https://github.com/souadyoujilabadi/REMC-HP-model

Python implementation of the Replica Exchange Monte Carlo (REMC) algorithm for protein folding in the Hydrophobic-Polar (HP) model.

Souad YOUJIL ABADI

souad.youjil-abadi@etu.u-paris.fr

Master 2 : Biologie-Informatique

Projet court :

Jean-Christophe Gelly jean-christophe.gelly@u-paris.fr

Tatiana Galochkina tatiana.galochkina@u-paris.fr

1. INTRODUCTION

The Replica Exchange Monte Carlo (REMC) method has emerged as a powerful technique in simulating protein folding. A manifestation of this is illustrated in the paper "A replica exchange Monte Carlo algorithm for protein folding in the HP model" by Chris Thachuk et al.[1] This paper introduces the Monte Carlo algorithm which uses the Metropolis criterion to accept or reject proposed moves from one conformational configuration to another based on the energy difference between the states and the temperature of the system. The REMC also integrates the concept of temperature swapping where replicas exchange temperatures if the probability favors the swap.
In REMC, each replica of the system is simulated using Monte Carlo algorithm, and the replicas temperatures are periodically exchanged with each other . This exchange allows the system to explore a wider range of temperatures and overcome energy barriers that may be difficult to overcome at a single temperature.
As described in a referenced paper, the states of the replicas are effectively propagated from high temperatures to lower temperatures through replica exchanges this propagation facilitates the mixing of the Markov chain (a model describing a sequence of possible events in which the probability of each event depends only on the state attained in the previous even) by capitalizing on the rapid relaxation at elevated temperatures, where the entropy of the distribution is significantly high.[2]

In our implementation, the Monte Carlo simulation provides conformations which might be of minimal localized energy. Post this process, the temperatures of replicas are exchanged. We utilize both minimal and maximal temperature values and maintain uniform exchange rates in order to evenly distribute the temperatures among the replicas.

Within the in vivo environment, proteins adapt to guided configurations, reminiscent of an energetic funnel-shaped landscape with a distinctive global minimum. Although several factors can shape this landscape, our study predominantly focuses on one dimension: the interaction between hydrophobic residues. To this end, we've employed Dill’s HP (Hydrophobic-Polar) model. The primary objective within this methodology is to augment interactions among hydrophobic residues. Indeed, hydrophobic residues tend to cluster together in the interior of a protein to minimize their exposure to the solvent. This hydrophobic effect is a major driving force for protein folding.

2. MATERIAL AND METHODS

Protein Modeling:

FASTA sequences are the initial input, representing the primary structure of proteins. For computational feasibility, these sequences undergo simplification via Dill's HP model. In this model, amino acids are broadly classified into hydrophobic (H) and polar (P). The model's energy calculation focuses predominantly on hydrophobic residues, as they play a pivotal role in determining a protein's native conformation – the conformation with the lowest potential energy. Specifically, every interaction between two residues that are topological neighbors i.e. spatially adjacent hydrophobic amino acids not succeeding each other in the sequence contributes an energy value of -1.

Algorithm Implementation:

The implemented algorithm incorporates one of two different movement types during each Monte Carlo simulation:

> VSHD Moves: Comprising end, corner, and crankshaft movements.
> Pull Moves: As per the Chris Thachuk et al. findings[1], pull moves produce more favorable results in protein folding simulations.

For this reason, we introduce a weighting system that assigns a specific probability to pull moves versus VSHD. Pull moves are assigned a higher probability (0.4) compared to VSHD moves.

3. RESULTS AND DISCUSSION

The following results were generated using the seed value 1234543.

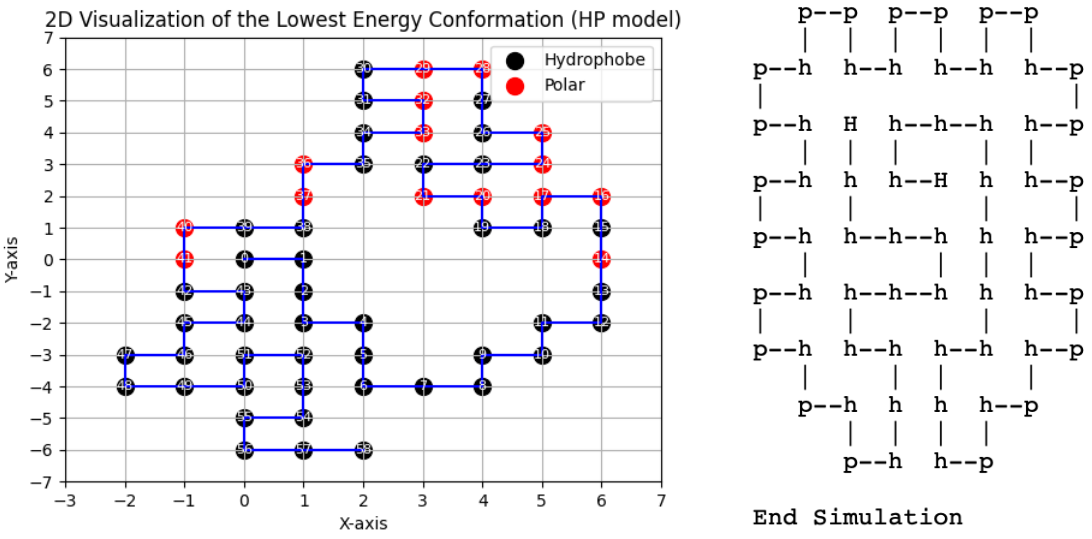

Fig 1. 2D visualization of the lowest energy conformation found by the REMC algorithm, our python implementation (left), Chris Thachuk et al. C++ implementation (right), for the same HP sequence. Energy: -19 (left) vs -42 (right).

Replica	1	2	3	4	5	Time before MP (sec)	Time after MP (sec)
MC	-19	-11	-10	-13	-16	-	-
REMC 10	-19	-12	-10	-12	-16	14.69	3.61

Table 1. Energies obtained for each of the five replicas after 500 iterations of MC (without exchanging temperatures) and 10 iterations of REMC algorithm. Time execution before and after parallelization using the multiprocessing (MP) Python's module.

The simulations were conducted under the following parameters:
MC number of iterations = 500, REMC number of iterations = 10, minimum and maximum temperatures at 160 and 220 respectively.
Similar results were obtained when the REMC number of iterations was set at 100.
The simulation is parallelized using the multiprocessing module to speed up the computation.
The parallelization is done at the level of the Monte Carlo search, i.e. each replica is simulated in parallel.

The performance of the program did not achieve the desired outcomes in terms of protein folding energies. The energies acquired were, at best, half of the expected values (-19 vs -42).
There is a suspicion of potential implementation errors in the "pull" movements, which would have otherwise enabled better mobility of the protein structures.

In terms of results reproducibility, since our program was implemented in Python, we cannot reproduce the same authors' result by using the same seed. Indeed, even if we seed both C++ and Python's random number generators with the same value, they'll produce different sequences of random numbers because they use different algorithms and have different implementations. However, for a correct implementation of the algorithm, we expect the program to converge to a global minimum.

REFERENCES

1. Thachuk, C., Shmygelska, A. & Hoos, H. H. A replica exchange Monte Carlo algorithm for protein folding in the HP model. *BMC Bioinformatics* **8**, 342 (2007).

2. Iba, Y. Extended Ensemble Monte Carlo. *Int. J. Mod. Phys. C* **12**, 623–656 (2001).