DNA データベースに対する効率的な相関問合せ手法の提案

八木　隆一†　　直井　悠馬†　　塩川　浩昭††

† 筑波大学大学院理工情報生命学術院　〒 305–8577 茨城県つくば市天王台 1-1-1
†† 筑波大学計算科学研究センター　〒 305–8577 茨城県つくば市天王台 1-1-1
E-mail: †{yagi,naoi}@kde.cs.tsukuba.ac.jp, ††shiokawa@cs.tsukuba.ac.jp

あらまし　DNA データベースに対する相関問合せとは，クエリとして DNA シーケンスが与えられたときクエリと共起して出現する部分シーケンスを検出する問題であり，生命情報科学分野で広く応用されている．この問題において DNA データベース内の全てのシーケンスの部分シーケンスが候補となるため，大量の DNA シーケンスからなる大規模 DNA データベースに対する相関問合せでは計算コストが大きくなる．本論文ではクエリとの相関値が閾値以上となる部分シーケンスを得ることを目的とする．本論文ではそのためのナイーブな手法ならびにその高速化手法を提案し，実データを用いた評価実験により提案手法の有効性を検証した．

キーワード　時系列データ処理，データ構造・索引，問合せ処理

1 序　　論

近年文字列に関する様々な研究が多くされている．文字列の類似度や相関に関する研究は，データベース分野や生命情報科学分野等において基本的な問題であり，データクリーニングやゲノムシーケンスアセンブリングなどに数多く採用されている．例えば，地球上のいかなる生物の DNA も A，C，G，T の 4 文字で構成される文字列として表すことができ，生物学的に近い種は似たような DNA シーケンスとなる可能性がある．そのため DNA シーケンスの類似度や相関に関する問題は文字列処理と同等の問題として考えることができる [4]．また，膨大な情報量から目的に合わせた有用な情報を得るためのデータマイニング等でも文字列処理に関する技術は欠かせないものとなっている．本論文の目的は DNA データベースに対する相関問合せ手法を提案することである．複数の文字列系列から構成された DNA データベースにおいて，クエリとして DNA シーケンスを与える場合を考える．このとき，クエリと共起して出現する DNA の部分系列構造を列挙することを目指す．

相関問合せにおける課題は，DNA データベース内の全ての部分系列構造が解候補となる点にある．ある DNA シーケンス s の長さを m としたとき，s は $\frac{m(m+1)}{2} = \mathcal{O}(m^2)$ 個の部分系列構造を持つ．すなわち，DNA データベースに登録された DNA シーケンスの数を N としたとき，相関問合せは (Nm^2) 個の解候補の中からクエリと共起して出現する部分系列構造を探索する必要がある．近年，次世代シーケンサーの登場により，DNA シーケンスの長さは長くなるとともに，DNA データベースに登録される DNA シーケンスの数も膨大となっている．ゆえに，DNA データベースに対する相関問合せは膨大な計算コストを必要とする．

相関問合せの高速化は，これまでグラフデータベースに関する研究が中心となって進められてきた [1–3]．例えば，文献 [1] ではクエリとの相関値が閾値より大きくなる部分グラフをデータベースから列挙する手法を提案している．また，文献 [2] ではクエリとの相関値が最も高くなる k 個の部分グラフを検出する手法が提案されている．しかしながら，本研究の対象とする DNA データベースにおいては相関問合せの高速化に関する議論が十分にされいない．

そこで本論文では DNA データベースにおける相関問合せの高速化手法を提案する．具体的には，クエリとなる DNA シーケンスとある閾値を与えたとき，クエリに対する相関値が閾値よりも大きくなる部分系列構造を DNA データベースから効率的に全列挙する手法を考える．前述したグラフデータベースに対する相関問合せの高速化手法では，問合せ処理を効率化するために，解候補となる部分グラフを高速に絞り込むための枝刈り手法を活用している．本論文ではこのアプローチを DNA データベースにおける相関問合せの効率化に応用する．具体的には，提案手法では DNA データベースにおいて部分系列構造の出現頻度に着眼し，この出現頻度からクエリと部分系列構造の出現頻度の下限値を推定する．その後，この下限値を活用して，閾値を満たさない部分系列構造を解候補から貪欲的に除外していく．本論文では実際の DNA データベースを用いて部分系列構造をすべて計算するナイーブな手法と提案手法の比較を行った．その結果として，提案手法はナイーブな手法と比較して 6.21 倍高速に解を列挙できることを確認した．

2 基 本 事 項

2.1 前 提 知 識

本研究で対象とする文字列は A, C, G, T の 4 文字で構成される DNA シーケンスである．本論文では DNA シーケンスを s とし，DNA シーケンス s の長さ（s に含まれる文字の数）を m_s とする．ただし，表現を簡潔にするため，文脈上明らかな場合は m_s を m と表記する．本論文では DNA シーケンス s の任意の部分文字列を s の部分系列構造と呼ぶ．s の取り得る全ての部分系列構造からなる集合を $\mathcal{B}(s)$ とする．2 つの DNA シーケ

ンス s と s' が与えられたとき，$s \in \mathcal{B}(s')$ であるならば，$s \subseteq s'$ と表記する．N 個の DNA シーケンスで構成される DNA データベースを $\mathcal{D} = \{s_1, s_2, \ldots, s_N\}$ とし，$\mathcal{D}$ に含まれるシーケンスの数を $|\mathcal{D}|$ と表す．すなわち $|\mathcal{D}| = N$ である．また，$\mathcal{D}$ においてシーケンス s を含むシーケンスの集合を s の射影データベースと呼び，$\mathcal{D}_s$ と表す．すなわち，$\mathcal{D}_s = \{s' \in \mathcal{D} \mid s \subseteq s'\}$ である．

本論文では $\mathcal{D}$ におけるシーケンス s の出現頻度の指標として支持度 $supp(s;\mathcal{D}) = \frac{|\mathcal{D}_s|}{|\mathcal{D}|}$ を用いる．本論文では特に断りがない限り，$supp(s;\mathcal{D})$ を $supp(s)$ と表す．また，2 つのシーケンス s_1, s_2 が同時に出現する頻度の指標である結合支持度を $supp(s_1, s_2) = \frac{|\mathcal{D}_{s_1} \cap \mathcal{D}_{s_2}|}{|\mathcal{D}|}$ と表す．支持度には逆単調性があり，$s_1 \subseteq s_2$ であるとき，$supp(s_2) \leqq supp(s_1)$ である．さらに，結合支持度の定義から $supp(s_1, s_2) \leqq supp(s_1)$ が明らかに成り立つため，$supp(s_1, s_2) \leqq supp(s_2)$ となることに注意されたい．

本研究では 2 つのシーケンスの相関値の指標としてピアソン相関係数 [11] を採用する．定義を以下に示す．

定義 1.（ピアソン相関係数）

$$\phi(s_1, s_2) = \frac{supp(s_1, s_2) - supp(s_1)supp(s_2)}{\sqrt{supp(s_1)supp(s_2)(1 - supp(s_1))(1 - supp(s_2))}}$$

$\phi(s_1, s_2)$ は $[-1, 1]$ の範囲の値をとる．$\phi(s_1, s_2)$ が正の値になるとき s_1 と s_2 は正の相関があり，負の値になるときは負の相関がある．また，$supp(s_1), supp(s_2)$ が 0 または 1 となるとき，$\phi(s_1, s_2) = 0$ と定義する．

2.2 問題定義

本研究では DNA データベースにおける相関問合せについて考える．この相関問合せではクエリとして DNA シーケンスと任意の閾値を与え，DNA データベースから閾値以上の相関値を示す部分系列構造を全て列挙する．より詳細な問題定義を以下に示す．

定義 2.（DNA データベースに対する相関問合せ）
DNA データベース $\mathcal{D} = \{s_1, s_2, \ldots, s_N\}$，クエリシーケンス q，閾値 $\theta \in [-1, 1]$ が与えらたとき，相関問合せは部分系列構造集合 $\mathcal{T}_\theta(q)$ を検出する問題である．ただし，$\mathcal{T}_\theta(q)$ は以下のように定義する．

$$\mathcal{T}_\theta(q) = \left\{ s \in \bigcup_{s' \in \mathcal{D}} \mathcal{B}(s') \mid \phi(q, s) \geqq \theta \right\}.$$

1 節で述べたように，定義 2 に示す問題の計算量は $\mathcal{O}(Nm^2)$ である．しかしながら，これまでこの問題を効率的に解くための手法は我々の知る限り提案されていない．

3 提案手法

本節では定義 2 で示した相関問合せ問題を高速に計算する提案手法を説明する．提案手法の基本的なアイデアは部分系列構造の出現頻度に着眼することで解になり得ないものを枝刈りすることである．本節ではまず 3.1 節において提案手法の用いる枝刈り手法の詳細を説明する．その後，3.2 節において提案手法のアルゴリズムの全体像を示す．

3.1 部分系列構造の枝刈り

本節では提案手法が採用する部分系列構造の枝刈り手法についてその詳細を説明する．

クエリの射影データベースの構築：クエリ q とある部分系列構造 s について考える．提案手法では $\phi(q, s) \geqq \theta$ を効率的に見つけるために，q と s が正の相関を持つ必要があることは自明である．このことから，q と s は DNA データベース $\mathcal{D}$ において少なくとも 1 回以上は共起して出現する必要があることは明らかである．したがって，提案手法ではまず，解候補の探索を $\mathcal{D}$ から $\mathcal{D}_q$ へと限定する．$|\mathcal{D}_q| \leqq |\mathcal{D}|$ であるため，DNA データベース全体から探索を行う場合と比較して，効率的な探索が行える可能性が高い．

$\mathcal{D}_q$ を $\mathcal{D}$ から構築するためには，データベース内の各 DNA シーケンス s' に対して，$q \subseteq s'$ であるかどうかを判定する必要がある．この判定処理は単純には $\mathcal{O}(m^2)$ の計算時間を必要とするため，本研究では文字列の部分系列包含判定手法 [7] を採用する．この判定手法は $\mathcal{O}(m)$ での包含判定を可能とするものである．この包含判定処理については他の手法に代替することも可能であり，例えば文献 [8] で提案されている部分系列構造のパターン列挙手法などを利用することもできる．

出現頻度下限値に基づく枝刈り：上述した射影データベース $\mathcal{D}_q$ は，クエリ q が DNA データベース内において普遍的に包含される場合，データベースのサイズを削減できないため，処理の十分な効率化ができない．そこで，提案手法はクエリ q と部分系列構造 s が正の相関値を持つ場合の，出現頻度の下限値を推定することで，探索対象となる DNA シーケンスの数をさらに削減する．本研究ではまず以下の定理を示す．ただし，紙面の都合により定理の証明は省略する．

定理 1.（出現頻度の下限値）
クエリ q，相関値の閾値 θ が与えられているものとする．このとき，ある部分系列構造 s が $\phi(q, s) \geqq \theta$ となるための必要条件は以下の不等式が成立することである．

$$supp(s;\mathcal{D}_q) \geqq \frac{lower(q, s)}{supp(q)}.$$

ただし，$lower(q, s)$ は以下のように定義される関数である．

$$lower(q, s) = \frac{supp(q)}{\theta^{-2}(1 - supp(q)) + supp(q)}.$$

定理 1 は，$\mathcal{D}_q$ から解を列挙する際に s の出現頻度が $\frac{lower(q,s)}{supp(q)}$ より大きなもののみ計算すれば良いことを示している．これにより，提案手法は $\mathcal{D}_q$ に含まれる全ての部分系列構造に対して相関値を計算する必要がなくなり，計算の効率化が可能となる．

規則に基づく枝刈り：提案手法では探索をさらに効率化するために，以下の定理に基づくヒューリスティックな枝刈り規則を設ける．各定理の証明は紙面の都合により省略する．

定理 2.（枝刈り規則 1）
クエリ q と部分系列構造 s，ならびに相関値の閾値 θ が与えられたとき，$q \subseteq s$ であるならば，$\phi(q, s) \geqq \theta$ である．

定理 3.（枝刈り規則 2）
クエリ q と相関値の閾値 θ が与えられたとき，2 つの部分系列構造 s と s' が $s \subseteq s'$ であり，$supp(q,s) = supp(q,s')$ を満たすとする．このとき，$\phi(q,s) < \theta$ であるならば $\phi(q,s') < \theta$ が成立する．

定理 4.（枝刈り規則 3）
クエリ q と相関値の閾値 θ が与えられたとき，2 つの部分系列構造 s と s' が $s \subseteq s'$ であるとする．このとき，$supp(q,s') < f(q,s)$ ならば，$\phi(q,s') < \theta$ が成立する．ただし，$f(q,s)$ は以下のように定義される関数である．

$$f(q,s) = \theta\sqrt{supp(q)(1-supp(q))supp(g)(1-supp(g))} + supp(q)supp(g)$$

3.2 アルゴリズム

提案手法のアルゴリズムについて説明する．提案手法ではクエリ q が与えられたとき，クエリ q の射影データベース $\mathcal{D}_q$ を構築する．その後，定理 1 に示した下限値に基づいて，解候補となる部分系列構造を列挙して，定理 2, 3, 4 の枝刈り規則を適用しながら効率的に閾値以上の相関値を持つ部分系列構造を検出する．

詳細なアルゴリズムを Algorithm 1 に示す．Algorithm 1 は DNA データベース $\mathcal{D} = \{s_1, s_2, \ldots, s_N\}$，クエリ q，および相関値の閾値 θ が与えられたとき，相関部分系列構造 $\mathcal{T}_\theta(q) = \{s \in \mathcal{D} \mid \phi(q,s) \geqq \theta\}$ を検出する．まず，クエリ q の射影データベース $\mathcal{D}_q$ を取得する (1 行目)．次に，$\mathcal{D}_q$ に対して定理 1 を適用して，部分系列構造集合 $\mathcal{C}$ を取得する (2 行目)．このとき，部分系列構造の出現頻度を効率的に計算するために，文献 [5] で提案された頻出部分系列列挙手法を利用する．次に，部分系列構造 $s \in \mathcal{C}$ に対して定理 2, 3, 4 に基づく枝刈りを実行する（3〜12 行目）．まず（4〜6 行目）では，s に対して定理 2 を適用する．s が定理 2 の条件を満たした場合，s を $\mathcal{T}_\theta(q)$ に追加する．定理 2 の条件を満たさない場合は，次の手順により定理 3 と定理 4 を適用しながら解を探索する（6〜12 行目）．まず，$\phi(q,s)$ を計算し，相関値が θ 以上となる場合，s を $\mathcal{T}_\theta(q)$ に追加する（7〜9 行目）．次に，定理 3, 4 を用いて，$\mathcal{C}$ 内の未探索の部分系列構造 s' の枝刈りを行う（10〜12 行目）．

Algorithm 1 Threshold-based method
Input: A DNA database $\mathcal{D}$, query q, threshold θ.
Output: $\mathcal{T}_\theta(q) = \{s \in \mathcal{D} : \phi(q,s) \geqq \theta\}$.
1: Obtain $\mathcal{D}_q$, $\mathcal{T}_\theta(q) = \emptyset$;
2: Obtain $\mathcal{C}$ by Theorem 1;
3: **for each** $s \in \mathcal{C}$ in length-descending order **do**
4: **if** s contains q **then**
5: Add s to $\mathcal{T}_\theta(q)$;
6: **else**
7: Compute $\phi(q,s)$;
8: **if** $\phi(q,s) \geqq \theta$ **then**
9: Add s to $\mathcal{T}_\theta(q)$;
10: **else**
11: Remove $s' \in \mathcal{C}$ by Theorem 3;
12: Remove $s' \in \mathcal{C}$ by Theorem 4;
13: **return** $\mathcal{T}_\theta(q)$;

4 評価実験

本節では現実の DNA データベースを用いて，提案手法の問合せ処理時間と精度を評価する．

4.1 実験設定

評価実験では提案手法とナイーブなアルゴリズムの比較を行う．ナイーブなアルゴリズムの詳細は次のとおりである．
(1) DNA シーケンス $s \in \mathcal{D}$ に対して部分系列構造集合 $\mathcal{B}(s)$ を取得する．
(2) 各部分系列構造 $s' \in \mathcal{B}(s)$ に対して相関値 $\phi(q,s')$ を計算し，$\phi(q,s') \geqq \theta$ であれば s' を $\mathcal{T}_\theta(q)$ に追加する．
(3) $\mathcal{D}$ 内の全ての DNA シーケンスに対して (1)〜(2) を繰り返す．

表 1: データセットの詳細

データセット	N	平均系列長	最小系列長	最大系列長
GEN20kS	20,000	5,000	4,829	5,109
GEN20kM	20,000	10,000	9,843	10,154

提案手法およびナイーブなアルゴリズムは C++を用いて実装し，-O3 の最適化オプションでコンパイルをした．実験は DELL PowerEdge R740XD Intel Xeon Platinum 8280M 2.7 GHz CPU と 768GB RDIMM を搭載した Linux サーバー上で実行した．

評価実験では文献 [12] で提供されている実際の DNA データベースを用いる．本実験では平均系列長の異なる 2 つのデータベースを採用し，系列長に対する性能の変化を観察する．データセットの詳細は表 1 に示す．

4.2 問合せ処理時間の評価

提案手法の問合せ処理時間の評価を行う．本実験で計測した問合せ処理時間は $\mathcal{D}_q$ の生成，候補部分系列構造の検出，諸定理とルールを利用した候補部分系列構造から解の検出にかかる時間の総計である．本実験では 10 種類のクエリを用意し，1 つのクエリに対して相関問合せにかかる平均時間を報告する．図 1, 2 に結果を示す．

実験結果より閾値 θ の値が大きくなるほど問合せ処理時間が短くなっていることがわかる．これは閾値の増加とともに Algorithm 1 の 2 行目で生成される解候補の数が少なくなるためであると考えられる．また，ナイーブな手法は閾値の値に関わらず全ての部分系列構造とクエリとの相関値を求めるため，問合せ処理時間は一定である．その値は図 1 において 24,402 秒，図 2 において 41,483 秒であり，いずれの場合も提案手法がナイーブな手法に比べて問合せ処理時間の観点で優れていることが明らかである．今回計測した閾値において，最も問合せ処理時間がかかっている $\theta = 0.3$ のときでもナイーブな手法に対して提案手法は約 6.21 倍の処理時間である．

さらに定理 3, 4 により多くの解候補 $s \in \mathcal{C}$ が除外されてい

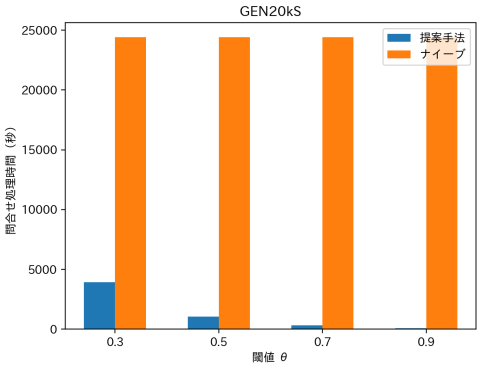

図 1: GEN20kS に対する問合せ処理時間

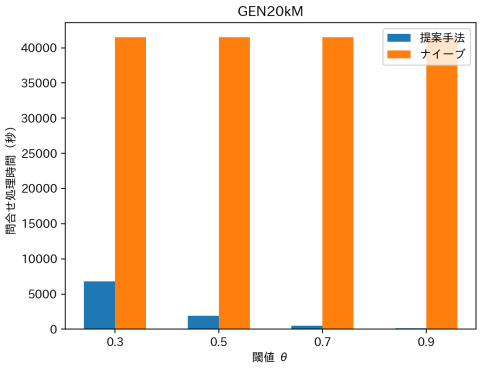

図 2: GEN20kM に対する問合せ処理時間

表 2: 問合せ処理精度：*racall*

	$\theta = 0.3$	$\theta = 0.5$	$\theta = 0.7$	$\theta = 0.9$
GEN20kS	1.0	1.0	1.0	1.0
GEN20kM	1.0	1.0	1.0	1.0

ることが確認できた．Algorithm 1 の 7 行目における相関値 $\phi(q, s)$ の計算では，$supp(s)$ を求める必要がありコストがかかる．そのため定理 3, 4 により解候補を効率的に減らすことで，提案手法の問合せ処理時間を大幅に短くしている．

4.3 問合せ処理精度の評価

ナイーブな手法に対し，提案手法はいくつかの諸定理により相関値の計算を行う部分系列構造の数を減らしている．そこで提案手法によって得た解とナイーブな手法によって得た解を比較することで，提案手法の問合せ処理精度の評価を行った．提案手法が $\phi(q, s) \geqq \theta$ と判定し，実際に成り立つ部分系列構造の数を TP とし，$\phi(q, s) < \theta$ と判定したが，実際には $\phi(q, s) \geqq \theta$ であった部分系列構造の数を FN とする．このとき，

$$recall = \frac{TP}{TP + FN}$$

を求める．表 2 にその結果を示す．

表 2 より提案手法はナイーブな手法と同様の解を得ることが確認できた．

5 結　　論

本研究では DNA データベースに対する相関問合せ手法の提案を行った．本論文では枝刈りに基づく手法をを示した．実データを用いた評価実験により，提案手法はナイーブなアルゴリズムに比べて実行時間の観点において優位性があることを実験的に確認するとともに，DNA データベースへの応用ができることを確認した．

謝　　辞

本研究の一部は，JST さきがけ (JPMJPR2033) ならびに JSPS 科研費 (JP22K17894) の支援を受けたものである．

文　　献

[1] Y.Ke, J.Cheng, and W.Ng. Correlation search in graph databases. In Proc. of KDD, pages 390-399, New York, NY, USA, 2007. ACM.

[2] Yiping Ke, James Cheng, and Jeffrey Xu Yu. Top-k Correlative Graph Mining. In Proc. SIAM International Conference on Data Mining , Pages 1038-1049, 2009.

[3] 直井悠馬, 真次彰平, 塩川浩昭．グラフデータベースに対する相関問合せ手法の高速化．第 14 回データ工学と情報マネジメントに関するフォーラム (DEIM 2022), G34-1, 2022 年 2 月 27 日-3 月 2 日.

[4] Ryuichi Yagi, Hiroaki Shiokawa. Fast Top-k Similar Sequence Search on DNA Databases. In Proceedings of the 24th International Conference on Information Integration and Web Intelligence (iiWAS2022), pp.145-150, Virtual Conference, November 2022.

[5] Yuta Tsuboi. Mining Frequent Substrings. In IPSJ SIGNL-158 (in Japanese), 2003.

[6] Jian Pei, Jiawei Han, B.Mortazavi-Asl, Q.Chen H.Pinto, U.Dayal, and M.-C. Hsu. Prefixspan: Mining sequential patterns efficiently by prefix-projected pattern growth. In Proc. of International Conference on Data Engineering, pp.215-224, Heidelberg, Germany, April 2001.

[7] D.E. Knuth, J.H. Morris, and V.R. Pratt, Fast pattern matching in strings. SIAM Journal on Computing, Vol. 6, No. 2, pp.323-350, June 1977.

[8] Rakesh Agrawal and Ramakrishnan Srikant. Mining sequential patterns. In Proc. of International Conference of Data Engineering, pp.3-14. IEEE Press, 6-10 1995.

[9] Xifeng Yan and Jiawei Han. gspan: Graph-based substructure pattern mining. In Proc. of International Conference on Data Mining, 2002.

[10] Akihiro Inokuchi, Takashi Washio, and Hiroshi Motoda. An Apriori-based Algorithm for Mining Frequent Substructures from Graph Data. In Proc. of Conference on Principles and Practice of Knowledge Discovery and Data Mining, 2000.

[11] H.Reynolds. The analysis of cross-classifications. The Free Press, New York, 1977.

[12] Haoyu Zhang, Qin Zhang. 2019. MinJoin: Efficient Edit Similarity Joins via Local Hash Minima. In The 25th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD'19), August 4-8, 2019, Anchorage, AK, USA. ACM, New York, NY, USA, 11 pages.