Secondary structure assignment of proteins in the absence of sequence information

Sammy Khalife, Thérèse E. Malliavin, Leo Liberti

Node classification allows secondary structure assignment of proteins in the absence of sequence information

Sammy Khalife [1], Thérèse E. Malliavin [2], and Leo Liberti [1]

[1]LIX, CNRS, Ecole Polytechnique, Institut Polytechnique de Paris, 91128, Palaiseau, France
[2]CNRS and Institut Pasteur UMR 3528, 75015, Paris, France
khalife@lix.polytechnique.fr
therese.malliavin@pasteur.fr
liberti@lix.polytechnique.fr

Abstract. The structure of proteins is organized in a hierarchy among which the secondary structure elements, α-helix, β-strand and loop, are the basic bricks. The determination of secondary structure elements usually requires the knowledge of the whole structure. Nevertheless, in numerous experimental circumstances, the protein structure is partially known. The detection of secondary structures from these partial structures is hampered by the lack of sequence information.
In this article we present a methodology to assign secondary structures in this context, and introduce the related algorithm called *Sequoia*. Sequoia allows to compute an estimation of secondary structure elements from the values of local distances and angles between the protein atoms, and relies on an message passing neural network seeing the topology of the given protein as a graph. The vertices of this graph are protein residues, and its edges are weighted by values of distances and pseudo-dihedral angles generalizing the backbone angles ϕ and ψ. Any pair of residues, independently of its covalent bonds along the primary sequence of the protein, is tagged with this distance and angle information. Sequoia permits the automatic detection of the secondary structure elements, with an F1-score larger than 80% for most of the cases, when α-helices and β-strands are predicted. In contrast to the approaches classically used in structural biology, such as DSSP, Sequoia is sensitive to the variations of geometry at the limit between two secondary structure elements. Due to its general modeling frame, Sequoia is able to handle graphs containing only Cα atoms, which is particularly useful on low resolution structural input.
Availability and implementation: Sequoia source code can be found at
https://github.com/Khalife/Sequoia with additional documentation.

1 Introduction

Since three decades, the development of structural biology has been driven by the intention to relate the function of molecular objects to the physico-chemical rules at the atomic level. In that frame, tools for the geometric analysis of the protein graph, consisting of atoms and residues, are essential. The protein structure is historically described as a hierarchy of molecular objects: (i) the individual protein residue; (ii) the secondary structure elements (α helices, β strands and loops), which are formed by stretches of residues covalently connected according to the sequence order; (iii) the combination of secondary structure elements, such as the parallel or anti-parallel β sheets formed from associations of β strands through hydrogen bonds; (iv) the tertiary structural motifs [10,2,37], where the association of secondary structure elements is most often stabilized through the formation of a hydrophobic core between residue sidechains; (v) the quaternary structure, where protein domains and/or individual proteins or biomolecules interact to form larger molecular assemblies. The levels (iv) and (v) of the hierarchy define the 3D structure of folded proteins or of assemblies of folded proteins. It should be noted that this hierarchy is strongly based on a description of proteins as polymers, formed of a succession of covalently bonded amino-acids. Moreover, the succession of protein residues along the primary sequence is often used as an input to classical methods for secondary structure prediction [19,13], in particular to detect hydrogen bonds between backbone atoms, and to characterize the α helices and β strands. To the best of our knowledge, all of the current methods for the determination of secondary structure from inter-atomic

distances and angles also use the amino acid sequence assignment. In the present work, we propose to bypass the sequence information.

Our work is motivated by the fact that the sequence of covalently bonded amino-acids is often partially or totally missing. Disordered regions of proteins are not visible in electronic density maps obtained using X-ray crystallography or electronic microscopy. Low resolution structures obtained by X-ray crystallography or electronic microscopy may only display a partial number of protein atoms, such as, for example, the α Carbons. In Nuclear Magnetic Resonance (NMR), the partial assignment of spectra can lead to a similar missing information in sequential assignment.

During the last decade, the explosion of the fields of artificial intelligence and machine learning has driven the consistent development of methods coming from these fields and applied to biology problems. Graph representations combined with deep learning methods or generative models have proved to be relevant for several applications dealing with the complex geometry of protein structures, such as protein-ligand interaction [27] or protein design [18]. In order to harness their experimental performance, we propose a convolutional message passing approach to integrate geometric features of proteins into a convolutional graph neural network, which automatically detects the type of secondary structure elements (α helices, β strands and loops) using the distance and angle information between heavy backbone atoms as its sole input. Specifically, we do not consider any input coming from the existence of covalent bonds between successive residues along the primary sequence. Consequently, the approach can be applied to structures that are determined only partially. We also point out that this is a methodological rather than biological study. Consequently, we aim at showing that our proposed methodology works well in general, meaning we do not fine-tune it for specific proteins.

The approach proposed here, named *Sequoia*, is computationally tested on protein structures determined using X-ray crystallography or NMR. We evaluate the effect of noise level in the input data, as well as the prediction efficiency of Sequoia for various secondary structure elements and protein graphs. On all atom protein structures, Sequoia predicts α helices and β strands with F1 scores respectively mostly better than 95% and 90% and the joint prediction of α helices and β strands displays a F1 score mostly larger than 80%. One should notice that this comparison is calculated with respect to the results with DSSP [19,13]. Sequoia also displays robustness with respect to noisy inputs and missing residues in the graph, as well as for sparse C_α graphs. Interestingly, most of our prediction errors is observed for residues located at the extremities of secondary structure elements, which undergo continuous geometrical transformations, rather than in the discontinuous description from [19,13].

The rest of this paper is organized as follows. Section 2 presents the protein descriptors, their robustness to noisy measurements, and the Sequoia architecture, along with a simple but solid baseline named FOS. Section 3 describes the results. Discussions and conclusions are given in Section 4.

2 Methodology

2.1 Graph description

We consider a natural geometric representation of molecules with n atoms in terms of an $n \times 3$ *realization matrix* where the i-th row is a vector in $\mathbb{R}^3$ corresponding to the Euclidean position of the i-th atom of the molecule, for $i \leq n$. This representation corresponds to the steady state of the protein, enforcing a *molecular rigidity assumption* [29]. For the Sequoia prediction purposes, we represent such structure by means of a simple, undirected, edge-weighted graph $G = (V, E, d)$, where V is the set of atoms, and E is the set of atom pairs $\{i, j\}$ with known distance value d_{ij}. A graph is a very relevant model for describing protein structure and has been widely used [17,31,21,7,12,15].

Two different networks will be considered: one full network with all heavy backbone atoms and one simplified network containing only the Carbons α. In the full network, the heavy atoms are grouped into to subsets corresponding to protein residues, in a way similar to the definition of spin systems in NMR [25].

The graph of residues will be defined by two methods:

- $\mathcal{A}$: a k-nearest neighbors graph $G_k = (V, A)$, where V is the set of all residues in the protein and $(r_1, r_2) \in A$ if and only if r_2 is one of the k nearest neighbors of r_1;

– $\mathcal{B}$: a threshold based graph $G'_\tau = (V, E)$, where V is the set of all residues in the protein and $(r_1, r_2) \in E$ if and only if the measured distance between r_1 and r_2 is lower than the threshold τ.

Both of these constructions require the notion of distance between two residues. For the remaining of this article, we define the distance between two residues as *the minimum distance between the respective atoms composing them*.

A priori, the method $\mathcal{B}$ appears more "natural" than $\mathcal{A}$ since: (a) it is formally undirected; and (b) the threshold τ can be set to a value corresponding to the physical requirements of structural biology. Our experiments (data not shown) revealed that both methods lead to very similar results when $k = 2$ in $\mathcal{A}$ and $\tau = 3$Å in $\mathcal{B}$. We thus decided to use method $\mathcal{A}$ to build the graph of the protein structure.

In addition to distance information, angle information between heavy backbone atoms will be added to the edges of the protein graph. The selected angles will be a generalization of the backbone dihedral angles ϕ and ψ, described below. This generalization will permit the computation of these angles for any pairs of protein residues, covalently bonded or not. In that way, no sequence information is present in the protein graph input of the neural network.

2.2 Protein descriptors and neural network inputs

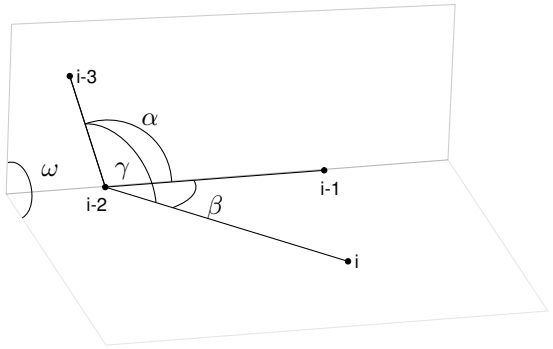

Fig. 1. Two planes made by a group of four atoms $i-3$, $i-2$, $i-1$ and i [22]. ω represents the dihedral angle between the two planes. The angles α, β and γ are calculated from the distances between atoms using Eq 3.

The backbone dihedral angles ϕ and ψ are classically defined between the atoms belonging to successive residues $r-1$, r and $r+1$ in the protein primary sequence:

– the carbon atom of the carbonyl group from residue $r-1$, the nitrogen atom, the carbon-α atom, and the carbon atom of the carbonyl group from residue r
– the nitrogen atom, the carbon-α atom and the carbon atom of the carbonyl group from residue r, and the nitrogen atom from residue $r+1$.

In the present work, this definition will be generalized to any couple of residues being closer in the space than the threshold τ.

Using the atomic coordinates determining the protein structures, it is straightforward to determine the dihedral angles. However, in the case when only the distances between atoms are known, it can be shown that using poly-spherical coordinates [34], or alternatively a Clifford algebraic formulation [22], the cosine of the dihedral angles $\cos\omega$ can be computed using only distances between atoms.

If ω represents the dihedral angle between the plane defined by the three atoms $i-3, i-2, i-1$ and $i-2, i-1, i$, the cosine law for trihedron (Figure 1 and [22]) can be written in the following way:

$$\cos\gamma = \cos\alpha\cos\beta + \sin\alpha\sin\beta\cos\omega \tag{1}$$

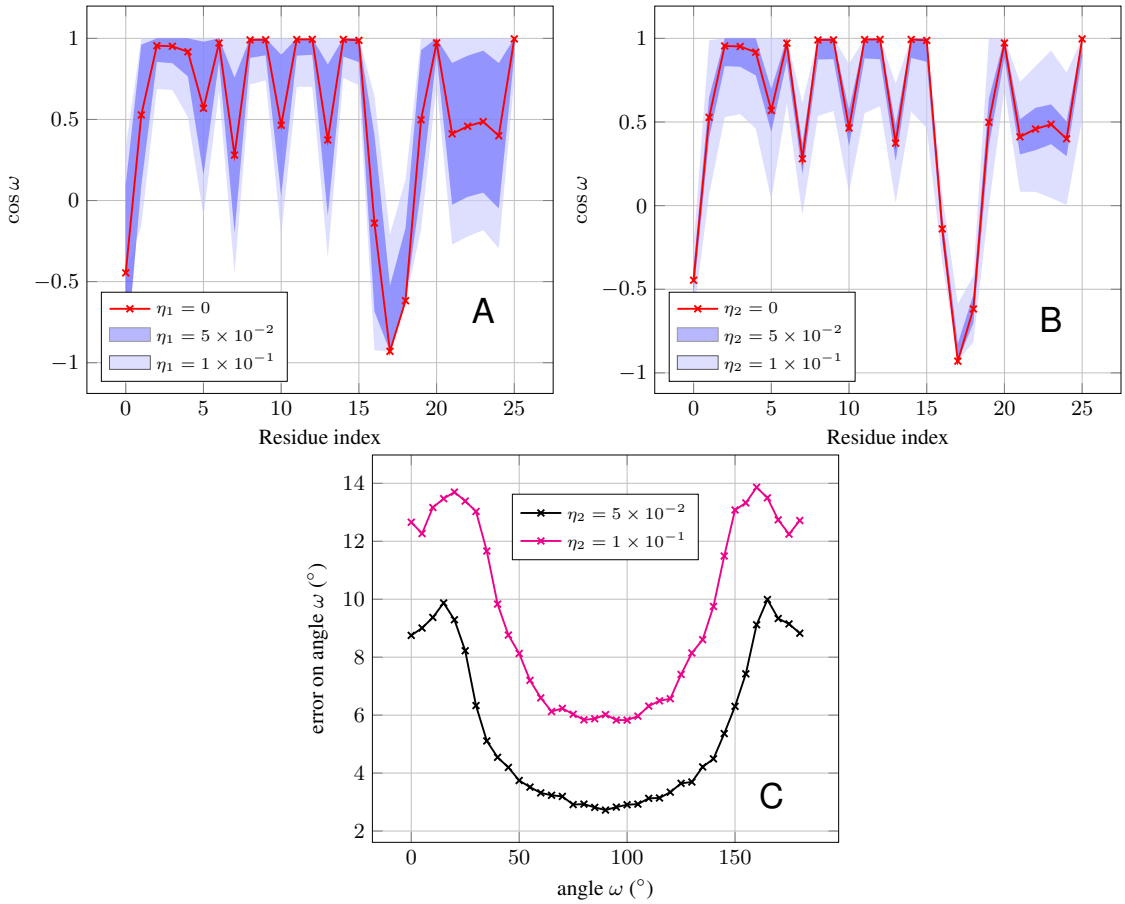

Fig. 2. A, B: Impact of noise on the cosine of dihedral angle ω on the 25 first residues of the protein 1M22 of PDB. The noise level is defined by the parameters: (A) η_1 (Å) on the distances and (B) η_2 on $\cos\omega$. The red curves defines the true $\cos\omega$ values, and the darkcyan and palecyan surfaces define the variations of $\cos\omega$ according to the noise level. C: Variation of the error on the angle ω with respect to the value of ω. The average error is calculated by Monte Carlo using 1000 realisation of Gaussian noise. The error on ω is computed as the standard deviation of $\omega - \arccos(\cos(\omega) + \varepsilon)$, where $\varepsilon = \mathcal{N}(0, \eta_2)$

which yields using relation between $\cos$ and $\sin$:

$$\cos\omega = \frac{\cos\gamma - \cos\alpha\cos\beta}{\sqrt{1-\cos^2\alpha}\sqrt{1-\cos^2\beta}} \tag{2}$$

Furthermore, using the planar cosine law, $\cos\alpha$, $\cos\beta$ and $\cos\gamma$ are given by:

$$\cos\begin{bmatrix}\alpha\\ \beta\\ \gamma\end{bmatrix} = \begin{bmatrix} f(d_{i-1,i-2}, d_{i-2,i-3}, d_{i-3,i-1}) \\ f(d_{i-1,i-2}, d_{i-2,i}, d_{i-1,i}) \\ f(d_{i-3,i-2}, d_{i-2,i}, d_{i-3,i}) \end{bmatrix} \tag{3}$$

where $d_{i,j}$ is the distance between atom i and j, and:

$$f(x,y,z) = \frac{-z^2+x^2+y^2}{2xy} \tag{4}$$

Using Equations 3, and 4, Eq. (2) can be reformulated as:

$$\cos\omega = \frac{2d^2_{i-2,i-1}\Delta_i - (d_{i-3,i-2,i-1})(d_{i-2,i-1,i})}{\Gamma_i\sqrt{4d^2_{i-2,i-1}d^2_{i-2,i-3} - (d^2_{i-2,i-1,i})}} \tag{5}$$

with:

$$\begin{aligned} d_{i-3,i-2,i-1} &= d^2_{i-3,i-2} + d^2_{i-2,i-1} - d^2_{i-3,i-1} \\ d_{i-2,i-1,i} &= d^2_{i-2,i-1} + d^2_{i-2,i} - d^2_{i-1,i} \\ \Delta_i &= d^2_{i-3,i-2} + d^2_{i-2,i} - d^2_{i-3,i} \\ \Gamma_i &= \sqrt{4d^2_{i-1,i-2}d^2_{i-2,i} - (d^2_{i-3,i-2,i-1})} \end{aligned}$$

Eq. 5 allows the calculation of backbone angles ϕ and ψ depending on the set of considered atoms $i-3, i-2, i-1$ and $i-2, i-1, i$, as recalled at the beginning of the subsection. Thus, using this Equation, we generalize the notion of ϕ and ψ angles to any pair of residues k and l in the protein, by considering the relevant atoms in the residues. Then if residue k and l are connected in the graph, the edge features x_{kl} are defined as $x_{kl} = (d_{kl}, \cos\phi_{kl}, \cos\psi_{kl})$ from the distance d_{kl} between the two residues, and the cosines of the pseudo-dihedral angles ϕ_{kl} and ψ_{kl}.

In addition to a graph containing all backbone atoms, we also tested the prediction of secondary structure on a simplified graph containing only C_α atoms. In that case, the edge between C_α atoms of residues a and b is labeled by $x_{ab} = (d_{ab}, \cos\Phi_{ab})$, d_{ab} being the distance between C_α atoms and one pseudo dihedral angle Φ_{ab} being defined using the equation 5 where atoms i-1 and i-2 are the atoms C_α of residues a and b, the atoms i and i-3 being two different atoms C_α the closest respectively of atoms i-1 and i-2.

2.3 Testing the noise robustness of dihedral angle computation

In practice, imprecision on distance measurements may lead to greater errors in the dihedral angle estimates. Indeed, the imprecision will lead to numerical errors on $\cos\omega$ as the cosine law for trihedron (Eq. 1) is no more valid. The relationship between the inter-atomic distances and the dihedral angle ω (Eq. 5) can be reformulated as a functional relationship: $\cos\omega = g(D)$ where D is the matrix containing all distances between the atoms i, $i-1$, $i-2$, $i-3$. An estimation of noisy dihedral angles can be obtained with the following equation:

$$(\cos\omega)_\epsilon = g(\mathsf{proj}(D+\epsilon))$$

where ϵ is a (4,4) symmetric matrix verifying: $\forall i, \epsilon_{i,i} = 0$, $\forall i < j, \epsilon_{i,j} \sim \mathcal{N}(0, \eta_1)$, and proj is the projection operator onto the cone of symmetric positive semidefinite (PSD) matrices. The proj operator avoids to consider

Algorithm 1 proj operator onto the cone of symmetric positive semidefinite (PSD) matrices

1: **Input** D: Symmetric matrix
2: **Output** D_{proj}: Projected matrix onto the cone of symmetric PSD matrices
3: $G = -\frac{1}{2}JDJ$, where $J = I_n - \frac{1}{n}\mathbf{1}\mathbf{1}^\top$ (n: number of atoms) , with $G = PDP^{-1}$, P real matrix such that $PP^{-1} = I_n$
4: D_+ diagonal matrix such that $D_+(i,i) = \max(0, D_{i,i})$
5: $E = PD_+P^{-1}$
6: $D_{\mathsf{proj}} = \mathbf{1}\mathsf{diag}(E)^T - 2E + \mathsf{diag}(E)\mathbf{1}^\top$
7: Return D_{proj}

matrices representing non-Euclidean 3D objects, in which case, the denominator of the right hand side in Equation 5 could be zero.

To transform the matrix D to a matrix corresponding to a Euclidean 3D molecular object, the proj operator takes as input a symmetric matrix D, and returns its projection D_{proj} onto the cone of symmetric positive semidefinite (PSD) matrices. This projection is obtained using the procedure described in Algorithm 1 (see [8] for details about this transformation).

In order to estimate the impact of noise addition, we conducted experiments on the ϕ angles of the first 25 residues of a protein (1M22) extracted from Dataset A (presented in Sect. 2.7). Results obtained for a thousand Monte-Carlo simulations are depicted in Figure 2A for noise levels $\eta_1 \in \{0.05, 0.1\}$ Å. We conclude that noisy distances will impact significantly dihedral angles when the imprecision is greater than 0.05Å.

Eq. (2), formulated as $\cos\omega = h(\alpha, \beta, \gamma)$, shows that the dihedral angle ω can be computed only based on angles α, β, γ. If these angles were to be computed with another method than distances, the impact on the dihedral angles might be reduced. In order to evaluate the robustness of our features to the imprecision on angles α, β and γ (Figure 1), we conducted a similar experimental analysis:

$$\cos\omega = h((\alpha, \beta, \gamma) + \epsilon)$$

where $\epsilon \in \mathbb{R}^3 \sim \mathcal{N}(\mathbf{0}, \eta_2 \times \mathbf{1})$, with $\mathcal{N}$ being the normal distribution, $\mathbf{0} = (0,0,0)$, $\mathbf{1} = (1,1,1)$, and η_2 being the relative amplitude of the noise on the cosines. Similarly to the evaluation of noise effect on distances, we considered a thousand Monte-Carlo simulations. The results are depicted in Figure 2B for noise levels $\eta_2 \in \{0.05, 0.1\}$. They show that adding noise to angles α, β and γ has less impact to dihedral angles ω than adding noise to the distances between atoms i, $i-1$, $i-2$, $i-3$. Following the results of these numerical experiments, the robustness to noise of Sequoia will be tested in the following by adding noise to $\cos\omega$. The error induced on ω by adding noise on $\cos\omega$ was also estimated using Monte Carlo simulations (Figure 2C). Depending on the regions of ω values and on the noise level η_2, the error was comprised between 3 and 14°.

2.4 Simple baseline with first order statistics

A first order statistics (FOS), considered as the baseline for the prediction of secondary structure, was defined for comparison purposes with Sequoia. For a fair comparison (Section 2.5), the baseline will also be sequence agnostic. FOS considers the neighborhood of a residue in the graph and compute the average and variances of the cosine of the dihedral angles ϕ and ψ in this neighborhood. The average and variances are then used as features for supervised classification as further explained in Section 2.6.

The idea of this baseline is based on the following remark. Along each β strand element, the protein backbone extends locally in a straight direction whereas along α helices, the backbone displays locally a spiral. These very different local geometries should have an impact on the moving average of cosine of dihedral angles ϕ and ψ, which leads to the FOS definition.

2.5 Sequoia: a Message Passing Neural (MPN) network

One of the advantages of modeling the protein as a graph of residues is to harness the experimental performance of graph neural networks (GNNs). For the sake of generality, we adopt the formulation of message passing

[14] which describes the core idea of GNNs. In the following, the variable t represents a time increment of the parameters of the model, and h_v^t the hidden variable state of node v at time t. The initial hidden states of the model h_v^0 are set to the features considered, which in the frame of this article are the cosines of the pseudo dihedral angles between residues. During the message passing phase, the hidden states h_v^t of each each node in the graph are updated based on messages m_v^t according to

$$m_v^{t+1} = \sum_{w \in N(v)} M_t(h_v^t, h_w^t, e_v^w) h_v^{t+1} = U_t(h_v^t, m_v^{t+1}) \tag{6}$$

where M_t is a message function and U_t a vertex update function. After T iterations, the final output of the node is computed with a readout function R:

$$y_v = R(h_{v \in G}^T)$$

The choice of the family of M_t and U_t and R lead to the design of the graph neural network, as explored in several references for various applications (e.g. Convolutional Network [9], Gated Graph Neural Network [24] or Molecular Graph Convolutions [20]). The learning of the parameters is then performed using standard back-propagation, interpreting the parameter t as the index of the neural network layer. The choice of functions M_t, U_t and R for our experiments is described in Section 2.7.

2.6 Secondary structure prediction with node classification

Based on our formulation, the attribution of a secondary structure to a residue can naturally be formulated as a node classification problem. If y represents the label variable, then we consider three situations:

- α-None: attribution to an α helix element. $y \in \{0, 1\}$.
- β-None: attribution to a β strand element. $y \in \{0, 1\}$.
- $\alpha - \beta-$None: attribution to an α helix, to a β-strand, or to other. $y \in \{0, 1, 2\}$
- All: attribution to all secondary structure elements defined in DSSP [19], leading to 8 classes: $y \in \{0, ..., 7\}$.

On the one hand, the FOS method translates into a simple classification problem that we approach with standard supervised learning methods. On the other hand, the MPN method leads to the training of a message passing neural network. The details of the classifier used for FOS and MPN architecture are described below.

2.7 Practical implementation

FOS As detailed in Section 2.4, the first order statistics (FOS) formulation leads to a simple classification problem with features belonging to $\mathbb{R}^3$. We used a k-nearest neighbors as the classifier for our baseline.

MPN The design of our message passing neural network (MPN) is based on the continuous kernel-based convolutional operator from [14], also known as the edge-conditioned convolution from [38]. Our implementation is based on the two high-level APIs pytorch [32] and pytorch-geometric [11].

We used a two layer kernel-based convolutional, where two message passing schemes are performed sequentially on the hidden states. In our case, for each of the two layers, the message function M_t and the vertex update function U_t are defined as:

$$M_t(h_v^t, h_w^t, e_v^w) = h_w^t \,.\mathcal{N}(e_v^w)$$

$$U_t(h_v^t, m_v^{t+1}) = \Theta \,.\, h_v^t + m_v^{t+1}$$

where $\mathcal{N}$ is a 4 layer linear perceptron with Rectifier Linear Unit (ReLU) activations between each layer and Θ is a linear operator. Finally, the readout function R is a softmax function composed with a two-layers linear perceptron to output after the two main layers a predicted label y_v for each v.

Our initial formulation leads to one dimensional discrete node feature corresponding to the type of amino acid residue for the node and the edge features defined above as x_{kl} and x_{ab} and containing distances and cosines of pseudo-dihedral angles. However, we noticed a gain in performance by aggregating edge features in the neighborhood of a node into its features. This behavior is somehow similar to the experiments led in [14] where edge features constructed from the node features were added to the graph. In our case the transformation goes from edges to nodes. We conjecture it to be a consequence of data augmentation [5].

Datasets of protein structures

Dataset A: Dataset A is composed of 3621 protein X-ray crystallographic structures downloaded from the server PISCES [39]. These structures correspond to a set of Protein Data Bank (PDB) [3] entries for which structures have been determined at a resolution better than 1.6 Å, and with R factors better than 0.25. The set of PDB entries and protein chains present in dataset A has been chosen [39] in order that the percentage of sequence identity between any pair of chains is smaller than 20%, to avoid statistical bias on the protein sequences.

Dataset B: Dataset B is composed of 226 protein structures obtained by processing the database of NMR chemical shifts used for the training of the neural network TALOS-N [35]. For 226 proteins of this database, a structure was determined by NMR. We decided to pick up the first conformer of these NMR structures to build a NMR structure database. The list of proteins and chains used in Datasets A and B are available in the Supplementary Material.

Validation of Sequoia results The secondary structures predictions obtained using Sequoia were compared to the output of DSSP [19], a classical software for the determination of secondary structures. Training samples corresponds to 50% of the samples in Dataset A and the validation samples to other 25% of Dataset A. Finally Test A correspond the 25% remaining samples of Dataset A, and Test B corresponds to the whole Dataset B.

2.8 Use of Sequoia on information coming from EM maps

Predictions were also realized in the context of low resolution structural information, by analyzing positions of atoms C_α predicted from electronic microscopy (EM) maps. To do so, we used the output of a deep-learning approach, Deeptracer [36,33], which predict positions of protein atoms from the image recorded from EM single particle analysis. Several entries from the EMDB, which will be described below, were used as inputs for the Web server of Deeptracer[1] , and the early output containing only atoms C_α was used to feed Sequoia.

3 Results

3.1 Prediction of secondary structure elements

Several experiments have been conducted to investigate the efficiency of Sequoia. First, the Sequoia results have been compared to the FOS baseline in order to estimate the performance improvement brought by a cutting-edge machine learning approach (Figure 3). Second, the robustness of Sequoia has been investigated (Figure 4) by performing runs on: (i) graphs with noisy edge attributes ; (ii) graphs with variable sparsity for vertices. The rational for exploring these aspects is the presence of noise in all experimental techniques of structural biology. The reason for analyzing the aspect (ii) is rather the numerous protein structures for which regions are not visible due to various experimental problems described in the introduction.

The results obtained for secondary structure assignment are reported in Figure 3 for the noise-free tests, and for the noise level $\eta_2 \in \{0.05, 0.1\}$. The first order statistics (FOS) method, introduced in Section 2.4,

[1] https://deeptracer.uw.edu/home

provides a solid baseline with prediction success rates (dashed lines) larger than 50% for graphs with all backbone atoms, in three cases: α-Other, β-Other, α-β-Other. In the case of noise-free test the best F1-scores are obtained using $k = 20$ in the nearest neighbors classifier, whereas for the test in presence of noise, $k = 60$ is required in the classifier to obtain the best F1-scores.

Interestingly, the message passing neural network Sequoia (Figure 3, continuous lines) provides improvement with respect to FOS by a wide margin (5% to more than 10%). The best improvement is obtained for the classification α-β-Other (cyan curves). Furthermore, the improvement increases with the addition of noise, which proves Sequoia is more robust.

The best prediction results are obtained in all cases for the classification α - Other (Figure 3: green continuous curves). This is certainly due to the very narrow interval of dihedral angles corresponding to the definition of the α-helix, which makes the angles values more discriminant. Adding the β-strand (black and cyan curves) induces a decrease of success rate as the dihedral angles defining the β-strand sample larger value intervals. Finally predicting a full classification requires to take into account the whole set of dihedral values measured in the proteins, which sample much larger intervals and display large heterogeneity in regions outside of α-helices and β-strands. Consequently, the results obtained for predicting the eight types of secondary structure described in DSSP (magenta curves) are, in all cases, behind from other predictions by 10 to 20%. This behavior is expected as the power of a classification approach heavily depends on the number of predicted classes.

The statistical approaches FOS and Sequoia display different behaviors on data-sets A and B displayed respectively on right and left panels of Figure 3. For the classification α-Other (green curves), the success rates are better for NMR (B) than for X-ray (A) data-sets. For other classifications, the results are inverted as the success rates are better for X-ray (A) than for NMR (B) structures, specially in the presence of noise η_2. The difference even goes up to 6% for classification All. The smaller success observed in the case of NMR solution structures is not surprising as the larger flexibility in solution which reduces the precision of these structures and consequently hampers the learning procedure. In addition, as described in Section 2.7, the Dataset B is only used for testing purpose and not for training.

In the case when only C_α atoms were included in the calculation, the prediction of secondary structures (Figure 3, lower panels) displays features similar to those observed when all atoms were included in the graph. In case where no noise was added to the angle/distance information, the F1-scores were the most decreased, but the decrease was bigger for FOS than for Sequoia. Overall, the prediction of α helix alone (green curves) keeps quite similar scores than in the case of all backbone heavy atoms were considered. There is a marked decrease of the success as soon as more than one type of secondary structure is considered. The C_α networks seems thus to have less discriminating features between different secondary structures than the all network of heavy backbone atoms. When the Datasets A (left panels) and B (right panels) are compared, the improvement for α-Other (green curves) in the Dataset B is similar that the one observed for all atoms. For the classification All, the proteins of Dataset B display significantly worse results than those of Dataset A.

3.2 Effect of degraded input

The effect of degraded input was investigated by reducing randomly the number of residues in the graph (left panels) or by increasing the number of connected neighbors described by the hyperparameter k (right panels) introduced in Section 2.1. In the graph including all backbone heavy atoms, several percentages of residues ablation from the graph network were considered (Figure 4, upper left panel). It is remarkable that the prediction by Sequoia is reduced from less than 10% for all ablation levels smaller than 20%. For larger ablation levels, the success rate decreases strongly but, for the prediction of α or β elements, is mostly reduced of about 20% for an ablation level of 50%. The two Datasets A and B (continuous and dashed lines) displays similar resistance to ablation for all predictions.

The influence of the hyperparameter k, defining for each residue, the number of neighbor residues connected by an edge in the graph, was also investigated (Figure 4, upper right panel). Hyperparameter values k of 3, 4, 5 have been explored whereas a value of $k = 2$ was used in the previous analyses (Figure 3). The predictions are more robust to the increase of k than they were to the ablation of residues. Sequoia displays

improved success rates along the number of neighbors for all types of investigated predictions. As the neighbor residues are added to the graph basing only on a distance criterion, they are shared between residues close in the primary sequence and other far apart in the primary sequence. The increase of success rates observed when adding more neighbor residues, gives an insight that the generalized definition of ϕ and ψ proposed in the present work, is quite efficient to decipher between residues close and far apart in the primary sequence. Indeed, the detection of secondary structure elements favor the residues close in the primary sequence to the detriment of the residues far apart in the primary sequence.

The effect of degraded input was also tested on the simplified network containing only atoms C_α (Figure 4, bottom panels). Concerning the random ablation of residues, the results on C_α graphs are quite similar (Figure 4, bottom left panel) to the results obtained on the backbone atom graph, with an overall reduction of scores of about 5% for ablation percentages up to 20%. For ablation percentages larger than 20%, the α-Other (green curves) prediction is much more affected than the predictions β-Other (black curves) and α-β-Other (cyan curves) predictions which display relatively flat variations according to the increase of ablation. This might be related to the difference of geometry between an α helix and a straight line corresponding to a β region. Indeed, in a helix defined by points, the removal of points has a larger influence on the perception of the geometric figure than in a straight line. The influence of the hyperparameter k was also investigated (Figure 4, bottom right panel) for the graph containing only atoms C_α. The observed trends were similar to those for the graph including all backbone heavy atoms. Nevertheless, the increase of F1 score is less marked and corresponds rather to a plateau of values.

3.3 Positions of Sequoia erroneous predictions

The error cases in Sequoia prediction were examined for Dataset A in the absence of noise (Figure 5, upper panel). For each erroneously predicted residue, the distance of the residue to the extremity of the corresponding secondary structure element was determined. For classifications α-Other and β-Other, a large majority of the erroneous predictions ($w = 2$) were located at the limits within the two first or the two last residues of a secondary structure element, most of them being the first or the last residue ($w = 1$). These erroneous predictions are the sign of different points of view on the limits of secondary structure elements. DSSP handles a discrete classification, whereas Sequoia is sensitive to the geometrical deformations close to the limits which leads to to exclude the limit residues from the detection of the element. If one would exclude the limit residues from the initial definition of the secondary structure element, the success rates in Figure 3 would drop for Sequoia from 94.4 up to about 97% for the prediction α-Other.

The positions of the residues erroneously assigned to secondary structure elements in a C_α graph (Figure 5, bottom panel), displays a quite striking difference from the predictions realized in the graph including backbone heavy atoms. Indeed, the erroneous β-Other predictions are in majority located at the extremity of the β strands, but in a lesser extend that for the graph built from backbone heavy atoms (Figure 5). At the contrary, the erroneous α-Other predictions are more often located at the extremity of the α helices than in the all-atom graph. This difference of behavior between the graph of Carbons α and the all atom graph is related to the differences in the geometry of an helix and a straight line mentioned above.

3.4 Examples of Sequoia use

Examples of Sequoia predictions are given for three proteins displaying only α helices, only β strands or both types of secondary structures (Figure 6). It is visible that the Sequoia predictions of α helices and β strands are in quite good agreement with the DSSP predictions. The missing residues in the prediction of secondary structure elements are mostly located at the extremities of the elements in agreement with the previous analyses of Figure 5.

The efficiency of Sequoia prediction was also tested on C_α positions determined using Deeptracer [36,33] on three EM maps obtained from the Electronic Microscopy Data Bank (EMDB) [2]: EMD-23927 [16], EMD-30915 [28], EMD-30942 (to be published). These entries were chosen as they correspond to different protein

[2] www.ebi.ac.uk/pdbe/emdb/

complexes (affinity captured human p97 hexamer, *Salmonella flagella* MS-ring protein FliF 1-456, apo spike protein of SARS-CoV2). They were obtained by single particle reconstruction and correspond to medium resolution data, for which the determination of atomic positions is not straightforward. The resolutions for the entries EMD-23927, EMD-30915, EMD-30942 were respectively of: 4.22, 3.45 and 4.46 Å, and no corresponding PDB entry has been described in EMDB for these data.

The EM maps were uploaded to the Deeptracer Web server [3] and the deep-learning prediction of atoms positions was run using the default parameters. The output containing only C_α atoms was downloaded and given to the Sequoia prediction tool trained on the database of C_α graphs with the classifications α-Other and α-β-Other. The results of the prediction are displayed in Figure 7. The predicted α helices and β strands are drawn in cartoon whereas the residues predicted to belong to the classification Other are drawn as spheres. Sequoia is able to catch quite a number of the secondary structure elements expected in these structures.

4 Discussion

The main outcome of this work is to propose a method for predicting the secondary structure elements of proteins using as input the distances between atoms and not requiring the knowledge of residue succession in protein sequence. To the best of our knowledge, this is the first time in the literature that secondary structures are predicted in such a frame. We showed above that this approach was made possible by a generalization of dihedral backbone angles ϕ and ψ for (i) the case of couples of residues, covalently bonded in the protein sequence or not, as well as for (ii) the case of a C_α atoms graph.

The type of neural network used for the Sequoia prediction is also an innovative aspect of the approach, as it is a message passing neural network (MPN). Although MPN approaches have already been used in the context of ligand docking [12,42,41], this type of neural network is used here for the first time in the context of protein structure prediction. In order to apply the MPN approach, we have constructed a graph on the protein residues in which the existence of an edge depends only on a threshold distance between the residues vertices, and not on their involvement in a covalent bond and is thus independent from the sequence information. This approach can exploit an essential advantage of MPN methods when dealing with fragments of protein structures, as it is the case if NMR spectra are partially assigned, if disordered regions of the protein are not observed, and if one deals with medium-resolution EM maps.

Sequoia performs better than first order statistics (FOS), and is resistant to noise. The classifications producing the best success rates are α-Other, β-Other and α-β-Other, in agreement with the knowledge on the ranges of dihedral angles in proteins. The three classifications α-Other, β-Other and α-β-Other, obtained by Sequoia, are successful at percentages mostly larger than 80% even for the less precise dataset B formed with NMR structures. Sequoia approach is also remarkably resistant to the ablation of protein residues and to the variation of distance threshold between residues.

The examination of individual residue errors in Sequoia revealed that most of these errors are located within the two first or last residues of the considered secondary structure elements. The origin of such errors arises from the choice of the method DSSP [19] as reference for validating Sequoia. Indeed, DSSP implements a discrete classification of residues among secondary structures in which the prediction jumps from one to another value at the limits of secondary structure elements, without continuous interpolation. Such discontinuity disagrees obviously with the protein structure variations which occur continuously along the protein backbone, as shown in the approach screwfit [4], based on a modeling of the protein backbone in terms of a curve with intrinsic torsion.

Sequoia represents also a step toward a coarse-grained perspective of the interval Branch-and-Prune (iBP) approach [23,26]. Indeed, iBP, as well as Sequoia, is based on the use of distances and angles [40] inputs, and was up-to-now, an algorithm basing the protein structure determination on a tree building, each tree level corresponding to atoms. With the help of Sequoia, it should be now possible to consider the replacement of certain groups of atoms by secondary structure elements. In that way, the tree will be simplified and the combinatorial problems due to algorithm complexity reduced.

[3] `deeptracer.uw.edu/home`

Sequoia displays results on a graph containing only atoms C_α, which are similar than the results obtained considering all backbone heavy atoms. Unsurprisingly, the reduced input information produces a decrease of the F1 scores. Nevertheless, Sequoia displays a reasonable robustness with respect to the reduction of the information from the molecular graph. Similarly, Sequoia shows constant success rates or even improvements when the complexity of the graph is increased by increasing the number of neighbors described by the hyperparameter k.

In cryo electronic microscopy (EM), the detection of secondary structure elements in the medium resolution EM maps is a fundamental step for connecting EM signal to structural information. The analysis of C_α graphs performed here have some relationship to the EM maps as the C_α atoms can be considered as a simplified description of the residue electronic density or of the EM map voxel. In this frame, it is interesting to see that the F1 score of 70-80% observed for Sequoia for many of the test described in the present work, are of similar order of values than the success rates observed for deep learning analysis of EM maps [30,36].

References

1. Abbott, S., Iudin, A., Korir, P.K., Somasundharam, S., Patwardhan, A.: EMDB Web Resources. Curr Protoc Bioinformatics **61**(1), 1–5 (03 2018)
2. Andreeva, A., Kulesha, E., Gough, J., Murzin, A.G.: The SCOP database in 2020: expanded classification of representative family and superfamily domains of known protein structures. Nucleic Acids Res **48**, D376–D382 (2020)
3. Berman, H., Bhat, T., Bourne, P., Feng, Z., Gilliland, G., Weissig, H., Westbrook, J.: The Protein Data Bank and the challenge of structural genomics. Nat Struct Biol **7 Suppl**, 957–959 (2000)
4. Calligari, P.A., Kneller, G.R.: ScrewFit: combining localization and description of protein secondary structure. Acta Crystallogr D Biol Crystallogr **68**, 1690–1693 (2012)
5. Chen, S., Dobriban, E., Lee, J.: A group-theoretic framework for data augmentation. Journal of Machine Learning Research **21**, 1–71 (2020)
6. DeLano, W.: The PyMOL Molecular Graphics System, Version 1.2r3pre (2002)
7. Di Paola, L., Giuliani, A.: Protein contact network topology: a natural language for allostery. Curr Opin Struct Biol **31**, 43–48 (2015)
8. Dokmanic, I., Parhizkar, R., Ranieri, J., Vetterli, M.: Euclidean distance matrices: essential theory, algorithms, and applications. IEEE Signal Processing Magazine **32**, 12–30 (2015)
9. Duvenaud, D., Maclaurin, D., Aguilera-Iparraguirre, J., Gómez-Bombarelli, R., Hirzel, T., Aspuru-Guzik, A., Adams, R.: Convolutional networks on graphs for learning molecular fingerprints. In: Advances in neural information processing systems. p. 2224 (2015)
10. El-Gebali, S., Mistry, J., Bateman, A., Eddy, S.R., Luciani, A., Potter, S.C., Qureshi, M., Richardson, L.J., Salazar, G.A., Smart, A., Sonnhammer, E.L.L., Hirsh, L., Paladin, L., Piovesan, D., Tosatto, S.C.E., Finn, R.D.: The Pfam protein families database in 2019. Nucleic Acids Res **47**(D1), D427–D432 (01 2019)
11. Fey, M., Lenssen, J.E.: Fast graph representation learning with PyTorch Geometric. In: ICLR Workshop on Representation Learning on Graphs and Manifolds (2019)
12. Fout, A., Byrd, J., Shariat, B., Ben-Hur, A.: Protein interface prediction using graph convolutional networks. In: Advances in neural information processing systems. p. 6530 (2017)
13. Frishman, D., Argos, P.: Knowledge-based protein secondary structure assignment. Proteins **23**, 566–579 (1995)
14. Gilmer, J., Schoenholz, S., Riley, P., Vinyals, O., Dahl, G.: Neural message passing for quantum chemistry. In: International Conference on Machine Learning (2017)
15. Heal, J., Bartlett, G., Wood, C., Thomson, A., Woolfson, D.: Applying graph theory to protein structures: an Atlas of coiled coils. Bioinformatics **34**, 3316–3323 (2018)
16. Hoq, M.R., Vago, F.S., Li, K., Kovaliov, M., Nicholas, R.J., Huryn, D.M., Wipf, P., Jiang, W., Thompson, D.H.: Affinity Capture of p97 with Small-Molecule Ligand Bait Reveals a 3.6 Å Double-Hexamer Cryoelectron Microscopy Structure. ACS Nano **15**, 8376–8385 (2021)
17. Huan, J., Wang, W., Bandyopadhyay, D., Snoeyink, J., Prins, J., Tropsha, A.: Mining protein family specific residue packing patterns from protein structure graphs. In: Research in computational molecular biology. p. 308 (2004)
18. Ingraham, J., Garg, V.K., Barzilay, R., Jaakkola, T.S.: Generative models for graph-based protein design. Neural Information Processing Systems 32 (2021)
19. Kabsch, W., Sander, C.: Dictionary of protein secondary structure: pattern recognition of hydrogen-bonded and geometrical features. Biopolymers **22**, 2577–2637 (1983)

20. Kearnes, S., McCloskey, K., Berndl, M., Pande, V., Riley, P.: Molecular graph convolutions: moving beyond fingerprints. Journal of computer-aided molecular design **30**, 595–608 (2016)
21. Krishnan, A., Zbilut, J., Tomita, M., Giuliani, A.: Proteins as networks: usefulness of graph theory in protein science. Curr Protein Pept Sci **9**, 28–38 (2008)
22. Lavor, C., Alves, R., Figueiredo, W., Petraglia, A., Maculan, N.: Clifford algebra and the discretizable molecular distance geometry problem. Advances in Applied Clifford Algebras **25**, 925–942 (2015)
23. Lavor, C., Lee, J., Lee-St. John, A., Liberti, L., Mucherino, A., Sviridenko, M.: Discretization orders for distance geometry problems. Optimization Letters **6**, 783–796 (2012)
24. Li, Y., Tarlow, D., Brockschmidt, M., Zemel, R.: Gated graph sequence neural networks. International Conference on Learning Representation (2016)
25. Lian, L., Roberts, G.: Protein NMR Spectroscopy: Practical Techniques and Applications. Wiley (2011)
26. Liberti, L., Lavor, C., Maculan, N., Mucherino, A.: Euclidean distance geometry and applications. SIAM Review **56**, 3–69 (2014)
27. Lim, J., Ryu, S., Park, K., Choe, Y.J., Ham, J., Kim, W.Y.: Predicting drug–target interaction using a novel graph neural network with 3D structure-embedded graph representation. Journal of Chemical Information and Modeling **59**, 3981–3988 (2019)
28. Liu, Y., Soh, W.T., Kishikawa, J.I., Hirose, M., Nakayama, E.E., Li, S., Sasai, M., Suzuki, T., Tada, A., Arakawa, A., Matsuoka, S., Akamatsu, K., Matsuda, M., Ono, C., Torii, S., Kishida, K., Jin, H., Nakai, W., Arase, N., Nakagawa, A., Matsumoto, M., Nakazaki, Y., Shindo, Y., Kohyama, M., Tomii, K., Ohmura, K., Ohshima, S., Okamoto, T., Yamamoto, M., Nakagami, H., Matsuura, Y., Nakagawa, A., Kato, T., Okada, M., Standley, D.M., Shioda, T., Arase, H.: An infectivity-enhancing site on the SARS-CoV-2 spike protein targeted by antibodies. Cell **184**, 3452–3466 (2021)
29. Luisi, P.: Molecular conformational rigidity: An approach to quantification. Naturwissenschaften **64**, 569–574 (1977)
30. Maddhuri Venkata Subramaniya, S.R., Terashi, G., Kihara, D.: Protein secondary structure detection in intermediate resolution cryo-EM maps using deep learning. Nat Methods **16**, 911–917 (2019)
31. Mason, O., Verwoerd, M.: Graph theory and networks in Biology. IET Syst Biol **1**, 89–119 (2007)
32. Paszke, A., Gross, S., Chintala, S., Chanan, G., Yang, E., DeVito, Z., Lin, Z., Desmaison, A., Antiga, L., Lerer, A.: Automatic differentiation in PyTorch. NIPS 2017 Autodiff Workshop (2017)
33. Pfab, J., Phan, N., Si, D.: DeepTracer for fast de novo cryo-EM protein structure modeling and special studies on CoV-related complexes. Proc Natl Acad Sci U S A **118**, e2017525118 (2021)
34. Porogelov, A.: Geometry. Mir Publishers, Moscow (1987)
35. Shen, Y., Bax, A.: Protein structural information derived from NMR chemical shift with the neural network program TALOS-N. Methods Mol Biol **1260**, 17–32 (2015)
36. Si, D., Moritz, S.A., Pfab, J., Hou, J., Cao, R., Wang, L., Wu, T., Cheng, J.: Deep Learning to Predict Protein Backbone Structure from High-Resolution Cryo-EM Density Maps. Sci Rep **10**, 4282 (2020)
37. Sillitoe, I., Bordin, N., Dawson, N., Waman, V.P., Ashford, P., Scholes, H.M., Pang, C.S.M., Woodridge, L., Rauer, C., Sen, N., Abbasian, M., Le Cornu, S., Lam, S.D., Berka, K., Varekova, I.H., Svobodova, R., Lees, J., Orengo, C.A.: CATH: increased structural coverage of functional space. Nucleic Acids Res **49**, D266–D273 (2021)
38. Simonovsky, M., Komodakis, N.: Dynamic edge-conditioned filters in convolutional neural networks on graphs. In: Proceedings of the IEEE conference on computer vision and pattern recognition(CVPR). p. 3693 (2017)
39. Wang, G., Dunbrack, R.: PISCES: a protein sequence culling server. Bioinformatics **19**, 1589–1591 (2003)
40. Worley, B., Delhommel, F., Cordier, F., Malliavin, T., Bardiaux, B., Wolff, N., Nilges, M., Lavor, C., Liberti, L.: Tuning interval Branch-and-Prune for protein structure determination. Journal of Global Optimization **72**, 109–127 (2018)
41. Zhao, T., Hu, Y., Valsdottir, L., Zang, T., Peng, J.: Identifying drug–target interactions based on graph convolutional network and deep neural network. Brief Bioinform **22**, 2141–2150 (2021)
42. Zhu, H., Du, X., Yao, Y.: ConvsPPIS: Identifying Protein-protein Interaction Sites by an Ensemble Convolutional Neural Network with Feature Graph. Current Bioinformatics **15**, 368–378 (2020)

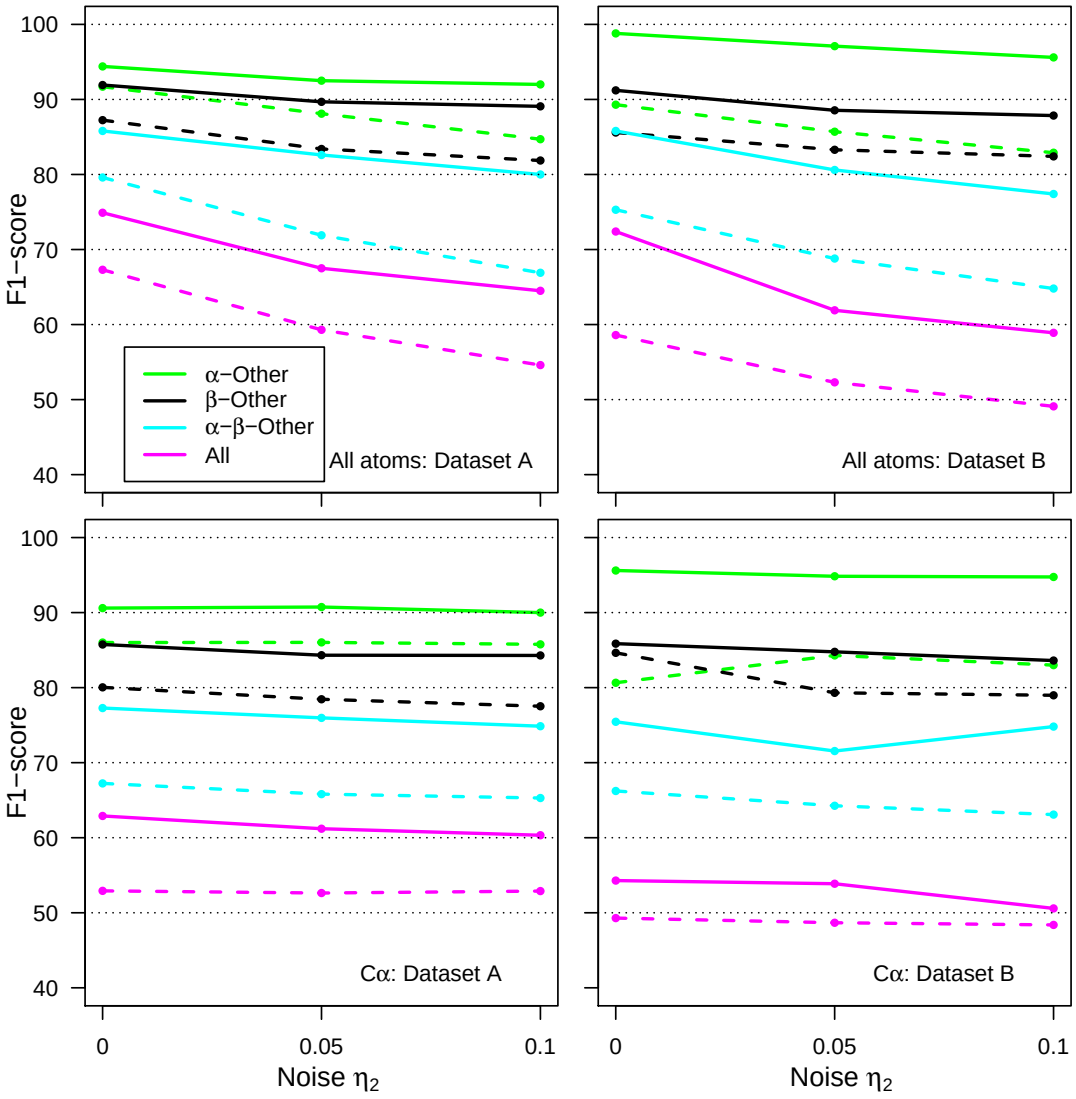

Fig. 3. Sequoia and FOS predictions of secondary structure elements. The predictions (F1-score) are displayed for all atoms (upper panels) and atoms C_α (lower panels), for datasets A (left panels) and B (right panels) and for Sequoia (continuous lines) and FOS (dashed lines). The F1-scores are plotted according to the noise level added to $\cos\omega$ (Section 2.3).

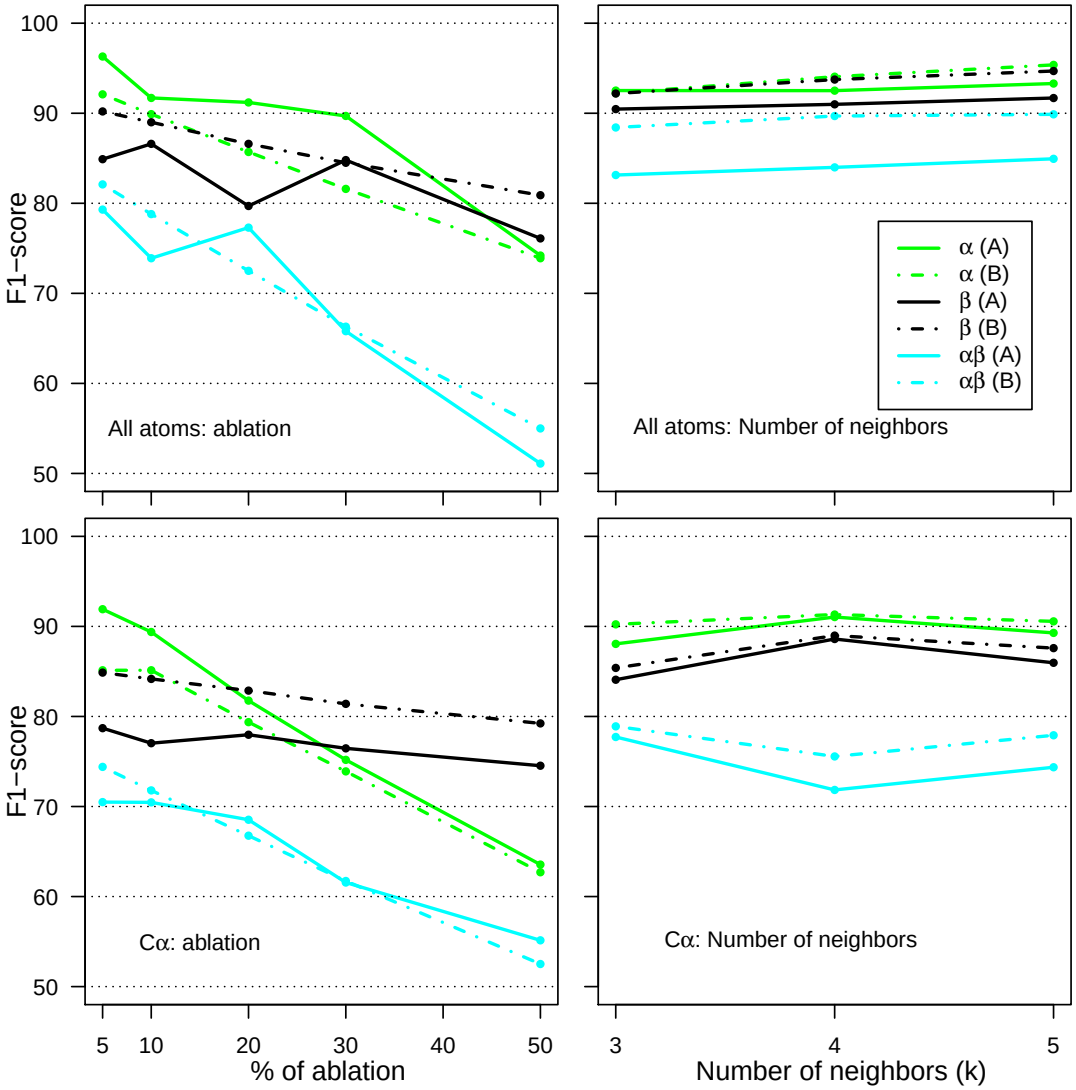

Fig. 4. Effect of input degradation. The Sequoia predictions (F1-score) are displayed for all atoms (upper panels) and atoms C_α (lower panels) and for percentages of ablation (left panels) and number of considered neighbors in the graph (right panels). The predictions are displayed for datasets A (continuous lines) and B (dotdash lines). The predictions are plotted according to the percentage of ablation (left panels) or to the number of considered neighbors in the graph (right panels).

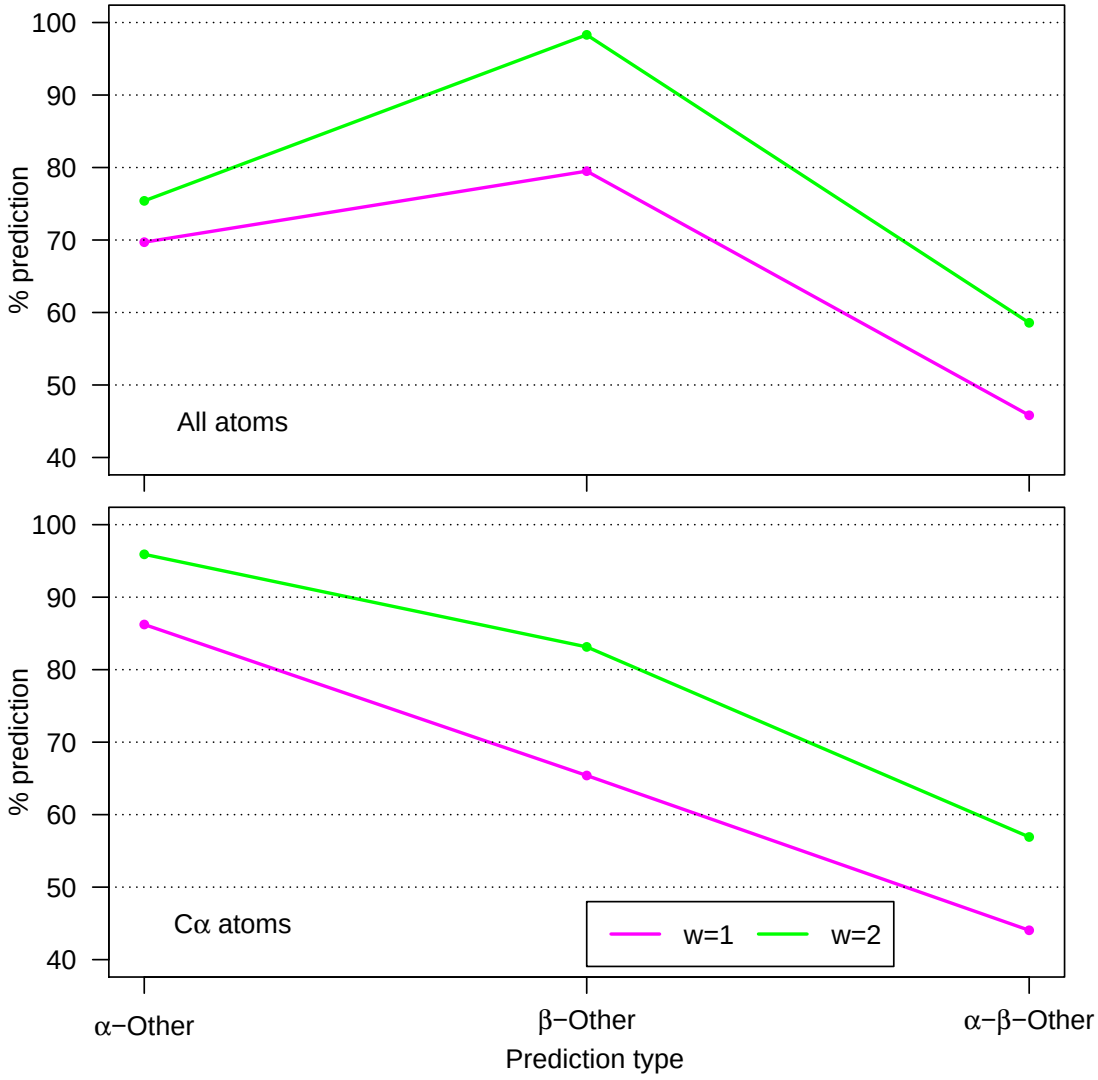

Fig. 5. Position of Sequoia errors in secondary structure elements. The percentages of erroneous Sequoia predictions within the first or last residue of the element ($w = 1$, magenta curve) or within the two first or two last residues of the element ($w = 2$, green curve) are plotted according to the type of prediction (α-Other, β-Other, α-β-Other).

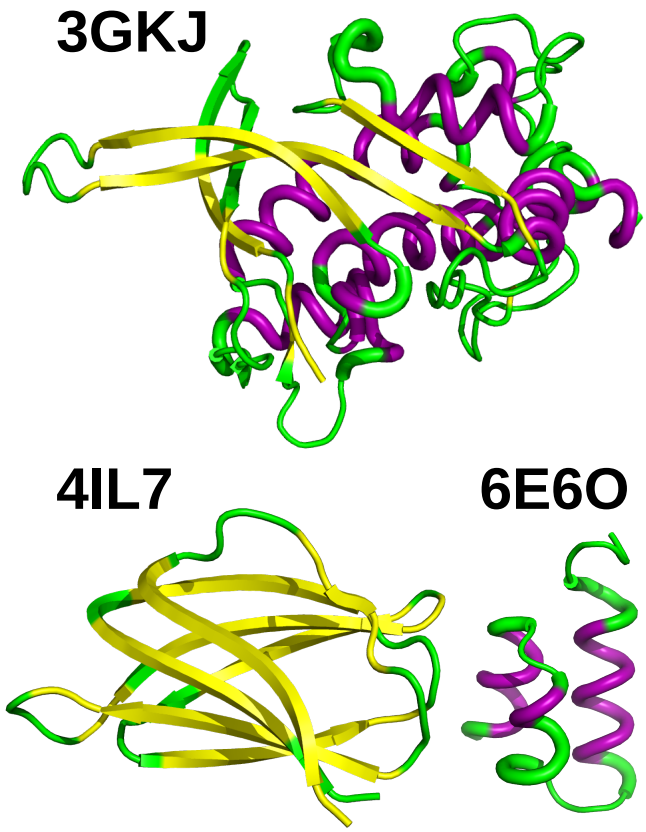

Fig. 6. Three examples of structures taken from dataset A. The structures are drawn in cartoon according to the DSSP predictions [19] whereas the regions predicted by Sequoia as α helices or β strands are colored in purple and yellow. The structures are labeled with the names of the corresponding PDB entries. The structure images were produced using pymol [6].

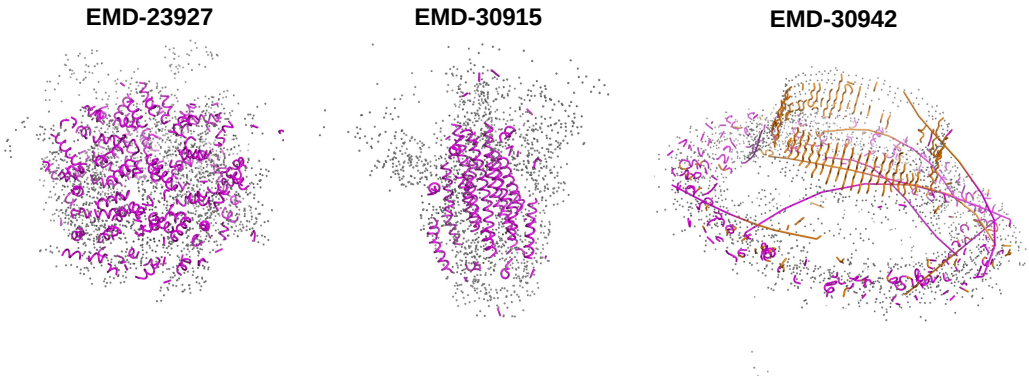

Fig. 7. Results of Sequoia on outputs of the Deeptracer Web site [36] `deeptracer.uw.edu`. The Sequoia prediction α-Other was run on EMD-23927 and EMD-30915 whereas the Sequoia prediction α-β-Other was run on EMD-30942. In each panel, the predicted α helices and β strands are drawn as cartoon, and other residues as grey spheres, and is labeled by the corresponding entry in [1]. The detected α helices are colored in magenta and the β strands in orange.The structure images were produced using pymol [6].