Vol.2016-MPS-108 No.32
Vol.2016-BIO-46 No.32
2016/7/6

最適グラフ発見に基づく蛋白質表面からの結合部位抽出におけるグラフの抽象化

八鳥 真弥[1,a)]　三井 拓真[1]　大川 剛直[1,b)]

概要：蛋白質は局所的な部位において低分子化合物 (リガンド) と結合して，機能を発現するものが多く，局所的な構造の比較が蛋白質の機能を知るための手掛りになる．本研究では，蛋白質表面をグラフで表し，類似グラフ探索を行う．そして，結合部位になる可能性が高い構造に高い評価値を与える評価関数により，最適グラフを発見することで，結合部位を予測する．類似グラフ探索の際，計算コスト削減のために，グラフを抽象化する手法を提案する．提案手法を蛋白質の結合部位予測へ適用した結果，計算コストを削減し，精度の高い予測を実現できた．

Graph Summarization in Binding Site Extraction from Protein Molecular Surface Based on Detecting Optimal Graph

Masaya Yatori[1,a)]　Takuma Mitsui[1]　Takenao Ohkawa[1,b)]

1. はじめに

蛋白質の多くは，低分子化合物 (リガンド) などの他の化合物と結合することでその機能を発現する．同一のリガンドが結合する結合部位は構造が類似していることから，蛋白質の結合部位の構造の類似性が，結合部位の特定や結合に起因する機能を解明する上での手掛かりとなると考えられている．また，結合部位は，蛋白質の凹んだ部分 (以下，ポケットと呼ぶ) に存在することが多い．そこで，本研究では蛋白質表面のポケットに対応する表面データを対象に，その局所的な類似性に着目して，結合部位を予測する手法を開発する．

複数の蛋白質に共通して見られる表面構造のことを表面モチーフと呼ぶ [1]．同一のリガンドに結合する蛋白質の結合部位は類似しているため，これらの蛋白質の間で表面モチーフが発見された場合，その表面モチーフが結合部位であると考えられる．しかしながら，共通する構造という観点のみに基づいて抽出した表面モチーフは，同一のリガンドに結合しない蛋白質においても普遍的に見られる構造である可能性がある．すなわち，表面モチーフについて，それが結合部位に固有の構造であるのか，あるいは，普遍的に見られる構造であるのかを識別する必要がある．そこで本研究では，結合部位を予測したい蛋白質 (以下，対象蛋白質と呼ぶ) を，その他の複数の蛋白質 (以下，参照蛋白質と呼ぶ) と比較するにあたり，参照蛋白質を，対象蛋白質と同一のリガンドに結合する蛋白質グループ (以下，ポジティブグループと呼ぶ) とそれ以外の蛋白質グループ (以下，ネガティブグループと呼ぶ) に分類し，前者のグループに頻出し，後者からはあまり発見されないような局所部位を結合部位として予測する．

提案手法では，蛋白質の表面データをグラフで表し，類似グラフ探索を行うことにより，頻出部位の発見を試みる．対象蛋白質のグラフの部分グラフを列挙し，それぞれの部分グラフについて類似するグラフを参照蛋白質から探索する．このとき，ポジティブグループの多くの蛋白質から類似グラフが発見でき，ネガティブグループの蛋白質からは類似グラフが発見されないような部分グラフに高い評価を与える評価関数を導入し，最も高い評価を与えられた部分グラフを含むポケットを結合部位と予測する．

[1] 神戸大学大学院 システム情報学研究科
Kobe University, Graduate School of System Informatics
[a)] yatori@cs25.scitec.kobe-.ac.jp
[b)] ohkawa@kobe-u.ac.jp

Vol.2016-MPS-108 No.32
Vol.2016-BIO-46 No.32
2016/7/6

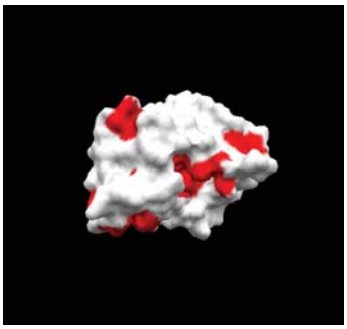

図 1　Extracted pockets from protein surface(red parts)

蛋白質表面を記述したグラフはサイズが大きいため，グラフ探索に要する計算コストが高く，将来的に想定されるデータの拡張や，蛋白質表面全体の網羅的な探索などのためにも効率化が必要である．近年，巨大なグラフを扱うための 1 つの方法として，グラフの抽象化に関する研究がいくつかなされている [2], [3]．そこで，類似する頂点同士を 1 つの頂点に集約してできる抽象グラフを用いて，類似グラフ探索を行うことで計算コストの削減をする手法を導入する．類似グラフ探索では，頂点数が少ない部分グラフから頂点を追加して，より大きい部分グラフを探索していく．頂点数が少ない部分グラフの類似グラフは，参照蛋白質において大量に発見できるため，それらすべてを探索する必要があり，計算コストがかかる．一方で，頂点数が少ない部分グラフはたまたま評価が良くなる可能性が高く，頂点数が多い部分グラフの方が結合部位となる可能性が高いことから，頂点数が少ない間は，抽象グラフを用いることで，探索するグラフの数を抑え，計算コストを削減し，精度の高い予測を目指す．

2. 最適グラフ発見による結合部位抽出

結合部位は結合現象に適した固有の構造を包含しているため，本研究では，同一のリガンドに結合する蛋白質に頻出し，それ以外の蛋白質ではあまり見られない構造を結合部位と予測する．蛋白質の構造データには，蛋白質の表面形状と物性に関するデータを集積したデータベースである eF-site[4] に登録されている蛋白質分子表面データを用いる．

2.1 蛋白質のポケット抽出

結合部位は蛋白質表面のポケットに存在することが多いことから，蛋白質表面のポケットに対応する表面データを対象に，結合部位を予測することを考える．ポケットの抽出にはインターネット上で利用できるポケット抽出ツールである CASTp[*1][5] を用いる．CASTp を用いて蛋白質分子表面のポケットの部分を抽出した例を図 **1** に示す．図中の赤色の箇所がポケットである．

[*1] http://sts.bioe.uic.edu/castp/

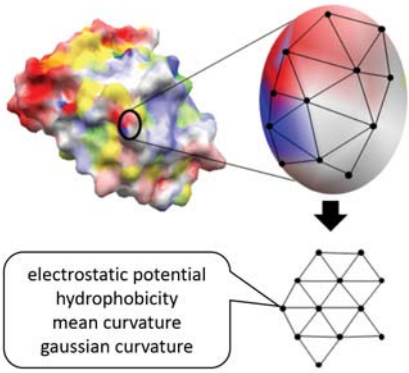

図 2　Graph represention of protein surface

2.2 蛋白質表面のグラフ表現

蛋白質のポケットの表面データをグラフで表す．eF-site に登録されている蛋白質は，efvet という XML ファイルとして格納されており，各頂点の 3 次元座標や疎水性，静電ポテンシャル，曲率などの物性情報と頂点間の辺の情報が記されている [6]．そこで，蛋白質表面のデータを，4 つの物性情報を持つ頂点とそれらを結ぶ辺から成るグラフで表すことができる．ポケット表面のデータをグラフで表すイメージを図 **2** に示す．

2.3 結合部位抽出手法

2.3.1 概要

本研究では，入力データとして，結合するリガンドによりグループ分けされている蛋白質の表面データを与える．入力データのうち 1 つの蛋白質を対象蛋白質 ts とし，その他の複数の蛋白質を参照蛋白質 $S = \{s_1, \cdots, s_m\}$ とする．参照蛋白質は，対象蛋白質と同一のリガンドに結合するポジティブグループと同一のリガンドに結合しないネガティブグループに分けられる．

まず，全ての蛋白質の分子表面からポケットを抽出し，グラフで表現する．次に，対象蛋白質の各ポケットの各部分グラフに対して，参照蛋白質から類似グラフを探索する．ポジティブグループ内で類似グラフを多く発見でき，ネガティブグループ内で類似グラフをあまり発見できない部分グラフに良い評価を与える評価関数により，最適グラフを発見する．

そして，対象蛋白質の各ポケットについて，列挙された部分グラフの中で最も良い評価関数の値をそのポケットのスコアとし，最も高いスコアが与えられたポケットを結合部位として予測する．図 **3** にそのイメージを示す．

2.3.2 評価関数

各参照蛋白質 s_i に対して，s_i がポジティブグループに属するとき，$\xi(s_i) = 1$ と表し，s_i がネガティブグループに属するとき，$\xi(s_i) = 0$ と表す．また，対象蛋白質 ts のポケットのある部分グラフ I に対して，s_i が I と類似した

Vol.2016-MPS-108 No.32
Vol.2016-BIO-46 No.32
2016/7/6

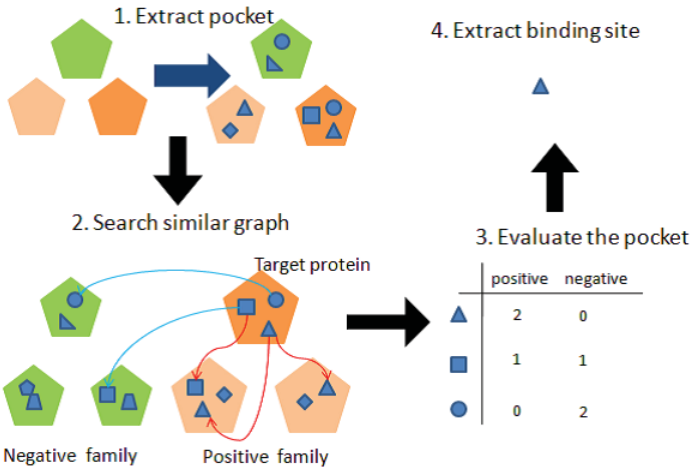

図 3　Overview of proposed method

グラフを持つとき $I(s_i)=1$，持たないとき $I(s_i)=0$ と表す．対象蛋白質のある部分グラフ I によって，参照蛋白質集合 S は，$S_j(I)=\{s\in S|I(s)=j\}(j=0,1)$ の 2 つの部分集合に分割される．つまり，S_1 が部分グラフ I に類似したグラフを持つ蛋白質の集合，S_0 が部分グラフ I に類似したグラフを持たない蛋白質の集合である．S_j の蛋白質の数を $|S_j|$，S_j のうちポジティブグループの蛋白質の数を $P_j(I)=\{s\in S|I(s)=j,\xi(s)=1\}\,(j=0,1)$ とし，その比率を $\theta_j(I)=P_j(I)/|S_j|$ とおく $(0\leq\theta_j(I)\leq 1)$．つまり，$\theta_1(I)$ が大きいほど，部分グラフ I に対する類似グラフが存在するポジティブグループの参照蛋白質が多いということである．また，$\theta_0(I)$ が小さいほど，部分グラフ I に対する類似グラフが存在しない参照蛋白質の多くがネガティブグループの蛋白質であると言える．各部分グラフ I に対する，これらの変数は**表 1** のように整理できる．なお，表中の $|\tau|$ はポジティブグループに属する蛋白質の数を表す．

この分割がポジティブグループとネガティブグループをうまく分割できるとき，対象蛋白質の部分グラフは結合部位である可能性が高いと考えられる．そのような評価が可能な関数を，文献 [7] で提案されている不純度関数 $\psi(x)=2x(1-x)\quad(0\leq x\leq 1)$ を元に考える．不純度関数は，$\psi(1/2)$ で最大値をとり，$\psi(0)=\psi(1)=0$ で最小値をとる上に凸な関数である．この関数を用いて，

$$G_{S,\xi}(I)=\psi(\theta_0(I))|S_0(I)|+\psi(\theta_1(I))|S_1(I)| \qquad (1)$$

のような，評価関数を考える．この $G_{S,\xi}(I)$ は，$\theta_1(I)$ が大きく，$\theta_0(I)$ が小さいほど値が小さくなる．つまり，部分グラフ I により，ポジティブグループの蛋白質とネガティブグループの蛋白質をうまく分割できていればいるほど値は小さく，結合部位となる可能性が高いといえる．しかし，式 (1) はネガティブグループで類似グラフを持つ蛋白質が頻出し，ポジティブグループで類似グラフを持つ蛋白質がほとんど出現しないような部分グラフも値が小さくなる．そこで，以下の条件を設け，そのような部分グラフが高い評価になることを防ぐ．

$$\frac{P_1(I)}{|\tau|}P_1(I)\geq\frac{|S_1(I)|}{|S|} \qquad (2)$$

表 1　Contingency table

	positive group	negative group	Σrow
match	$P_1(I)$		$\|S_1(I)\|$
unmatch	$P_0(I)$		$\|S_0(I)\|$
$\Sigma column$	$\|\tau\|$	$\|S\|-\|\tau\|$	$\|S\|$

2.3.3　類似グラフ探索と枝刈り

対象蛋白質の各ポケットのグラフ $p^1,\cdots,p^n$（n はその蛋白質に存在するポケットの数である）に関して，そのグラフのあらゆる部分グラフ I と類似しているグラフを，全ての参照蛋白質 $s_i(i=1,2,\cdots,m)$ の全てのポケットのグラフから探索する．gApprox[8] という拡張パターンの列挙方法を用いることで，対象蛋白質のグラフ $p^1,\cdots,p^n$ の全ての部分グラフを列挙する．これにより，対象蛋白質のポケットのグラフ内に存在するあらゆる部分グラフをもれなく，かつ重複なく，列挙することが可能である．そして，参照蛋白質のグラフから類似グラフ探索を完了した対象蛋白質の部分グラフ I に対して，評価関数を用いて評価を行う．

蛋白質のポケットのグラフサイズは大きいので，ポケット内の部分グラフの探索には膨大な計算を要する．そのため，類似グラフ探索時に，論文 [7] で提案されている枝刈りの考え方を導入する．グラフパターンを拡大するに従い，そのパターンに類似した参照蛋白質のグラフの数は単調減少する．このため，以降のパターン拡張時における式 (1) の評価関数の下限を計算できる．そして，現時点で類似グラフ探索を完了した全ての部分グラフ I における式 (1) の最小値と，計算した下限値を比較し，下限値が大きい場合には，そのパターンの拡張は行わないようにする．なぜなら，それ以上の拡張で，現時点の評価値よりもよくなる部分グラフは存在せず，最適グラフを見つけるうえでさらなる部分グラフの拡張は意味を成さないからである．これにより，探索空間の削減を行い，計算コストを削減する．

2.3.4　ポケットのスコアリング

探索された部分グラフの中で最も良い評価値をそのポケットのスコアとする．しかし，最適グラフを判定する際には，ポジティブグループでの類似グラフ発見蛋白質数とネガティブグループでの類似グラフ発見蛋白質数以外に，そのグラフパターンのサイズも考慮すべきである．なぜなら，サイズの小さなパターンの場合，より大きいパターンと比較して，偶然良い評価値になる可能性が高いと考えられるからである．しかしながら，上述した枝刈りを用いる場合にサイズを考慮した関数を用いると文献 [7] で述べられていた健全性を保持できない．そこで，類似グラフ探索後に，探索された全ての部分グラフを対象として，そのグラフのサイズを考慮した式 (3) の評価関数により新たに評

Vol.2016-MPS-108 No.32
Vol.2016-BIO-46 No.32
2016/7/6

価する．

$$G'_{S,\xi}(I) = \psi(\theta_0(I))|S_0(I)| + \psi(\theta_1(I))|S_1(I)|/h \qquad (3)$$

ここで，h は頂点数であり，グラフのサイズを表す．

探索された全ての部分グラフ I_{set} の評価値を式 (3) の評価関数により計算し，最も良い評価値をそのポケットのスコア $p^i_{score} = \max_{I \in I_{set}}\{G'_{S,\xi}(I)\}$ にする．対象蛋白質の全ポケット $p^1, \ldots p^n$ のスコアリングを終えた後，最も良いスコアのポケット $p_{max} = \max_i\{p^i_{score}\}$ を結合部位として出力する．なお，枝刈りが，式 (3) で定義した $G'_{S,\xi}(I)$ ではなく，式 (1) の $G_{S,\xi}(I)$ に基づいて実行されるため，$G'_{S,\xi}(I)$ に関して最適な部分グラフが見落とされる可能性がある．このため，関数 $G'_{S,\xi}(I)$ に関する最適部分グラフを厳密に得ることはできず，近似解として得られることに注意されたい．

3. グラフ抽象化による最適グラフ発見

蛋白質表面のポケットのグラフに関して，あらゆる部分グラフを列挙することは枝刈りを行ったとしても，計算コストが非常に大きくなる．そこで，提案手法では，グラフを抽象化することで、この課題の解決を試みる．

3.1 抽象グラフと頂点のクラスタリング

抽象グラフは，頂点に付与された物性情報が類似している頂点同士を 1 つに集約して再構成したグラフを意味する．抽象グラフの頂点（抽象頂点）には，それを構成する元のグラフの頂点の数の情報を付与する．そして，2 つの抽象頂点の各々を構成する元々の頂点間に辺が存在する場合は，抽象頂点間も辺で結ぶ．また，元のグラフの頂点間に存在する辺の数を抽象頂点間の辺に付与する．この抽象グラフに存在しないグラフパターンは実際のグラフでも存在しないことが保証される．**図 4** に抽象グラフのイメージ図を示す．図 4 の（a）は元のグラフを表しており，類似した頂点を同色で表している．（b）はグラフ内の類似した頂点同士を 1 つに集約したイメージであり，上記で示した情報を抽象頂点と辺に付与することで（c）に示す抽象グラフが得られる．

グラフを抽象化する際に，頂点が類似しているかどうかを判断する必要がある．類似グラフ探索で用いる全てのグラフの頂点について，4 つの物性情報を用いてクラスタリングを行い，同じクラスタに属する頂点同士を類似した頂点と判断する．このとき，全てのグラフの全ての頂点を対象としてクラスタリングを適用するにはデータ量が多いので，大規模なデータに対しても効率的にクラスタリングが可能な CURE[9] を用いる．

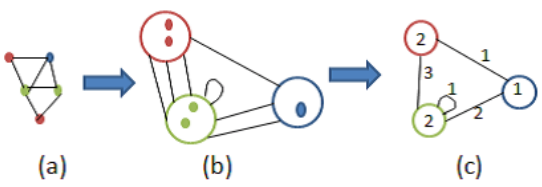

図 4 An example of summary graph

3.2 抽象グラフを用いた類似グラフ探索

最適グラフ発見において，グラフのサイズは重要な要素であり，サイズが小さな部分グラフ I が最適グラフになる可能性は極めて低い．また，サイズの小さな部分グラフ I に類似するグラフは参照蛋白質から大量に発見されるため，計算コストが高くなる．そこで，最終的な部分グラフ I のサイズが一定以上であると想定し，最小グラフサイズ α を導入する．そして，サイズが α 以下の部分グラフ I に対しては最適性に関する評価を行わず，ポジティブグループに属する参照蛋白質 $s_i \in \tau$ のグラフから生成された抽象グラフを用いることで，高速な類似グラフ探索を実現する．対象蛋白質の各ポケットのグラフ $p^1, \cdots, p^n$ に対して抽象グラフの生成を行わない理由は，元のグラフでは存在しない抽象グラフのグラフパターンまで列挙してしまい，無駄な処理を行う可能性が高いからである．また，ネガティブグループに属する参照蛋白質に対して抽象グラフの生成を行わない理由は，枝刈りの性質上，対象蛋白質の部分グラフ I を列挙する段階では，ネガティブグループの参照蛋白質の類似グラフ探索の結果は必要なく，また上述の通り，最小グラフサイズ α 以下の部分グラフ I に対しては，グラフの最適性に関する評価を行わないからである．

抽象グラフを導入した類似グラフ探索においては，列挙された部分グラフ I に対して，類似したグラフがポジティブグループの蛋白質から得られた抽象グラフに存在するか否かを判断する．その際の判断基準は以下の通りである．まず，グラフ G の抽象グラフを SG とし，$SG \prec G$ と表記する．抽象グラフ SG を構成する抽象頂点の集合を $SV(SG)$，抽象グラフ SG を構成する辺の集合を $SE(SG) \in SV(SG) \times SV(SG)$ とする．また，抽象頂点 $sv \in SV(SG)$ に対して，sv の属するクラスタを $c(sv)$，sv を構成する元のグラフ G の頂点数を $NUM(sv)$ とする．そして，抽象グラフの辺 $(su, sv) \in SE$ に対して，su を構成する元のグラフ G の頂点と sv を構成する元のグラフ G の頂点の間に存在する辺数を $NUM(su, sv)$ とする．

このとき，部分グラフ I が，抽象グラフ SG に存在するか否かを判定する方法は以下の通りである．

(1) 部分グラフ I 内に存在するクラスタに属する頂点の数が，そのクラスタと同じ抽象頂点の $NUM(sv)$ 以下である．

(2) 部分グラフ I 内に存在するクラスタのペアを結ぶ辺の数が，そのクラスタのペアと同じ抽象焦点間の辺の $NUM(su, sv)$ 以下である．

この 2 つを満たす場合に，部分グラフ I が抽象グラフ SG

Vol.2016-MPS-108 No.32
Vol.2016-BIO-46 No.32
2016/7/6

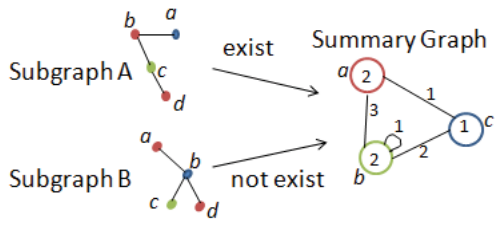

図 5　Search summary graph for similar graph

に存在するといい，$SG \prec G$ となる G に関して，I と類似したグラフが存在することを意味する．図 5 に具体例を示す．グラフの頂点の色がクラスタの種類を表し，赤色，青色，緑色の各クラスタを RED，$BLUE$，$GREEN$ と呼ぶものとする．まず，Subgraph A が Summary graph に存在するか判定する．Subgraph A は $RED, BLUE, GREEN$ の頂点がそれぞれ 2 個，1 個，1 個であるのに対して，Summary graph は，それぞれ 2 個，1 個，2 個である．よって，Subgraph A の全てのクラスタに関して，条件 1 を満たしている．また，Subgraph A の RED と $BLUE$ を結ぶ辺の数が 1 個，RED と $GREEN$ を結ぶ辺の数が 2 個であるのに対して，Summary graph は，それぞれ 1 個，3 個である．よって，Subgraph A の全てのクラスタのペアに関して，条件 2 も満たしている．したがって，Subgraph A は Summary graph に存在する．同様に，Subgraph B が Summary graph に存在するか判定する．条件 1 に関しては，Subgraph A と同様に満たしている．しかし，Subgraph B の RED と $BLUE$ を結ぶ辺の数が 2 個であるのに対して，Summary graph は 1 個しかないので，条件 2 は満たさない．よって，Subgraph B は Summary graph に存在しない．

抽象グラフを導入した類似グラフ探索では，部分グラフ I のサイズが最小グラフサイズ α を超える場合のみ，元のアルゴリズムと同様に，参照蛋白質から類似グラフ探索を行う．部分グラフ I のサイズが α 以下の場合は，類似グラフ探索は行わず，以下の処理を行う．

(1) 部分グラフ I のサイズが 1 のとき，ポジティブグループの各蛋白質 $s_i \in \tau$ の各ポケットの抽象グラフ $SG_i^j (j = 1, \cdots, n)$ から類似した（同じクラスタの）頂点を探索する．
- 類似した抽象頂点が 1 つも存在しないとき，部分グラフ I の拡張を打ち止めにする．
- それ以外のとき，類似した抽象頂点が存在する抽象グラフの集合 $RGS(I) = rgs_1, \cdots, rgs_m$ を記憶する．

(2) 部分グラフ I のサイズが 1 より大きいとき，I の拡張前のグラフ I' に類似したグラフが存在する抽象グラフのリスト $RGS(I') = rgs_1, \cdots, rgs_m$ の各抽象グラフ rgs_i について，I の類似グラフが存在するか判定する．そして，I に類似したグラフが存在する抽象グラフのリスト $RGS(I) = rgs_1, \cdots, rgs_l (l \leq m)$ を記憶する．$RGS(I)$ が空集合なら I の拡張を打ち止めにする．

(3) 部分グラフ I のサイズが最小グラフサイズ α と同じである場合，全ての参照蛋白質の全てのポケットに対して，部分グラフを探索し，類似グラフ集合 $RG(I)$ に発見した類似グラフを登録する．

4. 評価実験及び考察

4.1 評価実験

提案手法を用いて蛋白質の結合部位を予測し，正しい結合部位が特定できているかどうかの評価を行う．評価実験に使用するデータセットは表 2 に示すように，5 種類のリガンドに結合する 37 種類の蛋白質データから構成されている．表の Protein 欄における蛋白質は，PDB で用いられている PDB-ID により記載されている．

実験は，データセットの蛋白質の 1 つを対象蛋白質，それ以外の蛋白質を参照蛋白質として行う．対象蛋白質の結合部位を含むポケットを予測し，予測したポケットが実際に結合部位を含むとき，予測に成功したと考える．予測の対象とした対象蛋白質のうち，成功した割合により，予測精度を評価する．

4.2 抽象化を用いた手法に関する結果

グラフの抽象化を用いた手法による計算コストの削減効果について検証する．抽象グラフを利用せず最適グラフ探索した場合と，抽象グラフを利用した類似グラフ探索手法の場合で，その実行時間と結合部位予測精度を比較した．まず，抽象グラフの頂点数を 1500,2000,2500,3000 と変化させた場合と，抽象グラフを用いない場合の結合部位予測に要した処理時間を図 6 に示す．なお，最小グラフサイズ $\alpha = 6$ である．横軸はリガンド名であり，そのリガンドに結合する蛋白質のグループを表している．縦軸はそのグループに含まれる蛋白質の結合部位の予測に要した時間の

表 2　Ligands and protein groups?

Ligand	Protein
MTX[1]	3dau,3cl9,1e7w,1d1g,1df7
BTN[2]	3g8c,2zsc,3ew2,2c4i,2f01,1bdo,1stp
UMP[3]	2jar,2qch,2bsy,1seh,1f7n
STI[4]	3k5v,3hec,3gvu,2pl0,2oiq 1xbb,1t46,1opj,1iep
DAN[5]	2vk6,2f25,1z4v,1w0o,1rv0 1v3d,1usr,1sli,2qwc,1eus,2sim

[1] methotrexate

[2] biotin

[3] 2'-deoxyuridine 5'-monophosphate

[4] 4-(4-methyl-piperazin-1-ylmethyl)-N-[4-methyl-3-(4-pyridin-3-yl-pyrimidin-2-ylamino)-phenyl]-benzamide

[5] 2-deoxy-2,3-dehydro-N-acetyl-neuraminic acid

Vol.2016-MPS-108 No.32
Vol.2016-BIO-46 No.32
2016/7/6

平均であり，単位は分である．抽象グラフを類似グラフ探索に用いることで，結合部位の予測処理はどの蛋白質においても，頂点数が少ない方が予測にかかる時間は短かった．

次に，抽象グラフの頂点数を変化させた場合と，抽象グラフを用いない場合の結合部位の予測精度を**図 7** に示す．横軸はリガンド名であり，縦軸はそのグループ内で結合部位を予測できた割合を表している．全体的には頂点数 2000 と頂点数 2500 の精度が良かったが，抽象グラフを用いない場合より，精度が悪くなる場合が多かった．

本来，抽象グラフの利用の有無に依存して，結合部位の予測が異なることはあまり良いことではない．抽象グラフを利用することで精度が悪化する傾向にある原因としては，類似グラフ探索の際に，抽象グラフを用いない手法において，2 つのグラフ間で類似すると判断された頂点が，グラフを抽象化することで類似していると判定されず，拡張を打ち止めにしたことが挙げられる．一方で，全ての場合において精度が悪くなる訳ではなく，精度が向上する場合もある．これは，グラフの抽象化により，枝刈りの条件が変化することに依るものと考えられる．すなわち，抽象グラフを利用する手法において，探索の基準となる部分グラフのサイズが最小グラフサイズ α 以下の場合は枝刈りの閾値の更新は行わない．これに対して，グラフの抽象化を行わない手法では，グラフのサイズが小さい場合においても枝刈りの閾値を更新するため，大きなサイズのグラフに成長した段階で評価関数 $G'_{S,\xi}(I)$ が高くなる可能性があるパターンを，グラフサイズが小さい初期段階で枝刈りしてしまう恐れがある．本研究で提案したグラフの抽象化を導入した手法においては，グラフサイズが小さいときの枝刈りの更新が行われず，より広範囲の探索が可能となり，より最適なグラフを見落とすことなく発見できると考えられる．以上が，結合部位の予測結果がグラフの抽象化を用いる場合と用いない場合で変化する主な原因と考えられる．

4.3 最小グラフサイズの考察

一般に，抽象グラフの利用により，類似グラフ探索の効率化が図られるが，探索過程において抽象グラフが利用される範囲により，効率化の度合いが変化すると考えられる．そこで，最小グラフサイズ α の違いによる実行時間と予測精度の変化について考察する．最小グラフサイズ α を 6，8，10，12，14 と変化させた場合の結合部位予測に要した処理時間を**図 8** に示す．なお，抽象グラフの頂点数は 2500 である．図 6 と同様，横軸はリガンド名，縦軸はそのグループに含まれる蛋白質の結合部位の予測に要した時間の平均である．全体的に最小グラフサイズ α が大きいほど，予測時間は短くなる．これは，α の値を大きくするほど，抽象グラフを利用する範囲が拡大され，効率化に寄与すると考えられるためである．

次に，結合部位の予測精度について比較した結果を**図 9**

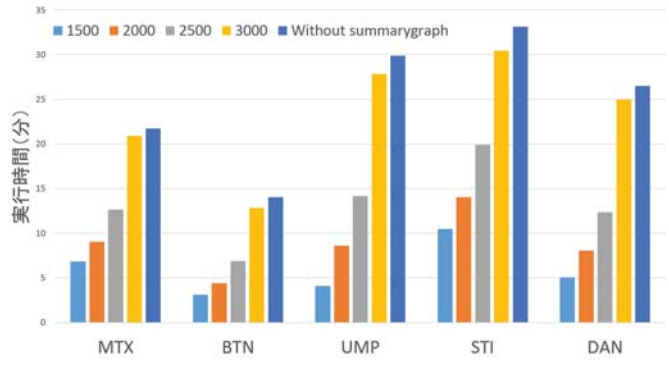

図 6 Total execution time for predicting binding site with different number of clusters

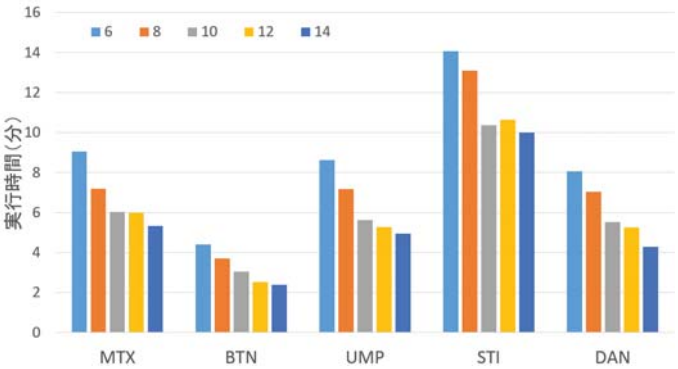

図 8 Total execution time for predicting binding site with different number of α

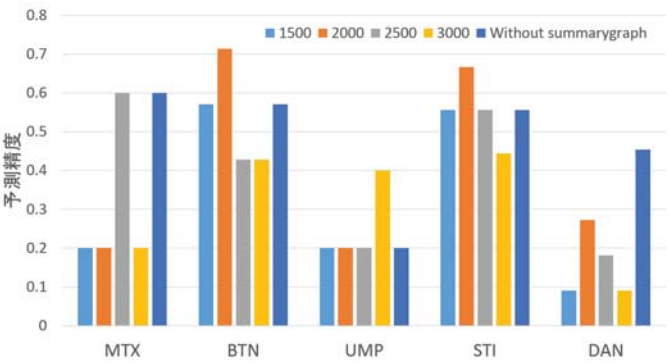

図 7 Success rate of prediction with different number of clusters

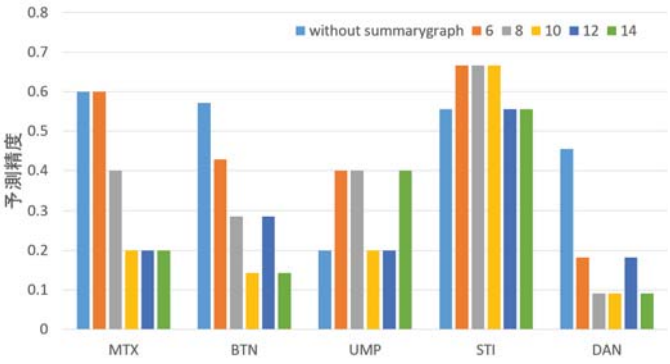

図 9 Success rate of prediction with different number of α

に示す．横軸はリガンド名であり，縦軸はそのグループ内で結合部位を予測できた割合を表している．全体的に最小グラフサイズ α が大きいほど，抽象グラフを利用する範囲が拡大されるため精度は悪くなっているが，少しばらつきがある．これは，抽象グラフを利用する手法において，ポジティブグループに属する蛋白質でのみ抽象グラフを生成し，類似グラフ探索を行うので，類似グラフがあるネガティブグループに属する蛋白質の数に関わらず，枝刈りしてしまうことが原因であると考えられる．

5. 結論

本研究では，蛋白質の分子表面データをもとに，その結合部位を抽象グラフを用いて予測する手法を提案した．提案手法では，蛋白質の分子表面データをグラフで表現し，類似グラフ探索を行い，対象蛋白質のあらゆる部分グラフについて，結合部位となる可能性の高さを評価関数により定量化することで，結合部位の抽出問題を最適グラフ発見問題として定式化した．類似グラフ探索の際，蛋白質のグラフサイズを小さくする抽象グラフを用いることで，計算コストを削減した．

その結果，抽象グラフを利用した探索手法により，蛋白質の結合部位の予測に要する時間を抽象化を用いなかった場合と比較して，約 30 %短縮することに成功した．一方で，予測精度に関しては，抽象グラフを利用する範囲によって，抽象化をしない場合と比べてやや低下するケースも見られた．今後の課題として，抽象グラフの各頂点や辺に付与されている抽象化前のグラフの頂点や辺に関する情報をもとに，どの程度類似グラフが存在しうるかを確率的な観点から判断する方法について検討することが挙げられる．また，データセットを増やして抽象グラフを利用した探索手法による探索空間の変化の影響を考察する必要があると考えられる．

本研究の一部は科学研究費・基礎研究 (B) 24300056 の補助による．

参考文献

[1] N. L. Shrestha and T. Ohkawa, “Filtering Protein Surface Motifs Using Negative Instances of Active Sites Candidates,” *Analysis of Biological Data: A Soft Computing Approach*, pp. 133-152 (2007)

[2] S. Navlakha, R. Rastogi, and N. Shrivastava, “Graph summarization with bounded error,” *Proceedings of the 2008 ACM SIGMOD International Conference on Management of Data,*, pp. 419-432 (2008)

[3] K. LeFevre and E. Terzi, “GraSS: Graph structure summarization,” *SIAM International Conference on Data Mining 2010,*, pp. 454-465 (2010)

[4] K. Kinoshita and H. Nakamura, “Identification of the ligand binding sites on the molecular surface of proteins,” *Protein Scitence*, Vol. 14, pp. 711-718 (2007)

[5] J. Dundas, Z. Ouyang, J. Tseng, A. Binkowski, Y. Turpaz and J. Liang, “CASTp: computed atas of surface topography of proteins with structural and topographical mapping of functionally annotated residues,” *Nucleic Acids Research*, Vol. 34, pp. W116-W118 (2006)

[6] K. Kinoshita and H. Nakamura, “eF-cite and PDBjViewer: Database and viewer for protein functional sites,” *Bioinformatics*, Vol. 20, No. 8, pp. 1329-1330 (2004)

[7] S. Morishita and J. Sese, “Transversing itemset lattices with statistical metric pruning,” *Proceedings of the Nineteenth ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems*, pp. 226-236(2000)

[8] C. Chent, X. Yan, F. Zhu, and J. Han, “gApprox: Mining frequent approximate patterns from a massive network,” *Proceedings of International Confernce on Data Mining*, pp. 4445-450(2007)

[9] S. Guha, R. Rastogi, and K. Shim, “CURE: An efficient clustering algorithm for large database,” *ACM SIGMOD Record*, Vol. 27, No. 2, pp. 73-84 (1998)