

複合構造グラフからの頻出強相関パターン発見

山 本 翼^{†1} 三 好 裕 樹^{†1}
尾 崎 知 伸^{†2} 大 川 剛 直^{†1}

近年、生物学の代謝パスウェイに代表される、頂点に複数の（構造）データが付与されている複合的なグラフ、すなわち複合構造グラフが大量に蓄積されるようになりつつある。本稿では、複合構造グラフデータベース中に存在する強い依存性をとらえるために、強相関パターンと呼ばれる構成要素間に強い依存性の存在するパターンを発見するアルゴリズム HFMG を提案する。HFMG は、複合構造グラフの頂点およびグラフ構造全体に着目し、その両者において高い依存性を持つパターンのみを抽出する。これにより、依存性が低く偶発的で冗長なパターンの生成が回避されるとともに、頻度は低い強い依存性を持つパターンの発見が期待される。代謝パスウェイを用いた実験を通じ、生成されるパターン数の抑制およびより低頻度におけるパターン発見が達成され、その有効性が確認された。

Discovery of Frequent Hyperclique Patterns in Multi-structured Graph Databases

TSUBASA YAMAMOTO,^{†1} YUUKI MIYOSHI,^{†1}
TOMONOBU OZAKI^{†2} and TAKENAO OHKAWA^{†1}

“Multi-structured graphs” are complex graphs whose vertices consist of a set of structured data such as itemsets, sequences and so on. In some application areas, *e.g.* bioinformatics, the whole structure of the target data can be modeled or represented naturally in multi-structured graphs. In this paper, to catch the strong affinity relationships in multi-structured graphs, we propose a pattern mining algorithm named HFMG for discovering hyperclique patterns, *i.e.* novel and meaningful frequent patterns whose components are highly correlated with each other. HFMG considers two kinds of correlations among components and it can enumerate all hyperclique patterns efficiently by employing the powerful pruning based on the properties of correlation. The effectiveness of HFMG is confirmed through the experiments with real datasets.

1. はじめに

グラフは、ソーシャルネットワークやタンパク質間ネットワークなど、物事の関係性を表す表現方法として広く利用されている。ここで、グラフで表現できる代表的なデータの1つとして、生物の代謝パスウェイについて考える。代謝パスウェイとは、酵素によって化合物を別の化合物へ変換する一連の化学反応のネットワークであり、反応に関与する酵素を頂点、酵素間の反応を辺とすることにより、グラフとして表現することが可能である。図1に代謝パスウェイの例を示す。図中において、長方形で表現された酵素が頂点に、また化合物を介する酵素間の反応関係が辺に、それぞれ対応する。一方、図2に示すように、頂点である各酵素には、酵素名 (Gene Name) や識別番号 (Orthology) に加え、酵素を構成するアミノ酸配列 (AA seq) や、特徴的配列であるモチーフの集合 (Motif) などの様々な情報を関連付けることが可能である。したがって、代謝パスウェイをより精密かつ自然に表現するためには、単なるグラフでは不十分であり、各頂点に複数のデータが付与された複合的なグラフ、すなわち複合構造グラフ¹⁴⁾が必要とされる。

代謝パスウェイ以外にも、複合構造グラフを利用することで、より精密かつ自然に表現することのできるグラフデータが存在する。たとえば、タンパク質に対しては、それを構成するアミノ酸配列のほか立体構造や表面形状など、種々の属性を考えることができる。したがって、タンパク質を頂点、それらの相互作用を辺とするタンパク質間相互作用ネットワークは、複合構造グラフとして自然に定式化することが可能である。またウェブサイトは、ウェブページを頂点、ハイパーリンクを辺とするネットワークと見なすことができるが、より詳細には、各ページに、内容を表すキーワードの集合や、ページの構成を表す HTML のタグ構造を割り当てた複合構造グラフとしてとらえることが可能である。

一方、複合構造グラフからの知識発見を考えた場合、専門家が持つ領域知識などを用いて、頂点に付与された複数のデータを単一のアイテムへと抽象化し、一般的なグラフマイニング手法¹⁵⁾を適用することも考えられる。しかし、必ずしも領域知識が適用できるとは限らず、また、抽象化によりデータが本来持っている情報が欠落してしまうので、有益な知識の発見が阻害される可能性も否定できない。これらのことから、今後ますます増大が予想さ

^{†1} 神戸大学大学院工学研究科

Graduate School of Engineering, Kobe University

^{†2} 神戸大学自然科学系先端融合研究環

Organization of Advanced Science and Technology, Kobe University

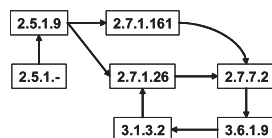


図 1 代謝パスウェイの例

Fig. 1 An example of metabolic pathway.

Gene Name	FLAD1
Orthology	KO: K00953 [EC:2.7.7.2]
Motif	MoCF_biosynth ATP_bind_3 PAPS_reduct
AA Seq	MGWDLGTRLFQRQEQRS RLSRIWLEKTRVF...

図 2 酵素データの例

Fig. 2 An example of enzyme data.

れる複合構造グラフを対象とした知識発見手法、なかでも特に、頂点に付与されたデータを抽象化することなく直接扱うことのできる手法の開発は重要な課題の 1 つであるといえる。

これまでに、複合構造グラフデータベースからの知識発見を目的に、データベース中に頻出するパターンを発見する手法が提案されている¹⁴⁾。しかし、単純な頻出パターン発見では、特に頂点構造の複合性に起因し、「得られるパターン数が爆発的に増大する」という頻出パターン発見特有の問題が顕著に現れる。したがって、複合構造グラフデータベースからの知識発見をより現実的なものとするためには、何らかの基準を用いて求めるべきパターンを限定し、より意味のあるパターンのみを発見する手法が必要不可欠である。これまで、グラフなどの構造データを対象としたパターンの限定手法として、(1) パターンに対して制約を与える¹⁸⁾、(2) (飽和パターンなどの) 代表的なパターンのみを列挙する¹⁶⁾ などの手法が提案されている。また、複合構造グラフデータを対象に、一部、同様の手法も提案されている¹⁴⁾。これに対し本稿では、関連パターン発見³⁾の考えに基づき、パターンが満たすべき新たな基準として「パターンとその構成要素間の依存性」を利用する手法^{12),13)}を複合構造グラフに導入することを提案する。形式的な定義は 2 章で与えるが、複合構造グラフパターンは、(1) 頂点中の (構造) データ、(2) 頂点 (データの集合) および (3) グラフ構造 (頂点間の連結関係) の 3 種の構成要素からなる 3 層構造のパターンであり、各頂点は (構造) データを構成要素とし、またグラフ全体は頂点を構成要素としている。このことに基づき、頂点

と頂点内のデータ間の依存性を内部相互依存性 (internal mutual dependency)、グラフ構造と頂点間の依存性を外部相互依存性 (external mutual dependency) とそれぞれ呼び、両者において強い依存性を持つパターン、すなわち強相関パターン (hyperclique patterns) のみを効率的に抽出することを目的としたアルゴリズム HFMG を提案する。

ここで、代謝パスウェイを例に、複合構造グラフパターンにおける依存性とそれを考慮することの意義について説明する。たとえば、モチーフ集合中に 'ATP_bind_3' と 'PAPS_reduct' を含み、かつアミノ酸配列中に配列 (部分文字列) 〈WDLG〉を含む酵素は、頂点のパターンとして $\{[ATP_bind_3, PAPS_reduct], \langle WDLG \rangle\}$ のように表現される。また同様に、モチーフ集合中に 'MoCF_biosynth' を、アミノ酸配列中に 〈T〉を含む酵素は、頂点パターン $\{[MoCF_biosynth], \langle T \rangle\}$ と表現される。さらに、これらの酵素の接続関係、すなわち代謝パスウェイの一部が、頂点パターンのグラフとして表現される。このとき、たとえば頂点パターン $\{[MoCF_biosynth], \langle T \rangle\}$ において、それを構成する各要素 $\{MoCF_biosynth\}$ や $\langle T \rangle$ と、頂点パターンとの依存関係が内部相互依存性である。具体的には、モチーフとして MoCF_biosynth を持つ酵素のうち、どの程度がアミノ酸配列に 〈T〉を含むのか、また逆に、アミノ酸配列に 〈T〉を含む酵素のうちどの程度がモチーフ MoCF_biosynth を持つかなど、頂点パターン中のある条件を満たす酵素が、他の (すべての) 条件をどの程度満たすのかに着目する。これにより、冗長なパターンを排除することが可能となる。たとえば、頂点パターン $\{[MoCF_biosynth], \langle T \rangle\}$ を考える。アミノ酸配列は、20 種のアミノ酸から構成されているので、ほとんどすべての酵素は、その配列中に 〈T〉という部分文字列を含むと考えられる。したがって、酵素がモチーフ MoCF_biosynth を持つか否かにかかわらず、〈T〉はアミノ酸配列に含まれることになり、両者 (酵素がそのアミノ酸配列中に 〈T〉を含むことと、モチーフとして MoCF_biosynth を持つこと) の間に依存関係があるとは考えられない。いい換えれば、頂点パターン $\{[MoCF_biosynth], \langle T \rangle\}$ を考える際、 $\{MoCF_biosynth\}$ に対して 〈T〉は冗長、すなわち 〈T〉の有無が解釈などへ影響を与えない、ということである。一方、別の頂点パターンとして $\{[ATP_bind_3, PAPS_reduct], \langle WDLG \rangle\}$ を考える。このとき仮に、2 つのモチーフ $\{ATP_bind_3, PAPS_reduct\}$ を持つ酵素や、アミノ酸配列に 〈WDLG〉を含む酵素がともに少ない場合でも、両者の依存関係が高いのであれば、 $\{ATP_bind_3, PAPS_reduct\}$ や 〈WDLG〉単体ではなく、その集合 $\{[ATP_bind_3, PAPS_reduct], \langle WDLG \rangle\}$ として意味があり、有益であると考えられる。通常の頻出パターン発見では、低頻度が原因で他の大量のパターンに埋もれてしまうようなパターンであっても、依存性を考慮することで、有益なパターンの 1 つとして抽出することが期待できる。

同様の議論は、グラフパターン全体に対しても適用できる．外部相互依存性とは、酵素が満たす条件の集合である頂点パターンと、その連結関係である（部分）代謝パスウェイに相当するグラフパターンとの依存関係である．具体的には、ある条件を満たす酵素を含む代謝パスウェイが、連結関係も考慮したうえで、どの程度他の頂点パターンも含むのかに着目する．これにより、仮に頻度が低いとしても、依存性の高い頂点パターンが集まって複合構造グラフパターンが形成されている場合、それは少数の生物種に特有の重要な代謝を表す非常に有益なパターンであると考えることができる．

このように依存性を考慮することで、複合構造グラフデータベースの少数のトランザクションにしか出現しないが、強い依存性を持つ意味のあるパターンの発見が可能になることが期待できる．加えて、依存性に基づく新たな枝刈り手法を導入することで、パターン発見全体の効率向上も期待できる．

以下に本稿の構成を示す．2 章では、準備として、複合構造グラフデータベースに関する用語を与える．3 章で、内部および外部相互依存性を定義し、ついで 4 章で、頻出強相関パターン発見アルゴリズム HFMG を提案する．5 章で関連研究について述べ、6 章で実験結果を示す．最後に 7 章でまとめを行う．

2. 複合構造グラフ

本章では文献 14) などに従い、複合構造グラフに関する定義を導入する．なお、本稿は無向グラフを対象とするが、提案する枠組み自体は、有向グラフにも容易に拡張可能である．

複合構造グラフ $G = (V_G, E_G)$ は、頂点集合 V_G と辺集合 $E_G \subseteq V_G \times V_G$ から構成される．各頂点 $v \in V_G$ は、 n 個のデータからなるデータ集合で構成される．頂点 v を構成するデータ集合を $s(v) = [elm_1^v, \dots, elm_n^v] \in [\text{dom}(A_1), \dots, \text{dom}(A_n)]$ と表記する．ここで $\text{dom}(A_i)$ は、アイテム集合や系列など、 i 番目の属性 A_i のクラスを表す．頂点 v と v' を結ぶ辺を (v, v') と表記する．各辺 $(v, v') \in E_G$ には、 $l(v, v')$ で参照されるラベルが 1 つ与えられる．複合構造グラフにおいて、各頂点を構成するデータ集合を内部構造と呼び、辺による頂点の接続関係を外部構造と呼ぶ．図 3 に複合構造グラフの例を示す．図中のグラフは、模式化した代謝パスウェイを意図しており、酵素に相当するグラフ中の各頂点は、(1) モチーフの集合を表すアイテム集合と、(2) アミノ酸配列を表す系列から構成される．たとえば V_{G_1} 中の頂点 v_{13} に対する $s(v_{13}) = [\{a, c\}, \langle AEACBE \rangle]$ は、酵素 v_{13} がアミノ酸配列 $AEACBE$ からなり、2 つのモチーフ a と c を持つことを表す．

アイテム集合における空集合や、系列における空列に対応し、頂点が属性 A_i を持たない、

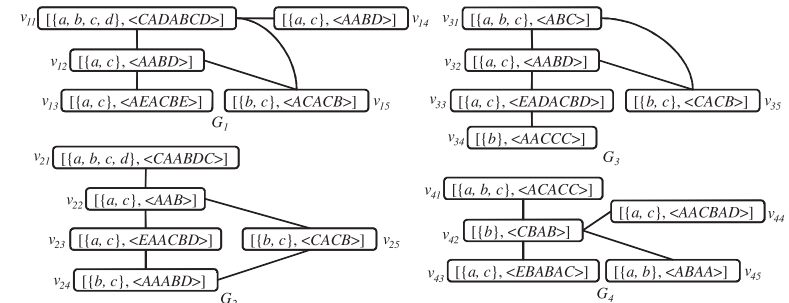


図 3 複合構造グラフデータベース D
Fig. 3 An example of multi-structured graph database D .

または考慮しないことを表す記号として、 ϕ を導入する．3 つの頂点 v, x, y をそれぞれ構成するデータ集合 $s(v) = [v_1, v_2, \dots, v_n]$, $s(x) = [x_1, x_2, \dots, x_n]$, $s(y) = [y_1, y_2, \dots, y_n]$ に対し、すべての v_i, x_i, y_i が、(1) $v_i = x_i = y_i = \phi$, (2) $v_i \neq \phi, x_i = v_i, y_i = \phi$, または (3) $v_i \neq \phi, x_i = \phi, y_i = v_i$ のいずれかの条件を満たすとき、 $s(v) = s(x) \cup s(y)$ と表記する．すなわち直感的には、 $s(v) = s(x) \cup s(y)$ とは、 v を構成するデータ集合を 2 つに分ける操作に対応する．一方、 $s(v)$ と $s(x)$ 中のすべての v_i, x_i が (1) $x_i = \phi$, または (2) $x_i = v_i$ を満たすとき、 $s(x) \subseteq s(v)$ と表記する．

本稿では、属性 A_i 中のデータ $x, y \in \text{dom}(A_i)$ に対し、後述するデータパターンの支持度（式 (3)）が特殊化により単調減少する、すなわち $x \preceq_i y \rightarrow \text{sup}_D^e(x) \geq \text{sup}_D^e(y)$ を満たす半順序関係 $\preceq_i$ を考える．また、 $x \preceq_i y$ が成り立つとき、 x は y に出現するという． $s(v) = [elm_1^v, \dots, elm_n^v]$ および $s(u) = [p_1^u, \dots, p_n^u]$ なる 2 つの頂点 v と u に対し、すべての p_i^u が対応する要素 elm_i^v に出現するとき、すなわち $\forall i [p_i^u \preceq_i elm_i^v]$ が成り立つとき、 u は v に出現すると定義し、 $u \preceq v$ と表記する．出現について、図 4 を例に説明する．アイテム集合に対して、 $\preceq$ として部分集合関係 ($\subseteq$) を用いる．このとき、アイテム集合 i_1 と i_2 に対して $i_2 \subseteq i_1$ なので、 i_2 は i_1 に出現する．一方、 $i_3 \not\subseteq i_1$ なので $i_3 \preceq i_1$ は成り立たない．系列データに対しては、半順序関係 $\preceq$ として部分系列関係 $\sqsubseteq$ を用いる．系列データ $t = \langle t_1 t_2 \dots t_n \rangle$ と $s = \langle s_1 s_2 \dots s_m \rangle$ に対し、 s が t の部分系列、すなわち $\forall i \in [1, \dots, m] \exists j_i \in [1, \dots, n] (s_i = t_{j_i} \wedge 1 \leq j_1 < \dots < j_m \leq n)$ が成り立つとき、この関係を $s \sqsubseteq t$ と定義する．このとき、図 4 中の系列 s_1 と s_2 に対して、 $s_2 \preceq s_1$ は成り立つが $s_3 \preceq s_1$ は成り立たない．さらに、アイテム集合と系列からなる 2 つのデータ集合 $s(sp_1) = [i_1, s_1]$,

アイテム集合	$i_1 = \{a, b, c, e, g\}, i_2 = \{a, b, e\}, i_3 = \{a, d\}$
アイテム系列	$s_1 = \langle ABABC \rangle, s_2 = \langle BABC \rangle, s_3 = \langle ABCA \rangle$
データ集合	$s(sp_1) = [i_1, s_1] = [\{a, b, c, e, g\}, \langle ABABC \rangle]$ $s(sp_2) = [i_2, s_2] = [\{a, b, e\}, \langle BABC \rangle]$

図 4 2 クラスのデータとデータ集合

Fig. 4 Examples of data and sets of data.

$s(sp_2) = [i_2, s_2]$ に対し, $i_2 \subseteq i_1$ および $s_2 \sqsubseteq s_1$ がともに成り立つので $sp_2 \preceq sp_1$ である.

2 つの複合構造グラフ $G = (V_G, E_G)$, $G' = (V_{G'}, E_{G'})$ に対し,

$$(1) \forall v \in V_G [v \preceq f(v)], \quad (2) \forall (u, v) \in E_G [l(u, v) = l(f(u), f(v))]$$

を満たす単射 $f: V_G \rightarrow V_{G'}$ が存在するとき, G は G' に出現するといいい, $G \preceq G'$ と表記する. また $G \preceq G'$ のとき, G を G' の部分複合構造グラフと呼ぶ.

次に, 複合構造グラフデータベースにおけるパターンとその支持度を導入する. データベース中のグラフに対する部分複合構造グラフを複合構造グラフパターン (以後, グラフパターン) と呼ぶ. また, グラフパターン中の各頂点を構成するデータ集合を頂点パターン, 頂点パターンの構成要素をデータパターンと呼ぶ. 複合構造グラフデータベース $D = \{G_1, \dots, G_M\}$ に対し, グラフパターン $p = (V_p, E_p)$ および p 中の頂点パターン $u \in V_p$, u 中の i 番目のデータパターン $elm_i \in s(u)$ の支持度を, それぞれ以下のように定義する.

$$sup_D^G(p) = |\{G \in D \mid p \preceq G\}| / M \quad (1)$$

$$sup_D^s(u) = \sum_{(V_G, E_G) \in D} |\{v \in V_G \mid u \preceq v\}| / N \quad (2)$$

$$sup_D^e(elm_i) = \sum_{(V_G, E_G) \in D} |\{v \in V_G \mid elm_i \preceq elm_i^v\}| / N \quad (3)$$

ここで $N = \sum_{(V_G, E_G) \in D} |V_G|$ はデータベース中の総頂点数, elm_i^v は頂点 v の i 番目のデータである. また, グラフパターンの支持度はデータベース中の総トランザクション数 M に基づいているのに対し, 頂点パターンおよびデータパターンの支持度は総頂点数 N を基準としている. 複合構造グラフデータベース D と利用者により与えられる最小支持度 σ ($0 < \sigma \leq 1$) に対し, $sup_D^G(p) \geq \sigma$ を満たすパターンを頻出複合構造グラフパターン (以後, 頻出グラフパターン) と呼ぶ.

3. 複合構造グラフパターンにおける相互依存性

本章では, 複合構造グラフパターンにおける 2 つの相互依存性である, 内部相互依存性および外部相互依存性を導入する.

3.1 内部相互依存性

内部相互依存性とは, グラフパターン $p = (V_p, E_p)$ 中の各頂点パターン $u \in V_p$ に対して定義されるものであり, u とその構成要素であるデータパターン $elm_i^u \in s(u)$ 間の依存性を表す. 複合構造グラフデータベース D に対する, 頂点パターン u の内部相互依存性を以下のように定義する. なおこの定義は, アイテム集合における相互依存性^{12),13)} の自然な拡張である.

$$hconf_D^{in}(u) = \min_{elm \in s(u), elm \neq \phi} \{sup_D^s(u) / sup_D^e(elm)\} \\ = sup_D^s(u) / \max_{elm \in s(u), elm \neq \phi} \{sup_D^e(elm)\} \quad (4)$$

内部相互依存性は, ある頂点 v においてデータパターン elm が出現した際の, 同一の頂点 v 中に頂点パターン u が出現する条件付き確率の下限値である. したがって, $0 \leq hconf_D^{in}(u) \leq 1$ であり, また値が大きいほど依存性が強い.

たとえば図 5 中のパターン P_a の頂点 u_1 に対する内部相互依存性は, データパターン $\{a, b, c\}$ が出現した際の頂点パターン $[\{a, b, c\}, \langle ABC \rangle]$ が出現する確率と, データパターン $\langle ABC \rangle$ が出現した際の頂点パターン $[\{a, b, c\}, \langle ABC \rangle]$ が出現する確率の下限値である. 同様に, u_2 および u_3 の内部相互依存性が計算できる. 利用者によって与えられた最小内部相互依存度 h_{in} ($0 \leq h_{in} \leq 1$) に対し, 頂点パターン u が $hconf_D^{in}(u) \geq h_{in}$ を満たすとき, u は内部相互依存性を持つという. またグラフパターン $p = (V_p, E_p)$ に対し, p 中の全頂点パターンが内部相互依存性を持つ, すなわち $\forall u \in V_p [hconf_D^{in}(u) \geq h_{in}]$ を満たすとき, p を内部強相関パターンと定義する. 非内部強相関パターンでは, 頂点パターンの支持度と各データパターンの支持度が大きく異なるので, 理解が困難であると考えられる. また, 内部相互依存性を考慮することで, 支持度が高く, どの頂点にも現れるような, ある意味で冗長なデータパターンの排除が期待できる.

内部相互依存性に関し, 以下の性質を導入する.

性質 1 (内部相互依存性の上界値) 複合構造グラフデータベース D と, $s(s) = s_\alpha \cup s_\beta$ および $s(s') = s_\alpha \cup s'_\beta$ なる 2 つの頂点パターン s, s' に対し, $s_\alpha \neq \emptyset$ かつ $s_\beta \preceq s'_\beta$ なら

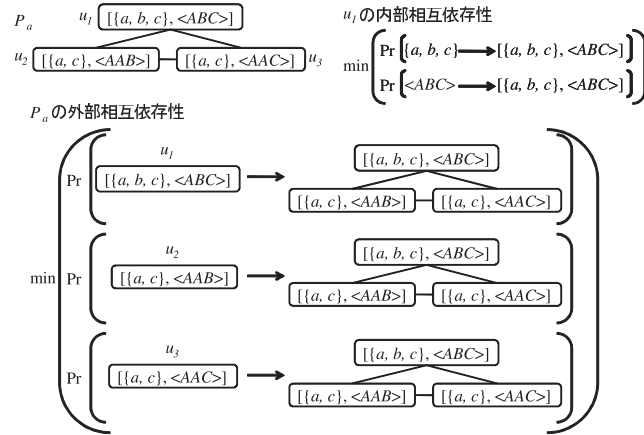


図 5 内部相互依存性および外部相互依存性の例
Fig. 5 Internal mutual dependency and external mutual dependency.

ば，以下の不等式が成り立つ．

$$up(s, s_\alpha) = sup_D^s(s) / \max_{elm \in s_\alpha, elm \neq \phi} \{sup_D^e(elm)\} \geq hconf_D^{in}(s')$$

証明 1 $s_\alpha \subseteq s(s')$ より， $\max_{elm \in s_\alpha, elm \neq \phi} \{sup_D^e(elm)\} \leq \max_{elm' \in s(s'), elm' \neq \phi} \{sup_D^e(elm')\}$ である．また $s \preceq s'$ より， $sup_D^s(s) \geq sup_D^s(s')$ が成り立つ．よって， $up(s, s_\alpha) = sup_D^s(s) / \max_{elm \in s_\alpha, elm \neq \phi} \{sup_D^e(elm)\} \geq sup_D^s(s') / \max_{elm' \in s(s'), elm' \neq \phi} \{sup_D^e(elm')\} = hconf_D^{in}(s')$ が成り立つ． $\square$

以下に，図 3 中のデータベース D および図 6 中の頂点パターンを対象に，内部相互依存性およびその上界値の計算例を示す．グラフパターン P_1 中の $s(u_1) = [\{a, b, c\}, \langle ABC \rangle]$ なる頂点パターン u_1 および $s(u_2) = [\{a, c\}, \langle AAB \rangle]$ なる頂点パターン u_2 に対し，

$$hconf_D^{in}(u_1) = sup_D^s(u_1) / \max\{sup_D^e(\{a, b, c\}), sup_D^e(\langle ABC \rangle)\} \\ = (3/20) / \max\{(3/20), (4/20)\} = 0.75 \quad (5)$$

$$hconf_D^{in}(u_2) = sup_D^s(u_2) / \max\{sup_D^e(\{a, c\}), sup_D^e(\langle AAB \rangle)\} \\ = (10/20) / \max\{(14/20), (12/20)\} = 0.71 \quad (6)$$

となる．また $s_\alpha = [\{a, c\}, \phi]$ としたとき， P_7 中の $s(u_3) = [\{a, c\}, \langle E \rangle]$ なる頂点パターン u_3 に対する上界値は，以下のように計算される．

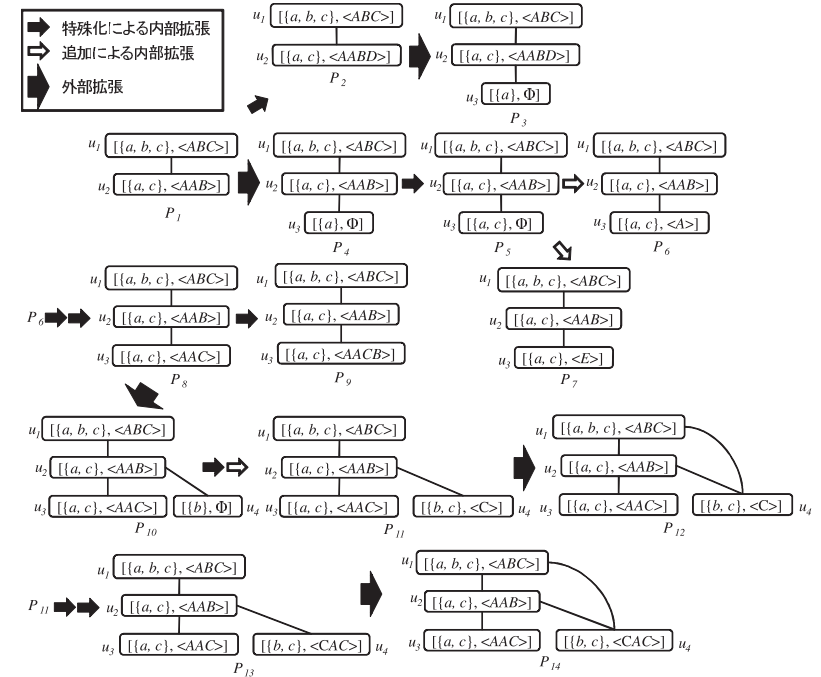


図 6 パターン列挙の過程の例
Fig. 6 An example of search space.

$$up(u_3, [\{a, c\}, \phi]) = sup_D^s(u_3) / \max\{sup_D^e(\{a, c\})\} = (4/20) / (14/20) = 0.29 \quad (7)$$

内部相互依存性の上界値は，パターン発見における枝刈り基準として利用可能である．すなわち，頂点パターン s が $up(s, s_\alpha) < h_{in}$ を満たせば， $s(s') = s_\alpha \cup s'_\beta$ ， $s_\beta \preceq s'_\beta$ なる頂点パターン s' に対して， $hconf_D^{in}(s') < h_{in}$ が成り立つ．したがって， s より特殊な頂点 s' を持つグラフパターンは，内部強相関パターンとはなりえない．

3.2 外部相互依存性

外部相互依存性とは，グラフパターンに対して定義されるものであり，グラフパターンとその構成要素である頂点パターン間の依存性である．複合構造グラフデータベース D における，グラフパターン $p = (V_p, E_p)$ に対する外部相互依存性を以下のように定義する．なお下式では，頂点パターン u を 1 つの頂点からなるグラフパターンとして扱っている．

$$hconf_D^{ex}(p) = \min_{u \in V_p} \{sup_D^G(p) / sup_D^G(u)\} = sup_D^G(p) / \max_{u \in V_p} \{sup_D^G(u)\} \quad (8)$$

外部相互依存性は、頂点パターン u に対し、 u と接続する形でグラフパターン p が存在する条件付き確率の下限值である。 $0 \leq hconf_D^{ex}(p) \leq 1$ であり、値が大きいほど依存性が強い。たとえば図 5 のパターン P_a の外部相互依存性は、頂点パターン u_1 が出現した際のグラフパターン P_a が出現する確率、頂点パターン u_2 が出現した際のグラフパターン P_a が出現する確率、および頂点パターン u_3 が出現した際のグラフパターン P_a が出現する確率の下限值である。

利用者によって与えられた最小外部相互依存度 h_{ex} ($0 \leq h_{ex} \leq 1$) に対し、グラフパターン p が $hconf_D^{ex}(p) \geq h_{ex}$ を満たすとき、 p を外部強相関パターンと呼ぶ。外部強相関パターンでは、頂点パターンとグラフパターンの支持度の値が近い。それゆえ、データベース上で、ある頂点パターンが現れる周囲には、必ず同じ程度の支持度を持つ頂点パターンが存在し、それらがグラフパターンを形成していると考えることができ、理解がしやすい。逆に、非外部強相関パターンは、頂点パターンとグラフパターンの支持度が大きく異なり、ある頂点パターンの周囲に偶然支持度の高い頂点パターンが存在したという場合が考えられる。外部相互依存性を考慮することで、このような無条件に支持度の高い頂点パターンを持つグラフパターンの生成を抑制することが可能となる。

外部相互依存性に関し、以下の性質を導入する。

性質 2 (外部相互依存性の上界値) 複合構造グラフデータベース D と、 $p \preceq p'$ なる 2 つのグラフパターン $p = (V_p, E_p)$ と $p' = (V_{p'}, E_{p'})$ に対し、 $V_p = V_\alpha \cup V_\beta$ 、 $V_{p'} = V_\alpha \cup V_{\beta'}$ 、 $V_\alpha \neq \emptyset$ 、 $V_\alpha \cap V_\beta = \emptyset$ とする。このとき、以下の不等式が成り立つ。

$$up(p, V_\alpha) = sup_D^G(p) / \max_{u \in V_\alpha} \{sup_D^G(u)\} \geq hconf_D^{ex}(p')$$

証明 2 $V_\alpha \subseteq V_{p'}$ より $\max_{u \in V_\alpha} \{sup_D^G(u)\} \leq \max_{u' \in V_{p'}} \{sup_D^G(u')\}$ が成り立つ。一方 $p \preceq p'$ より $sup_D^G(p) \geq sup_D^G(p')$ である。これにより、 $up(p, V_\alpha) = sup_D^G(p) / \max_{u \in V_\alpha} \{sup_D^G(u)\} \geq sup_D^G(p') / \max_{u' \in V_{p'}} \{sup_D^G(u')\} = hconf_D^{ex}(p')$ が成り立つ。□

以下に、図 3 中のデータベース D および図 6 中のグラフパターン $P_{13}, P_{14}, P_8 = (V_{P_8}, E_{P_8})$ を対象とした、外部相互依存性およびその上界値の計算例を示す。

$$hconf_D^{ex}(P_{13}) = sup_D^G(P_{13}) / \max \left\{ \begin{array}{l} sup_D^G(\{a, b, c\}, \langle ABC \rangle), sup_D^G(\{a, c\}, \langle AAB \rangle), \\ sup_D^G(\{a, c\}, \langle AAC \rangle), sup_D^G(\{b, c\}, \langle CAC \rangle) \end{array} \right\} \\ = (3/4) / \max\{(3/4), (4/4), (4/4), (4/4)\} = 0.75 \quad (9)$$

$$up(P_{14}, V_{P_8}) = sup_D^G(P_{14}) / \max \left\{ \begin{array}{l} sup_D^G(\{a, b, c\}, \langle ABC \rangle), sup_D^G(\{a, c\}, \langle AAB \rangle), \\ sup_D^G(\{a, c\}, \langle AAC \rangle) \end{array} \right\} \\ = (2/4) / \max\{(3/4), (4/4), (4/4)\} = 0.5 \quad (10)$$

内部相互依存性同様、外部相互依存性の上界値もパターン発見における枝刈り基準として利用可能である。すなわち、グラフパターン $p = (V_p, E_p)$ が $up(p, s_\alpha) < h_{ex}$ を満たせば、 $p' = (V_{p'}, E_{p'})$ s.t. $V_p \subseteq V_{p'}$ なるグラフパターン $p' = (V_{p'}, E_{p'})$ に対し、 $hconf_D^{ex}(p') < h_{ex}$ が成り立つ。したがって、そのようなグラフパターン p' は、外部強相関パターンとはならない。

4. 複合構造グラフからの強相関パターン発見

複合構造グラフデータベース D 、最小支持度 σ 、最小内部相互依存度 h_{in} および最小外部相互依存度 h_{ex} が与えられたとき、 $sup_D^G(p) \geq \sigma$ 、 $\forall u \in V_p [hconf_D^{in}(u) \geq h_{in}]$ 、 $hconf_D^{ex}(p) \geq h_{ex}$ を満たすグラフパターン $p = (V_p, E_p)$ を頻出強相関パターンと定義する。

本章ではまず、頻出強相関パターン発見アルゴリズムの基礎として、頻出複合構造グラフパターン発見アルゴリズム FMG¹⁴⁾ を概観する。その後、FMG を基礎とし、複合グラフデータベース中のすべての頻出強相関パターンを発見するアルゴリズム HFMG を提案する。

4.1 頻出パターン発見アルゴリズム FMG

複合構造グラフデータベースからの頻出グラフパターン発見アルゴリズム FMG¹⁴⁾ は、ラベル付きグラフマイニング手法 gSpan¹⁵⁾ の素直な拡張であり、標準形判定と逆探索に基づくアルゴリズムである。

アルゴリズム FMG の擬似コードを図 7 に示す。図 7 において、isCanonical (FMG-Enum の 11 行目) が標準形判定に相当する。詳細は文献 14) や 15) に委ねるが、標準形とは、同型パターン集合における代表元である。各グラフパターンをコードワード¹⁵⁾ と呼ばれる文字列で表現した際、同型パターン集合において、(文字列の意味で) 最小のコードワードを持つグラフパターンが標準形と定義される。標準形を考慮することで、同型グラフの重複生成が回避される。一方コードワードの生成には、グラフパターン p を深さ優先探索することにより得られる全域木を利用する。このとき、探索の開始点を根、最終点を最右頂

Algorithm FMG(D, σ)

```

1: set  $I$  be a set of all initial patterns
2: for each  $p \in I$ 
3:   FMG-Enum( $p, D, \sigma$ )
4: end for

```

Subroutine FMG-Enum(p, D, σ)

```

1: if  $\text{sup}_D^G(p) < \sigma$  then return
2: if degree of  $\text{rmv}(p) > 1$  then goto line 11
3:  $P_{ins} := \text{internalExpansionBySpecialization}(p)$ 
4: for each  $t \in P_{ins}$ 
5:   FMG-Enum( $t, D, \sigma$ )
6: end for
7:  $P_{ina} := \text{internalExpansionByAddition}(p)$ 
8: for each  $t \in P_{ina}$ 
9:   FMG-Enum( $t, D, \sigma$ )
10: end for
11: if  $\neg \text{isCanonical}(p)$  then return
12: output  $p$ 
13:  $P_{ex} := \text{externalExpansion}(p)$ 
14: for each  $t \in P_{ex}$ 
15:   FMG-Enum( $t, D, \sigma$ )
16: end for

```

図 7 FMG の擬似コード
Fig. 7 Pseudo code of FMG.

点 ($\text{rmv}(p)$ と表記), 全域木上で根から最右頂点へ至る経路を最右パスと呼ぶ。

次に, FMG におけるパターン列挙について説明する. FMG ではまず, 大きさ 1 のデータパターンをちょうど 1 つ持つグラフパターン p の集合 I を考え (FMG の 1 行目), その各々に対して FMG-Enum を適用することで, より特殊なパターンを生成する (FMG の 2-4 行目). なお, データの大きさは, アイテム集合ならばアイテムの数, 系列ならば系列長など, 各構造のクラスによってそれぞれ定義されるとする. たとえば, 図 3 中のデータベース D を対象とした場合, パターン I としては, $\{[a], \phi\}, \{[b], \phi\}, \{[c], \phi\}, \{[d], \phi\}, [\phi, \langle A \rangle], [\phi, \langle B \rangle], [\phi, \langle C \rangle], [\phi, \langle D \rangle], [\phi, \langle E \rangle]$ が考えられる.

FMG-Enum では重複を回避したうえでの頻出グラフパターンの網羅的な列挙のため, 各グラフパターン p に対し, まず支持度の検査 (1 行目) と内部拡張の必要性の検査 (2 行目) を行う. その後, (1) 頂点パターン中のデータパターンを特殊化する特殊化による内部拡張 (internalExpansionBySpecialization, 3 行目), (2) 頂点パターンに新たなデータパ

ターンを追加する追加による内部拡張 (internalExpansionByAddition, 7 行目), (3) グラフパターンに新たな辺を追加する外部拡張 (externalExpansion, 13 行目), の 3 種類の拡張が適用される. さらに, p より得られた各グラフパターン t に対し, FMG-Enum を再帰的に呼び出す (5, 9, 15 行目) ことで網羅的な探索が実現される.

支持度の検査 (1 行目) では, 支持度の逆単調性に基づき, p の支持度が最小支持度 σ 未満 ($\text{sup}_D^G(p) < \sigma$) であれば, それ以降のパターンの拡張を回避する. 内部拡張必要性の検査 (2 行目) は, 最右頂点の度数に基づき行われる. 次数が 1 を超える場合, すでに内部拡張は適用済みと判断し, 特殊化による内部拡張および追加による内部拡張をスキップする. これにより, 重複パターンの列挙が回避される.

標準形ではないパターンに対して外部拡張を行っても, 標準形となるパターンが生成されることはない¹⁴⁾. したがって, 標準形であるグラフパターン p に対してのみ外部拡張が適用される (13-16 行目). 外部拡張とは, p に対して辺を 1 つ追加する操作であり, 標準形を考慮し, (1) 最右パス上の頂点と最右頂点間に辺を追加する, (2) 最右パス上の頂点から新たな頂点をともなう辺を追加することにより達成される. なお (2) で導入される新たな頂点は, 新たなグラフパターンに対する最右頂点に相当し, I 中の 1 つの要素が割り当てられる.

一方, 外部拡張により得られたグラフパターンは, 現時点では標準形でなくても, その後の内部拡張により標準形となるものが存在する¹⁴⁾. したがって, 標準形であるか否かにかかわらず, p に対して, 追加による内部拡張 (7-10 行目) および特殊化による内部拡張 (3-6 行目) が適用される. 特殊化による内部拡張とは, $\text{rmv}(p)$ 中の (ϕ ではない最右の) データパターンを特殊化する操作であり, アイテム集合ならば ECLAT¹⁷⁾ や LCM¹⁰⁾, 系列ならば PrefixSpan⁸⁾ や BIDE¹¹⁾ など, 各データの属性のクラスに応じたアルゴリズムによって実現される. また追加による内部拡張とは, 直感的には, $\text{rmv}(p)$ に大きさ 1 のデータパターンを追加する操作であり, 正確には, $\text{rmv}(p)$ 中の ϕ を大きさ 1 のデータパターンに置き換える操作である. 追加による内部拡張は, データパターンである ϕ の特殊化と見なすことができ, 操作の観点からは, 特殊化による内部拡張と同様である. しかしこの操作は, パターンの解釈や半順序関係の判定において無視することのできた ϕ を他のデータパターンへと置換するものであり, 結果として, 考慮すべきデータパターンの増加をともなうものである. その意味で, 特殊化による内部拡張とは異なる拡張として扱っている.

図 6 に, 図 3 中のデータベース D を対象とした FMG のパターン列挙過程の一部を示す. P_1 に対して外部拡張を適用することで, P_4 が列挙される. P_4 は標準形ではないので, FMG が出力する解ではない. しかし, P_4 に対して特殊化による内部拡張を適用すること

で、標準形のパターン P_5 が列挙される。 P_5 に対して追加による内部拡張を適用すると、 P_6 と P_7 を得ることができる。

4.2 頻出強相関パターン発見アルゴリズム HFMG

頻出強相関パターン発見アルゴリズム HFMG は、内部および外部相互依存性とその上界値に基づく 5 つの枝刈りを FMG に導入したものであり、FMG の自然な拡張となっている。以下ではまず、各枝刈り手法について説明し、その後全体のアルゴリズムを示す。なお、下記の枝刈り手法の説明において、 ϕ_i は i 番目のデータが ϕ であることを表す。

枝刈り 1：内部相互依存性の上界値に基づく特殊化による内部拡張に関する枝刈り

$s(rmv(p)) = [elm_1, \dots, elm_{i-1}, elm_i, \phi_{i+1}, \dots, \phi_n]$ なるグラフパターン p に対する、特殊化による内部拡張により得られるグラフパターン t は、 $s(rmv(t)) = [elm_1, \dots, elm_{i-1}, elm'_i, \phi_{i+1}, \dots, \phi_n]$ *s.t.* $elm_i \preceq elm'_i$ なる最右頂点を持つ。このとき、 $s_\alpha = [elm_1, \dots, elm_{i-1}, \phi_i, \dots, \phi_n]$ とすると、 $s_\beta = [\phi_1, \dots, \phi_{i-1}, elm_i, \phi_{i+1}, \dots, \phi_n] \preceq s'_\beta = [\phi_1, \dots, \phi_{i-1}, elm'_i, \phi_{i+1}, \dots, \phi_n]$ より $up(rmv(t)) = up(rmv(t), [elm_1, \dots, elm_{i-1}, \phi_i, \dots, \phi_n]) \geq hconf_D^{in}(rmv(t))$ が成り立つ。したがって $h_{in} > up(rmv(t))$ のとき、 t および t に対する拡張により得られるグラフパターンは強相関パターンにならない。

枝刈り 2：内部相互依存性の上界値に基づく追加による内部拡張に関する枝刈り

$s(rmv(p)) = [elm_1, \dots, elm_{i-1}, \phi_i, \dots, \phi_n]$ なるグラフパターン p に対する、追加による内部拡張により得られるグラフパターン t は、 $s(rmv(t)) = [elm_1, \dots, elm_{i-1}, \phi_i, \dots, elm_j, \dots, \phi_n]$ なる最右頂点を持つ。このとき、 $s_\alpha = s(rmv(p))$ とすると、 $s_\beta = [\phi_1, \dots, \phi_n] \preceq s'_\beta = [\phi_1, \dots, elm_j, \dots, \phi_n]$ より $up(rmv(t), s(rmv(p))) \geq hconf_D^{in}(rmv(t))$ が成り立つ。したがって $h_{in} > up(rmv(t), s(rmv(p)))$ のとき、 t および t に対する拡張により得られるグラフパターンは強相関パターンにならない。

枝刈り 3：内部相互依存性に基づく外部拡張に関する枝刈り

グラフパターン $p = (V_p, E_p)$ と、 p に対する外部拡張により得られるパターン $t = (V_t, E_t)$ に対し、 $V_p \subseteq V_t$ が成り立つ。また、内部拡張は外部拡張適用前の最右頂点にのみ適用される。したがって $h_{in} > hconf_D^{in}(rmv(p))$ なるグラフパターン p に対する外部拡張により得られるグラフパターン t および t に対する拡張により得られるグラフパターンは強相関パターンにはならない。

枝刈り 4：外部相互依存性に基づく外部拡張に関する枝刈り

外部相互依存性の上界値の定義より、グラフパターン $p = (V_p, E_p)$ に対し、 $up(p, V_p) = sup_D^G(p) / \max_{u \in V_p} \{sup_D^G(u)\} = hconf_D^{ex}(p)$ が成り立つ。したがって、 p が $h_{ex} > hconf_D^{ex}(p)$

を満たすとき、 p に対する外部拡張により得られるグラフパターン t および t に対する拡張により得られるグラフパターンは強相関パターンにはならない。

枝刈り 5：外部相互依存性の上界値に基づく外部拡張に関する枝刈り

グラフパターン $p = (V_p, E_p)$ と、 p に対する外部拡張により得られるパターン $t = (V_t, E_p)$ を考える。このとき、 $s_\alpha = V_p \setminus \{rmv(p)\}$ とすると、 $p \preceq t$ および $s_\alpha \subseteq V_t$ より、 $s_\beta = \{rmv(p)\} \preceq s'_\beta = V_t \setminus s_\alpha$ が成り立つ。これより、 $up(t) = up(p, s_\alpha) \geq hconf_D^{ex}(t)$ が成り立つ。したがって、 $h_{ex} > up(p)$ のとき、 t および t に対する拡張により得られるグラフパターンは強相関パターンにはならない。

上記の枝刈り手法を導入した強相関パターン発見アルゴリズム HFMG を図 8 に示す。

Algorithm HFMG($D, \sigma, h_{in}, h_{ex}$)
1: set I be a set of all initial patterns
2: for each $p \in I$
3: HFMG-Enum($p, D, \sigma, h_{in}, h_{ex}$)
4: end for
Subroutine HFMG-Enum($p, D, \sigma, h_{in}, h_{ex}$)
1: if $sup_D^G(p) < \sigma$ then return
2: if degree of $rmv(p) > 1$ then goto line 13
3: $P_{ins} := \text{internalExpansionBySpecialization}(p)$
4: for each $t \in P_{ins}$
5: if $up(rmv(t)) \geq h_{in}$ then //枝刈り 1
6: HFMG-Enum($t, D, \sigma, h_{in}, h_{ex}$)
7: end for
8: $P_{ina} := \text{internalExpansionByAddition}(p)$
9: for each $t \in P_{ina}$
10: if $up(rmv(t), s(rmv(p))) \geq h_{in}$ then //枝刈り 2
11: HFMG-Enum($t, D, \sigma, h_{in}, h_{ex}$)
12: end for
13: if $\neg \text{isCanonical}(p)$ then return
14: if $hconf_D^{in}(rmv(p)) < h_{in}$ then return //枝刈り 3
15: if $hconf_D^{ex}(p) < h_{ex}$ then return //枝刈り 4
16: output p
17: $P_{ex} := \text{externalExpansion}(p, \sigma)$
18: for each $t \in P_{ex}$
19: if $up(t) \geq h_{ex}$ then //枝刈り 5
20: HFMG-Enum($t, D, \sigma, h_{in}, h_{ex}$)
21: end for

図 8 HFMG の擬似コード
Fig. 8 Pseudo code of HFMG.

HFMG において、枝刈り 1 が HFMG-Enum の 5 行目、枝刈り 2 が 10 行目、枝刈り 3 が 14 行目、枝刈り 4 が 15 行目、枝刈り 5 が 19 行目、にそれぞれ対応する。

HFMG-Enum の 5 行目では、その時点での反復で着目するグラフパターン p に対して、特殊化による内部拡張を適用して生成されたグラフパターン t に対し、最右頂点 $rmv(t)$ の内部相互依存性の上界値 $up(rmv(t))$ を計算する。ここで $up(rmv(t)) < h_{in}$ ならば、 t を拡張して得られる子パターンは $rmv(t)$ が内部相互依存性を持たないので、 t の拡張を枝刈りする（枝刈り 1）。同様に 10 行目では、 p に対して追加による内部拡張を適用して生成された t に対し、最右頂点 $rmv(t)$ の内部相互依存性の上界値 $up(rmv(t), s(rmv(p)))$ を計算する。このとき $up(rmv(t), s(rmv(p))) < h_{in}$ ならば、 t を拡張して得られる子パターンは $rmv(t)$ が内部相互依存性を持たないので、 t の拡張を枝刈りする（枝刈り 2）。枝刈り 3 と 4 は、それぞれ 14, 15 行目で適用される。グラフパターン p に対し、その内部および外部相互依存性の値そのものを利用して枝刈りを適用する。一方 19 行目では、 p を外部拡張して得られるグラフパターン t の外部相互依存性の上界値 $up(t)$ を計算する。 $up(t) < h_{ex}$ ならば、 t を拡張して得られる子パターンは、必ず t を部分構造として持っており、かつ t が外部相互依存性を持たないので、 t の拡張を枝刈りする（枝刈り 5）。

ここで、図 6 を用いて HFMG の挙動について説明する。 $\sigma = 0.5$, $h_{in} = 0.6$, $h_{ex} = 0.6$ とするとき、 P_1 に対して、式 (5) と式 (6) より、 $hconf_D^{in}(u_1) \geq h_{in}$, $hconf_D^{in}(u_2) \geq h_{in}$ が成り立つ。また、 $hconf_D^{ex}(P_1) = \sup_D^G(P_1) / \max\{\sup_D^G(u_1), \sup_D^G(u_2)\} = (3/4) / \max\{(3/4), (4/4)\} = 0.75 \geq h_{ex}$ が成り立つ。したがって、 P_1 は頻出強相関パターンである。 P_1 に対して特殊化による内部拡張を適用することで P_2 が得られる。HFMG-Enum の 5 行目で、 $rmv(P_2) = u_2$ に対して、内部相互依存性の上界値を計算すると、 $up(u_2) = \sup_D^s(u_2) / \max\{\sup_D^e(\{a, c\})\} = (9/20) / (14/20) = 0.64 \geq h_{in}$ より、次の HFMG-Enum の反復に移る。その後 15 行目で外部相互依存性を計算すると、 $hconf_D^{ex}(P_2) = \sup_D^G(P_2) / \max\{\sup_D^G(u_1), \sup_D^G(u_2)\} = (2/4) / \max\{(3/4), (4/4)\} = 0.5 < h_{ex}$ となる。このとき、枝刈り 4 に基づき、 P_2 と 17 行目以降で行われる P_2 に対する外部拡張を枝刈りする。したがって、 P_3 が列挙されることはない。

P_1 に対して外部拡張と特殊化による内部拡張を適用することによって、 P_5 が列挙される。 P_5 に対して追加による内部拡張を行うと P_6 と P_7 が列挙される。10 行目で $rmv(P_7) = u_3$ に対して内部相互依存性の上界値を計算すると、式 (7) より $up(u_3, [\{a, c\}, \phi]) < h_{in}$ である。このとき、枝刈り 2 に基づき、 P_7 と P_7 に対する拡張を枝刈りする。一方、 $up(rmv(P_6), [\{a, c\}, \phi]) \geq h_{in}$ より、 P_6 は枝刈りされない。

P_6 に対して特殊化による内部拡張を 2 回適用すると P_8 が列挙される。 P_8 に対し、14 行目で $rmv(P_8) = u_3$ の内部相互依存性を計算すると、 $hconf_D^{in}(u_3) = \sup_D^s(u_3) / \max\{\sup_D^e(\{a, c\}), \sup_D^e(\langle AAC \rangle)\} = (8/20) / \max\{(14/20), (9/20)\} = 0.6 \geq h_{in}$ である。また、同様に 15 行目で外部相互依存性を計算すると、 $hconf_D^{ex}(P_8) = \sup_D^G(P_8) / \max\{\sup_D^G(u_1), \sup_D^G(u_2), \sup_D^G(u_3)\} = (3/4) / \max\{(3/4), (4/4), (4/4)\} = 0.75 \geq h_{ex}$ が成り立つ。したがって、 P_8 は頻出強相関パターンである。 P_8 に対して特殊化による内部拡張を行うと P_9 が列挙される。このとき、5 行目で $rmv(P_9) = u_3$ に対して内部相互依存性の上界値を計算すると、 $up(u_3, [\{a, c\}, \phi]) = \sup_D^s(u_3) / \max(\sup_D^e(\{a, c\})) = (4/20) / (14/20) = 0.29 < h_{in}$ である。よって、枝刈り 1 に基づき、 P_9 と P_9 に対する拡張を枝刈りする。

一方 P_8 に対して外部拡張を行うと P_{10} が列挙され、さらに内部拡張を 2 回適用すると P_{11} が列挙される。10 行目で $rmv(P_{11}) = u_4$ に対して内部相互依存性の上界値を計算すると、 $up(u_4, [\{b, c\}, \phi]) = \sup_D^s(u_4) / \max\{\sup_D^e(\{b, c\})\} = (7/20) / (8/20) = 0.88 \geq h_{in}$ である。よって、次の HFMG-Enum の反復に移り、特殊化による内部拡張を 2 回適用することで、 P_{13} を列挙できる。なお、 P_{13} は頻出強相関パターンである。その後 14 行目で $rmv(P_{11}) = u_4$ の内部相互依存性を計算すると、 $hconf_D^{in}(u_4) = \sup_D^s(u_4) / \max\{\sup_D^e(\{b, c\}), \sup_D^e(\langle C \rangle)\} = (7/20) / \max\{(8/20), (14/20)\} = 0.5 < h_{in}$ である。このとき、枝刈り 3 に基づき、 P_{11} と 17 行目以降で行われる P_{11} に対する外部拡張を枝刈りする。よって、 P_{12} が列挙されることはない。

P_{13} に対して外部拡張を行うと P_{14} が列挙される。19 行目で外部相互依存性の上界値を計算すると、 $up(P_{13}) = \sup_D^G(P_{13}) / \max\{\sup_D^G(u_1), \sup_D^G(u_2), \sup_D^G(u_3), \sup_D^G(u_3)\} = (2/4) / \max\{(3/4), (4/4), (4/4), (4/4)\} = 0.5 < h_{ex}$ である。したがって、枝刈り 5 に基づき、 P_{14} と P_{14} に対する拡張を枝刈りする。

5. 関連研究

近年、グラフデータベースからの頻出パターン発見に関する研究はさかに行われているが、頂点に様々なデータが存在する複合構造グラフを対象とするものは少ない。DAG Miner²⁾ は、頂点がアイテム集合からなる有向非巡回グラフ (DAG) データベースから、pyramid pattern と呼ばれる特別な形状をした頻出パターンを発見する手法である。頂点の候補となるアイテム集合をあらかじめすべて求めておき、そのアイテム集合を頂点とする pyramid pattern を発見する。

また、頂点の類似性に着目した手法として、COPINE⁹⁾ や CoPam⁶⁾ があげられる。COPINE は、頂点がアイテム集合からなる単一グラフを対象に、共通のアイテム集合を持つ頂点からなるグラフパターンを発見する手法である。一方 CoPam は、頂点が特徴ベクトルからなる単一グラフから、密結合かつ各頂点の特徴ベクトルが類似しているグラフパターンを発見する手法である。これらの手法と比較し、HFMMG は頂点にアイテム集合だけでなく系列データなど複数の（構造）データを含む複合構造グラフを扱うことができ、頂点に類似性ではなく依存性の存在するパターンのみを発見する点で異なる。

強相関パターンは、アイテム集合マイニングの分野で提案され^{12),13)}、支持度は低いが強相関依存性を有するパターンの発見を目的としている。グラフデータから強相関パターンを発見する手法として、これまでに HSG⁷⁾ が提案されている。HSG では、グラフデータベースから相互依存性の存在するグラフパターンの集合を列挙する。一方 HFMMG では、(1) 頂点のデータ集合と各データ間の内部相互依存性、(2) グラフ全体と頂点間の外部相互依存性に着目し、パターンを列挙する。

6. 評価実験

提案手法の有効性を評価するために、FMG および HFMMG を JAVA 言語で実装し、Windows マシン (CPU: Intel(R) Xeon(R) CPU X5260 3.3 GHz, メインメモリ: 32 GByte) 上で評価実験を行った。評価には、KEGG^{4),*1)}の代謝パスウェイデータを用いた。実験で使ったデータを表 1 に示す。代謝パスウェイデータは、KEGG 上で公開されている特定の機能の代謝パスウェイを各生物種ごとに複合構造グラフで表現したものである。なお、生物種によって代謝機能が存在しない場合や、代謝機能の解明があまり進んでいない場合が存在することを考慮し、本来 KEGG 中に存在する代謝パスウェイデータの一部を使用してい

表 1 データセット
Table 1 Data sets.

Name	#	Ave. Node	Ave. Edge	Ave. Item	# Item	Ave. Sequence
Riboflavin	100	7.8	7.1	2.2	100	309.5
Sulfur	500	7.5	8.8	3.0	482	375.9
Cysteine	751	15.2	33.5	4.2	1,086	428.5

: データ数, Ave. Node : 平均頂点数, Ave. Edge : 平均辺数, Ave. Item : アイテム集合の平均アイテム数, # Item : アイテム集合の全アイテム数, Ave. Sequence : 平均系列長

*1 <http://www.genome.ad.jp/kegg/>

る。酵素と酵素の反応関係を辺で結び、その反応に関与する化合物を辺ラベルとして与える。また、今回は反応関係の方向を無視し、無向辺として扱う。酵素が持つ情報として、酵素の機能を発現する 20 種類のアミノ酸からなる配列が与えられており、これを系列データとして扱う。系列長は生物種、酵素によって異なる。また、アミノ酸配列中に認められる小さい構造部分を指すモチーフが複数存在し、これをアイテム集合データとして扱う。本実験では、酵素が持つ 2 つの情報に着目し、代謝パスウェイデータの各頂点に与えるデータ集合を [アイテム集合, 系列] と定義した。

6.1 実験 1

提案手法の性能を評価するために、Riboflavin データを用いて FMG と HFMMG の性能比較実験を行った。Riboflavin データは、比較的小さなデータベースであり、すべての頻出パターンの列挙が比較的短時間で終了することを想定した設定である。

最小支持度 (σ) を徐々に下げながら、実行時間 (time), 頂点数 2 以上の解パターン数 (pat.), 生成した候補パターン数 (cand.) を計測した。ここで、解パターンとは FMG では頻出パターン、HFMMG では頻出強相関パターンを指す。また生成した候補パターンとは内部および外部拡張によって生成された子パターンすべてを指す。したがって、頻出パターンや非頻出パターン、HFMMG では、依存性の存在しないパターンもすべて含む。なお、頂点に含まれる最小のデータ数を 2 とし、最小内部相互依存度と最小外部相互依存度をともに 0.7 とした場合 ($h_{in} = h_{ex} = 0.7$) と、0.5 にした場合 ($h_{in} = h_{ex} = 0.5$) で実験を行った。実験結果を表 2 に示す。なお、図中の ‘-’ は、2 時間以内に実験結果が得られなかったため、途中で実験を中断したこと、 < 0.1 は値が 0.1 より小さいことをそれぞれ示す。

FMG では、 $\sigma = 0.7$ で頻出パターンが約 55 万個得られ、生成された候補パターンは約 1 億 2 千万個と非常に多く、 $\sigma = 0.6$ の時点でパターンの列挙が困難となったが、HFMMG では、より低い支持度のパターンを発見することに成功した。また、FMG において生成された候補パターンは、高支持度の場合でも非常に多い。一方、HFMMG は、高支持度の場合は少ないが、最小支持度が低くなるにつれて膨大な数の候補パターンを生成していることが分かる。これらのことから、複合構造グラフから頻出パターンを発見する際は、列挙されるパターンに対する何らかの制限を加え、候補として生成されるパターンを減らすことが重要であるといえる。

6.2 実験 2

相互依存性の影響を評価するため、Riboflavin データを対象に、最小相互依存度 h_{in} および h_{ex} をそれぞれ変化させ、実行時間 (time), 頂点数 2 以上の解パターン数 (pat.), 生成

表 2 実験 1 の結果
Table 2 Result of Experiment 1.

σ	time(sec.)			pat. (#/1,000)			cand. (#/1,000)		
	FMG	HFMG	HFMG	FMG	HFMG	HFMG	FMG	HFMG	HFMG
		(0.7)	(0.5)		(0.7)	(0.5)		(0.7)	(0.5)
0.9	3.0	1.0	1.1	1.2	0	0	70.2	0	<0.1
0.8	69.6	1.2	1.3	19.0	0	<0.1	1,864.7	<0.1	<0.1
0.7	6,289.6	1.7	1.8	551.1	<0.1	<0.1	125,095.0	1.2	3.0
0.6	—	2.3	4.9	—	0.1	0.5	—	12.3	67.9
0.5	—	4.7	44.6	—	0.6	5.8	—	76.4	1,383.6
0.4	—	6.6	861.9	—	0.9	107.4	—	121.0	38,067.3
0.3	—	7.4	3,127.6	—	1.0	397.5	—	128.5	155,501.1
0.2	—	8.0	3,361.3	—	1.0	444.7	—	128.5	171,525.1
0.1	—	10.0	3,544.7	—	1.0	444.9	—	130.0	172,756.7
0.05	—	29.6	4,982.2	—	1.0	449.8	—	205.9	205,424.7
0.03	—	353.2	—	—	53.6	—	—	20,209.8	—

表 3 実験 2 の結果
Table 3 Result of Experiment 2.

		time (sec.)		pat. (#/1,000)		cand. (#/1,000)	
h_{ex}	h_{in}	$\sigma = 0.5$	$\sigma = 0.1$	$\sigma = 0.5$	$\sigma = 0.1$	$\sigma = 0.5$	$\sigma = 0.1$
0.5	0.7	7.9	125.0	2.2	56.5	269.2	7,828.7
	0.6	18.4	843.2	3.6	213.1	630.6	47,720.5
	0.5	44.3	3,549.4	5.8	444.9	1,383.6	172,756.7
	0.4	91.7	9,118.2	10.7	689.7	3,036.9	441,411.9
	0.3	345.0	51,509.2	29.8	1,624.7	10,923.4	2,373,220.7
h_{in}	h_{ex}	$\sigma = 0.5$	$\sigma = 0.1$	$\sigma = 0.5$	$\sigma = 0.1$	$\sigma = 0.5$	$\sigma = 0.1$
0.5	0.7	25.0	79.9	2.4	5.3	599.5	2,275.1
	0.6	42.4	619.4	5.2	60.3	1,261.1	24,270.7
	0.5	44.3	3,549.4	5.8	444.9	1,383.6	172,756.7
	0.4	44.5	12,221.7	5.8	1,912.1	1,383.6	688,983.2
	0.3	45.1	33,268.1	5.8	7,309.1	1,383.6	2,134,111.2

した候補パターン数 (cand.) を計測した。なお実験 1 と同様、頂点に含まれる最小のデータ数を 2 とした。また最小支持度は、 $\sigma = 0.5$ と $\sigma = 0.1$ を利用した。実験結果を表 3 に示す。

最小外部相互依存度を $h_{ex} = 0.5$ に固定し、最小内部相互依存度 (h_{in}) を減少させた場合、実行時間や解パターン数、候補パターン数がともに増加していることが分かる。特に、

$h_{in} = 0.4$ や $h_{in} = 0.3$ ではその傾向が顕著である。これは、相互依存性があると判断される頂点パターンの数が増加した結果であると考えられる。一方、最小外部相互依存度 (h_{ex}) の影響に関しては、 $\sigma = 0.1$ に対しては内部相互依存性と同様の傾向が認められるが、 $\sigma = 0.5$ に対しては $h_{ex} = 0.5$ 以下で、実行時間や得られるパターン数において違いが見られなかった。これは、支持度の影響によるものであると考えられる。

次に、支持度による違いについて考察する。 h_{in} を 0.7 から 0.3 へ減少させた場合、 $\sigma = 0.5$ では約 13 倍、 $\sigma = 0.1$ では約 28 倍の解パターンが抽出される。また、 h_{ex} を 0.7 から 0.3 へと減少させた場合は、 $\sigma = 0.5$ では約 2 倍の解パターンしか得られないが、 $\sigma = 0.1$ では約 1,380 倍程度もの解パターンが獲得されている。これらの結果から、相互依存度の影響は、特に低支持度の場合において顕著であることが推測される。

6.3 実験 3

HFMG の性能を評価するために、Sulfur データおよび Cysteine データを用いて評価実験を行った。 σ 、 h_{in} および h_{ex} の各パラメータを変えながら、実行時間 (time)、頂点数 2 以上の頻出強相関パターン数 (pat.)、生成した候補パターン数 (cand.) を計測した。また、発見されるパターンを (1) アイテム集合パターンの最小のアイテム数を 2、(2) 系列パターンの最小系列長を 5、という条件を満たすものに限定した。実験結果を表 4 に示す。

最小内部相互依存度や最小外部相互依存度を 0.9 と設定したとき、ほとんどパターンは発見されなかった。特に、最小内部相互依存度を 0.9 と設定した場合、最小外部相互依存度を下げてもパターンが発見されていない。これは、頂点のパターン集合に内部相互依存性がないと判断されるため、外部拡張が適用されないためである。一方、候補パターン数と強相関パターン数を比較すると、約 1,000 倍程度の差が存在する。したがって、データベースの中に存在する多くの偶発的で依存性のないパターンを排除し、強相関パターンのみを効率的に発見することができたといえる。また、両データともに $\sigma = 0.03$ の場合で強相関パターンの発見に成功しており、低支持度のパターンの発見に有効であることを確認できた。

これらの実験を通して、HFMG が複合構造グラフデータベースから頻出強相関パターンを発見できることを確認することができた。単純なアルゴリズム FMG を用いれば解の数が爆発してパターンを発見できないような場合でも、得られる解の種類を強相関パターンに限定し、HFMG を適用することで、パターンの発見が可能となる。

ここで、HFMG を適用して得られたパターンの例を図 9 に示す。このパターンは、Cysteine データを用いて、 $\sigma = 0.3$ 、 $h_{in} = 0.5$ 、 $h_{ex} = 0.5$ と設定したときに得られたものである。各頂点は、モチーフの集合とアミノ酸配列から構成されており、Cysteine データで

表 4 実験 3 の結果
Table 4 Result of Experiment 3.

σ	h_{in}	h_{ex}	time[sec.]		pat.(#/1,000)		cand.(#/1,000)	
			Sulfur	Cysteine	Sulfur	Cysteine	Sulfur	Cysteine
0.3	0.9	0.9	16.9	63.6	0	0	<0.1	<0.1
		0.7	16.8	62.6	0	0	<0.1	<0.1
		0.5	16.8	64.4	0	0	<0.1	<0.1
	0.7	0.9	16.8	63.0	0	0	<0.1	0.2
		0.7	16.6	69.0	0	<0.1	<0.1	1.0
		0.5	16.8	84.6	0	0.2	24	3.1
	0.5	0.9	16.9	74.5	0	0	100	7.3
		0.7	16.9	151.1	0	0.3	100	45.2
		0.5	16.7	1,202.9	<0.1	4.7	112	559.0
0.1	0.9	0.9	36.8	146.4	0	0	<0.1	<0.1
		0.7	37.6	142.9	0	0	<0.1	<0.1
		0.5	37.2	147.2	0	0	<0.1	<0.1
	0.7	0.9	45.2	146.2	0	0	48.4	1.6
		0.7	46.2	160.3	0	<0.1	48.4	3.3
		0.5	44.4	192.6	0	0.2	48.4	7.7
	0.5	0.9	77.2	179.5	0	0	296.0	51.0
		0.7	83.7	371.1	0.1	0.3	368.1	104.2
		0.5	83.7	2,980.8	0.1	5.5	369.6	956.2
0.05	0.9	0.9	64.8	271.2	0	0	<0.1	<0.1
		0.7	65.0	268.5	0	0	<0.1	<0.1
		0.5	64.9	271.2	0	0	<0.1	<0.1
	0.7	0.9	74.7	272.5	0	0	48.4	1.9
		0.7	75.8	291.2	0	<0.1	48.4	3.6
		0.5	73.2	327.7	0	0.2	48.4	8.2
	0.5	0.9	113.2	321.8	<0.1	0	401.6	81.5
		0.7	1,161.3	579.1	24.0	0.3	18,822.0	149.2
		0.5	2,936.8	4,864.3	64.7	19.2	49,844.6	5,730.6
0.03	0.9	0.9	165.6	490.0	0	0	0.4	<0.1
		0.7	174.0	484.0	0	0	0.4	<0.1
		0.5	173.2	486.0	0	0	0.4	<0.1
	0.7	0.9	181.0	499.2	0	0	295.1	8.1
		0.7	179.7	540.4	0	<0.1	295.1	36.5
		0.5	182.8	652.8	0	0.2	295.1	105.1
	0.5	0.9	266.6	746.4	<0.1	0	1,321.1	1,703.4
		0.7	1,749.8	1,230.4	24.2	0.3	21,289.0	1,981.0
		0.5	4,752.1	9,822.5	66.1	19.2	55,417.9	13,271.2

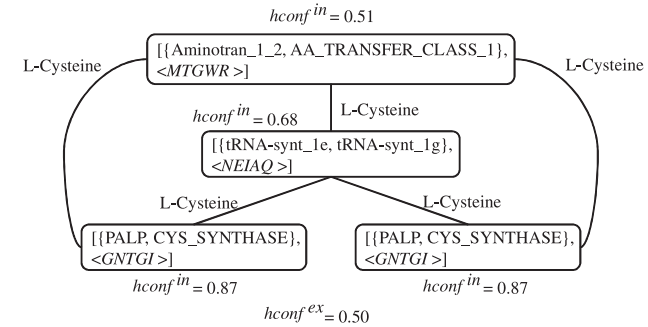


図 9 Cysteine データから発見された頻出強相関パターンの例

Fig. 9 An example of a discovered frequent hyperclique pattern from Cysteine database.

重要な L-Cysteine という化合物がすべての辺に辺ラベルとして存在する．したがって，多くの生物種中に L-Cysteine を中心とする密な酵素反応が存在することが確認でき，その酵素反応には共通のモチーフとそれに対応するアミノ酸配列が必要であるといったことが推測できる．このように，従来のグラフマイニング手法^{15),16)}ではその表現力の不足により，また FMG ではその効率の悪さが原因で発見することが困難であった複合構造グラフパターンを，HFMG を用いることで獲得できることが確認された．

7. ま と め

本稿では，グラフの頂点に複数のデータが存在する複合構造グラフを対象とする頻出パターン発見において，得られるパターンの数が爆発するという問題を解決する手法として，内部構造と外部構造に相互依存性が存在する強相関パターン発見手法 HFMG を提案した．HFMG により，グラフ構造だけでなく，対象が本来持つ情報を損なわずにパターンを抽出し，データベースに存在する依存性をとらえることができ，さらに従来では発見できなかった支持度が低く依存性の高いパターンを抽出することができた．

本稿では特に，代謝パスウェイデータを対象に評価を行ったが，HFMG の他領域への適用可能性の検討やその有効性評価は，今後の大きな課題の 1 つである．また別の課題として，複合構造グラフ以外の複雑な構造を持つデータに対するパターン発見手法の開発があげられる．一般に，複合的なデータを扱う場合には，頻出パターン発見を正面から行うことは困難となる．そこで，パターンの全列挙を試みるのではなく，近似的な解を得る手法^{1),5)}などを利用して，有益であると期待できるパターンのみを効率良く抽出する手法を開発する

ことも、今後の重要な課題であると考えている。

謝辞 有益なご指摘を賜りました査読者の方々に深く感謝いたします。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 21700168 および基盤研究（B）：課題番号 20300038）による。

参 考 文 献

- 1) Chen, C., Yan, X., Zhu, F. and Han, J.: gApprox: Mining Frequent Approximate Patterns from a Massive Network, *Proc. 7th IEEE International Conference on Data Mining (ICDM'07)*, pp.445–450 (2007).
- 2) Chen, Y.L., Kao, H. and Ko, M.: Mining DAG Patterns from DAG Databases, *Proc. 5th International Conference on Web-Age Information Management (WAIM'04)*, pp.579–588 (2004).
- 3) Han, J., Cheng, H., Xin, D. and Yan, X.: Frequent pattern mining: Current status and future directions, *Data Mining and Knowledge Discovery*, Vol.15, No.1, pp.55–86 (2007).
- 4) Kanehisa, M., Goto, S., Kawashima, S., Okuno, Y. and Hattori, M.: The KEGG resource for deciphering the genome, *Nucleic Acids Research*, Vol.32, pp.D277–D280 (2004).
- 5) Ketkar, N.S., Holder, L.B. and Cook, D.J.: Mining in the Proximity of Subgraphs, *Proc. Workshop on Link Analysis: Dynamics and Static of Large Networks (LinkKDD2006)*, pp.37–44 (2006).
- 6) Moser, F., Colak, R., Rafiey, A. and Ester, M.: Mining cohesive patterns from graphs with feature vectors, *Proc. 9th SIAM Conference on Data Mining*, pp.593–604 (2009).
- 7) Ozaki, T. and Ohkawa, T.: Mining Correlated Subgraphs in Graph Databases, *The 12th Pacific-Asia Conference on Knowledge Discovery and Data Mining (PAKDD'08)*, pp.272–283 (2008).
- 8) Pei, J., Han, J., Mortazavi-Asl, B., Pnto, H., Chen, Q., Dayal, U. and Hsu, M.: PrefixSpan: Mining Sequential Patterns Efficiently by Prefix-Projected Pattern Growth, *Proc. 17th International Conference on Data Engineering (ICDE'01)*, pp.215–224 (2001).
- 9) Seki, M. and Sese, J.: Identification of Active Biological Networks and Common Expression Conditions, *Proc. 8th IEEE International Conference on BioInformatics and BioEngineering (BIBE'08)*, pp.8–10 (2008).
- 10) Uno, T., Asai, T., Uchida, Y. and Arimura, H.: An Efficient Algorithm for Enumerating Closed Patterns in Transaction Databases, *Proc. 7th International Conference on Discovery Science (DS'04)*, pp.16–31 (2004).
- 11) Wang, J. and Han, J.: BIDE: Efficient Mining of Frequent Closed Sequences, *Proc. 20th International Conference on Data Engineering (ICDE'04)*, pp.79–90 (2004).
- 12) Xiong, H., Tan, P.N. and Kumar, V.: Hyperclique pattern discovery, *Data Mining and Knowledge Discovery*, Vol.13, No.2, pp.219–242 (2006).
- 13) Xiong, H., Tan, P.N. and Kumar, V.: Mining strong affinity association patterns in data sets with skewed support distribution, *Proc. 3rd IEEE International Conference on Data Mining (ICDM'03)*, pp.387–394 (2003).
- 14) 山本 翼, 尾崎知伸, 大川剛直: 構造データ集合からなるグラフデータベースからの頻出パターン発見, *情報処理学会論文誌: データベース*, Vol.1, No.1, pp.26–35 (2008).
- 15) Yan, X. and Han, J.: gSpan: Graph-Based Substructure Pattern Mining, *Proc. 2nd IEEE International Conference on Data Mining (ICDM'02)*, pp.721–724 (2002).
- 16) Yan, X. and Han, J.: CloseGraph: Mining closed frequent graph patterns, *Proc. 9th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp.286–295 (2003).
- 17) Zaki, M.J.: Scalable Algorithms for Association Mining, *IEEE Trans. Knowledge and Data Engineering*, Vol.12, No.3, pp.372–390 (2000).
- 18) Zhu, F., Yan, X., Han, J. and Yu, P.S.: gPrune: A Constraint Pushing Framework for Graph Pattern Mining, *Proc. 11th Pacific-Asia Conference on Knowledge Discovery and Data Mining (PAKDD'07)*, pp.388–400 (2007).

(平成 21 年 3 月 20 日受付)

(平成 21 年 7 月 10 日採録)

(担当編集委員 森本 康彦)



山本 翼

1984 年生。2007 年神戸大学工学部情報知能工学科卒業。2009 年神戸大学大学院工学研究科情報知能学専攻博士前期課程修了。現在、富士通株式会社勤務。修士（工学）。在学中、主に構造データマイニングの研究に従事。



三好 裕樹

1986 年生。2009 年神戸大学工学部情報知能工学科卒業。現在、神戸大学大学院工学研究科情報知能学専攻博士前期課程に在学中。構造データマイニングの研究に従事。



尾崎 知伸

1973 年生。1996 年慶應義塾大学総合政策学部卒業。1998 年慶應義塾大学大学院政策・メディア研究科前期修士課程修了。2002 年同研究科講師。2005 年神戸大学大学院自然科学研究科助手。現在、神戸大学自然科学系先端融合研究環助教。博士（政策・メディア）。帰納論理プログラミング、構造データマイニング等の研究に従事。人工知能学会会員。



大川 剛直（正会員）

1963 年生。1986 年大阪大学工学部通信工学科卒業。1988 年大阪大学大学院工学研究科通信工学専攻博士前期課程修了。大阪大学助手、講師、助教授を経て、2005 年神戸大学大学院自然科学研究科教授。現在、神戸大学大学院工学研究科教授。博士（工学）。知的ソフトウェア、バイオインフォマティクス等の研究に従事。IEEE、人工知能学会、電子情報通信学会、電気学会等各会員。