数理モデルによるDNAメモリの容量解析

山本雅人[†1,†2] 柏村聡[†3] 大内東[†1,†2]

著者らは，約16.8Mのアドレス空間を持つDNAメモリ（Nested Primer Molecular Memory: NPMM）を構築した．このDNAメモリでは，データはNested PCRとよばれる実験的操作によって，正しいアドレスを指定したときのみ取り出せる．本論文では，DNAメモリのデータ取り出し操作であるアドレッシングの過程を数理モデル化し，容量の最大化を最適化問題として定式化することで，NPMMのスケールアップの限界を議論する．

Capacity Analysis of DNA Memory Based on Mathematical Model

Masahito Yamamoto,[†1,†2] Satoshi Kashiwamura[†3] and Azuma Ohuchi[†1,†2]

We have developed a DNA Memory called NPMM with over 10 million (16.8M) addresses. The data embedded into a unique address was correctly extracted through addressing processes based on the nested PCR. In this paper, the addressing process of NPMM was modeled by using the combinatorial optimization problem in order to discuss the limitation of the scaling-up problem of NPMM.

1. はじめに

DNA（デオキシリボ核酸）のその化学的安定性や微少性を利用した計算への応用の試みとして，DNAコンピューティングの研究がさかんになった．DNAコンピューティングの研究はこれまで，組合せ問題などを扱うことが多かった[1)–4)]．DNA分子が試験管内で短時間に並列的に反応を起こす性質を持つことから，従来の逐次計算型計算機では問題のサイズに対して指数的に計算時間がかかるNP完全やNP困難な問題などに対して有効と考えられてきたからである．しかしながら，DNAコンピューティングの多くのアルゴリズムは指数関数的な計算時間の増加を，並列に反応を起こす分子数の多さで解決しようとするものであり，この考え方は，いくらDNAが扱える分子数が多い（10^{12}分子程度）とはいえ，組合せ問題のサイズ増加にともなう解空間の増加は著しいものがあるため，10^{12}分子程度ではすぐに限界がきてしまう．

†1 北海道大学大学院情報科学研究科
Graduate School of Information Science and Technology, Hokkaido University
†2 CREST，独立行政法人科学技術振興機構
CREST, Japan Science and Technology Agent (JST)
†3 株式会社日立製作所
Hitachi, Ltd.

こういった問題点が次第に明らかになるにつれてDNAコンピューティング研究は大きな方向転換をし始めている．DNAタイルとよばれる単位ナノ構造をDNA分子によって作り，その単位構造を相互に結合させて周期的・非周期的なナノ構造を作る技術を開発することで，ナノスケールの電気回路の作成やタンパク質の整列，ドラッグデリバリーシステム（DDS）へ応用しようとする試みがある[5),6)]．また，一方で，DNA，RNAやタンパク質を組み合わせて人工生命を創ろうとするもので，人工的な遺伝子回路を構成して，それらを制御する技術を開発する構成的生物学（合成生物学）とよばれる研究もさかんになってきた[7)]．これらの方向性は当初の組合せ問題を解こうとする試みとはほど遠いが，DNAなどの生体分子の形状や機能を自由に制御する必要があるという意味で，これまでのDNAコンピューティング研究の延長線上にあるものといえる．

さらに，DNAの性質を利用したナノデバイスの構築も行われている．その中でもDNAメモリは，古くから行われている研究ではあるが，最近，精度や規模の面から注目を浴びつつある[8)–12)]．著者らは，これまでDNAの増幅反応（ポリメラーゼ連鎖反応（PCR））を用いたDNAメモリを提案し，徐々にスケールアップと精度向上を行ってきた[13),14)]．特に，最近では約

16.8M のアドレス空間を持つ DNA メモリの構築に成功して，実際にアドレッシング（データの読み取り）を実験的に行っている[15]．

16.8M のアドレス空間を持つ DNA メモリは，これまでの DNA コンピューティング研究で実現されてきたメモリの大きさとしては最大である．DNA メモリは，HDD などの記憶デバイスと同様の速度や精度でデータを読み書きするためのデバイスとしての応用は難しいが，暗号化や DNA インクといった応用面に期待が持たれている．たとえば，作成した DNA メモリを十分希釈すると，いくつかのアドレスがランダムに欠落するが，その欠落のパターンが再現困難であることを利用して，再構築困難な DNA メモリを作成できる．この DNA メモリを溶かしたインクをマスタとして，そのインクで書いた文書は，再構築困難な文書となりうる．その文書がマスタの DNA メモリのアドレスパターンと一致するかどうかを調べることによって，その文書の真贋検査などへの応用が可能である[16)–18)]．こういった応用面を考えても，アドレス空間が十分大きな DNA メモリはセキュリティの分野で有用である．

しかしながら，DNA メモリがどの程度までスケールアップが可能なのかということについてはこれまでほとんど議論がなされてこなかった．その理由として，これまで大容量とよぶべき規模の DNA メモリが開発されなかったために，スケールアップの問題が重要視されてこなかったこと，また，分子生物学実験に基づくメモリ操作を行うため，その正確性の議論が困難であったこと，などがあげられる．

本論文では，著者らが開発した大容量の DNA メモリのモデルを対象として，容量解析を行うことを目的とし，DNA メモリのアドレッシング動作における制約を数理モデルとして表現し，容量の最大化問題を組合せ最適化問題としてモデル化する方法を提案する．そして，実際にその最適化問題を解くことによって，DNA メモリの容量限界やアドレス分子の割当て戦略について議論する．

2. DNA メモリ

著者らが提案した DNA メモリは，Nested Primer Molecular Memory（以下，NPMM）とよぶものである[15)]．NPMM は，塩基配列でコードされたデータ部とデータ部の両端に階層的に付加されたアドレス部からなるものであり，データ部のデータを読み出すためにはアドレスとなるプライマ対を順次指定して繰り返し PCR 増幅（Nested PCR）を行う．アドレスが階層化されていることにより，より種類数の少ないプライマ対で大規模なアドレス空間を表現可能である．**図 1** は，本研究で構築した NPMM を模式的に表したものである．

図 1 約 16.8M の NPMM の塩基配列構造．アドレス部は，両端 3 階層（XY，$X \in \{A, B, C\}$，$Y \in \{L, R\}$）からなり，各階層には，(XYi，$i \in \{0, 1, 2 \ldots 15\}$) の 16 種類の配列が割り当てられている

Fig. 1 Sequence structure of each DNA strand in 16.8M-NPMM. The area expressing the address information consists of six layers (XY, $X \in \{A, B, C\}$, $Y \in \{L, R\}$) and sixteen sequences are defined in each layer (named XYi, $i \in \{0, 1, 2 \ldots 15\}$).

アドレス部については，内側から A 階層，B 階層，C 階層と各階層の両側に 16 種類ずつの配列を持っている．したがって，この NPMM のアドレス空間は，16^6 = 16,777,216 となる．便宜上，データ部の左側（5’ 末端側）の各階層を内側から AL，BL，CL とよび，右側の各階層を内側から AR，BR，CR とよぶこととし，アドレス部それぞれの 16 種類のアドレス配列には 0〜15 の数字を割り当て，AL5，BR11 などと各アドレスを表現することとする．

各アドレス配列，および，それらをつなぐためのリンカ配列は，20 の長さを持つ塩基配列で表されるが，それらは相互になるべく似た配列とならないように設計した．これは，アドレス配列を認識する際，その配列の相補鎖（その配列と 2 本鎖を形成する各塩基が相補対となっている塩基配列）を用いて，2 本鎖を形成させる反応を用いるため，これらの配列がある程度似ていると誤った反応を起こし，読み取りの動作が正しくできない．アドレス配列が相互になるべく似た配列にならないように設計をする方法は，これまで DNA コンピューティングの研究で数多くなされてきているが[19)–22)]，今回は，著者らが開発した 2-step search とよぶ方法によって設計を行った[23)]．

図 1 の NPMM で実際に必要な塩基配列は，3 階層で両側 16 配列のアドレス部分の DNA 配列として 96 本，および，各アドレス部をつなぐためのリンカ部の配列として 6 本，データ部として長さがそれぞれ，20，40，60 のデータ data20，data40，data60 の 3 本，の計 105 本である．しかしながら，実際には，これ以上のスケールアップを考えて，200 本を超える配列を設計した．そして，そのうちから二次構造（ヘアピンなど自身の配列どうしの結合にできる構造）をとりにくい配列 105 本を選択して使用した．

上記の構造を設計した後，実際に，約 16.8M のア

ドレス空間を持つ DNA メモリを構築する必要があるが，その詳細については本論文との直接の関連性が低いことから述べない（文献 13)–15) を参照）．しかしながら，試験管内では，約 16.8M（約 1.68×10^7）種類の DNA 分子が存在することになるが，理想的にすべての種類の DNA 分子が均等に作られたとしても，実験として現実に扱える分子数の制限により，それぞれが数千〜数万分子ずつしか存在していないことには注意が必要である．たとえば，試験管内で扱うことのできる総 DNA 分子数が 10^{12} 分子と仮定した場合，原理的には 1 アドレスを 1 分子で表現できても，10^{12} 以上のアドレス空間を実現することはできないし，すべてのアドレスを 1 分子ずつ正確に作ることは実験的に不可能である．さらに，たとえ 1 アドレスを 1 分子で表現した 10^{12} 分子があったとして，1 分子のみを抽出することは今の実験技術では困難である．

したがって，本論文では，1 アドレスにつき数百分子ずつが存在するような DNA メモリを想定する．ただし，実際のメモリ構築のための実験操作によって，各アドレスを表す DNA 分子がすべて均等な数ずつ存在するように作成することは困難であることには注意が必要である．これについては，後の数理モデルで再度議論する．また，各階層に割り当てる配列数は，今回構築したように 16 種類と限定されるものではない．この数を変更すれば，NPMM のアドレス空間も変わるし，実験的な実現困難さも変わる．本論文では，実験的な困難さを考慮しつつ，この各階層への配列の割当て数をどのように変更すれば NPMM のアドレス空間を大きくできるかについて議論する．

3. DNA メモリのアドレッシング

3.1 アドレッシング方法

本節では，NPMM からデータを読み取る操作について説明する．本論文では，この読み取り操作をアドレッシング（addressing）とよぶ．

アドレッシングには，Nested PCR とよばれる方法を用いた[24]．**図 2** に，アドレッシングの様子を示した．図では，アドレス [$CL0$, $BL2$, $AL4$, $AR5$, $BR3$, $CR1$] を指定したときに，そのアドレスに対応するデータのみを取り出す操作を示したものである．第 1 段階として，アドレス配列の一部（両端の外側）から，$CL0$ と $CR1$ という配列の相補的な配列（プライマとよぶ）を約 16.8M の NPMM が入っている試験管内に入れポリメラーゼ連鎖反応（PCR）による増幅を行う．PCR は，2 本鎖の解離，プライマの結合，ポリメラーゼ伸長という一連の反応を 1 サイクルとして，温度制御とともに数十サイクル（通常は 30 サイクル程度）行うものである．その結果，約 16.8M 種類の DNA 分子から $CL0$ と $CR1$ のみを含む配列だけが多数増幅され，その溶液を希釈することによって，溶液内には $CL0$ と $CR1$ のみを含むメモリ分子だけ残ることが期待できる．ただし，この増幅後に，$CL0$ と $CR1$ のみを含む DNA 分子の総数が，それ以外の DNA 分子の総数に対して数百倍以上多くなければ，希釈した際に望みの DNA 分子以外が混在することになる．このため，アドレッシングが正確に動作するための数理モデル構築の際には注意が必要である．

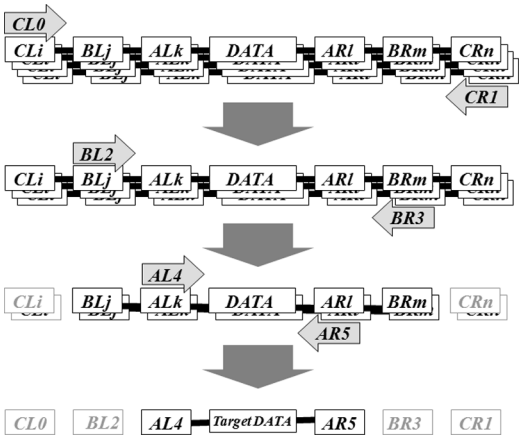

図 2 NPMM におけるデータ読み取り操作
Fig. 2 Operations for retrieving the target DNA from NPMM are implemented in the nested PCR.

さらに，その溶液に対して，次に内側のアドレス配列 $BL2$ と $BR3$ をプライマとして再度 PCR を行う．その結果，溶液には，前段階から含めると $CL0$, $CR1$，および，$BL2$, $BR3$ をすべて含むメモリ分子のみが残ることになる．この操作を最後の階層である $AL4$ と $AR5$ についても同様に行うことで，結果的にアドレス [$CL0$, $BL2$, $AL4$, $AR5$, $BR3$, $CR1$] のアドレス分子のみを含む DNA 分子が抽出可能である．抽出した配列データを読み取ることでデータの読み取りが可能となる．

3.2 アドレッシング検証実験

アドレッシングが適切に行われたかどうかを示すための実験を以下のように行った．16.8M のアドレス空間のすべてについて，長さ 20 のデータである data20 を付加しておく．一方，16.8M のアドレス空間のうちたった 1 つのアドレス [$CL0$, $BL0$, $AL0$, $AR0$, $BR0$, $CR0$]（すべて 0 のアドレス配列を使用するため All 0 とよぶ）には，通常の data20 を付加した分子のほかにさらに data40 を付加した分子を溶液中に混合す

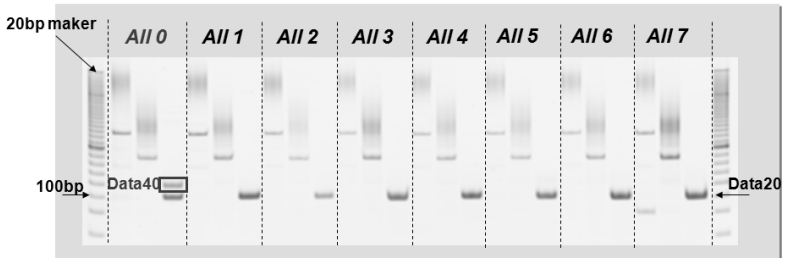

図 3 ゲル電気泳動によるアドレッシング結果
Fig. 3 Result of PolyAcrylamide Gel Electrophoresis for addressing.

る．すなわち，All 0 のアドレスに対しては，data20 と data40 の両方がアドレッシングの結果抽出されることを期待する．結果については，文献 15) に示されているので，ここではその一部である All 0 から All 7 までの 8 種類のアドレスに対するアドレッシング結果を**図 3** に示す．各アドレスに対する 3 つのレーンは左から C 階層，B 階層，A 階層のアドレッシング後の溶液を表しており，一番右端のレーンが最終的なアドレッシング結果を意味している．

図 3 を見て分かるとおり，All 0 のアドレスについては，data20 と data40 の 2 つのバンドが，それ以外のアドレスでは，data20 のみのバンドが検出されており，適切なアドレッシングが行われていることを示している．

4. 数理モデル

4.1 定 式 化

前章までの説明で，すでに約 16.8M の NPMM が実現できたといえるが，ではこの NPMM はどこまでスケールアップが可能であるだろうか．これまでに述べたように，DNA の 1 分子がいかに小さいとはいえ，NPMM に容量限界があることは明らかである．本論文では，この NPMM の容量限界について，数理モデルを構築して解析することにより議論を行った．以下では，その数理モデルについて詳述する．

NPMM でのアドレッシングは，各階層において両端のアドレス配列やその相補鎖をプライマ対として指定し PCR を行い，本来増幅すべき DNA 分子が十分な量増幅した後，次の階層のプライマ対を用いたアドレッシングへと進む．この際，適切なアドレッシングが実行されるためには，各階層の PCR について以下のような条件が満足されなければならない．

(a) PCR で増幅される分子の総数は，適切なプライマ量 P 以下でなければならない．PCR では，増幅したい鋳型 DNA にプライマがハイブリダイゼーション（アニール）して，ポリメラーゼ伸長することにより鋳型 DNA が複製される．この過程を何度も繰り返すことで鋳型 DNA を指数関数的に増幅することができるが，投入するプライマ量以上は絶対に増幅しえないことは明確である．

(b) 増幅後は，増幅されるべき DNA 分子数の総数 amp が増幅されない DNA 分子の総数 non_amp に対して十分大きくなければならない．すなわち，増幅後に amp が non_amp に対して少なくとも α 倍（数百倍〜数千倍）以上は存在していなければならない．これについては，3.1 節で述べた．

(c) 過剰なサイクル数の PCR 反応は，非特異的な（本来望むべき以外の）PCR 産物を生成する可能性が高いために避ける．すなわち，i 階層目の PCR におけるサイクル数 c_i はサイクル数の最大値 max_cycle 以下とする．これは，分子生物学的実験では，経験的に知られている事実である．

(d) 各アドレスに割り当てる DNA 配列は，互いに非特異的な反応をしないような配列である必要があるため，アドレス部に使用する DNA 配列の総数が用意可能な互いに似ていない配列数 N 以下である必要がある．

これらを考慮して，NPMM の容量最大化（アドレス空間の最大化）を目的とする最適化問題として定式化を行う．すなわち，上記の (a)〜(d) の条件をすべて満たす中で，各階層へ割り当てるアドレス分子の種類数 n_i を変えて，アドレス空間が最大となるものを探す．定式化された問題を**図 4** に示す．ただし，ここでは簡単化のために，各階層における両端のアドレス DNA の種類は同じであると仮定する．つまり，i 階層目の両端で用いられるアドレス DNA の種類数は同じであるため，n_i で表すとする．

以下では，図 4 の最適化問題の詳細について説明する．まず，目的関数は，総アドレス空間を最大化するため以下のとおりとなる．各階層に割り当てられるアドレス配列の種類数が n_i であり，両端に存在するため，n_i^2 を各階層ごとにかけたものが総アドレス空間

$$
\begin{aligned}
&\textit{maximize} && \prod_{i=1}^{L} n_i^2 \\
&\textit{sub.to} && n_i \geq 2 \ (i \in \{1,2,\dots L\}) && (1)\\
& && N \geq \sum_{i=1}^{L} 2n_i && (2)\\
& && v(2^{c_i}-1)\prod_{j=i+1}^{L} n_j^2 + 2vc_i(n_i-1)\prod_{j=i+1}^{L} n_j^2 \leq P \quad \forall i \in \{1,2,\dots L\} && (3)\\
& && amp_{c_i} > \alpha \times non\_amp_{c_i} \quad \forall i \in \{1,2,\dots L\} && (4)\\
& && 0 \leq c_i \leq max\_cycle \quad \forall i \in \{1,2,\dots L\} && (5)\\
& && P,\ L,\ N,\ v,\ \alpha,\ max\_cycle : \text{integer (given)}
\end{aligned}
$$

図 4 NPMM の容量解析のための最適化問題
Fig. 4 Combinatorial optimization problem for NPMM analysis.

となる．先に説明した約 16.8M の NPMM の場合は，各階層のアドレス配列数は 16 であったので，総アドレス空間は，$16^2 \times 16^2 \times 16^2 = 16{,}777{,}216$ である．

$$Capacity = \prod_{i=1}^{L} n_i^2.$$

次に，制約条件についてそれぞれ説明する．

式 (1) の条件は各階層に割り当てるアドレス DNA 配列数は 2 以上であることを示しており，$n_i = 1$ の場合は，階層とする意味がないため当然の制約である．この制約を満たすことは容易である．

式 (2) の制約は，アドレス配列に使用する塩基配列の種類数が，非特異的な反応を起こさないような互いに似ていない配列として設計可能な数 N を超えない，というものであり，上記の説明の (d) に相当する．実際には，リンカ部やデータ部の配列も設計したアドレス配列の候補から使用するため，右辺に加えた方がよいかもしれないが，ここでは，簡単のためはずしてある．

式 (3) と式 (4) の制約がこの数理モデルで最も重要なものである．ここで，v は，増幅前に各アドレスに対応する DNA 分子がどれだけ存在するか，というパラメータであり，たとえば，$v = 1$ とすると，各アドレスに対応する分子はたった 1 分子ずつしか存在していない状態を意味する．実験的には，1 分子のみを PCR で増幅することは困難とされているため，v は数十から数百分子存在することを仮定する．

この式 (3) と式 (4) の制約は，PCR の性質から導き出せるものである．ここで重要となる PCR の性質は以下のものである．

(x) DNA 分子の両端の配列がプライマ対とそれぞれ相補結合する DNA 分子は，PCR サイクルに対して指数関数的に増加する．(y) DNA 分子の片側の配列のみがプライマ対のどちらかと相補結合する DNA 分子は，PCR サイクルに対して線形増加する．(z) DNA 分子の両端の配列がプライマ対と相補結合しない DNA 分子は増えない．

この性質から，式 (3) の制約を説明する．PCR によって増幅するのは，上記の (x) と (y) の DNA 分子であるから，それらについて考える．(x) については，現在の階層 i より内側の階層のすべてのアドレスを含む DNA 分子が該当する．この種類数は $\prod_{j=i+1}^{L} n_j^2$ である．これに各アドレスの分子数 v をかけた $v\prod_{j=i+1}^{L} n_j^2$ が総分子数である．これらの分子がサイクル数とともに指数関数的（2 のサイクル乗）に増加するが，もとの DNA 分子を引いた分が，サイクルとともに増加する分子数 $v(2^{c_i}-1)\prod_{j=i+1}^{L} n_j^2$ となる

次に (y) については，プライマ対の片方のみと相補結合する分子数は，内側のアドレスをすべて含む DNA 分子の種類数 $\prod_{j=i+1}^{L} n_j^2$ に対して，i 階層目の配列数 n_i のうち，プライマ対と相補結合する数 1 を除いた数となるため，$(n_i-1)\prod_{j=i+1}^{L} n_j^2$ となる．各アドレスが v の分子数を持っており，右側で相補結合する場合と，左側で相補結合する場合の両方があるため，総分子数は，$2v(n_i-1)\prod_{j=i+1}^{L} n_j^2$ となる．これらがサイクルとともに線形（比例）に増加するため，サイクルごとの増加分子数は，$2vc_i(n_i-1)\prod_{j=i+1}^{L} n_j^2$ となる．

したがって，式 (3) の左辺は，上記 (x) と (y) のサイクルごとの増加分子数を表しており，その数が投入するプライマ量 P を超えないという制約となっている．

式 (4) での，amp_{c_i} と $non\_amp_{c_i}$ は，以下の式で表される．

$$amp_{c_i} = v \prod_{j=i+1}^{L} n_j^2 + v(2^{c_i} - 1) \prod_{j=i+1}^{L} n_j^2. \quad (6)$$

$$non\_amp_{c_i} = v(n_i^2 - 1) \prod_{j=i+1}^{L} n_j^2 + 2vc_i(n_i - 1) \prod_{j=i+1}^{L} n_j^2. \quad (7)$$

式 (6)，式 (7) のそれぞれ第 1 項は，増幅前の分子数を表し，第 2 項は，式 (3) の説明で示した PCR で増幅される分子数を表したものである．式 (6) は，増幅されるべき DNA 分子の増幅後の分子数を示しており，増幅前の分子数 $v\prod_{j=i+1}^{L} n_j^2$ に，上記の (x) の増幅分子数を足したものとなっている．同様に，式 (7) は，増幅されるべきでない DNA 分子の増幅前の分子数 $v(n_i^2-1)\prod_{j=i+1}^{L} n_j^2$ に，(y) の増幅分を足したものとなっている．

したがって，式 (4) は，PCR 後に増幅されるべき DNA 分子の総数が増幅されるべきでない DNA 分子の総数に対して，α 倍以上であることを示しており，上記の (b) に対応する．

4.2 NPMM の容量解析

この最適化問題を適切なパラメータを設定して解くことにより，NPMM の容量限界について議論することができる．パラメータとしては，NPMM の階層数 L，用意可能な互いに似ていない配列の設計可能数 N，PCR における最大サイクル数 max_cycle，および，適切なプライマ量 P，そして，PCR 増幅後に増幅対象 DNA 量 amp と非増幅対象 DNA 量 non_amp の濃度差に関わる係数 α がある．

一般的な PCR で用いられるパラメータを考慮し，$max\_cycle = 30$，$P = 30\,\mathrm{pmol}$（$100\,\mu$l 溶液を考えた場合の標準的な量），また，配列数 N については，長さ 20 の塩基配列で前述した 2-step search 法を用いた場合にすでに得られている配列数を考慮して 200 とした．ただし，16.8M の NPMM で使用した数 96 についても別途解析を行った．また，前述したように α は数百から数千であることが望ましいと考えられるため，α は，500 から 1,500 の範囲で変えた．さらに，v についても 50〜200 の範囲で変更した．設定したパラメータ値は**表 1** にまとめて示す．

表 1 設定したパラメータ値
Table 1 Parameter settings.

P	3.0×10^{13}
L	2，3，4
N	96，200
α	500〜1,500
v	50〜200
max_cycle	30

最適化問題は，枝刈りを用いた全探索を用いて解いた．各階層へのアドレス配列の割当て数を変えて目的関数の最大化を図るが，各階層，各サイクルにおいて，制約を満たさなくなった割当てや，これまで求めた最大容量を下回るような割当てについては，探索を打ち切り，次の割当てを探索する方法を用いた．割り当て方の列挙の方法によって，計算効率が変わることは容易に分かるが，本論文では計算時間が小さかったために特に工夫はしなかった．

まず，現在構築できている 16.8M の NPMM と同じ構造について解析を行った．階層数 $L = 3$ として，利用可能な配列数 N を 96 とした場合，パラメータ α と v が表 1 の範囲にあるすべての場合で，各階層に割り当てる配列数は 16 とした方が良いという実際の構築したメモリと同じ割当てが得られた．これは，制約の中で式 (3) や式 (4) を満たすことが厳しくない状況では，各階層に同じ数の配列数を割り当てる理論的な解が最適となることを示している．

しかしながら，利用できる配列数が多くなったり，階層数が多くなったりした場合には，状況は異なってくる．**表 2** は，階層数 L を 3 としたときの，α と v を変化させたときの限界容量を示したものである．$\alpha = 1{,}500$ の場合には，v によらず，一番外側の C 階層目に割り当てる配列数が 55 となっており，式 (4) による制約を満たすのが厳しい状況であることを示唆している．

$\alpha = 1{,}000$，$v = 100$ と固定して階層数 L を変化させた場合の限界容量（目的関数値）と，その際の各階層に割り当てる配列数を示したのが**表 3** である．

階層数 2 の場合については，やはり式 (3) や式 (4) の制約が厳しくならない条件のため，α や v が表 1 の範囲では，すべての階層に 50 ずつの配列を割り当てるのが良いという結果が得られた．

また，階層数 3 の場合では 170M 程度が限界となることが分かる．この数字は，上記の条件で実験を行う限り，この数字以上のアドレス数を持つ NPMM を作成した場合は，実験によってアドレッシングできないという意味で欠損するアドレス部が現れる可能性が高

表 2 容量解析結果（$L = 3$ の場合）：表の下段の括弧内の数字は，どの階層にどれだけの配列数を割り当てるかを示したものである．$\alpha = 50$, $v = 50$ のときは，(CL, BL, AL, AR, BR, CR) に (47, 38, 14, 14, 38, 47) が対応することを示している
Table 2 Theoretical capacity of case of $L = 3$. The numbers in brackets means the assignment of the number of address sequences.

$\alpha \setminus v$	50	100	200
500	625,200,016 (47,38,14,14,38,47)	317,623,684 (67,19,14,14,19,67)	181,279,296 (72,17,11,11,17,72)
1,000	317,623,684 (67,19,14,14,19,67)	176,278,729 (71,17,11,11,17,71)	89,170,249 (71,19,7,7,19,71)
1,500	214,036,900 (55,38,7,7,38,55)	108,056,025 (55,27,7,7,27,55)	53,509,225 (55,19,7,7,19,55)

表 3 容量解析結果（L を変えた場合）
Table 3 Theoretical capacity of case of $L = 2$, 3, 4.

階層数	$L = 2$	$L = 3$	$L = 4$
容量	6,250,000	176,278,729	180,069,561
（配列の割当て）	(50,50,50,50)	(71,17,11,11,17,71)	(71,21,3,3,3,3,21,71)

いことを意味する．

NPMM のアドレス配列を外側か内側のどちらにより多く割り当てた方が良いかについては，そう単純ではない．外側のアドレス数を多くすると，最初の増幅時に非増幅対象となる DNA 分子の種類が増えるため *non_amp* の総数も多くなる．その結果，増幅後に増幅対象 DNA の分子数 *amp* と *non_amp* の濃度差を十分大きくできない．また，外側のアドレス数を少なくすると，最初の増幅時に増幅対象 DNA の種類数が増えるので，増幅後に 1 種類あたりの分子数を大きくすることができない．今回提案した基本構造やパラメータにおいては，より多くの配列を外側に割り当てた方が良いという結果が得られた．

16.8M の NPMM については，すべての階層で使用したアドレス配列数は同じであった．また，互いに似ていない配列として 100 本程度のみを使用した．本論文の結果をふまえて，今後の大容量化を考えてみると，互いに予期しない相補結合をしないような配列をより多く用意して，各階層には，外側に対してより多くの配列を割り当てるのが良い，という戦略が示唆されるが，現在のアドレス配列数を倍にしても容量としては 10 倍程度にしか増えないということも意味している．このように設計する配列数と期待できる容量の関係を事前に議論できることは，この数理モデルの有効性を示しているといえる．

5. おわりに

本論文では，著者らが提案した DNA メモリである NPMM を対象として，さらなるスケールアップを実現するために，NPMM でのアドレッシングの過程を数理モデル化し，アドレッシングが適切に動作するための条件を提案した．そして，NPMM のスケールアップ問題を最適化問題として定式化し，それを解くことによって，NPMM の限界容量とアドレス配列の各階層への割当てについて議論した．提案した数理モデルでは，NPMM 構築の際に，各アドレスを示す DNA 分子が均等に生成されているという理想的な状況を想定しており，その意味では単純なモデルとなっているが，PCR が適切に動作するための条件を組み込んだ新しいモデルとなっている．したがって，まだモデルの改良をする余地は残っているものの，本論文で提案した解析手法は，これまで DNA メモリに関して議論されてこなかったスケールアップ問題（容量限界）についての数理的アプローチの 1 つとして有効であると考えている．

大容量の DNA メモリが構築できれば，構築時のアドレスの欠損がランダムに起こることを利用すると，どのアドレスが欠損していて，どのアドレスが欠損していないかという点で，同一のものを構築することが困難な DNA メモリの構築が可能となる．このような DNA メモリは，認証や真贋検査などに応用可能であると考えられている[25]．

謝辞 本論文を作成するにあたり，モデルの構築，実験に関するアドバイスなどをいただいた東京大学の萩谷昌己先生，陶山明先生，亀田充史さん，八重樫さつきさんに感謝します．

参考文献

1) Adleman, L.M.: Molecular Computation of Solutions to Combinatorial Problems, *Science*,

Vol.266, pp.1021–1024 (1994).
2) Lipton, R.: DNA solution of hard combinatorial problems, *Science*, Vol.268, pp.542–545 (1995).
3) Sakamoto, K., Gouzu, H., Komiya, K., Kiga, D., Yokoyama, S., Yokomori, T. and Hagiya, M.: Molecular computation by DNA hairpin formation, *Science*, Vol.288, pp.1223–1226 (2000).
4) Braich, R.S., Chelyapov, N., Johnson, C., Rothemund, P.W.K. and Adleman, L.: Solution of a 20-Variable 3-SAT Problem on a DNA Computer, *Science*, Vol.296, pp.499–502 (2002).
5) Yan, H., LaBean, T.H., Feng, L. and Reif, J.H.: Directed nucleation assembly of DNA tile complexes for barcode-patterned lattices, *Proc. National Academy of Sciences 100-14*, pp.8103–8108 (2003).
6) Winfree, E., Liu, F., Wenzler, L.A. and Seeman, N.C.: Design and Self-Assembly of Two-Dimensional DNA Crystals, *Nature*, Vol.394, pp.539–544 (1998).
7) 木賀大介，山村雅幸：構成的生物学—つくることで理解する生物学，人工知能学会誌，Vol.20, No.6, pp.715–721 (2005).
8) Baum, E.B.: Building an Associative Memory Vastly Larger Than the Brain, *Science*, Vol.268, pp.583–585 (1995).
9) Chen, J., Deaton, R. and Wang, Y.Z.: A DNA-based memory with in vitro learning and associative recall, *Natural Computing*, Vol.4, No.2, pp.83–101 (2005).
10) Wong, P.C., Wong, K.K. and Foote, H.: Organic Data Memory Using the DNA Approach, *Comm. ACM*, Vol.46, No.1, pp.95–98 (2003).
11) Neel, A., Garzon, M.H. and Penumatsa, P.: Improving the Quality of Semantic Retrieval in DNA-Based Memories with Learning, Lecture Notes in Computer Science, Vol.3213, pp.2489–2498 (2004).
12) Reif, J.H., LaBean, T.H., Pirrung, M., Rana, V.S., Guo, B., Kingsford, C. and Wickham, G.S.: Experimental Construction of Very Large Scale DNA Databases with Associative Search Capability, *DNA Computing*, Jonoska, N. and Seeman, N.C. (Eds.), Lecture Notes in Computer Science, Vol.2340, pp.231–247 (2002).
13) Kashiwamura, S., Yamamoto, M., Kameda, A. and Ohuchi, A.: Hierarchical DNA Memory based on Nested PCR, *DNA Computing*, Hagiya, M. and Ohuchi, A. (Eds.), Lecture Notes in Computer Science, Vol.2568, pp.112–123 (2003).
14) Kashiwamura, S., Yamamoto, M., Kameda, A., Shiba, T. and Ohuchi, A.: Potential for enlarging DNA memory: The validity of experimental operations of scaled-up nested primer molecular memory, *BioSystems*, Vol.80, pp.99–112 (2005).
15) Yamamoto, M., Kashiwamura S. and Ohuchi, A.: DNA Memory with 16.8M Addresses, *Preliminary Proc. DNA13*, pp.19–29 (2007).
16) Hashiyada, M.: Development of Biometric DNA Ink for Authentication Security, *Tohoku J. Exp. Med.*, Vol,204, pp.109–117 (2004).
17) Hashiyada, M., Itakura, Y., Nagashima, T., Nata, M. and Funayama, M.: Polymorphism of 17 STRs by multiplex analysis in Japanese population, *Forensic Sci. Int.*, Vol.133, pp.250–253 (2003).
18) Itakura, Y., Hashiyada, M., Nagashima, T. and Tsuji, S.: Proposal on Personal Identifiers Generated from the STR Information of DNA, *Int. J. Information Security*, Vol.1, pp.149–160 (2002).
19) Arita, M.: Comma-free design for DNA words, *Comm. ACM*, Vol.47, No.5, pp.99–100 (2004).
20) Deaton, R., Murphy, R.C., Garzon, M., Franceschetti, D.R. and Stevens, Jr, S.E.: Good Encoding for DNA-Based Solutions to Combinatorial Problems, *DNA Based Computers II, DIMACS Series in Discrete Mathematics and Theoretical Computer Science*, Landweber, L.F. and Baum, E.B. (Eds.), Vol.44, pp.247–258 (1999).
21) Tanaka, F., Kameda, A., Yamamoto, M. and Ohuchi, A.: Design of nucleic acid sequences for DNA computing based on a thermodynamic approach, *Nucleic Acids Research*, Vol.33, pp.903–911 (2005).
22) Tulpan, D.C., Hoos, H.H. and Condon, A.: Stochastic Local Search Algorithms for DNA word Design, *DNA Computing*, Hagiya, M. and Ohuchi, A. (Eds.), Lecture Notes in Computer Science, Vol.2568, pp.229–241 (2003).
23) Kashiwamura, S., Kameda, A., Yamamoto, M. and Ohuchi, A.: Two-Step Search for DNA Sequence Design, *IEICE Trans. Fundamentals of Electronics, Communications and Computer Sciences*, Vol.E87-A, No.6, pp.1446–1453 (2004).
24) McPherson, M.J., Quirke, P. and Taylor, G.R.: *PCR A Practical Approach*, pp.42–43, IRL Press, New York (1991).
25) Clelland, C.T., Risca, V. and Bancroft, C.: Hiding message in DNA microdots, *Nature*, Vol.399, pp.533–544 (1999).

26) Yurke, B., Turberfield, A.J., Mills Jr., A.P., Simmel, F.C. and Neumann, J.L.: A DNA-fuelled molecular machine made of DNA, *Nature*, Vol.406, pp.605–608 (2000).

(平成 19 年 8 月 8 日受付)
(平成 19 年 9 月 28 日再受付)
(平成 19 年 10 月 16 日採録)

山本　雅人（正会員）

昭和 43 年生．平成 8 年北海道大学大学院工学研究科システム情報工学専攻博士後期課程修了．平成 8 年より日本学術振興会特別研究員（PD）．平成 9 年より北海道大学大学院工学研究科助手．平成 12 年より北海道大学大学院工学研究科助教授．平成 19 年より北海道大学大学院情報科学研究科准教授．DNA コンピューティング，マルチエージェントシステムの研究に従事．博士（工学）．

柏村　聡

昭和 53 年生．平成 17 年北海道大学大学院情報科学研究科博士後期課程修了．平成 17 年より日本学術振興会特別研究員（PD）．博士（工学）．同年日立製作所入社．

大内　東（正会員）

昭和 20 年生．昭和 49 年北海道大学大学院工学研究科博士後期課程修了．同年より北海道大学工学部助手，助教授を経て，平成元年より北海道大学大学院工学研究科教授．平成 16 年より北海道大学大学院情報科学研究科教授．現在に至る．工学博士．