

GAによるHP folding問題の解法

1 概要

私は問 5(タンパク質のフォールディング問題) に取り組み、3 次元の HP 配列の最適化を遺伝的アルゴリズムを用いて解いた。2つの論文をもとに実装し、さらに独自の工夫として隔世遺伝を導入した。検証に用いたタンパク質は以下である。

- 配列長 20: (HP)2PH2PHP2HPH2P2HPH
- 配列長 48: P2HP2H2P2H2P5H10P6H2P2H2P2HP2H5

結果として配列長 20 の配列に関しては二次元、三次元ともに最適解を求めることができたが、配列長 48 に関しては一定の精度は得られたものの最適解を求めることはできなかった。

2 論文 [1]をもとにした基本実装

今回の基本実装に当たっては [1] を再現した。この論文では HP 配列の最適化問題を遺伝的アルゴリズムを用いて解いている。各遺伝子は HP 配列の姿勢を表す。HP 配列の姿勢は各アミノ酸の位置座標を配列にすることで表し、例えば図 1 であれば配列は $[[0, 0], [0, 1], \dots]$ となる。この時対称な形を可能な限り排除するため、配列の先頭を $[0, 0]$ (3次元の場合は $[0, 0, 0]$) に、その次の座標を $[1, 0]$ ($[1, 0, 0]$) に固定している (論文 2 を実装した際にミスをしたため、 $[1, 0]$ への固定はなくなってしまった)。適応度は水素結合の個数であり、水素結合が多いほど適応度が高くなる。適応度が最も高い配列が最適解となる。図 1 では 9 個の水素結合が存在する。各遺伝子は平面 (あるいは空間) 上での直鎖から始まり以下のアルゴリズムで全て突然変異され、交叉を経て次世代へと引き継がれる。

2.1 突然変異

モンテカルロ法と近い手法で突然変異を行う。各遺伝子について、変異前のエネルギーを E_1 、変異後のエネルギーを E_2 とすると、以下の確率で突然変異を行う。ただしエネルギーは $(\text{水素結合の数} + 1) \times (-1)$ であり、 c_m は世代を経るにつれて減少する定数である。水素結合の数に $+1$ しているのは、交叉の親選択の際に 0 除算を防ぐためである。

1. ランダムな位置を選ぶ。
2. その位置を中心としてランダムな角度で回転させる。二次元では 90 度、180 度、270 度のいずれか、三

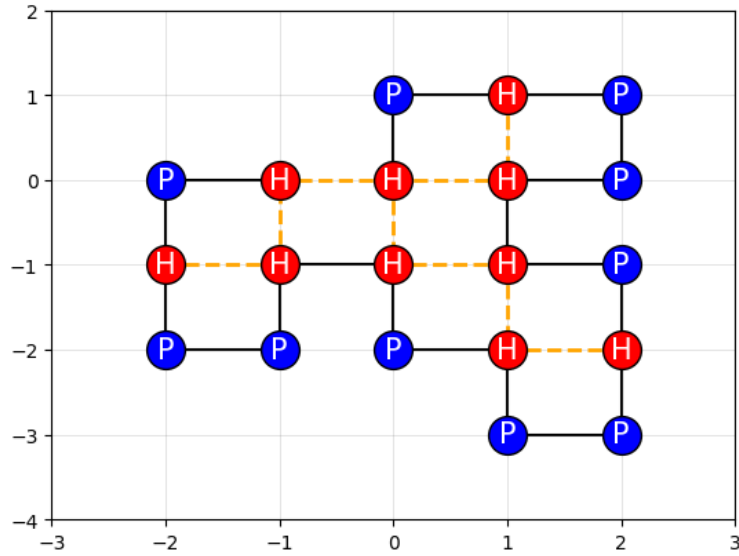


図 1: HP 配列の例

次元では回転軸が増え 5 つの角度から選択し回転させる。

3. 水素結合の数が増えた場合は採用する。そうでない場合でも局所最適にはまるのを防ぐため $\exp(\frac{E_1 - E_2}{c_m})$ の確率で採用する。
4. 3. で採用されなかった場合は他の角度で回転させる。
5. 1 4 を、変更が採用されるか全ての位置、角度を確認し終わるまで繰り返す。

ただし、複数のアミノ酸が同じ座標に存在することはできないので、このような構造は手順 2. の時点で直ちに棄却される。これは交叉でも同様である。

2.2 交叉

全ての遺伝子を突然変異させたのちに、最もエネルギーが安定な遺伝子を一体のみ新世代にそのまま追加する (elite selection)。その後交叉を行う。規定の population size に達するまで以下のアルゴリズムで新世代への追加を繰り返す。ただし E_{ave} は交叉前の親のエネルギーの平均であり、 c_c は世代を経るにつれて減少する定数である。

1. 2 つの遺伝子をランダムに選ぶ。この時より適合度の高い遺伝子を選ぶ確率が高くなるように選択確率を $p(S_i) = \frac{E_i}{\sum_j E_j}$ で定める。
2. それぞれの遺伝子を真ん中で付け替えて、新しい遺伝子を二つ作る。この際結合の角度はランダムに決定する。
3. 交換後の遺伝子が水素結合の数が増えた場合は採用する。そうでない場合は $\exp(\frac{E_{ave} - E_{new}}{c_c})$ の確率で採用する。
4. 3. で採用されなかった場合は他の角度で結合させる。
5. 1 4 を、変更が採用されるか全て確認し終わるまで繰り返す。
6. 交叉後の遺伝子を新世代に追加する。どの角度でも採用されなければ親二つをそのまま追加する。

2.3 パラメタ

population size, generation は 300 としている。突然変異は必ず起きる。突然変異の際の定数 c_m は 2.0 から始まり、5 世代ごとに 0.97 倍されていく。交叉の際の定数 c_c は 0.3 から始まり、5 世代ごとに 0.99 倍されていく。また、適応度が最も高い遺伝子は次世代にそのまま引き継がれる。

2.4 基本実装での精度

配列長 20 に関しては複数回試しても二次元 (最適は水素結合 9 個) では 8 個、三次元 (最適は 11) は 6,7 個までが限界であった。特に 3 次元については z 軸方向に十分曲がらないなどの問題があった。100-150 世代で収束することが多く、局所最適に陥りやすいことが分かる。文献にも遺伝的アルゴリズムは局所最適に陥りやすく三次元の解析に向かないと記されており、この結果は文献と一致している。配列長 48 に関しても二次元 (最適は 23) では最大 18、三次元 (最適は 29) では最大 16 個にとどまった。こちらでも三次元の方が収束が早いいためか、より芳しくない結果となった。

3 隔世遺伝の導入、論文 [2] の実装

より良い結果を得るため、まず隔世遺伝を導入した。具体的には交叉の際、新しい遺伝子を作るのに親の遺伝子を用いるのではなく、親の更に親の遺伝子を用いる確率を $p = 0.1$ で定め、それ以外の場合は従来通りに遺伝子を生成するようにした。これにより子世代の多様性が増し、局所最適に陥りにくくなると考えられた。また、論文 [2] をもとに実装を行った。論文 [1] では交叉の際必ず真ん中で結合を行っていたが、論文 [2] ではランダムな位置で結合を行うことで多様性を増すことができると記されていた。そこで交叉の際も突然変異同様、すべての位置で結合を試みた。

4 結果、考察

隔世遺伝及び論文 [2] の実装により、配列長 20 に関しては二次元、三次元ともに最適解を求めることができた。また、配列長 48 に関しては二次元 (最適は 23) では 21 個、三次元 (最適は 29) では 25 個の水素結合を持つ配列を求めることができた。これは基本実装よりも高い精度である。配列長 20 で求めた最適解を図 2 に示す。配列長 48 で求めた解を図 3 に示す。

配列長 20,48 について二次元、三次元共に五回ずつ試行した。表 1 に最適解の水素結合数と収束までにかかった世代数を示す。やはり安定はしないものの、基本実装よりも高い精度で最適解を求めることができた。十分な回数試行ができたとは言えないが、実験した範囲では配列長 48 のタンパク質の方が配列長 20 と比べて収束が明らかに早いため、配列長が長いほど収束が早い可能性がある。また、二次元より三次元の方がスコアが低い事例が複数あり、三次元の方が局所最適に収束しやすい可能性があるが、収束までの世代数に大きな差は見られなかった。基本実装では収束に 100-150 世代かかることが多かったのに対し最終的な実装では 50 100 世代で収束することが多かったが、これは一世代での計算回数が増えたため、収束が速くなった (局所最適に陥りやすくなった) とはいえない。以上より、隔世遺伝及び論文 [2] の実装により、配列長 20 のタンパク質で最適解を求めることができ、配列長 48 についてもより高い精度で解を求めることができた。

配列長/(最大結合数, 世代数)	2 次元	3 次元
20	(9,47),(9,88),(9,94),(9,179),(8,65)	(11,69),(11,81),(10,64),(9,144),(7,24)
48	(21,61),(21,62),(19,68),(19,36),(18,45)	(25,26),(25,29),(25,71),(22,52),(18,97)

表 1: 最適解の水素結合数と収束にかかった世代数

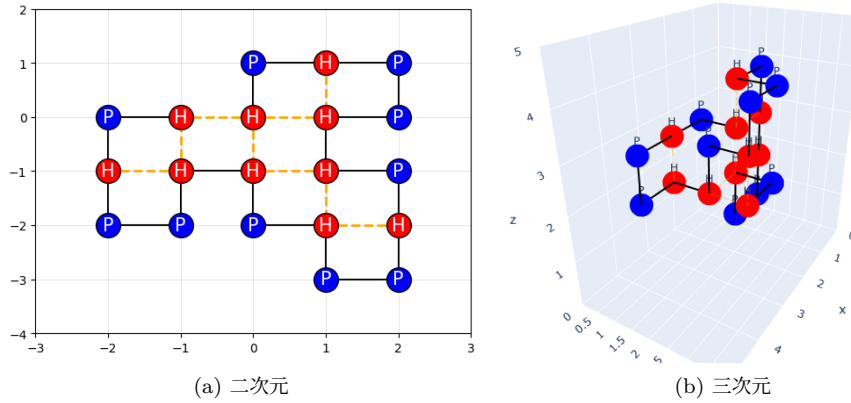


図 2: 配列長 20 の最適解

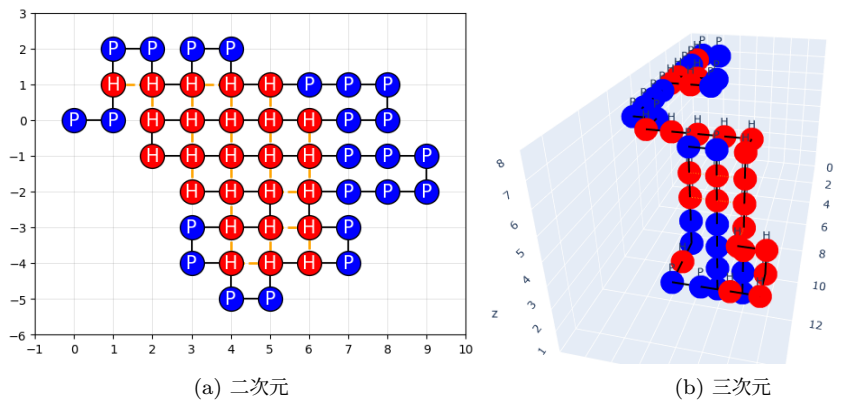


図 3: 配列長 48 の最適解

参考文献

- [1] Ron Unger and John Moult. Genetic algorithms for protein folding simulations. *Journal of molecular biology*, Vol. 231, No. 1, pp. 75–81, 1993.
- [2] Rainer König and Thomas Dandekar. Improving genetic algorithms for protein folding simulations by systematic crossover. *Biosystems*, Vol. 50, No. 1, pp. 17–25, 1999.