 2004] proposes a method that adopts a two-dimensional square lattice based on HP model. [Liang *et al.*, 2001] uses a hybrid algorithm consists of Monte Carlo optimization and an HP square model. [Cotta, 2003] develops a HP model with cubic lattice implementation. This method adopts a fitness function based on the Kronecker-delta function[7], distance between target residues, overlap involving the residues and free contact energy between target residues. A coefficient evaluates possible penalties due to violations movements.

Various contact and distance map predictors are also based on EAs. A distance matrix representation is presented in [Piccolboni and Mauri, 1998]. The method analyzes possible distances between each pair of amino acids for each protein. Fitness function is calculated using three terms: two penalty constraint factors and a hydrophobicity interaction term. This method also describes a repair algorithm and a penalization strategy for distance map unfeasible solutions. The method proposed in [Gupta *et al.*, 2005] starts with an initial random contact map population for a given amino acid sequence. A neural network and four physical protein properties (charge, sequence distance, neighborhood hydrophobicity and degree of vertices) are used in the fitness function. The most accurate contact map is selected after last generation. Later, this contact map is compared with a contact map template for each fold using graph theory. The maximum scoring template determines the fold of the protein. [Zhang *et al.*, 2007] proposes an EA that employs a 19-bit representation for a protein, where bits 0-8 represent each possible pairwise between amino acids, bits 9-12 represent a residue classification (polar, non-polar, acid or base), bits 13-15 represent which possible secondary structure a residue is among helix, sheet and coil. Bits 16-17 represent the sequence length and bits 18-19 represent the sequence separation. A GA is used to improve a radial basis function neural network. The method proposed in [Chen, 2010] is

[7]The function is 1 if the variables are equal, and 0 otherwise.

based on genetic algorithm classifiers (GaCs) for the long range contacts prediction. These contacts have a sequence separation between amino acids of more than 24 residues. This method incorporates the sequence profile centers (SPCs). An SPC represents an encoding vector for a residue pair that belong to the same long-range contact class or long-range non-contact class. Finally, [MacCallum, 2004] uses a Self-organizing map [Kohonen and Makisara, 1989] and Genetic Programming (GP) approach to predict protein contacts.

An EA that incorporates a local search phase is called a Memetic Algorithm. Multimeme Algorithms (a type of Memetic Algorithms) use several kinds of local searches. This paradigm is adopted in the following four methods. [Krasnogor *et al.*, 2002] introduce a Multimeme algorithm with a HP model and Functional Model Protein [Blackburne and Hirst, 2001] in two and three dimensions. This method incorporates a new mating strategy based on a contact map memory and analyzes if a new offspring is compatible with it. [Pelta and Krasnogor, 2005] proposed a combination of fuzzy logic and multimeme algorithms in HP models. Fuzzy logic is used as a modifier of the memepool local searchers, evaluating possible solutions. Another memetic algorithm was proposed in [Islam and Chetty, 2009]. This method uses a HP model and calculates the fitness function, with two new parameters called H-compliance and P-compliance. H-compliance measure of how compactly a residue is located to the H-core centre and P-compliance is a measure of how close the residue is to any of the sides of the lattice.

[Chen, 2010] proposed an ensemble of GA classifiers to predict long-range contacts. The individuals of the GA represent three amino acid windows and 20 properties obtained from the HSSP database of protein structure-sequence alignments [Dodge *et al.*, 1998] for each residue in such windows. The method also uses the sequence profile centers (SPCs).

Several prediction methods have considered PSP problem as a multi-objective optimization problem (MOP). A parallel multi-objective optimization was performed by using CHARMM energy function in [Calvo *et al.*, 2009]. [Shi *et al.*, 2004] proposed a multi-objective Feature Analysis and Selection Algorithm (MOFASA) in order to solve the Protein Fold Recognition (PFR) problem. In [Cutello *et al.*, 2006], a immune inspired Pareto archived evolutionary strategy (I-PAES) algorithm is used to explore the conformational space searching for the minimal interaction energies of bond and non-bond atoms. The method uses a torsion angles model representation and CHARMM equation as fitness function. [Judy *et al.*, 2009] proposed a MOEA, which represents protein structures by torsion angles. They modified the classical algorithm PAES, introducing two immune inspired operators. This algorithm, called MI-PAES uses adaptive probabilities of crossover, mutation and immune operation. Calvo et al. [Calvo *et al.*, 2011] also proposed a MOEA, called Pitagoras-PSP. This algorithm uses an evolutionary *ab initio* approach based on PAES.

The algorithm predicts protein backbone and side-chain torsion angles and it uses an energy function as fitness function. Mutation operators maintain values of torsion angles in feasible ranges according to secondary structure of residues.

A survey of evolutionary computation methods for tertiary structure prediction is shown in table 4.9.

Method	Reference	Acc.(%)	Dataset size	Description
	[Dandekar and Argos, 1992]	-	6	Tetrahedral lattice with direction vectors
	[Judson *et al.*, 1993]	-	72	Dihedral angle representation model
	[Unger and Moult, 1993]	-	8	Square lattice in the HP model
	[Dandekar and Argos, 1994]	-	5	Dihedral angle representation Φ, Ψ
	[Pedersen and Moult, 1997]	3.1 (RMSD)	28	Global free energy function
	[Piccolboni and Mauri, 1998]	-	3	Distance matrix representation
	[Cui *et al.*, 1998]	1.48-4.48 (RMSD)	5	Torsion angles, fitness interaction
	[Schulze-Kremer *et al.*, 2000]	1.08 (RMSD)	3	Force field model
	[Liang *et al.*, 2001]	-	8	Monte Carlo optimization and HP model.
	[Braden, 2002]	-	2	3-Dimensional protein representation
	[Krasnogor *et al.*, 2002]	-	200	Multimeme algorithm, HP model
	[Cotta, 2003]	-	8	HP model with cubic lattice implementation
	[MacCallum, 2004]	21.4	CASP5	Self-organizing map and GP approach
MOFASA	[Shi *et al.*, 2004]	53.0	27	Folding recognition
	[Gupta *et al.*, 2005]	69.0-88.0	24	Contact map representation
FANS	[Pelta and Krasnogor, 2005]	-	4	Fuzzy logic and multimeme algorithm
	[Cutello *et al.*, 2006]	3.6 (RMSD)	5	I-PAES, torsion angles model, CHARMM
	[Zhang *et al.*, 2007]	-	61	Residue properties representation
	[Kehyayan and Mansour, 2008]	9.43 (RMSD)	2	Fitness function CHARMM based
	[Calvo *et al.*, 2009]	1.8 (RMSD)	2	CHARM
	[Islam and Chetty, 2009]	-	9	Memetic algorithm, HP model
MI-PAES	[Judy *et al.*, 2009]	4.23 (RMSD)	4	Torsion angle model
	[Chen, 2010]	21.5	480	Sequence profile centers (SPCs)
Pitagoras-PSP	[Calvo *et al.*, 2011]	9.15 (RMSD)	CASP8	PAES, torsion angles

Table 4.9: Resume of evolutionary computation methods for tertiary structure prediction.

4.2.5 Case-based reasoning methods

Case-based reasoning (CBR) is a paradigm for analogical reasoning where experiences are represented as cases in a case base. Such cases are then retrieved and reused during problem solving. A case represents knowledge about a particular problem solving experience and includes a problem description, a solution to the problem and feedback on the success of the solution. The case base is a repository of cases, designed to support the efficient storage and retrieval of a large number of complex cases. CBR is particularly useful in domains that are poorly understood or evolving, where knowledge is difficult to formalize. CBR is based on the premise that similar problems have similar solutions. An advantage of CBR as a problem-solving paradigm is that it is applicable to a wide range of problems. In particular, CBR is particularly applicable to the biological domain, like protein structures. This is because biological systems are often homologous (rooted in evolution) and because biologists often use a form of reasoning similar to CBR, where experiments are designed and performed based on the similarity between features of a new system and those of known systems.

[Zhang *et al.*, 1993] proposes a memory-based reasoning method to predict protein torsion angles based on known structures. Their work is based on the premise that if two amino acids have similar physical properties and occur in a similar environment, then they should have similar structure (in terms of their Φ and Ψ angles).

[Conklin *et al.*, 1994] applies CBR to determine the three-dimensional structure of proteins from experimental crystallographic data. This work in molecular scene analysis concerns the automated reconstruction and interpretation of protein image data (in the form of a three-dimensional electron density map). Cases correspond to previously determined structures. Discovered spatial and visual concepts of a structure are used to index cases. Cases are retrieved from the case base through a pattern-matching process that involves the comparison of unidentified features in a novel electron density map (derived from an image-segmentation process) with motifs from known structures. This approach combines a bottom-up approach to image analysis. Image-processing techniques are applied to extract features from the maps, with a top-down approach and CBR is used to anticipate what motifs are likely to occur in the image.

The following three methods predict protein contact maps using a CBR approach. [Glasgow *et al.*, 2006] proposes a contact map predictor using sequence data. Case representation includes protein name, protein sequence, assignment of secondary structure to residues, structure class and protein contact map. The solution consists of a 3D backbone model of the protein structure computed for the input contact map. The method considers only alpha proteins. A similarity measure for comparing the query contact map with maps generated from structures in the PDB is derived using techniques

from machine vision. [Davies *et al.*, 2006] presents a CBR hierarchical method, in the sense that it considers protein contact maps at varying levels of structural complexity. In a bottom-up fashion, the method initially construct secondary structure motifs using the contact map and geometric knowledge of α-helices and β-strands. In particular, the method retrieves similar α-helix pair contact maps and adapts the known structures to predict alignments for the unknown structures. Case representation consists of protein name, primary sequence, assignment of secondary structure to residues, class of structure and the protein contact map.

A summary of CBR methods for tertiary structure prediction is shown in table 4.10.

4.2.6 Other predictive methods

In addition to the cited approaches to PSP, there are other important approaches, such as random forest algorithm, integer linear optimization and sparse inverse covariance. In this section, we will cover some of these strategies.

Various contact map predictors are based on mathematical models. The six following methods belong to this category. [Gao *et al.*, 2009] describes a consensus contact prediction method based on an integer linear programming model. This method evaluates its correlation by using maximum likelihood estimation[8] and extracts independent latent servers by using principal component analysis (PCA). A system of weights to maximize the differences between true and false contacts is also implemented. [Rajgaria *et al.*, 2009, Rajgaria *et al.*, 2010] proposes an integer linear optimization approach which uses a high resolution distance dependent force field to calculate the interaction energy between different residues of a protein. This method predicts the hydrophobic residue contacts in alpha proteins. The algorithm incorporates a set of constraints based on the commonly observed contact distances between various elements of a secondary structure and the possibility of adding new constraints to the model. [Wei *et al.*, 2011] improves a mathematical optimization model to predict the contacts in transmembrane alpha proteins. Physical constraints were incorporated in the mathematical model and a blind contact prediction scheme was tested on two different protein sets. [Jones *et al.*, 2012] develops PSICOV, a novel method which introduced the use of sparse inverse covariance estimation to the problem of protein contact prediction from coupled mutation correlation in the multiple sequence alignments. This method performs corrections for phylogenetic and entropic correlation noise and allows accurate discrimination of direct from indirectly coupled mutation correlations in the MSA. [Ashkenazy *et al.*, 2011] proposes a

[8] A method of estimating the parameters of a statistical model.

method for combining structural data from several templates to enhance contact map prediction of novel proteins. The use of multiple templates improves prediction of contact maps, and can also be used to reveal novel conformations.

Evolutionary information is employed in several methods, such as [Björkholm *et al.*, 2009], [Wang *et al.*, 2010] and [Marks *et al.*, 2011]. The first method presents a novel HMM method for contact map prediction using as training data homologous sequences, predicted secondary structure and a library of local neighborhoods (local descriptors of protein structure). The second method, called MULTICOM, describes a multi-level combination approach to improve the various steps in PSP combining complementary and alternative templates, alignments and models. This approach incorporates five automated PSP servers and one human predictor. The last method uses a maximum entropy model of the protein sequence based on statistics of MSA to infer evolutionary constraints from a set of homologous protein sequences. The inferred residue pair couplings constitutes enough information to define an accurate 3D protein fold model.

[Li *et al.*, 2011] develops ProC_S3, a set of Random Forest[9] algorithm-based models. Some characteristics of the algorithm are the use of a propensity matrix between residues and a set of seven amino acids groups based on probabilities.

[Eickholt *et al.*, 2011] performs a conformation ensemble approach. The method collects various models (SVMCon, TASSER and ROSSETA) and complementary information from a variety of methods to improve the residue-residue contact prediction.

A sequence-based protein contact map prediction method, named LRcon, based on logistic regression is detailed in [Yang and Chen, 2011]. A feature vector is fed into the logistic regression-based algorithm to make a consensus prediction for each residue pair.

JUSTcon is described in [Abu-Doleh *et al.*, 2011]. The method consists of multiple parallel stages that are based on adaptive neuro-fuzzy inference System (ANFIS) and K nearest neighbors (KNNs) classifier. A simple expert system selects the window size of a smart filter to ensure normal connectivity behaviors of residues pairs.

All these methods are summarized in table 4.11.

[9] An ensemble of random decision tree classifiers, that makes predictions by combining the predictions of the individual trees.

Method	Reference	Acc.(%)	Dataset size	Description
SBB	[Zhang *et al.*, 1993]	-	74	Torsion angles prediction, AA properties
	[Conklin *et al.*, 1994]	<1.0 (RMS)	402	Pattern-matching process, density map
	[Davies *et al.*, 2006]	1.24 (RMSD)	422	Contact map, SS prediction
	[Glasgow *et al.*, 2006]	1.86 (RMSD)	100	Contact map prediction

Table 4.10: Resume of CBR methods for tertiary structure prediction.

Method	Reference	Acc. (%)	Dataset size	Description
FragHMMent	[Björkholm *et al.*, 2009]	22.8	151	HMM
	[Gao *et al.*, 2009]	37.0	CASP7	Hybrid method
	[Rajgaria *et al.*, 2009]	66.0	48	Integer linear optimization
MULTICOM	[Wang *et al.*, 2010]	63.0 (GDT-TS)	120	Template-based approach
JUSTcon	[Abu-Doleh *et al.*, 2011]	45.2	450	Nearest neighbor-based algorithm
WMC	[Ashkenazy *et al.*, 2011]	23.6 (PCC)	CASP8	Template-based approaches
	[Eickholt *et al.*, 2011]	30.0	CASP9	Hybrid method
ProC_S3	[Li *et al.*, 2011]	26.9	1,490	Random Forest algorithm based model
EVfold	[Marks *et al.*, 2011]	2.7–4.8 (RMSD)	15	Max. entropy model, corr. mutations
	[Wei *et al.*, 2011]	56.0	5	Integer linear optimization
LRcon	[Yang and Chen, 2011]	41.5	846	Hybrid method
PSICOV	[Jones *et al.*, 2012]	>50.0	118	Sparse inverse covariance

Table 4.11: Resume of other methods for tertiary structure prediction.

Edition	# targets	Method	Reference	Acc.(%)
CASP10	145	Zhang-Server	[Zhang, 2009]	56.48
CASP9	144	QUARK	[Xu and Zhang, 2012]	56.13
CASP8	172	Zhang-Server	[Zhang, 2009]	64.83
CASP7	124	Zhang	[Zhang, 2009]	78.03
CASP6	87	TASSER-3D	[Zhang, 2009]	56.23
CASP5	67	Bujnicki-Janusz	[Kosinski *et al.*, 2003]	47.17

Table 4.12: Resume of latest CASP competitions.

Finally, table 4.12 summarize the main characteristics of the latest CASP competitions. First column indicates the edition of the competition. The second column shows the number of target proteins. The third and forth column present the name of the best method of each edition for PSP and its reference. The fifth column indicates the achieved accuracy (GDT-TS) of each method.

4.3 Summary

The most representative approaches for solving the protein structure (secondary and tertiary) prediction problem, are summarized in this chapter. All these methods have been classified according to three criterias. In first place, the approaches have been classified according to the type of predicted structure (secondary or tertiary). In second place, the classification was made according to the type of methodology of the approaches (statistical, soft computing or lazy approaches). Finally, some features of the implementation of the methods have been taken into account to classify the different approaches of a determined methodology.

Part III

Proposals

Chapter 5

Evolutionary approaches for the protein structure prediction

In this chapter, we detail the methodology of two proposed approaches for the protein structure prediction. The first approach corresponds to a protein contact map predictor. The second proposal is an algorithm for the secondary structure prediction. Both of them, are based on evolutionary algorithms, physico-chemical properties of amino acids and structural features of the proteins.

5.1 Multi-objective Evolutionary Contact Map Predictor (MECoMaP)

This section describes our proposal for contact maps prediction. In particular, our proposal is based on a multi-objective EA (MOEA). The prediction is based on three physico-chemical properties (hydrophobicity, polarity and charge), and other structural features (solvent accessibility and secondary structure). It is known that amino acid properties play an important role in the PSP problem [Gu and Bourne, 2003]. Several PSP methods rely on amino acids properties, *e.g.*, HP models [Unger and Moult, 1993].

In the following sections, we define the procedures, elements and evaluation measures used by our prediction method.

5.1.1 Methodology

Our proposal, called MECoMaP (Multi-objective Evolutionary Contact Map Predictor), is based on the Strength Pareto Evolutionary Algorithm (SPEA) [Ziztler and Thiele, 1998]. This algorithm uses an external population of

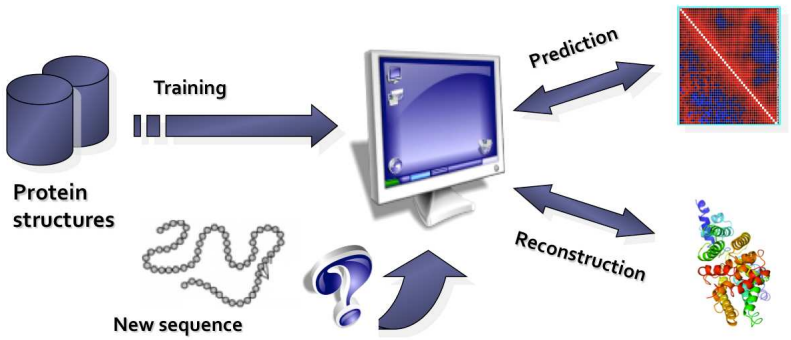


Figure 5.1: Experimental procedure scheme.

non-dominated solutions, which is obtained at the end of every generation. The algorithm is based on the strength concept. Recall that the strength of an individual x is given by the number of individuals that x dominates. The fitness of an individual is proportional to its strength, as will be detailed in the following. Each individual of the population represents a decision rule. In particular, rules are based on the previously mentioned amino acid properties. Basically rules specify a set of conditions on each property, that, if satisfied, predict a contact between two amino acids.

Figure 5.1 represents the experimental procedure to predict protein contact maps adopted in our work. First, protein sequences and distances between amino acids, as well as other complementary information, are obtained from PDB, with a procedure that will be described in section 5.1.4. This information constitutes the training set. Then, our algorithm is applied and generates our predictive model based on a set of rules. The model is then used for generating a contact map for each protein of the test dataset.

5.1.2 Physico-chemical properties of the amino acids

The most direct information we can extract from the primary sequence of a protein are physico-chemical characteristics of its residues (in this case hydrophobicity, polarity and net charge). With this information, we can generate representations of, for example, how the hydrophobicity varies along the sequence of the protein and obtain information about hydrophobic areas, which may help in the prediction of structural characteristics.

By definition, a substance is hydrophobic if it is not miscible with water. Hydrophobicity is then defined as the incapacity of interacting with the molecules of water by ion-dipole interactions or by hydrogen bonds. Hydrophobic amino acids are generally found in the inner layers of the proteins protected from direct contact with water. On the contrary, the hydrophilic amino acids are generally found on the outside of proteins as

well as in the active centers of enzymatically active proteins.

In chemistry, polarity refers to a separation of electric charge leading to a molecule or its chemical groups having an electric dipole or multipole moment.

The net charge is the algebraic sum of all the charged groups present in any amino acid, peptide or protein.

Amino acids can be classified according to these properties of their residues. There are four main classes:

- Non-polar or hydrophobic (Ala, Val, Leu, Ile, Pro, Phe, Trp, Met, Cys and Gly).
- Polar and uncharged (Asn, Gln, Ser, Thr and Tyr).
- Polar and negative charge (acidic) (Asp and Glu).
- Polar and positive charge (basic) (Arg, His and Lys).

In our proposal, we use the Kyte-Doolittle hydropathy profile [Kyte and Doolittle, 1982] for hydrophobicity, the Grantham's profile [Grantham, 1974] for polarity and Klein's scale for net charge [Klein *et al.*, 1984]. In table 5.1, we can see the property values for each amino acid according to the cited scales, normalized between -1 and 1 for hydrophobicity and polarity. For the hydrophobicity and polarity values, the more positive the value, the more hydrophobic or polar is the amino acid. For the net charge, a positive, negative or neutral charge, are represented with a 1, -1 or 0, respectively. We can see, for example that amino acid I has a hydrophobicity value equal to 1.0, which means that I is highly hydrophobic, a polarity of -0.93, which means that I is poorly polar and a neutral charge.

In addition to these properties, we also use two structural features of proteins: secondary structure prediction (SS) and solvent accessibility (SA) which are explained in the following section.

5.1.3 Structural features of protein residues

As mentioned before, MECoMaP uses also structural features of protein residues. In particular, it uses secondary structures and solvent accessibility. Secondary structure prediction consists of predicting the location of α-helices, β-sheets and turns from a sequence of amino acids. The location of these motifs could be used by approximation algorithms to obtain the tertiary structure of the protein. A 3-state representation of SS (helix, sheet or coil) is employed in our approach. The prediction is performed using PSIPRED [Jones, 1999].

Table 5.1: Values of different properties according to the cited scales for each amino acid. H represents the hydrophobicity, P the polarity and C the charge.

Prop.	A	C	D	E	F	G	H	I	K	L
H	0.40	0.56	-0.78	-0.78	0.62	-0.09	-0.71	1.00	-0.87	0.84
P	-0.21	-0.85	1.00	0.83	-0.93	0.01	0.36	-0.93	0.58	-1.00
C	0	0	-1	-1	0	0	0	0	1	0

Prop.	M	N	P	Q	R	S	T	V	W	Y
H	0.42	-0.78	-0.36	-0.78	-1.00	-0.18	-0.16	0.93	-0.20	-0.30
P	-0.80	0.65	-0.23	0.38	0.38	0.06	-0.09	-0.75	-0.88	-0.68
C	0	0	0	0	1	0	0	0	0	0

SA refers to the degree to which a residue interacts with the solvent molecules[1]. SA, also known as accessible surface area (ASA), represents the solvent exposed surface area of a residue in a protein. The studies of solvent accessibility have shown that the process of protein folding is driven to maximal compactness by solvent aversion of some residue. Therefore, knowledge of residue solvent accessibility provides us useful information for the prediction of the structure and function of a protein [Lo *et al.*, 2009, Cheng and Baldi, 2007]. Relative solvent accessibility (RSA) is required for the prediction. To calculate the RSA of a residue, we use the DSSP program [Kabsch and Sander, 1983], and then obtain the actual SA of each residue as described in [Bacardit *et al.*, 2009]. SA is divided by the maximum accessible surface in the extended conformation of its AA type. We finally obtain a 5-state representation (ranging from 0 to 4) for SA, where lower values mean a buried state and higher values represent exposed states. The prediction is performed using ICOS Server for the prediction of structural aspects of protein residues[2].

The algorithm may also use evolutionary information. We have included in our representation the evolutionary information obtained from PSI-BLAST [Altschul *et al.*, 1997] using non-redundant protein sequences database. Sequence alignment is a standard technique in bioinformatics for visualizing the relationships between residues in a collection of evolutionary or structurally related protein. Position-specific scoring matrices (PSSM) are obtained from sequence alignments. PSSM matrices determine the substitution scores between amino acids according to their positions in the alignment. Each cell of the matrix is calculated as the log_2 of the observed substitution frequency at a given position divided by the expected

[1]Solvents are substances that dissolve a solute in a solution. They are composed of polar molecules, such as water.

[2]http://cruncher.cs.nott.ac.uk/psp/prediction

substitution frequency at that position. Thus, a positive score (ratio > 1) indicates that the observed frequency exceeds the expected frequency, suggesting that this substitution is surprisingly favored. A negative score (ratio < 1) indicates the opposite: the observed substitution frequency is lower than the expected frequency, suggesting that the substitution is not favored. This information has been used to represent each residue in the window. We normalize the PSSM values between -1 and 1.

5.1.4 Data preparation

For each protein of the training set, we obtain all the information used by MECoMaP. First, we extract the amino acid sequences as well as the distances between these pairs of amino acids from PDB. As discussed in section 2.7, PDB is a repository of files containing information about various structures, including proteins. These files, one per protein, contain structural information of proteins and consist of several sections. Each section provides different features of the protein structure. In our case, we want to know the amino acid sequence of the protein and the distances between pairs of amino acids. In Primary Structure Section of each file, we can find the SEQRES record, containing a list of the consecutive chemical components that form a determined polymer. The following example corresponding to *1ego* protein whose structure is in the file *1ego.pdb*:

```
SEQRES 1 A 85 MET GLN THR VAL THR PHE GLY ARG SER GLY
SEQRES 2 A 85 CYS  VAL ARG ALA LYS  ASP LEU ALA GLU LYS
SEQRES 3 A 85 GLU  ARG ASP  ASP PHE GLN TYR GLN TYR VAL
SEQRES 4 A 85 ALA  GLU GLY ILE  THR LYS  GLU ASP  LEU GLN
SEQRES 5 A 85 GLY  LYS  PRO VAL GLU THR VAL PRO  GLN ILE
SEQRES 6 A 85 GLN  GLN HIS  ILE  GLY GLY TYR THR ASP PHE
SEQRES 7 A 85 VAL  LYS  GLU ASN LEU  ASP ALA
```

The first column identifies the entry as a SEQRES record, the second and third columns indicate a serial number and the amino acid chain (a protein may be formed by several chains) respectively, the forth column reports the total number of residues of the chain and the rest of the columns represent the succession of amino acids in the chain following the classification discussed in section 2.2.

In order to obtain the distances between pairs of amino acid, we can use the Coordinate Section in the PDB files, which includes information (in $\mathring{A}$) about the spatial coordinates of the atoms of the protein. An example of information stored in an ATOM record is the following:

```
ATOM 1 CA MET A 1 -8.629 -3.431 8.016 1.00 2.16 N
ATOM 2 N  MET A 1 -8.571 -3.525 6.539 1.00 1.46 C
ATOM 3 C  MET A 1 -7.219 -4.124 6.158 1.00 1.29 C
```

ATOM 4 O MET A 1 -6.329 -4.216 6.997 1.00 1.47 O
ATOM 5 CB MET A 1 -8.718 -2.135 5.906 1.00 1.29 C

The first column identifies the ATOM record, the second column indicates a serial number, and the third column represents the chemical component of the atom (*e.g.* CA for alpha carbon). The next three columns determine the type of amino acid, the chain of the protein and the position of this amino acid in the chain. The following three columns indicates the orthogonal coordinates of the atom in the x, y and z axis, respectively, in $\mathring{A}$. The occupancy values[3] and temperature factor[4] are indicated in the tenth and eleventh columns respectively. The last column represents the chemical symbol of the atom.

With this information, we calculate the distances between each pair of amino acids which form the protein chain. Two amino acids are considered to be in contact if their distance is lower than a determined threshold (generally $8\mathring{A}$). In order to calculate these distances, we select the respective alpha or beta carbons of the atoms, and use the Euclidean distance:

$$D_{ij} = \|r_i, r_j\| = \sqrt{((x_i - x_j)^2 + (y_i - y_j)^2 + (z_i - z_j)^2)}$$

where $r_i = (x_i, y_i, z_i)$ and $r_j = (x_j, y_j, z_j)$ are the geometrical center (C_β in this case) of amino acids i and j.

After processing the required PDB files, we produce two files per protein with the format <*protein_name*>*.seq* and <*protein_name*>*.dist*. The first file stores the amino acid sequence of the protein. For example, the amino acid sequence for protein *1ego* is stored as:

> 1EGO
MQTVIFGRSGCPYCVRAKDLAEKLSNERDDFQYQYVDIR
DLQQKAGKPVETVPQIFVDQQHIGGYTDFAAWVKENLDA

The dist file stores the distances between pairs of amino acid as shown in the following example:

85
1 2 3.812781924002473
1 3 6.922394744595254
1 4 10.50783607599585
1 5 13.68879023142659
1 6 16.89158574557166
1 7 19.51940478088407
...

[3]This value is used to indicate the fraction of molecules that have each of the conformations.

[4]An indicator of thermal motion about an atom.

where the first line represents the length of the protein sequence, and each line shows the position of each pair of amino acid in the sequence and the distance, in $\mathring{A}$, between them.

In the same way, secondary structure and solvent accessibility information are stored in files named *<protein_name>.psipred* and *<protein_name>.sapred*, respectively. An example of our *psipred* file is shown below:

C,9
C,1
E,2
E,8
E,3
E,2
C,1
C,9
H,4
H,6
H,8
H,8
H,6
H,6
C,7
C,9
...

where each line represents a residue of a determined protein sequence with the corresponding secondary structure prediction (C: coil, H: helix and E: strand), and a confidence factor (0=low, 9=high), which indicates the reliability of the prediction.

The format of our *sapred* file is also shown in the following example:

2344333323213303430332333233302000001111010000200200144313442 1...

where each position i indicates the SA prediction of residue i in the protein sequence. All the preparation of the data is summarized and represented in figure 5.2. After processing the *pdb* information, we obtain the sequence and distance files. PSIPred and SAPred return the predictions of SS and SA respectively. These predictions are stored in *psipred* and *sapred* files. All this information constitutes the input of our predictive algorithm MECoMaP.

In the following we address the solutions adopted for what regards the representation, the genetic operators and the fitness function used by the EA.

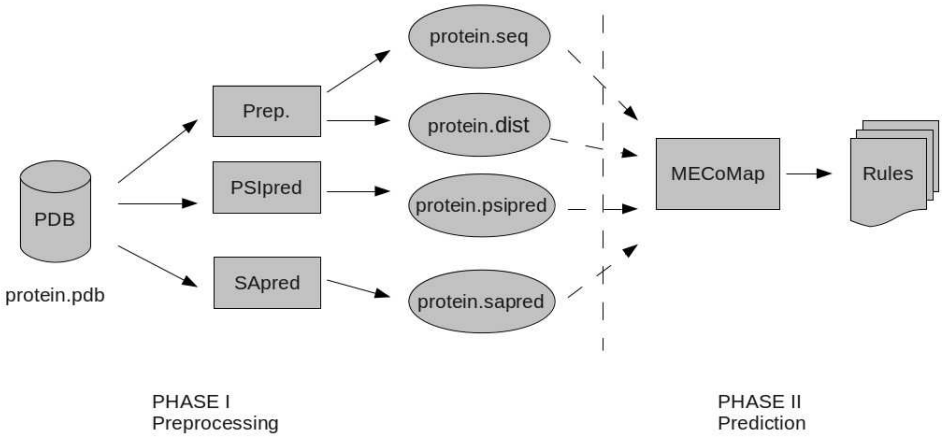


Figure 5.2: Preprocessing procedure scheme.

5.1.5 Encoding

Each individual in our algorithm represents a decision rule which determines whether amino acids i and j are in contact, with $1 \leq i < j \leq L$, where L is the sequence length. For this purpose, we use two windows of ± 3 residues centered around the two target amino acids i and j. Therefore, one window is relative to amino acids $i-3, i-2, i-1, i, i+1, i+2, i+3$ and the other one is relative to amino acids $j-3, j-2, j-1, j, j+1, j+2, j+3$. For each amino acid k belonging to the two windows, we define the descriptor Q_k (where $k \in \{i-3, i-2, i-1, i, i+1, i+2, i+3, j-3, j-2, j-1, j, j+1, j+2, j+3\}$) which represents a set of conditions for the amino acid k, as shown in equation 5.1.

$$Q_k = \{H_{min}, H_{max}, P_{min}, P_{max}, C, SS, SA\} \tag{5.1}$$
$$where$$
$$-1 \leq H_{min} < H_{max} \leq 1$$
$$-1 \leq P_{min} < P_{max} \leq 1$$
$$C \in \{-1, 0, 1\}$$
$$SS \in \{-1, 0, 1, 2\}$$
$$SA \in \{-1, 0, 1, 2, 3, 4\}$$

We define the decision rule $R_{i,j}$ for amino acids i and j, encoded in each individual of our algorithm, as shown in equation 5.2, for each $k \in \{i-3, i-2, i-1, i, i+1, i+2, i+3, j-3, j-2, j-1, j, j+1, j+2, j+3\}$.

$$R_{i,j} = \{Q_k\} \tag{5.2}$$

Given a test sequence $t_1 \ldots t_{L'}$, where L' is the test sequence length, and a pair of amino acids t_a and t_b ($1 \leq a < b \leq L'$), the algorithm predict a contact between these amino acids if there exist any rule $R_{i,j}$ ($1 \leq i < j \leq L$) that covers the pair (t_a, t_b).

A rule $R_{i,j}$ covers the pair (t_a, t_b) if that pair satisfies Q_k for all $k \in \{a-3, a-2, a-1, a, a+1, a+2, a+3, b-3, b-2, b-1, b, b+1, b+2, b+3\}$. The pair ($t_a$, t_b) satisfies Q_k if it fulfills the following equations for all k.

$$H_{min} \leq H(t_k) \leq H_{max}$$
$$P_{min} \leq P(t_k) \leq P_{max}$$
$$C(t_k) = C$$
$$SS(t_k) = SS$$
$$SA(t_k) = SA$$

where $H(t_k)$ is the hydrophobicity of the amino acid t_k, $P(t_k)$ its polarity, $C(t_k)$ its charge, $SS(t_k)$ its secondary structure and $SA(t_k)$ its solvent accessibility.

Figure 5.3 shows an example of an individual $R_{i,j}$. $R_{i,j}$ is constituted by 98 attributes (7 attributes per 14 amino acids). More specifically, an example of encoding for the element Q_i of an individual $R_{i,j}$ is shown in figure 5.4.

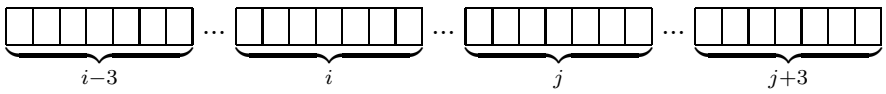


Figure 5.3: Example of a complete individual.

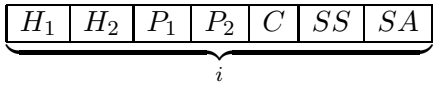
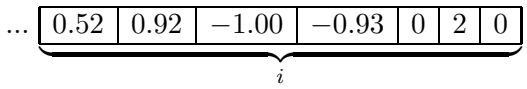


Figure 5.4: Example of encoding for the element Q_i of an individual $R_{i,j}$. H_1, H_2, P_1 and P_2 are lower and upper bounds for the hydrophobicity and polarity and volume values. C represents the charge value of the residue. SS represents secondary structure and SA indicates the solvent accessibility value.

Figure 5.5 shows an example of a generated $R_{i,j}$ individual and its translation into a decision rule. This rule indicates that if the hydrophobicity of amino acid in position i is between 0.52 and 0.92 and the SA value of amino acid in position $j+2$ is equal to 1, among other requirements, a contact is established.



$$if\ H_i \in [0.52, 0.92]\ and\ P_i \in [-1.00, -0.93]\ and$$
$$C_i = 0\ and\ SS_i = 2\ and\ SA_i = 0\ and$$
$$H_j \in [0.32, 0.82]\ and\ P_{j+1} \in [-0.41, -0.01]\ and$$
$$C_{j+1} = 0\ and\ SS_{j+1} = 2\ and\ SA_{j+2} = 1\ then\ contact$$

Figure 5.5: Example of a decision rule.

The main advantage of our representation is the easily interpretation of the generated decision rules by experts in the fields. The information extracted from these rules could provide useful insights into protein structure prediction problem. To the best of our knowledge, similar representations have not been considered in the literature.

5.1.6 Fitness Function

The aim of the algorithm is to find both general and precise rules for identifying residue-residue contacts. We consider two objectives to be optimized, rule coverage and rule accuracy. Rule coverage represents the proportion of contacts covered by each rule, while rule accuracy evaluates the correctly predicted contacts rate by each rule. Therefore, $Rule\ coverage = C/C_t$ and $Rule\ accuracy = C/C_p$, where C is the number of correctly predicted contacts of a protein, C_t is the total number of contacts of the protein and C_p is the number of predicted contacts. The fitness of an

individual is equal to the number of individuals that it dominates. The formula for the fitness function for an individual i is detailed in equation 5.3, where j represents an individual of the population and N represents the size of the population.

$$fitness(i) = \sum_{j}^{N} f(i,j) \tag{5.3}$$

The function $f(i,j)$ is described in equation 5.4 and calculates if the individual i dominates the individual j ($i \succeq j$) according to the two objectives used, where $i \succeq j$ *iff* $o_t(i) \geq o_t(j)$ for all $t = 1..M$, and $o_t(i) > o_t(j)$ for some $t \in 1..M$, where o_t are the objectives and M=2.

$$f(i,j) = \begin{cases} 1 \; if \; i \succeq j \\ 0 \; otherwise \end{cases} \tag{5.4}$$

5.1.7 Genetic Operators

The algorithm starts with a randomly initialized population and is run for a maximum number of generations. If the fitness of the best individual does not increase over twenty generations, the algorithm is stopped and a solution is provided. In order to obtain the next generation, individuals are selected with a tournament selection mechanism of size two. Crossover and mutation are then applied in order to generate offsprings.

Various crossover operators have been tested. In particular, we have tested the performances of one-point, two-points, uniform and BLX-α crossovers. These crossover operators act at the level of the amino acid properties. For instance, one-point crossover randomly selects a point inside two parents and then builds the offspring using one part of each parent. It follows that the resulting rule has to be tested for validity, since it could contain incorrect ranges. An example of one-point crossover is shown in figure 5.6.

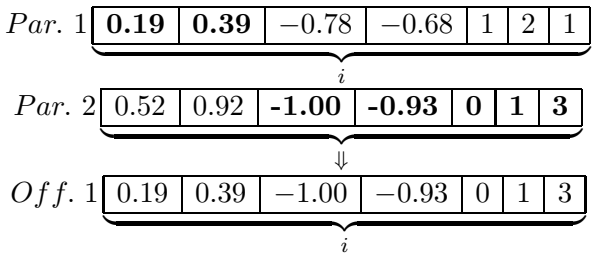


Figure 5.6: An example of one-point crossover for the element Q_i of two parent individuals $Par.1$ and $Par.2$ and the offspring individual $Off.1$. The random cut is established between H_2 and P_1.

BLX-α crossover creates a new offspring $R_{i,j}$, where the values of the elements of Q_k (for each $k \in \{i-3, i-2, i-1, i, i+1, i+2, i+3, j-3, j-2, j-1, j, j+1, j+2, j+3\}$) are mutated within an interval delimited by the maximum and minimum values of the two parent individuals for the same element of Q_k. This crossover operator can be seen as a linear combination of the two parents. After having performed several runs of the algorithm, the best results were obtained when the two-points crossover was used, which was then adopted as standard crossover in the algorithm.

We have applied two different mutation operators. The first operator, called Gaussian operator, mutates a randomly selected element of Q_k of an individual $R_{i,j}$, where $k \in \{i-3, i-2, i-1, i, i+1, i+2, i+3, j-3, j-2, j-1, j, j+1, j+2, j+3\}$. The value of this element is increased or decreased (for H and P) with a probability of 0.5. The increment value is randomly chosen between allowed ranges for each properties following a Gaussian distribution. If the values of a mutated individual are not within the allowed ranges for each properties, the mutation is discarded. For the charge, SS and SA, we randomly select a new value among the allowed ranges. An example of this mutation operator is shown in figure 5.7.

0.52	**0.75**	0.15	0.60	0	0	3	$\Rightarrow$	0.52	**0.95**	0.15	0.60	0	0	3
			i								i			

Figure 5.7: Example of Gaussian mutation for the element Q_i of an individual $R_{i,j}$ with an increment value of +0.2 for the H_2 property.

A second mutation operator, called Enlarge operator, randomly selects an element of Q_k of an individual, that is related to a given property, and varies its range, to all the allowed values. For instance, if the property is the hydrophobicity, this operator varies the range to [-1, 1]. This means that the rule does not take into account this property in this case. For the SS and SA elements, we set a negative value (-1) to indicate this constraint. An example of this mutation operator is represented in figure 5.8. Both types of mutation are applied to the entire population.

0.1	0.5	**0.38**	**0.50**	1	1	4	$\Rightarrow$	0.1	0.5	**-1.00**	**1.00**	1	1	4
			i								i			

Figure 5.8: Example of enlarge mutation operator for the element Q_i of an individual $R_{i,j}$ for P_1 and P_2 properties.

The parameter settings of the algorithm are shown in table 5.2. This setting was determined after several preliminary runs.

Table 5.2: Parameter setting used in the experiments.

Population size	100
Crossover probability	0.5
Gaussian mutation probability	0.5
Enlarge mutation probability	0.1
Max number of generations	100
Tournament size	2

5.1.8 Algorithm

The pseudocode of MECoMaP is shown in Algorithm 2. The evolutionary process is repeated $numIt$ times where $numIt$ is the number of iterations. The algorithm starts by randomly initialize the population. Then, it evaluates the current population P and the Pareto front is determined. Non-dominated solutions, which constitute the external population A, will be included in the population P' of the next generation. As already mentioned, four genetic operators are used: a binary tournament selection operator, a 2-point crossover operator and two mutation operators. The first 50% of the individuals in P' is formed by the non-dominated individuals (external population A) and by the selected individuals with the binary tournament selection operator. The other 50% of the individuals in P' is created using the 2-point crossover operator. Mutation is applied to the whole population, except to the Pareto front individuals, at the end of each generation. This process is repeated a maximum number of generations $maxGen$. The final set of rules ($Results$ in the code) is incrementally built. At the end of each generation, the algorithm adds to the final set the best rules found by the EA.

This is done in the following way: first, the best individual, according to its $F-measure$, is selected and added to the final solution. Then the next best individual is added, and the global $F-measure$ of the final solution is calculated. This process is repeated until the addition of a rule causes the $F-measure$ of the final solution to decrease. The $F-measure$ is defined as in equation 5.5:

$$Fmeasure = 2 \cdot \frac{Rule\ coverage \cdot Rule\ accuracy}{Rule\ coverage + Rule\ accuracy} \tag{5.5}$$

Repeated or redundant rules are not included in the final solution. Each pair of rules $(R_{a,b}, R_{c,d})$ is checked, where a, b and c, d are two pairs of amino acids in contact. If we find that $R_{a,b}$ is contained in $R_{c,d}$, then $R_{a,b}$ is removed from our final rule set. In this context, a rule $R_{a,b}$ is contained in another rule $R_{c,d}$ if the values of the elements of Q_k (for each $k \in [a-3, a+3] \cup [b-3, b+3]$) and the values of the elements of $Q_{k'}$ (for each $k' \in [c-3, c+3] \cup [d-3, d+3]$)

Algorithm 2 MECoMaP Algorithm for Contact Map Prediction

INPUT set of protein subsequences M, maximum number of iterations $numIt$, maximum number of generations $maxGen$, size of the population S.
OUTPUT set of generated rules $Results$.

```
begin
  num ← 0, Results ← ∅
  while (num < numIt) do
    Initialize P
    Evaluate population P
    i ← 0, A ← ∅
    while (i < maxGen) do
      Find Non-dominated solutions P
      Update Non-dominated solutions set A
      P' ← A
      P' ← Selection Method with binary tournament(P)
      P' ← 2-point Crossover Method with binary tournament(P)
      P' ← Mutation Method(P)
      P ← P'
      i ← i + 1
      Evaluate population P
    end while
    Results ← the best combination of rules from P
    num ← num + 1
  end while
  return Results
end
```

satisfy the conditions shown in equation 5.6.

$$\begin{gathered} H_{min} \geq H'_{min} \wedge H_{max} \leq H'_{max} \\ P_{min} \geq P'_{min} \wedge P_{max} \leq P'_{max} \\ C = C' \\ SS = SS' \\ SA = SA' \end{gathered} \tag{5.6}$$

5.1.9 Efficient evaluation of the individuals

In order to reduce the execution time of our method, we have implemented an AVL tree [Adelson-Velskii *et al.*, 1962] to order and classify the training examples according to their property values. A binary search tree is a binary tree with the property that all the elements stored in the left subtree of any node x are less than or equal to the item stored in x, and all the items stored in the right subtree of x are higher than the element stored in x. An AVL tree is a self-balancing binary search tree where the heights of the two child subtrees of any node differ by at most one. The time of the operations on an AVL tree is $O(logn)$ on average, where n is the number of elements. Each

node determines a condition of a property (e.g. hydrophobicity of amino acid i < 0) and each leave represents a list with the training examples that fulfills all the conditions imposed in the predecessor nodes. Each level of the tree represents a determined property of a determined position of an amino acid. An example of the described structure is represented in figure 5.9.

The main goal of the implementation of this structure is the reduction of time complexity of the algorithm by means of a fast evaluation of examples from the dataset. This tree organizes the information in such a way that it is not necessary to process all the examples to evaluate individuals (candidate decision rules) from the genetic population. This structure is also similar to the Efficient Evaluation Structure (EES) described in [Giraldez *et al.*, 2005].

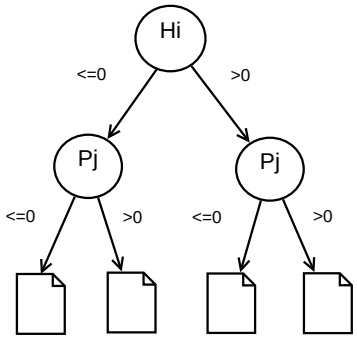


Figure 5.9: Example of AVL tree. Each leaf node represents a list with the training examples that fulfill the conditions imposed by its predecessor nodes, in this case they are referred to the hydrophobicity (H) of amino acid i and to the polarity (P) of amino acid j.

5.1.10 MECoMaP application

MECoMaP application is a Java multi-threading application (figure 5.10) which allows the user to easily generate contact map predictions from a protein training and test set (PDB files). On the Settings tab, the user can choose the input directory (protein data set) and output directory to store the results of the prediction. On the Options tab, the user can select the type of predictive algorithm. Error tab allows the user to select which type of prediction error measure will be calculated among absolute error $|d_i - d^*|$, relative $|d_i - d^*|/d^*$ and mean squared error $|d_i - d^*|^2$ where d_i corresponds to estimated distance, and d^* is the real distance between a pair of amino acids. On the Train&Test tab, the application allows users to select the type of validation to be used by the algorithm (cross-validation or leave-one-out). On PDB file tab, user can extract the required information from PDB files, such as distances and sequences of proteins of the training set, choosing a input directory of the PDB files. On the Genetic Algorithm

tab, the user can select the settings referred to the evolutionary algorithm. These options are: type and probability of crossover operator, probability of the mutation operator, number of individuals of the population, maximum number of generations, number of executions, minimum sequence separation between amino acids and number of folds for the validation.

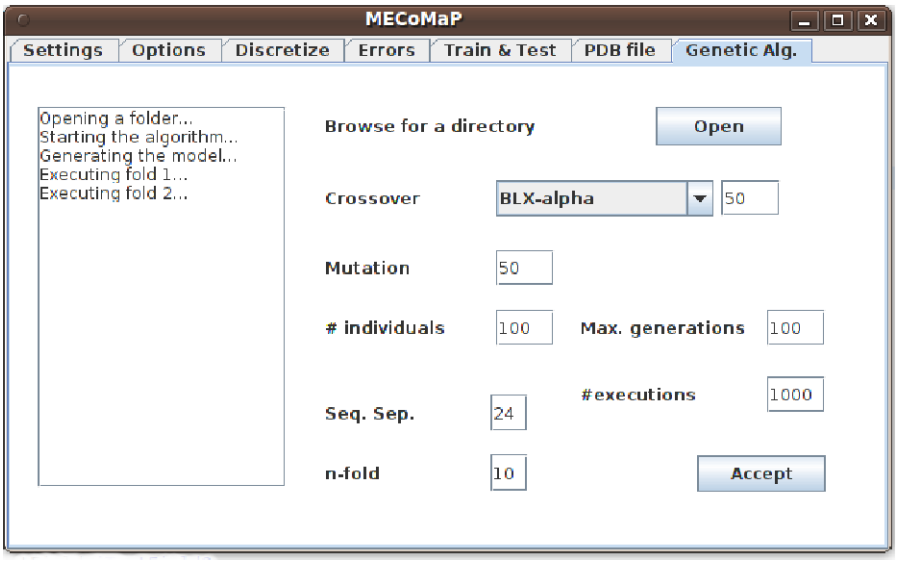


Figure 5.10: Screenshot of MECoMaP application.

The application also generates a graphical representation of the results, such as contact and distance maps.

An example of a generated distance map for $1A7GE$ protein is shown in figure 5.11. In such distance map, the X and Y-axis represent the protein sequence and each cell of the matrix shows the distance between a pair of amino acids represented by a different color. The ranges of distances and its corresponding color, are shown in the legend of the distance map. So, a cell with red color represents a contact or proximity of contact for a determined pair of amino acids. On the other hand, a blue cell represents a high distance in angstroms and consequently a non-contact. The upper triangle of the matrix corresponds to real distances and the lower triangle represents the estimated distances.

On the Discretize tab, the user can select the type of discretization for the distance or contact map representation among discretization by frequency, by width or no discretization. The number of intervals of distances for the legend of distance map representation can also be selected.

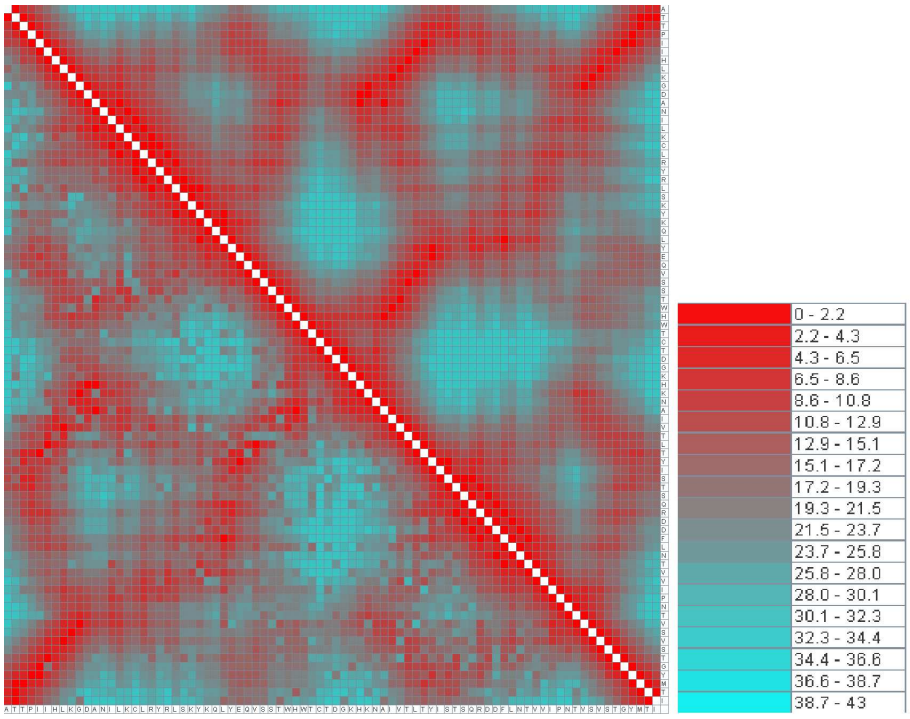


Figure 5.11: Example of a generated distance map for 1A7GE protein by MECoMaP application. Distances in angstroms.

5.2 Protein Secondary Structure Predictor based on Evolutionary Computation

In this section, we describe our proposal for predicting secondary structure (α-helices and β-sheets) from sequences of amino acids. We believe that EAs are good candidates for tackling this problem. In fact, secondary structure prediction problem can be seen as a search problem, where the search space is represented by all the possible folding rules. Such a space is very complex, and has huge size. EAs have proven to be particularly good in this kind of domains, due to their search ability and their capability of escaping from local optima.

In our proposal, prediction is made *ab initio*, i.e., without any known protein structure as a starting template of the search. The result of the algorithm is a model for the prediction of the beginning and the end of regions in the amino acid sequence that correspond to either an α-helix or to a β-strand. In particular, the model corresponds to a set of decision rules.

Existing methods, typically fail at predicting motifs boundaries. [Wilson *et al.*, 2002]. In particular, β-sheet determination is more difficult to predict than an α-helix [FarzadFard *et al.*, 2008].

5.2.1 Methodology

Our proposal identifies α-helices and β-strands within a sequence of amino acids. Both α-helix and β-strand, are a subsequence of amino acids. Figure 5.12 shows an example of a sequence that may correspond to an α-helix. Each amino acid in the sequence is identified by its position, being amino acids in positions N-cap and C-cap those that immediately precede or follow the beginning or the end of the structure, respectively. So the aim of our proposal is to predict whether or not an amino acid lies in either position N-cap or C-cap.

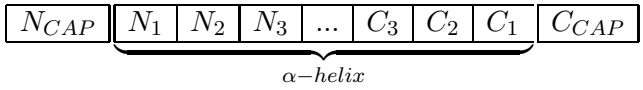


Figure 5.12: Relevant positions in an α-helix.

The experimental procedure to perform is represented in figure 5.13. During the data acquisition stage, the α-helix and β-strand sequences are extracted from the Protein Data Bank (PDB) [1]. Later, these sequences are used for training our evolutionary algorithm. Finally, a set of rules are generated. These rules specify different beginnings and ends of the motifs. We test our algorithm by applying these rules to a set of known protein sequences, thus used as test set.

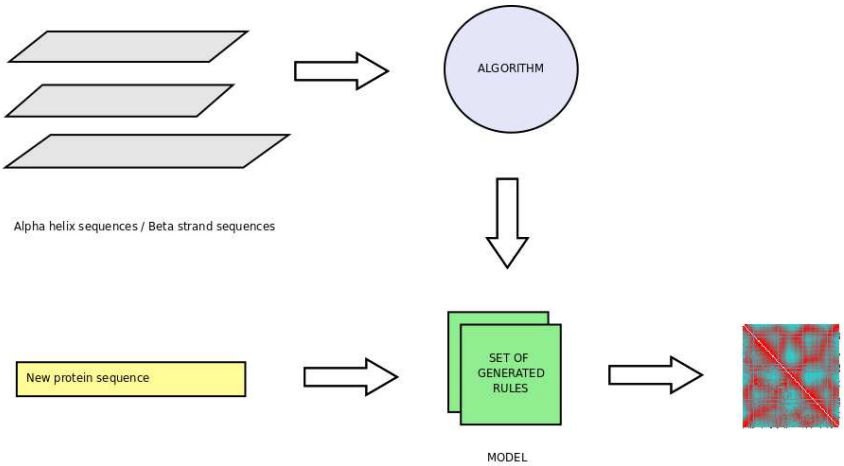


Figure 5.13: Experimental and prediction procedure.

In the following we discuss the various solutions we adopted for what regards the fitness, the representation and the genetic operators used.

5.2.2 Encoding

Each individual of the population represents a window of 2-amino acids. An individual may represent either the beginning or the end of an α-helix or a β-sheet (N-cap, N_1 or C_1, C-cap positions).

For each position we consider the same properties used in the tertiary structure prediction, *i.e.*, hydrophobicity, polarity and charge. We used the same normalized values reported in table 5.1.

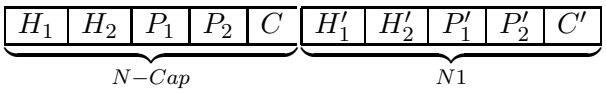


Figure 5.14: Example of encoded chromosome for a beginning of an α-helix or a β-strand.

Figure 5.14 presents an example of individual. Positions H_1, H_2, H'_1, H'_2 represent the hydrophobicity values of the first and second amino acid of the window respectively, e.g., H_1, H_2 represent an interval of real numbers which determine the hydrophobicity values for the amino acid in N-cap position. Positions P_1, P_2, P'_1, P'_2 represent the polarity values according to Grant scale of the first and second amino acid respectively, e.g., P_1, P_2 represent an interval of real numbers which determine the polarity values for the amino acid in N-cap position. Positions C and C' represents the net charge values of the two amino acid.

5.2.3 Fitness Function

We have chosen as fitness of individuals the F-measure (see equation 5.5) to evaluate the rules for the prediction of beginnings and ends of secondary motifs. The higher the fitness, the better the individual.

Furthermore, we also consider some physical-chemical properties (polarity and charge) information of the amino acids in positions N-cap, N_1 or C_1, C-cap, if the rule is relative to a beginning or an end of a motif, respectively. It has been demonstrated that there are molecules with asymmetrical distributions of charge in the limits of a α-helix [Richardson *et al.*, 1998]. This means that the residues in limits of the helix are polar, so the fitness of these individuals is increased. Moreover, in [Doig *et al.*, 1995, Fonseca *et al.*, 2007], it has been proven that many helices present a positive charge in its last turn and a negative charge at its first turn. On the other hand, we also consider some specifications for the β-sheet capping prediction. It has been demonstrated that hydrophobic amino acids have a high propensity to be at N_1 and C_1 positions (especially V, I, Y, and W) in a β-sheet. In addition, many strands present a negative charge in C-cap and a positive charge in N-cap positions [FarzadFard *et al.*, 2008].

We increase the score of those individuals that fulfill one requirements in a 50%, and in a 100% for those individuals that present the two properties. In this manner, those individuals which are more likely to belong to a structural motif, will have a higher fitness value.

5.2.4 Genetic Operators

Individuals are selected with a roulette wheel mechanism (see section 3.2.3). Elitism is applied, thus the best individual of the population is always preserved in the next generation.

Uniform crossover is used in order to generate offsprings. Crossover is applied with a 1.0 probability. Mutation is applied with a probability of 0.5. If mutation is applied, one gene of the individual is randomly selected, and its value is increased or decreased by 0.01. If the selected gene is relative to the charge of the amino acid, then its value is randomly changed to one of the other two allowed possibilities. After that an individual has been mutated, it is checked for validity, i.e., its values are within the ranges allowed for each properties [-1,1] for Hydrophobicity and Polarity and -1,0 or 1 for the Net charge. If the encoded rule is not valid, then the mutation is discarded.

5.2.5 Algorithm

Algorithm 3 Algorithm for Secondary Structure Prediction

INPUT set of protein subsequences M, maximum number of iterations $numIt$, maximum number of generations $maxGen$, size of the population S.
OUTPUT set of generated rules $Results$.

begin
  $num \leftarrow 0$, $Results \leftarrow \emptyset$
  while $(num < numIt)$ **do**
    Initialize P
    Evaluate population P
    $i \leftarrow 0$
    while $(i < maxGen)$ **do**
      $P' \leftarrow$ Elitist Selection(P)
      $P' \leftarrow$ Roulette wheel Selection(P)
      $P' \leftarrow$ Uniform Crossover(P)
      $P' \leftarrow$ Mutation Method(P)
      $P \leftarrow P'$
      $i \leftarrow i + 1$
      Evaluate population P
    end while
    $Results \leftarrow$ the best combination of rules from P
    $num \leftarrow num + 1$
  end while
  return $Results$
end

The pseudocode of this proposal is shown in algorithm 3. The evolutionary process is repeated $numIt$ times where $numIt$ is the number of iterations. In the algorithm, the population size of the algorithm is set to 100. At the beginning of the algorithm, the initial population is randomly initialized. After having evaluated the initial population, the first generation is created by using the genetic operators described in the previous section.

This process is repeated a maximum number of generations $maxGen$. If the fitness of the best individual does not increase over twenty generations, the algorithm is stopped and a solution is provided.

At the end of the evolutionary process, the best individuals from each populations are extracted, and together they form the proposed solution. *Results* set is formed in an incrementally way, as for MECoMaP, and constitutes the output of our algorithm. Repeated or redundant rules are not included in the final solution.

The optimal number of rules necessary for the prediction is unknown. For this reason, we tested with a different number of iterations of the algorithm, more specifically from 10 to $1,000$. In order to select the rules that will form the solution proposed by the algorithm, those rules identified by the a higher value of the F-measure are selected.

In the experiments proposed, we used the following parameters for the EA. The population size is set to 100. Crossover and mutation probabilities are set to 1.0 and 0.5, respectively. The maximum number generations is set to 100.

Four populations are evolved separately: one contains individuals that encode rules identifying the beginning of an α-helix, while the others contain individuals representing rules for the end of the helix, encodes rules identifying the beginning of a β-strand, and individuals representing rules for the end of a β-strand.

5.3 Summary and conclusions

In this chapter, we have detailed the methodology of our two evolutionary proposals for PSP: an evolutionary contact map predictor, called MECoMaP and a secondary structure predictor which predicts the starts and ends of SS motifs. Our first proposal is a multi-objective evolutionary contact map predictor based on physico-chemical properties of the amino acids as well as two structural features of residues (SS and SA). We have detailed the preparation of the training and test datasets and the features of the evolutionary components of the algorithm (encoding, fitness function and genetic operators). A pseudocode of the algorithm is also provided. A description about a new efficient evaluation of the individuals is also included in this part of the chapter. In the second part of this chapter, we describe our protein secondary structure predictor based on evolutionary computation.

The encoding, fitness function, and genetic operators are also described. The details of the implementation of the evolutionary algorithm are included at the end of the chapter. We can conclude that both two proposals are based on evolutionary computation which are an appropriate way to tackle this kind of problems, as it was detailed in chapter 1. The prediction is based on a set of amino acid properties which are very important in the folding process. The output of the algorithm consist of a set of rules that can easily be interpreted and analyzed by experts in the field. The fusion of all these features represent, as far as we concerned, a novel and significant methodology in the PSP field.

Part IV

Results

Chapter 6

Tertiary structure prediction experiments

6.1 MECoMaP results

In this chapter we will present the results obtained by MECoMaP. We have performed experiments on four different datasets, two preliminary studies and an analysis of generated predicting rules.

6.1.1 Data sets

The first protein data set (DS1) [Fariselli *et al.*, 2001] consists of 173 non-redundant proteins with sequence identity lower than 25%. As in [Fariselli *et al.*, 2001], four subsets have been obtained according to the sequence length (L_s): $L_s < 100$, $100 <= L_s < 170$, $170 <= L_s < 300$, $L_s >= 300$. The minimum and maximum lengths of proteins are 31 and 753 amino acids, respectively.

The second data set (DS2) comprehends 53 non-redundant and non-homologous globulin proteins and is detailed in [Cheng and Baldi, 2007]. In this case, proteins are classified according to their SCOP class [Murzin *et al.*, 1995] as described in chapter 2. The sequence identity of DS2 dataset is also lower than 25%. The minimum and maximum lengths of proteins are 52 and 198 amino acids, respectively.

The third data set we used, is described in [Zhang *et al.*, 2005]. This data set (DS3) includes 48 non-homologous proteins. DS3 is divided into five subsets according to L_s: $L_s < 100$, $100 <= L_s < 200$, $200 <= L_s < 300$, $300 <= L_s < 400$, $L_s >= 400$. The minimum and maximum lengths of proteins are, in this case, 98 and 458 amino acids, respectively.

The forth data set (DS4) is detailed in [Jones *et al.*, 2012]. A total of 150 non-homologous proteins are contained in this data set. The sequence length of the proteins varies between 50 and 275 amino acids.

All the experimentations were performed under the same conditions that appeared in the cited articles. A threshold of 8 angstroms (Å) was established to determine a contact as in [Fariselli *et al.*, 2001]. In order to avoid the effect of learning local contacts, we set the same minimum sequence separation between each pair of amino acids to establish a contact as in the reference works. In Appendix C, we provide four tables with the PDB code of all the proteins used in the experiments.

6.1.2 Preliminary studies

Before presenting the results obtained by our algorithm on the datasets, we present results of two preliminary studies. The first study was conducted in order to determine the distribution of the number of contacts according to the sequence separation between the pairs of amino acids. In this study, we have used the protein data set DS1. The result of this study is shown in figure 6.1. The X-axis represents the different possible values of sequence separations (the number of residues between those that are in contact) and the Y-axis represents the number of residue-residue contacts. From this graph, we can conclude that the vast majority (97%) of contact occurrences are established with a sequence separation lower than 140 amino acids. Therefore, we discard all the possible contacts with a sequence separation higher than 140 during the training phase in all the experimentations. Using this constraint, we can considerably reduce the computational time. Similar contact distributions were obtained for the other datasets.

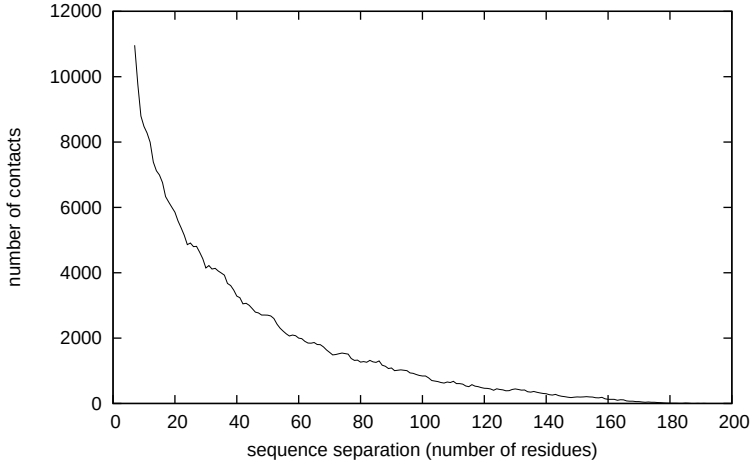


Figure 6.1: Sequence separation vs. number of contacts.

The other preliminary experiment we present, is aimed at verifying if the encoding adopted by MECoMaP provides enough information in order to perform a good classification. To this aim, we have compared the results

obtained by MECoMaP with those obtained by five well known classifiers: Naive Bayes (NB), C4.5 classifier tree, Nearest Neighbor approach with $k = 1$ (IB1), Neural Network (ANN) and Support Vector Machine (SVM). For this experimentation we have used DS1 ($L_s < 100$), DS2, DS3 and DS4. We have set the same experimental conditions in all the cases: a minimum sequence separation of 6 amino acids and a 3-fold cross-validation was performed. From all the extracted data, we have built four files in ARFF format, with all the training data information. The positive class (contact) is represented with 1 and the negative class (no contact) is represented with 0. Table 6.1 specifies the number of examples contained in each data set as well as the number of positive (contact) and negative (non contact) examples.

Table 6.1: Number of total, positive (class=1) and negative (class=0) examples in the four data sets for the WEKA experimentation.

Data set	#Pos. ex.	#Neg. ex.	# Total
DS1	6922	117027	123949
DS2	5530	166386	171916
DS3	18486	37502	55988
DS4	44444	1119751	1512823

We have used the WEKA [Hall *et al.*, 2009] implementation of C4.5 (J48), Naive Bayes (NB), IB1, Multilayer Perceptron (ANN) and Sequential Minimal Optimization algorithm (SMO) which represents a SVM. We used the default setting of the algorithms.

Table 6.2 shows the results of this experiment. The results obtained by MECoMaP are within normal values of accuracy and coverage rates for the contact map prediction [Cheng and Baldi, 2007]. These results confirm that our encoding provides enough information for a good performance of a learning classifier. Furthermore, we can also notice that MECoMaP achieved the best results for this experiment in the majority of the cases. High values of coverage are achieved by NB for DS2 and DS4, however, the accuracy rate is significantly low in these cases, so these results are overcome on average by our algorithm. On the other hand, NN achieves higher values of accuracy than MECoMaP on DS3, but its coverage is much lower than the coverage obtained by our method. In addition, we performed a statistical test (Friedman test) on the results shown in table 6.2 and we found that the differences of accuracy values for DS2 and DS4 are statistically significant.

Table 6.2: Average accuracy, coverage and standard deviation values obtained for different Weka classification algorithms for the DS1, DS2, DS3 and DS4 protein data sets with the same experimental settings.

Methods	DS1 Data Set		DS2 Data Set		DS3 Data Set		DS4 Data Set	
	Acc.$_{\mu\pm\sigma}$	Cov.$_{\mu\pm\sigma}$	Acc.$_{\mu\pm\sigma}$	Cov.$_{\mu\pm\sigma}$	Acc.$_{\mu\pm\sigma}$	Cov.$_{\mu\pm\sigma}$	Acc.$_{\mu\pm\sigma}$	Cov.$_{\mu\pm\sigma}$
IB1	$0.11_{\pm 0.37}$	$0.11_{\pm 0.27}$	$0.05_{\pm 0.01}$	$0.05_{\pm 0.01}$	$0.08_{\pm 0.00}$	$0.08_{\pm 0.00}$	$0.07_{\pm 0.00}$	$0.07_{\pm 0.00}$
J48	$0.33_{\pm 0.31}$	$0.03_{\pm 0.22}$	$0.10_{\pm 0.05}$	$0.08_{\pm 0.04}$	$0.35_{\pm 0.03}$	$0.41_{\pm 0.02}$	$0.08_{\pm 0.01}$	$0.07_{\pm 0.01}$
NB	$0.14_{\pm 0.34}$	$0.20_{\pm 0.39}$	$0.09_{\pm 0.02}$	$0.20_{\pm 0.02}$	$0.19_{\pm 0.02}$	$0.05_{\pm 0.08}$	$0.08_{\pm 0.02}$	$0.32_{\pm 0.03}$
NN	$0.24_{\pm 0.05}$	$0.10_{\pm 0.02}$	$0.12_{\pm 0.07}$	$0.04_{\pm 0.27}$	$0.42_{\pm 0.01}$	$0.07_{\pm 0.01}$	$0.12_{\pm 0.00}$	$0.03_{\pm 0.00}$
SMO	$0.14_{\pm 0.14}$	$0.03_{\pm 0.09}$	$0.04_{\pm 0.02}$	$0.06_{\pm 0.03}$	$0.08_{\pm 0.01}$	$0.09_{\pm 0.01}$	$0.04_{\pm 0.01}$	$0.09_{\pm 0.01}$
MECoMaP	$0.54_{\pm 0.24}$	$0.21_{\pm 0.18}$	$0.38_{\pm 0.09}$	$0.12_{\pm 0.01}$	$0.37_{\pm 0.08}$	$0.39_{\pm 0.02}$	$0.36_{\pm 0.09}$	$0.11_{\pm 0.03}$

In order to further verify the correct performance of our approach, in figure 6.2 we show the different Pareto fronts for ten generations (from generation 10 to 100 with an interval of 10) of a typical execution of the algorithm on DS1 ($L_s < 100$). Each different symbol represents an individual of the Pareto front in different generations. The X-axis represents the coverage and the Y-axis shows the accuracy rate. These two measures are the two objectives subject of optimization. We can notice how the quality of individuals improve over the generations. This result confirms that the algorithm is optimizing the two objectives.

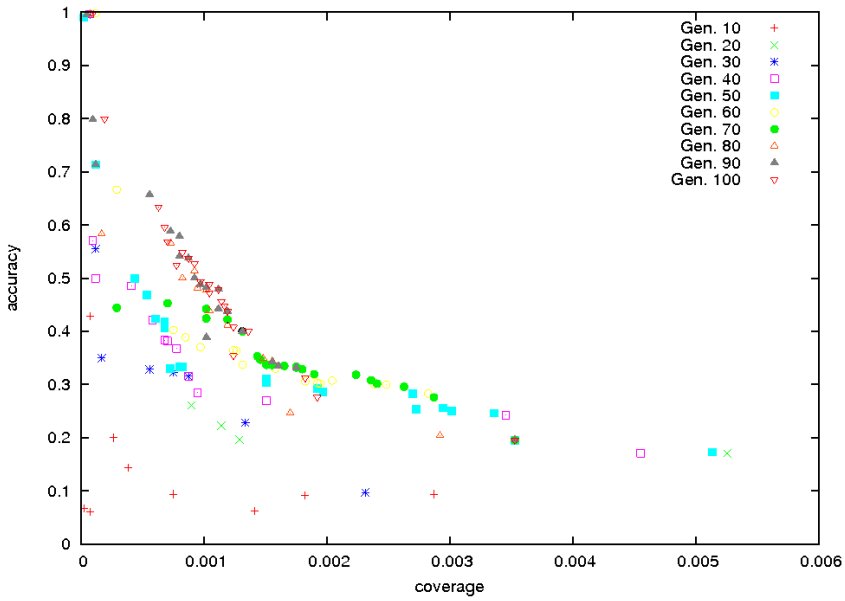


Figure 6.2: Pareto fronts for an execution in different generations.

Since the algorithm incrementally adds decision rules to a final set of rules, and since the optimal and exact number of rules is unknown, we have performed various experiments varying the numbers of runs of the EA, where to a higher number of runs corresponds a higher number of rules. The aim of this was to test whether or not a higher number of rules would yield better results. From these, we have concluded that the best results were obtained when the algorithm was run for 1000 iterations.

In the following experiments we employ three statistical measures which are used in CASP competitions [Monastyrskyy *et al.*, 2011], *i.e.*, coverage, accuracy and X_d, to evaluate the performance of protein structure predictors. In particular, in CASP, coverage indicates what percentage of contacts have been correctly identified. Accuracy reflects the number of correctly predicted contacts. X_d represents the distribution accuracy of the predicted contacts. X_d is defined by equation 6.1

$$X_d = \sum_{i=1}^{15} \frac{P_i - P_a}{i} \tag{6.1}$$

where P_i represents the percentage of predicted pairs with a distance between $4(i-1)$ and $4i$ and P_a represents the percentage of total pairs with a distance between $4(i-1)$ and $4i$. Coverage represents the number of predicted contacts divided by the number of desired contacts.

MECoMaPwas implemented in Java using a multithreading architecture. Furthermore, due to the enormous volume of data, all the experiments were run on a 64-bit workstation, with 32 GB DDR SDRAM and four dual-core processors. We have also performed four experiments on three different datasets described in the following.

6.1.3 First experimentation

Table 6.3 shows the results obtained on DS1. As in [Fariselli *et al.*, 2001], we used a minimum sequence separation of 7 residues and a 3-fold cross-validation. We have compared our results with the ones reported in [Fariselli *et al.*, 2001] using the same data set. The first column of the table reports the sequence length range of each subset of proteins, while the second column represents the number of proteins of each subset. The third column shows the average accuracy rate obtained by MECoMaP, and finally, the fourth column presents the average accuracy rate obtained by the reference algorithm [Fariselli *et al.*, 2001]. Standard deviation for accuracy is also reported. We can notice how the accuracy rate decreases when the length of the sequences increases. This is due to the fact that, generally, *ab initio* methods only work well with peptides shorter than 150 amino acids [Fernandez *et al.*, 2009]. MECoMaP obtains better results than [Fariselli *et al.*, 2001] for proteins whose sequence length is lower than or equal to 100. We have obtained the same accuracy rates for the second subset ($100 \leq L < 170$), and similar accuracy rates for the third ($170 \leq L < 300$) and fourth group ($L \geq 300$).

Low values of standard deviation show us that our data results are not significantly spread compared to the results obtained by [Fariselli *et al.*, 2001]. Additionally, we performed a statistical test (Friedman test) with these values and we found that these differences are statistically significant. Positive values of X_d are achieved in all the cases. Therefore our predictor improves the performance of a random predictor (negative values of X_d).

Table 6.3: Efficiency of our method predicting DS1 protein data set.

		MECoMaP	[Fariselli *et al.*, 2001]
Protein length	#prot.	Acc.$_{\mu\pm\sigma}$	Acc.$_{\mu\pm\sigma}$
$L \leq 100$	65	$0.54_{\pm 0.24}$	$0.26_{\pm 0.39}$
$100 \leq L < 170$	57	$0.21_{\pm 0.16}$	$0.21_{\pm 0.32}$
$170 \leq L < 300$	30	$0.17_{\pm 0.08}$	$0.15_{\pm 0.22}$
$L \geq 300$	21	$0.10_{\pm 0.05}$	$0.11_{\pm 0.15}$
All proteins	173	$0.26_{\pm 0.13}$	$0.18_{\pm 0.32}$

6.1.4 Second experimentation

Table 6.4 presents the results for the DS2 data set. As in [Cheng and Baldi, 2007], a minimum sequence separation of 6 residues was used. The first column of the table presents the SCOP classification. The second column shows the number of proteins of each subset, and the third and fourth column show the average accuracy rate obtained by MECoMaP and by the algorithm presented in [Cheng and Baldi, 2007], respectively. Standard deviation is also reported. From this table, we can observe that MECoMaP achieves good results for the beta proteins prediction. This is explained by the fact that most of the rules generated by our algorithms predict β-sheets. This observation will be validated in a further analysis of the extracted rules presented in chapter 7. We also obtain better accuracy for the alpha, small and coil-coil classes. Also in these cases, we have performed a non-parametric statistical test (Friedman test) on the results. After executing the test, the obtained p-value was 0.025 (p-value < 0.05), so that the null hypothesis was rejected, thus the differences of the results are statistically significant. We have also obtained positive values of X_d in all the cases.

6.1.5 Third experimentation

A third experiment compares our proposal with a neural network method (RBFNN) proposed in [Zhang *et al.*, 2005]. This method used the protein data set DS3. RBFNN uses an input and hidden layer with 20 nodes, the learning rate is set to 0.01, and the goal rate is set to 0.001. Coverage was used to evaluate the performance of RBFNN. We used the same experiment settings as in [Zhang *et al.*, 2005]. The dataset is divided in five subdatasets according to the sequence length. Table 6.5 shows the results of this experiment. The first column indicates the sequence length of each subset. The second column shows the number of proteins contained in each dataset. The third column represents the coverage rate of our algorithm and the

Table 6.4: Efficiency of our method predicting DS2 protein data set.

		MECoMaP	[Cheng and Baldi, 2007]
SCOP Class	#prot.	$\text{Acc.}_{\mu\pm\sigma}$	$\text{Acc.}_{\mu\pm\sigma}$
alpha	11	$0.30_{\pm 0.13}$	$0.24_{\pm 0.13}$
beta	10	$0.42_{\pm 0.12}$	$0.38_{\pm 0.16}$
$a+b$	15	$0.38_{\pm 0.09}$	$0.45_{\pm 0.10}$
a/b	7	$0.22_{\pm 0.05}$	$0.37_{\pm 0.07}$
small	4	$0.50_{\pm 0.01}$	$0.36_{\pm 0.07}$
coil-coil	1	$0.25_{\pm 0.00}$	$0.22_{\pm 0.00}$
All proteins	48	$0.38_{\pm 0.08}$	$0.37_{\pm 0.14}$

forth column represents the coverage rate of RBFNN. As we can see from this table, the average coverage is largely improved by our method in all the cases. Only on the third subset MECoMaP obtained worse results. This is due to the fact that only one protein is used as training set in this case and it seems to be insufficient to build an effective knowledge model.

Table 6.5: Efficiency of our method predicting DS3 protein data set.

		MECoMaP	[Zhang *et al.*, 2005]
Protein length	#prot.	$\text{Cov.}_{\mu\pm\sigma}$	Cov.
$L \leq 100$	10	$0.41_{\pm 0.12}$	0.26
$100 \leq L < 200$	13	$0.62_{\pm 0.15}$	0.30
$200 \leq L < 300$	2	$0.15_{\pm 0.13}$	0.31
$300 \leq L < 400$	13	$0.29_{\pm 0.10}$	0.26
$L \geq 400$	9	$0.75_{\pm 0.08}$	0.26
All proteins	48	$0.44_{\pm 0.12}$	0.27

6.1.6 Fourth experimentation

The vast majority of methods in PSP use evolutionary information, such as multiple sequence alignments or PSSM matrices (*e.g.* [Fariselli and Casadio, 1999, Zhao and Karypis, 2002, Xue *et al.*, 2009]). The use of this information, can help obtaining a significant increment of the accuracy in the predictions. To confirm this information, we have performed a experimentation in WEKA. We have analyzed the effect of using different attributes for the same DS1 training examples. The first training dataset only includes physico-chemical attributes: hydrophobicity, polarity and net charge (H, P and C). A second dataset includes the

cited physico-chemical and two attributes related to structural features: secondary structure and solvent accessibility. The third dataset also includes all the previous attributes and PSSM attributes. The results are

Table 6.6: Efficiency of Ib1 WEKA classifier predicting DS1 protein dataset with different sets of attributes.

Method	Attributes	Acc.$_{\mu\pm\sigma}$	Cov.$_{\mu\pm\sigma}$
IB1	H,P,C	$0.087_{\pm 0.05}$	$0.084_{\pm 0.08}$
IB1	H,P,C,SS,SA	$0.115_{\pm 0.07}$	$0.112_{\pm 0.09}$
IB1	H,P,C,SS,SA,PSSM	$0.169_{\pm 0.08}$	$0.164_{\pm 0.12}$

shown in table 6.6. For the experimentation, we have used IB1 WEKA classifier. We can notice an improvement of the results when PSSM attributes are used, with an increment of 5.4 percentage points in accuracy regarding the second dataset and 7 percentage points regarding the first dataset. We can also notice that the algorithm obtains better results with the second set of attributes than with the first set. Motivated by these results, we have included PSSM information as a new attribute in a second version of our algorithm. We have calculated the PSSM matrix for each protein of the training set using PSIBLAST. These PSSM values represent the substitution frequencies at a given position divided by the expected substitution frequency for a determined amino acid. According to the formalization described in section 5.1, we define a new descriptor Q_k (where $k \in \{i{-}3, i{-}2, i{-}1, i, i{+}1, i{+}2, i{+}3, j{-}3, j{-}2, j{-}1, j, j{+}1, j{+}2, j{+}3\}$) which represents a set of conditions for the amino acid k, as shown in equation 6.2. This descriptor determines the new encoding of the algorithm, where $PSSM_{min}$ and $PSSM_{max}$ represent the minimum and maximum PSSM values for each residue k and for the 20 different amino acids. These values are normalized between -1 and 1.

$$
\begin{gathered}
Q_k = \{H_{min}, H_{max}, P_{min}, P_{max}, C, SS, SA, PSSM_{min}^{1..20}, PSSM_{max}^{1..20}\} \\
where \\
-1 \leq H_{min} < H_{max} \leq 1 \\
-1 \leq P_{min} < P_{max} \leq 1 \\
C \in \{-1, 0, 1\} \\
SS \in \{-1, 0, 1, 2\} \\
SA \in \{-1, 0, 1, 2, 3, 4\} \\
-1 \leq PSSM_{min}^{1..20} < PSSM_{max}^{1..20} \leq 1
\end{gathered}
\tag{6.2}
$$

Table 6.7 shows the obtained results for DS1, using the new encoding and the same experimental settings of the first experiment. It can be noticed that the accuracy values have been increased in all the cases comparing with the results in table 6.3. This study confirms the importance of evolutionary information in PSP. After executing a non-parametric statistical test (Friedman test) on the results, we obtain that the differences are statistically significant (p-value = 0.025).

Table 6.7: Efficiency of our method predicting DS1 protein data set including PSSM attributtes.

Protein length	#prot.	Acc.$_{\mu\pm\sigma}$
$L \leq 100$	65	$0.61_{\pm 0.25}$
$100 \leq L < 170$	57	$0.25_{\pm 0.15}$
$170 \leq L < 300$	30	$0.20_{\pm 0.11}$
$L \geq 300$	21	$0.13_{\pm 0.10}$
All proteins	173	$0.29_{\pm 0.15}$

Figure 6.3: Generated contact map of protein 5PTI.

At the end of the execution, the program generates a resulting contact map for each protein test. In figure 6.3, we show an example for the protein 5PTI from DS1 data set. We can appreciate that the lower triangular (predicted contacts) is largely similar to the upper one (real contacts).

6.2 Analysis of MECoMaP predicting rules

In order to further evaluate the results obtained by MECoMaP, we have analyzed the set of rules obtained on DS1. A set of 10,244 rules were extracted by the algorithm after an execution of 1,000 iterations on DS1. With this study, we aim at analyzing the properties of the amino acids that are predicted to be in contact. These residues are identified by i and j. This would allow us to draw conclusions about the influence that these properties have on the protein folding problem.

First, we have analyzed the properties of the amino acid i in the rules set. The histograms in figures 6.4 and 6.5 show the relative frequency of hydrophobicity and polarity values for the amino acid i. The properties values have been discretized into five groups in intervals of 0.5 from -1 to 1 for the hydrophobicity and polarity. In these figures, each interval I contains all the rules whose interval $[Hmin, Hmax]$ (for hydrophobicity and polarity) is totally or partially included in I. Therefore, note that the same rule could be contained in one or more groups. Although all the study is referred to target residue i, the target residue j presents similar behavior.

From the graphs, we can notice that a vast majority of amino acids in contact present high values of hydrophobicity. Furthermore, a high percentage of contacts have non-polar residues. These conclusions were expected, because hydrophobic and non-polar amino acids tend to be located in the core of the protein. The core of proteins contains much less space than other protein regions, and contacts among amino acids are more frequent. Therefore, these type of residues have more probabilities to be in contact [Gupta *et al.*, 2005]. We have not observed any clear conclusion regarding the net charge of amino acids i and j individually.

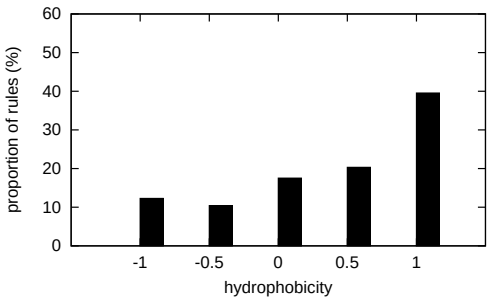


Figure 6.4: Relative frequency of hydrophobicity values for amino acid i in our predicted rules.

Figure 6.6 and 6.7 represent the relative frequencies of values of solvent accessibility and secondary structure respectively. As we can see in figure 6.6, lower values of solvent accessibility are the most represented values

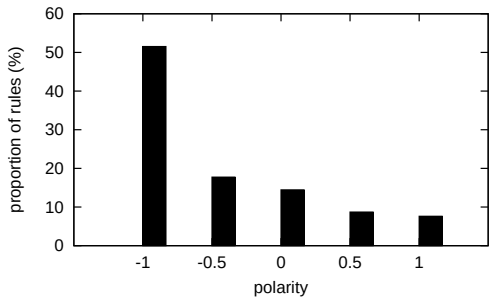


Figure 6.5: Relative frequency of polarity values for amino acid i in our predicted rules.

in the rules. This is due to the fact that amino acids with low values of solvent accessibility are also present in the core of the protein and thus are often in contact. We can appreciate from figure 6.7, that a high number of rules (77%) present secondary structure values of type E (β-sheets). This is explained by the separation in the sequence constraint, which was set to 7. In fact, in this way, intra turn and intra α-helix contacts are avoided. However, this fact does not affect the long-range β-sheet contacts and, therefore, they predominate in our set of rules.

We have also analyzed the relation between the properties of amino acids i and j that are predicted to be in contact. In figures 6.8 and 6.9 we show the hydrophobicity and polarity regions, respectively, for amino acids i and j covered by our predicted rules. The representation of the regions is based on overlapping translucent rectangles whose area covers the range of hydrophobicities or polarities of amino acids i and j that are included in the rules.

From figure 6.8, we can notice that the obtained rules predict contacts between amino acids whose hydrophobicity is high, especially when both amino acids are hydrophobic (values close to 1.0). As we can observe in figure 6.9, non-polar amino acids (values close to -1.0) are more likely to be in contact, according to our rules. These results are consistent with those obtained in figures 6.4 and 6.5. In figure 6.10 we show the relative frequency of charge values for amino acids i and j in the rules. We found that amino acids with charge 0 are often in contact (in 79.3% of cases) according to the rules.

Figure 6.11 shows an example of the two best resulting rules, generated in the experiment 1, according to their $F-measure$. The coverage of $Rule1$ is 0.002, while the accuracy rate is 0.4. On the other hand, coverage value of $Rule2$ is 0.015 and its accuracy rate is 0.65. If we inspect the first rule, we can infer that the hydrophobicity value for the amino acid i lies between 0.52 and 0.92, the polarity value between -1.0 and -0.93, neutral charge 0,

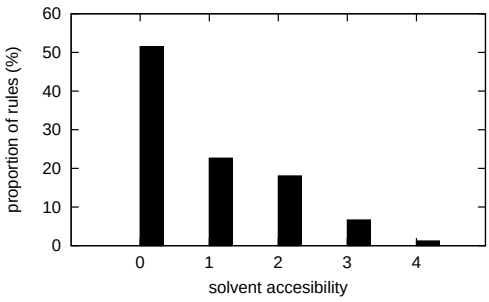


Figure 6.6: Relative frequency of solvent accessibility values for amino acid i in our predicted rules.

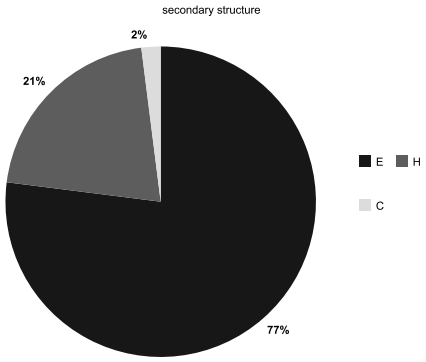


Figure 6.7: Relative frequency of secondary structures for amino acid i in our predicted rules. We consider H for α-helix, E for β-sheets and C for coil.

solvent accessibility 0 and secondary structure 2 (β-sheet). Therefore, the amino acid i could be L (Lysine) or F (Phenylalanine), which fulfills all these features according to the cited scales. As it can be noticed the produced rules are easily interpretable by experts in the field.

6.3 Summary and conclusions

In this chapter we present the experiments and results obtained by our multiobjective evolutionary algorithm, called MECoMaP on four different datasets (DS1, DS2, DS3 and DS4) described in [Fariselli *et al.*, 2001, Cheng and Baldi, 2007, Zhang *et al.*, 2005, Jones *et al.*, 2012] respectively. Our algorithm is tested in the same experimental conditions described in the literature for each dataset, obtaining encouraging results. Statistical tests show that the differences between the methods are statistically significant. We have also analyzed the set of rules obtained on DS1 by MECoMaP. The objective of this analysis is to extract some conclusions about the

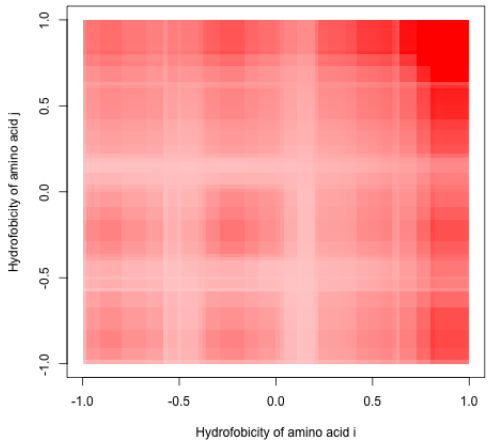


Figure 6.8: Hydrophobicity regions for amino acids i and j covered by our predicted rules.

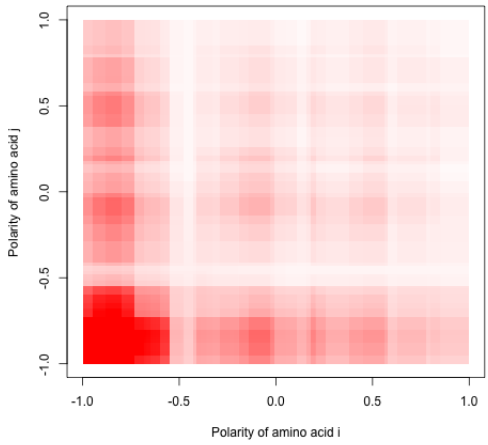


Figure 6.9: Polarity regions for amino acids i and j covered by our predicted rules.

folding protein inferred from the rules. These conclusions are related to the physico-chemical properties of amino acids and the predicted structural features used by the two approaches. For instance, these studies reflect the most commonly used values of the different properties for amino acids in contact, as well as the most frequent attributes in the rules. This information could be used in order to perform an attribute selection, reducing in this

Amino acid j		Amino acid i		
		-1	0	+1
	+1	0.8	6.8	0.3
	0	2.5	79.3	5.4
	-1	0.3	3.5	0.8

Figure 6.10: Relative frequency (%) of charge values for amino acids i and j in our predicted rules.

$$\begin{gathered}
\textit{Rule } 1: \textit{ if } H_i \in [0.52, 0.92] \textit{ and } P_i \in [-1.00, -0.93] \textit{ and} \\
C_i = 0 \textit{ and } SS_i = 2 \textit{ and } SA_i = 0 \textit{ and} \\
H_j \in [0.32, 0.82] \textit{ and } P_{j+1} \in [-0.41, -0.01] \textit{ and} \\
C_{j+1} = 0 \textit{ and } SS_{j+1} = 2 \textit{ and } SA_{j+2} = 1 \textit{ then contact}
\end{gathered}$$

$$\begin{gathered}
\textit{Rule } 2: \textit{ if } H_{i-1} \in [0.20, 0.82] \textit{ and } P_i \in [-0.41, -0.01] \textit{ and} \\
C_i = 0 \textit{ and } SS_{i+1} = 2 \textit{ and } SA_{i+2} = 1 \textit{ and} \\
H_j \in [0.45, 0.62] \textit{ and } P_{j+1} \in [-0.73, -0.01] \textit{ and} \\
\textit{and } SS_j = 2 \textit{ and } SS_{j+1} = 2 \textit{ then contact}
\end{gathered}$$

$$\begin{gathered}
\ldots \\
\textit{else no contact};
\end{gathered}$$

Figure 6.11: An example of the best two rules obtained by MECoMaP on DS1.

way the computational costs of the algorithms and possibly improving their performance. An advantage of these rules, is that can easily be interpreted and analyzed by experts in the field in order to obtain more insight on the protein folding process. Furthermore, we have also analyzed a set of predictive contact map rules generated by the Infobiotics predictor [Bacardit *et al.*, 2012], a collaborative research project together with the University of Nottingham. This study is shown in Appendix A.

Chapter 7

Secondary structure prediction experiments

In this chapter we present the results obtained by our secondary structure predictor. Moreover, we also describe a statistical analysis of amino acid pairs propensities in cap positions of SS motifs.

7.1 Preliminary statistical analysis

The analysis proposed in this section is aimed at studying the different propensities of each pair of amino acids in the studied positions, i.e, N-cap, N_1, C-cap and C_1 (start and end of either a helix or a sheet).

Knowing the propensities of each pair of amino acids at these positions would allow us to extract useful information about the properties of those amino acid that are located in the beginnings or ends of the different secondary structure motifs.

Our data set includes 163461 α-helix and 216390 β-strand sequences extracted from a dataset which will be described in section 8.2, using the DSSP program [Kabsch and Sander, 1983]. To the best of our knowledge, no other approaches have used such a high number of secondary structure states sequences for a similar study. We have computed the global propensities for each pair of amino acids in N-cap, N_1 positions and C_1, C-cap positions. In this analysis, we have used equation 7.1, which calculates the relative frequency of the XY pair at positions $i, i+1$ (N-cap,N_1 or C_1, C-cap) in helices or strands and the relative frequency of that pair in the total protein set.

$$P^g_{X_iY_{i+1}} = \frac{n^{helix}_{X_iY_{i+1}}}{\sum_{AB} n^{helix}_{A_iB_{i+1}}} / \frac{n^{total}_{XY}}{\sum_{AB} n^{total}_{AB}} \tag{7.1}$$

In the equation, $P^g_{X_iY_{i+1}}$ represents the global propensity for a XY amino acid pair, X_iY_{i+1} represents a pair of amino acids in a helix or sheet capping

position, and A_iB_{i+1} stands for a pair of amino acids in any consecutive position in a protein sequence.

We represent the propensities with a matrix, where each propensity is represented by a different color. A cell with blue color represents a high likelihood for this pair. On the other hand, a red cell represents a low propensity.

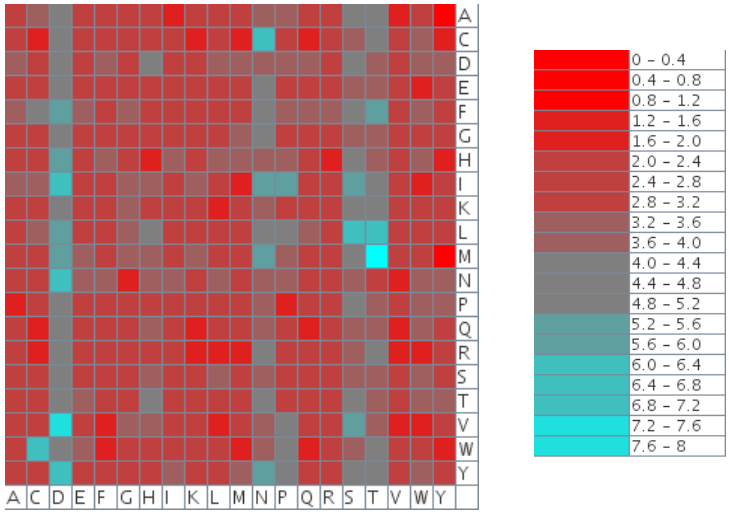


Figure 7.1: Propensity matrix for N-cap,N_1 helix positions.

Figure 7.1 shows a chart for N-cap, N_1 positions of the helices. The y axis represents the N-cap position and x axis represents the N_1 position. The most likely pairs of amino acids to be in N_1 position are D, N, S and T (Aspartic acid, Asparagine, Serine and Threonine, respectively), while practically any amino acid could be the N-cap position. All these amino acids have a polar side chain. The most frequent pair in a α-helix start is MT (Methionine and Threonine).

Figure 7.2 shows a chart for C_1, C-cap positions of the helices. The y axis, represents the C_1 position while the x axis represents the C-cap position. In this case amino acid P (Proline) has the lowest propensity in both C_1 and C-cap positions and the G amino acid (Glycine) at C_1 position. The most frequent pair in a α-helix end is QH (Glutamine and Histidine).

Figure 7.3 shows a matrix for Ncap, N_1 positions for β-strands. The y axis represents the N-cap position while N_1 position is represented on the x axis. From the matrix, we can conclude that the most likely pairs to be in N_1 positions are G, P, N and D (Glycine, Proline, Asparagine and Aspartic acid respectively), with a high number of amino acid in a N-cap position. All these amino acids have a common characteristic: they have a small residue. The most frequent pair in a β-strand start is PV (Proline

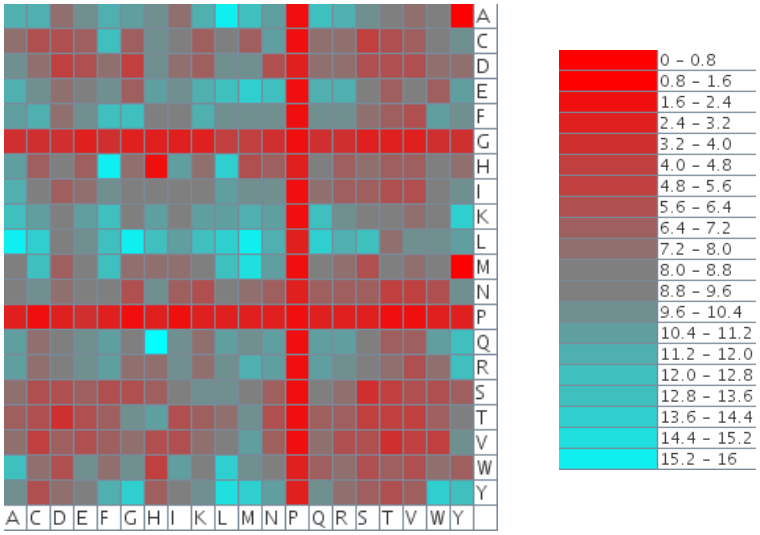


Figure 7.2: Propensity matrix for C_1, C-cap helix positions.

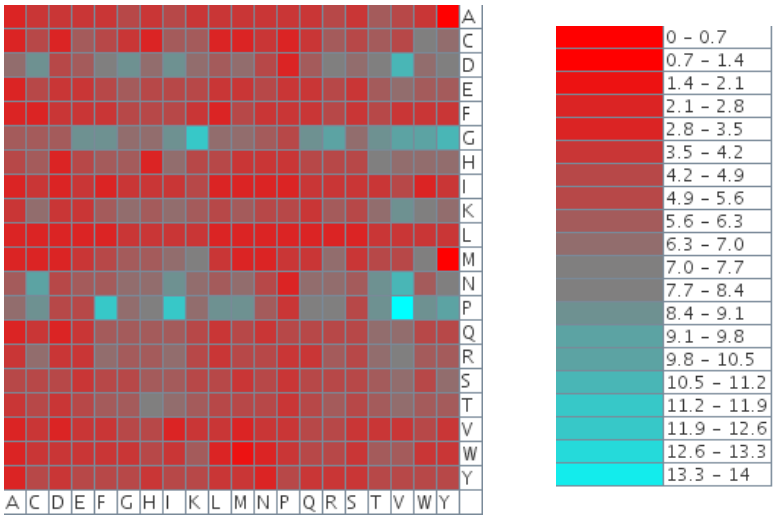


Figure 7.3: Propensity matrix for N-cap, N_1 strand positions.

and Valine).

Figure 7.4 shows the results for C_1, C-cap positions of a β-strand. The y axis, represents the C_1 position and x axis represents the C-cap position. In this case the amino acids N, D and P (Proline) have the lowest propensity to be in position C_1. The most likely pairs to be in C_1 are I, V and Y (Isoleucine, Valine and Tyrosine respectively), with practically any amino acid in a C-cap position. They are all hydrophobic residues. The most

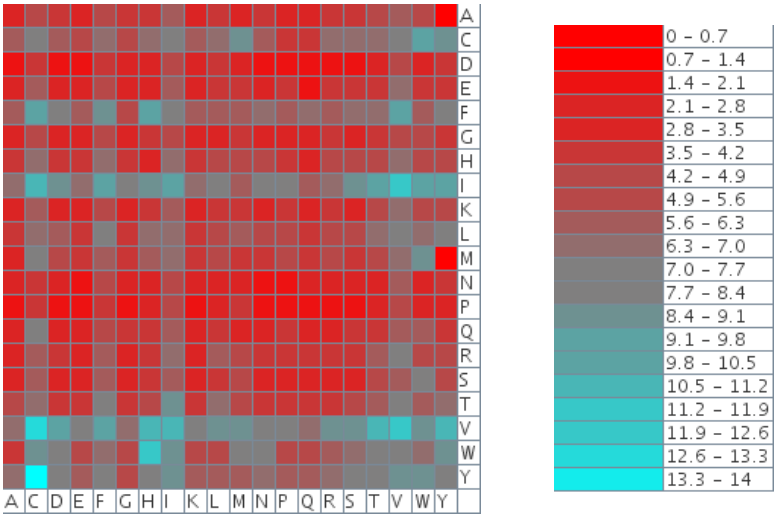


Figure 7.4: Propensity matrix for C_1-Ccap strand positions.

frequent pair in a β-strand end is YC (Tyrosine and Cysteine).

7.2 Experiments and discussion

In this section, we present the experimentation performed in order to assess the validity of the SS predictor. Protein secondary structure is obtained from amino acid sequences so it is necessary to obtain a set of known protein sequences. We extract this data from PDB.

7.2.1 Data sets

A set of $12,830$ non-homologous and non-redundant proteins with a homology lower than 30% were obtained from PDB, using the PDB Advanced Search. We have only selected the structures which contain protein chains and not DNA or RNA chains. The complete list of the $12,830$ PDB protein identifiers can be downloaded in [3]. The DSSP program [Kabsch and Sander, 1983] was used in order to extract the secondary structure relative to α-helix and β-sheet states of each protein based on the atomic coordinates in the PDB file. Once we have located the motifs in the protein sequence, we extract the amino acids from N-cap to C-cap positions of the helix or sheet (figure 5.12), which are the amino acids who are in relevant positions in a α-helix or β-sheet. We have randomly selected a subset of 5000 α-helix and 5000 β-strand sequences without replacement from the initial set, with a minimum size of four residues, and length less than 150 residues. Coils and no-motifs protein sequences are included as

negative examples. A 10-fold cross-validation has been applied. The data set is divided into 10 subsets, and the holdout method is repeated 10 times. Each time, one of the 10 subsets is used as the test set and the other 9 subsets are put together to form a training set. Then the average result across all 10 trials is computed. A model is obtained for each fold. This model consists of a set of rules that identify beginnings and ends of a α-helix or of a β-strand.

For each fold, we compute the following three measures:

- Recall represents the percentage of correctly identified positive cases. In our case, recall indicates what percentage of beginnings or ends of motifs have been correctly identified.

- Precision is a measure to evaluate the false positive rate. Precision reflects the number of real predicted examples.

- Specificity, or true negative rate, measures the percentage of correctly identified negative cases. In this case, specificity reflects what percentage of cases which are not beginnings or ends of motifs, have been correctly identified.

7.2.2 Alpha experimentation

Table 7.1 and table 7.2 show the obtained results for the helix capping prediction algorithms (starts and ends of helix) for a different number of executions, ranging from 10 to $1,000$. The first column specifies the number of execution of the algorithm, the second column gives the average recall obtained. The third and fourth columns provide the average specificity and precision, respectively. For each measure, standard deviation, shown between brackets, is also provided. It can be noticed that for the α-helix capping prediction, the algorithm obtained extremely high specificity, with an average of 0.99. The average recall is about 0.64, in C-cap and about 0.62 in N-cap prediction. The overall precision obtained shows a low rate of error in the prediction with an average of about 0.7. All the measures vary weakly depending on the number of iterations. We can also notice that producing a model with more rules (the more executions the more rules will be part of the model produced) does help in increasing the recall and the precision.

Other approaches were developed to predict starts of helix. The start position are correctly predicted for approximately 30% of all predicted helices in [Wilson *et al.*, 2002]. The number of correctly predicted α-helix start positions was improved from 30% to 38% in [Wilson *et al.*, 2004]. These results are widely outperformed by our approach, as our algorithm predicts about 60% of the start positions correctly. We have not found references for the C-cap helix prediction in literature.

Table 7.1: Average results for the prediction of the beginning of α-helices obtained for different number of iterations. Standard deviation is reported between brackets.

It.	$Recall_{\mu\pm\sigma}$	$Spec._{\mu\pm\sigma}$	$Prec._{\mu\pm\sigma}$
10	$0.604_{\pm 0.103}$	$0.993_{\pm 0.001}$	$0.693_{\pm 0.024}$
20	$0.635_{\pm 0.096}$	$0.991_{\pm 0.002}$	$0.687_{\pm 0.023}$
30	$0.638_{\pm 0.066}$	$0.993_{\pm 0.000}$	$0.692_{\pm 0.012}$
40	$0.623_{\pm 0.055}$	$0.995_{\pm 0.000}$	$0.732_{\pm 0.010}$
100	$0.657_{\pm 0.043}$	$0.991_{\pm 0.003}$	$0.738_{\pm 0.008}$
500	$0.687_{\pm 0.032}$	$0.995_{\pm 0.002}$	$0.756_{\pm 0.009}$
1000	$0.690_{\pm 0.027}$	$0.996_{\pm 0.002}$	$0.768_{\pm 0.009}$

Table 7.2: Average results for the prediction of the end of α-helices obtained for different number of iterations. Standard deviation is reported between brackets.

It.	$Recall_{\mu\pm\sigma}$	$Spec._{\mu\pm\sigma}$	$Prec._{\mu\pm\sigma}$
10	$0.633_{\pm 0.160}$	$0.993_{\pm 0.002}$	$0.665_{\pm 0.022}$
20	$0.643_{\pm 0.196}$	$0.993_{\pm 0.003}$	$0.694_{\pm 0.049}$
30	$0.656_{\pm 0.066}$	$0.992_{\pm 0.002}$	$0.668_{\pm 0.031}$
40	$0.634_{\pm 0.101}$	$0.992_{\pm 0.001}$	$0.640_{\pm 0.013}$
100	$0.657_{\pm 0.080}$	$0.993_{\pm 0.020}$	$0.666_{\pm 0.008}$
500	$0.667_{\pm 0.140}$	$0.994_{\pm 0.003}$	$0.680_{\pm 0.019}$
1000	$0.685_{\pm 0.159}$	$0.995_{\pm 0.001}$	$0.685_{\pm 0.021}$

Table 7.3: Average results for the prediction of the beginning of β-strands obtained for different number of iterations. Standard deviation is reported between brackets.

It.	$Recall_{\mu\pm\sigma}$	$Spec._{\mu\pm\sigma}$	$Prec._{\mu\pm\sigma}$
10	$0.151_{\pm 0.083}$	$0.995_{\pm 0.001}$	$0.671_{\pm 0.039}$
20	$0.163_{\pm 0.040}$	$0.988_{\pm 0.001}$	$0.596_{\pm 0.018}$
30	$0.198_{\pm 0.015}$	$0.994_{\pm 0.000}$	$0.551_{\pm 0.019}$
40	$0.187_{\pm 0.055}$	$0.995_{\pm 0.001}$	$0.565_{\pm 0.044}$
100	$0.223_{\pm 0.060}$	$0.995_{\pm 0.001}$	$0.586_{\pm 0.007}$
500	$0.256_{\pm 0.044}$	$0.996_{\pm 0.002}$	$0.603_{\pm 0.023}$
1000	$0.278_{\pm 0.069}$	$0.994_{\pm 0.001}$	$0.618_{\pm 0.018}$

Our algorithm is capable of producing satisfactory results using an elevated number of sequences (5,000 helix sequences). To the best of our knowledge, no other approaches have used such a high number of sequences in α-helix capping regions prediction.

7.2.3 Beta experimentation

Results relative to the prediction of β-strands capping are given in Table 7.3 and Table 7.4.

The algorithm obtained a recall of 0.28 in N-cap, and about 0.65 in C-cap prediction for 1,000 iterations in both cases. These results are slightly less accurate than those relative to the helix prediction for the N-cap prediction. However, results for C-cap prediction represents a good result as well, and it means that on average, 65% of the ends of β-strands are recognized as such. The precision of the model is also satisfactory. It is about 0.60, in N-cap and about 0.70 in C-cap prediction. This means that model obtained commits few classification errors. High levels of specificity are shown in both cases.

Unlike α-helices [Wilson *et al.*, 2004], to the best of our knowledge, there are not previous results reported in the literature for the β-sheet capping prediction.

It is also interesting to inspect the behavior of our EA. Figure 7.5 shows a graphical representation of the maximum and average fitness values at different generations relative to a typical run for the beginning of β-strands. We can notice that the maximum fitness is achieved very early, at about generation seven, and then it is stable. This may suggest that we should try to increase the mutation probability, or apply a mutation operator that introduces more changes in an individual, in order to increase diversity in the population. Another strategy, could apply some local search method with a given probability. Such local search would help in improving the

Table 7.4: Average results for the prediction of the end of β-strands obtained for different number of iterations. Standard deviation is reported between brackets.

It.	$Recall_{\mu\pm\sigma}$	$Spec._{\mu\pm\sigma}$	$Prec._{\mu\pm\sigma}$
10	$0.489_{\pm 0.101}$	$0.990_{\pm 0.001}$	$0.615_{\pm 0.014}$
20	$0.524_{\pm 0.040}$	$0.991_{\pm 0.001}$	$0.663_{\pm 0.008}$
30	$0.516_{\pm 0.014}$	$0.994_{\pm 0.000}$	$0.760_{\pm 0.019}$
40	$0.586_{\pm 0.059}$	$0.993_{\pm 0.001}$	$0.728_{\pm 0.019}$
100	$0.605_{\pm 0.025}$	$0.992_{\pm 0.002}$	$0.754_{\pm 0.007}$
500	$0.626_{\pm 0.036}$	$0.993_{\pm 0.003}$	$0.765_{\pm 0.025}$
1000	$0.648_{\pm 0.074}$	$0.995_{\pm 0.015}$	$0.780_{\pm 0.017}$

fitness of the individuals.

On the other hand, the average fitness increases constantly, and tends to converge to the maximum fitness toward the end of the run.

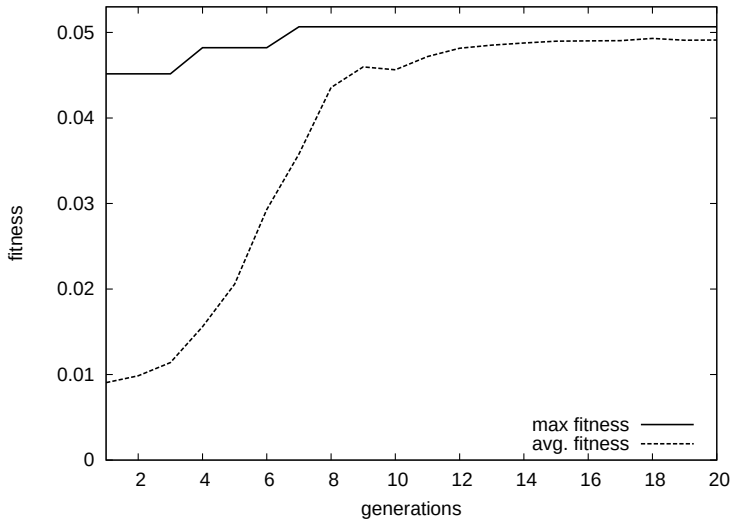


Figure 7.5: Maximum Fitness vs. Average Fitness.

7.3 Summary and conclusions

In this chapter, we have described the experiment settings and results obtained by our secondary structure predictor. We have also included a study of amino acid pairs propensities in cap positions of α-helices and β-strands. This algorithm predicts the beginnings and ends of these SS motifs obtaining very good results. Moreover, the algorithm has been tested using

a high number of sequences. We believe that this represent an important factor. In fact, the number of protein sequences available increase by the day, and thus, having a method that is scalable would be very important.

Part V

Conclusions

Chapter 8

Conclusions and Future works

The main goal of this thesis was the introduction of a novel evolutionary methodology to predict protein structure from the primary sequence of amino acids.

- On the one hand, we proposed a multi-objective evolutionary algorithm approach for protein contact map prediction. Our algorithm generates a set of rules for residue-residue contact prediction using a representation based on amino acid properties. The rules forming the final solution express a set of conditions on specific physico-chemical properties and structure information of amino acids. Such rules can easily be interpreted and analyzed by experts in the field in order to obtain more insight on the protein folding process.

  Our approach have been tested on four different protein data bases, which appear in the literature, obtaining good results. A statistical study of our set of rules have been performed. Some conclusions about the folding protein can be inferred from the rules. These conclusions are related to the physico-chemical properties of amino acids (hydrophobicity and polarity) and two predicted structural features (SA and SS) used by our approach.

  The main advantage of our representation is the easily interpretation of the generated decision rules by experts in the fields. The information extracted from these rules could provide useful insights into protein structure prediction problem. The use a set of amino acid properties, which include very important information in the folding process [Gu and Bourne, 2003], represents another advantage of our algorithm. To the best of our knowledge, similar representations have not been considered in the literature.

  As for future work, we intend to expand this study to other significant

amino acid properties, e.g., isoelectric point and steric parameter. Furthermore, we are planning to include evolutionary information like Position-Specific Score Matrix (PSSM). This information must be encoded in the representation of the algorithm. We also intend to study the possibility of using self-adaptable parameters for controlling the genetic operators used in the algorithm. Another future development is the application of the algorithm to larger proteins data set, in order to test the validity of our proposal in these cases, where the resulting rules set could cover more of the search space.

As in [Li *et al.*, 2011] and [Bacardit *et al.*, 2012], we intend to incorporate a balancing of the data. The ratio of positive cases (contacts) and negative cases (non contacts) are 1:1 and 1:2 in these works. The inclusion of domain protein information could also improve the results in the prediction as in [Calvo *et al.*, 2011]. PSICOV [Jones *et al.*, 2012] incorporates an improvement on multiple sequence alignment that we could adapt to our algorithm using sparse inverse covariance estimations.

- On the other hand, we have proposed an evolutionary algorithm for α-helix and β-strand capping prediction from sequences of amino acid. Three amino acids properties (hydrophobicity, polarity and net charge) have been incorporated in the fitness function. These properties help in improving the search process performed by the algorithm.

  We have also performed a statistical analysis aimed at discovering the amino acid propensities in capping positions in $163,461$ α-helices and $216,390$ β-strands extracted from PDB using the DSSP program. For each pair of possible amino acid, N-cap and N_1 positions and C_1 and C-cap positions have been taken into account. This study provided us with useful information for the prediction of secondary structure. In fact, this information could be used for modifying the fitness function, improving in this way the evolutionary search. A study of each single amino acid has been also developed in each position. From this study, we could individuate which amino acid is more probable to appear in one of the positions taken into consideration.

  In order to test the validity of the proposed algorithm, we performed a set of experiments using $5,000$ α-helix and $5,000$ β-strand sequences extracted from a protein data set obtained from Protein Data Bank and composed by $12,830$ non-redundant and non-homologous protein with a homology rate lower than 30%. To the best of our knowledge, no other approaches have used such a high number of sequences in α-helix capping regions prediction. Results obtained on the prediction of α-helices are very encouraging and in particular, the accuracy characterizing the prediction models obtained is very high

independently from the number of generated rules. As far as the experiments on the prediction of β-sheets, we have not found other results in the literature to contrast our results. However, also in this case, the accuracy obtained is satisfactory, even if the results are slightly worse than those obtained for the α-helices. Protein secondary structure prediction is essential for the three-dimensional structure modeling, accurate sequence alignment and function prediction.

Future works will be focused on the analysis of different properties to be included in the fitness function in order to increase the quality of the prediction model, such as, e.g., residue size, which has a significant relevance according to our statistical study. We will also expand the number of residues in the window of amino acids. Furthermore, we are studying the possibility of incorporating a local search phase in the algorithm that will help to improve individuals. We also intend to extend our experimentation to other dataset of protein sequences. Moreover, we could incorporate the use of propensities for the SS formation, as in [Chen *et al.*, 2009]. In this aspect, we have developed a study (see Appendix B) of amino acid propensities in the formation of SS motifs (α-helix, β-sheets and coils). The incorporation of SA in our encoding, as in [Faraggi *et al.*, 2012], can achieve an improvement of the accuracy in SS prediction. The use of specific domains and protein families can also increase the accuracy of the prediction. Several methods are applied only to a determined family or protein domain [Olson *et al.*, 2012].

Part VI

Appendix

Appendix A

Analysis of Infobiotics predicting rules

In collaboration with the Interdisciplinary Computing and Complex Systems research group ($ICOS_2$) of the School of Computer Science of the University of Nottingham, we have performed an analysis of a set of rules obtained by the Infobiotics contact map predictor presented in CASP9 [Bacardit *et al.*, 2012]. Infobiotics was ranked among the best predictor participants to the competition. The system is based on the prediction of some structural features of protein residues such as SS, SA, Recursive Convex Hull (RCH) and coordination number (CN), an ensemble strategy used to facilitate the training process and a genetic algorithms- based rule learning system, called BioHEL, which generates the rule sets from which we can extract human-readable explanations and useful information about the contact map predictions[1]. Following the notation used in [Bacardit *et al.*, 2012], $r1$ and $r2$ denote the target amino acids. The representation used by the predictor contains information regarding two windows of ± 4 residues around $r1$ and $r2$. In addition to this, the system also uses a third window of ± 2 residues centered around the middle point in the chain between $r1$ and $r2$. The prediction is based on the PSSM profile and the predictions of SS, SA, CN and RCH contained in the three windows. Rules generated by Infobiotics are similar to those obtained by MECoMaP. An example of rule is shown in figure A.1.

In this rule, words in capital letter represent attributes, *e.g.* PredSA represents the prediction of SA, and characters preceded by the symbol _ , *e.g.* $\_r1$, indicates the particular position of the represented amino acid where, for instance, $r2$-1 is the amino acid which immediately precede $r2$ in the sequence. A list with all the possible attributes is shown in Table A.5.

We have analyzed a total of $1,250$ rule sets with an average of $152.5(\pm 7.1)$ rules per set. Each rule contains an average of $8.4(\pm 2.9)$ attributes out of a

[1] http://icos.cs.nott.ac.uk/software/biohel.html

$$if\ Propensity \in [0.53, 1.51], PredSA\_r2 \leq 0.00,$$
$$PSSM\_r1E \in [-10.06, -4.78], PSSM\_r2 - 1\_Q \leq -3.57,$$
$$PSSM\_central\_Q \in [-12.98, 6.42],$$
$$PSSM\_central\_R \in [-2.96, 7.34]$$
$$then\ contact$$

Figure A.1: An example of rule obtained by Infobiotics.

total of 631 possible attributes.

A first study was aimed at discovering the most frequently used attributes in the rules obtained by BioHEL. Table A.1 presents the first twenty attributes according to their presence in the rules. We can see that, attributes PredSA_r1, propensity, PredSA_r2 and PredSS_r1 appear in about 20% of the whole total of rules. This result highlights the key role played by these attributes in the prediction. Attributes PredRCH_r1 and PredRCH_r2 appear in about 15% of the rules. The prediction of the coordination number (PredCN_r1 and PredCN_r2) appears in about 10% of the rules. PSSM attributes referred to amino acid E, D, N and K, appear in about 10% of the rules. Three of these are charged amino acids. Therefore, the net charge of the residue seems to be an influence factor in the formation of residue-residue contacts. To confirm this idea, we have computed the average rank of the PSSM elements corresponding to each amino acid type. Table A.2 contains the results of this analysis, reporting the average rank for the positions of the windows associated to the target residues. As we can observe in the rank, polar and charged amino acids occupy the first positions of the rank (D,E,N,K), giving us the idea of the importance of the polarity and charge properties in the formation of contacts. Next we find two aliphatic amino acids (I and V). Aromatic and tiny amino acids are in general low in the ranking.

We performed a second analysis in order to calculate the frequency of appearance of each pair of attributes in the rules. Table A.3 shows the top 20 most frequent pairs of attributes used in BioHEL's rules. We can observe a very clear trend in the most frequent pairs: a pair includes one attribute associated to each of the target residues ($r1$ and $r2$). This is an expected result since we are predicting the contact between these two residues. The most frequent attributes are those referred to SS and SA. In general the ranking of pairs follows the trends already identified in the ranking of attributes.

Table A.4 shows a comparison between the average and best rank of all the attributes aggregated by the type of attribute and the window positions

Table A.1: Top 20 most frequent attributes used in BioHEL's rules, where Ratio is the percentage of rules where the attribute appears.

Rank	Attribute	Ratio
1	PredSA_r1	22.3
2	propensity	20.1
3	PredSA_r2	18.4
4	PredSS_r1	17.4
5	PredSS_r2	15.7
6	PredRCH_r1	15.6
7	PredRCH_r2	13.9
8	PredSS_r1 1	13.7
9	PredSS_r2 -1	13.2
10	PredCN_r1	12.3
11	PSSM_r2 0 E	11.6
12	PSSM_r2 0 D	10.9
13	PredCN_r2	10.1
14	PredSS_r1 -1	10.0
15	PSSM_r1_0 D	10.0
17	PSSM_r1_0 E	9.9
18	PSSM_r2_0 N	9.4
19	PredRCH_freq_connecting_0	9.4
20	PSSM_r2_0_K	9.0

(*e.g.* $r1 \pm 4$ window, $r2 \pm 4$ window and amino acids of central window). Some conclusions can be extracted from this table. Attributes relative to the central positions are in the lowest positions of the rank. Therefore, we assume a low relevance of the central positions in the prediction of contacts. Although PredSA_r1 is the most frequent attribute, it occupies only the 10th position in the average rank. Thus, SA prediction loses importance for the amino acids which precede and follows the target residues. The first positions of the ranking are occupied by propensity, separation, length and prediction of SS, CN and RCH. In these cases, the difference between the average and best rank are barely noticeable.

We have also analyzed the relative frequencies of the values of the different attributes related to the residue $r1$ in the rules. Figure A.2 shows the frequencies of the different values of SA. In the studied rules, the most frequent value corresponds to the buried state (value=0) reaching a 48.29% probability. The sum of the three first states (0,1 and 2) reaches a total of 88.64% probability. These results indicate that, similar to the case of MECoMaP rules, amino acids which are not in an exposed surface, are

Table A.2: Rank of the evolutionary information attributes aggregated by their AA type.

AA Type	Target residue rank
D	13.5±1.5
E	13.5±2.5
N	19.5±2.5
K	21.5±1.5
Q	27.0±2.0
R	34.0±1.0
I	57.0±11.0
V	66.0±16.0
S	73.0±2.0
G	85.0±15.0
H	92.5±2.5
L	111.0±23.0
P	136.0±12.0
M	209.0±7.0
F	227.5±13.5
C	252.5±85.5
T	260.5±24.5
Y	270.5±7.5
W	292.0±19.0
A	490.5±22.5

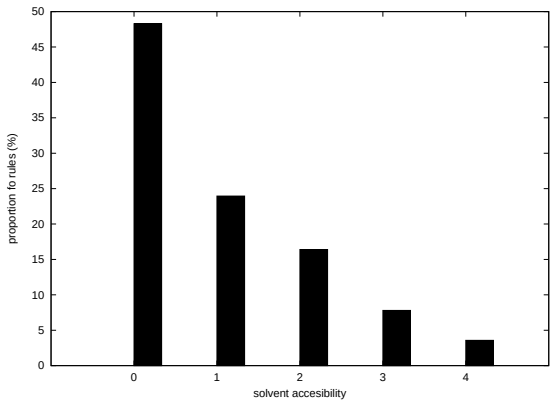


Figure A.2: Relative frequency of SA values for amino acid $r1$ in BioHEL rules.

Table A.3: Top 20 most frequent pairs of attributes used in BioHEL's rules, where Ratio is the percentage of rules containing the attributes.

Rank	Attribute 1	Attribute 2	Ratio
1	PredSS_r1	PredSS_r2	5.4037
2	PredSA_r1	PredSA_r2	4.7722
3	PredSA_r1	propensity	4.7537
4	PredSS_r1_1	PredSS_r2	4.1945
5	PredSS_r1	PredSS_r2_-1	3.9722
6	PredSS_r1_1	PredSS_r2_-1	3.7199
7	PredSA_r2	propensity	3.6903
8	PSSM_r2_0_E	PredSA_r1	3.4991
9	PSSM_r2_0_E	propensity	3.3260
10	PredSA_r1	PredSS_r2	3.2753
11	PredSS_r1	PredSS_r2_1	3.1855
12	PredRCH_r1	PredRCH_r2	3.1802
13	PSSM_r2_0_D	PredSA_r1	2.9923
14	PSSM_r1_0_E	propensity	2.9389
15	PredRCH_r2	PredSA_r1	2.9363
16	PredRCH_r1	PredSA_r2	2.8951
17	PredSA_r2	PredSS_r1	2.8735
18	PredSS_r1_-1	PredSS_r2	2.8676
19	PredSA_r1	PredSS_r2_-1	2.7636
20	PSSM_r2_0_K	PredSA_r1	2.7557

more probably to be in contact.

Figure A.3 shows the frequency of SS values in BioHEL rules. This attribute in the rules indicates that a residue can belong to one or several secondary structures, where C=coil, H=helix and E=sheet. A huge percentage of rules, 71.60%, determines the formation of β-sheets, just like in the case of MECoMaP. This is due to the fact that short range interactions between residues (located in α-helices) are avoided in the prediction, as BioHEL only use training examples with a minimum separation of 24 residues. β-sheets are generally formed by long range interactions [Gu and Bourne, 2003].

CN is the number of spatial neighbors of the residue within a specified distance threshold [Bacardit *et al.*, 2009]. The boundary of a sphere around a residue to determine the neighborhood is defined by a distance cutoff. This method appeared in [Kinjo *et al.*, 2005]. Figure A.4 shows the frequency of CN values in BioHEL rules. We conclude that the majority of the rules (42.78%) belongs to the last group (value 4). This value implies a higher

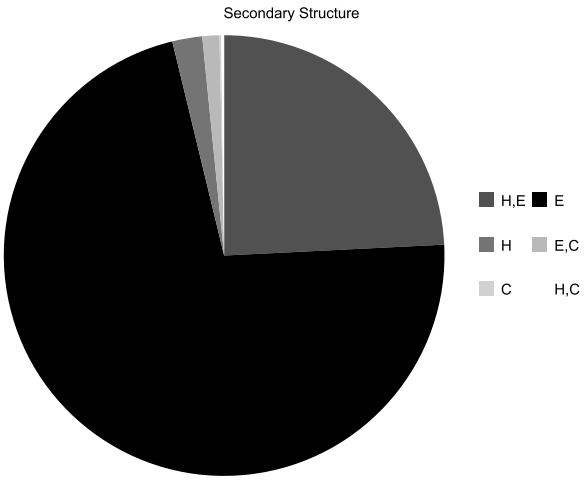


Figure A.3: Relative frequency of SS values for amino acid $r1$ in BioHEL rules.

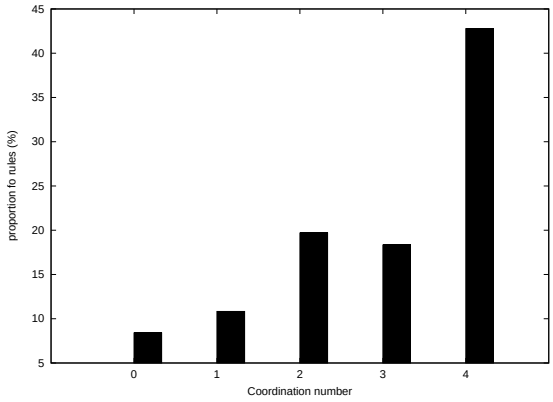


Figure A.4: Relative frequency of CN values for amino acid $r1$ in BioHEL rules.

concentration of residues close to the protein contact.

Figure A.5 shows the frequencies of propensity values of contact formation in the rules. This attribute determines the propensity of a pair of amino acids to be in contact. As could be expected, high values of propensity appear more frequently in the BioHEL rules.

Recursive Convex Hull (RCH) [Stout *et al.*, 2008] is a metric that calculates the degree of burial of a residue within the core of a protein. The topology of the protein is modelled using the concept of convex hull. Hulls were iteratively identified, and residues in each hull were assigned a

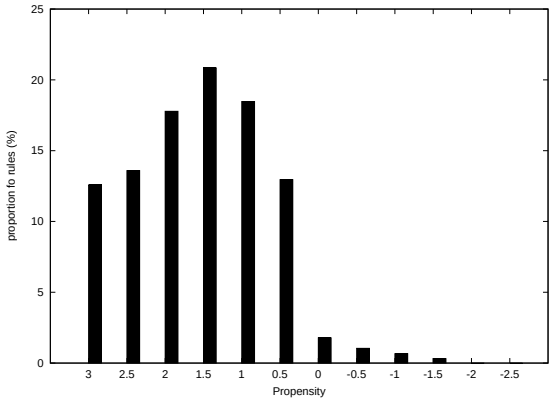


Figure A.5: Relative frequency of propensity values for amino acid $r1$ in BioHEL rules.

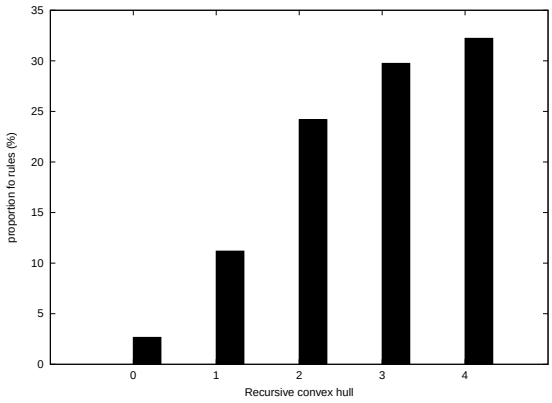


Figure A.6: Relative frequency of RCH values for amino acid $r1$ in BioHEL rules.

number [0..4], where 4 is the most buried state. Figure A.6 shows the relative frequency for the different RCH states. Results indicate that the majority of the rules, which include this attribute are referred to the buried states (value 2 = 24.18%, value 3 = 29.75% and value 4 = 32.22% respectively). This conclusion is in line with the outcomes of the SA study.

Separation between residues of the sequence could be a prediction factor that should be considered. Consequently, this information is included in the predictive rules of BioHEL. Figure A.7 shows the percentages of sequence lengths, ranging from 20 to 200 in intervals of 20. The vast majority of the rules which include this attribute, indicates a sequence separation between 24 and 80 (71.33%). The most frequent interval is the first one ($[20, 40]$), reaching a 30.49% of the rules.

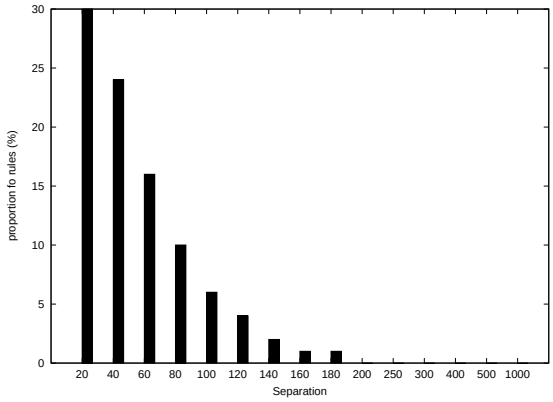


Figure A.7: Relative frequency of separation between residues for amino acid $r1$ in BioHEL rules.

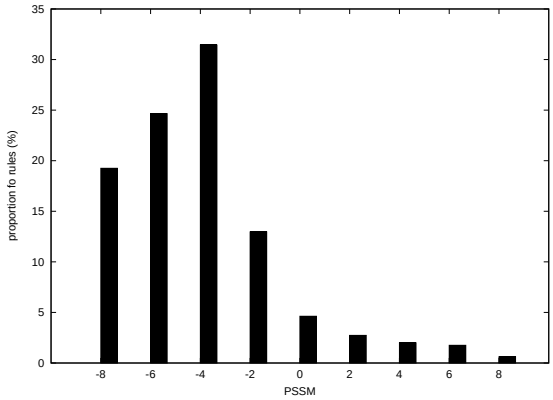


Figure A.8: Relative frequency of PSSM of amino acid D at position $r1$ in BioHEL rules.

Finally, figure A.8 shows the relative frequency of PSSM values. In this case we analyze the rules with the attribute PSSM_r1_0_D which is one of the most common in the rules. This attribute indicates the substitution score at $r1$ position for the amino acid D in the alignment, and can assume values in the interval $[-8, 8]$. In this case, the higher percentages correspond to negative values (88.29% of the respective rules). This indicates that this substitution is not beneficial for the contact formation.

Table A.4: Average and best rank of the information sources in BioHEL's rules grouped by amino acid window and sorted by average rank.

Type	Average rank	Best rank
propensity	2.0±0.0	2
separation	24.0±0.0	24
PredSS_freq_connecting	32.0±9.9	19
length	42.0±0.0	42
PredSS_r1	43.0±31.4	4
PredSS_r2	47.8±38.9	5
PredSS_freq_global	75.3±58.5	30
PredCN_r1	81.6±40.2	10
PredRCH_r1	82.7±40.1	6
PredSA_r1	82.8±42.6	1
PredRCH_r2	85.6±37.8	7
PredCN_r2	89.2±37.7	13
AA_freq_connecting	91.8±38.4	45
PredSA_freq_connecting	92.4±56.2	27
PredSA_r2	93.6±44.3	3
PredRCH_freq_connecting	114.2±48.9	18
PredCN_freq_connecting	118.2±49.2	63
PredCN_freq_global	133.2±66.6	31
PredRCH_freq_global	134.2±37.2	62
PredSA_freq_global	171.8±81.4	65
PredSS_central	282.2±39.5	232
AA_freq_global	290.9±63.7	181
PSSM_r1	322.6±130.8	15
PredCN_central	329.4±18.0	301
PSSM_r2	334.6±144.8	11
PredRCH_central	366.4±33.6	305
PredSA_central	408.8±41.4	331
PSSM_central	568.6±50.4	390

Table A.5: Definition of the attributes in the BioHEL's rules, where $r1$ = first residue in the pair of residues that are tested whether they are in contact or not, $r2$ = second residue in the pair and central = middle point in the chain between the two residues in the pair.

Attribute	Definition
separation	Chain separation of the pair of residues
propensity	Static contact propensity of the two AA types in the pair
length	Length of the protein chain
PredSS_r[1,2]_[-4 .. +4]	Pred. SS of pos. [-4 .. +4] in the r[1,2] window
PredSS_freq_connecting_[H/E/C]	Ratio of state [H/E/C] of pred. SS in the connecting segment
PredCN_r[1,2]_[-4 .. +4]	Pred. CN of pos. [-4 .. +4] in the r[1,2] window
PredCN_freq_connecting_[0 .. 4]	Ratio of state [0 .. 4] of pred. CN in the connecting segment
PredRCH_r[1,2]_[-4 .. +4]	Pred. RCH of pos. [-4 .. +4] in the r[1,2] window
PredRCH_freq_connecting_[0 .. 4]	Ratio of state [0 .. 4] of pred. RCH in the connecting segment
PredSA_r[1,2]_[-4 .. +4]	Pred. SA of pos. [-4 .. +4] in the r[1,2] window
PredSA_freq_connecting_[0 .. 4]	Ratio of state [0 .. 4] of pred. SA in the connecting segment
PredSS_freq_global_[H/E/C]	Ratio of state [H/E/C] of pred. SS in the whole chain
PredCN_freq_global_[0 .. 4]	Ratio of state [0 .. 4] of pred. CN in the whole chain
PredRCH_freq_global_[0 .. 4]	Ratio of state [0 .. 4] of pred. RCH in the whole chain
PredSA_freq_global_[0 .. 4]	Ratio of state [0 .. 4] of pred. SA in the whole chain
AA_freq_connecting_AA	Ratio of amino-acid type AA (one-letter code) in the connecting segment
PredSS_central_[-2 .. +2]	Pred. SS of pos. [-2 .. +2] in the central window
PredCN_central_[-2 .. +2]	Pred. CN of pos. [-2 .. +2] in the central window
PredRCH_central_[-2 .. +2]	Pred. RCH of pos. [-2 .. +2] in the central window
PredSA_central_[-2 .. +2]	Pred. SA of pos. [-2 .. +2] in the central window
AA_freq_global_AA	Ratio of amino-acid type AA (one-letter code) in the whole chain
PSSM_r[1,2]_[-4 .. +4]_AA	Column AA of the PSSM profile of pos. [-4 .. +4] in the r[1,2] window
PSSM_central_[-2 .. +2]_AA	Column AA of the PSSM profile of pos. [-2 .. +2] in the central window

Appendix B

Tables of experiments

AA	Alfa	Beta	Coil
A	9.74	6.46	7.50
R	5.56	4.60	4.99
N	3.71	3.42	4.75
D	5.49	4.50	4.88
C	1.23	1.78	2.34
Q	4.07	3.06	3.82
E	8.63	6.15	7.19
G	5.54	6.53	9.11
H	2.59	3.09	2.92
I	5.78	7.88	4.85
L	10.6	8.91	6.50
K	6.12	5.26	7.05
M	2.40	2.11	2.34
F	4.10	5.22	2.54
P	3.18	3.30	5.67
S	5.78	5.71	9.28
T	4.48	5.63	4.85
W	1.60	1.95	1.20
Y	3.36	4.17	2.48
V	6.05	10.26	5.74

Table B.1: Alpha helix, beta strand and coil amino acid propensities, in percentage terms, from 12,860 non-redundant PDB proteins sharing less than 30% sequence identity, using the DSSP program. Amino acids L, A and E (10.6, 9.74 and 8.63 % respectively) show a high propensity for alpha helices. Amino acids V, L and I (10.26, 8.91 and 7.88 % respectively) indicates high propensities for beta sheets. On the other hand, Amino acids S, G, A and K (9.28, 9.11, 7.50 and 7.05 % respectively) show high propensities for coils.

Appendix C

List of proteins

L<100	1MUP	1BP5_B
1OWW	1QO6	1BTJ_B
1J8K	1E8B	L>400
1Q38	1EBB	1H76
1QGB	1E88	1J7E_B
1FNA	L 200-300	1LOT_A
1FBR	1F5F	1KW2_A
1O9A_A	1NAR	1JNF
1O9A_B	L 300-400	1OD5_A
1TTG	1JQF	1J7E_A
1TTF	1D3K	1J78_A
L 100-200	1BTJ_A	1J78_B
1DV9	1TFD	
1KDK	1A8E	
1KDM	1B3E	
1D2S	1OQG	
1LHN	1N84	
1LHO	1LOT_B	
1LHU	1OQH	
1LHV	1BP5_A	

Table C.1: List of proteins for the third experiment of MECoMaP.

L<100	1msi	2sn3	1eca	5p21	2baa
1a1i_A	1mzm	2sxl	1erv	7rsa	2fha
1a1t_A	1nxb	3gat_A	1exg	L 170–299	L>300
1a68	1ocp	3mef_A	1hfc	1ad2	16pk
1a7i	1opd	4mt2	1ifc	1akz	1a8e
1acp	1pce	5pti	1jvr	1amm	1ads
1ah9	1plc	L 100–169	1kpf	1aol	1fts
1aho	1pou	1a62	1kte	1ap8	1axn
1aie	1ppt	1a6g	1bgf	1bf8	1b0m
1ail	1brf	1acz	1npk	1bjk	1bg2
1dun	1sco	1asx	1pdn_C	1byq_A	1bgp
1ajj	1spy	1aud_A	1pkp	1c3d	1bxo
1aoo	1sro	1ax3	1poa	1cdi	1dlc
1ap0	1tbn	1b10	1put	1cne	1irk
1ark	1tiv	1bc4	1ra9	1cnv	1iso
1awd	1tle	1bd8	1rcf	1csn	1kvu
1awj	1tsg	1bea	1rie	1ezm	1moq
1awo	1ubi	1bfe_A	1skz	3chy	1svb
1bbo	1uxd	1bfg	1tam	1juk	1uro_A
1bc8_C	2acy	2lef_A	1vsd	1kid	1ysc
1c5a	2adx	1bkf	1whi	1mml	2cae
1cfh	2bop_A	1bkr_A	2fsp	1mrj	2dpg
1ctj	2ech	1br0	2gdm	1nls	2pgd
1cyo	2fdn	1bsn	2ilk	1ppn	3grs
1fna	2fn2	1bv1	2lfb	1rgs	1arv
1hev	2fow	1bxa	2pil	1rhs	
1hrz_A	2hfh	1c25	2tgi	1thv	
1kbs	2hoa	1cew_I	2ucz	1vin	
1mbh	2hqi	1cfe	1mak	1xnb	
1mbj	1rof	1cyx	3lzt	1yub	

Table C.2: List of proteins for the first experiment of MECoMaP.

PDB name	Length	Type	PDB name	Length	Type
1IG5A	75	alpha	1XERA	103	a+b
1HXIA	112	alpha	1JSFA	130	a+b
1SKNP	74	alpha	1DZOA	120	a+b
1ELRA	128	alpha	1GRJA	151	a+b
1E29A	135	alpha	1MSCA	129	a+b
1CTJA	89	alpha	1CEWI	108	a+b
1J75A	57	alpha	1VHHA	157	a+b
1ECAA	136	alpha	1BUOA	121	a+b
1FIOA	190	alpha	1G2RA	94	a+b
1C75A	71	alpha	1E9MA	106	a+b
1HCRA	52	alpha	1E87A	117	a+b
1QJPA	137	beta	1H9OA	108	a+b
1D2SA	170	beta	1IDOA	184	a/b
1CQYA	99	beta	1CHDA	198	a/b
1BMGA	98	beta	1FUEA	163	a/b
1MAIA	119	beta	1CXQA	143	a/b
1AMXA	150	beta	1F4PA	147	a/b
1G3PA	192	beta	1ES8A	196	a/b
1RSYA	135	beta	1DMGA	172	a/b
1WHIA	122	beta	1A1HA	85	small
1HE7A	107	beta	9WGAB	171	small
1MWPA	96	a+b	2MADL	124	small
1QGVA	130	a+b	1EJGA	46	small
1DBUA	152	a+b	1AA0A	113	coil-coil

Table C.3: List of proteins for the second experiment of MECoMaP.

1A3A	2ARC	1dsx	1i4j	1nps
1ATL	2HS1	1eaz	1i58	1nrv
1BDO	2VXN	1ej0	1i5g	1ny1
1BEH	3BOR	1ej8	1i71	1p90
1BSG	3DQG	1ek0	1ihz	1pch
1CHD	1a6m	1f6b	1im5	1pko
1CJW	1a70	1fcy	1j3a	1qf9
1CZN	1aap	1fk5	1jbk	1qjp
1D1Q	1aba	1fl0	1jfu	1ql0
1FQT	1ag6	1fna	1jfx	1roa
1GMX	1aoe	1fvg	1jkx	1smx
1GZC	1atz	1fvk	1jl1	1svy
1IIB	1avs	1fx2	1jo8	1t8k
1IWD	1beb	1g2r	1jos	1tif
1JBE	1bkr	1g9o	1jvw	1tqg
1JO0	1brf	1gbs	1jwq	1tqh
1K7C	1c44	1gmi	1jyh	1vfy
1KQ6	1c52	1guu	1k6k	1vhu
1KQR	1c9o	1gz2	1k7j	1vp6
1LM4	1cc8	1h0p	1kid	1w0h
1NB9	1cke	1h2e	1ktg	1whi
1O1Z	1ctf	1h4x	1ku3	1wjx
1R26	1cxy	1h98	1kw4	1xdz
1RW1	1d0q	1hdo	1lo7	1xff
1RW7	1d4o	1hfc	1lpy	1xkr
1RYB	1dbx	1hh8	1m4j	2cua
1TZV	1dix	1htw	1m8a	2mhr
1VJK	1dlw	1hxn	1mk0	2phy
1VMB	1dmg	1i1j	1mug	2tps
1WKC	1dqg	1i1n	1ne2	5ptp

Table C.4: List of proteins for the WEKA experiment.

Appendix D

Glossary

3_{10}-helix: A secondary structure motif that occurs most often as a single turn transition. Similar to an alpha helix, but more tightly coiled. Has 3 residues per helical turn, and 10 atoms in the ring closed by the hydrogen bond. Minimum length of 3 residues.

β-bridge: A single pair β-sheet hydrogen bond formation with only one amino acid on each strand.

π-helix: A secondary structure helix motif with 5 residues per helical turn and a minimum length of 5 residues.

Bend: A secondary structure motif. The only non-hydrogen-bond based assignment.

BLAST: A set of programs used to perform fast sequence alignmnents between two or more gene or protein sequences. The program is used to search a protein or nucleotide sequence database for a match against a query protein or nucleotide sequence. The statistical significance of each match is given in order to indicate the degree of similarity between related or homologous sequences.

DNA: DNA or deoxyribonucleic acid is the molecule that encodes genetic information necessary for all cellular functions. DNA is composed of the sugar deoxyribose, phosphate groups, and the bases adenine (A), thymine (T), guanine (G) and cytosine (C). The DNA molecule is normally a double helix in which the two strands are bound through complementarity of the bases. Cs from one strand hydrogen bind to Gs of the other strand and vice versa. Similarly, As and Ts hydrogen bind to each other.

Electron microscopy: Imaging technique that uses a beam of high energy electrons to produce a magnified image of an object.

mRNA: A RNA molecule that is transcribed from DNA and is translated into the amino acid sequence of a polypeptide

Nuclear magnetic resonance: Property that magnetic active nucleic have in a magnetic field when responding to applied electromagnetic pulses or perturbations. The measurements of such responses provide rich structural, dynamic, and kinetic information about the molecule.

Ramachandran plot: A two-dimensional plot showing the backbone conformational angles psi versus phi in a polypeptide. Various regions of the plot indicate specific secondary conformation. The phi angle is around the C-N bond and the psi angle is around the CA-C bond. Ramachandran plots are used to identify areas in a structure with geometric problems.

Ribosome: Particle composed of ribosomal RNAs and proteins that is the site of protein synthesis.

RNA: RNA is a usually single-stranded nucleic acid similar to DNA but containing the sugar ribose rather than deoxyribose and the base uracil (U) rather than thymine (T). RNA has structural, genetic, and enzymatic roles. There are three major types of RNA: messenger RNA (mRNA), transfer RNA (tRNA), and ribosomal RNA (rRNA).

tRNA: RNA molecule thata carries amino acids to the ribosome during protein synthesis

X-ray crystallography: A method used to determine the detailed three-dimensional structure of molecules present in a crystal in which an X-ray beam is aimed at the crystal and the resulting scattered rays are studied.

Appendix E

Acronyms

AA: Amino acid.

ANN: Artificial neural network.

ARFF: Attribute-Relation File Format.

BLOSUM: Block Substitution Matrix.

CASP: Critical Assessment of Techniques for Protein Structure Prediction.

CATH: Class, Architecture, Topology and Homologous superfamily.

CBR: Case-based reasoning.

CHARM: Chemistry at HARvard Macromolecular Mechanics.

CM: Contact map.

CN: Contact number.

DNA: Deoxyribonucleic acid

DS: Data set.

DSSP: Dictionary of secondary structure of proteins.

EC: Evolutionary computation.

GA: Genetic algorithm.

GDT-TS: Global Distance Test — Total Score.

HSMM: Hidden semi-Markov model.

HSSP: Homology derived Secondary Structure of Proteins.

MCC: Mathew correlation coeficient.

MECoMaP: Multi-objective Evolutionary Contact Map Predictor

MOEA: Multi-objective evolutionary algorithm.

NMR: Nuclear magnetic resonance.

NN: Nearest neighbour.

PCC: Pearson correlation coeficient.

PDB: Protein Data Bank.

PSIBLAST: Position-Specific Iterative Basic Local Alignment Search Tool.

PSP: Protein structure prediction.

PSSM: Position-specific scoring matrices.

RBFNN: Radial basis function neural network.

RCH: Recursive convex hull.

RMSD: Root mean squared deviation.

RNA: Ribonucleic acid

RSA: Relative solvent accessibility.

SA: Solvent accessibility.

SCOP: Structural Classification of Proteins database.

SMO: Sequential minimal optimization algorithm.

SOV: Segment Overlap Measure

SPEA: Strength Pareto Evolutionary Algorithm.

SS: Secondary structure.

SVM: Support vector machines.

TM-score: Template Modeling Score.

Part VII

Bibliography

Bibliography

[Abu-Doleh *et al.*, 2011] Abu-Doleh, A.A., Al-Jarrah, O.M. and Alkhateeb, A. Protein contact map prediction using multi-stage hybrid intelligence inference systems. *J Biomed Inform* .

[Adamczak *et al.*, 2005] Adamczak, R., Porollo, A. and Meller, J. Combining prediction of secondary structure and solvent accessibility in proteins. *Proteins: Structure, Function, and Bioinformatics*, 59(3), 467–475.

[Adelson-Velskii *et al.*, 1962] Adelson-Velskii, G. and Landis, E.M. An algorithm for the organization of information. *Proceedings of the USSR Academy of Sciences*, 146, p. 263–266 (Russian). English translation: Ricci, M.J. in *Soviet Math*, 3, p. 1259–1263.

[Altschul *et al.*, 1997] Altschul, S.F., Madden, T.L., Schäffer, A.A., Zhang, J., Zhang, Z. and Miller, W. and Lipman, D.J. Gapped BLAST and PSI-BLAST: a new generation of protein database search programs. *Nucleic Acids Res*, 25(17), 3389-402.

[Anfinsen, 1972] Anfinsen, C. The formation and stabilization of protein structure. *The Biochemical journal*, 128, 737–749.

[Asai *et al.*, 1993] Asai, K., Hayamizu, S. and Handa, K. Prediction of protein secondary structure by the hidden Markov model. *Comput Appl Biosci*, 9(2), 141–146.

[Asencio *et al.*, 2011a] Asencio, G., Aguilar-Ruiz, J.S. and Márquez, A.E. Prediction of protein distance maps by assembling fragments according to physicochemical similarities. 5th International Conference on Practical Applications of Computational Biology and Bioinformatics (PACBB 2011). *Advances in Intelligent and Soft Computing*, 93, pp. 271-278.

[Asencio *et al.*, 2011b] Asencio, G., Aguilar-Ruiz, J.S. and Márquez, A.E. A nearest neighbour-based approach for viral protein structure prediction. 9th European Conference on Evolutionary Computation, Machine

Learning and Data Mining in Bioinformatics (EvoBio 2011). *Lecture Notes in Computer Science*, 6623, pp. 69-76.

[Asencio *et al.*, 2011c] Asencio, G., Aguilar-Ruiz, J.S. and Márquez. A.E. Predicción de Mapas de Distancia de Proteínas basados en Vecinos más Cercanos. *Actas de la XIV Conferencia de la Asociación Española para la Inteligencia Artificial (CAEPIA 2011).*

[Asencio *et al.*, 2012] Asencio, G., Aguilar-Ruiz, J.S., Márquez, A.E., Ruiz, R. and Santiesteban, C.E. Prediction of Mitochondrial Matrix Protein Structures Based on Feature Selection and Fragment Assembly. 10th European Conference on Evolutionary Computation, Machine Learning and Data Mining in Bioinformatics (EvoBio 2012), *Lecture Notes in Computer Science*, 7246, pp. 156-167.

[Ashkenazy *et al.*, 2011] Ashkenazy, H., Unger, R. and Kliger, Y. Hidden conformations in protein structures. *Bioinformatics*, 27(14), 1941–1947.

[Aydin *et al.*, 2006] Aydin, Z., Altunbasak, Y. and Borodovsky, M. Protein secondary structure prediction for a single-sequence using hidden semi-Markov models. *BMC Bioinformatics*, 7, 178.

[Bacardit *et al.*, 2009] Stout, M., Bacardit, J., Hirst, J. and Krasnogor, N. Prediction of recursive convex hull class assignments for protein residues. *Bioinformatics*, 24(7), p. 916-923.

[Bacardit *et al.*, 2012] Bacardit, J., Widera, P., Márquez-Chamorro, A.E., Divina, F., Aguilar-Ruiz, J.S., Krasnogor, N.. Contact map prediction using a large-scale ensemble of rule sets and the fusion of multiple predicted structural features. *Bioinformatics* (IF: 5.468), 28(19), p. 2441-2448.

[Bairoch *et al.*, 2005] Bairoch, A., Apweiler, R., Wu, CH. Barker, WC., Boeckmann, B. and Ferro, S. The Universal Protein Resource (UniProt). *Nucl. Acids Res.*, 33:1, 154-159.

[Baum and Petrie, 1966] Baum, L.E. and Petrie, T. Statistical Inference for Probabilistic Functions of Finite State Markov Chains. *The Annals of Mathematical Statistics*, 37(6), 1554–1563.

[Berg and Stryer, 2008] Berg, J.M. and Stryer, L. Biochemistry. *Reverté.*

[Berman *et al.*, 2000] Berman, H.M., Westbrook, J., Feng, Z., Gilliland, G., Bhat, T.N., Weissig, H., Shindyalov, I.N., Bourne P.E. The Protein Data Bank. *Nucleic Acids Research*, 28, 235-242.

[Berman *et al.*, 2003] Berman, H.M., Henrick, K. and Nakmura, H. Announcing the worldwide Protein Data Bank. *Nature Structural Biology*, 10(12), 98.

[Biou *et al.*, 1988] Biou, V., Gibrat, JF., Levin, JM., Robson, B. and Garnier, J. Secondary structure prediction: combination of three different methods. *Protein Eng.*, 2(3), 185–191.

[Björkholm *et al.*, 2009] Björkholm, P., Daniluk, P., Kryshtafovych, A., Fidelis, K., Andersson, R., Hvidsten, T.R. Using multi-data hidden markov models trained on local neighborhoods of protein structure to predict residue-residue contacts. *Bioinformatics*, 25(10), 1264–1270.

[Blackburne and Hirst, 2001] Blackburne, B.P. and Hirst J.D. Evolution of functional model proteins. *J Chem Phy*, 115(4), 1935-42.

[Bohr *et al.*, 1990] Bohr, H., Bohr, J., Brunak, S., Cotterill, R.M.J., Fredholm, H., Lautrupt, B. and Petersen, S.B. A novel approach to prediction of the 3-dimensional structures of protein backbones by neural networks *FEBS Letters*, 261 (1), 43-46.

[Bordoloi *et al.*, 2012] Bordoloi, H., Sarma, KK. Protein Structure Prediction Using Multiple Artificial Neural Network Classifier. *Soft computing techniques in vision science. Studies in Computational Intelligence*, 395, 137–146.

[Bowie *et al.*, 1991] Bowie J.U., Luthy R., Eisenberg D. A method to identify protein sequences that fold into a known three-dimensional structure. *Science*, 253, 164-170.

[Braden, 2002] Braden, K. A Simple Approach to Protein Structure Prediction Using Genetic Algorithms. *Stanford University*, 426, 36-44.

[Brooks *et al.*, 1983] Brooks, B.R., Bruccoleri, R.E., Olafson, B.D., States, D.J., Swaminathan, S. and Karplus, M. CHARMM: A program for macromolecular energy, minimization, and dynamics calculations. *J. Comp. Chem.*, 4, 187-217.

[Bystroff *et al.*, 2000] Bystroff, C., Thorsson, V. and Baker, D. HMMSTR: A hidden Markov model for local sequence-structure correlations in proteins, *J Mol Biol*, 301, p. 173-190..

[Calvo *et al.*, 2009] Calvo, J.C., Ortega J. Parallel protein structure prediction by multiobjective optimization. *Parallel, Distributed and Network-based Processing*, 12(4), 407–413.

[Calvo *et al.*, 2011] Calvo, J.C., Ortega, J. and Anguita, M. Pitagoras-psp: Including domain knowledge in a multi-objective approach for protein structure prediction. *Neurocomputing*, 74(16), 2675–2682.

[Chandonia *et al.*, 1999] Chandonia, JM. and Karplus, M. New methods for accurate prediction of protein secondary structure. *Proteins: Structure, Function, and Bioinformatics*, 35(3), 293–306.

[Chandonia, 2007] Chandonia, J.M. StrBioLib: a Java library for development of custom computational structural biology applications. *Bioinformatics*, 23 (15), 2018-2020.

[Chatterjee *et al.*, 2011] Chatterjee, P., Basu, S., Kundu, M., Nasipuri, M. and Plewczynski, D. PSP-MCSVM: brainstorming consensus prediction of protein secondary structures using two-stage multiclass support vector machines. *Journal of Molecular Modeling*, 17(9), 2191–2201.

[Chen *et al.*, 2005] Chen P, Huanglt, W.B, Zhu, Y., Li, Y. Prediction of Contact Map Integrated PNN with Conformational Energy *Proc. IEEE Int. Joint Conf. Neur. Net. (IJCNN 05)*, 499-502.

[Chen *et al.*, 2009] Chen, C., Chen, L. and Zou, X. Prediction of Protein Secondary Structure Content by Using the Concept of Chous Pseudo Amino Acid Composition and Support Vector Machine *Protein and Peptide Letters*, 16(1), 27-31.

[Chen, 2010] Chen, P. Prediction of protein long-range contacts using an ensemble of genetic algorithm classifiers with sequence profile centers. *BMC Structural Biology*, 10(1), doi:10.1186/1472-6807-10-S1-S2.

[Cheng and Baldi, 2007] Cheng, J. and Baldi, P. Improved residue contact prediction using support vector machines and a large feature set. *Bioinformatics*, 8, 113.

[Chou *et al.*, 1974] Chou, PY., Fasman, G.D. Prediction of protein conformation. *Biochemistry*, 13(2), 222-45.

[Cohen, 2004] Cohen J. Bioinformatics — An Introduction for Computer Scientists *ACM Computing Surveys*, 36 (2), 122-158.

[Cole *et al.*, 2008] Cole, C., Barber, JD., and Barton, GJ. The Jpred 3 secondary structure prediction server. *Nucl. Acids Res.*, 36(2), 197–201.

[Conklin *et al.*, 1994] Conklin, D., Fortier, S., Glasgow, J.I. Knowledge Discovery of Multilevel Protein Motifs. *Proceedings of the Second International Conference on Intelligent Systems for Molecular Biology*, 2, 96-102.

[Corne *et al.*, 2000] Corne, D.W., Knowles, J.D., Oates, M.J. The Pareto Envelope-based Selection Algorithm for Multiobjective Optimization. *Proceedings of the Parallel Problem Solving from Nature VI Conference*, Lecture Notes in Computer Science, 1917, 839–848.

[Cortes *et al.*, 2011] Asencio-Cortes, G., Aguilar-Ruiz, J.S. Predicting protein distance maps according to physicochemical properties. *J Integr Bioinform*, 8(3), 181.

[Cotta, 2003] Cotta, C. Protein Structure Prediction Using Evolutionary Algorithms Hybridized with Backtracking *Lecture Notes in Computer Science*, 2687, 321-328.

[Cuff *et al.*, 1998] Cuff, J.A., Clamp, M.E., Siddiqui A.S., Finlay, M., Barton G.J. JPred: a consensus secondary structure prediction server. *Bioinformatics*, 14, 892-893.

[Cuff *et al.*, 1999] Cuff, JA., Barton, GJ. Evaluation and improvement of multiple sequence methods for protein secondary structure prediction. *Proteins: Structure, Function, and Bioinformatics*, 34(4), 508–519.

[Cui *et al.*, 1998] Cui, Y., Chen, R.S. and Hung, W. Protein Folding Simulation With Genetic Algorithm and Supersecondary Structure Constraints. *Proteins: Structure, Function and Genetics*, 31, 247-257.

[Cutello *et al.*, 2006] Cutello, V., Narzisi, G. and Nicosia, G. A multi-objective evolutionary approach to the protein structure prediction problem. *J. R. Soc. Interface*, 3, 139–151.

[Dandekar and Argos, 1992] Dandekar, T. and Argos, P. Potential of genetic algorithms in protein folding and protein engineering simulations. *Protein Engineering*, 5, 637-645.

[Dandekar and Argos, 1994] Dandekar, T. and Argos, P. Folding the main chain of small proteins with the genetic algorithm. *J. Mol. Biol.*, 236, 844–861.

[Davies *et al.*, 2006] Davies, J., Glasgow, J.I. and Kuo, T. Visio-Spatial Case-Based Reasoning: A Case Study in Prediction of Protein Structure. *Computational Intelligence*, 22(3-4), 194-207.

[Day *et al.*, 2002] Day, R.O., Zydallis, J.B., Pachter, G.L. Solving the protein structure prediction problem through a multiobjective genetic algorithm. *Nanotechnology*, 2, 32–35.

[Deb *et al.*, 2002] Deb, K., Pratap, A., Agarwal, S and Meyarivan, T. A Fast and Elitist Multiobjective Genetic Algorithm: NSGA-II. *IEEE Transactions on Evolutionary Computation*, 6:2, 182-197.

[Deleage and Roux, 1987] Deleage, G. and Roux, B. An algorithm for protein secondary structure prediction based on class prediction. *Protein Eng.*, 1(4), 289–294.

[Di Lena *et al.*, 2010] Di Lena, P., Fariselli, P., Margara, L., Vassura, M. and Casadio, R. Fast overlapping of protein contact maps by alignment of eigenvectors. *Bioinformatics*, 26(18), 2250–2258.

[Di Lena *et al.*, 2012] Di Lena, P., Nagata, K. and Baldi, P. Deep Architectures for Protein Contact Map Prediction. *Bioinformatics*.

[Dill, 1985] Dill, K.A. Dominant forces in protein folding. *Biochemistry*, 24, 1501.

[Ding *et al.*, 2001] Ding, C.H., Dubchak I. Multi-class protein fold recognition using support vector machines and neural networks. *Bioinformatics*, 17(4), 349-58.

[Dodge *et al.*, 1998] Dodge, C., Schneider, R. and Sander, C. The hssp database of protein structure-sequence alignments and family profiles. *Nucleic Acids Res*, 26(1), 313–315.

[Doig *et al.*, 1995] Doig, A.J., Baldwin R.L. N- and C-capping preferences for all 20 amino acids in alpha-helical peptides, *Protein Science*, 4(7), p. 1325-1336.

[Dong *et al.*, 2005] Dong, S., Liu, P., Cao, Y. and Du, Z. Grid Computing Methodology for Protein Structure Prediction and Analysis *ISPA Workshops*, 3759, 257-266.

[Duarte *et al.*, 2010] Duarte, J.M., Sathyapriya, R., Stehr, H., Filippis, I. and Lappe, M. Optimal contact definition for reconstruction of contact maps. *BMC Bioinformatics*, 11, 283.

[Eickholt *et al.*, 2011] Eickholt, J., Wang, Z. and Cheng, J. A conformation ensemble approach to protein residue-residue contact. *BMC Struct Biol*, 11, 38.

[Faraggi *et al.*, 2009] Faraggi, E., Yang, Y., Zhang, S. and Zhou, Y. Predicting continuous local structure and the effect of its substitution for secondary structure in fragment-free protein structure prediction. *Structure*, 17(11), 1515–1527.

[Faraggi *et al.*, 2012] Faraggi, E., Zhang, T., Yang, Y., Kurgan, L. and Zhou, Y. SPINE X: Improving protein secondary structure prediction by multi-step learning coupled with prediction of solvent accessible surface area and backbone torsion angles. *Journal of Computational Chemistry*, 33(3), 259–267.

[Fariselli *et al.*, 2001] Fariselli, P., Olmea, O., Valencia, A. and Casadio, R. Prediction of contact map with neural networks and correlated mutations. *Protein Engineering*, 14, 133–154.

[Fariselli and Casadio, 1999] Fariselli, P. and Casadio, R. A neural network based predictor of residue contacts in proteins. *Protein Engineering*, 12, 15-21.

[FarzadFard *et al.*, 2008] FarzadFard, F., Gharaei, N., Pezeshk, H. and Marashi, S. Beta-Sheet capping: Signals that initiate and terminate beta-sheet formation, *J. Structural Biology*, 161, p. 101-110.

[Faure *et al.*, 2008] Faure, G., Bornot, A., de Brevern, A.G. Protein contacts, inter-residue interactions and side-chain modelling. *Biochimie*, 90(4), 626–639.

[Fernandez *et al.*, 2009] Fernandez, M., Paredes, A., Ortiz, L. and Rosas, J. Sistema predictor de estructuras de proteínas utilizando dinámica molecular (modypp). *Revista Internacional de Sistemas Computacionales y Electrónicos* pp. 6–16.

[Fogel *et al.*, 1966] Fogel, L.J., Owens, A. J. and Walsh, M. Artificial Intelligence through Simulated Evolution. *Wiley.*

[Fonseca *et al.*, 1993] Fonseca, C.M., Fleming P.J. Genetic algorithms for multiobjective optimization: Formulation, discussion and generalization. S. Forrest (Ed.), *Proc. 5th Int. Conf. on Genetic Algorithms*, 416–423.

[Fonseca *et al.*, 2007] Fonseca, N.A., Camacho, R., Magalhaes, A.L. Amino acid pairing at the N- and C-termini of helical segments in proteins, *Proteins*, 70, p. 188-196.

[Fornes *et al.*, 2009] Fornes, O., Aragues, R., Espadaler, J., Marti-Renom, M.A., Sali, A., and Oliva, B. ModLink: improving fold recognition by using protein-protein interactions. *Bioinformatics*, 25, 1506-1512.

[Francesco *et al.*, 1996] Francesco, V., Garnier, J., Munson, PJ. Improving protein secondary structure prediction with aligned homologous sequences. *Protein Science*, 5(1), 106–113.

[Frishman and Argos, 1996] Frishman, D. and Argos, P. Incorporation of non-local interactions in protein secondary structure prediction from the amino acid sequence. *Protein Engineering*, 9, 133-142.

[Furuta *et al.*, 2009] Furuta, T., Shimizu, K. and Terada, T. Accurate prediction of native tertiary structure of protein using molecular dynamics simulation with the aid of the knowledge of secondary structures. *Chemical Physics Letters*, 472, 134–139.

[Gao *et al.*, 2009] Gao, X., Bu, D. and Xu, J., Li, M. Improving consensus contact prediction via server correlation reduction. *BMC Struct Biol*, 9, 28.

[Garnier *et al.*, 1978] Garnier, J., Osguthorpe, D.J. and Robson, B. Analysis of the accuracy and implications of simple methods for predicting the secondary structure of globular proteins. *J. Mol. Biol.*, 120, 97-120.

[Geourjon and Deleage, 1994] Geourjon, C. and Deleage, G. SOPM: a self-optimized method for protein secondary structure prediction. *Protein Eng.*, 7(2), 157–164.

[Geourjon and Deleage, 1995] Geourjon, C. and Deleage, G. SOPMA: significant improvements in protein secondary structure prediction by consensus prediction from multiple alignments. *Bioinformatics*, 11(6), 584–601.

[Giraldez *et al.*, 2005] Giraldez, R., Aguilar-Ruiz, J.S. Riquelme,J.C. Knowledge-based Fast Evaluation for Evolutionary Learning. *IEEE Transactions on Systems, Man and Cybernetics, Part C*, Vol 35(2) p. 254-261.

[Glasgow *et al.*, 2006] Glasgow, J., Kuo, T. and Davies, J. Protein structure from contact maps: A case-based reasoning approach *Inf Sys Front*, 8, 29-36.

[Goldberg, 1989] Goldberg, E. Genetic Algorithms in Search, Optimization, and Machine Learning. Addison-Wesley Publishing Company..

[González *et al.*, 2007] González, J.R., Pelta, D.A. On Using Fuzzy Contact Maps for Protein Structure Comparison *IEEE*, 1, 1650-1655.

[Gorodkin *et al.*, 1999] Gorodkin, J., Lund, O., Andersen, C.A. and Brunak, S. Using Sequence Motifs for Enhanced Neural Network Prediction of Protein Distance Constraints. *Proc. Int. Conf. Intell. Syst. Mol. Biol.*, 95-105.

[Grantham, 1974] Grantham, R. Amino acid difference formula to help explain protein evolution. *J. J. Mol. Bio.*, 185, 862–864.

[Greenwood *et al.*, 1999] Greenwood, G.W., Shin, J.M. and Lee, B. and Fogel, G.B. A survey of Recent Work on evolutionary approaches to the Protein Folding Problem. *IEEE*, 99, 488-495.

[Gu and Bourne, 2003] Gu, J. and Bourne, P.E. *Structural Bioinformatics (Methods of Biochemical Analysis).* Wiley-Blackwell.

[Guo *et al.*, 2004] Guo, J., Chen, H., Sun, Z. and Lin, Y. A novel method for protein secondary structure prediction using dual-layer SVM and profiles. *Proteins Structure Function and Bioinformatics*, 54, 738-743.

[Gupta *et al.*, 2005] Gupta, N., Mangal, N. and Biswas, S. Evolution and similarity evaluation of protein structures in contact map space. *Proteins: Structure, Function, and Bioinformatics*, 59, 196–204.

[Hall *et al.*, 2009] Hall, M., Frank, E., Holmes, G., B., P., Reutemann, P. and Witten, I. *The weka data mining software: An update.* SIGKDD Explorations 11.

[Han *et al.*, 2005] Han, S., Lee, B., Yu, ST., Jeong, C., Lee, S. and Kim, D. Fold recognition by combining profile profile alignment and support vector machine. *Bioinformatics*, 21, 2667-2673.

[Hardin, 2002] Hardin, C., Pogorelov, Taras V. and Luthey-Schulten, Z. Ab initio protein structure prediction. *Courrent Opinion in Structural Biology*, 12, 176-181.

[Holland, 1975] Holland, J.H. Adaptation in Natural and Artificial Systems, University of Michigan Press, Ann Arbor.

[Holley *et al.*, 1989] Holley, LH. and Karplus, M. Protein secondary structure prediction with a neural network. *PNAS*, 86(1), 152–156.

[Horn and Nafpliotis, 1993] Horn, J. and Nafpliotis, N. Multiobjective Optimization Using the Niched Pareto. *Genetic Algorithms*, IlliGAL Report 93005, University of Illinois, Urbana, Champaign, July.

[Hu *et al.*, 2004] Hu, HJ. Improved protein secondary structure prediction using support vector machine with a new encoding scheme and an advanced tertiary classifier. *NanoBioscience, IEEE Transactions on*, 3(4), 265–271.

[Hua and Sun, 2001] Hua, S. and Sun, Z. A novel method of protein secondary structure prediction with high segment overlap measure: Support Vector Machine Approach. *J. Mol. Biol.*, 308, 397-407.

[Huang *et al.*, 2003] Huang, C., Lin, C. and Pal, N. Hierarchical learning architecture with automatic fearture selection for multiclass protein fold classification. *IEEE transactions on NanoBioscience*, 2(4), 221–232.

[Islam and Chetty, 2009] Islam, K. and Chetty, M. Novel Memetic Algorithm for Protein Structure Prediction. *Lecture Notes in Artificial Intelligence*, 5866, 412-421.

[Janin, 2010] Janin, J. Protein-protein docking tested in blind predictions: the CAPRI experiment. *Mol Biosyst*, 6, 2351-2362.

[Jaravine *et al.*, 2006] Jaravine, V., Ibraghimov, I. and Yu Orekhov, V. Removal of a time barrier for high-resolution multidimensional nmr spectroscopy. *Nat Meth*, 3(8), 605–607.

[Jones *et al.*, 2012] Jones, D.T., Buchan, D.W.A., Cozzetto, D. and Pontil, M. Psicov: precise structural contact prediction using sparse

inverse covariance estimation on large multiple sequence alignments. *Bioinformatics*, 28(2), 184–190.

[Jones, 1999] Jones, D. Protein secondary structure prediction based on position-specific scoring matrices. *Journal of Molecular Biology*, 292, 195–202.

[Joo *et al.*, 2004] Joo, K., Kim, I., Kim, S. and Lee, J. Prediction of the Secondary Structures of Proteins by Using PREDICT, a Nearest Neighbor Method on Pattern Space. *Journal of the Korean Physical Society*, 45(6), 1441–1449, December.

[Judson *et al.*, 1993] Judson RS, Jaeger EP, Treasurywala AM, Peterson ML Conformational searching methods for small molecules. II. Genetic algorithm approach. *J. Comp. Chem.*, 14, 1407-1414.

[Judy *et al.*, 2009] Judy, M.V., Ravichandran, K.S. and Murugesan, K. A multi-objective evolutionary algorithm for protein structure prediction with immune operators. *Comput Methods Biomech Biomed Engin*, 12(4), 407–413.

[Kabsch and Sander, 1983] Kabsch, W. and Sander, C. Dictionary of protein secondary structure: pattern recognition of hydrogen-bonded and geometrical features, *Biopolymers*, 22(12), p. 2577-2637.

[Karplus, 2009] Karplus, K. SAM-T08, HMM-based protein structure prediction. *Nucl. Acids Res.*, 37(2), 492-497.

[Karypis, 2006] Karypis, G. YASSPP: Better Kernels and Coding Schemes Lead to Improvements in Protein Secondary Structure Prediction. *Proteins: Structure, Functions and Bioinformatics*, 64, 575-586.

[Katzman *et al.*, 2008] Katzman, S., Barrett, C., Thiltgen, G. and Karchin, R. PREDICT-2ND: a tool for generalized protein local structure prediction. *Bioinformatics*, 24, 2453-2459.

[Kawashima *et al.*, 2008] Kawashima, S., Pokarowski, P., Pokarowska, M., Kolinski, A., Katayama, T. and Kanehisa, M. Aaindex: amino acid index database, progress report 2008. *Nucleic Acids Res*, 36(Database issue), D202–D205.

[Kehyayan and Mansour, 2008] Kehyayan, C. and Mansour, N. Evolutionary Algorithm for Protein Structure Prediction. *International Conference on Advanced Computer Theory and Engineering*, 199, 133-154.

[Kihara, 2005] Kihara, D. The effect of long-range interactions on the secondary structure formation of proteins. *Protein Sci*, 14(8), 1955–1963.

[Kim *et al.*, 2006] Kim, S-Y., Sim, J. and Lee, J. Fuzzy k-nearest neighbor method for protein secondary structure prediction and its parallel implementation. *Proceedings of the 2006 international conference on Computational Intelligence and Bioinformatics* - Volume Part III, 444-453.

[Kim and Park, 2003] Kim, H. and Park, H. Prediction of protein relative solvent accessibility with support vector machines and long-range interaction 3D local descriptor. *Proteins*, 54, 557-562.

[Kim, 2004] Kim, S. Protein beta-turn prediction using nearest-neighbor method. *Bioinformatics*, 20(1), 40-44.

[King *et al.*, 1990] King, RD., Sternberg, MJE. Machine learning approach for the prediction of protein secondary structure. *Journal of Molecular Biology*, 216(2), 441–457.

[King *et al.*, 1996] King, R.D., Sternberg M.J.E. Identification and application of the concepts important for accurate and reliable protein secondary structure prediction. *Protein Science*, 5, 2298-2310.

[Kinjo *et al.*, 2005] Kinjo, A.R., Horimoto, K., and Nishikawa, K. Predicting absolute contact numbers of native protein structure from amino acid sequence. *Proteins*, 58, 158–165.

[Klein *et al.*, 1984] Klein, P., Kanehisa, M. and DeLisi, C. Prediction of protein function from sequence properties: Discriminant analysis of a data base. *Biochim. Biophys.*, 787, 221–226.

[Kloczkowski *et al.*, 2002] Kloczkowski, A., Ting, KL., Jernigan, RL. and Garnier, J. Combining the GOR V algorithm with evolutionary information for protein secondary structure prediction from amino acid sequence. *Proteins: Structure, Function, and Bioinformatics*, 49(2), 154–166.

[Kloczkowski *et al.*, 2009] Kloczkowski, A., Jernigan, R., Wu, Z., Song, G., Yang, L., Kolinski, A. and Pokarowski, P. Distance matrix-based approach to protein structure prediction. *Journal of Structural and Functional Genomics*, 10, 67–81.

[Kneller *et al.*, 1990] Kneller, D.G., Cohen, F.E. and Langridge, R. Improvements in Protein Secondary Structure Prediction by An Enhanced Neural Network. *J. Mol. Biol.*, 214, 171-182.

[Kneller *et al.*, 1990] Kneller, DG., Cohen, FE. and Langridge, R. Improvements in protein secondary structure prediction by an enhanced neural network.. *J. Mol. Biol.*, 214(1), 171–182.

[Kohonen and Makisara, 1989] Kohonen, T. and Makisara, K. The self-organizing feature maps. *Phys. Scripta*, 39, 168-172.

[Kosinski *et al.*, 2003] Kosinski, J., Cymerman, I.A., Feder, M., Kurowski, M.A., Sasin, J.M. and Bujnicki, J.M. A Frankensteins monster approach to comparative modeling: merging the finest fragments of fold-recognition models and iterative model refinement aided by 3D structure evaluation. *Proteins*, 53 Suppl 6, 369-79.

[Koza, 1992] Koza, J. R. Genetic Programming: On the Programming of Computers by Natural Selection. *MIT Press, Cambridge, MA.*.

[Krasnogor *et al.*, 2002] Krasnogor, N., Blackbourne, B.P., Burke, E.K., Hirst, J.D. Multimeme Algorithms for Protein Structure Prediction. *Lecture Notes in Computer Science*, 2439, 769-778.

[Kyte and Doolittle, 1982] Kyte, J. and Doolittle, R. A simple method for displaying the hydropathic character of a protein. *J. J. Mol. Bio.*, 157, 105–132.

[Lattman, 2004] Lattman, E. The state of the protein structure initiative. *Proteins*, 54(4), 611–615.

[Lavor *et al.*, 2011] Lavor, C., Liberti, L., Maculan, N. and Mucherino, A. Recent advances on the discretizable molecular distance geometry problem. *European Journal of Operational Research.*

[Leng *et al.*, 1993] Leng, B., Buchanan, B.G., Nicholas, H.B. Protein secondary structure prediction using two-level case-based reasoning. *J Comput Biol.* Spring 1(1), 25-38.

[Levin *et al.*, 1986] Levin, JM., Robson, B. and Garnier, J. An algorithm for secondary structure determination in proteins based on sequence similarity. *FEBS letters*, 205(2), 303–308.

[Levin *et al.*, 1997] Levin, JM. Exploring the limits of nearest neighbour secondary structure prediction. *Protein Eng.*, 10(7), 771–776.

[Li *et al.*, 2011] Li, Y., Fang, Y. and Fang, J. Predicting residue-residue contacts using random forest models. *Bioinformatics*, 27(24), 3379–3384.

[Liang *et al.*, 2001] Liang, F., Wonh, W.H. Evolutionary monte carlo for protein folding simulations. *J. Chem. Phy.*, 115(7), 3374-3380.

[Lim, 1974] Lim, V.I. Algorithms for Prediction of a-Helical and b-Structural Regions in Globular Proteins. *J. Mol. Biol.*, 88, 857-872.

[Lin *et al.*, 2005] Lin, K., Simossis, V.A., Taylor, W.R. and Heringa, J. A simple and fast secondary structure prediction method using hidden neural networks. *Bioinformatics*, 21, 152-159.

[Lin *et al.*, 2005] Lin, H., Chang, J., Wu, K., Sung, T. and Hsu, W. HYPROSP II-A knowledge-based hybrid method for protein secondary structure prediction based on local prediction confidence. *Bioinformatics*, 21(15), 3227–3233.

[Lippi and Frasconi, 2009] Lippi, M. and Frasconi, P. Prediction of protein beta-residue contacts by markov logic networks with grounding-specific weights. *Bioinformatics*, 25(18), 2326–2333.

[Liu *et al.*, 2005] Liu, G., Zhou, C., Zhu, Y. and Zhou, W. Prediction of Contact Maps in Proteins Based on Recurrent Neural Network with Bias Units *LNCS*, 3498, 686-690.

[Liu *et al.*, 2006] Liu Z., Yuanxian, Z.W. Prediction of Contact Maps Using Modified Transiently Chaotic Neural Network *LNCS*, 3973, 696-701.

[Livs *et al.*, 1998] Livs, P., Goldman, N., Thorne, JL., Jones, DT. PASSML: combining evolutionary inference and protein secondary structure prediction. *Bioinformatics*, 14(8), 726–733.

[Lo *et al.*, 2009] Lo, A., Chiu, Y.Y., Rødland, E.A., Lyu, P.C., Sung, T.Y., Hsu, W.L. Predicting helix-helix interactions from residue contacts in membrane proteins. *Bioinformatics*, 25(8), 996–1003.

[Lo Conte *et al.*, 2000] Lo Conte, L., Ailey, B., Hubbard, T. J., Brenner, S.E., Murzin, A.G. and Chothia, C. SCOP: A Structural Classification of Proteins database. *Nucleic Acids Research*, 28 (1), 257–259.

[Márquez *et al.*, 2009] Márquez, A.E., Aguilar-Ruiz, J.S. and Anguiano, E. Marco de Referencia en la Calidad de la Predicción de Mapas de Contacto de Proteínas. In: *Actas de la XIII Conferencia de la Asociación Española para la Inteligencia Artificial (CAEPIA 2009)*, pp. 11-19.

[Márquez *et al.*, 2010a] Márquez, A.E., Divina, F., Aguilar-Ruiz, J.S. and Asencio, G. Alpha helix prediction based on evolutionary computation. Proceedings of the 5th IAPR international conference on Pattern recognition in bioinformatics (PRIB 2010), *Lecture Notes in Computer Science*, 6282, pp. 358-367.

[Márquez *et al.*, 2010b] Márquez, A.E., Aguilar-Ruiz, J.S. and Anguiano, E. Definición de umbral mínimo para la predicción de estructura secundaria de proteínas. *Actas del XV Congreso Español sobre Tecnologías y Lógica Fuzzy (ESTYLF 2010)*, pp. 465-470.

[Márquez *et al.*, 2011a] Márquez, A.E., Divina, F., Aguilar-Ruiz, J.S. A Multi-objective Genetic Algorithm for the Protein Structure Prediction. In: *Proceedings of the 11th Annual ACM on Intelligent Systems Design and Applications (ISDA 2011)*, pp.1086-1090.

[Márquez *et al.*, 2011b] Márquez, A.E., Divina, F., Aguilar-Ruiz, J.S. Protein Secondary Structures Prediction based on Evolutionary Computation. *Applied computing Review*, ACM SIGAPP, 11-4, pp. 17-25.

[Márquez *et al.*, 2011c] Márquez, A.E., Divina, F., Aguilar-Ruiz, J.S. Evolutionary Computation for the Prediction of Secondary Protein Structures. *Proceedings of the 26th Annual ACM Symposium on Applied Computing (SAC-2011)*, pp. 1087-1092.

[Márquez *et al.*, 2011d] Márquez, A.E., Divina, F., Aguilar-Ruiz, J.S. Evolutionary Protein Contact Maps Prediction based on Amino Acid Properties. 6th International Conference on Hybrid Artificial Intelligent Systems (HAIS 2011), *Lecture Notes in Computer Science*, 6678, pp. 303-310.

[Márquez *et al.*, 2011e] Márquez, A.E., Divina, F., Aguilar-Ruiz, J.S. and Asencio, G. Residue-residue Contact Prediction based on Evolutionary Computation. A E Marquez, F Divina, J S Aguilar-Ruiz, G Asencio. 5th International Conference on Practical Applications of Computational Biology and Bioinformatics (PACBB 2011), *Advances in Intelligent and Soft Computing*, 93/2011, pp. 279-283.

[Márquez *et al.*, 2011f] Márquez, A.E., Divina, F., Aguilar-Ruiz, J.S. and Asencio, G. An Evolutionary Approach for Protein Contact Map Prediction. 9th European Conference on Evolutionary Computation, Machine Learning and Data Mining in Bioinformatics (EvoBio 2011), *Lecture Notes in Computer Science*, 6623, pp. 101-110.

[Márquez *et al.*, 2011g] Márquez, A.E., Divina, F., Aguilar-Ruiz, J.S. and Asencio, G. Un Algoritmo Genético para la Predicción de Mapas de Contacto Basado en Propiedades de Aminoácidos. *Actas de la XIV Conferencia de la Asociación Española para la Inteligencia Artificial (CAEPIA 2011).*

[Márquez *et al.*, 2012a] Márquez, A.E., Asencio, G., Divina, F., Aguilar-Ruiz, J.S. Evolutionary Decision Rules for Predicting Protein Contact Maps. *Pattern Analysis and Applications, PAAA*, Springer, (IF: 1.097), September 2012, pp. 1-13.

[Márquez *et al.*, 2012b] Márquez, A.E., Divina, F., Aguilar-Ruiz, J.S., Bacardit, J. and Asencio, G. A NSGA-II Algorithm for the Residue-Residue Contact Prediction. In: 10th European Conference on Evolutionary Computation, Machine Learning and Data Mining in Bioinformatics (EvoBio 2012), *Lecture Notes in Computer Science*, 7246, pp. 234-244.

[MacCallum, 2004] MacCallum, R.M. Striped sheets and protein contact prediction. *Bioinformatics*, 20, 224-231.

[Marks *et al.*, 2011] Marks, D.S., Colwell, L.J., Sheridan, R., Hopf, T.A., Pagnani, A., Zecchina, R. and Sander, C. Protein 3d structure computed from evolutionary sequence variation. *PLoS ONE*, 6(12), 766.

[McGuffin *et al.*, 2000] McGuffin, L.J, Bryson K. and Jones, D.T. The PSIPRED protein structure prediction server. *Bioinformatics*, 16, 404-405.

[Mehta *et al.*, 1995] Mehta, PK., Heringa, J. and Argos, P. A simple and fast approach to prediction of protein secondary structure from multiply aligned sequences with accuracy above 70%. *Protein Science*, 4(12), 2517–2525.

[Mittleman *et al.*, 2003] Mittelman, D., Sadreyev, R. and Grishin, N. Probabilistic scoring measures for profile–profile comparison yield more accurate short seed alignments. *Bioinformatics*, 19, 1531–1539.

[Monastyrskyy *et al.*, 2011] Monastyrskyy, B., Fidelis, K., Tramontano, A. and Kryshtafovych, A. Evaluation of residue-residue contact predictions in casp9. *Proteins: Structure, Function, and Bioinformatics*, 79(S10), 119–125.

[Montgomerie *et al.*, 2006] Montgomerie, S., Sundararaj, S., Gallin, WJ. and Wishart, DS. Improving the accuracy of protein secondary structure prediction using structural alignment. *BMC Bioinformatics*, 7, 301.

[Muggleton *et al.*, 1992] Muggleton, S., King, RD. and Stenberg, MJE. Protein secondary structure prediction using logic-based machine learning. *Protein Eng.*, 5(7), 647–657.

[Munson *et al.*, 1994] Munson, P.J., Di-Francesco, V. and Porrelli, R. Protein secondary structure prediction using periodic-quadratic-logistic models: Statistical and technical issues. *Proceedings of 27th Hawaii International Conference on Sysiem Sciences*, 5, 375-384.

[Murzin *et al.*, 1995] Murzin, A., Brenner, S., Hubbard, T. and Chothia, C. Scop: a structural classification of proteins database for the investigation of sequences and structures. *Journal of Molecular Biology*, 247, 536–540.

[Muskal *et al.*, 1992] Muskal, SM. and Kim, SH. Predicting protein secondary structure content: A tandem neural network approach. *Journal of Molecular Biology*, 3(5), 713–727.

[Nagata *et al.*, 2012] Nagata, K., Randall, A. and Baldi, P. Sidepro: A novel machine learning approach for the fast and accurate prediction of side-chain conformations. *Proteins*, 80(1), 142–153.

[Olmea *et al.*, 1999] Olmea, O., Rost, B. and Valencia, A. Effective Use of Sequence Correlation and Conservation in Fold Recognition. *J. Mol. Biol.*, 295, 1221-1239.

[Olson *et al.*, 2012] Olson, A.L., Thompson, R.J., Melander, C. and Cavanagh, J. Chemical shift assignments and secondary structure prediction of the C-terminal domain of the response regulator BfmR from Acinetobacter baumannii. *Biomol NMR Assign*, (in print).

[Orengo *et al.*, 1997] Orengo, C.A., Michie, A.D., Jones, S., Jones, D.T., Swindells, M.B. and Thornton, J.M. CATH–a hierarchic classification of protein domain structures. *Structure*, 5(8), 1093–1108.

[Ouali *et al.*, 2000] Ouali, M. and King, RD. Cascaded multiple classifiers for secondary structure prediction. *Protein Science*, 9(6), 1162–1176.

[Pedersen and Moult, 1997] Pedersen, J.T. and Moult, J. Protein folding simulations with genetic algorithms and a detailed molecular description. *J. Mol. Biol.*, 269, 240–259.

[Pelta and Krasnogor, 2005] Pelta, D. and Krasnogor, N. Multimeme Algorithms Using Fuzzy Logic Based Memes For Protein Structure Prediction. *Studies in Fuzziness and Soft Computing*, 166, 49-64.

[Petersen *et al.*, 2000] Petersen, T.N., Lundegaard, C., Nielsen, M., Bohr, J., Brunak, S., Gippert, G.P., Lund O. Prediction of Protein Secondary Structure at 80% Accuracy. *Proteins: Structure, Function, and Genetics*, 41, 17-20.

[Petersen and Taylor, 2003] Petersen, K. and Taylor, W.R. Modelling zinc-binding proteins with GADGET: genetic algorithm and distance geometry for exploring topology. *J. Mol. Biol*, 325, 1039-1059.

[Piccolboni and Mauri, 1998] Piccolboni, A. and Mauri, G. Aplication of evolutionary algorithms to protein foldin prediction *Lecture Notes in Computer Science*, 1363, 123-135.

[Plaxco *et al.*, 1998] Plaxco, K.W., Simons, K.T. and Baker, D. Contact order, transition state placement and the refolding rates of single domain proteins. *J Mol Biol*, 277(4), 985–994.

[Pollastri *et al.*, 2002] Pollastri, G., Przybylski, D., Rost, B. and Baldi, P. Improving the Prediction of Protein Secondary Structure in Three and Eight Classes Using Recurrent Neural Networks and Profiles. *Proteins: Structure, Functions and Bioinformatics*, 47-2, 228-235.

[Pollastri *et al.*, 2007] Pollastri, G., Martin, AJM., Mooney, C. and Vullo, A. Accurate prediction of protein secondary structure and solvent accessibility by consensus combiners of sequence and structure information. *BMC Bioinformatics*, 8, 201.

[Pollastri and Baldi, 2002] Pollastri, G. and Baldi, P. Prediction of Contact Maps by Recurrent Neural Propagation From All Four Cardinal Corners *Bioinformatics*, 1 (1), 1-9.

[Pollastri and Mclysaght, 2005] Pollastri, G. and Mclysaght, A. Porter: a new, accurate server for protein secondary structure prediction. *Bioinformatics*, 21(8), 1719–1720.

[Punta and Rost, 2005] Punta, M. and Rost, B. PROFcon: novel prediction of long-range contacts. *Bioinformatics*, 21, 2960-2968.

[Qian *et al.*, 1988] Qian N., Sejnowski T.J. Predicting the secondary structure of globular proteins using neural network models. *J. Mol. Biol.*, 202, 865-884.

[Qu *et al.*, 2012] Qu, W., Yang, B., Jiang, B. and Wang, L. HYBP-PSSP: a hybrid back propagation method for predicting protein secondary structure. *Neural computing and applications*, 21(2), 337–349.

[Rajgaria *et al.*, 2009] Rajgaria, R., McAllister, S.R., Floudas, C.A. Towards accurate residue-residue hydrophobic contact prediction for alpha helical proteins via integer linear optimization. *Proteins*, 74(4), 929–947.

[Rajgaria *et al.*, 2010] Rajgaria, R., Wei, Y., Floudas, C.A. Contact prediction for beta and alpha-beta proteins using integer linear optimization and its impact on the first principles 3d structure prediction method astro-fold. *Proteins*, 78(8), 1825–1846.

[Ramachandran *et al.*, 1965] Ramakrishnan, C., Ramachandran, G.N. Stereochemical criteria for polypeptide and protein chain conformation. *Byophys Journal*, 5, 909-933.

[Ramanathan *et al.*, 2008] Ramanathan, A., Agarwal, P.K. and Langmead, C.J. Using Tensor Analysis to characterize Contact-map Dynamics of Proteins *School of Computer Science Carnegie Mellon University Pittsburgh*, 1-24.

[Raval *et al.*, 2002] Raval, A., Ghahramani, Z., Wild, DL. Bayesian network model for protein fold and remote homologue recognition. *Bioinformatics*, 18, 788-801.

[Rechenberg, 1973] Rechenberg, I. Evolutionsstrategie: Optimierung technischer Systeme nach Prinzipien der biologischen Evolution. *Friedrich Frommann.*

[Richardson *et al.*, 1998] Richardson, J.S., Richardson, D.C. Amino Acid Preferences for Specific Locations at the Ends of Alpha Helices, *Science*, 240, p. 1648-1652.

[Riis *et al.*, 1996] Riis, SK. and Krogh, A. Improving Prediction of Protein Secondary Structure Using Structured Neural Networks and Multiple Sequence Alignments. *Journal of Computational Biology*, 3(1), 163–183.

[Robles *et al.*, 2004] Robles, V., Larranaga, P., Pena, PM. and Menasalvas, E. Bayesian network multi-classifiers for protein secondary structure prediction. *Artificial Intelligence in Medicine*, 31(2), 117–136.

[Robson, 1974] Robson, B. Analysis of the Code Relating Sequence to Conformation in Globular Proteins *Biochemistry Jurnal*, 141, 853-867.

[Rost *et al.*, 1994] Rost, B., Sander, C. and Schneider, R. Redefining the goals of protein secondary structure prediction. *Journal of Molecular Biology*, 235, 13-26.

[Rost and Sander, 1993] Rost, B. and Sander, C. Prediction of protein secondary structure at better than 70% accuracy. *J. Mol. Biol.*, 232, 584-599.

[Rost and Sander, 1994] Rost, B. and Sander, C. Combining evolutionary information and neural networks to predict protein secondary structure. *Proteins: Structure, Function, and Bioinformatics*, 19(1), 55–72.

[Roy *et al.*, 2010] Roy, A., Kucukural, A. and Zhang, Y. I-tasser: a unified platform for automated protein structure and function prediction. *Nat Protoc*, 5(4), 725–738.

[Salamov *et al.*, 1997] Salamov, A.A., Solovyev, V.V. Protein Secondary Structure Prediction Using Local Alignments. *J. Mol. Biol.*, 268,31-36.

[Santiesteban *et al.*, 2011] Santiesteban, C.E., Márquez, A.E., Asencio, G., Aguilar-Ruiz, J.S. A Decision Tree-Based Method for Protein Contact Map Prediction. 9th European Conference on Evolutionary Computation, Machine Learning and Data Mining in Bioinformatics (EvoBio 2011), *Lecture Notes in Computer Science*, 6623, pp. 224-233.

[Santiesteban *et al.*, 2012] Santiesteban, C.E., Asencio, G., Márquez, A.E., Aguilar-Ruiz, J.S. Short-Range Interactions and Decision Tree-Based Protein Contact Map Predictor. 10th European Conference on Evolutionary Computation, Machine Learning and Data Mining in Bioinformatics (EvoBio 2012), *Lecture Notes in Computer Science*, 7246, pp. 224-233.

[Schmidler *et al.*, 2000] Schmidler, SC., Liu, JS., Brutlag, DL. Bayesian Segmentation of Protein Secondary Structure. *Journal of Computational Biology*, 7(1-2), 233–248.

[Schulze-Kremer *et al.*, 2000] Schulze-Kremer, S. Genetic Algorithms and Protein Folding. *Protein Structure Prediction: Methods and protocols*, 9, 175-222.

[Selbig *et al.*, 1999] Selbig, J., Mevissen, T. and Lengauer, T. Decision tree-based formation of consensus protein secondary structure prediction. *Bioinformatics*, 15(12), 1039–1046.

[Sen *et al.*, 2005] Sen, TZ., Jernigan, RL., Garnier, J. and Kloczkowski, A. GOR V server for protein secondary structure prediction. *Bioinformatics*, 21(11), 2787–2788.

[Service, 2005] Service, R. Structural biology: structural genomics, round 2. *Science*, 307, 1554–1558.

[Shi *et al.*, 2004] Shi, S.Y.M, Suganthan, N. Multi-Class Protein Fold Recognition Using Multi-Objective Evolutionary Algorithms. *KanGAL Report*, 2004007, 1–7.

[Simons *et al.*, 1999] Simons, KT., Bonneau, R., Ruczinski, I. and Baker, D. Ab initio protein structure prediction of CASP III targets using ROSETTA. *Proteins: Structure, Function, and Bioinformatics*, 37(3), 171–176.

[Song and Burrage, 2006] Song, J. and Burrage, K. Predicting residue-wise contact orders in proteins by support vector regression. *BMC Bioinformatics*, 7, 425.

[Srinivas *et al.*, 1995] Srinivas N., Debl K. Multiobjetive Optimization Using Nondominated Sorting in Genetic Algorithms. *Evolutionary Computation*, 2, 221–248.

[Stolorz *et al.*, 1992] Stolorz, P., Lapedes, A. and Xia, Y. Predicting protein secondary structure using neural net and statistical methods. *Journal of Molecular Biology*, 225(2), 363–377.

[Stout *et al.*, 2008] Stout, M., Bacardit, J. and Hirst, J. D. and Krasnogor, N. Prediction of recursive convex hull class assignments for protein residues. *Bioinformatics*, 24(7), 916–923.

[Subramani *et al.*, 2012] Subramani, A., Wei, Y., Floudas, CA. ASTRO-FOLD 2.0: An enhanced framework for protein structure prediction. *AIChE Journal*, 58(5), 1619–1637.

[Tegge *et al.*, 2009] Tegge, AN., Wang, Z., Eickholt, J. and Cheng, J. NNcon: Improved Protein Contact Map Prediction Using 2D-Recursive Neural Networks. *Nucleic Acids Research*, 37(2), 515-518.

[Tiwari *et al.*, 2008] Tiwari, S., Koch, P., Fadel, G. and Deb, K. AMGA: an archive-based micro genetic algorithm for multi-objective optimization. *Proceedings Genetic and Evolutionary Computation Conference - GECCO*, pp. 729-736.

[Toona, 2012] Toona, G.W. A dynamical approach to contact distance based protein structure determination. *Journal of Molecular Graphics and Modelling*, 32, 75–81.

[Tradigo, 2009] Tradigo G. On the Integration of Protein Contact Map Predictions *IEEE*, 1-5.

[Unger and Moult, 1993] Unger, R. and Moult, J. Genetic algorithms for protein folding simulations. *J. Mol. Biol.*, 231, 75-81.

[Unger, 2004] Unger, R. The Genetic Algorithm Approach to Protein Structure Prediction. *Structure and Bonding.*, 110, 153-175.

[Vajda and Kozakov, 2009] Vajda, S. and Kozakov, D. Convergence and combination of methods in protein-protein docking. *Curr Opin Struct Biol*, 19, 164-170.

[Vassura *et al.*, 2008] Vassura, M., Margara, L., Di Lena, P., Medri, F., Fariselli, P. and Casadio, R. Ft-comar: fault tolerant three-dimensional structure reconstruction from protein contact maps. *Bioinformatics*, 24(10), 1313–1315.

[Vassura *et al.*, 2011] Vassura, M., Di Lena, P., Margara, L., Mirto, M., Aloisio, G., Fariselli, P. and Casadio, R. Blurring contact maps of thousands of proteins: what we can learn by reconstructing 3d structure. *BioData Min*, 4(1), 1.

[Venclovas *et al.*, 1999] Venclovas, C., Zemla, A., Fidelis, K. and Moult, J. Some measures of comparative performance in the three CASPs. *Proteins: Structure, Function, and Genetics*, 34, 220-223.

[Vullo *et al.*, 2006] Vullo, A., Walsh, I. and Pollastri, G. A two-stage approach for improved prediction of residue contact maps *BMC Bioinformatics*, *7 (180)*, 1-12.

[Walsh *et al.*, 2009] Walsh, I., Bau, D., Martin, A., Mooney, C., Vullo, A. and Pollastri, G. Ab initio and template-based prediction of multi-class distance maps by two-dimensional recursive neural networks. *BMC Structural Biology*, 9(1), 5.

[Wang *et al.*, 2010] Wang, Z., Eickholt, J. and Cheng, J. Multicom: a multi-level combination approach to protein structure prediction and its assessments in casp8. *Bioinformatics*, 26(7), 882–888.

[Wang *et al.*, 2011] Wang, Z., Zhao, F., Peng, J., Xu, J. Protein 8-class secondary structure prediction using conditional neural fields. *Proteomics*, 11(19), 3768–3792.

[Ward *et al.*, 2003] Ward, J.J., McGuffin, L.J., Buxton, B.F., Jone, D.T. Secondary structure prediction with support vector machines. *Bioinformatics*, 13, 1650-1655.

[Wei *et al.*, 2011] Wei, Y., Floudas, C.A. Enhanced inter-helical residue contact prediction in transmembrane proteins. *Chem Eng Sci*, 66(19), 4356–4369.

[Wilson *et al.*, 2002] Wilson, C.L., Hubbard, S.J., Doig, A.J. A critical assessment of the secondary structure prediction of alpha-helices and their N-termini in proteins. *Protein Eng.*, 15, p.545-554.

[Wilson *et al.*, 2004] Wilson, C.L., Boardman, P.E., Doig, A.J., Hubbard, S.J. Improved prediction for N-termini of alpha-helices using empirical information. *Proteins*, 57(2), p.322-330.

[Wood *et al.*, 2005] Wood, MJ., Hirst, JD. Protein secondary structure prediction with dihedral angles. *Proteins: Structure, Function, and Bioinformatics*, 59(3), 476–481.

[Wu *et al.*, 2011] Wu, S., Szilagyi, A. and Zhang, Y. Improving protein structure prediction using multiple sequence-based contact predictions. *Structure*, 19(8), 1182–1191.

[Wu and Zhang, 2008] Wu, S. and Zhang, Y. A comprehensive assessment of sequence-based and template-based methods for protein contact prediction. *Bioinformatics*, 24(7), 924–931.

[Xu and Zhang, 2012] Xu, D. and Zhang, Y. Ab initio protein structure assembly using continuous structure fragments and optimized knowledge-based force field. *Proteins*, 80, 1715-1735.

[Xue *et al.*, 2009] Xue, B., Faraggi, E. and Zhou, Y. Predicting residue-residue contact maps by a two-layer, integrated neural-network method. *Proteins*, 76(1), 176–183.

[Yang *et al.*, 2011] Yang, B., Wu, Q., Ying, Z. and Sui, H. Predicting protein secondary structure using a mixed-modal SVM method in a compound pyramid model. *Knowledge-Based Systems*, 24(2), 304–313.

[Yang and Chen, 2011] Yang, J.Y. and Chen, X. A consensus approach to predicting protein contact map via logistic regression. In: J. Chen, J. Wang, A. Zelikovsky *Bioinformatics Research and Applications* - 7th International Symposium, ISBRA 2011, Changsha, China, May 27-29, 2011. Proceedings, *Lecture Notes in Computer Science*, vol. 6674, pp. 136–147. Springer.

[Yi and Lander, 1993] Yi, T.M. and Lander E.S. Protein secondary structure prediction using nearest-neighbor methods. *J.Mol.Biol.*, 232, 1117-1129.

[Zaki *et al.*, 2003] Zaki, M.J., Jin, S. and Bystroff, C. Mining Residue Contacts in Proteins Using Local Structure Predictions *IEEE Transactions on Systems, Man, and Cybernetics. Part B: Cybernetics*, 33(5), 789-801.

[Zemla, 2003] Zemla, A. LGA: a method for finding 3D similarities in protein structures. *Nucl. Acids Res.*, 31:13, 3370-4.

[Zeng *et al.*, 2004] Zeng, S.Y., Kang, L.S. and Ding, L.X. An orthogonal multi-objective evolutionary algorithm for multi-objective optimization problems with constraints. *Evol Comput.*, 12(1), 77-98.

[Zhang *et al.*, 1992] Zhang, X., Mesirov, JP., Waltz DL. Hybrid system for protein secondary structure prediction. *Journal of Molecular Biology*, 225(4), 1049–1063.

[Zhang *et al.*, 1993] Zhang, X., Fetrow, J.S., Waltz, D.L., Rennie, W.A. Automatic Derivation of Substructures Yields Novel Structural Building Blocks in Globular Proteins. *Proceedings of the First International Conference on Intelligent Systems for Molecular Biology*, 438-446.

[Zhang *et al.*, 2005] Zhang, G., Huang, D. and Quan, Z. Combining a binary input encoding scheme with rbfnn for globulin protein inter-residue contact map prediction. *Pattern Recognition Letters*, 16(10), 1543–1553.

[Zhang *et al.*, 2007] Zhang, G., Han. K. Hepatitis C virus contact map prediction based on binary strategy. *Comp. Biol. and Chem.*, 31, 233-238.

[Zhang and Huang, 2004] Zhang, G.Z. and Huang, D.S. Prediction of inter-residue contacts map based on genetic algorithm optimized radial basis function neural network and binary input encoding scheme *Journal of Computer-Aided Molecular Design*, 18 797-810.

[Zhang and Skolnick, 2004] Zhang, Y. and Skolnick, J. Scoring function for automated assessment of protein structure template quality. *Proteins*, 57, 702-710.

[Zhang, 2009] Zhang, Y. I-TASSER: fully automated protein structure prediction in CASP8. *Proteins: Structure, Function, and Bioinformatics*, 77, 100-113.

[Zhao and Karypis, 2002] Zhao, Y. and Karypis, G. Prediction of Contact Maps Using Support Vector Machines *Proceedings of Third IEEE Symposium on Bioinformatics and Bioengineering, BIBE 2003*, 26-33.

[Zhou *et al.*, 2008] Zhou, J., Arndt, D., Wishart, D.S., Lin, G., Shi, Y., Zhou, J., Arndt, D., Wishart, D.S. and Lin, G. Protein contact order prediction from primary sequences *BMC Bioinformatics*, 9 (255), 1-21.

[Zhou *et al.*, 2010] Zhou, T., Shu, N. and Hovmöller, S. A novel method for accurate one-dimensional protein structure prediction based on fragment matching. *Bioinformatics*, 26(4), 470–477.

[Zhou *et al.*, 2011] Zhou, Y., Duan, Y., Yang, Y., Faraggi, E. and Lei, H. Trends in template/fragment-free protein structure prediction. *Theoretical Chemistry Accounts: Theory, Computation, and Modeling (Theoretica Chimica Acta)*, 128, 3–16.

[Ziztler *et al.*, 2001] Zitzler, E., Laumnns, M. and Thiele, L. SPEA2: Improving the Strength Pareto Evolutionary Algorithm. *Technical Report*, 103, Zürich,Switzerland: Computer Engineering and Networks Laboratory (TIK), Swiss Federal Institute of Technology (ETH), May.

[Ziztler and Thiele, 1998] Zitzler, E. and Thiele, L. An evolutionary algorithm for multiobjective optimization: The strength Pareto Approach. *Technical Report*, 43, Zürich,Switzerland: Computer Engineering and Networks Laboratory (TIK), Swiss Federal Institute of Technology (ETH).

[Ziztzler and Thiele, 1999] Zitzler, E. and Thiele, L. Multiobjective evolutionary algorithms: a comparative case study and the strength

pareto approach. *Evolutionary Computation, IEEE Transactions on*, 3(4), 257–271.

[Zvelebil *et al.*, 1987] Zvelebil, MJ., Barton, GJ., Taylor, WR., Sternberg, MJ. Prediction of protein secondary structure and active sites using the alignment of homologous sequences. *Journal of Molecular Biology*, 195(4), 957–961.

[1] http://www.wwpdb.org , Protein Data Bank web.

[2] ftp://ftp.wwpdb.org , Protein Data Bank online repository.

[3] http://www.upo.es/eps/marquez/proteins.txt , Complete list of PDB protein identifiers.