情報処理学会論文誌　数理モデル化と応用　Vol.6　No.3　87–99 (Dec. 2013)

CMO問題に対する改良版EOを用いた発見的解法

中田 章宏[1,a)]　田村 慶一[1,b)]　北上 始[1,c)]　高橋 誉文[1]

受付日 2013年1月30日，再受付日 2013年3月22日，
採録日 2013年4月23日

概要：蛋白質の類似構造を抽出する手法として蛋白質立体構造アラインメントがある．蛋白質立体構造アラインメントは構造的な類似性を使い，比較する蛋白質どうしの残基間の対応関係を求める．蛋白質立体構造アラインメントを組合せ最適化問題として定式化したのが CMO（Contact Map Overlap）問題である．CMO 問題はコンタクトマップと呼ばれるグラフについて，頂点間のアラインメントにより保存される共通コンタクトと呼ばれる構造の数を最大化する問題である．CMO 問題は，NP 困難な問題の 1 つであることが知られており，進化計算を用いた手法などの研究が行われている．本論文では，CMO 問題に対する改良版 EO を用いた発見的解法を提案する．提案手法の特徴は，(1) 世代交代に改良版 EO を用いること，(2) 動的計画法を用いて初期個体を作成すること，(3) 改良版 EO における状態遷移に即時移動戦略ではなく，最良移動戦略を用いることである．提案手法を実際に実装し，評価実験を行った結果，EO による発見的解法よりも評価の高い最良解が得られた．

キーワード：CMO 問題，Extremal Optimization，発見的解法

Heuristic Using Modified EO for CMO Problem

Akihiro Nakada[1,a)]　Keiichi Tamura[1,b)]　Hajime Kitakami[1,c)]　Yoshifumi Takahashi[1]

Received: January 30, 2013, Revised: March 22, 2013,
Accepted: April 23, 2013

Abstract: Proteins are important biochemical compounds that have biogenic functions for biological activities. The three-dimensional structures of proteins are closely related to its biological functions, and therefore, techniques for comparing them have been studied. Many of these techniques for comparing protein structures are based on protein structure alignment, which is one of the most effective methods. CMO (Contact Map Overlap) is formulated as combinatorial optimization to find the optimal structure alignments. In this paper, we propose a novel heuristic using Modified Extremal Optimization (MEO) for CMO. Our MEO-based heuristic is characterized by three features. First, the proposed heuristic uses MEO for alternation generations. Second, an initial solution is created by dynamic programming (DP). Third, state transition is executed using the best admissible move strategy.

Keywords: Contact Map Overlap, Extremal Optimization, heuristic

1. はじめに

蛋白質は酵素，抗体やホルモンなど，我々の生命活動を支える生体機能を持つ重要な物質の 1 つである．蛋白質は，DNA 塩基配列から翻訳されたアミノ酸配列から作られ，特有のかたち（三次元の立体構造）を形成する．立体構造がその蛋白質が持つ生体機能を決定するといわれているがその関係性は十分に解明されていない．ただし，アミノ酸配列が似ていなくても立体構造が類似する蛋白質どうしはその生体機能がお互いに類似しているといわれており，蛋白質の立体構造を比較する研究 [1], [2] がさかんに行われている．

蛋白質の立体構造を比較するときに必要とされている機能が類似構造の抽出であり，類似構造を抽出するために広く利用されているのが蛋白質立体構造アラインメント [3]

[1] 広島市立大学大学院情報科学研究科
Graduate School of Information Sciences, Hiroshima City University, Hiroshima 731–3194, Japan
[a)] mw67025@edu.ipc.hiroshima-cu.ac.jp
[b)] ktamura@hiroshima-cu.ac.jp
[c)] kitakami@hiroshima-cu.ac.jp

© 2013 Information Processing Society of Japan

である．蛋白質立体構造アラインメントは構造的な類似性を使い，比較する蛋白質どうしの残基間の対応関係を求める．蛋白質はアミノ酸が鎖状にペプチド結合した高分子であり，元々のアミノ酸部分を残基と呼ぶ．残基間の対応関係は蛋白質を構成するアミノ酸の数が増えるとともにその組合せが膨大となり，最適なアラインメントを求めることは，バイオインフォマティクスにおいて最も難しい問題の1つとして知られている [4]．

蛋白質立体構造アラインメントを組合せ最適化問題として定式化したのが CMO（Contact Map Overlap）問題 [5], [6] である．CMO 問題では，蛋白質の残基を頂点とし，近接する残基どうしを辺で結んだコンタクトマップと呼ばれるグラフを作成する．CMO 問題は，コンタクトマップ間のアラインメントにより保存される共通コンタクトと呼ばれるオーバラップ構造の数を最大化する問題として定義される．CMO 問題を解くことで，共通コンタクトの数が類似度，また，アラインメントが示す残基間の対応関係が類似構造として抽出される．

CMO 問題は，NP 困難な問題の 1 つであることが知られており，分枝限定法や線形計画法にラグランジュ緩和を組み合わせた手法がその解法として提案されている．また，厳密解を求めるには非常に多くの計算量が必要なため，実用的な観点から，進化計算や EO（Extremal Optimization）[7], [8] などの発見的解法を用いた手法の研究も行われている．その中でも EO を用いた CMO 問題の発見的解法 [9] は他の手法と比較して，より良い最良解が得られることが示されている．

本論文では，改良版 EO [10] を用いた CMO 問題の発見的解法を提案する．提案手法は，

（1）世代交代に改良版 EO を用いる，
（2）初期個体は動的計画法を用いて作成する，
（3）改良版 EO における状態遷移に即時移動戦略ではなく，最良移動戦略を用いる，

という 3 つの特徴を持っている．

改良版 EO では，複数の近傍個体の中から一番良い個体を次世代の個体として選択する．複数の近傍個体を生成し，状態遷移を繰り返すため，EO と比較して局所解に陥りにくい手法であることが示されている．また，動的計画法を用いて初期アラインメントを作成し，その初期アラインメントを初期個体とする．動的計画法で求めたアラインメント結果を初期個体として用いることで，ある程度最適な構造からスタートできる．

提案手法を実際に実装し，先行研究で用いられている実際の蛋白質立体構造データを PDBj（Protein Data Bank Japan）[11] から取得し，提案手法の評価を行った．評価実験の結果，提案手法は EO を用いた発見的解法と比較して，評価の高い最良解を求めることができた．また，先行研究と比較しても評価の高い最良解を求めることができており，CMO 問題に対する新しい解法として有効であることを示すことができた．

本論文の構成は以下のとおりである．2 章では，関連研究について述べる．3 章では，CMO について，その問題定義を示す．4 章では改良版 EO について説明し，5 章では提案手法を示す．6 章で評価実験の結果を示し，7 章において，本論文をまとめる．

2. 関連研究

CMO 問題の初期研究においては，CMO 問題を整数計画問題として定義し，整数計画問題を解くことで厳密解を求めることがさかんに行われていた．CMO 問題は NP 困難な問題であることが示されているため [12]，線形計画問題に置き換えて定式化するアプローチ [13]，線形計画問題に対してラグランジュ緩和を導入した手法 [14], [15], [16]，緩和問題を作成し，分枝限定法を用いた手法 [17] などが提案されている．

また，CMO 問題を他の組合せ最適化問題に置き換えるアプローチも提案されている．これは，すでに最適な解法が提案されている他の組合せ最適化問題に置き換えることで CMO 問題の評価の高い最良解を高速に求める試みである．CMO 問題を最大クリーク問題に置き換える手法 [18], [19] や，CMO 問題を MCS 問題に置き換え，動的計画法を用いて MCS 問題を解く手法 [20] が提案されている．

このように，CMO 問題の解法は，緩和問題を工夫することや別問題に置き換えるアプローチが多いが，蛋白質の大きさが大きくなるほど計算時間が増加することが問題となっている．また，近年，wwPDB など蛋白質立体構造データを収集管理する組織で格納されている蛋白質立体構造データは指数関数的に増えており，比較する蛋白質の数が多くなるほど処理時間が膨大となる．そこで，厳密解を求めるのではなく，進化計算などの発見的解法を用いて高速に近似解を求める手法も研究が行われてきている [9], [21]．

本研究に最も関連する EO を用いた発見的解法 [9] は，Greedy アルゴリズムによる初期解生成と，EO による個体の世代交代を組み合わせた手法となっている．しかしながら，EO を用いた発見的解法には，大域的な変化が起こりにくい問題では，局所解へ陥りやすいという問題点がある．本研究では，改良版 EO を用いているため，この問題を解決できる．

また，文献 [21] では，遺伝的アルゴリズムを用い，個体の突然変異に EO を使用した手法を提案している．複数個体を使用する遺伝的アルゴリズムと正確な比較を行うことはできないが，本研究では，改良版 EO を用いることで，遺伝的アルゴリズムよりも探索の集中性をあげることができ，個体単位での探索性能は提案手法の方が優れているといえる．

© 2013 Information Processing Society of Japan

3. CMO 問題

蛋白質の残基を頂点，近接する残基間をコンタクトエッジと呼ばれる辺で結んだグラフをコンタクトマップと呼ぶ．コンタクトマップの各頂点は残基の中心座標と結び付けられる．残基の中心座標として，本研究では他の CMO 問題を対象とした研究と同様に Cα 原子の座標を用いる．また，蛋白質 v の i 番目の残基と，j 番目の残基は，それぞれ，i 番目の頂点 v_i，j 番目の頂点 v_j として表現される．頂点 v_i と頂点 v_j とが辺で結ばれている場合，残基 i と残基 j 間の距離が，与えられたカットオフ距離 *cutoff* 未満であることを示す．

図 1 にコンタクトマップの例を示す．図 1 の上部に蛋白質 A の残基の空間的な構造を示し，図 1 の下部に蛋白質 A のコンタクトマップを示す．ここで，蛋白質 A の残基 1 に着目すると，残基 1 の中心座標から *cutoff* 以内に残基 3 と残基 5 が存在する．よって，コンタクトマップにおいて，頂点 A_1 と頂点 A_3 間にコンタクトエッジ (A_1, A_3)，頂点 A_1 と頂点 A_5 間にコンタクトエッジ (A_1, A_5) が作成されている．同様に，残基 3 と残基 5 間の距離も *cutoff* 以内であるため，コンタクトエッジ (A_3, A_5) が作成されている．

ここで，蛋白質 v のコンタクトマップ CM_v を，$CM_v = (RV_v, CE_k)$ と表現する．ただし，$RV_v = \{v_1, v_2, \cdots, v_n\}$ は頂点集合であり，

$$CE_v = \{(v_i, v_j) \mid v_i \in RV_v, v_j \in RV_v, \\ i < j, dist(v_i, v_j) < cutoff\} \quad (1)$$

はコンタクトエッジの集合を表す．ただし，関数 *dist* は残基 i と残基 j の中心座標間の距離を返す関数とする．たとえば，図 1 の蛋白質 A のコンタクトマップは，$CM_{\mathrm{A}} = (RV_{\mathrm{A}}, CE_{\mathrm{A}})$ と表し，このとき，$RV_{\mathrm{A}} = \{A_1, A_2, A_3, A_4, A_5\}$，$CE_{\mathrm{A}} = \{(A_1, A_3), (A_1, A_5), (A_3, A_5)\}$ である．

蛋白質 v と蛋白質 w とをそれぞれ表現するコンタクトマップ CM_v と CM_w の部分頂点集合（$RV_v^+ \subseteq RV_v$，$RV_w^+ \subseteq RV_w$）間を 1 対 1 に対応付けることをアラインメントという．また，アラインメントされた頂点のペアをアラインメントペアと呼ぶ．ここで，このアラインメントを全単射として，

$$\phi : {RV_v}^+ \to {RV_w}^+, v_i \mapsto w_{\phi(i)}, \quad (2)$$

と定義すると，アラインメントペア集合 AL^ϕ は，

$$AL^\phi = \{(v_i, w_{\phi(i)}) \mid v_i \in {RV_v}^+, w_{\phi(i)} \in {RV_w}^+\}, \quad (3)$$

と表現することができる．

図 2 は 2 つの蛋白質 A と蛋白質 B のコンタクトマップ間に作成されたアラインメントの例を示している．点線で結ばれた頂点どうしがアラインメントされた残基を示している．この例では，3 つのアラインメントペア (A_1, B_1)，(A_3, B_3)，(A_5, B_4) が存在する．よって，$AL^\phi = \{(A_1, B_1), (A_3, B_3), (A_5, B_4)\}$ となる．

ただし，アラインメントペアは次の条件を満たす必要がある．

$$i < j \;\Rightarrow\; \phi(i) < \phi(j). \quad (4)$$

この制約は，アラインメントペア間に交差が生じないことを示している．

ここで，アラインメントペア $(v_i, w_{\phi(i)})$ と $(v_j, w_{\phi(j)})$ とについて，頂点 v_i と頂点 v_j 間と，頂点 $w_{\phi(i)}$ と頂点 $w_{\phi(j)}$ 間とにコンタクトエッジが存在する場合，つまり，$(v_i, v_j) \in CE_v$ かつ $(w_{\phi(i)}, w_{\phi(j)}) \in CE_w$ が成り立つ場合，コンタクトマップがオーバラップするといい，このオーバラップのことを共通コンタクトと呼ぶ．

たとえば，**図 3** において，(A_1, B_1) と (A_3, B_3) の 2 つのアラインメントペアに着目する．ここで，頂点 A_1 と頂点 A_3 の間，また，頂点 B_1 と頂点 B_3 の間にコンタクトエッジ（図中の太線）が存在するため，(A_1, B_1) と (A_3, B_3) の 2 つのアラインメントペアに共通コンタクトが 1 つ存在する．

CMO 問題はこの共通コンタクト数を最大化するアラインメントペア集合を求める問題である．具体的には，以下のコスト関数 f を最大化する問題として定義される．

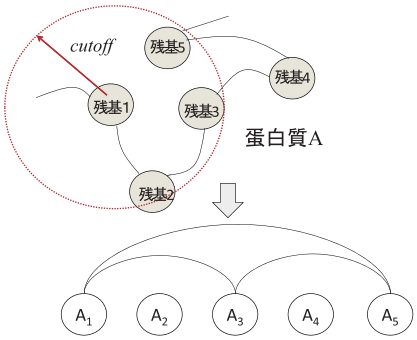

図 1　コンタクトマップの例
Fig. 1　Example of contact map.

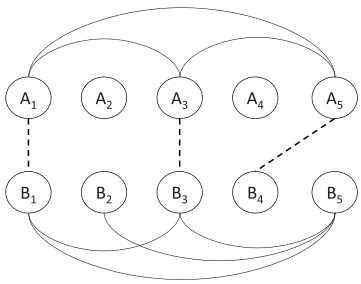

図 2　アラインメントの例
Fig. 2　Example of alignment.

© 2013 Information Processing Society of Japan

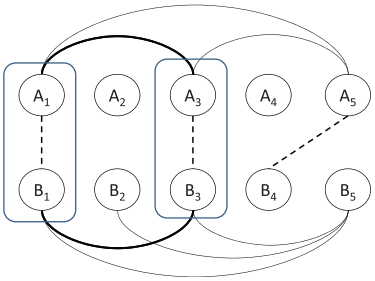

図 3　共通コンタクトの例
Fig. 3　Example of common contact.

$$f(AL^{\phi}) = \sum_{(v_i,w_{\phi(i)})\in AL^{\phi},(v_j,w_{\phi(j)})\in AL^{\phi}(i<j)} g(v_i,w_{\phi(i)},v_j,w_{\phi(j)})$$

$$g(v_i,w_{\phi(i)},v_j,w_{\phi(j)}) = \begin{cases} 1 & if\ (v_i,v_j)\in CE_v \\ & and\ (w_{\phi(i)},w_{\phi(j)}) \\ & \quad \in CE_w \\ 0 & \text{otherwise.} \end{cases} \tag{5}$$

図 1 の例では，共通コンタクトは 1 つ存在する．よって，このコスト関数 f は値として 1 を返す．

4. 改良版 EO

EO [7], [8] は個体の中で適応度が悪い構成要素を選択し，その構成要素を状態遷移することで，個体の適応度を向上させていく発見的解法である．構造物において強度が弱い部分を補強することでその構造物の強度が向上するという考えに基づいている．EO は，スピングラス，グラフの分割問題，グラフの彩色問題，巡回セールスマン問題，センサデータのトラッキング，画像の対応付けに応用されるロバストな点対応付け，クラスタリングやコミュニティ抽出問題などに応用されており，様々な分野で有効性が示されている [7], [22], [23], [24], [25], [26].

EO のアルゴリズムを Algorithm 1 に示す．個体 I は n 個の構成要素 O_i（$1 \le i \le n$）から構成される．ここで，λ_i を構成要素 O_i の適応度とする．また，$\mathcal{F}$ は個体の適応度を返す関数であり，この値が大きいほど適応度が大きい．初期個体の生成方法，構成要素とその適応度の計算方法や状態遷移の方法については，アプリケーションごとに異なるため，EO 自体に明確な方法は示されてはいない．最初に，適応度 λ_i が最も小さい構成要素 O_i を状態遷移の候補として選択する．そして，当該構成要素を対象として，I に対して状態遷移を行い，次世代の個体とする．以下，構成要素の選択と状態遷移を繰り返しながら，解の探索を進めていく．

改良版 EO では，複数の近傍個体を生成し，近傍個体の中で最良の個体を次世代の個体として選択する．個体のコピーを複数個用意し，コピーして作成した各個体についてルーレット選択を用いて構成要素を選択する．そして，選択した構成要素を状態遷移させる．最後に，近傍個体中で一番適応度の高い個体を次世代の個体として選択する．通常の EO ならば，もし選択された結果が改悪となるにしても，その状態遷移を行うしかないが，改良版 EO ならば近傍個体を複数生成するため改良となる個体が生成される可能性も高まり，結果，改悪へと進む可能性が減る．

改良版 EO のアルゴリズムを Algorithm 2 に示す．最初に個体 I をランダムに生成する．そして，ユーザが指定した最大世代数 $gmax$ になるまで次の処理を繰り返す．最初に，個体 I のコピー $I_{neighbor}$ を生成する．次に，構成要素の適応度を使用してルーレット選択で構成要素 O_k を 1 つ

Algorithm 1: EO

input ：最大世代数 $gmax$
output: 最良個体 I_{best}

```
初期個体 I をランダムに生成
I_best = I
g = 0
while g < gmax do
    個体 I のすべての構成要素 O_i に関して，その適応度 λ_i を算出
    適応度 λ_i が最も悪い構成要素 O_k を選択
    構成要素 O_k を対象として個体 I を状態遷移
    if F(I) > F(I_best) then
        I_best = I
    g++
return I_best
```

Algorithm 2: 改良版 EO

input ：最大世代数 $gmax$，近傍個体生成数 $nmax$
output: 最良個体 I_{best}

```
初期個体 I をランダムに生成
I_best = I
g = 0
while g < gmax do
    n = 0
    N = φ
    I のすべての構成要素 O_i に関して，その適応度 λ_i を算出
    while n < nmax do
        I_neighbor = I
        適応度 λ_i のルーレット選択により構成要素 O_k を選択
        構成要素 O_k を対象として I_neighbor を状態遷移
        N = N ∪ I_neighbor
        n++
    I =best (N)
    if F(I) > F(I_best) then
        I_best = I
    g++
return I_best
```

© 2013 Information Processing Society of Japan

選択する．そして，構成要素 O_k を対象として $I_{neighbor}$ を状態遷移し，$I_{neighbor}$ を N に格納する．最後に，N の中から最も適応度が高い個体を取り出し，次世代の個体とする．ここで，関数 `best` は個体の集合から最も適応度が高い個体を取り出す関数である．

5. 提案手法

本章では，提案手法である改良版 EO を用いた CMO 問題の発見的解法の詳細内容を示す．

5.1 個体と構成要素の定義

改良版 EO を CMO 問題に適用するにあたり，はじめに個体とその構成要素を定義する必要がある．蛋白質 v と蛋白質 w を表現するコンタクトマップ $CM_v = (RV_v, CE_v)$ と $CM_w = (RV_w, CE_w)$ とすると，本研究では，アラインメントペア集合 AL^{ϕ} をそのまま個体 I として定義する．

また，アラインメントペアを構成する頂点 1 つ 1 つを構成要素 O_i（$\in RV_v \cup RV_w$）とする．たとえば，図 2 では，$I = \{(A_1, B_1), (A_3, B_3), (A_5, B_4)\}$ であり，個体は $O_1 = A_1$，$O_2 = B_1$，$O_3 = A_3$，$O_4 = B_3$，$O_5 = A_5$，$O_6 = B_4$ の 6 つの構成要素から構成される．

5.2 適応度の定義

蛋白質 v と蛋白質 w を表現するコンタクトマップを $CM_v = (RV_v, CE_v)$ と $CM_w = (RV_w, CE_w)$ とすると，個体 I の適応度である大域的適応度は 3 章で示したコスト関数 f を用い，

$$global\_fitness(I) = \frac{f(I)}{min(|CE_v|, |CE_w|)}, \tag{6}$$

と定義する．たとえば，図 2 の例では，$|CE_{\mathrm{A}}| = 3$, $|CE_{\mathrm{B}}| = 4$ で，$f(I) = 1$ であるため，$global\_fitness(I) = 1/3$ となる．

ここで，頂点 v_k と頂点 $w_{\phi(k)}$ に接続しているコンタクトエッジの中で共通コンタクトであるコンタクトエッジの数をそれぞれ $com(v_k)$，$com(w_{\phi(k)})$ とする．

$$\begin{aligned} com(v_k) &= com(w_{\phi(k)}) \\ &= \sum_{(v_j, w_{\phi(j)}) \in AL^{\phi}} g(v_k, w_{\phi(k)}, v_j, w_{\phi(j)}). \end{aligned} \tag{7}$$

構成要素 O_i の適応度である局所的適応度は，構成要素 O_i に対応する頂点の次数で頂点が持つ共通コンタクト数を割った値とする．

$$local\_fitness(O_i) = \begin{cases} \frac{com(v_k)}{dig(v_k)} & if\ O_i = v_k \in CV_v \\ \frac{com(w_{\phi(k)})}{dig(w_{\phi(k)})} & if\ O_i = w_{\phi(k)} \in CV_w. \end{cases} \tag{8}$$

上記の式で，頂点の次数を $dig(v_k)$, $dig(w_{\phi(k)})$ とする．ただし，次数が 0 である頂点はつねに局所的適応度は 0 とする．

5.3 初期個体

初期個体となる初期アラインメントは残基間の構造的な類似度を使い，動的計画法を用いて作成する．高速かつある程度精度の高い構造アラインメント結果を求めるために，動的計画法を用いて，蛋白質立体構造間の構造アラインメントを求める手法 [27] がある．本研究では，コンタクトマップを用いて残基間の構造的な類似度を求め，動的計画法を用いて初期アラインメントを作成する．動的計画法を用いることで，文献 [9] の手法と比較して，より最適な個体から解の探索をスタートすることができる．

最初に，比較する 2 つの蛋白質の残基間の構造的な類似度をスコア関数（スコア行列）として定義する．残基間の類似度 $s_{i,j}$ は以下の定義式で求める．

$$\begin{aligned} &s_{i,j} \\ &= \alpha \times \Big(\frac{min(dig(v_i), dig(w_j))}{max(dig(v_i), dig(w_j))} + \frac{min(sd(v_i), sd(w_j))}{max(sd(v_i), sd(w_j))} \Big), \end{aligned} \tag{9}$$

ここで，$dig(v_i)$ と $dig(w_j)$ は頂点の次数であり，$sd(v_i)$ と $sd(w_i)$ はコンタクトエッジで接続している他の頂点が示す残基との空間的な距離の総和である．また，α は係数である．

次に，残基の並びを配列要素として，最適アラインメントを求める．はじめに，動的計画法を用いて，スコア行列 D の各要素 $D_{i,j}$ を計算する．

$$\begin{aligned} &D_{i,0} \leftarrow i \times g \quad i = 0, \cdots, n \\ &D_{0,j} \leftarrow j \times g \quad j = 0, \cdots, m \\ &D_{i,j} \leftarrow min \begin{cases} D_{i-1,j} + g \\ D_{i,j-1} + g \\ D_{i-1,j-1} + \frac{max(S) - s_{i,j}}{max(S)} \end{cases} \end{aligned} \tag{10}$$

ここで，n と m はそれぞれ，2 つの蛋白質の残基数であり，式中の $max(S)$ は類似行列 S の最大要素の値，$s_{i,j}$ は類似行列 S の第 (i, j) 要素の値である．また，g はペナルティスコアであり，次の値を用いる．

$$g = \frac{\sum_{k=1}^{n} \sum_{l=1}^{m} \frac{max(S) - s_{k,l}}{max(S)}}{n \times m}. \tag{11}$$

スコア行列が算出できたら，スコア $D_{n,m}$ から最大値を算出する経路をトレースバックしていく．つまり，スコア $D_{n,m}$ を算出するのに，上，左上，左のどちらの要素の数値が採用されたかトレースバックする．

5.4 アルゴリズム

提案手法のアルゴリズムを Algorithm 3 に示す．最初に，入力した蛋白質の座標配列データからコンタクトマップと類似度行列 S を作成する．次に，動的計画法を用いて初期アラインメントを求める．そして，初期アラ

© 2013 Information Processing Society of Japan

Algorithm 3: 提案手法

input ：蛋白質 A と蛋白質 B の座標配列データ，カットオフ値 $cutoff$，最大世代数 $gmax$，近傍個体生成数 $nmax$
output: 最良個体 I_{best} が持つアラインメントペア集合 AL^{ϕ}

コンタクトマップ CM_A と CM_B を作成し，類似度行列 S を生成する．
類似度行列 S を用いて動的計画法により初期アラインメントを生成し，初期アラインメントを I とする．
$I_{best} = I$
$g = 0$
while $g < gmax$ **do**
    I の全構成要素 O_i について，局所的適応度 $local\_fitness(O_i)$ を算出する．
    NI = **make_neighbor_individuals**(I,$nmax$)
    I =**best**(NI)
    if $global\_fitness(I) > global\_fitness(I_{best})$ **then**
        $I_{best} = I$
    g++
return 最良個体 I_{best} が持つアラインメントペア集合 AL^{ϕ}

Algorithm 4: make_neighbor_individuals

input ：個体 I，最大近傍個体生成数 $nmax$
output: 近傍個体集合 NI

$n = 0$
$NI = \phi$
while $n < nmax$ **do**
    I のコピーを作成し，$I_{neighbor}$ に保存する．
    局所的適応度 $local\_fitness(O_i)$ のルーレット選択により，$I_{neighbor}$ の構成要素 O_k を選択する．
    選択した構成要素 O_k が示す頂点について，当該頂点を移動することで作成可能なアラインメントペアをすべて調べ，個体の大域的適応度が最も大きくなるアラインメントペアを選択し，置き換える．
    $NI = NI \cup I_{neighbor}$
    n++
return NI

インメントを初期個体，また現時点の最良解として設定する．続いて，ユーザが指定した世代数まで改良版 EO を用いて，状態遷移を繰り返す．最初に，構成要素についてその適応度 $local\_fitness(O_i)$ を求める．次に，関数 **make_neighbor_indivisuals** を呼び出し，個体の近傍個体となる複数の個体（近傍個体集合 NI とする）を生成する．近傍個体集合 NI から個体の適応度が最良の個体を 1 つ選択し，次世代の個体とする．もし，次世代の個体が最良個体よりも評価の高い個体ならば最良個体としてその個体のコピーを保存する．

Algorithm 4 に関数 **make_neighbor_individuals** の内容を示す．最初に，個体 I のコピーを作成し，$I_{neighbor}$ に保存する．$I_{neighbor}$ の構成要素をその局所的適応度を用いて，ルーレット選択で 1 つ選択する．次に，選択した構成要素を状態遷移する．状態遷移の方法については，次節に示す．状態遷移を行った $I_{neighbor}$ を近傍個体集合 NI に保存する．この一連の処理を $nmax$ 回繰り返すことで，$nmax$ 個の近傍個体を作成し，作成した近傍個体集合 NI を返す．

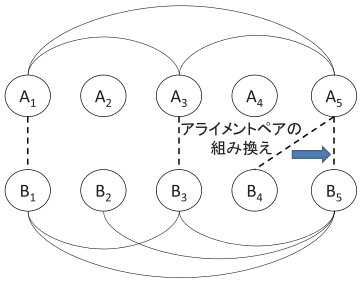

図 4　状態遷移の例
Fig. 4　Example of change state.

5.5　状態遷移

構成要素の状態遷移はアラインメントペアの組み換えにより行う．たとえば，構成要素 O_k が状態遷移の候補として選択され，$O_k = v_k$ と仮定する．アラインメントペア $(v_k, w_{\phi(k)})$ について，v_k を蛋白質 v の他の頂点に変更する．逆に，$O_k = w_{\phi(k)}$ と仮定すると，アラインメントペア $(v_k, w_{\phi(k)})$ について，$w_{\phi(k)}$ を蛋白質 w の他の頂点に変更する．

状態遷移することで，アラインメントペアで交差が生じないという制約を満たす必要がある．ここで，アラインメントペア $(v_k, w_{\phi(k)})$ について，$l < k$ となる最大の値 l を持つアラインメントペアを $(v_l, w_{\phi(l)})$ とし，$k < u$ となる最小の値 u を持つアラインメントペアを $(v_u, w_{\phi(u)})$ とする．もし，$O_k = v_k$ である場合，頂点 v_k を $l < k' < u$，$k' \neq k$ を満たす頂点 $v_{k'}$ に置き換える．また，$O_k = w_{\phi(k)}$ の場合，頂点 $w_{\phi(k)}$ を $\phi(l) < k' < \phi(u)$，$k' \neq \phi(k)$ を満たす頂点 $w_{k'}$ に置き換える．

ただし，最大の値 l を持つアラインメントペアが存在しない場合は，選択する頂点の最小の添字番号は 1 となる．また，最小の値 u を持つアラインメントペアが存在しない場合は，選択する頂点の最大の添字番号は，$O_k = v_k$ である場合は，蛋白質 v の頂点数となり，$O_k = w_{\phi(k)}$ の場合，頂点 w の頂点数となる．

図 4 に状態遷移におけるアラインメントペアの組み換えの例を示す．図 4 の例では，B_4 が選択された構成要素で，アラインメントペア (A_5, B_4) をアラインメントペア (A_5, B_5) に組み換えた例を示している．このアラインメントペアの組み換えで，組み換え前と比較して共通コンタクトの数が増加し，組み換え前と比較して評価の高いアラインメントとなっていることが分かる．

アラインメントの組み換えでは，ランダムに最初に選ん

© 2013 Information Processing Society of Japan

だ他の頂点を選択する即時移動戦略と，組み換え可能なすべての頂点の候補の中から個体の大域的適応度の高くなる頂点を選択する最良移動戦略の 2 種類が考えられる．本研究では，最良移動戦略を用い，組み換え可能なすべての頂点の中から個体の大域的適応度が最も大きくなる頂点を選び，アラインメントの組み換えをする．

6. 評価実験

提案手法を評価するために評価実験を行った．本章では評価実験の実験結果を示す．

6.1 データセット

評価実験では，文献 [9] と文献 [21] の実験結果と本研究とを比較するために，両文献内で共通に用いられている蛋白質立体構造データ 25 件を評価用の蛋白質立体構造データとして用いた．また，先行研究との比較と加えて，網羅的な組合せで提案手法の有効性を確認するために，オリジナルの Skolnick データセット 40 件も実験に用いた．

文献 [9] と文献 [21] で用いられていた蛋白質立体構造データ 25 件の内訳は，9 件が Sokol テストデータセット [18] で，残り 16 件は，オリジナルの Skolnick データセット 40 件（文献 [14]）から Xu らが文献 [28] において抜き出して用いた 16 件の蛋白質立体構造データである．ただし，この 16 件の蛋白質立体構造データは，オリジナルの Skolnick データセットとは同じ蛋白質であるが，オリジナルとは異なる鎖を用いているため PDB の番号が若干異なっている．以後，オリジナルの Skolnick データセットと区別するために本論文では Xu らのデータセットと呼ぶ．

表 1 に Sokol テストデータセットの蛋白質立体構造データ 9 件を示す．Sokol テストデータセットは，Carr ら [13] が蛋白質立体構造アラインメントを評価するために用いたテストデータセットであり，ヒト科，ウシ科，キジ科，ウミヘビ科の 4 種が持つ蛋白質立体構造データで構造が類似している 3 種類の蛋白質ドメインの蛋白質から構成されている．評価実験では，コンタクトマップは $cutoff = 6.75$ Å として作成した．

表 1　Sokol テストデータセット
Table 1　Sokol test data set.

PDB ID	残基数	コンタクトエッジ数
1bpi	58	195
1knt	55	192
2knt	58	200
5pti	58	190
1vii	36	120
1cph	21	65
3ebx	73	275
6ebx	62	205
1era	62	208

表 2 に Xu らのデータセットを示す．オリジナルの Skolnick データセットから 16 件の蛋白質立体構造データを抜き出したデータセットである．Skolnick データセットは Sokol テストデータセットと比較して，残基数が多く，SCOP データベースにおける 5 種の分類に分類される球状蛋白質から構成されている．オリジナルの Skolnick データセットの一覧を**表 3** に示す．Xu らのデータセットは 1 番目から 4 番目の分類から 16 件の蛋白質を取り出している．Xu らのデータセットにおいても，同じく，コンタクトマップは $cutoff = 6.75$ Å として作成した．

6.2 実験内容

評価実験では，次の 5 つの実験を行った．実験 1 では，最良個体の共通コンタクト数の平均，標準偏差について，提案手法と EO による発見的解法とを比較する．また，論文上の数値のみであるが先行研究との比較も行う．実験 2 では，提案手法の状態遷移の効果を調べるために，即時移動戦略と最良移動戦略とを比較し，実験 3 では，近傍個体生成数を変化させたときの比較をする．実験 4 では，同世代交代数による，提案手法と EO による発見的解法との比較を行う．実験 5 では，オリジナルの Skolnick データセット 40 件の全組合せによる比較結果を示す．

実験 1 から実験 4 は，CPU: Intel (R) Core (TM) 2 Quad CPU Q6700 @2.66 GHz，Memory: 2 GB，HDD: 250 GB を搭載した PC 上で評価実験を行った．また，実験 5 は，CPU: Intel（R）Xeon X5560 プロセッサ（2.80 GHz）× 2CPU，Memory: 12 GB DDR-3 SDRAM（1,333 MHz，ECC，2 GB × 6），HDD: 1 TB を搭載した PC 上で評価実験を行った．

実験 1 から実験 4 では，文献 [9] と文献 [21] に示されて

表 2　Xu らのデータセット
Table 2　Xu et al.'s data set.

PDB ID	残基数	コンタクトエッジ数
1b00A	122	488
1b00B	122	423
1bawA	105	387
1dbwA	125	474
1qmpA	125	454
1qmpC	125	452
4tmyA	118	473
1byoA	99	355
1dpsB	154	586
1dpsC	154	585
1nat	119	435
1amk	250	1,086
2pcy	99	357
8timA	247	930
1aw2B	254	1,043
1b9bA	252	953

© 2013 Information Processing Society of Japan

表 3　Skolnick データセット
Table 3　Skolnick data set.

	フォールド	分類属性	PDB 番号
1	Flavodoxin-like	CheY-related	1b00, 1dbw, 1nat, 1ntr, 1qmp(A,B,C,D), 3chy, 4tmy(A,B)
2	Cupredoxin-like	Plastocyanin/azurin-like	1baw, 1byo(A,B), 1kdi, 1nin, 1pla, 2b3i, 2pcy, 2plt
3	TIM beta/alpha-barrel	Triosephosphate isomerase (TIM)	1amk, 1aw2, 1b9b, 1btm, 1hti,1tmh, 1tre, 1tri, 1ydv, 3ypi, 8tim
4	Ferritin-like	Ferritin	1b71, 1bcf, 1dps, 1hfa, 1ier, 1rcd
5	Microbial ribonucleases	Fungal ribonucleases	1rn1(A,B,C)

表 4　最良個体の共通コンタクト数の平均と標準偏差（Sokol テストデータセット）
Table 4　Average of number of common contacts and standard deviation of best solutions (Sokol test data set).

PDB 番号		平均		標準偏差	
蛋白質 A	蛋白質 B	提案手法	EO	提案手法	EO
1bpi	1knt	**161**	154	0.507	0.935
1bpi	2knt	**174**	170	0.674	0.520
1bpi	5pti	**187**	182	0.466	0
1knt	2knt	**185**	179	0	0
1knt	5pti	**160**	154	0.498	0.946
1vii	1cph	53	**54**	0.922	0.0225
2knt	5pti	**172**	170	1.019	0.508
3ebx	1era	**166**	164	3.473	0.994
3ebx	6ebx	**195**	190	1.917	0
6ebx	1era	**181**	179	1.607	0

表 5　最良個体の共通コンタクト数の平均と標準偏差（Xu らのデータセット，フラボドキシンに似た形状を持つ蛋白質データの組合せ）
Table 5　Average of number of common contacts and standard deviation of best solutions (Xu et al.'s data set with flavodoxin-like flold).

PDB 番号		平均		標準偏差	
蛋白質 A	蛋白質 B	提案手法	EO	提案手法	EO
1b00A	1dbwA	**287**	261	5.300	2.527
1b00A	1nat	**300**	279	1.539	3.070
1b00A	1qmpC	**317**	283	5.884	3.846
1nat	1b00B	**291**	278	3.530	3.736
1nat	1dbwA	**370**	358	3.660	1.847
1nat	4tmyA	**321**	317	7.647	3.901
1qmpC	1b00B	**294**	276	3.657	2.380
1qmpC	4tmyA	**328**	312	5.094	3.150
4tmyA	1b00B	**252**	238	3.779	3.520

いる 32 組の蛋白質立体構造データの組合せについて測定を行う．Sokol テストデータセットに関する 10 組の組合せと，Xu らのデータセットについては 22 組の組合せとで合計 32 組の組合せとなっている．文献 [9] では 33 組となっているが，1 組重複した記載があり，文献 [21] では 32 組となっている．さらに，Xu らのデータセットの 22 組の組合せについては，フラボドキシンに似た形状を持つ蛋白質立体構造データを組み合わせた 9 組の組合せと，また，異なる形状を持つ蛋白質立体構造データを組み合わせた 13 組の組合せから構成される．

6.3　実験 1

実験 1 では，提案手法と EO を用いた発見的解法とで最良個体の共通コンタクト数の比較をする．ただし，実験で使用する EO を用いた発見的解法は，文献 [9] に示された手法とは初期解として動的計画法を用いている点が異なる．EO を用いた発見的解法では，100 秒間，世代交代を繰り返す．提案手法では，近傍個体生成数を 100 と設定し，同じく，100 秒間，世代交代を繰り返す．また，それぞれ 30 回ずつ実行し，得られた最良個体の共通コンタクト数の平均と標準偏差を求める．

Sokol テストデータセットにおける組合せの実験結果を**表 4** に示す．Sokol のテストデータセットにおける 10 組の組合せについては，10 組中 9 組が提案手法の方が良い結果が得られた．ただし，若干の差はあるものの，両者ほぼ同じ結果が得られ，共通コンタクト数は近差である．これは，Sokol テストデータセットに含まれる蛋白質データは残基数が少なく，大部分が類似する構造であるため，EO による発見的解法でも十分に良い最良個体が得られるためである．

次に，Xu らのデータセットにおけるフラボドキシンに似た形状を持つ蛋白質のデータセットの組合せ（**表 5**）と，Xu らのデータセットにおける異なる形状を持つ蛋白質データの組合せ（**表 6**）においても，提案手法の方が EO を用いた発見的解法よりも良い結果が得られた．標準偏差の値からも提案手法の方が EO を用いた発見的解法よりも値が大きい部分で分散していることが分かる．Xu らのデータセットは，Sokol テストデータセットと比較して残基数が多く，また，部分的な構造が似ているため，局所解に陥りやすい問題となっているため，EO を用いた発見的解法は局所最適解に陥っている．

表 4, 表 5 と表 6 の結果から，EO を用いた発見的解法よりも提案手法の方が優れているといえる．これは，提案手法は複数の近傍個体を作成するために，EO を用いた発見的解法よりも極端な改悪の方向に向かうことが少なく，また，近傍個体から最良個体を選択しているため変化も期待でき，局所最適解から抜け出せる手法となっているためである．

© 2013 Information Processing Society of Japan

表 6　最良個体の共通コンタクト数の平均と標準偏差（Xu らのデータセット，異なる形状を持つ蛋白質データの組合せ）

Table 6　Average of number of common contacts and standard deviation of best solutions (Xu et al.'s data set with different fold).

PDB 番号		平均		標準偏差	
蛋白質 A	蛋白質 B	提案手法	EO	提案手法	EO
1b00A	1bawA	**158**	151	5.300	2.527
1b00A	1byoA	**157**	147	3.936	2.083
1b00A	1dpsB	**251**	228	5.884	2.279
1nat	1amk	**237**	176	13.06	3.964
1nat	1dpsB	**267**	234	6.185	2.775
1qmpC	2pcy	**159**	156	3.591	1.801
1qmpA	8timA	**239**	193	6.328	5.128
4tmyA	1bawA	150	**152**	3.520	2.585
4tmyA	1amk	**210**	163	13.193	4.866
4tmyA	1dpsC	**237**	213	5.932	2.445
1bawA	1aw2B	**176**	135	2.940	3.543
1bawA	1b9bA	**186**	142	4.994	2.465
1bawA	1dpsB	**182**	167	3.722	2.112

次に，**表 7** に文献 [9] と文献 [21] との論文上の数値での比較結果を示す．文献 [9] はオリジナルの EO による発見的解法であり，本実験で用いた EO による発見的解法と初期解生成方法が異なる．文献 [21] は遺伝的アルゴリズムを用いており，複数個体を用いる手法で，遺伝的アルゴリズムの突然変異操作に EO を用いている．実験環境が異なるのと，複数個体を用いる手法との正確な比較はできないが，現時点でどの手法が数値的に優れた最適解が得られているかという観点で比較を行う．よって，提案手法と本研究で実装した EO の結果は，平均値ではなく 30 試行中の最良値を掲載している．

まず，全体の結果であるが，表 7 の太字が最も共通コンタクト数が良い結果を示している．各手法において最も良い結果が出た総数は，同数である結果も重複してカウントして，提案手法が 14 件，本研究で実装した EO による発見的解法が 9 件，文献 [9] の手法が 2 件，文献 [21] の手法が 12 件であり，提案手法が最も優れていた．

次に，個体数が 1 つである提案手法と文献 [9] の手法を直接比較すると，32 組中 27 組において提案手法の方が共通コンタクト数が多く，良い結果が得られた．提案手法と文献 [21] の手法を比較すると，32 組中 19 組において提案手法の方が共通コンタクト数が多く，良い結果が得られた．

論文上の数値での比較結果ではあるが，提案手法の方が良い結果が得られているといえる．ただし，Xu らのデータセットにおけるフラボドキシンに似た形状を持つ蛋白質のデータセットの組合せにおいて差が大きい．遺伝的アルゴリズムを用いた手法では，個体を交差するときにアラインメント数の増減が期待でき，初期個体のアラインメント結果に依存しにくい．一方，提案手法ではアラインメント

表 7　先行研究との比較

Table 7　Comparison of previous studies.

PDB 番号		共通コンタクト数			
蛋白質 A	蛋白質 B	提案手法	EO	文献 [9]	文献 [21]
1bpi	1knt	**161**	**161**	149	150
1bpi	2knt	175	**177**	174	174
1bpi	5pti	188	188	**189**	**189**
1knt	2knt	**185**	**185**	156	160
1knt	5pti	**161**	**161**	153	157
1vii	1cph	54	**57**	36	41
2knt	5pti	174	**176**	172	172
3ebx	1era	172	**177**	114	133
3ebx	6ebx	**197**	**197**	150	164
6ebx	1era	183	**185**	155	167
1b00A	1dbwA	297	162	259	**329**
1b00A	1nat	303	294	314	**363**
1b00A	1qmpC	**332**	301	322	325
1nat	1b00B	302	298	330	**367**
1nat	1dbwA	**378**	375	229	316
1nat	4tmyA	338	337	311	**355**
1qmpc	1b00B	300	291	**339**	338
1qmpc	4tmyA	337	328	323	**355**
4tmya	1b00B	262	253	302	**304**
1b00A	1bawA	170	162	124	**194**
1b00A	1byoA	166	157	119	**179**
1b00A	1dpsB	260	242	199	**310**
1nat	1amk	**267**	187	170	211
1nat	1dpsB	**281**	252	210	140
1qmpC	2pcy	**168**	165	124	128
1qmpA	8timA	**257**	212	172	227
4tmyA	1bawA	159	164	141	**194**
4tmyA	1amk	**237**	179	157	229
4tmyA	1dpsC	249	226	189	**292**
1bawA	1aw2B	**181**	147	116	163
1bawA	1b9bA	**195**	153	99	181
1bawA	1dpsB	**192**	179	116	127

数は初期個体から変化しないため，初期個体のアラインメント結果に依存してしまう．アラインメント数の適応的な変化を行うアルゴリズムの開発が必要だと考えられる．

また，本研究で用いた EO による発見的解法と文献 [9] のオリジナルの EO との比較を行った．本研究で用いた EO による発見的解法と文献 [9] のオリジナルの EO による発見的解法は初期解生成方法が異なるが，本研究で用いた EO による発見的解法の方が，32 組中 25 組が共通コンタクト数が多く，良い結果が得られた．この結果から，動的計画法による初期解生成が効果があることが分かる．

6.4　実験 2

実験 2 では，即時移動戦略と最良移動戦略を比較する．提案手法における近傍個体生成数を 100 と設定し，100 秒間，状態遷移を繰り返す．また，それぞれ 30 回ずつ実行

© 2013 Information Processing Society of Japan

表 8　移動戦略の比較（実験 2）
Table 8　Comparison of move strategies.

PDB 番号		移動戦略	
蛋白質 A	蛋白質 B	最良移動戦略	即時移動戦略
1bpi	1knt	160	**161**
1bpi	2knt	**174**	168
1bpi	5pti	**187**	185
1knt	2knt	185	**188**
1knt	5pti	160	**161**
1vii	1cph	**53**	42
2knt	5pti	**172**	166
3ebx	1era	**167**	157
3ebx	6ebx	**196**	185
6ebx	1era	**181**	174
1b00A	1dbwA	**289**	286
1b00A	1nat	**301**	295
1b00A	1qmpC	312	**314**
1nat	1b00B	289	**299**
1nat	1dbwA	**370**	368
1nat	4tmyA	**315**	301
1qmpc	1b00B	289	**304**
1qmpc	4tmyA	326	**342**
4tmya	1b00B	253	**270**
1b00A	1bawA	**159**	142
1b00A	1byoA	**156**	142
1b00A	1dpsB	**253**	233
1nat	1amk	**240**	219
1nat	1dpsB	**263**	261
1qmpC	2pcy	**155**	136
1qmpA	8timA	**240**	226
4tmyA	1bawA	**148**	124
4tmyA	1amk	**207**	194
4tmyA	1dpsC	238	**240**
1bawA	1aw2B	179	**180**
1bawA	1b9bA	**182**	162
1bawA	1dpsB	**182**	173

表 9　近傍個体生成数の比較（実験 3）
Table 9　Comparison of number of neighbors.

PDB 番号		共通コンタクト数		
蛋白質 A	蛋白質 B	10 個	50 個	100 個
1bpi	1knt	160	160	160
1bpi	2knt	173	174	174
1bpi	5pti	187	187	187
1knt	2knt	185	185	185
1knt	5pti	160	161	160
1vii	1cph	53	53	53
2knt	5pti	172	171	172
3ebx	1era	169	167	167
3ebx	6ebx	196	195	196
6ebx	1era	183	181	181
1b00A	1dbwA	286	285	289
1b00A	1nat	303	300	301
1b00A	1qmpC	322	320	312
1nat	1b00B	291	293	289
1nat	1qmpC	372	372	370
1nat	4tmyA	317	319	315
1qmpc	1b00B	288	295	289
1qmpc	4tmyA	324	326	326
4tmya	1b00B	255	253	253
1b00A	1bawA	159	161	159
1b00A	1byoA	155	157	156
1b00A	1dpsB	257	251	253
1nat	1amk	246	231	240
1nat	1dpsB	271	266	263
1qmpC	2pcy	159	160	155
1qmpA	8timA	235	240	240
4tmyA	1bawA	150	146	148
4tmyA	1amk	217	212	207
4tmyA	1dpsC	236	237	238
1bawA	1aw2B	176	175	179
1bawA	1b9bA	185	181	182
1bawA	1dpsB	186	183	182

し，得られた最良個体の共通コンタクト数の平均をとる．実験結果を**表 8** に示す．表 8 は 32 組の組合せを 1 つにまとめて掲載をしている．結果としては，ほぼ最良移動戦略での状態遷移の方が良い結果が得られ，22 組の組合せで値が優っている．

劣っていた組合せについては，最良の値をとり続けることで，逆に局所解に陥った可能性がある．即時移動戦略はランダム性が高い．よって，ランダム性を取り入れることで，このように局所解に陥る場合も局所解から抜け出せる可能性があるため，最良移動戦略にランダム性を取り入れることでこのようなデータに対応できるようにする必要がある．

6.5　実験 3

実験 3 では，提案手法における近傍個体生成数を評価する．提案手法における近傍個体生成数を 10，50，100 と設定し，100 秒間，状態遷移を繰り返す．また，それぞれ 30 回ずつ実行し，得られた最良個体の共通コンタクト数の平均をとる．実験結果を**表 9** に示す．表 9 は 32 組の組合せを 1 つにまとめて掲載をしている．近傍個体生成数を 10，50，100 と増やすと，若干ではあるが増えている組合せもあるが，大幅な変化はあまりない．これは，アラインメントは連続領域であるため，個体から生成される近傍個体のバリエーションが限定されるためだと考えられる．

6.6　実験 4

実験 4 では，提案手法と EO を用いた発見的解法とで最良個体の共通コンタクト数の比較をするが，実験 1 とは異なり，世代交代数を固定して比較を行う．同一世代交代数であれば，近傍個体を作成し評価回数が多くなる提案手法

© 2013 Information Processing Society of Japan

表 10　世代数固定での比較（実験 4）
Table 10　Comparison of number of neighbors.

PDB 番号		世代交代数 = 100 世代		世代交代数 = 1,000 世代		世代交代数 = 10,000 世代	
蛋白質 A	蛋白質 B	提案手法	EO	提案手法	EO	提案手法	EO
1bpi	1knt	161	155	161	155	161	155
1bpi	2knt	174	170	174	171	174	171
1bpi	5pti	188	188	188	188	188	188
1knt	2knt	185	185	185	185	185	185
1knt	5pti	161	156	161	152	161	155
1vii	1cph	53	45	54	50	54	52
2knt	5pti	172	169	172	168	172	170
3ebx	1era	176	148	178	166	178	161
3ebx	6ebx	197	192	197	195	197	197
6ebx	1era	184	180	184	184	184	184
1b00A	1dbwA	292	249	312	257	321	258
1b00A	1nat	311	274	318	279	320	277
1b00A	1qmpC	324	283	331	296	331	285
1nat	1b00B	316	280	322	278	321	279
1nat	1qmpC	371	342	371	358	371	361
1nat	4tmyA	316	291	329	303	329	301
1qmpc	1b00B	311	283	318	285	320	291
1qmpc	4tmyA	340	314	350	312	350	315
4tmya	1b00B	278	233	287	238	290	235
1b00A	1bawA	151	131	156	136	161	137
1b00A	1byoA	146	132	156	135	161	140
1b00A	1dpsB	239	204	269	208	273	220
1nat	1amk	235	160	264	164	278	164
1nat	1dpsB	261	216	294	222	294	235
1qmpC	2pcy	145	132	153	135	155	143
1qmpA	8timA	232	185	275	190	308	189
4tmyA	1bawA	136	118	147	121	160	136
4tmyA	1amk	205	141	219	133	238	144
4tmyA	1dpsC	246	217	264	221	268	221
1bawA	1aw2B	184	133	200	134	212	133
1bawA	1b9bA	168	120	186	131	196	137
1bawA	1dpsB	184	162	190	164	192	162

の方がより多くの計算時間を必要とするが，同一世代での違いを比較するという観点で比較を行う．実験では，100 世代，1,000 世代，10,000 世代と世代交代を繰り返す．提案手法では，近傍個体生成数を 100 と設定している．それぞれ 30 回ずつ実行し，得られた最良個体の共通コンタクト数の平均を求める．

表 10 に実験結果を示す．世代交代数が固定であると，近傍個体を作成し評価回数が多くなる提案手法の方が優位である．すでに，100 世代目において，提案手法の方が EO を用いた発見的解法と比較すると優れた結果となっている．提案手法と EO を用いた発見的解法は世代交代数を増やすと，増加していく傾向ではあるが，両者に数値に差があり，EO を用いた発見的解法は初期個体周辺の探索しか行うことができておらず，局所最適解に陥っているが，提案手法は 100 世代の段階でそこから抜け出しているため数値が大きく改善されている．

6.7　実験 5

網羅的な組合せで提案手法の有効性を確認するために，オリジナルの Skolnick データセット 40 件を用い，すべての蛋白質立体構造データの組合せで評価を行った．蛋白質 A と蛋白質 B となった場合を区別して組合せを 1,560 組と，区別しない 780 組で，提案手法と EO を用いた発見的解法とで最良個体の共通コンタクト数の比較をする．EO を用いた発見的解法では，100 秒間，世代交代を繰り返す．提案手法では，近傍個体生成数を 100 と設定し，同じく，100 秒間，世代交代を繰り返す．また，それぞれ 3 回ずつ実行し，得られた最良個体の共通コンタクト数の平均を求める．

表 11 に 1,560 組の組合せにおける実験結果を掲載する．

© 2013 Information Processing Society of Japan

表 11　Skolnick データセットの全組合せ（実験 5）
Table 11　All matching of Skolnick data set.

PDB 番号	提案手法	EO	PDB 番号	提案手法	EO
1b00	39	0	2b3i	39	0
1dbw	39	0	2pcy	35	4
1nat	37	2	2plt	39	0
1ntr	39	0	1amk	39	0
1qmpA	36	3	1aw2	39	0
1qmpB	36	3	1b9b	39	0
1qmpC	35	4	1btm	39	0
1qmpD	36	3	1htiA	39	0
4tmyA	39	0	1tmh	39	0
4tmyB	37	2	1tre	39	0
3chy	39	0	1tri	39	0
1rn1A	37	2	3ypi	39	0
1rn1B	37	2	8tim	39	0
1rn1C	37	2	1ydv	39	0
1baw	35	4	1dps	39	0
1byoA	36	3	1b71	39	0
1byoB	35	4	1bcf	39	0
1kdi	39	0	1fha	39	0
1nin	39	0	1ier	39	0
1pla	39	0	1rcd	39	0

紙面の都合上，各蛋白質立体構造データにおいて，共通コンタクト数の平均値が大きい方の総数を記載する．たとえば，1b00 であれば，提案手法が蛋白質立体構造データ 1b00 と他の 39 件の蛋白質立体構造データとの 39 組の組合せにおいて，39 組が提案手法の方が優れていたことを示す．表 11 から分かるように，オリジナルの Skolnick データセットの総組合せでは，提案手法が EO を用いた発見的解法と比較して，1,560 組の組合せ中，劣っていたのは 38 件であった．また，780 組の組合せの場合は，780 組の組合せ中，提案手法が EO を用いた発見的解法と比較して劣っていた組合せは 19 件であった．

7. まとめ

本論文では改良版 EO を用いた CMO 問題の発見的解法を提案した．提案手法は，(1) 世代交代に改良版 EO を用いる，(2) 初期個体は動的計画法を用いて作成する，(3) 改良版 EO における状態遷移に即時移動戦略ではなく，最良移動戦略を用いる，という 3 つの特徴を持っている．提案手法を実際に実装し，文献 [9] と文献 [21] とで用いられている蛋白質立体構造データを用い，32 組の組合せにおいて，提案手法の評価を行った．評価実験の結果，提案手法は 27 組の蛋白質の組合せにおいて，EO を用いた発見的解法と同等，また，評価の高い最良解を求めることができ，CMO 問題に対する新しい解法として有効であることを示すことができた．特に，大規模で局所解に陥りやすい問題において，提案手法は EO を用いた発見的解法と比較して，良い解を求めることができることを確認できた．また，先行研究との比較も行い，提案手法の優位性を示すこともできた．これからの課題として，初期個体の生成方法の工夫，また，突然変異によるアラインメント数の動的な増減などがあげられる．また，多様性という観点では個体数が 1 つであるのは限界があるので，複数個体を用いた手法などの検討なども必要である．

謝辞　本研究の一部は，文部科学省・科学研究費補助金（若手研究（B），課題番号：23700124），日本学術振興会・科学研究費補助金（基盤研究（C），課題番号：20500137）の支援により行われた．

参考文献

[1] Branden, C.I. and Tooze, J.: *Introduction to Protein Structure*, Garland Publishing (1999).
[2] Lancia, G. and Istrail, S.: Protein Structure Comparison: Algorithms and Applications, *Mathematical Methods for Protein Structure Analysis and Design*, pp.1–33 (2003).
[3] Taylor, W.R. and Orengo, C.A.: Protein Structure Alignment, *Journal of Molecular Biology*, Vol.208, No.1, pp.1–22 (1989).
[4] Sippl, M.J. and Wiederstein, M.: A note on difficult structure alignment problems, *Bioinformatics*, Vol.24, No.3, pp.426–427 (2008).
[5] Godzik, A. and Skolnick, J.: Flexible algorithm for direct multiple alignment of protein structures and sequences, *Computer Applications in the Biosciences*, Vol.10, No.6, pp.587–596 (1994).
[6] Andonov, R., Malod-Dognin, N. and Yanev, N.: Maximum Contact Map Overlap Revisited, *Journal of Computational Biology*, Vol.18, No.1, pp.27–41 (2011).
[7] Boettcher, S. and Percus, A.G.: Extremal Optimization: Methods derived from Co-Evolution., *Proc. Genetic and Evolutionary Computation Conference*, pp.825–832 (1999).
[8] Boettcher, S. and Percus, A.: Nature's way of optimizing, *Artificial Intelligence*, Vol.119, No.1-2, pp.275–286 (2000).
[9] Lu, H., Yang, G. and Yeung, L.F.: Extremal Optimization for the Protein Structure Alignment, *Proc. 2009 IEEE International Conference on Bioinformatics and Biomedicine*, pp.15–19 (2009).
[10] 田村慶一，森　康真，北上　始：Extremal Optimization による調停グラフの交差数減少，情報処理学会論文誌：数理モデル化と応用，Vol.49, No.4, pp.105–116 (2008).
[11] 日本蛋白質構造データバンク（PDBj: Protein Data Bank Japan）：入手先 ⟨http://www.pdbj.org/index_j.html⟩.
[12] Goldman, D., Papadimitriou, C.H. and Istrail, S.: Algorithmic Aspects of Protein Structure Similarity, *Proc. 40th Annual Symposium on Foundations of Computer Science*, pp.512–522 (1999).
[13] Carr, R.D., Lancia, G., Istrail, S. and Genomics, C.: Branch-and-Cut Algorithms for Independent Set Problems: Integrality Gap and an Application to Protein Structure Alignment, SAND Report SAND2000-2171, Sandia National Laboratories (2000).
[14] Lancia, G., Carr, R., Walenz, B. and Istrail, S.: 101 optimal PDB structure alignments: A branch-and-cut algorithm for the maximum contact map overlap problem, *Proc. 5th Annual International Conference on Computational Biology*, pp.193–202 (2001).

© 2013 Information Processing Society of Japan

[15] Caprara, A. and Lancia, G.: Structural alignment of largesize proteins via Lagrangian relaxation, *Proc. 6th Annual International Conference on Computational Biology*, pp.100–108 (2002).
[16] Caprara, A., Carr, R.D., Istrail, S., Lancia, G. and Walenz, B.: 1001 Optimal PDB Structure Alignments: Integer Programming Methods for Finding the Maximum Contact Map Overlap, *Journal of Computational Biology*, Vol.11, No.1, pp.27–52 (2004).
[17] Xie, W. and Sahinidis, N.V.: A Reduction-Based Exact Algorithm for the Contact Map Overlap Problem, *Journal of Computational Biology*, Vol.14, No.5, pp.637–654 (2007).
[18] Strickland, D.M., Barnes, E. and Sokol, J.S.: Optimal Protein Structure Alignment Using Maximum Cliques, *Operations Research*, Vol.53, pp.389–402 (2005).
[19] Balaji, S., Swaminathan, V. and Kannan, K.: A Simple Algorithm for Maximum Clique and Matching Protein Structures, *International Journal of Combinatorial Optimization Problems and Informatics*, Vol.1, No.2, pp.2–11 (2010).
[20] Jain, B.J. and Lappe, M.: Joining softassign and dynamic programming for the contact map overlap problem, *Proc. 1st International Conference on Bioinformatics Research and Development*, pp.410–423 (2007).
[21] Lu, H., Yang, G. and Yeung, L.F.: A similarity matrix-based hybrid algorithm for the contact map overlaps problem, *Computers in Biology and Medicine*, Vol.41, pp.247–252 (2011).
[22] Svenson, P.: Extremal optimization for sensor report pre-processing, *Proc. SPIE*, Vol.5429, pp.162–171 (2004).
[23] Meshoul, S. and Batouche, M.: Robust Point Correspondence for Image Registration Using Optimization with Extremal Dynamics, *Proc. DAGM-Symposium 2002*, pp.330–337 (2002).
[24] Boettcher, S. and Percus, A.G.: Extremal Optimization at the Phase Transition of the 3-Coloring Problem, *Physical Review E*, Vol.69, 066703 (2004).
[25] Duch, J. and Arenas, A.: Community detection in complex networks using Extremal Optimization, *Physical Review E*, Vol.72, 027104 (2005).
[26] Zhou, T., Bai, W.-J., Cheng, L.-J. and Wang, B.-H.: Continuous extremal optimization for Lennard-Jones Clusters, *Physical Review E*, Vol.72, 016702 (2005).
[27] Taylor, W.R.: Protein structure comparison using iterated double dynamic programming, *Protein Science*, Vol.8, No.3, pp.654–665 (1999).
[28] Xu, J., Jiao, F. and Berger, B.: A parameterized algorithm for protein structure alignment, *Proc. 10th Annual International Conference on Research in Computational Molecular Biology, RECOMB '06*, pp.488–499 (2006).

中田 章宏　（学生会員）

2012 年広島市立大学情報科学部知能工学科卒業．同年同大学大学院情報科学研究科知能工学専攻修士課程入学，現在に至る．

田村 慶一　（正会員）

広島市立大学大学院情報科学研究科准教授．博士（情報科学）．1998 年九州大学工学部情報工学科卒業．2000 年同大学大学院システム情報科学研究科知能システム学専攻修士課程修了．2003 年同大学院システム情報科学府知能システム学専攻博士後期課程単位取得のうえ満期退学．広島市立大学情報科学部助手，広島市立大学大学院情報科学研究科助教，同講師を経て，2011 年より現職．データマイニングとその並列処理，進化計算に関する研究に従事．IEEE（SMC Hiroshima Chapter Secretary），日本データベース学会，人工知能学会，知能情報ファジイ学会各会員．

北上 始　（正会員）

広島市立大学大学院情報科学研究科教授．博士（工学）．1976 年東北大学大学院工学研究科博士前期課程修了．同年富士通株式会社，以後，富士通研究所，新生代コンピュータ技術開発機構（ICOT）主任研究員，国立遺伝学研究所客員助教授を経て，1994 年広島市立大学情報科学部教授，2007 年より現職．主な著書『データベースと知識発見』（共著），『生命情報学』（共著）ほか．1985 年情報処理学会 25 周年記念論文賞，2003 年日本工学教育協会論文・論説賞ほか．IEEE，ACM，情報処理学会（論文誌数理モデルと応用編集委員，一般情報教育委員会委員），電子情報通信学会，人工知能学会（評議員），日本バイオインフォマティクス学会，日本データベース学会（論文誌編集委員）各会員．

高橋 誉文

2008 年広島市立大学情報科学部知能情報システム工学科卒業．2010 年同大学大学院情報科学研究科博士前期課程修了．同年同博士後期課程進学．

© 2013 Information Processing Society of Japan