アルゴリズム 59－3
（1997. 11. 28）

PCR実験におけるプライマー数の最小化の計算量的性質と解法

土井 晃一郎　今井 浩

東京大学大学院理学系研究科情報科学専攻
〒113 東京都文京区本郷 7-3-1

ポリメラーゼ連鎖反応（PCR）実験において、よいプライマーを選択することは生物学において大変重要である。今までに作られてきたプライマー選択に対するプログラムは1つのDNA配列に対して1組のプライマーを選択するものであった。我々は、プライマーが複数の箇所につくことを許すことにより全てのDNA配列を増幅するような少ないプライマーの集合を選択することを考える。このプライマー選択問題を定式化して、その問題の計算量的性質を解析し、単純な定式化に対する貪欲アルゴリズムを実装して、酵母の配列に対して適用する。

Complexity properties and algorithms for minimizing number of primers in PCR experiments

Koichiro Doi　Hiroshi Imai

Department of Information Science, Faculty of Science, University of Tokyo
7–3–1 Hongo, Bunkyo-ku, Tokyo 113, Japan

Selecting a good collection of primers is very important for polymerase chain reaction (PCR) experiments in biology. Most existing algorithms for primer selection are concerned with computing a primer pair for each DNA sequence. We consider selecting a small set of primers which amplify all DNA sequences by allowing each primer matches several places. We formulate the primer selection problem and analyze the complexity of these problems and we implement the greedy algorithm for the simple formulation and apply it to real yeast data.

1　はじめに

PCR(ポリメラーゼ連鎖反応) という生物のDNAの解析のときに有用な実験がある。この実験は、プライマーと呼ばれる方向性のある配列をDNA配列のマッチする部分にくっつけ、正しく向きあった2つのプライマーにはさまれた部分のコピーをたくさん作るものである。（図1参照）このPCRにおいて使うプライマーの設計は大変重要であり、[11, 15]では、1つだけ増幅を得るような1組のプライマーを計算していて、[12]では、複数の増幅を得るプライマーを1つのDNA配列に対して1組のプライマーを計算している。

ここでは、複数のDNA配列に対して複数の増幅を得るときの最小の数のプライマーを求めることを考える。プライマーの数を少なくすればPCR実験のコストが減少するからである。現在は

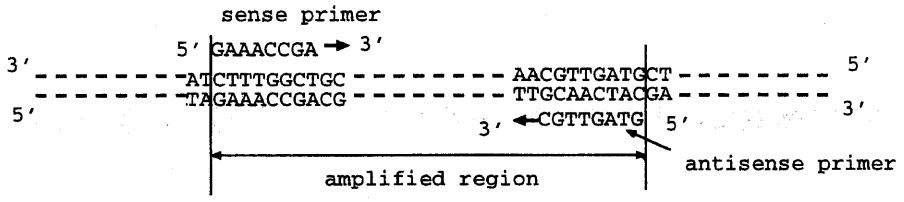

図 1: DNA 配列とプライマー

全DNA 配列が知られているものがあり、そのようなDNA 配列に対してはこのような計算が出来る。(全DNA 配列が知られているものには酵母、大腸菌などがある。) [13, 14] ではプライマーの向きについては考えずに解析、実験をしている。しかし、プライマーの向きは重要であり、本稿ではそれを考慮した解析、実験について述べる。

2 問題の定式化

問題の解析をする際には、A,T,G,C といった 4 つの塩基によってDNA が encoding されているといったことは考えないで、文字列 (DNA 配列) とそれにマッチする箇所としてプライマーの集合が与えられたとする。

入力: m_j $(1 \leq j \leq n)$ の長さの n 本の文字列 (DNA 配列)
集合族 $S = \{S_1, S_2, ..., S_l\} \cdots$ 集合 1 つがプライマーに対応、
$(j, p, r/l) \in S_i$ $(1 \leq j \leq n, 1 \leq p \leq m) \cdots j$ 本目の文字列の p 番目の要素に右/左 方向にマッチする。

まず、一番単純にどのDNA 配列からも増幅が得られるという条件で定式化すると以下のようになる。

定義 1 (プライマー選択問題) どの文字列も右、左 2 つのプライマーではさまれる部分があるような最小の数のプライマーを求める問題。

$$minimize \quad |S'|$$
$$s.t. \quad \forall j \ \exists S_i, S_{i'} \in S', \ (j, p_1, \mathrm{r}) \in S_i, (j, p_2, \mathrm{l}) \in S_{i'}, \ p_2 - p_1 + 1 > 0.$$

さて、ここからは増幅配列の長さというものに着目していく。そのためには以下の生物的制約を考えなくてはいけません。

定義 2 (増幅制約) プライマーによる増幅は、間に他のプライマーがマッチしない正しく向きあった 2 つのプライマーに対してのみ行なわれる。

図 2 で説明すると、増幅制約とは S_1,S_3 間の増幅は作られずに、S_1,S_2 間の増幅ばかりが作られてしまうことである。

最初の定式化ではプライマー自身の長さを考えていないし、電気泳動で識別できるようにするためには増幅配列がある程度以上の長さでなくてはいけない。このことを考慮して、まず、増幅制約がないと考えてあるプライマーの間隔がある一定以上の長さであるという条件で定式化すると以下のようになる。

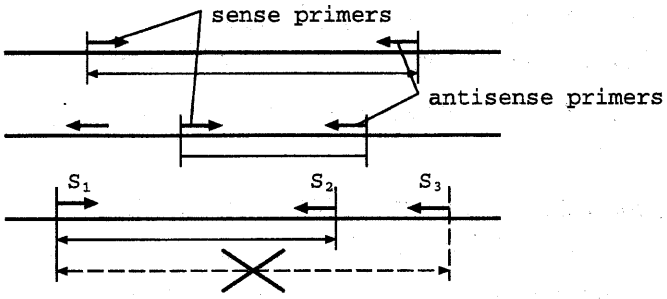

図 2: 増幅制約

定義 3 (間隔制約付きプライマー選択問題) 全ての *DNA* 配列に対して右、左2つのプライマーではさまれる部分の長さが k 以上あるような最小の数のプライマーを求める問題。を求める問題。*(*プライマー選択問題の定義の $p_2 - p_1 + 1 > 0$ を $p_2 - p_1 + 1 \geq k$ に変えたもの*)*

増幅制約を考えた問題を、最小制約付きプライマー選択問題とする。

定義 4 (最小制約付きプライマー選択問題) 全ての *DNA* 配列に対して長さが k 以上である増幅が存在するような最小の数のプライマーを求める問題。

$$
\begin{aligned}
&\textit{minimize} \quad |S'| \\
&\quad s.t. \quad \forall j \ \exists S_i, S_{i'} \in S',\ \exists (j, p_1, \mathrm{r}) \in S_i, (j, p_2, \mathrm{l}) \in S_{i'},\ p_2 - p_1 + 1 \geq k \\
&\qquad \forall p' \ (p_1 \leq p' \leq p_2)\ \forall S_{i''} \in S' \ (j, p', \mathrm{r}), (j, p', \mathrm{l}) \notin S_{i''} \text{（増幅制約）}
\end{aligned}
$$

電気泳動により長さで DNA 配列の有無を判定するには、各DNA 配列に対して長さが異なる増幅配列が存在しなくてはいけません。このことを定式化すると以下のようになる。

定義 5 (相違制約付きプライマー選択問題) 全ての *DNA* 配列に対して増幅が存在し、その増幅の長さが異なっているものが各 *DNA* 配列に対して存在するような最小の数のプライマーを求める問題。

$$
\begin{aligned}
&\textit{minimize} \quad |S'| \\
&\quad s.t. \quad \forall j \ \exists S_i, S_{i'} \in S',\ \exists (j, p_1, \mathrm{r}) \in S_i, (j, p_2, \mathrm{l}) \in S_{i'},\ p_2 - p_1 + 1 > 0 \\
&\qquad \forall p' \ (p_1 \leq p' \leq p_2)\ \forall S_{i''} \in S' \ (j, p', \mathrm{r}), (j, p', \mathrm{l}) \notin S_{i''} \text{（増幅制約）} \\
&\qquad \forall j' \ (\neq j) \text{ 増幅制約を満たす } (j', p_3, \mathrm{r}), (j', p_4, \mathrm{l}) \text{ は } p_4 - p_3 + 1 \neq p_2 - p_1 + 1.
\end{aligned}
$$

3 問題の計算量的性質

前章で定式化した問題がどのような計算量的性質をもつかこの章では述べる。これらの問題がNP hard の問題であることはすぐいえるので、近似の性質について見ていく。この章の定理は [1, 2, 6, 7, 9, 16, 17] の近似に関する結果に基づいて土井 [4] が証明している。

定理 1 $NP \subset TIME(n^{O(\log\log n)})$ でないならば任意の ϵ に対して、プライマー選択問題を解く $(1-\epsilon)\ln n$ 以内の近似比の多項式時間近似アルゴリズムは存在しない。

定理 2 $P \neq NP$ ならばプライマー選択問題には、どんな定数の近似比の多項式時間近似アルゴリズムは存在しない。

この定理は、集合被覆問題からの変換により証明出来ます。逆にどのような近似アルゴリズムが存在するかというと、以下のような定理が成り立ちます。

定理 3 プライマー選択問題には、$O(\log mn)$ の近似比の多項式時間アルゴリズムが存在する。$(m = \max(m_1, ..., m_n))$

これは、集合被覆問題に変換してその貪欲アルゴリズムを適用することによって証明出来ます。この貪欲アルゴリズムについては 4 章で考えます。

定理 4 $NP \subset TIME(n^{O(\log\log n)})$ でないならば任意の ϵ に対して、間隔制約、最小制約付きプライマー選択問題を解く $(1-\epsilon)\ln n$ 以内の近似比の多項式時間近似アルゴリズムは存在しない。

定理 5 $P \neq NP$ ならば間隔制約、最小制約付きプライマー選択問題には、どんな定数の近似比の多項式時間近似アルゴリズムは存在しない。

定理 6 間隔制約付きプライマー選択問題には、$O(\log mn)$ の近似比の多項式時間アルゴリズムが存在する。$(m = \max(m_1, ..., m_n))$

間隔制約をプライマー選択問題に付加したとしても近似の性質は変わりません。しかし、増幅制約を考えた最小制約付きプライマー選択問題となると定理 6 のようなアルゴリズムが存在するかどうかは分かっていない。

それに対して相違制約付きプライマー選択問題は以下のような定理が成り立っています。

定理 7 $NP \neq coRP$ であるならば、相違制約付きプライマー選択問題は任意の $\epsilon > 0$ に対して、近似比 $n^{\frac{1}{2}-\epsilon}$ 以下の多項式時間近似アルゴリズムは存在しない。

定理 8 $P \neq NP$ であるならば、相違制約付き問題 4 は任意の $\epsilon > 0$ に対して、近似比 $n^{\frac{1}{8}-\epsilon}$ 以下の多項式時間近似アルゴリズムは存在しない。

この 2 つの定理はグラフ点彩色問題からの変換により証明することが出来ます。これらの定理により、近似の面からいってプライマー選択問題と相違制約付きプライマー選択問題とは問題の難しさに差があることが分かる。

4 アルゴリズム

前章までの結果を考慮して、このプライマーを選択する問題に対するアルゴリズムを考える。解きたい問題は最小制約、相違制約の双方が付加されたプライマー選択問題である。しかし、前章により相違制約付きプライマー選択問題はプライマー選択問題よりも近似という面に置いて本質的に難しいということを示している。また、大規模な問題に適用していくことを考えているのであまり複雑なことは出来ない。土井、今井 [5] では、定理 3 の証明に使った集合被覆問題に対する貪欲アルゴリズムを拡張していくことによって最小制約や相違制約を満たす解を得ることを考えている。土井、今井 [5] でのアルゴリズムとその実験について簡単にこの章と次章で述べる。

4.1 プライマー選択問題に対する貪欲アルゴリズム

DNA 配列中の文字 (A,T,G,C) を台集合の要素、プライマーを部分集合と考えることによってプライマー選択問題は集合被覆問題と考えることが出来る。プライマーは図 3 のようにそのプライマーがマッチする位置から、その向きにある要素を全てカバーするものと考える。

このとき、まだカバーされていない要素をもっとも多くカバーするようなプライマーを順に選択していくという集合被覆問題に対する貪欲アルゴリズムを考える。このアルゴリズムは定理 3 の証明で使ったものであり、近似精度が保証されている。これを基本のアルゴリズムとし、これ以降はこのアルゴリズムを拡張していくことを考える。アルゴリズムは

1. DNA 配列をスキャンして、プライマーの候補となりうる配列毎にマッチする位置をまとめる。
2. まだカバーされていない要素をもっとも多くカバーするようなプライマーを順に選択していく。
3. すべての要素がカバーされるか、これ以上カバー出来なくなれば終了する。

という流れでなる。(これ以降の拡張によりカバーできない要素が出てくる場合があるので 3 はこのようにしておく。)

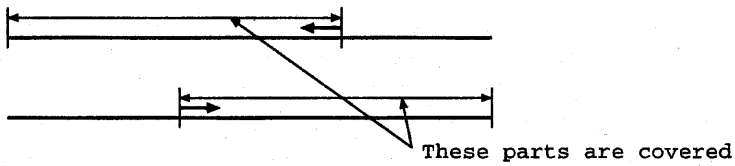

図 3: プライマーのカバーする要素

4.2 最小制約のための拡張

ここでは、アルゴリズムに対して 2 種類の拡張を考える。

1 つは定理 6 の証明に使われている間隔制約付きプライマー選択問題に対するアルゴリズムである。これは図 4 のように右向きにマッチするプライマーのカバーする要素を k だけずらしたところから考えるようにもとのアルゴリズムから変更したものである。そうすれば各々の DNA 配列に対して k 以上離れた位置にマッチするプライマーが存在するような解が出てくるが、増幅制約を考えると k 以上の増幅が全 DNA 配列に得られるかどうかは分からない。この拡張を間隔制約拡張とし、k をパラメータとする。

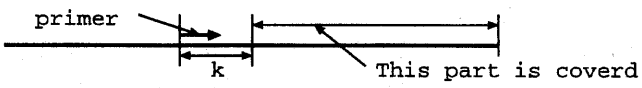

図 4: プライマーのカバーする要素の変更

そこで、もう 1 つ増幅制約を考えて、ある定数以上の増幅が得られるようなアルゴリズムを考

表 1: 最小制約のための拡張の効果

k (間隔制約拡張のパラメータ)	0	50	50	100	100
t (最小制約拡張のパラメータ)	0	0	50	0	100
プライマーの数	106	117	186	129	183
プライマーのマッチする位置の数	3965	4348	3274	4601	2561
増幅配列の数	1053	1154	834	1224	602
50 未満の長さの増幅配列の数	465	513	0	535	0
長さにより区別可能な増幅配列の数	170	143	174	157	206
長さにより区別可能な DNA 配列の数	135	121	146	127	167
50 から 500 の長さの増幅で区別可能な DNA 配列の数	111	95	115	118	120
増幅されない配列の数	0	0	15	0	98
50 以下の増幅しか持たない DNA 配列の数	124	111	0	100	0
100 以下の増幅しか持たない DNA 配列の数	189	201	94	169	0

える。短い増幅を生成しないように 3.1 の貪欲アルゴリズムに禁止条件をつけることによって実現する。もっと具体的にいうとすでに選択されたプライマーのマッチする位置の矢印方向 t 以内の位置にマッチしてしまうようなプライマーは選択しないような禁止条件をつけます。そうすれば短い増幅は出てくることはない。最初の拡張では近似比は保証されていたが、この拡張をしてしまうと近似比は保証されていない。この拡張を最小制約拡張とし、t をパラメータとする。

4.3 相違制約のための拡張

今度は相違制約をみたすために同じ長さの増幅を生成しないようなアルゴリズムを考える。このために、次のような禁止条件をつける。貪欲アルゴリズムですでに選ばれたプライマーと、今、候補として考えているプライマーとで、同じ長さの増幅を作るようならそれは候補から外す。そうすれば、同じ長さの増幅が出てくることはない。この拡張をしてしまうと近似精度の保証はなくなる。この拡張を相違制約拡張とする。

5 実験

前章で考えたアルゴリズムを 500 の酵母配列に対して適用した結果が表 1、2 であるプライマーの長さは 8 のものだけを考えている。ここで注目するのは、50 から 500 の長さで他の DNA 配列にはない長さの増幅配列を持つ DNA の数を多くすることである。もちろん、そのときのプライマーの数は少なくなっていた方がよい。

表 1 では、最小制約をみたすための拡張による効果をみている。増幅制約を考えずに間隔制約拡張をしただけでは、長さの短い増幅を減らすのには貢献しないということが分かります。それに対して、短い増幅を作ることを禁止してしまえばプライマーの数は多くなっているが短い増幅のないものが得られている。しかし、増幅されない DNA 配列も出てきてしまう。

表 2 では、相違制約をみたすための拡張による効果をみている。増幅される配列を持たない DNA 配列も多くでてしまうが、$k = t = 50$ のときは 174 のプライマーによって 263 の DNA 配列が識

表 2: 相違制約のため拡張の効果

k (間隔制約拡張のパラメータ)	50		100	
t (最小制約拡張のパラメータ)	50		100	
相違制約拡張	無	有	無	有
プライマーの数	186	174	183	50
プライマーのマッチする位置の数	3274	2414	2561	1299
増幅配列の数	834	453	602	243
50 未満の長さの増幅配列の数	0	0	0	0
長さにより区別可能な増幅配列の数	174	422	167	228
長さにより区別可能な DNA 配列の数	146	317	167	199
50 から 500 の長さの増幅で区別可能な DNA 配列の数	115	263	120	133
増幅されない配列の数	15	170	98	199
50 以下の増幅しか持たない DNA 配列の数	0	0	0	0
100 以下の増幅しか持たない DNA 配列の数	94	30	0	0

別出来るような解が得られている。これは、1 つの DNA 配列に対して 1 組のプライマーを設計したときに比べて $\frac{1}{3}$ の数のプライマーでいいということである。

6 まとめ

本稿では、プライマー数最小化に対する解析と実験について述べた。ある程度の成果は得ているが、更なる理論的解析、実験が必要であると思われる。また、今後は約 6000 の酵母の全 DNA 配列、そしてさらに大きな規模の問題に適用することを考えている。それには、全 DNA 配列を識別できるように 1 つのプライマーの集合を設計するのは不可能なので、複数回の PCR 実験によって全 DNA 配列を識別することを考えて、それに対応するプライマー設計をしなくてはならない。

謝辞

プライマー選択問題に関して様々なアドバイスをいただいた東京大学医科学研究所ヒトゲノム解析センター 伊藤隆司氏に感謝致します。また、本研究の一部は文部省科学研究費重点領域「ゲノムサイエンス」の援助を受けた。

参考文献

[1] M.Bellare, S.Goldwasser, C.Lund and A.Russell, Efficient Probabilistically Checkable Proofs and Applications to Approximation, Proc. 25th Symposium on the Theory of Computing, pp.294-304, 1993.

[2] M.Bellare, S.Goldwasser and M.Sudan, Free Bits, PCPs and Non-Approximability— Towards Tight Results, Proc. 36th Symposium on Foundations of Computer Science, pp.422-431, 1995. The 3rd version is available from ECCC, http://www.eccc.uni-trier.de/eccc/

[3] J.S.Chamberlain, R.A.Gibbs, J.E.Ranier, P.N.Nguyen and C.T.Caskey, Deletion Screening of the Duchenne Muscular Dystrophy Locus via Multiplex DNA Amplification, Nucleic Acids Research 16, pp.11141-11156, 1988.

[4] K.Doi, Complexity of Primer Selection Problem in Genome Informatics, Preprint, Department of Information Science, University of Tokyo, May 1997.

[5] K.Doi and H.Imai, Greedy Algorithms for Finding a Small Set of Primers Satisfying Cover and Length Resolution Conditions in PCR Experiments, to appear in Proc. Genome Informatics Workshop 1997.

[6] U.Feige, A Threshold of $\ln n$ for Approximating Set Cover, Proc. 28th Symposium on Theory of Computing, pp314-318, 1996.

[7] J.Håstad, Clique is Hard to Approximate within $n^{1-\epsilon}$, Proc. 37th Symposium on Foundations of Computer Science, pp.627-636, 1996.

[8] T.Ito and Y.Sakaki, Fluorescent Differential Display Method for High-speed Scanning of Tissue or Cell-specific Transcripts, Methods Mol. Genet. 8, pp.229-245, 1996.

[9] C.Lund and M.Yannakakis, On the Hardness of Approximating Minimization Problems, Journal of ACM 41, pp.960-981, 1994.

[10] M.Mitsuhashi, Technical Report: Part 2. Basic requirements for designing Optimal PCR Primers, Journal of Clinical Laboratory Analysis 10, pp.285-293, 1996.

[11] M.Mitsuhashi, A.Cooper, M.Ogura, T.Shinagawa, K.Yano and T.Hosokawa, Oligonucleotide Probe Design – a New Approach, Nature 367, pp.759-761, 1994.

[12] P.Nicodème and J.Steyaert, Selecting Optimal Oligonucleotide Primers for Multiplex PCR, Proc. 5 th International Conference on Intelligent Systems for Molecular Biology, AAAI press, pp.210-213, 1997.

[13] W.R.Pearson, G.Robins, D.E.Wrege, and T.Zhang, A New Approach to Primer Selection in Polymerase Chain Reaction Experiments, Proc. Third International Conference on Intelligent Systems for Molecular Biology, AAAI Press, pp.285-291, 1995.

[14] W.R.Pearson, G.Robins, D.E.Wrege, and T.Zhang, On the Primer Selection Problem in Polymerase Chain Reaction Experiments, Discrete Appl. Math. 71, pp.231-246, 1996.

[15] T.Rowe, J.Sharefkin, S.Q.Yan and C.W.Diffenbach, A Computer Program for Selection of Oligonucleotide Primers for Polymerase Chain Reactions, Nucleic Acids Research 18, pp.1757-1761, 1990.

[16] P.Slavik, A Tight Analysis of the Greedy Algorithm for Set Cover, Proc. 28th Symposium on Theory of Computing, pp.435-441, 1996.

[17] A.Srinivasan, Improved Approximation of Packing and Covering Problems, Proc. 27th Symposium on Theory of Computing, pp.268-276, 1995.

[18] J.Welsh and M.McClelland, Fingerprinting Genomes Using PCR with Arbitrary Primers, Nucleic Acids Research 18, pp.7213-7218, 1990.