

H-032

自由エネルギーに基づいた塩基配列設計に関する研究

田中 文昭* 亀田 充史† 山本 雅人‡ 大内 東‡

Email: {fumiaki, kameda, masahito, ohuchi}@dna-comp.org

1 序論

DNAは相補対の形成によって、選択的、自律的に結合して二本鎖を形成する性質がある。この性質を利用して、ボトムアップなナノテクノロジーや分子マシンへの応用を試みるのがDNAコンピューティングの一つの目標である。これらの実現のためには、DNAの塩基配列を適切に設計し、意図した結合のみが形成される必要がある。

従来の塩基配列設計では、結合の安定性の指標として相補対の数（以下、 BP ）を用いていたが、これは粗い近似に過ぎないために、意図しない結合を防ぐことができないという問題があった。

そこで、本稿では、結合の安定性の指標として最小自由エネルギー（以下、 ΔG_{min} ）を用いた塩基配列設計法について述べる。特に、自身の相補鎖とのみ結合するような塩基配列集合の設計問題に関してアルゴリズムの提案および評価を行う。

2 提案手法

2.1 対象問題

制限時間 T 内で設計された塩基配列集合を $P = \{U_0, U_1, \dots, U_{n-1}\}$, $U_i \in P$ の相補配列を V_i とする。このとき、目的関数は $MAX|P|$ であり、 P 内の塩基配列が満たすべき制約条件は以下の2つである。

1. $\forall U_i \in P$ が V_i と結合した際の T_m が、 $T_m^- < T_m < T_m^+$ となる。

2. 以下の全ての組み合わせで $\Delta G_{min} > \Delta G_{min}^*$ となる。

$$(a) \langle U_i, U_j U_k \rangle (0 \leq i, j, k \leq n-1)$$

$$(b) \langle U_i, U_j V_k \rangle (0 \leq i, j, k \leq n-1), i \neq k$$

$$(c) \langle U_i, V_j U_k \rangle (0 \leq i, j, k \leq n-1), i \neq j$$

$$(d) \langle U_i, V_j V_k \rangle (0 \leq i, j, k \leq n-1), (i \neq j) \wedge (i \neq k)$$

ここで、 $\langle X, Y \rangle$ は塩基配列 X と Y の組み合わせを表し、 XY は X と Y をこの順で連結した塩基配列を表す。例えば、 $X = 5'\text{-GGCC-}3'$, $Y = 5'\text{-AATT-}3'$ とすると、 $\langle X, XY \rangle$ は、 $5'\text{-GGCC-}3'$ と $5'\text{-GGCCAATT-}3'$ との組み合わせを表す。

また、本稿では $\forall U_i (i \in P)$ の配列長を l と定義し、 T_m^- , T_m^+ をそれぞれ $(T_m^{ave} - 1.5)$, $(T_m^{ave} + 1.5)$ とする。ここで、 T_m^{ave} は長さ l の 10,000 本の塩基配列をランダムに出力した際の T_m の平均値である。

2.2 アルゴリズムの概要

塩基配列設計は、組合せ最適化問題の一種であるが、 ΔG_{min} 計算は計算コストがかかるため、限られた時間では多数の解候補を探索できない。従って、GA や SA 等のメタヒューリスティクスを用いて大域的な探索を行うよりも、局所的な探索が適していると考えられる。そこで本アルゴリズムでは generate-and-test 法を採用している [1]。Generate-and-test 法とは、ランダムに解候補を生成し、解候補が全ての制約条件を満たした場合にのみ解集合に加えるという手続きを繰り返すアルゴリズムである。

アルゴリズムのフローチャートを図1に示す。ここでは、3つのフィルターを用いて制約条件の判定を行う。 T_m フィルターと ΔG_{min} フィルターは、それぞれ T_m と ΔG_{min} に関する制約の判定を行うために必須であり、 ΔG_{gre} フィルターは計算時間を削減するために用いるフィルターである。 ΔG_{min} フィルターで弾くべき候補の塩基配列を ΔG_{gre} フィルターで前もって弾くことによって、 ΔG_{min} 計算を省略することができるため、全体の計算時間の削減が期待できる。また、フィルターは計算量の昇順に並べられているため、計算量の大きいフィルターでの判定回数を抑えることができ、これによ

*北海道大学大学院工学研究科

†CREST, Japan Science and Technology Corporation

‡北海道大学大学院情報科学研究科

り全体の計算時間を削減できる。ここで、すでに設計された塩基配列の数を n_c とすると、 $n_c + 1$ 番目の候補となる塩基配列を判定する際の各フィルターの計算量は順に $O(l), O(n_c^2 l^2), O(n_c^3 l^3)$ である。

T_m と ΔG_{min} に関しては、従来研究と同様に nearest-neighbor model と dynamic programming を用いて算出している [2]。

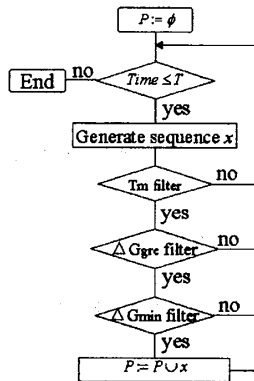


図1: アルゴリズムのフローチャート。

2.3 ΔG_{gre} フィルター

貪欲法により計算された ΔG_{min} を ΔG_{gre} と表記する。 ΔG_{gre} フィルターでは、 ΔG_{gre} を計算し、 $\Delta G_{gre} \leq \Delta G_{min}^*$ となる塩基配列を弾く。dynamic programming で ΔG_{min} を計算する場合と比較して、貪欲法による ΔG_{gre} 計算は計算量が小さい。さらに、 $\Delta G_{min} \leq \Delta G_{gre}$ が成り立つことに注意すると、 $\Delta G_{gre} \leq \Delta G_{min}^*$ ならば、 $\Delta G_{min} \leq \Delta G_{min}^*$ が成り立つことがわかる。従って、 ΔG_{gre} フィルターで弾かれた塩基配列は、 ΔG_{min} フィルターにも必ず弾かれる塩基配列であるため、制約条件を満たす塩基配列を誤って排除することがない。

貪欲法による ΔG_{gre} 計算の模式図を図2に示す。まず、二本の塩基配列を互いにシフトさせていき、連続相補となる箇所の ΔG を計算する。その中で、最小となる ΔG_0 を探し、その箇所の相補対を固定する。次に、固定した相補対の左右の塩基配列についても同様にそれぞれシフトさせていき、 ΔG が最小となるような連続相補の箇所を探し出し、相補対を固定する。以下、上記の手続きを繰り返すことによって相補対を固定していき、最終的に得られた構造の ΔG を ΔG_{gre} とする。詳細は参考文献 [2] を参照されたい。

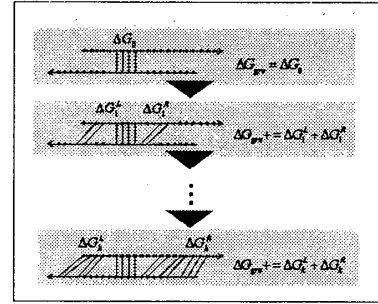


図2: 貪欲法による ΔG_{min} 計算の模式図。

3 実験

3.1 化学実験による BP と ΔG_{min} との比較

ΔG_{min} に基づく塩基配列設計の妥当性を検証するため、電気泳動を用いて二本の DNA が結合する際の BP および ΔG_{min} の値を調査した [3]。

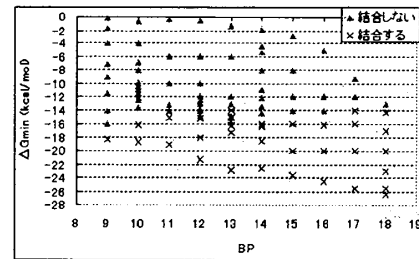


図3: DNA 間の結合における BP と ΔG_{min} との比較。

結果を図3に示す。結果から、結合する DNA 対と結合しない DNA 対とを分離する能力は、BP よりも ΔG_{min} の方が高く、 $\Delta G_{min} = -14.0$ を閾値としておおよそ分離できることがわかる。従って、自身の相補鎖とのみ結合する塩基配列集合を設計することを考えた場合、従来の BP を用いた手法では意図しない結合を形成する塩基配列を排除することが困難であり、 ΔG_{min} を用いた方がより効率的にそのような塩基配列を排除できることがわかる。

3.2 計算機実験による ΔG_{gre} フィルターの有効性の検証

ΔG_{gre} フィルターの有効性を検証するため、 $T = 10$ 時間として ΔG_{gre} フィルターを用いないアルゴリズムとの比較を行った。

まず、DNA コンピューティングでよく使用される、

配列長が 20mer の塩基配列に関して調査を行った。 ΔG_{min}^* の値は前節での結果を踏まえて、 $\Delta G_{min}^* = -14.0, -13.0, -12.0, -11.0$ の 4 パターンとしている。結果を図 4 に示す。この図から、 ΔG_{gre} フィルターを用いたアルゴリズムの方が 10~20 本多い塩基配列集合を設計することに成功しており、全ての閾値においてほぼ同様のパフォーマンスを示している。従って、配列長が 20mer の場合には、 ΔG_{min}^* が実用的な範囲で設定されていれば、 ΔG_{gre} フィルターは有効に機能することが示された。

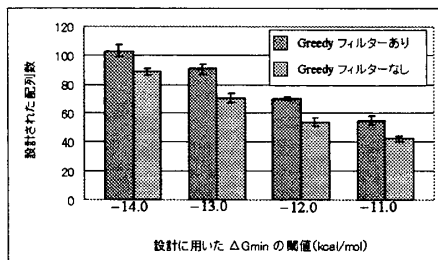


図 4: 配列長が 20mer の場合の ΔG_{gre} フィルターの有効性。計算時間は 10 時間。各値はシードを変更して 5 試行繰り返した際の平均値であり、エラーバーは標準偏差を示している。

次に、配列長と ΔG_{gre} フィルターの有効性との関係について調査した。配列長は 20, 30, 40, 50mer の 4 パターンとし、 $\Delta G_{min}^* = -14.0$ としている。結果を図 5 に示す。結果から、配列長が長くなるに従って ΔG_{gre} フィルターが有効に機能しないことがわかる。この理由は、配列長が長くなるほど ΔG_{gre} と ΔG_{min} との差は大きくなるため、 ΔG_{gre} フィルターで効率的に塩基配列を排除することができないためと考えられる。実際、 ΔG_{gre} フィルターが正しく塩基配列を排除できた割合 $R_{gre} = N_{gre}/N_{min}$ を調査すると、配列長が 20, 30, 40, 50mer となるに従い、 $R_{gre} = 0.89, 0.90, 0.86, 0.77$ となった。ここで、 N_{min} は ΔG_{gre} フィルターが存在しない場合に ΔG_{min} フィルターで弾かれる塩基配列数であり、 N_{gre} は N_{min} のうち、 ΔG_{gre} フィルターで弾くことができた塩基配列数である。結果から、 ΔG_{gre} フィルターが有効に機能するのは、~50mer 程度の配列長の場合であることがわかる。DNA コンピューティングでは 15~40mer の塩基配列が用いられる事が多いため、 ΔG_{gre} フィルターは実際の塩基配列設計で有用であるといえる。

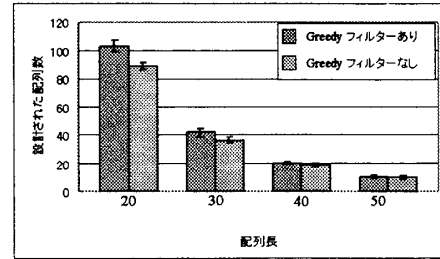


図 5: 配列長と ΔG_{gre} フィルターの有効性との関係。 ΔG_{min} の閾値は -14.0 kcal/mol。計算時間は 10 時間。各値はシードを変更して 5 試行繰り返した際の平均値であり、エラーバーは標準偏差を示している。

4 結論

本稿では、DNA が結合する際の安定性の指標として ΔG_{min} を用いた塩基配列設計手法の提案、および評価について述べた。化学実験の結果より、従来の BP を用いた手法よりも ΔG_{min} が塩基配列設計に適していることが確認できた。また、計算機実験の結果より、 ΔG_{gre} フィルターが計算時間削減に有効であることが示された。従って、提案アルゴリズムは、DNA コンピューティングでの塩基配列設計に有用であると考えられる。

参考文献

- [1] M. Arita, A. Nishikawa, M. Hagiya, K. Komiya, H. Gouzu, and K. Sakamoto. Improving sequence design for dna computing. In *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO-2000)*, pages 875–882, 2000.
- [2] Fumiaki Tanaka, Atsushi Kameda, Masahito Yamamoto, and Azuma Ohuchi. Design of nucleic acid sequences for DNA computing based on a thermodynamic approach. *Nucleic Acids Res*, 33(3):903–11, 2005.
- [3] Fumiaki Tanaka, Atsushi Kameda, Masahito Yamamoto, and Azuma Ohuchi. Specificity of hybridization between DNA sequences based on free energy. In *Proceedings of 11th International Workshop on DNA-Based Computers*, 2005.