1D1-2in

単一グラフ系列からの頻出パターン列挙

Mining Frequent Transformation Subsequences in a Graph Sequence

山岡歩 Ayumu Yamaoka　猪口明博 Akihiro Inokuchi　鷲尾隆 Takashi Washio

大阪大学 産業科学研究所
The Institute of Scientific and Industrial Research, Osaka University

Recently, much attention has been given to frequent pattern mining from graph sequences, because graph sequences can be used to model objects for many real world applications. For example, we can represent a human network as a single graph sequence. In this paper, we propose a method for mining frequent transformation subsequence patterns from a graph sequence. To efficiently mine the patterns, we propose a scalable support measure in the graph sequence data. Our performance study shows that the proposed method is efficient and scalable for mining FTSs from a graph sequence.

1. はじめに

近年グラフ系列からの頻出パターンマイニングであるグラフ系列マイニングが注目されている [Inokuchi 2008, Inokuchi 2010]．例えば，人間関係ネットワークにおいて，人をグラフの頂点，人と人の関係をグラフの辺で表すと，ある時点での人間関係ネットワークをグラフにより表現することが出来る．さらに，人がネットワークに参加，脱退することによりグラフの頂点や辺は増減する．すなわち，時間の経過とともにその構造が変化する人間関係ネットワークは，グラフの系列として表すことが可能である．また，Web におけるリンク構造や遺伝子ネットワークなど，変化する構造を持つものは数多く存在するため，変化する構造を表現した大規模グラフ系列をマイニングし，グラフ系列に埋もれた構造変化パターンを発見することは有用であると考えられる．

グラフ系列マイニングの手法として GTRACE(Graph TRAnsformation sequenCE mining) が提案されている [Inokuchi 2008]．これは図 1(a) に示すグラフ系列の集合から図 1(b) に示すような頻出パターンを列挙する手法である．ここで，GTRACE を適用できるデータは比較的小さく短いグラフ系列の集合である．しかし，より一般的なデータである単一グラフ系列に対して GTRACE を適用することは難しい．

そこで，本稿では単一の長いグラフ系列から FTS(Frequent Transformation Subsequences) と呼ばれる頻出パターンを列挙する手法として SiGTRACE を提案する．また，評価実験から提案手法が効率的であることを示す．本稿では無向グラフについてのみ議論しているが，提案手法は一般性を失うことなく有向グラフに対しても適用可能である．

2. グラフ系列の表現

図 1(a) は観測されたグラフ系列の例である．$g^{(j)}$ $(j \in \mathbb{N})$ はグラフ系列における j 番目のグラフである．観測されたグラフ系列に対して，以下の 2 つの仮定を置く．

- 連続する 2 つのグラフ $g^{(j)}$ と $g^{(j+1)}$ の間で，ごく一部の構造のみが変化する．
- 各グラフは疎グラフである．

前述の人間関係ネットワークを考えたとき，人間関係が連続する 2 グラフ間で大きく変わることは考え難く，さらに多くの人が他の大多数の人と関係を有しているとも考え難い．つまり，上記の仮定があてはまる．その他，実世界の多くの構造の変化はこれらの仮定を満たしていると考えられる．

連絡先: 山岡歩，大阪大学 産業科学研究所，567-0047 大阪府茨木市美穂ケ丘 8-1，yamaoka@ar.sanken.osaka-u.ac.jp

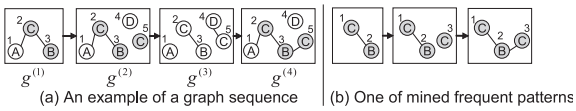

図 1: グラフ系列の例と探索された頻出パターンの例

ラベル付きグラフを $g = (V, E, L, f)$ という四つ組で表す．ここで $V = \{v_1, \cdots, v_n\}$，$E = \{(v, v') \mid (v, v') \in V \times V\}$，$L$ はそれぞれ頂点集合，辺集合，頂点ラベルと辺ラベルの集合であり，$f : V \cup E \to L$ である．グラフ系列 d をグラフ $g^{(j)}$ を用いて $d = \langle g^{(1)}\ g^{(2)} \cdots g^{(n)} \rangle$ と表す．また，グラフの各頂点 v には頂点 ID $id(v)$ を付与する．頂点 ID の集合を $ID_V(d) = \{id(v) \mid v \in V(g^{(j)}), g^{(j)} \in d\}$，各辺に対応する頂点 ID の組の集合を $ID_E(d) = \{(id(v), id(v')) \mid (v, v') \in E(g^{(j)}), g^{(j)} \in d\}$ と表す．

ここで，グラフ系列を簡潔に表現するために，系列中の連続する 2 つのグラフ $g^{(j)}$ と $g^{(j+1)}$ の差異に着目する．

定義 1. 観測されたグラフ系列 $d = \langle g^{(1)} \cdots g^{(n)} \rangle$ の各グラフ $g^{(j)}$ を外部状態グラフと呼ぶ．さらに，連続する 2 つの外部状態グラフ $g^{(j)}, g^{(j+1)}$ の間を補完するグラフ系列を $d^{(j)} = \langle g^{(j,1)} \cdots g^{(j,m_j)} \rangle$ と表し，各グラフ $g^{(j,k)}$ を内部状態グラフと呼ぶ．ただし，$g^{(j,1)} = g^{(j)}, g^{(j,m_j)} = g^{(j+1)}$ とする．グラフ系列 d は補完系列 $d = \langle d^{(1)} \cdots d^{(n-1)} \rangle$ で表される． ■

さらに，連続する外部状態グラフの編集距離は最小のものに制限する．

定義 2. 頂点や辺の追加，削除，ラベルの変更を変換の最小単位とし，編集距離 1 とする．内部状態グラフ系列 $d^{(j)} = \langle g^{(j,1)}\ g^{(j,2)} \cdots g^{(j,m_j)} \rangle$ の連続する 2 つの内部状態グラフの編集距離は 1 である．また，内部状態グラフ系列中の任意の 2 つの内部状態グラフの編集距離は最小である． ■

変換は次に示す変換規則によって表される．

定義 3. $g^{(j,k)}$ を $g^{(j,k+1)}$ へ変換する変換規則を $tr^{(j,k)}_{[o_{jk}, l_{jk}]}$ で表す．ただし，以下の 3 つの条件を満たす．

表 1: グラフ系列データのための変換規則

頂点追加 $vi^{(j,k)}_{[u,l]}$	ラベルが l，頂点 ID が u である頂点を $g^{(j,k)}$ へ追加し，$g^{(j,k+1)}$ へ変換
頂点削除 $vd^{(j,k)}_{[u,\bullet]}$	頂点 ID が u である頂点を $g^{(j,k)}$ から削除し $g^{(j,k+1)}$ へ変換
頂点ラベル変更 $vr^{(j,k)}_{[u,l]}$	頂点 ID が u である頂点のラベルを l に変更し，$g^{(j,k)}$ を $g^{(j,k+1)}$ へ変換
辺追加 $ei^{(j,k)}_{[(u_1,u_2),l]}$	頂点 ID が u_1 と u_2 である頂点間にラベル l の辺を追加し，$g^{(j,k)}$ を $g^{(j,k+1)}$ へ変換
辺削除 $ed^{(j,k)}_{[(u_1,u_2),\bullet]}$	頂点 ID が u_1 と u_2 である頂点間から辺を削除し，$g^{(j,k)}$ を $g^{(j,k+1)}$ へ変換
辺ラベル変更 $er^{(j,k)}_{[(u_1,u_2),l]}$	頂点 ID が u_1 と u_2 である頂点間の辺のラベルを l へ変更し，$g^{(j,k)}$ を $g^{(j,k+1)}$ へ変換

頂点削除と辺削除の変換規則，vd と ed は頂点 ID の指定のみで変換可能なので，引数 l はダミー変数であり，‘•’ で表す．$u_1 \le u_2$ を必ず満たす．

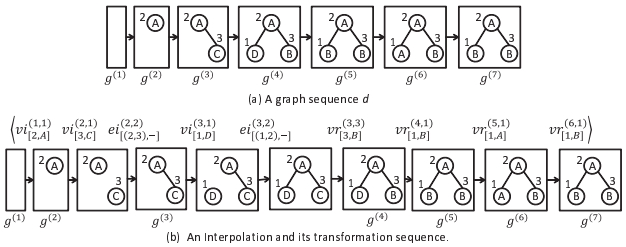

図 2: グラフ系列とその変換規則

- $tr \in \{vi,\ vd,\ vr,\ ei,\ ed,\ er\}$
- $o_{jk} \in ID_V(d)$ あるいは $o_{jk} \in ID_E(d)$
- $l_{jk} \in L$ ■

本稿では，簡単に表記するため変換規則 $tr^{(j,k)}_{[o_{jk},l_{jk}]}$ を $tr^{(j,k)}_{[o,l]}$ と略記する．GTRACE では，表 1 に示す 6 つの変換規則を定義し，グラフ間の差異を表現する．変換規則を用いて表現されたグラフ系列を変換系列と呼び，以下のように定義する．

定義 4. 内部状態グラフ系列 $d^{(j)} = \langle g^{(j,1)} \cdots g^{(j,m_j)} \rangle$ を変換規則を用いて $s^{(j)}_d = \langle tr^{(j,1)}_{[o,l]} \cdots tr^{(j,m_j-1)}_{[o,l]} \rangle$ と表し，内部状態グラフ変換系列と呼ぶ．また，外部状態グラフ系列 $d = \langle g^{(1)} \cdots g^{(n)} \rangle$ を $s^{(j)}_d$ を用いて $s_d = \langle s^{(1)}_d \cdots s^{(n-1)}_d \rangle$ と表し，外部状態グラフ変換系列と呼ぶ． ■

連続する 2 つの外部状態グラフ間でごく一部の構造のみが変化するという仮定の下で，変換系列には 2 つのグラフ間の差異のみが変換規則を用いて表されているので，グラフ表現を直接用いた系列の表現よりも簡潔である．

例 1. 図 2(a) に示すグラフ系列は図 2(b) に示すように変換規則を用いて表すことができ，その変換系列は $\langle vi^{(1,1)}_{[2,A]}\ vi^{(2,1)}_{[3,C]}\ ei^{(2,2)}_{[(2,3),-]}\ vi^{(3,1)}_{[1,D]}\ ei^{(3,2)}_{[(1,2),-]}\ vr^{(3,3)}_{[3,B]}\ vr^{(4,1)}_{[1,B]}\ vr^{(5,1)}_{[1,A]}\ vr^{(6,1)}_{[1,B]} \rangle$ となる．

変換系列 $s_d = \langle s^{(1)}_d \cdots s^{(n-1)}_d \rangle$ について，長さ $n-1$ を $|s_d|$ と表し，大きさ $||s_d||$ を s_d に存在する変換規則の数と定義する．例 1 の変換規則の長さと大きさはそれぞれ 6 と 9 である．

3. 単一グラフ系列からの FTS の探索

本節では本稿の問題を定義し，単一グラフ系列マイニング手法 SiGTRACE を提案する．ここで，単一グラフ系列から FTS を列挙するため，変換系列パターンの出現を定義する．

定義 5. 入力グラフ系列 d の変換系列 $s_d = \langle s^{(1)}_d \cdots s^{(n-1)}_d \rangle$ と，あるパターン p の変換系列 $s_p = \langle s^{(1)}_p \cdots s^{(n')}_p \rangle$ が与えられ，次の条件 $(a), (b), (c)$ を満たす単射関数対 (ϕ, ψ) が存在するとき，s_p は s_d に出現するという．

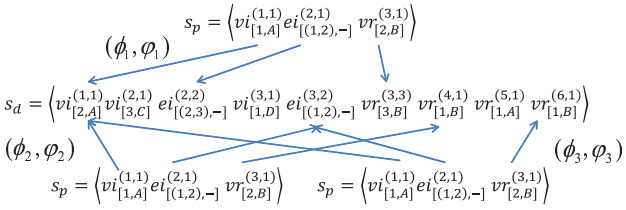

図 3: s_d に存在する s_p の出現

(a) $\forall\ tr^{(j,k)}_{[o,l]} \in s_p \Rightarrow \exists k', tr^{(\phi(j),k')}_{[o',l]} \in s_d$

(b) ϕ は以下を満たし，かつ整数を返す関数．
$1 \le \phi(1) < \phi(2) < \cdots < \phi(n') \le n-1$

(c) ψ は以下を満たす関数．
$\psi(u) \in ID_V(d)$

ここで，もし $tr^{(j,k)}_{[o,l]} \in s_p$ が頂点 ID u の頂点に関する変換規則ならば，$o' = \psi(u)$ であり，頂点 ID u_1 と u_2 の辺に関する変換規則ならば，$o' = (\psi(u_1), \psi(u_2))$ である．s_p が s_d に出現するとき，$s_p \sqsubseteq s_d$ と表す． ■

定義 5 の条件 (a) は s_p におけるすべての変換規則に対応する変換規則が s_d に必ず存在することを述べており，条件 (b) は内部状態グラフ変換系列の順序が維持されることを述べている．また条件 (c) は s_p におけるすべての頂点に対応する頂点が s_d に必ず存在することを述べている．s_d における s_p の出現を発見する計算量は部分グラフ同型問題と同一である．

例 2. 図 3 に示される変換系列 $s_p = \langle vi^{(1,1)}_{[1,A]} ei^{(2,1)}_{[(1,2),-]} vr^{(3,1)}_{[2,B]} \rangle$ は s_d に 3 回出現する．これらの出現は図 3 に示すようにそれぞれ (ϕ_1, ψ_1), (ϕ_2, ψ_2), (ϕ_3, ψ_3) で与えられる．

頻出パターンを効率よく列挙するために，逆単調性を満たす支持度を定義することが重要である．しかし，s_d に存在する s_p の出現数を s_p の支持度と定義すると支持度は逆単調性を満たさない．例えば，図 3 の s_p は s_d に 3 回出現しているが，s_p の部分系列 $s'_p = \langle vi^{(1,1)}_{[1,A]} ei^{(2,1)}_{[(1,2),-]} \rangle$ は 2 回しか出現していない．そこで，本稿では s_p の変換規則に対応する s_d の変換規則の最小数を用いた支持度を提案する．

定義 6. s_d における s_p のすべての出現 $\{ (\phi_1, \psi_1), \cdots, (\phi_n, \psi_n) \}$ が与えられたとき，s_d における s_p の支持度を以下の式で定義する．

$$\sigma(s_p) = \min_{tr^{(j,k)}_{[o,l]} \in s_p} \left| \bigcup_{i=1,\ldots,n} \{ tr^{(\phi_i(j),k')}_{[o'_i,l]} \in s_d \} \right| \quad ■$$

最小支持度 σ' 以上の変換部分系列を頻出変換部分系列 (FTS:Frequent Transformation Subsequence) と呼ぶ．定義 6 により，支持度の逆単調性は次のように保証される．

補題 1. $s'_p \sqsubset s_p \Rightarrow \sigma(s'_p) \ge \sigma(s_p)$ ■

定義 6 は単一グラフから頻出部分グラフを列挙する手法 [Kuramochi 2004, Fiedler 2007] の支持度の定義 [Bringmann 2008] を拡張したものである．そのため，s_d に存在する s_p のすべての出現の集合 M が既知であれば s_p の支持度は $O(|M|)$ で計算が可能である．

例 3. 図 3 では s_p の出現 (ϕ_1, ψ_1), (ϕ_2, ψ_2), (ϕ_3, ψ_3) が存在する．それぞれの出現において表 2 の 2 列目に示すように s_p の 1 番目の変換規則 $vi^{(1,1)}_{[1,A]}$ は s_d における $vi^{(1,1)}_{[2,A]}$ に対応する．同様に，s_p の他の変換規則の対応を表 2 にまとめる．定義 6 より，s_d における s_p の支持度は以下のようになる．

表 2: s_d に存在する s_p の支持度の計算

	$tr^{(\phi_1(j),k')}_{[o'_1,l]} \in s_d$	$tr^{(\phi_2(j),k')}_{[o'_2,l]} \in s_d$	$tr^{(\phi_3(j),k')}_{[o'_3,l]} \in s_d$	$\lvert\bigcup_{i=1,2,3}\{tr^{(\phi_i(j),k')}_{[o'_i,l]}\}\rvert$
$vi^{(1,1)}_{[1,A]} \in s_p$	$vi^{(1,1)}_{[2,A]}$	$vi^{(1,1)}_{[2,A]}$	$vi^{(1,1)}_{[2,A]}$	$\lvert\{vi^{(1,1)}_{[2,A]}\}\rvert = 1$
$ei^{(2,1)}_{[(1,2),-]} \in s_p$	$ei^{(2,2)}_{[(2,3),-]}$	$ei^{(3,2)}_{[(1,2),-]}$	$ei^{(3,2)}_{[(1,2),-]}$	$\lvert\{ei^{(2,2)}_{[(2,3),-]}, ei^{(3,2)}_{[(1,2),-]}\}\rvert = 2$
$vr^{(3,1)}_{[2,B]} \in s_p$	$vr^{(3,3)}_{[3,B]}$	$vr^{(4,1)}_{[1,B]}$	$vr^{(6,1)}_{[1,B]}$	$\lvert\{vr^{(3,3)}_{[3,B]}, vr^{(4,1)}_{[1,B]}, vr^{(6,1)}_{[1,B]}\}\rvert = 3$

Input: FTS s_p, グラフ系列 d の変換系列 s_d,
最小支持度 σ', ウィンドウ幅 w.
Output: FTS の集合 S.

SiGTRACE(s_p, s_d, σ', w, S)

1: insert s_p into S;
2: set C to $\emptyset$;
3: scan s_d once, and find all TRs $tr^{(j,k)}_{[o,l]}$ and
their occurrences, such that $s_p \Diamond tr^{(j,k)}_{[o,l]}$ occurs
in windows of width w of s_d
based on the Pattern Growth principle;
insert $s_p \Diamond tr^{(j,k)}_{[o,l]}$ in C
and compute its support using Definition 6;
4: **for each** frequent $s_p \Diamond tr^{(j,k)}_{[o,l]} \in C$ **do**
5: **Call SiGTRACE**($s_p \Diamond tr^{(j,k)}_{[o,l]}$, s_d, σ', w, S);
6: **return;**

図 4: FTS の列挙アルゴリズム

$$\sigma(s_p) = \min\{|\{vi^{(1,1)}_{[2,A]}\}|, |\{ei^{(2,2)}_{[(2,3),-]}, ei^{(3,2)}_{[(1,2),-]}\}|, |\{vr^{(3,3)}_{[3,B]}, vr^{(4,1)}_{[1,B]}, vr^{(6,1)}_{[1,B]}\}|\} = 1$$

さて，定義 5 で変換部分系列の出現について定義したが，実用上の理由から互いに近い変換規則からなる変換部分系列 s'_d に興味が有ることが多い．文献 [Katoh 2010] ではウィンドウを用いてアイテムの出現を制限しており，本稿においても出現をウィンドウ幅 w 内で出現するものに制限する．

定義 7. グラフ系列 d の変換系列 $s_d = \langle s_d^{(1)} \cdots s_d^{(n-1)} \rangle$ が与えられたとき，s_d のウィンドウとは s_d の連続な部分系列 $s_w = \langle s_d^{(i)} \cdots s_d^{(i+w-1)} \rangle$ $(i = 1, \cdots, n-w+1)$ である．ここで，$w \geq 1$ はウィンドウ幅である．s_d における s_p の出現 (ϕ, ψ) が与えられたとき，$\phi(|s_p|) - \phi(1) < w$ であれば，(ϕ, ψ) は幅 w のウィンドウに存在する． ■

出現をウィンドウ幅 w で制限しても支持度の逆単調性は保持される．また，s_d からすべての FTS を列挙する計算時間は s_d の大きさと長さの増加に従って，指数的に上昇する [Inokuchi 2008] ため，出現をウィンドウ幅 w 内のものに制限することでグラフ系列から効率よく FTS を列挙できる．

例 4. ウィンドウ幅 $w = 4$ のとき，図 3 に示す 3 つの出現のうち (ϕ_3, ψ_3) は $\phi_3(|s_p|) - \phi_3(1) = \phi_3(3) - \phi_3(1) = 6 - 1 = 5 \not< w = 4$ となるため，s_d 中の出現と認めない．

以上を用いて本稿の問題設定を以下に示す．

問題 1. グラフ系列 d，ウィンドウ幅 w，最小支持度 σ' が入力として与えられたとき，d の変換系列 s_d における w 内に出現する頻出変換部分系列 (FTS) をすべて列挙する．

図 4 は単一グラフ系列 d から深さ優先探索ですべての FTS を探索し，その内で最小支持度 σ' 以上のものを S に蓄積する提案手法 SiGTRACE の擬似コードである．3 行目で SiGTRACE は PrefixSpan [Pei 2001] と同様に，Pattern Growth principle に従い FTS の末尾に変換規則を再帰的に加えることですべての FTS を列挙する．ここで，$s_p \Diamond tr^{(j,k)}_{[o,l]}$ は s_p の末尾に変換規則 $tr^{(j,k)}_{[o,l]}$ が追加された変換系列を示す．

実際には SiGTRACE は FTS の集合の部分集合である “関連のある FTS” を列挙する．FTS の関連性は次のように定義される．

表 3: 実験時のパラメータ

データ生成時のパラメータ	初期値
変換規則が付加される確率	$p_i = 80\%$
付加される変換規則が頂点に関する変換規則である確率	$p_v = 80\%$
データ頂点 ID 数	$\lvert V\rvert = 15$
パターン平均頂点 ID 数	$\lvert V'_{avg}\rvert = 8$
頂点ラベル数	$\lvert L_v\rvert = 5$
辺ラベル数	$\lvert L_e\rvert = 5$
パターンの種類	$N = 3$
辺存在確率	p_e =15%
外部状態グラフでの平均編集距離	d_{ist} =2
パターンの長さ	$l_{ength} = 5$
各パターンを埋め込む数	$t = 10$
SiGTRACE 実行時のパラメータ	初期値
最小支持度	$\sigma' = 10$
ウィンドウ幅	$w = 5$

定義 8. 変換系列 s の和グラフが連結グラフであれば，s は関連がある，という．ここで，s の和グラフを $g_u(s) = (V, E)$ と定義する．

$$V = \{u \mid tr^{(j,k)}_{[u,l]} \in s, tr \in \{vi, vd, vr\}\} \cup \{u_1, u_2 \mid tr^{(j,k)}_{[(u_1,u_2),l]} \in s, tr \in \{ei, ed, er\}\}$$

$$E = \{(u_1, u_2) \mid tr^{(j,k)}_{[(u_1,u_2),l]} \in s, tr \in \{ei, ed, er\}\}$$ ■

GTRACE の支持度の定義を文献 [Bringmann 2008] を用いた支持度の定義に置き換えることで，グラフ系列の集合ではなく単一グラフ系列からすべての rFTS をマイニングできる．また，このアルゴリズムを “SiGTRACE” と呼ぶ．

4. 実験と議論

提案手法を Java を用いて実装した．実験は Intel Xeon CPU W3565 3.20GHz のプロセッサ，12.0GB のメインメモリ，Windows 7 Enterprise 64bit の OS を搭載した計算機で行った．提案手法の性能を人工グラフ系列データを用いて評価した．人工データ生成時のパラメータは表 3 に示される通りである．

まず，N 個のグラフ系列 $d'_1, \cdots, d'_N$ を以下の通りに生成する．はじめに，平均 $|V'_{avg}|/2$ の頂点を持ち，2 頂点間に辺の存在する確率が p_e のグラフ $g^{(1)}$ を生成する．次に d'_i の変換系列の末尾に d_{ist} 個の変換規則を追加することで頂点 ID 数が $|V'_{avg}|$ 個のグラフ系列を生成する．追加される変換規則は確率 $p_v \times p_i$ で頂点の追加，確率 $p_v \times \frac{1-p_i}{2}$ で頂点の削除，確率 $(1-p_v) \times p_i$ で辺の追加，というように選ばれる．また，変換規則の頂点ラベル，辺ラベルはそれぞれ $|L_v|$ 個の頂点ラベル，$|L_e|$ 個の辺ラベルからランダムに選ばれる．以上の過程は d'_i の変換系列が連結かつ長さが l_{ength} となるまで続けられる．従って，p_i が小さい，あるいは $|V'_{avg}|$ が大きいとき，生成された変換系列は長くなる．同様に，頂点 ID 数が $|V|$ 個のグラフ系列 d を生成する．最後に，d に d'_i ($i = 1, \cdots, N$) を t 回ずつ互いに重ならないように上書きすることで入力データとなるグラフ系列を生成する．表 3 に示される初期値で生成されたグラフ系列 d の変換系列 s_d の長さは 210 であった．さらに，グラフ系列 d'_i をそれぞれ 10 回上書きし，d'_i の変換系列の長さを 5 としたため，SiGTRACE の最小支持度とウィンドウ幅の初期値をそれぞれ 10 と 5 とした．

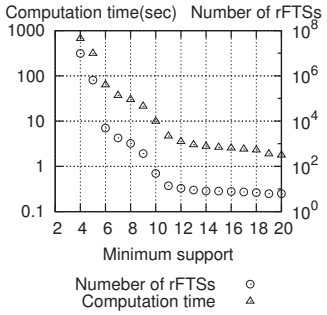

図 5: Result for various σ'

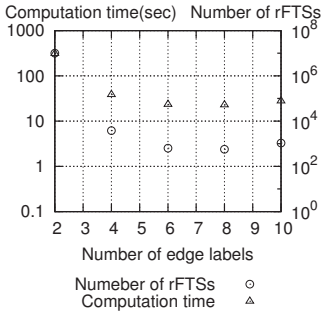

図 6: Result for various $|L_e|$

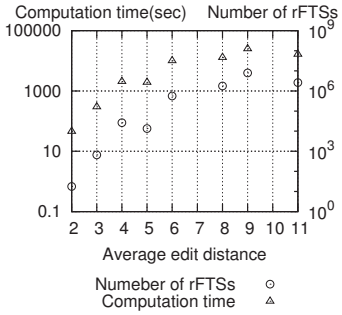

図 7: Result for various d_{ist}

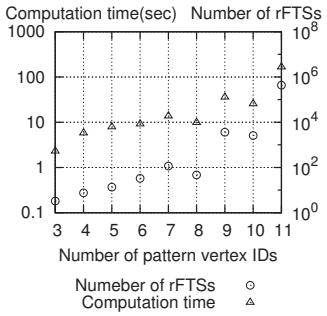

図 8: Result for various $|V'_{avg}|$

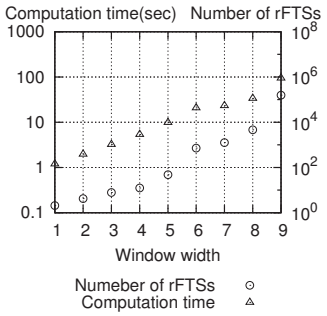

図 9: Result for various w

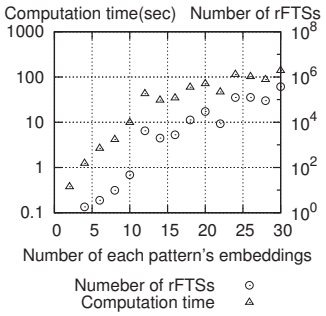

図 10: Result for various t

図 5 から図 10 はそれぞれ σ', $|L_e|$, d_{ist}, $|V'_{avg}|$, w, t を初期値から個別に変化させて実験したときの SiGTRACE の計算時間とマイニングされた rFTS の数を示している．すべてのデータ集合の場合について，SiGTRACE がデータを生成する過程で上書きしたすべての rFTS を列挙したことを確認した．図 5 は最小支持度 σ' の減少に伴って計算時間が指数的に増加することを示している．これは σ' が減少すると s_d に含まれている rFTS の数が指数的に増加するためである．図 6 は辺ラベル数の増加に伴って計算時間，rFTS の数が減少することを示している．これは辺ラベル数が増加すると s_d に出現する変換部分系列の多様性が増加し，各変換部分系列の支持度が減少するためである．図 7 は平均編集距離 d_{ist} の増加に伴って計算時間，rFTS の数が増加することを示している．d_{ist} が増加すると s_d に出現する変換部分系列の多様性は増加するが，各変換部分系列の支持度は減少しない．これは N 個のグラフ系列 d'_i の変換系列が s_d において t 回出現しているためである．図 8 はパターン頂点 ID 数 $|V'_{avg}|$ を変化させたときの計算時間と rFTS の数を示している．$|V'_{avg}|$ が増加すると d'_i の変換部分系列の数が増加するため，計算時間，rFTS の数が増加する．図 9 はウィンドウ幅 w の減少に伴って計算時間，rFTS の数が減少することを示している．この結果から出現をウィンドウ幅 w に制限することで互いに近い変換規則から構成される rFTS を効率的に列挙できることがわかる．図 10 は d に d'_i を上書きする回数 t の増加に伴って計算時間，rFTS の数が増加することを示している．これは t が増加することで，それぞれの変換部分系列の支持度が増加するためである．以上に示した計算時間と列挙された頻出パターンの数に関する傾向は従来の頻出パターンマイニング手法と同様である [Han 2006].

5. まとめ

本稿では単一グラフ系列からすべての rFTS を列挙する手法として SiGTRACE を提案した．また性能評価実験により提案手法が単一グラフ系列からの頻出パターン列挙に対して有効であることを示した．近年，グラフ系列の集合から FRISS(Frequent, Relevant, and Induced Subgraph Subsequences) と呼ばれる頻出パターンの列挙手法が提案されている [Inokuchi 2010]．本稿で提案した原理はその手法にも適用できる．今後，単一グラフ系列から FRISS を列挙する手法の開発する予定である．

参考文献

[Bringmann 2008] Bringmann, B., and Nijssen, S.: What is Frequent in a Single Graph. *Proc. of PAKDD 2008*, pp. 858–863, (2008)

[Fiedler 2007] Fiedler, M., and Borgelt, C.: Support Computation for Mining Frequent Subgraphs in a Single Graph. *Proc. of MLG 2007*, (2007)

[Han 2006] Han, J., et. al.: Data Mining: Concepts and Techniques *Morgan Kaufmann*, (2006)

[Inokuchi 2008] Inokuchi, A., and Washio, T.: A Fast Method to Mine Frequent Subsequences from Graph Sequence Data. *Proc. of ICDM 2008*, pp. 303–312. (2008)

[Inokuchi 2010] Inokuchi, A., and Washio, T.: Mining Frequent Graph Sequence Patterns Induced by Vertices. *Proc. of SDM 2010*, pp. 466–477 (2010)

[Katoh 2010] Katoh, T., et. al.: Mining Frequent k-Partite Episodes from Event Sequences, *Joint JSAI-isAI Workshops, Lecture Notes in Artificial Intelligence 6284*, pp. 331–344, (2010)

[Kuramochi 2004] Kuramochi, M., and Karypis, G.: Finding Frequent Patterns in a Large Sparse Graph. *Proc. of SDM 2004*, (2004)

[Pei 2001] Pei, J., et. al.: PrefixSpan: Mining Sequential Patterns by Prefix-Projected Growth, *Proc. of ICDE 2001*, pp. 2–6. (2001)