Neural Translation and Evolutionary Multiple-Agent Networks for *Ab Initio* Protein Structure Prediction

Abishrant Panday
Harvard University
abishrantpanday@college.harvard.edu

Joyce Tian
Harvard University
joycetian@college.harvard.edu

Abstract

The three-dimensional structure of proteins, which is constrained by interactions between amino acids, is crucial to their function. However, the protein structure prediction (PSP) problem is NP-complete, and specifically the computation of a tertiary structure given only the amino acid sequence is a yet unsolved challenge. In this work, we present new computational approaches to predict the three-dimensional structure of proteins. Our approaches are based on neural machine translation and multi-agent evolutionary algorithms on cubic lattices. These approaches are developed using amino acid residue and structure information obtained from the Protein Data Bank and ProteinNet12 databases. Our methods are comparable to those in recent publications on short-sequence proteins while requiring a smaller convoluted search space due to more refined inputs, and further may provide valuable insights when used to examine potential protein misfolding patterns which are commonly found in neurodegenerative diseases.

1 Introduction

Proteins are macromolecules found within all living organisms that perform essential inter- and intra-cellular functions, which are intimately linked with their three dimensional structures and conformations [14]. However, there is a large divide within all known protein sequences and knowledge regarding their associated 3-D structures [3]; further, proteins that misfold have been shown to be responsible for a number of fatal diseases, including Alzheimer's, Huntington's, and Parkinson's disease [15, 16]. Therefore, efficient solutions to the protein structure prediction (PSP) problem are crucial for both the severity and scope of questions they address and the potential further insights such solutions may introduce for the perspective of motivations behind key misfolds.

Proteins are uniquely defined by their primary sequence, a polymer consisting of up to 20 unique amino acid residues [17]. However, experimentally determining a protein's 3-D structure tends to be time-consuming and expensive due to the inherent costs associated with common techniques such as X-ray crystallography, NMR, and scanning electron microscopy, each of which may provide differing structure results due to medium differences. On the theoretical front, computation is equally time-consuming due to the combinatorial explosion of possible structures as one moves from a linear amino acid chain to a conformation in three dimensions, as well as the general uncertainty associated with the various cost functions used to model intermolecular interactions; as such, 3-D PSP is considered an NP-complete problem [2, 18].

Potential solutions to PSP fall into two broad approaches: those that utilize genetic homologs and facets of the database of known protein structures, and those that use only the amino acid system and a model of the amino acid interactions to determine structure [19]. The former approach has arisen due to an expanding library of experimentally-verified structures and ongoing genome-mapping efforts since the early 2000s. The latter, termed *ab initio* PSP, is a weaker form of Anfinsen's dogma, which posits that an observed protein structure is a unique and stable free-energy minimum [1]. In our paper, we tackle *ab initio* structure determination, a computationally harder task [20], as it provides

.

more potential applications in understanding the ramifications of aforementioned neurodegenerative diseases, whose misfolded proteins don't all have homologs and similar metadata [4].

Moreover, within the scope of the PSP, we utilize what is known as an on-lattice model to represent the configuration of amino acids for simulation. As opposed to off-lattice models, which don't constrain the space of possible points and hence require far more computational power [4], on-lattice models reduce the dimensionality of the PSP, which allows for more nuance within the size and structure of protein chains and the complexity of their associated potential energy function [21].

Because of the computationally expansive search space for protein conformations, biologically-informed algorithms have become a prominent approach to finding potential solutions [22]. Within this space, the main algorithms that have historically been used to solve the PSP are genetic algorithms [23], differential evolution methods [24], and particle swarm optimization [25]. Of these approaches, we focus on a modified genetic algorithm (GA), which are search algorithms based on notions of natural selection and evolution [26]. Genetic algorithms consist of an overall fitness function that serves as an indicator of solution quality. Each solution is an individual within a base population that undergoes 'mutations' and 'recombinations' to generate new solutions, whose viability is determined through the fitness function. After each step of the genetic algorithm, some individuals are selected and (re)combined to form a new base population, which then repeats the same process [5].

Finally, multi-agent networks (MANs) are an approach to solving high-dimensionality, complex problems by utilizing an agent-interaction framework. In this approach, agents, which are comprised of independent processes, interact with the environment and other agents under certain circumstances in order to solve a more complex problem [27]. Within this paradigm, each agent can choose to interact with the environment, follow its own directives, or cooperate with other agents, which allows complex problems to be tackled in independently moving subparts [28, 29].

In section 2, we discuss relevant recent literature within the translation and multi-agent network frameworks whose insights we seek to utilize. In section 3, we detail our language translation based model for determining secondary structures and follow that with a generalized protein structure prediction model utilizing evolutionary multi-agent networks in 4. Section 5 then describes our data cleaning and training methods, which is followed by an analysis of our results in section 6 and a conclusion. All relevant code is hosted at `https://github.com/jtianesq/protein-prediction-nmt-evo`

2 Literature review

Our algorithmic design consists of an applied neural machine translation model and an evolutionary multi-agent network, both active fields in their general formulation but sparser with regards to their application in protein structure prediction. Bahdanau et al. [8] overcame the bottleneck of a fixed-length encoding in the traditional encoder-decoder translation framework by allowing the model to soft-search parts of a source sentence relevant to predicting a target word. This is achieved through a decoder attention mechanism which takes a sequence of encoder annotations providing information in each input word surrounding. To construct this mechanism, the encoder was created using a bidirectional RNN (biRNN), wherein each annotation summarizes preceding and following words and is used by the decoder to compute gradients for soft alignment.

While the approach of Bahdanau et al. increased translation performance significantly, particularly with regards to flexibility in sentence lengths, there exist only a few recent papers that directly utilize neural machine translation as a strategy to address the PSP probem. One such paper, by Cao et al. [9], constructs a "ProLan" language based on the UniProtKB database and translates it into the "GOLan" language, which is based on the gene ontology (GO) terms from a protein ID. Within this framework, protein sequences are constructed into sentences to be translated by randomly fragmenting a given protein into $k-$mers, each of which is a word. When comparing to state of the art prediction models in PANNZER, DeepGO, and FANN-GO, the ProLanGO formalism of Cao et al. performed worse, but it was comparable or better to traditional machine language translation models.

In order to contextualize the efficacy of our proposed neural translation scheme for protein secondary structure prediction, which is based on the jointly align and translate model in Bahdanu et al. [8] described above, we note a recent review of protein structure prediction by Hansen et al. [10]. Here, a comparison of reported accuracies for contextual secondary structure predictors showed average accuracy of around 84% for the 2013-2019 period. These models, however, utilized evolutionary profiles in determining final structures, taking advantage of homologous sequences and other dependencies not directly expressed within the data. We look to determine whether a model operating on an *ab initio* paradigm can determine secondary structures with comparable accuracy.

On the front of multi-agent networks, we note a lack of published works where MANs are applied to the protein structure problem, especially when considering only methods that follow from first principles, *ab initio* designs. The main publication that operates within this space is by de Lima Correa et al. [2], who utilized database information to construct a 3-D protein structure via the use of a multi-agent system. The authors utilized database heuristics such as an angle probability list and information on backbone and side-chain angles that informed and simplified their search space. Their system was then tested on 8 target proteins, which we will also test our model against, in order to calculate secondary structure analysis and average RMSD.

Within this regime, Campeotto et al. [30] constructed a multi-agent network to predict protein structure, wherein agents represent a unique fraction of the overall protein, thus fragmenting the overall chain amongst the agents who then try to determine low-energy folds per section. The best folds are then aggregated in order to return a fully folded protein. This methodology differs from our approach, where the protein itself is the same per agent and the actual three dimensional rotations and translations of bonds are what agents operate on. Our reasoning for preferring this model is that it better models the effect that neighbouring amino acid residues have on the energy profile of each proposed rotation and translation.

3 Asynchronous bi-directional neural machine translation

The secondary structure of a protein is defined by interactions between segments of the amino acid chain; for instance, hydrogen bonding between polar residues lower local potential energy and thus formation of secondary structures such as α-helices, β-sheets, among others [31]. To model the transition from primary to secondary structure units, we decided to implement the RNNsearch algorithm specified by Bahdanau et al. [8]. Both foreign language translation, the predominant use of the RNNsearch algorithm in prior literature, and the translation of a primary sequence to its secondary structure units involve largely local effects. However, protein translation is impacted by multiple loci for both languages and protein structure determination for which we cannot suppose any a priori distribution, and thus bidirectional models like RNNsearch, as opposed to exclusively feed-forward or feed-backward models, lend themselves more naturally to providing such a framework.

3.1 Implementation

Our RNNsearch-based algorithm, **ProteinSearch**, largely follows the original RNNsearch algorithm with notable divergences in vocabulary treatments, scoring, and translation directionality. Like [8], given an input sequence $x = x_1 \dots x_n$ and target output sequence $y = y_1 \dots y_n$, **ProteinSearch** utilizes a bidirectional RNN as an encoder, whose forward component $\overrightarrow{f} = (\overrightarrow{f_1}, \dots, \overrightarrow{f_n})$ applies a conditional gate recurrent unit (GRUcond) using an activation function g and context vector c_k such that $\overrightarrow{s_k} = g(\overrightarrow{s_{k-1}}, y_{k-1}, c_k)$ for each forward hidden state $\overrightarrow{s_k}$ and whose backward component $\overleftarrow{f} = (\overleftarrow{f_1}, \dots, \overleftarrow{f_n})$ reads over $y_n, \dots, y_1$ such that $\overleftarrow{s_k} = g(\overleftarrow{s_{k-1}}, y_{n-k-1}, c_k)$ for each backwards hidden state $\overleftarrow{s_k}$. Annotations thus still consist of a concatenation of the corresponding forward and backward state $s_j = [\overrightarrow{s_k}; \overleftarrow{s_k}]^T$. The context vectors $c_0, \dots c_n$ use an alignment measure between the ith hidden state and kth annotation computed as the observed conditional probability of the value of h_k yielding the most probable ith output.

ProteinSearch differs from traditional translation models in that it is fed the entire vocabulary for both primary and secondary sequences due to the small sizes for both, and therefore does not require a token for unknown characters. Further, its scoring function for a given translation is based solely on levels of correctness, where one secondary structure proposal may be more similar to the actual secondary structure than another possibility (e.g. a β-strand interpretation would be more similar to a desired β-sheet output than an α-helix interpretation). Finally, the major change within our translation framework is that **ProteinSearch** also uses bi-directional decoding in addition to encoding, constructed identically to the encoding RNN, so as to allow for translation to be informed by both potential reverse hidden states as well as the former. We believe this network construction better reflects the significance in inter-residue dependence within secondary structure conformations, both forwards and backwards within the chain, and further explores more distant relationships within the primary sequence.

4 Evolutionary multi-agent network

In agent-based modeling, the individual actions of each agent and their interactions with each other and the environment are combined in order to evaluate an overall effect on a complex system. The framework of multi-agent networks consists of an agent class, with some initial condition and a step function that determines the actions an agent takes at each time-step. The final component of the model is a scheduler, which controls the order in which agents are activated. The most common scheduler in multi-agent network implementations is random activation, which activates all agents once per step in random order. This activation of all agents in every step is a common formulation in standard theory [11] and implementation, such as Mesa, NetLogo, and Repast. Our multi-agent network deviates from this strategy by activating one random agent at each step.

4.1 Free Energy Function

The objective function that our algorithm seeks to minimize in order to find appropriate 3-D protein structure is the total free energy of interaction between nearby amino acid residues. We construct the basic framework for our free energy by considering the change in Gibbs free energy, ΔG, per amino acid residue, as measured relative to a zero energy of interaction between two adjacent glycines [12]. In order to determine the energy of interaction between any two amino acids, we first grouped them into the following five distinct categories: positively charged residues (H, K, R), negatively charged residues (D, Q), aromatic residues (F, W, Y), aliphatic residues (I, L, P, V), non-charged polar residues (E, N, S, T), small residues (A, G), and sulfur-containing residues (C, M). Given these designations, we then calculate the free energy between two amino acids as

$$f_{ij} = c_{ij} \nu_{ij} \left(\Delta G_i \right), \tag{1}$$

where c_{ij} is proportional to inverse distance between the amino acid side-chains and ν_{ij} scles relative the interaction between two residues based on their categories at the given distance.

4.2 Implementation

Our multi-agent evolutionary algorithm, **MultiFold**, instantiates one **AminoAgent** x_i to represent the location of each amino-acid residue in an input primary sequence $a_1 \ldots a_n$, wherein each chain is held within an **AminoModel**. Each x_i holds knowledge of a main-chain and side-chain location and are placed on a 3D cubic lattice such that for all $i \in \{1, \ldots, n-1\}$, x_{i-1} and x_i's main-chain positions are 1 unit apart, and all main-chain and side-chain pairs are 1 unit apart, where each unit is 3.6 Angstroms. Maintaining both main-chain and side-chain components for each amino-acid representation prevents over-compression of the conformations and further allows us to model the energy impact of internally-facing versus externally-facing side-chains relative to the skeleton structure of the main chains.

In each step of the **AminoModel**, one **AminoAgent** $x_i, i \in \{1, \ldots, n-1\}$ is chosen per some choice function c to mutate its position. The bond (x_{i-1}, x_i) is randomly rotated such that the resulting conformation is non-overlapping within the cubic lattice, where specifically all following relative conformations (x_k, x_{k+1}) where $k \geq i$ are held constant; each step mutation thus emulates the gradual folding proteins undergo in actuality. Then, every e steps, the **AminoModel** is annealed with a function f which, given the current model and the k prior models examined which have the lowest global energies, chooses the top model with probability related to the fraction of total steps taken, $p(\text{best model}) \propto \frac{i}{n}$, and otherwise randomly selects from the k other models. Our annealing function f allows us to explore the conformation space moreso at the beginning of the algorithm and less so when Finally, after n total steps are taken, the model with the lowest global energy thus far is output.

Algorithm 1: MULTIFOLD outputs the model whose agent positions minimize global energy

```
Input: A primary amino-acid sequence $a = a_1 \ldots a_n$, integer specifications for the number of
       best models to store $k$, the total steps $n$, and the intervals for each epoch $e$
Output: The minimum-energy model in the top $k$ models stored at the end of $n$ steps
$model = AminoModel(a)$;
$topk \leftarrow [\,]$;
$energies \leftarrow [\,]$;
for $i \leftarrow 1$ to $n$ do
    $global\_energy \leftarrow model.step()$;
    if $i \mod e = 0$ then
        $energies \leftarrow global\_energy$;
        $topk \leftarrow (global\_energy, model)$
        $topk.sort()[: k]$;
        $model \leftarrow f(topk + [model], i, n)$;
    end
end
return $topk[0]$
```

Table 1: A comparison of the reported accuracies of secondary structure prediction models based on various datasets.

Predictor	Year Published	Dataset	Protein Count	Accuracy (%)
MUFOLD-SS (Fang et al., 2018a)	2018	Easy case	226	88.20
MUFOLD-SS (Fang et al., 2018a)	2018	Hard case	95	83.37
PORTER 5.0 (Torrisi et al., 2018)	2018	Full set	3154	84.19
SPOT-ID (Hanson et al., 2018b)	2018	TEST-2018	250	86.18
PROTEINSEARCH (Panday & Tian, 2019)*	2019	PDB	150	79.48

5 Methods

5.1 Data acquisition and cleaning

In order to train and validate our models, we utilized the ProteinNet12 [13] and RCSB Protein Data Bank (PDB) datasets. Due to the size of the ProteinNet12 database, which was in excess of 10 terabytes, we randomly sampled 50 gigabytes of data to download. The dataset itself comes split into training and validation subsets, the relative proportion of which our sampling left unchanged. Given our goal of creating *ab initio* algorithms for protein structure prediction, we cleaned the ProteinNet12 dataset by separating and discretizing secondary structures and parsing out homologs and evolutionary sequence alignments. For PDB data, we created an algorithm to query pdb files from the RCSB database and, from each file, extract the primary and secondary structures.

5.2 Model Training

Each dataset was pruned to single-strand proteins with sub-128 residues whose structures were measured in aqueous environments to best standardize potential structural relationships. For the secondary structure prediction, structures with notations within the .PDB files were noted for Q3 secondary structure prediction (designation of helix (H), strand (E), coil (C), or none (X)), while DSSP was run on each residue for further Q3 elucidation and for Q8 secondary structure designations (3-turn helix (G), 4-turn helix (H), 5-turn helix (I), β-stand (E), β-bridge (B), turn (T), bend (S), and others (C)). The latter characterization is more fine-grained, with each designation falling into a broader Q3 category; as such, the scoring function for Q8 structure prediction gave a 0.5 score for matching Q3 categories given differing Q8 prediction and actual values.

The **ProteinSearch** algorithm was trained on an 80/20 split on each pruned dataset for secondary structure prediction, wherein the input was the primary sequence in the form of one-letter abbreviations for each amino acid residue, and the expected output would be the one-to-one Q3 or Q8 designation for each amino acid. The **MultiFold** algorithm takes the same primary sequence input and outputs a 3D rendering of the sequence on a simple cubic lattice, such that the distance between two adjacent points on the lattice represents 3.6 Angstroms. For comparison, we converted each 3D-representation into its RMSD-minimizing cubic lattice representation using Mann et al.'s LatFit program [32], wherein we then aimed to minimize the RMSD error of the solution output after 100,000 steps based on altering k, e, c, and f.

6 Results

6.1 Secondary structure prediction

Accuracy: Utilizing the **ProteinSearch** algorithm, 150 sub-128-residue amino-acid sequences which were not used for training were randomly sampled from the PDB and their corresponding secondary structures (SS) were computed. These determinations were then checked by referencing the experimentally-determined secondary structures found in the database or through DSSP. The average accuracy, in secondary structure determination was then computed:

$$\text{accuracy } = \frac{\sum \# \text{ of characters in SS matching experimental data}}{\sum \text{ total } \# \text{ characters in SS}} \tag{2}$$

We compared the average accuracy of **ProteinSearch** with state-of-the-art algorithms for protein structure determination from [10], which can be seen in **Table 1**. Our results indicate that, overall, our algorithm for determining secondary structure is close to the accuracy exhibited by state-of-the-art techniques. We theorize that the lower accuracy in our experiment resulted from randomly sampling sub-128-residue amino acids, since our RNN, similar to the translator created by [8], performs better on smaller inputs. We verify this theory in the experiment to follow, which shows that our average accuracy on smaller inputs is comparable to and better than some aspects of SOTA models.

Q-index: We calculate another metric for determining the accuracy of the model, this time with respect to specific secondary structure components, called the $Q-$index:

$$Q_S(P/E) = \frac{\sum \# \text{ S secondary structures correctly predicted}}{\sum \# \text{ S secondary structures experimentally verified}} \tag{3}$$

For each secondary structure, S, the Q-index measures the accuracy of the model in correctly predicting the structure with respect to experimentally verified measurements. We tabulate our computed Q-indices for eight proteins, $\{$1AB1, 1ACW, 1L2Y, 1DFN, 2P81, 1K43, 2MR9, 1WQC$\}$ in **Table 2**. These proteins are the same ones used in a protein structure determination study by de Lima Correa [2], which allows us to analyze both overall accuracy of **ProteinSearch** and its ability to identify specific secondary structure categories.

Table 2: An overview of Q-index measures for specific proteins as output by **ProteinSearch**.

PDB ID	$\%Q_H(P/E)$	$\%Q_E(P/E)$	$\%Q_T(P/E)$	$\%Q_C(P/E)$	Avg (%)
1AB1	100 (20/20)	100 (4/4)	100 (5/5)	88.2 (15/17)	95.7
1ACW	100 (9/9)	100 (10/10)	40 (4/5)	20 (1/5)	82.8
1L2Y	100 (12/12)	-	-	87.5 (6/8)	90.0
1DFN	-	100 (16/16)	55.5 (4/9)	80.0 (4/5)	80.0
2P81	100 (27/27)	-	60 (4/5)	91.6 (11/12)	95.5
1K43	-	100 (6/6)	40.0 (5/5)	100 (1/3)	85.7
2MR9	100 (30/30)	-	66.6 (6/9)	60.0 (2/5)	86.3
1WQC	100 (18/18)	-	-	100 (8/8)	100

We first note that for secondary structures H, E, which represent α-helices and β-strands, respectively, our model has 100% accuracy, which we believe stems from the fact that both structures are very well-defined spatially by polar, hydrophilic, and electrostatic forces. In contrast, structures T, C, which represent turns and coils, respectively, have a significantly lower $Q-$index. This is most likely because amino acids that determine turns are impacted more heavily by the t-mean-square deviation of atomic positions between the computed structure and experimentally determined ones; in particular, turns are usually informed by interactions with larger neighbouring chains, which lowers the ability of our model to discriminate them as accurately. Coils, meanwhile, are generally harder to classify for most algorithms, including the multi-agent method from [2], because they constitute a larger class that is less well-defined structurally and through electrostatic interactions. Comparing the results to those obtained by de Lima Correa et al.[2], we find that our model exhibits higher Q-scores on average, while retaining the same relative difficulty in correctly classifying coils.

6.2 3-D structure prediction

We measure the efficacy of **MultiFold** in determining three dimensional structures from amino acids by computing the root mean square deviation (RMSD) of atomic positions between the computed structure and experimentally determined ones. The RMSD for each solution was found by mapping the first and second residue positions to those specified within the PDB file, translating all other positions accordingly, and then taking the average of the mean squared distances between each subsequent predicted and actual residue location. We find that the RMSD values for **MultiFold**, which are detailed in **Table 3**, are comparable to those computed by de Lima Correa et al. [2]. In **Figure 1**, we depict a particularly well-fit 3-D representation, which qualitatively can be seen to strongly resemble the structure found within PDB. We theorize that the slightly higher RMSD values for some proteins is a result of hardware limitations that required the number of generations in **MultiFold** to remain small enough for computability in reasonable time. We propose running these experiments again when using a cloud cluster or supercomputer array in order to obtain the best average RMSD per protein.

Table 3: An overview of average RMSD error produced by **MultiFold**.

PDB ID	RMSD (Å) in [2]	RMSD (Å) of **MultiFold**
1AB1	6.06	7.15
1ACW	2.77	4.02
1L2Y	1.86	1.55
1DFN	5.02	5.38
2P81	5.37	4.42
1K43	0.56	0.71
2MR9	1.90	1.00
1WQC	2.61	2.31

A
C
D
E
G
H
K
L
N
P
Q
V
Y

(a) 1WQC conformation on cubic lattice output by **MultiFold**.

(b) 1WQC 3D structure from PDB.

Figure 1: A visual comparison between 1WQC's lattice fold structure and the 3D structure recorded within PDB.

6.3 Disease-specific applications

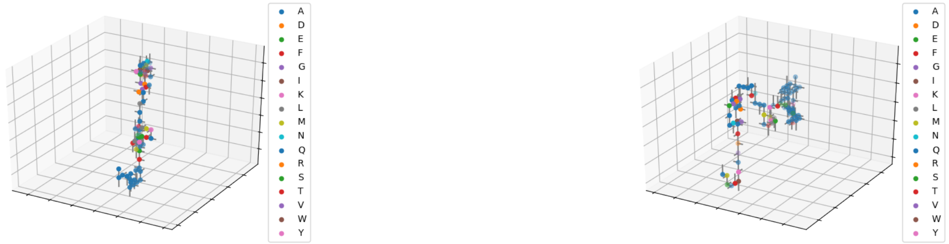
Beyond the theoretical importance of improving on solutions to the primary structure problem without the use of advanced heuristics in data preparation and manipulation, our paper was motivated to tackle this query due to the important role misfolded proteins play in many rare diseases.

6.3.1 Huntington's Disease

Huntington's disease, which is an incurable neurodegenerative disease, is caused by a mutation in the $IT - 15$ gene that increases the number of CAG nucleotide repeats; the resulting protein, huntingtin, then contains an excessive amount of glutamines [33]. These form a polyglutamine (or polyQ) tract at the end of huntingtin which causes misfolding and aggregation. We first examine the protein 3IO6, the end of which contains the glutamates which end up aggregating in Huntington's. Our model first determines a structure for 3IO6 **Figure 2a** whose RMSD is low, indicating the model's ability to accurately determine the beginning structure. Then, we linearly increase the number of glutamines (Q) at the end of the tail until the protein structure exhibits clumping, which happens at 36, **Figure 2b**. This corroborates experimental findings of a jump in 3-D structural disarray with increasing glutamine repeats [34]. We thus show that our model is effective in verifying abnormal protein folds for the 3IO6 protein associated with Huntington's disease. Moving forward, this means that our procedure could be used to modify other misfolding candidates in order to evaluate their structures and focus experimental studies, which tend to be more expensive and time consuming.

Table 4: An overview of RMSD error within Huntingtin polyQ-variants.

3IO6 Variant	RMSD (Å)
Control	1.93
3IO6 with 36Q tail	5.50



(a) 17-Q tail conformation of Huntingtin, as output by **Multifold**.

(b) 36-Q tail conformation of Huntingtin, as output by **Multifold**.

Figure 2: A visual comparison between the predicted conformations for a 17-length and 36-length poly-glutamine tail within 3IO6.

7 Conclusion

In this paper, we propose two potential *ab initio* algorithms which extend prior work regarding similar problems of constrained optimization and apply them in predicting secondary and tertiary structure given a protein's primary structure. The first proposed algorithm, **ProteinSearch**, aims to extend neural machine translation [8] to use in the one-to-one translation of primary to secondary structure and better incorporate holistic knowledge of the primary structure using a bi-directional decoder in addition to a bi-direction encoder. The second proposed algorithm, **MultiFold**, extends prior work on multi-agent networks to modeling the general energy transactions between agent representatives for each amino acid within a primary sequence, wherein the aim for each agent is to reduce individual amino acid free energy. Our algorithm proceeds to play this game through an evolutionary framework, wherein the agent positions and energies are sampled from a population such that **MultiFold** is more probable to explore brand new conformations at the beginning and then to progressively sample more local changes towards the end of its stepcount, wherein conformations are constrained to a cubic lattice to diminish the combinatorial explosion of possible folds. Our results are comparable to state-of-the-art work even when working from first-principles, which shows the promise in exploring and tuning these approaches further. Finally, we note a more focused application of **MultiFold** in identifying selective mutations within proteins with the propensity to misfold, which we will expand with further testing.

References

[1] Pedersen, J. T., & Moult, J. (1996). Genetic algorithms for protein structure prediction. *Current Opinion in Structural Biology*, 6(2), 227-231.

[2] de Lima Corrêa, L., Inostroza-Ponta, M., & Dorn, M. (2017, June). An evolutionary multi-agent algorithm to explore the high degree of selectivity in three-dimensional protein structures. In *2017 IEEE Congress on Evolutionary Computation (CEC)* (pp. 1111-1118). IEEE.

[3] Narloch, P. H., & Parpinelli, R. S. (2017, October). The protein structure prediction problem approached by a cascade differential evolution algorithm using ROSETTA. In *2017 Brazilian Conference on Intelligent Systems (BRACIS)* (pp. 294-299). IEEE.

[4] Rashid, M. A., Iqbal, S., Khatib, F., Hoque, M. T., & Sattar, A. (2016). Guided macro-mutation in a graded energy based genetic algorithm for protein structure prediction. *Computational biology and chemistry*, 61, 162-177.

[5] Borguesan, B., e Silva, M. B., Grisci, B., Inostroza-Ponta, M., & Dorn, M. (2015). APL: An angle probability list to improve knowledge-based metaheuristics for the three-dimensional protein structure prediction. *Computational biology and chemistry*, 59, 142-157.

[6] Deng, H., Jia, Y., & Zhang, Y. (2018). Protein structure prediction. *International Journal of Modern Physics B*, 32(18), 1840009.

[7] Kuhlman, B., & Bradley, P. (2019). Advances in protein structure prediction and design. *Nature Reviews Molecular Cell Biology*, 20(11), 681-697.

[8] Bahdanau, D., Cho, K., & Bengio, Y. (2014). Neural machine translation by jointly learning to align and translate. *arXiv preprint* arXiv:1409.0473.

[9] Cao, R., Freitas, C., Chan, L., Sun, M., Jiang, H., & Chen, Z. (2017). ProLanGO: protein function prediction using neural machine translation based on a recurrent neural network. *Molecules*, 22(10), 1732.

[10] Hanson, J., Paliwal, K. K., Litfin, T., Yang, Y., & Zhou, Y. (2019). Getting to know your neighbor: protein structure prediction comes of age with contextual machine learning. *Journal of Computational Biology.*

[11] Hanna, L., & Cagan, J. (2009). Evolutionary multi-agent systems: an adaptive and dynamic approach to optimization. *Journal of Mechanical Design*, 131(1), 011010.

[12] Wallqvist, A., & Ullner, M. (1994). A simplified amino acid potential for use in structure predictions of proteins. *Proteins: Structure, Function, and Bioinformatics*, 18(3), 267-280.

[13] AlQuraishi, M. (2019). ProteinNet: a standardized data set for machine learning of protein structure. *BMC bioinformatics*, 20(1), 311

[14] Lesk, A. M. (2013). Introduction to Bioinformatics, 4th Edn.

[15] Adam, S. (2003). Protein misfolding. *Nature Reviews Drug Discovery*, 426(6968), 78-102.

[16] Dobson, C. M. (2003). Protein folding and misfolding. *Nature,* 426(6968), 884.

[17] Anfinsen, C. B. (1973). Principles that govern the folding of protein chains. *Science*, 181(4096), 223-230.

[18] Guyeux, C., Côté, N. M. L., Bahi, J. M., & Bienia, W. (2014). Is protein folding problem really a NP-complete one? First investigations. *Journal of bioinformatics and computational biology*, 12(01), 1350017.

[19] Defay, T., & Cohen, F. E. (1995). Evaluation of current techniques for ab initio protein structure prediction. *Proteins: Structure, Function, and Bioinformatics*, 23(3), 431-445.

[20] Dodson, E. J. (2007). Computational biology: Protein predictions. *Nature*, 450(7167), 176.

[21] Miyazawa, S., & Jernigan, R. L. (1985). Estimation of effective interresidue contact energies from protein crystal structures: quasi-chemical approximation. *Macromolecules*, 18(3), 534-552.

[22] Benítez, C., Parpinelli, R. S., & Lopes, H. S. (2016, July). An ecologically-inspired parallel approach applied to the protein structure reconstruction from contact maps. In *Proceedings of the 2016 on Genetic and Evolutionary Computation Conference Companion* (pp. 1299-1306). ACM.

[23] Custódio, F. L., Barbosa, H. J., & Dardenne, L. E. (2014). A multiple minima genetic algorithm for protein structure prediction. *Applied Soft Computing*, 15, 88-99.

[24] Dorn, M., e Silva, M. B., Buriol, L. S., & Lamb, L. C. (2014). Three-dimensional protein structure prediction: Methods and computational strategies. *Computational biology and chemistry*, 53, 251-276.

[25] Eberhart, R., & Kennedy, J. (1995, November). Particle swarm optimization. In *Proceedings of the IEEE international conference on neural networks* (Vol. 4, pp. 1942-1948).

[26] Luke, S., 2009. Essentials of metaheuristics. 1 ed., Lulu. MacKerrel, A., 2010. Empirical force fields. *Springer*. chapter 2. pp. 45–69.

[27] Wooldridge, M. (2009). *An introduction to multiagent systems*. John Wiley & Sons.

[28] Jennings, N. R., Sycara, K., & Wooldridge, M. (1998). A roadmap of agent research and development. *Autonomous agents and multi-agent systems*, 1(1), 7-38.

[29] Wooldridge, M., & Jennings, N. R. (1995). Intelligent agents: Theory and practice. *The knowledge engineering review*, 10(2), 115-152.

[30] Campeotto, F., Dovier, A., & Pontelli, E. (2013, October). Protein structure prediction on GPU: a declarative approach in a multi-agent framework. In *2013 42nd International Conference on Parallel Processing* (pp. 474-479). IEEE.

[31] Lehninger, Albert L., David L. Nelson, Michael M. Cox, and Michael M. Cox. *Lehninger principles of biochemistry*. Macmillan, 2005.

[32] Mann, M., Saunders, R., Smith, C., Backofen, R., & Deane, C. M. (2012). Producing high-accuracy lattice models from protein atomic coordinates including side chains. *Advances in bioinformatics*, 2012.

[33] Arrasate, M., & Finkbeiner, S. (2012). Protein aggregates in Huntington's disease. *Experimental neurology*, 238(1), 1-11.

[34] Warner IV, J. B., Ruff, K. M., Tan, P. S., Lemke, E. A., Pappu, R. V., & Lashuel, H. A. (2017). Monomeric huntingtin exon 1 has similar overall structural features for wild-type and pathological polyglutamine lengths. *Journal of the American Chemical Society*, 139(41), 14456-14469.