Technical Report of the 2nd Workshop on Latent Dynamics
(Jun 22, 2011, Tokyo, Japan)

グラフ系列マイニング

猪口 明博*
Akihiro Inokuchi

Abstract: 人間関係ネットワークは人が頂点，関係が辺であるグラフで表現でき，人がネットワークに参加，脱退することで頂点や辺が増減する．すなわち人間関係ネットワークの構造変化はグラフの系列で表される．同様に状態遷移系に基づく掛かり受け解析器（Shift-Reduce Parser）内の状態は文節が頂点，係り受けが辺であるグラフで表現でき，遷移系列はグラフ系列で表される．このようにグラフ系列は構造とその構造変化を扱うのに適したデータ構造である．本講演ではグラフ系列マイニング問題とグラフ系列から頻出パターンを列挙する手法を紹介する．

1 はじめに

膨大なデータから有用な，あるいは興味のあるパターンを知識として発掘するデータマイニングの研究が盛んに行われている．有用性は人それぞれ異なるので定義するのは難しいが，一般に多くの事例を説明できる知識は有用と考えられる [17]．複数のアイテム集合のデータから頻出アイテム集合を列挙する Apriori アルゴリズム [1] が提案されて以来，様々なデータ構造に対して頻出パターン列挙手法が提案されている．近年では，頂点間連結関係と頂点や辺ラベルの情報からなるグラフ構造に頻出する部分グラフパターン [22, 11, 6] をマイニングする手法が提案されている．提案されているグラフマイニング手法は実用上，非常に効率的であるが，部分グラフ同型問題が NP 完全であるため，より大きな部分グラフをマイニングするのに多くの計算時間を要する．従って，既存手法をグラフ系列のような複数グラフからなる大きなグラフに対して適用することは困難である．

しかしながら，グラフの系列によるモデル化が適している実世界の対象は多く存在する．図 1(a) は 4 状態，5 頂点 ID からなるグラフ系列を示している． 例えば，人間関係ネットワークは人が頂点，関係が辺であるグラフで表現でき，人がコミュニティ（ネットワーク）に参加，脱退することで頂点や辺が増減する．同様に，遺伝子が頂点，相互関係が辺である遺伝子ネットワークは，進化の過程で遺伝子が新規獲得されたり，欠落，突然変異するグラフの系列で表現できる．

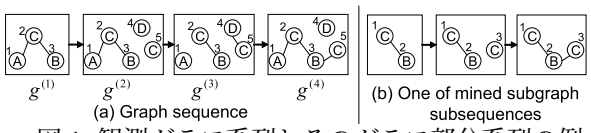

図 1: 観測グラフ系列とそのグラフ部分系列の例

このようなデータ解析上のニーズを背景として，我々は，グラフ系列をマイニングする手法 GTRACE[13], FRISSMiner[14] を提案した．本講演ではグラフ系列マイニング問題とグラフ系列から頻出パターンを列挙する手法を紹介する．

2 GTRACE

GTRACE は，図 1(a) に示すグラフ系列の集合から，それらに頻出する図 1(b) のような系列を列挙する手法である．GTRACE が対象とするグラフ系列は，以下を満たすグラフの系列である．

- 系列中でグラフの頂点数や辺数が増減する．
- 系列中で頂点ラベルや辺ラベルが変わる．
- 観測グラフ系列の中の連続する 2 つのグラフ $g^{(j)}$ と $g^{(j+1)}$ 間でその構造のごく一部のみが変化する．
- 各グラフは疎グラフである．

例えば，一度に大半の人間や遺伝子が入れ替わることはなく，更に各時点では個々の人間や遺伝子は他の一部としか関係を持たない人間関係ネットワークや遺伝子ネットワークのように，実世界の多くのグラフ変化は，これらの仮定を満たしている．

2.1 グラフ系列の表現形式

グラフ系列中で連続する 2 つのグラフのごく一部が変化するという仮定より，各グラフ $g^{(j)}$ をその全頂点，及びその間の辺で直接表す方法は冗長である．部分系列を

*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, e-mail inokuchi@ar.sanken.osaka-u.ac.jp,
The Institute of Scientific and Industrial Research, Osaka University, 8-1 Mihogaoka, Ibaraki, Osaka, 567-0047

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，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 $vd^{(j,k)}_{[u,\bullet]}$	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 $vr^{(j,k)}_{[u,l]}$	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 $ei^{(j,k)}_{[(u_1,u_2),l]}$	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 $ed^{(j,k)}_{[(u_1,u_2),\bullet]}$	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
辺ラベルの変更 $er^{(j,k)}_{[(u_1,u_2),l]}$	ID が u_1 と u_2 である頂点間の辺ラベルを l に変更し，$g^{(j,k)}$ を $g^{(j,k+1)}$ へ変換

効率よく探索するためには，計算コストと空間コストを抑えるためのグラフ系列の簡潔な表現が必要となる．そこで本節では，GTRACE が用いるグラフ系列の表現形式を説明する．

ラベル付きグラフ g を $g=(V,E,L,f)$ で表す．ここで，$V=\{v_1,\cdots,v_z\}$ は頂点集合，$E\subseteq\{(v,v')\mid(v,v')\in V\times V\}$ は辺集合，L は頂点と辺のラベル集合であり，$f:V\cup E\rightarrow L$ である．グラフ g の頂点集合，辺集合，ラベル集合を $V(g), E(g), L(g)$ と表す．また観測グラフ系列を $d=\langle g^{(1)}\cdots g^{(n)}\rangle$ と表す．$g^{(j)}$ は j 番目に観測されたグラフである．また，グラフ系列の ID の集合を $ID(d)=\{id(v)\mid v\in V(g^{(j)}), g^{(j)}\in d\}$ と定義する．

グラフ系列を簡潔に表現するため，グラフ系列中の連続する 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 $g^{(j)}$ を外部状態と呼ぶ．さらに，連続する *2* グラフ $g^{(j)}$ と $g^{(j+1)}$ の間を補間するグラフ系列を $d^{(j)}=\langle g^{(j,1)}\cdots g^{(j,m_j)}\rangle$ で表し，各 $g^{(j,k)}$ を内部状態と呼ぶ．ただし，$g^{(j,1)}=g^{(j)}$ かつ $g^{(j,m_j)}=g^{(j+1)}$ とする．グラフ系列 d は補間系列 $d=\langle d^{(1)}\cdots d^{(n-1)}\rangle$ で表される． ■

外部状態の順序は観測グラフ系列中のグラフの順序であるが，内部状態の順序は人工的に補間されたグラフの順序であり，$g^{(j)}$ と $g^{(j+1)}$ の間に様々な補間系列が考えられる．GTRACE は，グラフ系列マイニングの計算コストと空間コストを抑えるために，グラフ編集距離 [21] に基づき最短の補間系列を選択する．

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

本稿では，最小単位の変換を変換規則を用いて表す．

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．

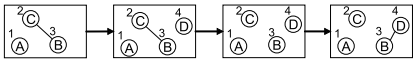

図 2: 関連性のない頂点を含む外部状態系列

- tr は頂点や辺の追加，削除，ラベル変更のいずれか．
- o_{jk} は変換される頂点 *ID*，あるいは辺の頂点 *ID* 対．
- l_{jk} は変換される頂点や辺のラベル． ■

本稿では簡単化のため変換規則 $tr^{(j,k)}_{[o_{jk},l_{jk}]}$ を $tr^{(j,k)}_{[o,l]}$ と略記する．GTRACE が用いる 6 種の変換規則を表 1 に示す．

以上より，変換系列を以下のように定義する．

定義 4 内部状態系列 $d^{(j)}=\langle g^{(j,1)}g^{(j,2)}\cdots g^{(j,m_j)}\rangle$ を変換規則を用いて $s^{(j)}_d=\langle tr^{(j,1)}_{[o,l]}tr^{(j,2)}_{[o,l]}\cdots tr^{(j,m_j-1)}_{[o,l]}\rangle$ と表し，内部状態変換系列と呼ぶ．さらに，外部状態系列 $d=\langle g^{(1)}\cdots g^{(n)}\rangle$ を内部状態変換系列の系列である外部状態変換系列 $s_d=\langle s^{(1)}_d s^{(2)}_d\cdots s^{(n-1)}_d\rangle$ で表す． ■

変換系列によるグラフ系列の表記は，グラフが徐々に変化するという仮定の下で，連続するグラフの差異のみに注目した表現形式であるので，グラフによる直接の系列表記に比べ簡潔である．また，如何なるグラフ系列も表 1 に示す 6 種の変換規則で表現可能である．

2.2 頻出変換部分系列のマイニング

本節ではグラフ系列の集合から頻出変換部分系列をマイニングする手法を示す．2.1 節で説明した外部状態の系列から頻出変換部分系列を列挙するために，変換系列 s'_d が変換系列 s_d の部分系列であるとき，$s'_d\sqsubseteq s_d$ と書く．詳細な定義については，文献 [13] を参照されたい．

GTRACE は，実用性の観点から出力される系列中の頂点と辺が互いに関連がある (relevant) 系列のみを列挙する．例えば，図 2 のグラフ系列では，ラベルが A で ID が 1 である頂点は，どの外部状態においても他の頂点と連結していないため，他の頂点と関連がないと考える．一方，頂点 2 と頂点 4 はどの外部状態においても直接は接続していないが，それらの頂点はラベル B をもつ頂点 3 と，1 番目の外部状態と 4 番目の外部状態でそれぞれ連結している．この場合，本稿では頂点 2 と 4 は頂点 3 を介して互いに関連があると考える．このように，図 2 における関連性のある系列の例として，頂点 2, 3, 4 を含み，頂点 1 を含まないものが考えられる．以上の外部状態系列の連結性の議論に基づいて，頂点と辺の ID の関連性を以下に定義する．

定義 5 外部状態系列 $d=\langle g^{(1)}\cdots g^{(n)}\rangle$ に対し，ラベルを持たない d の和グラフ $g_u(d)=(V_u,E_u)$ を以下のように定義する．

$V_u=\{id(v)\mid v\in V(g^{(j)}), g^{(j)}\in d\}$
$E_u=\{(id(v),id(v'))\mid(v,v')\in E(g^{(j)}), g^{(j)}\in d\}$ ■

1) **GTRACE**(DB, σ')
2) $G_u = \{g_u(d) \mid \langle tid, d\rangle \in DB\}$
3) for g =AcGM(G_u, σ'); untill $g \neq null${
4) $DB' = \bigcup_{\langle tid,d\rangle \in DB} proj(\langle tid, d\rangle, g)$
5) F' = SeqPatternMiner(DB', σ')
6) $F = F \cup \{\alpha \mid \alpha \in F' \wedge g_u(\alpha) = g\}$
7) }

図 3: GTRACE の概略

和グラフは変換系列に対しても同様に定義される．外部状態系列 d，あるいは変換系列 s_d の和グラフが連結であるとき，d，あるいは s_d の ID は互いに関連があると定義する．GTRACE は和グラフが連結である変換系列のみを列挙する．グラフ系列の集合 $DB = \{\langle tid, d\rangle \mid d = \langle g^{(1)} \cdots g^{(n)}\rangle\}$ に対し，変換部分系列 s_p の支持度 $\sigma(s_p)$ を $\sigma(s_p) = |\{tid \mid \langle tid, d\rangle \in DB, s_p \sqsubseteq s_d\}|$ と定義する．ここで，s_d は d の変換系列である．最小支持度 σ' 以上の支持度を有する部分系列を頻出変換部分系列 (Frequent Transformation Subsequence: FTS) と呼ぶ．関連研究同様，$s_{p1} \sqsubset s_{p2}$ ならば $\sigma(s_{p1}) \geq \sigma(s_{p2})$ である支持度の逆単調性が成り立つ．以上の定義により，グラフ系列マイニングを以下のように定義する．

問題 1 グラフ系列の集合 $DB = \{\langle tid, d\rangle \mid d = \langle g^{(1)} \cdots g^{(n)}\rangle\}$ と最小支持度 σ' が入力として与えられたとき，DB 中の *rFTS (relevant FTS)* を全て列挙する．

図 3 は DB から rFTS を全て列挙するアルゴリズムを示している．はじめに 2 行目で外部状態系列の集合 DB の和グラフ集合 G_u を計算する．3 行目の AcGM[12] は G_u から頻出連結部分グラフ g を 1 つずつ出力する関数であり，4 行目において g を用いて射影データ DB' を生成する．ここで得られる射影データは，和グラフが g と同型な変換系列の集合である．続いて，射影データ DB' に含まれる頻出変換部分系列を SeqPatternMiner で列挙する．SeqPatternMiner では，PrefixSpan[20] と同様に，得られた FTS の末尾に変換規則を 1 つずつ付加しながら FTS を探索し，最小支持度を下回ったら，バックトラックする．最後に，列挙された FTS の和グラフが g と同型ならば，それを rFTS として出力する．この処理は AcGM が g を出力する限り続けられる．

定義 6 グラフ系列 $\langle tid, d\rangle \in DB$ と連結グラフ g が与えられたとき，$\langle tid, d\rangle$ に対する射影 $proj$ を以下のように定義する．

$$proj(\langle tid, d\rangle, g) = \{\langle tid, s'_d\rangle \mid s'_d \sqsubseteq s_d,\ g_u(s'_d) = g,\ \nexists s''_d\ s.t.\ (s'_d \sqsubset s''_d \sqsubseteq s_d \wedge g_u(s''_d) = g)\} \quad \blacksquare$$

この射影により，1 つのグラフ系列 $\langle tid, d\rangle$ から複数の変換規則が出力されることに注意されたい．rFTS の和グラフは和グラフ集合 G_u において頻出連結部分グラフとなるので，もし和グラフの集合 G_u から連結な頻出部分グラフ g が得られれば，定義 6 により生成された射影系列から，和グラフが g である rFTS を全て列挙することができる．

2008 年に我々が提案した GTRACE は探索の過程において，関連のない FTS も列挙するため，改善の余地があった．そこで，rFTS のみを探索する GTRACE-RS[10] を提案した．GTRACE-RS は逆探索 [2, 3] に基づいた手法であり，従来の GTRACE に比べ，100 倍以上高速に全 rFTS を列挙できる．

3 FRISSMiner

変換規則を用いたグラフ系れ列の表現は，グラフが徐々に変化するという仮定のもとで，グラフ系列を簡潔に表現することが可能である．しかしながら，グラフ系列を観測する際（データを収集する際）に，時間分解能が低い場合，観測されたグラフ系列の連続する 2 つのグラフの間で，グラフの大部分が変化する可能性があるため，変換規則系列の長さは大きくなる．変換規則系列長が大きくなると，支持度の逆単調性より頻出パターンの部分パターンは頻出であるため，頻出パターンの集合が非常に大きくなり，GTRACE を適用することが困難になる．本節では，グラフ系列中の連続する 2 つのグラフの変化が小さくないといういう仮定のもとで，頻出パターンを効率良く列挙する手法 FRISSMiner を紹介する．

3.1 誘導部分グラフ系列

はじめに，2 つのグラフ系列 α と β の包含関係を以下のように定義する．

定義 7 グラフ系列 $\alpha = \langle a^{(1)} \cdots a^{(n)}\rangle$ と $\beta = \langle b^{(1)} \cdots b^{(m)}\rangle$ との間に，以下を満たす単射 $\phi : ID(\alpha) \to ID(\beta)$ と整数 $1 \leq j_1 < j_2 < \cdots < j_n \leq m$ が存在するとき, α を β の部分グラフ系列と呼び，$\alpha \sqsubseteq \beta$ と表わす.

- $a^{(1)} \sqsubseteq b^{(j_1)}, a^{(2)} \sqsubseteq b^{(j_2)}, \cdots, a^{(n)} \sqsubseteq b^{(j_n)}$,
- *for* $v \in V(a^{(i)})$ *and* $v' \in V(a^{(i')})$, *if* $id(v) = id(v')$, *then* $\exists(u \in V(b^{(j_i)})$ *and* $u' \in V(b^{(j_{i'})}))$ *s.t.* $\{id(u) = \phi(id(v)) \wedge id(u') = \phi(id(v'))\}$, $id(u) = id(u')$. $\blacksquare$

上記の定義において，1 つ目の条件は従来の系列マイニング [20] の部分系列の定義と同様である．2 つ目の条件は，グラフ系列 α の異なる状態 $a^{(i)}$ と $a^{(i')}$ の各々の頂点 v と v' が同じ ID をもつなら，ϕ によって写像された頂点 u と u' も同じ ID をもつことを意味している．

さらに理解可能な頻出パターン [14] をマイニングするために誘導部分グラフ系列を定義する．

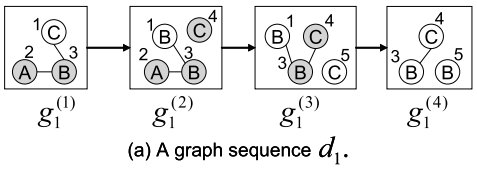

図 4: グラフ系列の包含関係

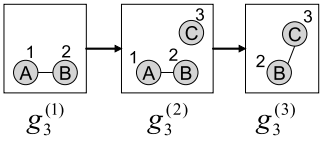

図 5: 図 4 のグラフ系列 d_1 の誘導部分グラフ系列 d_3

定義 8 グラフ系列 $\alpha = \langle a^{(1)} \cdots a^{(n)} \rangle$ をグラフ系列 $\beta = \langle b^{(1)} \cdots b^{(m)} \rangle$ の部分グラフ系列とする．ただし，$a^{(i)} \sqsubseteq b^{(j_i)}$ であり，α の *ID* u_α と u'_α は単射 ϕ によりそれぞれ β の *ID* u_β と u'_β に写像されるものとする．下記の *2* つの条件を満たすとき，α を β の誘導部分グラフ系列と呼び，この包含関係を $\alpha \sqsubseteq_i \beta$ と記す.

- *ID* が u_β である頂点が $b^{(j_i)}$ に存在するときに限り，*ID* が u_α である頂点が $a^{(i)}$ に存在する.
- 両端の *ID* が u_β と u'_β である頂点間の辺が $b^{(j_i)}$ に存在するときに限り，両端の *ID* が u_α と u'_α である頂点間の辺が $a^{(i)}$ に存在する． ■

グラフ g の誘導部分グラフは g の頂点とそれに接続する辺を除くことで生成できる．すなわち，誘導部分グラフは頂点集合により決められる [7]．同様に，グラフ系列 β の誘導部分グラフ系列は，ある ID を持つ頂点，それに接続する辺を削除することで生成することができる．

図 5 のグラフ系列 d_3 は図 4(a) のグラフ系列 d_1 の誘導部分グラフ系列である．一方，d_2 の 2 番目の状態で，ID が 1 と 2 である頂点間に辺がないので，図 4 (b) のグラフ系列 d_2 は，d_1 の誘導部分グラフ系列ではない．

誘導部分グラフ系列の包含関係 ($\sqsubseteq_i$) に基づいて，グラフ系列 α の支持度の定義を $\sigma_i(\alpha) = |\{tid \mid (\langle tid, d \rangle \in DB) \wedge (\alpha \sqsubseteq_i d)\}|$ と定義する．この支持度についても，支持度の逆単調性が成り立つ．さらに，マイニング問題を以下のように定義する.

問題 2 データベース $DB = \{\langle tid, d \rangle \mid d = \langle g^{(1)} \cdots g^{(l)} \rangle\}$ と最小支持度 σ' が入力として与えられたとき，和グラフが連結である頻出パターンの集合 $F = \{f \mid \sigma_i(f) \geq \sigma'\}$ を列挙することである．列挙される頻出パターンを頻出関連誘導部分グラフ系列（*FRISS*：*Frequent, Relevant, and Induced Subgraph Subsequence*）パターンと呼ぶ.

3.2 FRISS 列挙アルゴリズム

既存のグラフマイニングを拡張することによる，FRISS をマイニングする素朴なマイニング手法は，列挙された FRISS に頂点（あるいは辺）を再帰的に 1 つずつ追加して，支持度を計算し，最小支持度を下回ったときバックトラックする方法である．FRISS の定義より，FIRSS に追加する頂点は FRISS の和グラフが連結であることを満たしながら加えていく必要がある．しかし，再帰の深さが増加したとき，必要なメモリ量は急激に増加する．例えば，この素朴な手法による図 5 に示す頂点数 9 の FRISS をマイニングするための再帰の深さは 9 になる．しかし，グラフ系列の関連性，及び誘導部分グラフ系列の定義を巧みに用いることで，FRISSMiner の再帰の深さは FRISS に含まれる ID の数と外部状態数の和になる．例えば，図 5 の FRISS を探索するための再帰の深さは $3 + 3 = 6$ となる.

FRISSMiner のアルゴリズムの概略は GTRACE と同じである．ただし，射影の定義と SeqPatternMiner の実装方法がことなる．射影の定義を以下に示す.

定義 9 $\langle sid, d \rangle \in DB$ と連結グラフ g に対して，射影 *"proj"* を以下のように定義する.

$$proj(\langle sid, d \rangle, g) = \{\langle sid, d' \rangle \mid (d' \sqsubseteq_i d) \wedge (\perp \notin d') \wedge (g_u(d') = g) \wedge (\nexists d'' \; s.t. \; d' \sqsubset_i d'' \sqsubseteq_i d)\} \blacksquare$$

上記の定義で $\perp \notin d'$ は d' が頂点なしの外部状態を含まないことを意味している．「$\nexists d''$ $s.t.$ $d' \sqsubset_i d'' \sqsubseteq_i d$」は「$d' \sqsubseteq_i d \wedge g_u(d') = g \wedge \perp \notin d'$」を満たすグラフ系列 d' のうち極大なもののみを出力することを意味している.

GTRACE の射影が変換系列の集合を返すのに対して，FRISSMiner の射影はグラフ系列の集合を返す．また，GTRACE の SeqPatternMiner が変換系列を 1 つずつ末尾に付加するのに対して，FRISSMMiner の SeqPatternMiner はグラフを 1 つずつ末尾に付加する.

GTRACE，FRISSMiner ともに，それらの計算時間は入力のグラフ系列の数に対して線形的に増加する．また，GTRACE の計算時間は変換規則系列の平均長に，FRISSMiner の計算時間はグラフ系列の平均長に対して，指数関数的に増加する．本稿では詳細な評価実験結果を割愛するため，個々の論文を参照されたい.

表 2: グラフ系列の違い

	graph sequence	dynamic graph	evolving graph
頂点数	増減する	一定	一定 †
辺数	増減する	増減する	増加する †
頂点ラベル	変化する	変化しない	変化しない
辺ラベル	変化する	ラベルなし	変化しない

†: evolving graph の頂点はそれに繋がる辺とともに追加される．

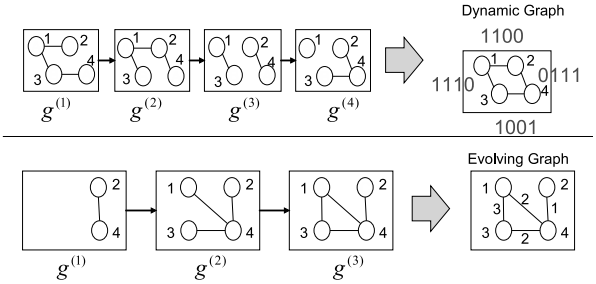

図 6: Dynamic graph と evolving graph
（簡単化のため，頂点ラベルを省略する）

4 議論

近年，グラフ系列 (dynamic graph [5] あるいは evolving graph [4]) から頻出パターンを列挙する問題が注目されはじめている [19, 18, 8, 23, 9, 16]．文献 [5] において，Borgwardt らは dynamic graph と呼ばれるグラフ系列から頻出パターンを列挙する手法を提案した．この文献では，グラフ系列中のグラフの辺数は増減するが，頂点数や頂点ラベルは変化せず，辺にはラベルがないものとしている．dynamic graph の特徴を表 2 の 3 列目目に要約する．また，図 6 の左上のグラフ系列は右上の dynamic graph で表わされる．この手法において，グラフ系列中の各状態の辺の存在と非存在は 1 と 0 によって表わされ，dynamic graph の各辺はこれらの 0 と 1 からなる系列によって表わされる．

一方，Berlingerio らは文献 [4] において，evolving graph と呼ばれるグラフ系列から頻出パターンを列挙する手法を提案した．この手法では，グラフの頂点や辺にはラベルがなく，頂点数や辺数は増加するが，減少はしないと仮定している．さらに，頂点に接続する辺は，必ずその頂点と同じ状態で追加されると仮定している．evolving graph の特徴を表 2 の 4 列目目に要約する．また，図 6 の左下のグラフ系列は右下の evolving graph で表わされる．evolving graph の各辺の数字はその辺が追加された状態の番号を表わしている．これらの 2 つの手法では，複雑なグラフ系列を 1 つのグラフで表現しているが，それらの手法では 1 節で述べた人間関係ネットワークや遺伝子

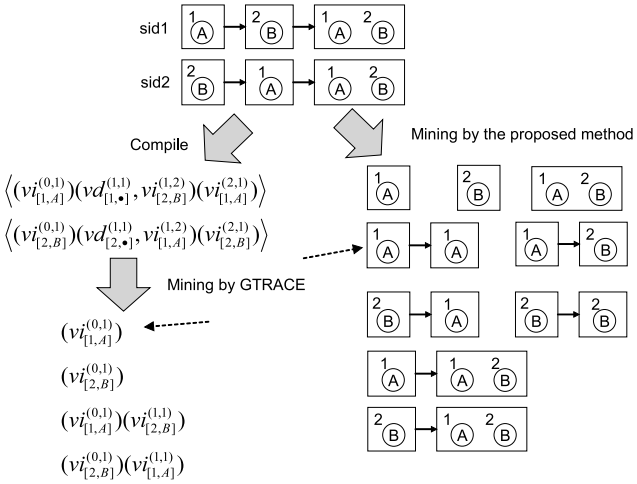

図 7: グラフ系列から FSS と FTS のマイニング

ネットワークのように頂点や辺数が増減し，ラベルが変化するグラフの系列を扱うことができない．従って，本稿で扱ったグラフ系列は dynamic graph や evolving graph よりも汎用なクラスの構造であると言える．

FRISSMiner や GTRACE はともに表 2 の 2 列目にまとめた汎用的なグラフ系列に適用することが可能である．図 7 の左下の 4 つの系列は上部の 2 つのグラフ系列から最小支持度 2 で列挙される全 FTS である．一方，右下の 9 つの系列は同じグラフ系列から最小支持度 2 で列挙される全 FSS (Frequent Subgraph Subsequence) である．FTS は入力である 2 つのグラフに共通して含まれる共通の変化であるのに対して，FSS は 2 つのグラフに含まれる共通の構造である．例えば，矢印で示されている 1 つ目の FTS はラベルが A である頂点が追加されたことを表わしてる．ただし，この FTS から，その頂点がいくつの外部状態で存在し続けたのかまでは分からない．一方， 矢印で示された 2 つ目の FSS はラベルが A である頂点が少なくとも 2 つの外部状態に存在したことをあらわしている．このように，FRISSMiner と GTRACE の入力は同じであるが，出力されるパターンの解釈は異なる．共通する変化のパターンを発見したいか，あるいは共通する構造を発見したいかの目的に応じて使い分ける必要がある．

5 おわりに

本稿ではグラフ系列マイニング問題を紹介し，我々が提案した GTRACE と FRISSMiner を紹介した．グラフ系列は構造の変化を表すための汎用的なデータ構造であり，グラフが徐々に変化するという仮定のもとで，変換系列はグラフ系列を簡潔に表現するデータ構造である．GTRACE や FRISSMiner はグラフ系列に現れる表層的

な関係である頻出パターンを列挙するアルゴリズムであり，構造変化の予測まではできない．構造変化を予測する問題などが，今後，重要な研究テーマであると考えられる．

参考文献

[1] R. Agrawal and R. Srikant. Fast Algorithms for Mining Association Rules in Large Databases. *proc. of Int'l Conf. on Very Large Data Bases*, 487-499, 1994.

[2] D. Avis and K. Fukuda. Reverse Search for Enumeration. *Discrete Applied Mathematics*, Vol. 65, pp. 21–46, 1996.

[3] T. Asai, et al. Efficient Tree Mining Using Reverse Search. Technical Report 218, Department of Informatics, Kyushu University, 2003.

[4] M. Berlingerio, et al. Mining Graph Evolution Rules. *Proc. of European Conf. on Principles and Practice of Knowledge Discovery in Databases*, pp. 115–130, 2009.

[5] K. M. Borgwardt, et al. Pattern Mining in Frequent Dynamic Subgraphs. *Proc. of Int'l Conf. on Data Mining*, pp. 818–822, 2006.

[6] D. J. Cook and L. B. Holder. Mining Graph Data. *Wiley-Interscience*, 2006.

[7] Chris Godsil and Gordon Royle. Algebraic Graph Theory *Springer*, 2001.

[8] 岸本，猪口，鷲尾．飽和系列パターンマイニングを用いたグラフ系列マイニングの高速化 人工知能学会第 24 回全国大会, 3A2-4, 2010.

[9] Y. Kabutoya, et al. Dynamic Network Motifs: Evolutionary Patterns of Substructures in Complex Networks. *Proc. of Asia-Pacific Web Conference*, pp. 321–326, 2011.

[10] 生田, 猪口, 鷲尾. 逆探索法によるグラフ系列マイニングの高速化第 3 回データ工学と情報マネジメントに関するフォーラム, B10-3, 2011.

[11] A. Inokuchi, et al. An Apriori-based Algorithm for Mining Frequent Substructures from Graph Data. *Proc. of European Conf. on Principles of Data Mining and Knowledge Discovery*, pp. 13–23, 2000.

[12] A. Inokuchi, et al. A Fast Algorithm for Mining Frequent Connected Subgraphs. *IBM Research Report*, RT0448, 2002.

[13] A. Inokuchi and T. Washio. A Fast Method to Mine Frequent Subsequences from Graph Sequence Data. *Proc. of Int'l Conf. on Data Mining*, pp. 303–312, 2008.

[14] A. Inokuchi and T. Washio. Mining Frequent Graph Sequence Patterns Induced by Vertices. *Proc. of SIAM Int'l Conf. on Data Mining*. pp. 466-477, 2010.

[15] A. Inokuchi and T. Washio. GTRACE2: Improving Performance Using Labeled Union Graphs. *Proc. of Pacific-Asia Conf. on Knowledge Discovery and Data Mining*. pp. 178-188, 2010.

[16] C. Jianguo, et al. A Mining Method of Frequent Tree Sequences. *Proc. of Int'l Conf. on Intelligent Computation Technology and Automation*, pp. 1032–1035, 2011.

[17] 元田浩. 明示的理解に魅せられて. 人工知能学会学会誌 pp.615-625,1999.

[18] C. W. Leung, et al. Mining interesting link formation rules in social networks. *Proc. of Int'l Conf. on Information and Knowledge Management* pp. 209-218, 2010.

[19] T. Ozaki and T. Ohkawa. Discovery of Correlated Sequential Subgraphs from a Sequence of Graphs *Proc. of Int'l Conf. on Advanced Data Mining and Applications.*, pp. 265–276, 2009.

[20] J. Pei, et al. SeqPatternMiner: Mining Sequential Patterns by Prefix-Projected Growth, *Proc. of Int'l Conf. on Data Eng.*, pp. 2–6, 2001.

[21] A. Sanfeliu and K. Fu. A Distance Measure Between Attributed elational Graphs for Pattern Recognition, *IEEE Transactions on Systems, Man and Cybernetic*, Vol. 13, pp. 353–362, 1983.

[22] T. Washio and H. Motoda. State of the Art of Graph-based Data Mining. *SIGKDD Explorations*, Vol. 5, Issue 1, pp. 59–68, 2003.

[23] 山岡，猪口，鷲尾．単一グラフ系列からの頻出パターン列挙 人工知能学会 第 25 回全国大会, 1D1-2in, 2011