A framework for improving protein structure predictions by teamwork

Luigi Palopoli[†] **Giorgio Terracina**[‡]

[†] D.I.M.E.T. - Università di Reggio Calabria
Via Graziella, Località Feo di Vito
89100 Reggio Calabria, Italy
[‡] Dipartimento di Matematica - Università della Calabria
Via P. Bucci, 87036 Rende, Cosenza, Italy
e-mail:palopoli@ing.unirc.it, terracina@mat.unical.it

Abstract

Predicting the three dimensional structure of proteins is a difficult task. In the last years several approaches have been proposed for performing this task taking into account different protein chemical and physical properties. As a result, a growing number of protein structure prediction tools is becoming available, some of them being specialized to work on either some aspects of the predictions or on some categories of proteins. In this context, it becomes useful to jointly apply different prediction techniques and combine their results in order to improve the quality of the predictions. However, several problems have to be solved in order to make this a viable possibility. In this paper we propose a framework allowing to *(i)* define a common reference applicative domain for different prediction techniques, *(ii)* characterize predictors through evaluating some quality parameters, *(iii)* characterize the performances of a team of predictors jointly applied over a prediction problem and *(iv)* obtain a unique prediction from the team. Finally, we highlight the application of this framework to the definition of a multi-agent system performing the team selection task, the integration of multiple, possibly heterogeneous, predictions and the translation of predictors inputs and outputs into a uniform data format.

Keywords: Protein structure prediction, heterogeneous representations, collaborative computation, multi-agent systems.

1 Introduction

In the last years several approaches have been proposed for predicting the three-dimensional structure of proteins (As an example see (Daron, Gunn, Friesner & McDermott 1998, Dudek, Ramnarayan & Ponder 1998, Galaktionov & Marshall 1994, Krasnogor, Hart, Smith & Pelta 1999, Olszewski & Yan 2000, Piccolboni & Mauri 1997, Rost 1998, Rost & Sander 1993, Rost, Schneider & Sander 1997)). Some other approaches have been proposed for predicting either protein secondary structures, e.g. (Baldi, Brunak, Frasconi, Pollastri & Soda 2000, Casadio, Compiani, Fariselli, Jacoboni, Martelli & Rossi 2001, Cuff & Barton 1999, Rost & Sander 1993), or contact maps (Olmea & Valencia 1997). In order to predict the three-dimensional structure of proteins, several approaches exploit protein chemical or physical properties of proteins, (e.g. (Daron et al. 1998, Dudek et al. 1998) work on energy minimization, whereas (Galaktionov & Marshall 1994) use intraglobular contacts), or evolutionary information (Piccolboni & Mauri 1997).

In general, the final goal of protein structure prediction techniques is that of obtaining a representation of the three dimensional atomic configuration of the protein starting either from its aminoacid sequence or from a set of chemical and physical properties representing it.

Some of the available tools for three dimensional protein structure prediction are described in (Bowers, Strauss & Baker 2000, Fischer & Eisenberg 1996, Gough, Karplus, Hughey & Chothia 2001, Guex & Peitsch 1997, Huber, Russell, Ayers & Torda 1999, Lund, Frimand, Gorodkin, Bohr, Bohr, Hansen & Brunak 1997, Meller & Elber 2001, Shindyalov & Bourne 2000). While the number of proposals and the accuracy of predictions is constantly growing, the existing tools are still not sufficiently accurate and reliable for predicting all kinds of proteins; moreover, each of them operates optimally only for a particular protein category.

In this context, it appears particularly interesting to jointly apply different prediction techniques and to combine their results for improving the quality of the overall prediction. However, several problems should be solved in order to make this a viable possibility. Indeed: *(i)* different prediction techniques often exploit quite different formalisms for representing their inputs and their outputs; *(ii)* approaches used by various prediction techniques for carrying out their activity are quite different; *(iii)* both the comparison and the integration of produced results appear particularly hard. Note that, in general, the application of a large number of prediction techniques might not lead to an improvement in the prediction quality and, in some cases, such a generalized application might be not even possible at all.

In this paper we propose a framework by which a suitable set of prediction techniques can be chosen to be applied together out of a given set of available ones. Thus, this paper proposes the formalization of a framework allowing to *(i)* define a common application domain for different prediction techniques, *(ii)* characterize the performances of single predictors when applied over a prediction problem, *(iii)* characterize the performances of a team of predictors to be jointly applied over a prediction problem and *(iv)* obtain a unique prediction from the team. The formalization of the framework will be obtained through the following steps:

1. the uniform and formal definition of a *generic* predictor to be used as the reference for describing any kind of predictor;
2. the definition of measures allowing to relate single predictions; this will constitute the basis to analyze the performances of predictors;

3. the definition of measures characterizing the performances of a single predictor, thus providing tools for formally dealing with predictors;
4. the characterization of the performances of a team of predictors; this is the core step of the proposed framework;
5. the derivation of a single prediction from the results yielded by predictors in a team; this is obtained by exploiting the results of the previous steps.

In particular, in *Step 1* we give the basic definitions for a formal and uniform description of tools that are generally heterogeneous in both their input and output structures and prediction techniques; moreover, we clearly state what kind of tools can be dealt with using our framework. In *Step 2* we define how two protein structure predictions are compared and related within the framework; this step is the foundation for *Step 3* in which we characterize the behaviour of a prediction tool looking at its *average* behaviour when applied on a specific set PS of proteins for which the structure is known and representing the reference evaluation set underlying one given prediction problem; in more detail, for each protein sequence p in PS the prediction yielded by the tool is related to the known structure of p to obtain some *precision* measures characterizing the behaviour of the tool on p. The set of measures obtained by applying the tool on PS are averaged to obtain overall *precision* measures characterizing the tool when applied on PS. Measures defined in this step are to be exploited to determine the appropriate set of prediction tools to be applied together out of a given set of available ones in a specific prediction problem. Definitions given in *Step 1* are exploited both in this and in the subsequent step for handling different tools independently of their input and output domains. In *Step 4* previously defined precision measures are extended to characterize teams of predictors; definitions provided in this step might be used to both validate the selection of the tools composing the team and to foretell the capability of the team to improve the prediction quality w.r.t. the separate application of single tools. Finally, in *Step 5* all information derived so far is exploited to derive single protein structure predictions from the predictions yielded by the tools forming a given selected team.

The framework proposed in this paper constitutes the theoretical foundation of the multi-agent system *X-MACoP* (XML Multi-Agent System for the collaborative prediction of protein structures) we proposed in (Garro, Terracina & Ursino 2002, Garro, Terracina & Palopoli 2002). *X-MACoP* supports the users in the prediction of the three-dimensional structure of proteins. In particular, the system automatically performs the following tasks: *(i)* selection of a team of predictors to be jointly applied to the prediction problem of interest for the user; *(ii)* integration of the predictions yielded by the predictors of the team for obtaining a unique prediction to be proposed to the user; *(iii)* (possible) translation of the predictor inputs and outputs in such a way that a user handles a unique data format.

It is worth pointing out that this paper does not propose a "meta-predictor", but defines a formal framework which provides the tools for combining various prediction techniques and for properly selecting teams of predictors in order to optimize results quality.

Before closing the present section, we briefly discuss about the system EVA (EValuation of Automatic protein structure prediction) (Eyrich, Marti-Renom, Przybylski, Madhusudhan, Fiser, Pazos, Valencia, Sali & Rost 2001). EVA continuously and automatically analyzes protein structure prediction servers in "real time". Its main goal is to provide a continuous, fully automated and statistically significant analysis of structure prediction servers in order to answer the question "How well could molecular biologists predict protein structures if they simply take the outputs from the available programs?". EVA provides very sophisticated tools for evaluating servers working on *(i)* prediction of protein structure in 1D (secondary structure), *(ii)* prediction of protein structures in 2D (inter-residues distances), *(iii)* prediction of protein structures in 3D (homology modeling and threading) and *(iv)* prediction of novel folds.

While related to our framework, (Eyrich et al. 2001) has a quite different purpose from our own. Indeed, we focus on extracting inter-relationships between prediction techniques, whereas (Eyrich et al. 2001) focuses on the characterization of single tools.

The plan of the paper is as follows. In Section 2 we give a formal definition of a generic predictor. In Section 3 we define some measures relating pairs of protein structures whereas, in Section 4, measures characterizing a prediction tool are given. Section 5 is devoted to the definition of measures relating sets of predictors allowing to define a technique for selecting predictor teams. In Section 6 we propose a technique for obtaining a unique prediction from the set of predictions provided by predictors in a team whereas, in Section 7, we give an example of application of our framework over three real prediction tools, namely the *Swiss-Model* (Guex & Peitsch 1997), the *CPH-models* (Lund et al. 1997) and the *DOE FOLD Server* (Fischer & Eisenberg 1996). In Section 8 we show how the proposed framework has been exploited for the definition of the multi-agent system *X-MACoP*. Finally, in Section 9 we draw our conclusions.

2 Formalizing the predictors

In order to define our framework for comparing different protein structure prediction tools, it is necessary to formally define the model used for representing the generic predictor. A predictor can be modeled as a function F which takes in input a protein sequence p and yields the associated three dimensional structure *tdp*. So, F can be defined as:

$$F : SD \rightarrow TD$$

where SD is the source domain and TD is the target domain of F. The source domain defines the kind of input needed by the predictor, i.e. the kind of representation used to describe the protein sequence to be analyzed. Analogously, the target domain indicates the representation used to describe the predicted protein structure. As already mentioned, in the literature, several protein structure prediction techniques have been proposed (Daron et al. 1998, Dudek et al. 1998, Galaktionov & Marshall 1994, Krasnogor et al. 1999, Olszewski & Yan 2000, Piccolboni & Mauri 1997, Rost 1998, Rost & Sander 1993, Rost et al. 1997) and a great variety of protein properties is exploited to perform the predictions (Daron et al. 1998, Dudek et al. 1998, Galaktionov & Marshall 1994, Piccolboni & Mauri 1997). This variety results in a large number of formalisms used to represent the protein sequences to be analyzed and of representations used to describe protein structures. For instance, a protein can be represented either as a sequence of DNA residues or as a sequence of aminoacids or by chemical properties associated to the aminoacids, e.g. {*Hydrophobic, Neutral, Hydrophilic*}, and so on. Analogously, three-dimensional

protein structures can be represented either by Cartesian coordinates of the backbone atoms or using dihedral angles representing rotations between peptide bonds linking the backbone atoms, and so on. Such a diversity makes it difficult to compare different prediction tools and the results they yield.

In order to be able to uniformly handle different predictors, it is useful to consider a reference domain for the representation of the protein sequences and a reference domain for the expression of their structures. In the following, we shall use SD_0 to indicate our standard reference source domain representing the protein sequence and TD_0 for the target domain representing the folded protein. They are described next.

2.1 The reference source domain SD_0

As far as the representation of the protein sequences is concerned, the most suited choice is that of referring to their aminoacid sequences. Therefore, let $\Sigma_S = \{A, R, N, D, C, E, Q, G, H, I, L, K, M, F, P, S, T, W, Y, V\}$ be the set of symbols representing the aminoacids (i.e., Alanine, Arginine, Asparagine, Aspartic acid, Cysteine, and so on) and let ε be a special symbol indicating the "unknown" value, the reference domain SD_0 is then:

$$SD_0 = \{p \mid p \in (\Sigma_S \cup \{\varepsilon\})^*\}$$

where $(\Sigma_S \cup \{\varepsilon\})^*$ is the set of all strings over $\Sigma_S \cup \{\varepsilon\}$. The occurrences of the ε symbols in p might be relative to portions of protein sequences that are either unknown or not required by the predictor and allow to have some portions of the protein representation undefined. The usefulness of ε symbols will appear clearer in the following.

Each symbol in p can be identified by its position within p and is indicated by $p[i]$. Analogously, $p[i, j]$ indicates the portion of p starting at position i and ending at position j.

2.2 The reference target domain TD_0

The three dimensional structure of a protein can be represented according to different formats. One of the most used representations exploits the *phi* (ϕ) and *psi* (ψ) angles (Hunter 1993). Backbone atoms are linked together by peptide bonds. Rotations around those links can be characterized as having essentially two degrees of rotational freedom, encoded in the ϕ and ψ angles. The conformation of a protein can be, therefore, described as a series of pairs of angles $\langle \phi, \psi \rangle$. In order to obtain this representation, however, it is necessary to have the positions of the chains of atoms N-C^α-C composing the backbone. However, several prediction tools are capable to predict just the positions of the C^α atoms (which, in any case, results in a good quality prediction). For this reason, we choose to represent the three-dimensional structure of a protein as a sequence of three dimensional relative positions of the C^α atoms composing the backbone. In this representation, the position of the i-th C^α atom is expressed w.r.t. the position of the $(i-1)$-th C^α atom of the chain. Formalizing, let $\Sigma_T = \{\langle \delta_x, \delta_y, \delta_z \rangle \mid \delta_x, \delta_y, \delta_z \in \mathcal{R}\}$, TD_0 is defined as:

$$TD_0 = \{tdp \mid tdp \in (\Sigma_T \cup \{\langle \varepsilon, \varepsilon, \varepsilon \rangle\})^*\}$$

Also in this case, $tdp[i]$ is used to indicate the i-th element in tdp, whereas $tdp[i, j]$ represents the portion of tdp starting at position i and ending at position j.

It is interesting to point out that given a protein sequence $p \in SD_0$ and the associated structure tdp as expressed in terms of TD_0, there exists a one-to-one correspondence between elements in p and elements in tdp.

2.3 Definition of valid domain families

In order to be able to compare the inputs and the outputs of the prediction tools into consideration, we restrict our discussion to families of source domains $SDF = \{SD_i\}$ and target domains $TDF = \{TD_i\}$ for which the following properties hold:

Property 1 *Source Domain Validity.* A source domain SD_i is *valid* if it is possible to define a function $\tau^S_{SD_i} : SD_i \to SD_0$ that allows to map elements of SD_i into elements of SD_0. ∎

Property 2 *Target Domain Validity.* A target domain TD_i is considered *valid* if it is possible to define a function $\tau^T_{TD_i} : TD_i \to TD_0$ that allows to map elements of TD_i into elements of TD_0. ∎

Note that, the properties above do not represent actual restrictions on the set of available prediction tools since most of them (Bowers et al. 2000, Fischer & Eisenberg 1996, Gough et al. 2001, Guex & Peitsch 1997, Huber et al. 1999, Lund et al. 1997, Meller & Elber 2001, Shindyalov & Bourne 2000) work on domains that satisfy the properties above. However, those properties are to be satisfied in order to be able to compare different predictors.

Now, given two families SDF and TDF of valid source and target domains resp., we can deal with predictors, without loss of generality, just in terms of the reference domains SD_0 and TD_0. Indeed, consider a protein structure prediction tool $F : SD_F \to TD_F$ working on *valid* domains SD_F and TD_F. Properties 1 and 2 assure that it is possible to define two functions $\tau^S_{SD_F}$ and $\tau^T_{TD_F}$ allowing to obtain values in SD_0 (resp., TD_0) from values in SD_F (resp., TD_F). Therefore, a generic predictor function $F : SD_F \to TD_F$ can be easily rewritten in terms of the reference domains SD_0 and TD_0 by exploiting the functions $\tau^S_{SD_F}$ and $\tau^T_{TD_F}$ as shown in the following diagram:

$$\begin{array}{cccc} & \tau^S_{SD_F} & & \tau^T_{TD_F} \\ F: & \{SD_F\} & \to & \{TD_F\} \\ \downarrow & \downarrow & & \downarrow \\ F': & \{SD_0\} & - - \to & \{TD_0\} \end{array}$$

Thus, w.l.o.g., in the following we will refer to the functions associated to the predictors as applied to the reference domains SD_0 and TD_0, unless otherwise specified.

Example 1 As an example of domain transformation, consider the source domain SD_1 in which protein sequences are represented by $tRNA$ basis sequences. A simple function $\tau^S_{SD_1}$ can be defined that associates each codon in a basis sequence $p \in SD_1$ with an aminoacid in SD_0. For instance, $\tau^S_{SD_1}(\langle GCA \rangle) =$**A***lanine*, $\tau^S_{SD_1}(\langle CGG \rangle) = a$**R***ginine*, $\tau^S_{SD_1}(\langle GAU \rangle) =$*asparagi*$\mathbf{N}e$ and so on. Therefore, the $tRNA$ sequence $p = GCA\ CGG\ GAU \ldots$ is translated into the aminoacid sequence $ARN \ldots$ ∎

3 Measures relating pairs of protein structures

The second step of our framework consists in the definition of measures relating pairs of protein structures. This is the base step allowing to analyze the behaviour of a predictor. In particular, we are interested in measuring the correctness of a protein structure prediction w.r.t. its *real* structure. In this section

we suppose that both the protein structure prediction and the real structure belong to TD_0.

Most of the protein structure prediction tools allow to correctly predict only portions of the protein, but it is usually not known which portions are correctly predicted. Therefore, given a structure prediction $tdp_k \in TDP_0$ associated to a protein sequence p we are interested in relating it with the known structure $tdp_0 \in TDP_0$. In particular, it is interesting to measure *(i)* the cover of tdp_k over tdp_0, i.e. the set of portions of tdp_0 correctly predicted by tdp_k, *(ii)* the local precision of tdp_k, i.e. to measure the longest contiguous correctly predicted protein portion and *(iii)* the global precision measuring the overall extension of correctly predicted protein structure.

Since both tdp_0 and tdp_k represent three dimensional structures, it is usually not correct to compare them "as they are". Indeed both representations may depend on the position of the protein w.r.t. the Cartesian axes. A preprocessing phase is therefore mandatory for obtaining a three dimensional superposition of the two protein structures. This can be done by exploiting the *Combinatorial Extension (CE) Method* (Shindyalov & Bourne 1998) applied on tdp_0 and tdp_k. After this, tdp_0 and tdp_k are analyzed through a pairwise comparison of their elements.

Definition 1 Define the *cover* of tdp_k over tdp_0 as the set

$$\begin{aligned} C(tdp_0, tdp_k) = \{ & tdp_k[i,j] \mid \\ & (\forall l)(i \le l \le j)(\delta(tdp_0, tdp_k, l) \le th_\delta), \\ & \delta(tdp_0, tdp_k, i-1) > th_\delta, \\ & \delta(tdp_0, tdp_k, j+1) > th_\delta, \\ & 1 \le i \le j \le |tdp_0| \} \end{aligned}$$

where δ is a function measuring the distance between two elements $tdp_k[l]$ and $tdp_0[l]$ and th_δ is the maximum acceptable value for δ. δ is defined as:

$$\delta(tdp_0, tdp_k, l) = \delta'(\pi(tdp_0, l), \pi(tdp_k, l))$$

Here $\pi(tdp, l)$ computes the three dimensional position of the l-th element of tdp assuming the first one at position $\langle 0,0,0 \rangle$ in Cartesian coordinates and composing the other $l-1$ relative positions up to $tdp[l]$, whereas δ' is as follows:

$$\begin{aligned} & \delta'(\langle x_1, y_1, z_1 \rangle, \langle x_2, y_2, z_2 \rangle) = \\ & \quad \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2 + (z_1 - z_2)^2} \end{aligned}$$

It is worth pointing out that dealing with exact match in this context is not meaningful.

■

Thus, the cover singles out portions of tdp_k that match with correspondent portions of tdp_0.

In the following definitions of local and global precision, we use the definition of cover $C(\cdot)$ as a function parameter in order to allow our framework to be easily extended to other kinds of cover definitions. Moreover, we shall use a function χ which receives two three-dimensional structures tdp_0 and tdp_k and applies the *CE Method* (Shindyalov & Bourne 1998) to tdp_0 and tdp_k in order to perform their superposition. χ returns the pair $\langle tdp'_0, tdp'_k \rangle$ corresponding to the superposed structures.

Definition 2 Define the *local precision* $P_l(tdp_0, tdp_k, C(\cdot))$ of the prediction tdp_k w.r.t. tdp_0 under the cover C as:

$$P_l(tdp_0, tdp_k, C(\cdot)) = \frac{max_{t_l \in C(\chi(tdp_0, tdp_k))} |t_l|}{|tdp_0|}$$

that is, the ratio between the maximum extension of a correctly predicted portion and the overall length of the protein. ■

Definition 3 Define the *global precision* $P_g(tdp_0, tdp_k, C(\cdot))$ of the prediction tdp_k w.r.t. tdp_0 under the cover definition C, as:

$$P_g(tdp_0, tdp_k, C(\cdot)) = \frac{\sum_{t_l \in C(\chi(tdp_0, tdp_k))} |t_l|}{|tdp_0|}$$

that is, the ratio between the overall extension of correctly predicted portions and the overall length of the protein. ■

As described above, other possible definitions of cover might be exploited. As an example, an alternative definition of C might take into account the presence of α-helices or β-strands in the protein structures (Hunter 1993).

4 Measures characterizing a prediction tool

Measures defined in the previous section relate *one* protein structure prediction with the known structure. However, we are interested in characterizing the behaviour of a prediction tool F over a set of proteins.

In order to characterize a prediction tool, it is necessary to compare protein structure predictions yielded by the tool with known protein structures. This can be done by applying the tool on a set PS of protein sequences whose precise three-dimensional structure is already known. PS and corresponding three-dimensional structures can be retrieved from a database storing both protein sequences and structures, such as the Protein Data Bank (Berman, Westbrook, Feng, Gilliland, Bhat, Weissig, Shindyalov & Bourne 2000). To simplify the presentation we suppose to have at disposal a particular predictor, that we call *exact predictor*, $F_0 : SD_0 \to TD_0$ which, for every protein sequence $p \in PS$ is able to yield the *exact* structure of p; this corresponds to retrieving the structure of p from the Protein Data Bank.

Recall that we refer to functions associated to the predictors as applied to the reference domains SD_0 and TD_0, unless otherwise specified (see Section 2.3).

Since we are interested in characterizing the performances of the predictors when they are applied over a specific prediction problem, the selection of the set PS of protein sequences which the tools are to be evaluated upon is a quite relevant task. Indeed, it is important that PS contains protein sequences "representing" the prediction problem we want to address. As an example, if we want to study proteins belonging to the *cupredoxins* family, it is important that PS contains those protein sequences of that family whose structure is already known.

So, let $PS \subseteq SD_0$ be a set of protein sequences whose structure is known. Then, $TDP_0 = \{tdp \mid tdp = F_0(p), p \in PS\} \subseteq TD_0$ represents the set of known structures of protein sequences of PS. Moreover, $TDP_F = \{tdp \mid tdp = F(p), p \in PS\} \subseteq TD_0$ represents the set of predictions yielded by F for the protein sequences in PS. Recall that for those protein structures or portions thereof that F is not able to predict, the special symbol ε is used.

Definition 4 Define the *cover of a predictor F over PS* the set

$$C_F(PS, C(\cdot)) = \{\langle p, C(\chi(F_0(p), F(p))) \rangle \mid p \in PS\}$$

i.e., the set of covers of each prediction w.r.t. the exact structure. Functions C and χ are as defined in the previous section. ■

Note that we have defined the cover of a predictor as a function of the cover between two single structures. The same idea can be used for defining its local and global precision coefficients. In order to define the precision coefficients associated to a predictor we have to take into account that existing predictors might exploit training sets of proteins for tuning their parameters and, therefore, the predictions of these proteins might be biased by the training process. This is dealt with by introducing, in the formulae below, a weighing function $\gamma(p, F)$.

Definition 5 Define the *local precision* $P_{F_l}(PS, C(\cdot))$ (resp., the *global precision* $P_{F_g}(PS, C(\cdot))$) of F over PS and under the cover C as:

$$P_{F_l}(PS, C(\cdot)) = \frac{\sum_{p \in PS} \gamma(p,F) \times P_l(\chi(F_0(p), F(p)), C(\cdot))}{|PS|}$$
$$P_{F_g}(PS, C(\cdot)) = \frac{\sum_{p \in PS} \gamma(p,F) \times P_g(\chi(F_0(p), F(p)), C(\cdot))}{|PS|}$$

where $\gamma(p, F)$ is a function used to take into account the possible exploitation of training sets and is defined as:

$$\gamma(p, F) = \begin{cases} 0.8 & \text{if } p \text{ belongs to the training set of } F \\ 1 & \text{otherwise} \end{cases}$$

■

Thus, the local (resp., global) precision of F is obtained by averaging the local (resp., global) precisions of its predictions over protein sequences in PS.

As will be clear in the following, the measures defined above play a central role in the selection of the teams of predictors to be jointly applied.

5 Measures relating different prediction tools

We are now able to illustrate the fourth step of our framework for relating different protein structure predictors. As pointed out in the Introduction, one of the main problems to be dealt with in this context is that of comparing the inputs and the outputs of different predictors as expressed with different formalisms. Properties 1 and 2 allow us to translate the values of single predictors domains into the values of the reference domains SD_0 and TD_0 and to express all the functions associated to the predictors in terms of SD_0 and TD_0 as explained in Section 2.3. In this section, as in the previous one, we suppose to have at disposal the exact predictor F_0 which, for every protein sequence p is able to yield the *exact* structure of p. Moreover, the set PS of protein sequences exploited below is supposed to correctly represent the prediction problem to be addressed.

We begin by defining the *affinity coefficient* of a pair of predictors. This can be exploited, along with precision coefficients defined above, to drive the selection of teams of predictors to be applied together over a prediction problem.

Definition 6 The *affinity* of two predictors F_1 and F_2 over a set of protein sequences PS under the cover definition C is defined as:

$$\varphi(F_1, F_2, C(\cdot), PS) = \frac{\sum_{p \in PS} P_g(\chi(F_1(p), F_2(p)), C(\cdot))}{|PS|}$$

where C, P_g and χ are as defined in Section 3. ■

Roughly speaking, the affinity coefficient gives a measure of how much F_1 and F_2 give the same results for the same input protein sequence. This is done by directly comparing the predictions of F_1 with those of F_2, instead of relating them with the protein structures yielded by the exact predictor F_0. This measure is useful to characterize the behaviour of the predictors when working on proteins for which the structure is not known (note, by the way that $\varphi(F_1, F_2, C(\cdot), PS) = \varphi(F_2, F_1, C(\cdot), PS)$).

In the previous sections we have defined the *cover* of a single predictor over a set of protein sequences. When considering several predictors working together as a team, it is meaningful to measure the, so as to say, *cumulative* or *global* cover of the team.

Definition 7 Given a team of predictors $\mathcal{T} = \{F_1, \ldots, F_n\}$ and a protein sequence $p \in PS$, define the *conjunctive cover* of the predictions of $\mathcal{T}$ over a protein sequence p the set

$$\begin{aligned} C_{\mathcal{T}_p,\wedge}(p) = \{ & tdp_0[i,j] \mid tdp_0 = F_0(p), \\ & 1 \le i \le j \le |tdp_0|, (\forall l)(i \le l \le j) \\ & ((\delta(\chi(F_0(p), F_1(p)), l) \le th_\delta) \wedge \\ & (\delta(\chi(F_0(p), F_2(p)), l) \le th_\delta) \wedge \ldots \wedge \\ & (\delta(\chi(F_0(p), F_n(p))) \le th_\delta), \\ & (\exists k)(\delta(\chi(F_0(p), F_k(p)), i-1) > th_\delta, \\ & \delta(\chi(F_0(p), F_k(p)), j+1) > th_\delta)\} \end{aligned}$$

and, analogously, define the *disjunctive cover* of $\mathcal{T}$ over a protein sequence p the set

$$\begin{aligned} C_{\mathcal{T}_p,\vee}(p) = \{ & tdp_0[i,j] \mid tdp_0 = F_0(p), \\ & 1 \le i \le j \le |tdp_0|, (\forall l)(i \le l \le j) \\ & ((\delta(\chi(F_0(p), F_1(p)), l) \le th_\delta) \vee \\ & (\delta(\chi(F_0(p), F_2(p)), l) \le th_\delta) \vee \ldots \vee \\ & (\delta(\chi(F_0(p), F_n(p)), l) \le th_\delta), \\ & (\forall k)(\delta(\chi(F_0(p), F_k(p)), i-1) > th_\delta, \\ & \delta(\chi(F_0(p), F_k(p)), j+1) > th_\delta)\} \end{aligned}$$

Definitions above, referring to a single protein sequence, are easily extended to the set PS of protein sequences as follows:

$$C_{\mathcal{T},\wedge}(PS) = \{\langle p, C_{\mathcal{T}_p,\wedge}(p)\rangle \mid p \in PS\}$$
$$C_{\mathcal{T},\vee}(PS) = \{\langle p, C_{\mathcal{T}_p,\vee}(p)\rangle \mid p \in PS\}$$

■

The definitions above are quite useful to measure the cumulative ability of a team of predictors $\mathcal{T}$ to correctly predict the structure of a protein; in particular, the *conjunctive* definition corresponds to a more "conservative" analysis attitude than the *disjunctive* one.

Using the definitions above, we next define the concepts of local and global precisions of a team $\mathcal{T}$ of predictors when applied over the set PS of protein sequences. Differently from precision definitions provided in the previous section, the following ones allow to characterize the behaviour of a team of predictors as a whole. In particular we can define local (resp., global) conjunctive and disjunctive precision coefficients associated to a team $\mathcal{T}$as follows:

Definition 8 Define the *local conjunctive precision* (resp., *local disjunctive precision*) of a team $\mathcal{T}$ of predictors over PS and under the cover $C_{\mathcal{T},\wedge}$ (resp., $C_{\mathcal{T},\vee}$) as:

$$P_{\mathcal{T}_l,\wedge}(PS, C_{\mathcal{T},\wedge}(\cdot)) = \frac{\sum_{p \in PS} \left(\frac{max_{t_l \in C_{\mathcal{T},\wedge}(p)} |t_l|}{|F_0(p)|} \right)}{|PS|}$$

$$P_{\mathcal{T}_l,\vee}(PS, C_{\mathcal{T},\vee}(\cdot)) = \frac{\sum_{p \in PS} \left(\frac{max_{t_l \in C_{\mathcal{T},\vee}(p)} |t_l|}{|F_0(p)|} \right)}{|PS|}$$

and the *global conjunctive precision* (resp., *global disjunctive precision*) of a team $\mathcal{T}$ of predictors as:

$$P_{\mathcal{T}_g,\wedge}(PS, C_{\mathcal{T},\wedge}(\cdot)) = \frac{\sum_{p \in PS} \left(\frac{\sum_{t_l \in C_{\mathcal{T},\wedge}(p)} |t_l|}{|F_0(p)|} \right)}{|PS|}$$

$$P_{\mathcal{T}_g,\vee}(PS, C_{\mathcal{T},\vee}(\cdot)) = \frac{\sum_{p \in PS} \left(\frac{\sum_{t_l \in C_{\mathcal{T},\vee}(p)} |t_l|}{|F_0(p)|} \right)}{|PS|}$$

■

Those measures, might be used to both validate the selection of the predictors to be included in a team and to foretell the capability of the team to improve the prediction quality w.r.t. the application of the single tools separately.

Note that, teams of predictors having high conjunctive precision coefficients have the characteristic to correctly predict the same portions of protein structures, resulting in a high confidence that such predictions are indeed reliable. This characteristic can be effectively exploited to predict the structure of unknown proteins.

In order to show how measures defined so far might be exploited to drive the selection of the team, consider the following examples.

Example 2 Suppose to have a set of three predictors $\{F_1, F_2, F_3\}$ such that F_1 works on energy minimization, F_2 exploits intraglobular contacts and F_3 exploits homology modeling techniques and a set PS of protein sequences representing the prediction problem to be addressed.

One may be interested in obtaining the team of predictors to be jointly used in order to attain sufficiently high precisions (high predictor affinity) minimizing, at the same time, the number of predictors to be used. This problem can be formalized as a linear programming problem as follows:

$$\begin{cases} min\ |\mathcal{T}| - \sum_{F_i,F_j \in \mathcal{T}} \varphi(F_i, F_j, C(\cdot), PS) \\ P_{\mathcal{T}_g,\vee}(PS) \geq k \\ \mathcal{T} \in \{\{F_1\},\{F_2\},\{F_3\},\{F_1,F_2\},\{F_2,F_3\}, \\ \quad \{F_1,F_2,F_3\}\} \end{cases}$$

Analogously, if one is interested in maximizing the global precision coefficient $P_{\mathcal{T}_g,\vee}$ defined in Section 5 by exploiting at most k predictors, the following linear programming problem can be formalized:

$$\begin{cases} max\ P_{\mathcal{T}_g,\vee}(PS) + \\ \quad \sum_{F_i,F_j \in \mathcal{T}} \varphi(F_i, F_j, C(\cdot), PS) \\ |\mathcal{T}| \leq k \\ \mathcal{T} \in \{\{F_1\},\{F_2\},\{F_3\},\{F_1,F_2\},\{F_2,F_3\}, \\ \quad \{F_1,F_2,F_3\}\} \end{cases}$$

■

Examples above clearly show some of the modeling features provided by our framework. In the next Section we show how the results provided by a team might be exploited for obtaining a unique prediction.

6 Generating a unique prediction from the predictions of a team

The last step of our framework is intended to define a technique for obtaining *one*, and as accurate as possible, prediction from the set of predictions yielded by a team $\mathcal{T}$. Predictors in $\mathcal{T}$ are supposed to work independently. It is important to stress the fact that, in our framework, the selection of the team is as important as the precision of the single predictors. In Section 8 we show how the team selection task can be carried out by exploiting a multi-agent system based on the framework described here.

Given a team of n predictors $\mathcal{T} = \{F_1, \ldots, F_n\}$, an input protein sequence p of length k and the set of protein structure predictions yielded by $\mathcal{T}$ when applied on p, in order to obtain a single prediction three support structures are used:

- A matrix TDP, having n rows and k columns, containing the n predictions. In particular, each row i of TDP is associated to the predictor F_i and contains its prediction. The element $TDP[i, j]$ refers to the j-th element of the prediction of F_i, i.e. $F_i(p)[j]$. We use the notation $TDP[i]$ to indicate the whole row i of TDP, i.e. $F_i(p)$. Predictions stored in TDP are supposed to be superposed as described in Section 3.

- An array P of n elements each storing the global precision coefficient of the corresponding predictor, i.e. $P[i] = P_{F_{i_g}}(PS, C(\cdot))$ (see Section 4 for the definition of $P_{F_{i_g}}(PS, C(\cdot))$).

- A matrix M having n rows and k columns in which each element $M[i, j]$ stores the plausibility of the element $TDP[i, j]$ to be a correct prediction. $M[i, j]$ is obtained as follows:

$$M[i,j] = \alpha \times \sum_{x \in \rho(TDP,i,j)} P[x] + (1-\alpha) \times |\rho(TDP, i, j)|$$

  here α is a weighting coefficient belonging to the real interval $[0, 1]$, used to weight the importance of the precision coefficients w.r.t. the quantity of equal predictions, whereas the function ρ takes in input the matrix TDP, two indices i and j and yields the set of rows $\{x\}$ containing predictions sufficiently similar to $TDP[i, j]$; formally:

$$\rho(TDP, i, j) = \{x \mid TDP[x, j] \neq \varepsilon, \delta(TDP[x], TDP[i], j) \leq th_\delta\}$$

The final prediction $tdp_{\mathcal{T}}$ is obtained by suitably composing the n predictions yielded by the team $\mathcal{T}$ according to the information stored in M. In particular, $tdp_{\mathcal{T}} = tdp_{\mathcal{T}}[1, k]$ where each element $tdp_{\mathcal{T}}[j]$ $(1 \leq j \leq k)$ is obtained as follows:

$$tdp_{\mathcal{T}}[j] = TDP[i, j]\ such\ that\ M[i, j] = max_{1 \leq x \leq n} M[x, j]$$

If more than one maximum exists in column j of M corresponding to different elements in TDP, the one corresponding to the predictor with the highest local precision coefficient (see Section 4) is chosen.

7 A practical example

In this section we propose a practical example of application of our framework. In particular, we analyze the behaviour of three predictors, namely the *Swiss-Model* (Guex & Peitsch 1997) (in the following SM), the *CPHmodels* (Lund et al. 1997) (in the following CM) and the *DOE FOLD Server* (Fischer & Eisenberg 1996) (in the following DF). While SM and CM are homology modeling based tools, DF exploits threading techniques to perform its predictions.

The three tools have been applied on a simple set of three protein sequences $PS = \{$*Azurin*, *Amicyanin*, *Plastocyanin*$\}$, belonging to the family *cupredoxins*,

for which the three dimensional structure is known. The reference three dimensional structures, i.e. the predictions yielded by the exact predictor F_0 in our framework, have been taken from the Protein Data Bank (Berman et al. 2000) and their data bank entries are 1AZU, 1AAN and 1PLC, resp.

First we have computed the affinity coefficients relating SM, CM and DF by applying the predictors on the three input protein sequences and comparing their results for each prediction; we obtained $\varphi(SM, CM, C(\cdot), PS) = 0.79$ which indicates that the results of SM and CM, when applied on protein sequences of PS, are indeed similar. Moreover, we computed $\varphi(SM, DF, C(\cdot), PS) = 0.42$ and $\varphi(CM, DF, C(\cdot), PS) = 0.4$ which tell us that the results of DF, when applied on protein sequences of PS, are not particularly related to neither SM nor CM.

The computation of the local precision of the single predictors on PS yielded $P_{SM_l}(PS, C(\cdot)) = 0.26$, $P_{CM_l}(PS, C(\cdot)) = 0.49$ and $P_{DF_l}(PS, C(\cdot)) = 0.06$ indicating that, on average, CM is capable to correctly predict longer contiguous portions. However, the analysis of the global precision coefficient returned $P_{SM_g}(PS, C(\cdot)) = 0.54$, $P_{CM_g}(PS, C(\cdot)) = 0.56$ and $P_{DF_g}(PS, C(\cdot)) = 0.3$ indicating that, overall, SM and CM have the same performances, whereas DF is less precise, when applied on the protein sequences included in PS.

We have then analyzed the performances of the team $\mathcal{T} = \{SM, CM\}$ obtaining $P_{\mathcal{T}_l,\wedge}(PS, C(\cdot)) = 0.32$, $P_{\mathcal{T}_l,\vee}(PS, C(\cdot)) = 0.61$, $P_{\mathcal{T}_g,\wedge}(PS, C(\cdot)) = 0.65$ and $P_{\mathcal{T}_g,\vee}(PS, C(\cdot)) = 0.71$ representing, respectively, the local conjunctive precision, the local disjunctive precision, the global conjunctive precision and the global disjunctive precision of the team $\mathcal{T}$. For instance, $P_{\mathcal{T}_g,\wedge}(PS, C(\cdot)) = 0.65$ indicates that, on average, the team is capable to *contemporarily* yield correct predictions for 65% of the protein structure.

Finally, in order to test our framework on a protein sequence for which the structure is not known, we have applied the team $\mathcal{T}' = \{SM, CM, DF\}$ to the *Uclacyanin 1* ($UCC1$ for short) from *Arabidopsis thaliana* which is a member of the phytocyanin family cupredoxins (Nersissian, Immoos, Hill, Hart, Williams, Herrmann & Valentine 1998). In particular, first we have applied the three tools to $UCC1$ in order to construct the prediction matrix TDP. The array P, storing the global precision coefficients of the tools is $P = [0.54, 0.56, 0.3]$. For the sake of clarity, in Figure 1 we have depicted, just for the first three aminoacids of $UCC1$, the correspondences between the predictions of the three tools. As an example, the OK relative to SM and DF indicates that SM and DF yield sufficiently similar predictions for the first aminoacid. Note that such matrices are not actually computed, but their entry values are implicitly encoded in the rows x of TDP yielded by the function ρ.

Recall that elements of the matrix M are derived as $M[i,j] = \alpha \times \sum_{x \in \rho(TDP,i,j)} P[x] + (1-\alpha) \times |\rho(TDP, i, j)|$. In our experiments, we have exploited a value of α equal to 0.5. Following the correspondences shown in Figure 1.a we can derive, for the first aminoacid of $UCC1$, the following values:

$$M[0,0] = \alpha \times (P[0] + P[2]) + (1-\alpha) \times 2 = \\ = 0.5 \times 0.84 + 0.5 \times 2 = 1.42$$
$$M[1,0] = \alpha \times P[1] + (1-\alpha) \times 1 = \\ = 0.5 \times 0.56 + 0.5 \times 1 = 0.78$$
$$M[2,0] = \alpha \times (P[0] + P[2]) + (1-\alpha) \times 2 = \\ = 0.5 \times 0.84 + 0.5 \times 2 = 1.42$$

As for the second and the third aminoacid of $UCC1$, following the correspondences depicted in Figures 1.b and 1.c resp., we have:

$$M[0,1] = \alpha \times (P[0] + P[1]) + (1-\alpha) \times 2 = 1.55$$
$$M[1,1] = \alpha \times (P[0] + P[1] + P[2]) + (1-\alpha) \times 3 = 2.2$$
$$M[2,1] = \alpha \times (P[1] + P[2]) + (1-\alpha) \times 2 = 1.43$$
$$M[0,2] = M[1,2] = M[2,2] = \\ = \alpha \times (P[0] + P[1] + P[2]) + 0.5 \times 3 = 2.2$$

Remaining elements of M are obtained analogously. In Figure 2 the first elements of M are shown. Recall that elements of the final prediction $tdp_{\mathcal{T}'}$ are obtained as $tdp_{\mathcal{T}'}[j] = TDP[i,j]$ such that $M[i,j] = max_{1 \leq x \leq n} M[x,j]$ and, if more than one maximum exists in column j of M, the one corresponding to the predictor with the highest local precision coefficient is chosen. In Figure 2 circled elements are those corresponding to the maximum values of M computed in a column-wise fashion. The final prediction $tdp_{\mathcal{T}'}$ is given by $tdp_{\mathcal{T}'} = [TDP[0,0], TDP[1,1], TDP[1,2], \ldots]$.

This example clearly illustrates the advantages of exploiting our framework in the context of collaborative protein structure prediction. Indeed, by analyzing both predictor precision coefficients and the coefficients associated to the team $\mathcal{T}$ it is easy to guess that, according to the results extracted using our test set PS, the exploitation of the DF tool would be not helpful to improve the prediction quality. This intuition is confirmed by the example where, by applying the team $\mathcal{T}'$, containing also DF, on a protein of unknown structure belonging to the same family as those composing PS, only the results yielded by SM and CM are exploited to construct the final team prediction.

8 Application of the framework to multi-agent systems

The framework described in this paper, has been exploited for the definition of a multi-agent system called *X-MACoP* (XML Multi-Agent system for the Collaborative Prediction of protein structures) supporting users in the prediction of the three-dimensional structure of proteins. Due to space constraints we give here only a brief description of the system. The interested reader can find all details in (Garro, Terracina & Ursino 2002, Garro, Terracina & Palopoli 2002).

X-MACoP automatically performs the following tasks: *(i)* selection of a team of predictors to be jointly applied to the prediction problem of interest for the user; *(ii)* integration of the predictions yielded by the predictors of the team for obtaining a unique prediction to be proposed to the user; *(iii)* (possible) translation of the predictor inputs and outputs in such a way that a user handles a unique data format.

X-MACoP consists of two kinds of agents, namely the User Agent and the Predictor Agent.

A generic User Agent UA_i is associated with a user u_i and assists her/him in carrying out prediction tasks (we call prediction task the set of activities necessary for predicting the three-dimensional structure of a protein). In particular, for each prediction task τ_k, associated with the prediction of the three-dimensional structure of a protein p_k,UA_i provides u_i with the following services:

- Support for defining the set PS of proteins to associate with the prediction task τ_k and describing the prediction problem of interest for the user.
- Computation of the precision coefficients of a Predictor P_j w.r.t. the proteins of PS.

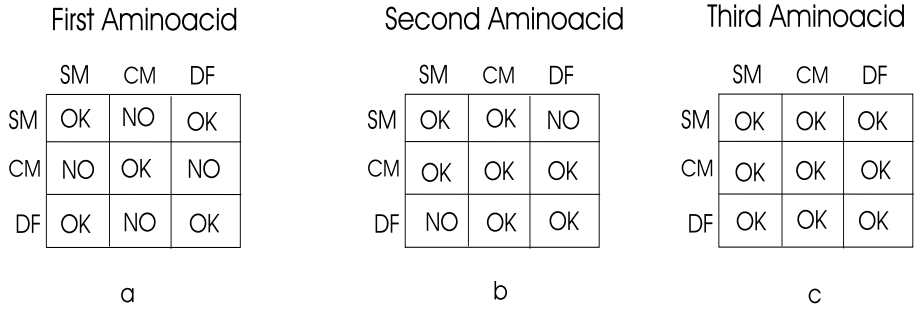

Figure 1: Some correspondences between the predictions of the three tools

1.42	1.55	2.20	....
0.78	2.20	2.20	....
1.42	1.43	2.20	....

Figure 2: The first columns of the derived matrix M

- Construction of a Predictor Team $\mathcal{T}_{ik}$ for UA_i and τ_k; such a team is constructed by considering both the precision of the predictors when applied on PS and some interest parameters set by u_i.
- Prediction of the three-dimensional structure of p_k obtained by integrating the results returned by the Predictor Agents of $\mathcal{T}_{ik}$ when applied on p_k.

A Predictor Agent is associated with a specific prediction tool and collaborates with both the User Agent and other Predictor Agents in the system for both defining predictor teams and carrying out prediction tasks. A Predictor Agent PA_j provides a User Agent UA_i, handling a prediction task τ_k, with the following services:

- prediction of the three-dimensional structure of the proteins composing the set PS which UA_i, or another Predictor Agent PA_j, supplied to it;
- suggestion of other Predictor Agents, unknown to UA_i, which might be considered for the composition of the Predictor Team $\mathcal{T}_{ik}$ associated with UA_i and τ_k;
- prediction of the three-dimensional structure of a protein p_k.

Finally, PA_j has in charge the management of the prediction tool associated with it and the (possible) translation of both input and output data from the format exploited in X-MACoP to the format required by its prediction tool, and vice versa.

When a user u_i needs to predict the three-dimensional structure of a protein, say p_k, the associated User Agent UA_i performs the following tasks:

- Construction of a suitable team $\mathcal{T}_{ik}$ of Predictor Agents;
- Request to the Agents of $\mathcal{T}_{ik}$ to predict the three-dimensional structure of p_k;
- Composition of the results returned by the Predictor Agents of $\mathcal{T}_{ik}$.

In order to construct $\mathcal{T}_{ik}$, UA_i interacts with the Predictor Agents stored in its ontology. In particular, if we consider one of these Predictor Agents, say PA_j, the interaction consists of the following steps:

- UA_i requires PA_j to predict the three-dimensional structure of the proteins of PS;
- PA_j requires its underlying prediction tool to carry out such a task; the results returned by the tool are, then, sent to UA_i;
- UA_i computes the precision degrees of PA_j predictions; if they correspond to user requirements, PA_j is put in $\mathcal{T}_{ik}$ and UA_i requires PA_j to suggest other Predictors. The computation of the precision degrees of a Predictor on a set of proteins is performed as described in Section 4.
- PA_j selects all Predictor Agents stored in its ontology, currently not present in the ontology of UA_i, and requires them to send to UA_i their prediction on the set of proteins of PS.
- After all these predictions have been received, UA_i computes their precision degrees; all Predictors having precision degrees corresponding to user requirements are put in $\mathcal{T}_{ik}$ and in the ontology of UA_i.
- The final team is obtained by computing the precision degrees of $\mathcal{T}_{ik}$ on PS, as described in Section 5, and by adjusting the team in order to satisfy user requirements on the number of predictors to be exploited and the desired precision degrees.
- Finally, UA_i sends to PA_j the list of Predictor Agents of its ontology. Let PA_q be one of them. If PA_q is not present in the ontology of PA_j, PA_q is inserted in the ontology of PA_j. In this way PA_j ontology is enriched with the addition of other Predictor Agents whose behaviour is presumably similar to that of PA_j. Note that this process becomes particularly interesting when many User Agents interact with different Predictor Agents. In order to maintain its ontology at a reasonable size, PA_j removes from it the Predictor Agents not selected in any team for a long period.

After $\mathcal{T}_{ik}$ has been constructed, UA_i asks each Predictor Agent belonging to $\mathcal{T}_{ik}$ to carry out and return its prediction on p_k. Then, UA_i integrates all thus obtained predictions for producing a unique prediction about the three-dimensional structure of p_k. This last task is performed as explained in Section 6.

9 Conclusions

In this paper we have proposed a framework allowing to *(i)* define a common application domain for different prediction tools, *(ii)* characterize single predictors, *(iii)* characterize the performances of a team of predictors to be jointly applied and *(iv)* obtain a unique prediction from the team. We have also shown that the framework provides a formalism allowing to easily handle different predictors.

Moreover we have shown how the proposed framework has been exploited for the definition of the *X-MACoP* multi-agent system supporting users in the task of predicting the three-dimensional structure of proteins.

As far as current and future work is concerned, we are implementing the *X-MACoP* multi-agent system based on the proposed framework. Moreover, we plan to extensively test our system in order to verify its effectiveness.

Acknowledgments. The Authors gratefully thank Bruno Rizzuti for several inspiring discussions about the topics of this paper.

References

Baldi, P., Brunak, S., Frasconi, P., Pollastri, G. & Soda, G. (2000), 'Bidirectional dynamics for protein secondary structure prediction.', *Sequence Learning: Paradigms, Algorithms, and Applications* pp. 99–120.

Berman, H., Westbrook, J., Feng, Z., Gilliland, G., Bhat, T., Weissig, H., Shindyalov, I. & Bourne, P. (2000), 'The protein data bank', *Nucleic Acids Research* pp. 235–242.

Bowers, P., Strauss, C. & Baker, D. (2000), 'De novo protein structure determination using sparse NMR data', *Journal of Biomolecular NMR* **18**, 311–318.

Casadio, R., Compiani, M., Fariselli, P., Jacoboni, I., Martelli, P. & Rossi, I. (2001), 'Tools for protein secondary structure prediction: from sequence to structure', *Protein Sequence Analysis in the Post Genomic Era* pp. 115–133.

Cuff, J. & Barton, G. (1999), 'Evaluation and improvement of multiple sequence methods for protein secondary structure prediction', *Proteins* **34**(4), 508–519.

Daron, M., Gunn, J., Friesner, R. & McDermott, A. (1998), 'Tertiary structure prediction of mixed proteins via energy minimization', *Proteins: Structure, Function, and Genetics* **33**(2), 240–252.

Dudek, M., Ramnarayan, K. & Ponder, J. (1998), 'Protein structure prediction using a combination of sequence homology and global energy minimization', *Journal of Computational Chemistry* **II**(19), 548–573.

Eyrich, V., Marti-Renom, M., Przybylski, D., Madhusudhan, M., Fiser, A., Pazos, F., Valencia, A., Sali, A. & Rost, B. (2001), 'Eva: continuous automatic evaluation of protein structure prediction servers', *Bioinformatics* **17**(12), 1242–1243.

Fischer, D. & Eisenberg, D. (1996), 'Protein fold recognition using sequence-derived predictions', *Protein Science* **5**(5), 947–955.

Galaktionov, S. & Marshall, G. (1994), Properties of intraglobular contacts in proteins: An approach to prediction of tertiary structure., *in* 'Proc. 27th Annual Hawaiian International Conference on Systems Sciences, Biotechnology Computing.', IEEE Computer Society Press, pp. 326–335.

Garro, A., Terracina, G. & Palopoli, L. (2002), Exploiting agents for improving protein structure prediction by teamwork, pp. Submitted for publication, Available from the authors.

Garro, A., Terracina, G. & Ursino, D. (2002), An XML multiagent system for the collaborative prediction of protein structures, *in* 'Proc. of Third International NAISO Symposium on Engineering of Intelligent Systems (EIS 2002)', Universidad de Màlaga, Malaga, Spain, p. Forthcoming.

Gough, J., Karplus, K., Hughey, R. & Chothia, C. (2001), 'Assignment of homology to genome sequences using a library of hidden markov models that represent all proteins of known structure.', *Journal of Molecular Biology* **313**(4), 903–919.

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

Huber, T., Russell, A., Ayers, D. & Torda, A. (1999), 'Sausage: Protein threading with flexible force fields', *Bioinformatics* **15**, 1064–1065.

Hunter, L. (1993), *Artificial Intelligence and Molecular Biology*, AAAI Press, Cambridge.

Krasnogor, N., Hart, W., Smith, J. & Pelta, D. (1999), Protein structure prediction with evolutionary algorithms, *in* 'Proceedings of the Genetic and Evolutionary Computation Conference', Morgan Kaufmann, Orlando, Florida, USA, pp. 1596–1601.

Lund, O., Frimand, K., Gorodkin, J., Bohr, H., Bohr, J., Hansen, J. & Brunak, S. (1997), 'Protein distance constraints predicted by neural networks and probability density functions.', *Protein Engineering* **10**, 1241–1248.

Meller, J. & Elber, R. (2001), 'Linear programming optimization and a double statistical filter for protein threading protocols', *Proteins* **45**(3), 241–261.

Nersissian, A., Immoos, C., Hill, M., Hart, P., Williams, G., Herrmann, R. & Valentine, J. (1998), 'Uclacyanins, stellacyanins, and plantacyanins are distinct subfamilies of phytocyanins: Plant-specific mononuclear blue copper proteins', *Protein Science* **7**, 1915–1929.

Olmea, O. & Valencia, A. (1997), 'Improving contact predictions by the combination of correlated mutations and other sources of sequence information', *Folding & Design* **2**, 525–532.

Olszewski, K. & Yan, L. (2000), 'From fold recognition to homology modeling. an analysis of protein modeling challenges at different levels of prediction complexity', *Computers and Chemistry* **24**, 499–510.

Piccolboni, A. & Mauri, G. (1997), Application of evolutionary algorithms to protein folding prediction, *in* 'Artificial Evolution', pp. 123–136.

Rost, B. (1998), 'Protein structure prediction in 1d, 2d, and 3d', *The Encyclopaedia of Computational Chemistry* **3**, 2242–2255.

Rost, B. & Sander, C. (1993), 'Prediction of protein secondary structure at better than 70% accuracy.', *Journal of Molecular Biology* **232**, 584–599.

Rost, B., Schneider, R. & Sander, C. (1997), 'Protein fold recognition by prediction-based threading', *Journal of Molecular Biology* **270**(3), 471–480.

Shindyalov, I. & Bourne, P. (1998), 'Protein structure alignment by incremental combinatorial extension (CE) of the optimal path', *Protein Engineering* **11**(9), 739–747.

Shindyalov, I. & Bourne, P. (2000), Improving alignments in HM protocol with intermediate sequences., *in* 'Fourth meeting on the critical assessment of techniques for protein structure prediction', pp. A–92.