Université Paris Cité

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

## M2 BI : Biologie-Informatique

Souad Youjil Abadi

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

Based on Thachuk, et al. A replica exchange Monte Carlo algorithm for protein folding in the HP model. *BMC Bioinformatics* 8, 342 (2007).
https://doi.org/10.1186/1471-2105-8-342

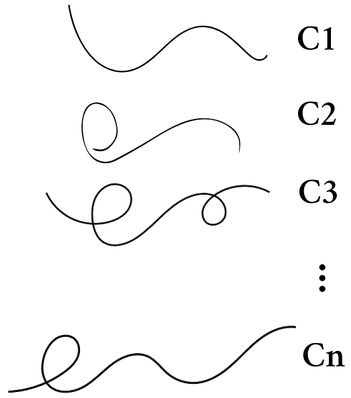
# INTRODUCTION

## Principle:

**Monte Carlo algorithm** : At each iteration it uses the Metropolis criterion to decide wether to accept or reject the new generated conformation.

V-M-G-L-H

**Protein sequence** → **random sampling**




→ **Potential energies evaluation**

**The Metropolis criterion states that :**

$$Pr[c \to c'] := \begin{cases} 1 & \text{if } \Delta E \leq 0, \quad \text{(if } \Delta E < 0\text{)} \\ e^{\frac{-\Delta E}{TK}} & \text{otherwise.} \quad \text{(if } \Delta E \Rightarrow 0\text{)} \end{cases}$$

ΔE : = E(c') - E(c) : Difference in energy between conformations c' and c

T : Temperature of the replica and K : 0.0019872 kcal/molK

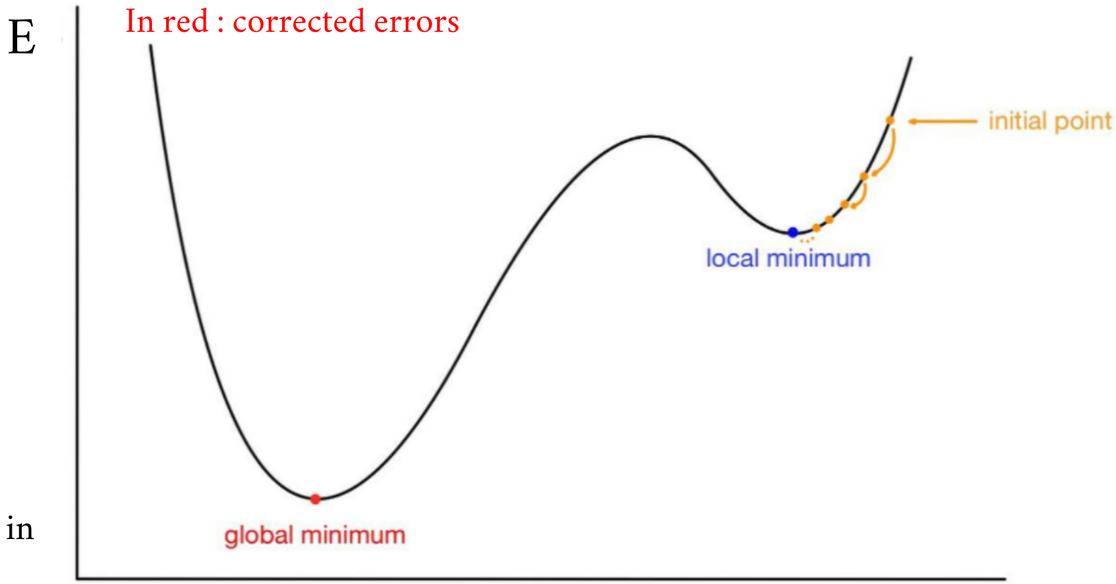
In red : corrected errors

**Large number of possible conformations**

**Influence of the temperature :**

$$\frac{1}{e^{\frac{\Delta E}{TK}}}$$

Higher temperatures ensures that the system explores the energy landscape and does not get trapped in local minima → REMC

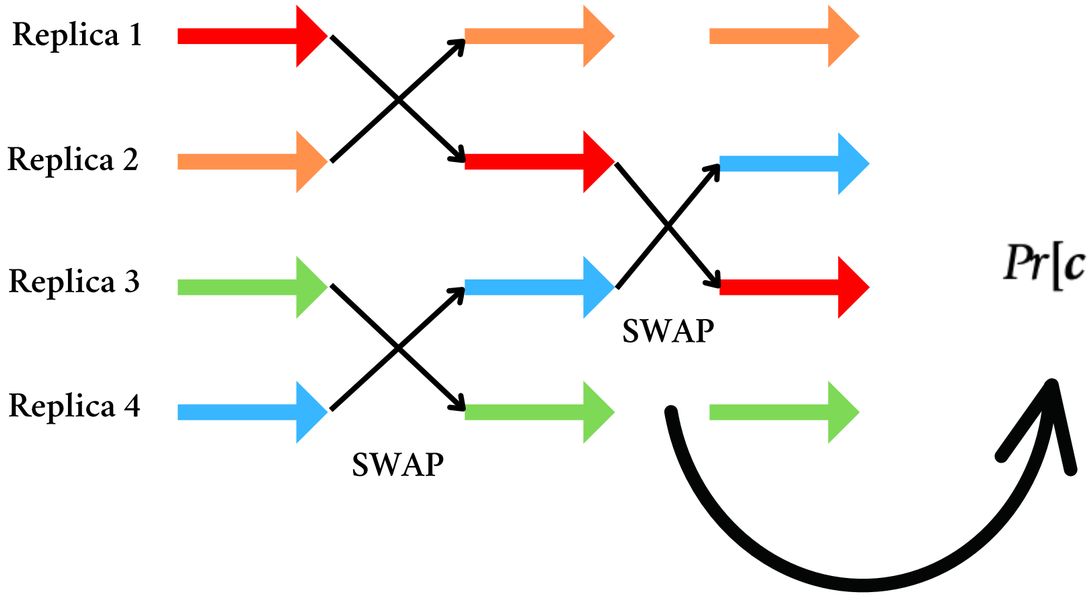


# INTRODUCTION

**Replica Exchange algorithm :**

Replicas are simulated at different temperatures, and exchanges between replicas are attempted using the Metropolis criterion.
This allows the system to escape from local minima and explore a wider range of conformations and energies.

**Simulltaneous MC replica exchange scheme**




$$Pr[c \rightarrow c'] := Pr[l(c_i) \leftrightarrow l(c_j)]$$

$$:= \begin{cases} 1 & \Delta \leq 0 \quad \text{(if } \Delta > 0\text{)} \\ e^{-\Delta} & \text{otherwise.} \quad \text{(if } \Delta < 0\text{)} \end{cases}$$

$$\Delta := (\beta_j - \beta_i)(E(c_i) - E(c_j)). \qquad \beta_i = \frac{1}{T_i K}$$

**Metropolis criterion to decide if the temperatures should be swapped**

K = 0.0001679010

# VSHD and Pull moves

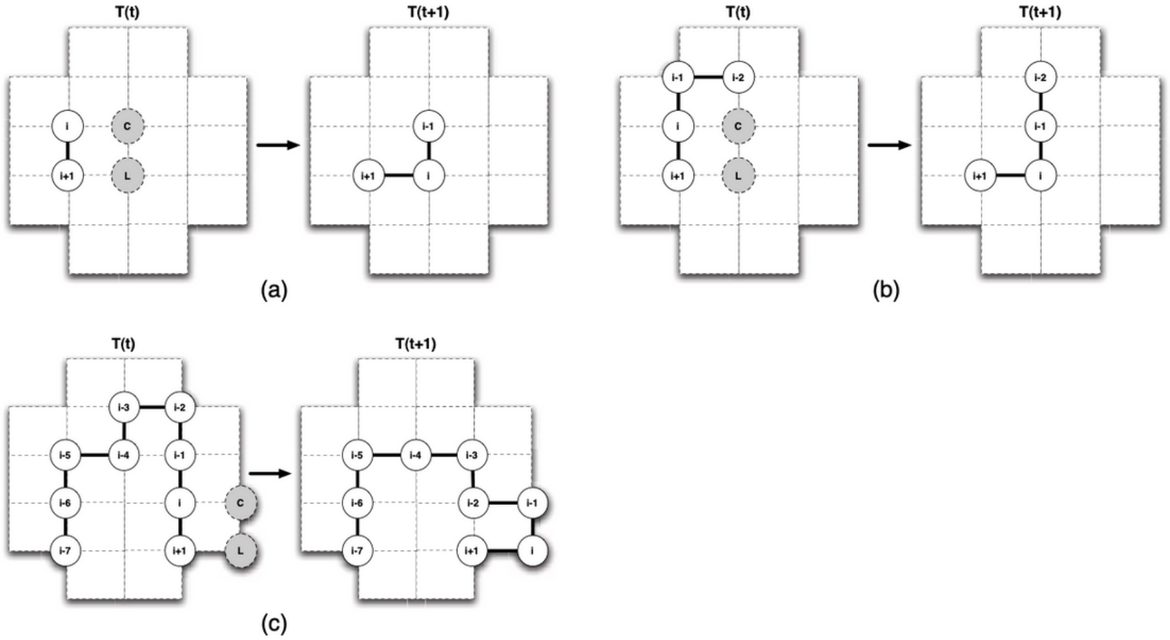



**Figure 3**
**Pull Moves.** This figure has been reproduced from [7] to illustrate the central idea behind this neighbourhood. In 3a, the simplest case where position *C* is occupied by residue *i* - 1 is shown. This move is equivalent to a corner move in the VSHD moveset. In 3b, residue *i* is moved to *L* and *i* - 1 to *C*. The chain is in a valid conformation and the move is finished. In 3c, residues *i* down to *i* - 3 must be pulled until a valid conformation is found.

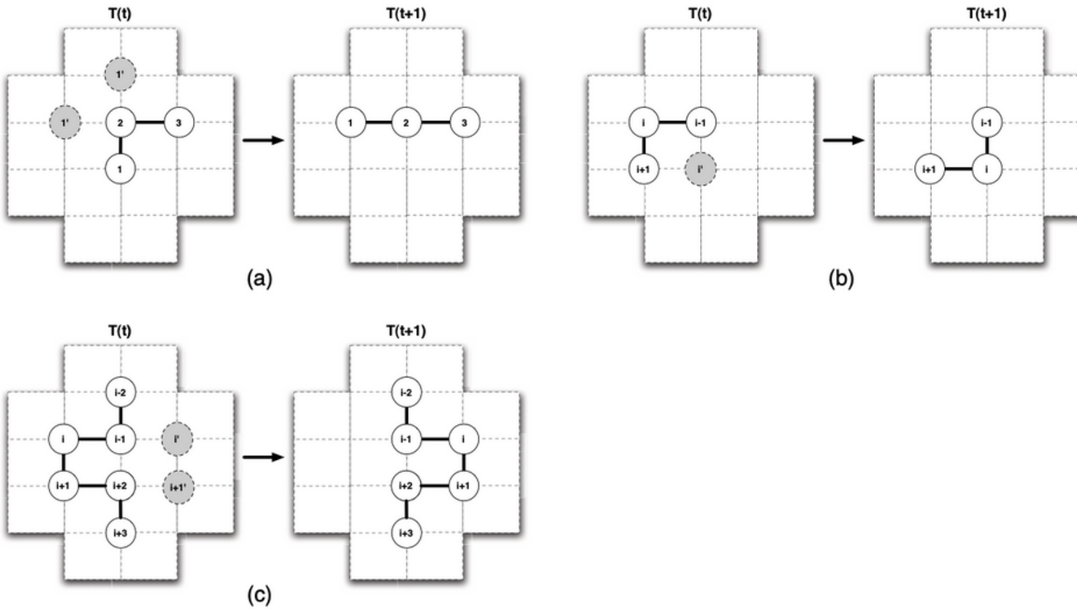



**Figure 2**
**VSHD Moves.**

Thachuk, et al. A replica exchange Monte Carlo algorithm for protein folding in the HP model.

# CLASS DEPENDENCIES

## Tools

read_fasta(filename)
monte_carlo(steps, conformation, pm_weight)
initialize_conformations(seq, num_rep, Tmin, Tmax, is_random)
swap_temperatures(conformations, flag)
visualize_2d_conformation(conformation)

Use

## Conformation

sequence: str
residues: list(Residue)
positions: list of lists
temperature: float
energy: int

sequence_to_residue_objects()
initialize_positions()
find_adjacent_empty_position(indx)
find_diagonal_empty_position(idx1, idx2)
move(adj_position, residue_index)
assign_initial_conformation(is_random)
end_move(residue_index)
corner_move(residue_index)
crankshaft_move(residue_index)
pull_move(residue_idx)
weighted_random_choice(choices, weights)
choose_movement_randomly(idx, pm_weight)
compute_energy()

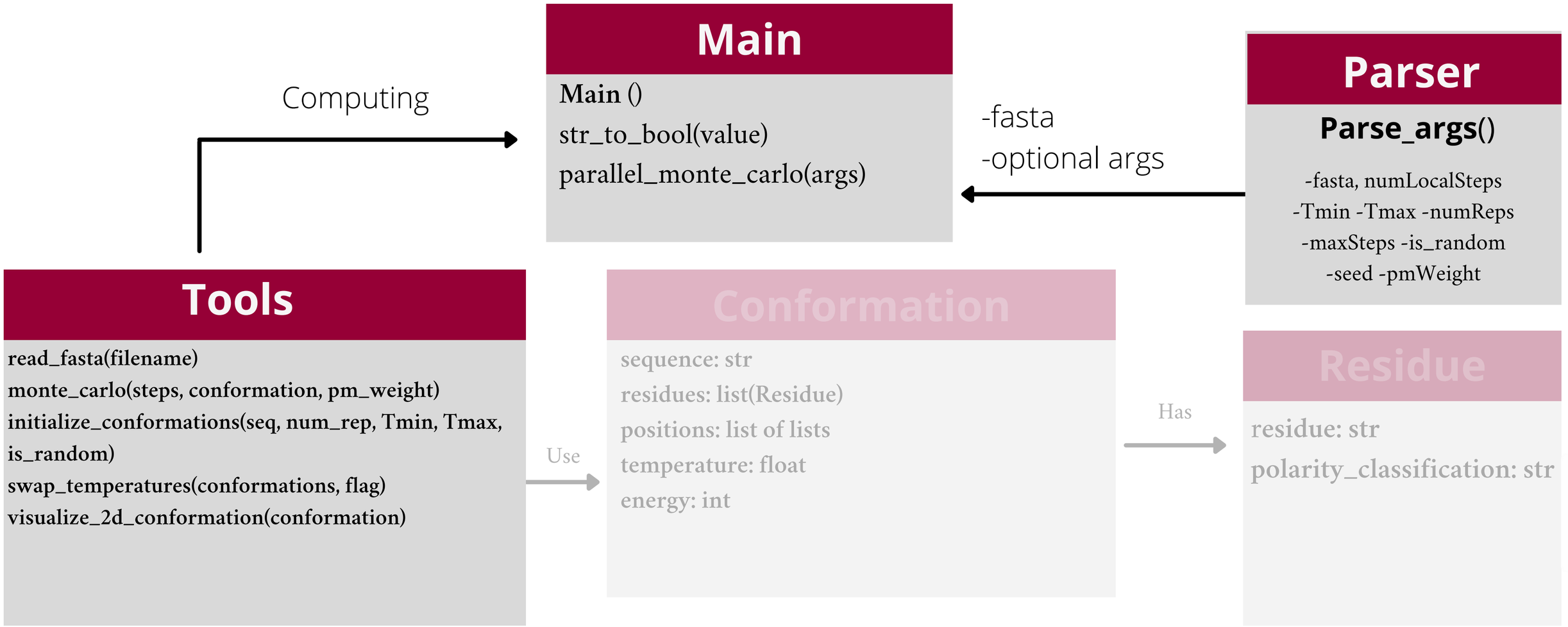
Has

## Residue

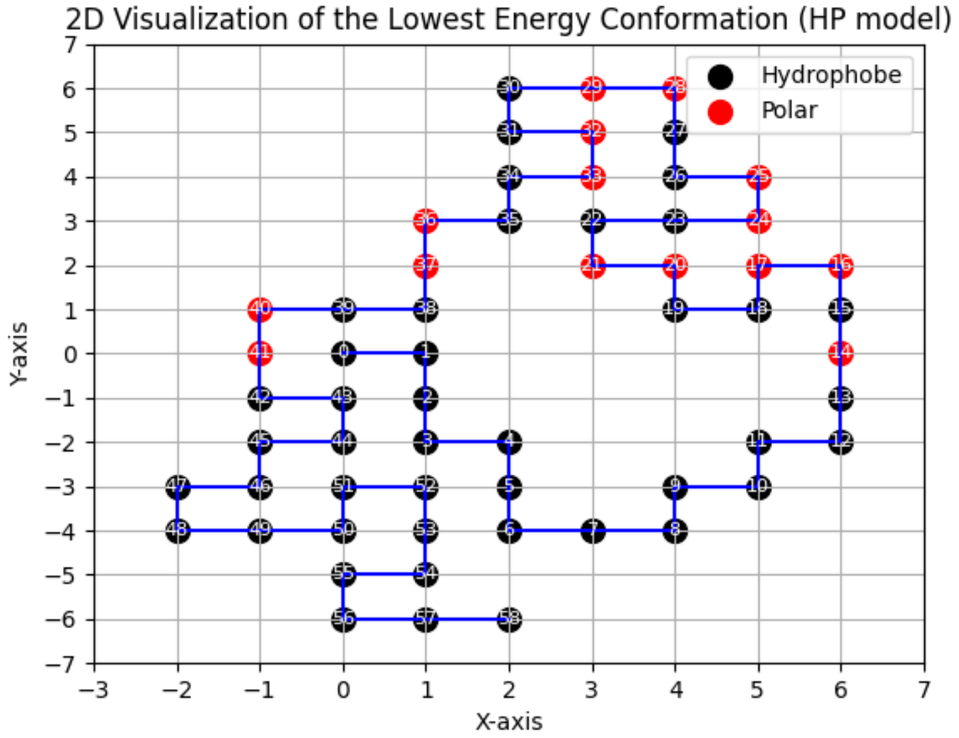
residue: str
polarity_classification: str

convert_to_hp()

# REMC PROGRAM (after parallelization using Pool)



# RESULTS




2D visualization of the lowest energy conformation found by the REMC algorithm
Python implementation

E: -19

```
    p--p  p--p  p--p
    |  |  |  |  |  |
p--h  h--h  h--h  h--p
|                    |
p--h  H  h--h--h  h--p
    |  |  |     |  |
p--h  h  h--H  h  h--p
|     |        |     |
p--h  h--h--h  h  h--p
    |        |  |  |
p--h  h--h--h  h  h--p
|     |        |     |
p--h  h--h  h--h  h--p
    |     |  |     |
    p--h  h  h  h--p
       |  |  |  |
       p--h  h--p

End Simulation
```

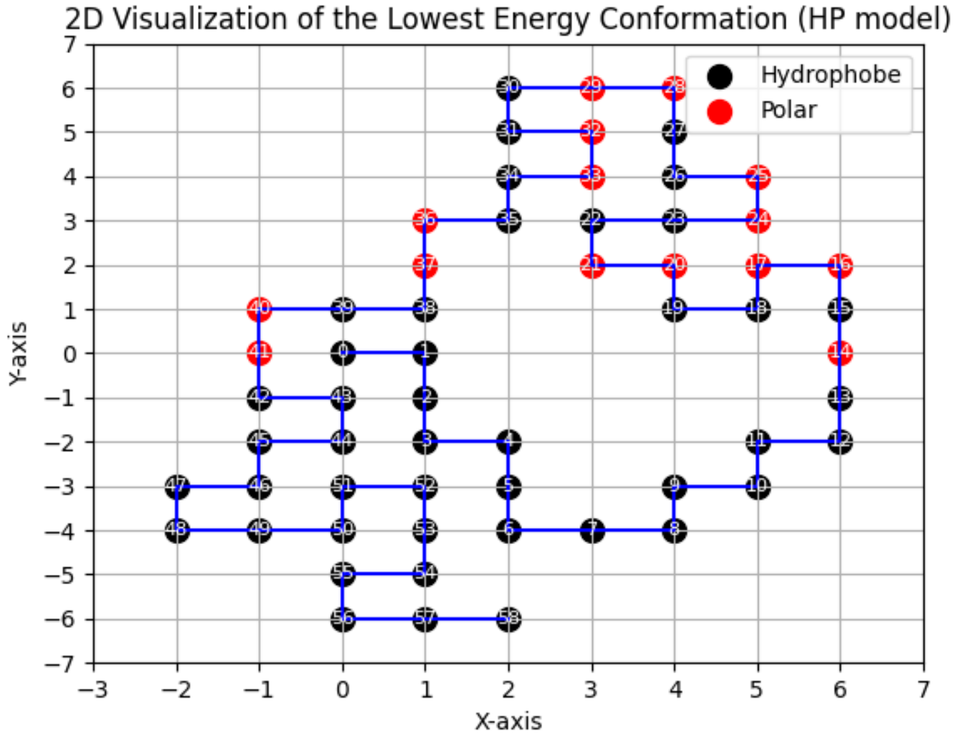
Chris Thachuk et al. C++ implementation

E: -42

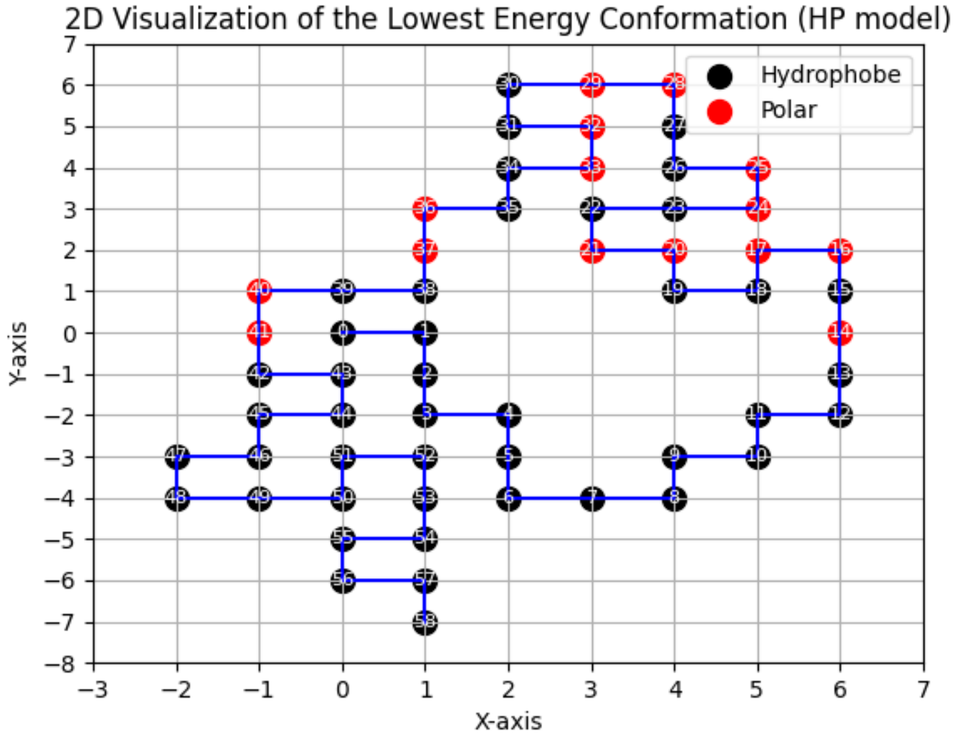
The performance of the program did not achieve the desired outcomes in terms of protein folding energies
The energies acquired were, at best, half of the expected values (-19 vs -42)
There is a suspicion of potential implementation errors in the pull movements

# RESULTS

The following results were generated using the same seed value 1234543




pmWeight = 0.4




pmWeight = 0

# RESULTS

| Rep | 1 | 2 | 3 | 4 | 5 | Time before MP (sec) | **Time after MP (sec)** |
|---|---|---|---|---|---|---|---|
| REMC 1 | -19 160.0 | -12 175.0 | -10 190.0 | -12 205.0 | -15 220.0 | 14.69 | 3.61 |
| REMC 2 | -19 175.0 | -12 160.0 | -10 205.0 | -12 190.0 | **-16 220.0** | | |
| REMC 3 | -19 175.0 | -12 205.0 | -10 160.0 | -12 220.0 | -16 190.0 | | |
| REMC 4 | -19 205.0 | -12 175.0 | -10 220.0 | **-13 160.0** | -16 190.0 | | |
| REMC 5 | -19 205.0 | -12 220.0 | -10 175.0 | -13 190.0 | -16 160.0 | | |
| REMC 10 | -19 160.0 | -12 190.0 | -10 175.0 | -13 220.0 | -16 205.0 | | |

Energy is initialised at 0, MC is successful, but REMC doesn't seem to have an effect to keep decreasing the potential energy

Energies obtained for each of the five replicas after 500 iterations of MC and 10 iterations of REMC algorithm. Similar results were obtained with 100 iterations. Time execution before and after parallelization using the multiprocessing (MP) Python's module.

# CONCLUSION & PERSPECTIVES

- Performances :
  Runs time have been successfully accelerated

  MC and REMC implementations were successful, however our pull move implantation seems to have an insignificant effect

- Possible improvements :
  Pull moves

**REFERENCES:**

*Thachuk, et al. A replica exchange Monte Carlo algorithm for protein folding in the HP model*

*Monte Carlo replica-exchange based ensemble docking of proteinconformations: Replica-exchange Ensemble Docking*

*Replica-Exchange Monte Carlo Method for Ar Fluid*

*Extended Ensemble Monte Carlo*