Deep Reinforcement Learning for Protein Folding in the Hydrophobic-Polar Model with Pull Moves

Haoyang Yu [1], John S. Schreck[2] and Wann-Jiun Ma[3]
[1] Duke University; [2] National Center for Atmospheric Research; [3] Research.ai, LLC

Introduction

We formulate, as a deep reinforcement learning problem, the protein folding problem of predicting a folded protein structure given an amino acid sequence in the Hydrophobic-Polar (HP) model [2] .

Motivation

- The ability to predict the folded structures and the folding dynamics given an amino acid sequence has a wide range of applications, such as disease prevention and computational drug design.
- Creating a computer agent to predict protein structures and simulate the folding process may be able to forgo the high degree of time, computational expenses, and expertise needed to use crystallography and NMR spectroscopy to identify the ground state structures and folding pathways.

Challenges

- The protein folding problem in the HP model is NP-complete [1].
- Exponentially large state space of possible configurations of proteins.

Deep Reinforcement Learning Algorithms

We implement a few standard DRL algorithms such as Policy/Value Iteration, DQN, and MCTS, as well as AlphaGo Zero with pretraining, which is a combination of MCTS with deep neural networks to approximate the policy and value function, to train a self-folding agent in a "lattice" environment. We add the pretraining portion to speed up the training process by pre-training the agent with MCTS without the neural network before transitioning to self-play using the neural network to evaluate states found with MCTS.

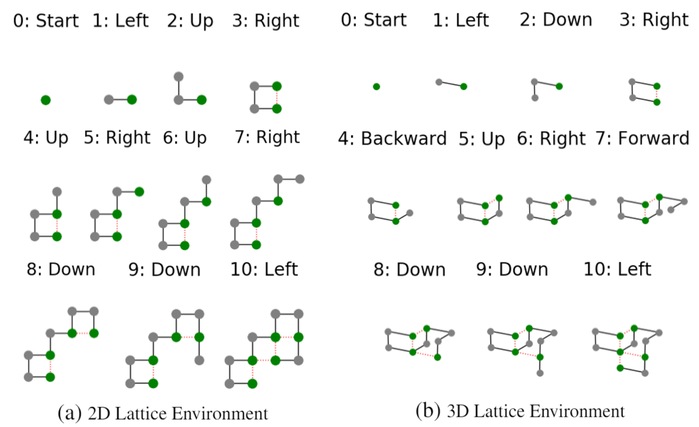
Method	Time / Sequence Length (2-D)	Time / Sequence Length (3-D)
Policy Iteration	15 mins/14	37 mins/5
Value Iteration	15 mins/14	37 mins/5
DQN	30 mins/20	10 mins/15
Prioritized DQN	30 mins/20	10 mins/15
Dueling DQN	30 mins/20	10 mins/15
MCTS	5 mins/20	3 mins/12
AlphaGo Zero +pretraining	<1 min/40	–

Table 1: Max sequence length and inference time to fold comparisons

All the algorithms besides AlphaGo Zero with pretraining are trained specifically for one sequence and times in Table 1 show the inference plus training time. The AlphaGo Zero approach is trained on a set of sequences (so that it may generalize to unseen sequences); here inference is fast but training slow.

Models

Lattice Environment – Build the protein up one amino acid residue at a time, with each additional amino acid residue being placed on the lattice adjacent to the last one placed.



(a) 2D Lattice Environment (b) 3D Lattice Environment

Figure 1: Folding a sequence in the 2D and 3D lattice environments. A green dot represents H, gray represents P, black lines represent covalent bonds connecting adjacent residues in the sequence, and red dot lines represent non-sequence-adjacent H-H bonds.

Chain Environment – Place the entire (unfolded) sequence onto the lattice, and through a sequence of "pull" moves [3], fold the sequence into a final folded structure.

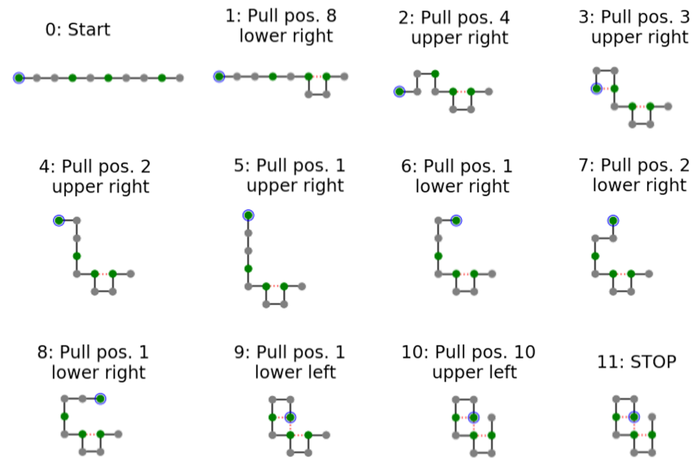


Figure 2: Folding a sequence in the 2D chain environment. The circled molecule represents position 1 of the sequence. Notice the Stop action; with no predetermined game length as in the lattice environment, under the chain environment, an agent must choose to stop moving.

Comparison

We compare our folding results with the results of traditional Monte Carlo simulations using Gillespie's acceptance rule, implemented in the chain environment. In the table below, we take the best result out of Policy/Value Iteration, DQN, and MCTS, and report it under DRL.

Length	Benchmark	Monte Carlo Sim.	DRL	AlphaGo Zero + pretraining
18	-4	-2	-2	-3
18	-8	-7	-6	-8
18	-9	-8	-7	-8
20	-9	-8	-6	-8
20	-10	-7	-8	-9
24	-9	-7	-6	-8
25	-8	-6	–	-7
36	-14	-10	–	-13

Table 2: 2D Free Energy Comparisons. A lower energy level means the algorithm found a structure closer to the ground state. The benchmark data and sequences are taken from `http://www.brown.edu/Research/Istrail_Lab/hp2dbenchmarks.html`.

Conclusion and Future Work

In this study, we implemented different deep RL algorithms to solve a challenging NP-Complete protein folding problem using the lattice environment, and tested the validity of different neural network architectures and algorithms. Within our knowledge, ours is the first study that implements DRL algorithms to the HP model in a 3-D lattice environment. We also implemented the chain environment with pull moves for simulating the dynamics of the protein folding process.
In future work, we plan to implement the DRL algorithms in the chain environment to evaluate the merits of using a physically motivated approach compared to the simpler, more popular lattice environment.

Acknowledgements

J. S. S. thanks Prof. Petr Šulc for useful discussions and for supplying a GPU node that was used for training throughout this work.

References

[1] B. Berger and T. Leighton. "Protein folding in the hydrophobic-hydrophilic (HP) model is NP-complete". In: *J. Comput. Biol.* 5.1 (1998), pp. 27–40.
[2] K. F. Lau and K. A. Dill. "A lattice statistical mechanics model of the conformational and sequence spaces of proteins". In: *Macromolecules* 22.10 (1989), pp. 3986–3997. eprint: `https://doi.org/10.1021/ma00200a030`.
[3] N. Lesh, M. Mitzenmacher, and S. Whitesides. "A Complete and Effective Move Set for Simplified Protein Folding". In: *Proceedings of the Seventh Annual International Conference on Research in Computational Molecular Biology*. RECOMB '03. Berlin, Germany: Association for Computing Machinery, 2003, pp. 188–195.