3A2-4

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

Improvement of the performance of graph sequence mining by using closed sequence mining

岸本 卓也　猪口 明博　鷲尾 隆
Takuya Kishimoto　Akihiro Inokuchi　Takashi Washio

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

There are many real-world applications suitable to model objects by using graph sequences. For example, a human network is represented by a graph where each human and each relationship between two humans correspond to vertices and an edge, respectively. If a person joins or leaves a human community, the numbers of vertices and edges in the graph increase or decrease. We have proposed a method called GTRACE for mining frequent patterns from graph sequences. However, it needs much computation time to mine from long and large graph sequences. In this paper, we propose a method which improves the performance of GTRACE and compare the improved GTRACE, called GTRACE(CloSpan), with the original GTRACE to evaluate their efficiencies applying them to synthetic datasets and a real dataset.

1. はじめに

情報技術の発展により，膨大な量のデータを蓄積することが可能となった．しかし，日々肥大化するデータは人間の理解力を超えたため，有益な情報が含まれていてもそのままでは理解できなくなっている．そこで，この膨大なデータから有益な情報を発見するため，近年データマイニングに関する研究は非常に注目され，盛んに研究されている．ここで有益性は人によって異なるため，数学的に厳密に定義することは困難であるが，一般に多くの事例を説明できる知識は有用であると考えられる [元田 99] ことから，データから頻繁に出現するパターンをマイニングする様々な手法が提案されている [Han 06]．例えば，小売店業界では経営戦略に役に立てるため，顧客が一回の購買で同時に購入した商品の集合を蓄えたデータベースから頻繁にかつ同時に購入される商品の集合をマイニングするバスケット分析が使われている．

バスケット分析は，（商品の）集合を蓄えたデータベースから頻繁に購入されるパターンをマイニングする問題であるが，Apriori アルゴリズムが提案されて以降，系列を蓄えたデータベースから頻繁に出現するパターンのマイニングや，順列木，無順序木，グラフ，グラフ系列など様々なデータ構造から頻繁に出現するパターンをマイニングする手法が提案されている．本研究が対象とするグラフ系列マイニングは，グラフ系列から頻出部分グラフ系列をマイニングする．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．このグラフ系列をマイニングすることにより，大規模なグラフ系列に埋もれた，頻繁に現れる構造の変化の発見が期待される．

我々はグラフ系列マイニングの手法として GTRACE (Graph Transformation Sequence Mining)[Inokuchi 08] を提案し，エンロン社の電子メールデータに適用した．GTRACE はグラフの変化に着目し，グラフ系列を簡潔に表現することで効率化を図っているが，10 状態，100 頂点程度の規模のグラフ系列にしか適用できない．そこで，本稿ではさらに効率化を図るために，飽和系列の概念を GTRACE に取り込んだ手法を提案する．そして，人工データと実データとしてエンロン社の電子メールデータに対して提案手法を適用し，その結果について考察を行う．

連絡先: 岸本 卓也，大阪大学 産業科学研究所，567-0047 大阪府茨木市美穂ヶ丘 8-1，kishimoto@ar.sanken.osaka-u.ac.jp

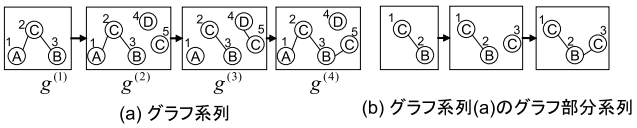

図 1　観測グラフ系列とそのグラフ部分系列の例

2. GTRACE

図 1(a) は観測されたグラフ系列の例を表している．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，一度に大半の人間が入れ替わることはなく，更に各時点では個々の人間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

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，ユニーク 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 $vd^{(j,k)}_{[u,\bullet]}$	ユニーク 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 $vr^{(j,k)}_{[u,l]}$	ユニーク 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 $ei^{(j,k)}_{[(u_1,u_2),l]}$	ユニーク 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 $ed^{(j,k)}_{[(u_1,u_2),\bullet]}$	ユニーク 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 $er^{(j,k)}_{[(u_1,u_2),l]}$	ユニーク 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

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，$V = \{v_1, v_2, \ldots, v_z\}$ は頂点集合，$E = \{(v, v') \mid (v, v') \in V \times V\}$ は辺集合，L は頂点と辺のラベル集合であり，$f : V \cup E \to L$ 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 $d = \langle g^{(1)}\, g^{(2)} \ldots g^{(n)} \rangle$ と表す．$g^{(j)}$ は j 番目に観測されたグラフである．$g^{(1)}$ を系列の先頭，$g^{(n)}$ を系列の末尾とする．グラフの各頂点 v はユニーク ID をもち，$id(v)$ と表す．頂点集合と辺集合に対するユニーク ID の集合 $ID_V(d)$ 及び $ID_E(d)$ を以下のように定義する．

$$ID_V(d) = \left\{ id(v) \mid v \in V(g^{(j)}), g^{(j)} \in d \right\}$$
$$ID_E(d) = \left\{ (id(v), id(v')) \mid (v, v') \in E(g^{(j)}), g^{(j)} \in d \right\}$$

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 $d = \langle g^{(1)} g^{(2)} \ldots g^{(n)} \rangle$ の各グラフ $g^{(j)}$ を外部状態と呼ぶ．さらに，連続する 2 つのグラフ $g^{(j)}$ と $g^{(j+1)}$ の間を補間するグラフ系列を $s^{(j)} = \langle g^{(j,1)} g^{(j,2)} \ldots g^{(j,m_j)} \rangle$ 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．観測グラフ系列 d は補間系列 $d = \langle s^{(1)} s^{(2)} \ldots s^{(n-1)} \rangle$ 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，グラフ編集距離に基づき最短長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 $s^{(j)} = \langle g^{(j,1)} \ldots g^{(j,m_j)} \rangle$ 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．ただし，

- tr は頂点や辺の追加，削除，ラベル変更のいずれか．
- $o_{jk} \in ID_V(d) \cup ID_E(d)$ は変換される頂点や辺のユニーク ID.
- $l_{jk} \in L$ は変換される頂点や辺のラベル． ■

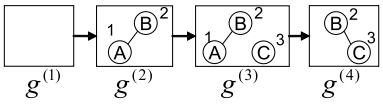

図 2　外部状態系列

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．例えば，j 番目の外部状態の k 番目と $k+1$ 番目の内部状態間で，ラベルが l でユニーク ID が u である頂点の追加を $vi^{(j,k)}_{[u,l]}$ で表す．頂点削除と辺削除はユニーク ID のみの指定で変換可能であるので，変換規則の引数 l はダミー変数であり，‘•’ で表す．

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

このような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，いか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マイニングするため，変換系列 $seq(d')$ が変換系列 $seq(d)$ の部分系列であるとき，$seq(d') \sqsubseteq seq(d)$ と書く．その定義の詳細は [Inokuchi 08] を参照されたい．

例 1. 図 2 の外部状態系列は

$$seq(d) = \langle \underline{vi^{(1,1)}_{[1,A]} vi^{(1,2)}_{[2,B]} ei^{(1,3)}_{[(1,2),-]}} vi^{(2,1)}_{[3,C]} ed^{(3,1)}_{[(1,2),\bullet]} \underline{vd^{(3,2)}_{[1,\bullet]} ei^{(3,3)}_{[(2,3),-]}} \rangle$$

と表される．以下の系列 $seq(d')$ は $seq(d)$ の部分系列であり，$seq(d')$ は $seq(d)$ 中の下線部に対応する．

$$seq(d') = \langle vi^{(1,1)}_{[1,B]} ei^{(1,2)}_{[(1,3),-]} vd^{(2,1)}_{[3,\bullet]} ei^{(2,2)}_{[(1,2),-]} \rangle$$

グラフ系列の集合 $DB = \left\{ (did_i, d_i) \mid d_i = \langle g_i^{(1)} \ \ldots \ g_i^{(n_i)} \rangle \right\}$ に対し，変換部分系列 $seq(d')$ の支持度 $\sigma(seq(d'))$ を

$$\sigma(seq(d')) = \frac{|\{did_i \mid (did_i, d_i) \in DB, seq(d') \sqsubseteq seq(d_i)\}|}{|DB|}$$

と定義する．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，$seq(d'_1) \sqsubset seq(d'_2)$ ならば $\sigma(seq(d'_1)) \geq \sigma(seq(d'_2))$ 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 $DB = \left\{ (did_i, d_i) | d_i = \langle g_i^{(1)} g_i^{(2)} \ldots g_i^{(n_i)} \rangle \right\}$ と最小支持度 σ' が入力として与えられたとき，DB 中の頻出変換部分系列を全て列挙する．

GTRACE は，FTS の末尾に深さ優先探索で変換規則を付加する PrefixSpan[Pei 01] を用いて，$seq(DB)$ から全 FTS を列挙する．

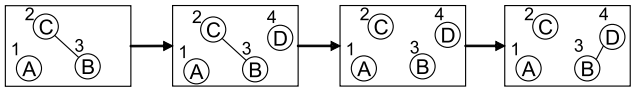

図 3　関連性のない頂点を含む外部状態系列

2.3　関連性のある FTS のマイニング

2.2 節では全ての頻出変換部分系列を列挙するアルゴリズムを示した．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系列 (relevant FTS：rFTS) のみを列挙する．例えば，図 3 のグラフ系列では，ラベルが A でユニーク 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 3 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，頂点と辺のユニーク ID の関連性を以下に定義する．

定義 5. 外部状態系列 $d = \langle g^{(1)} g^{(2)} \dots g^{(n)} \rangle$ 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．

$$V_u = \bigcup \left\{ id(v) \mid v \in V(g^{(j)}), g^{(j)} \in d \right\}$$
$$E_u = \bigcup \left\{ (id(v), id(v')) \mid (v, v') \in E(g^{(j)}), g^{(j)} \in d \right\} \quad \blacksquare$$

定義 5 に基づき，ユニーク ID 間の関連性を定義する．

定義 6. 外部状態系列 d の和グラフが連結グラフであるとき，d のユニーク ID は互いに関連がある．$\blacksquare$

和グラフは変換系列に対しても同様に定義される．和グラフの定義を用いて，rFTS のマイニングを以下のように定義する．

問題 2. グラフ系列の集合 $DB = \left\{ (did_i, d_i) | d_i = \langle g_i^{(1)} g_i^{(2)} \dots g_i^{(n_i)} \rangle \right\}$ 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変換部分系列を全て列挙する．

GTRACE は効率良く rFTS を列挙するため，はじめに，定義 5 に基づいて DB 中のグラフ系列の和グラフを計算する．次に，和グラフの集合から AcGM[Inokuchi 02] を用いて，頻出連結部分グラフを取り出す．頻出連結部分グラフ g_u が取り出されるたびに，2.2 節で述べた PrefixSpan を呼び出す．ここで PrefixSpan の入力である変換系列の集合は以下の射影により生成される．

定義 7. グラフ系列 d の変換系列 $seq(d)$ と連結グラフ g が与えられたとき，$seq(d)$ をその部分系列に射影する proj を以下のように定義する．

$$proj((did, seq(d)), g)$$
$$= \{(did, seq(d')) \mid seq(d') \sqsubseteq seq(d), g_u(d') = g \wedge \nexists seq(d'')$$
$$\text{s.t. } (seq(d') \sqsubset seq(d'') \sqsubseteq seq(d) \wedge g_u(d'') = g)\} \quad \blacksquare$$

この射影により一つのグラフ系列から複数の変換系列が生成される．そして，PrefixSpan により列挙された FTS の和グラフが g_u と同型ならば，それを rFTS として出力する．

3.　提案手法

本章では，既存の GTRACE に対して，その実行速度を向上させる新たな手法について述べる．既存の GTRACE の射影によって生成される変換系列は非常に多くなるため，GTRACE の内部で呼ばれる PrefixSpan の計算時間は非常に長くなる．そこで本稿では，GTRACE の内部で呼ばれる PrefixSpan を CloSpan[Yan 03] に置き換えることにより，出力される頻出変換系列パターンから冗長なパターンを除き，計算時間を短くした手法 GTRACE(CloSpan) を提案する．

系列パターンマイニングでは，アイテムをその生起順に並べた系列の集合から多くの系列に共通して出現するアイテムの生起パターンを探索する．従って，このアルゴリズムの PrefixSpan 及び CloSpan の入力はどちらも系列の集合であり，出力はそれぞれ頻出系列パターンの集合及び飽和頻出系列パターンの集合である．ここで，飽和は以下のように定義される．

定義 8. 頻出系列パターンの集合 FS において，ある系列パターン α の支持度が α と同じで，α を包含する系列が FS の中に存在しなければ，α は飽和であるという．$\blacksquare$

飽和頻出系列パターンの集合は頻出系列パターンの集合の部分集合であり，頻出系列パターンは飽和頻出系列パターンから生成可能であるため，飽和頻出系列パターンを探索することによって等価な情報をよりコンパクトに表すことが出来る．従って，CloSpan では飽和系列を含まない探索空間の一部を枝刈りしながら探索を行うため，CloSpan は PrefixSpan よりも計算時間が短くなる．

4.　評価実験

本章では，3 章で述べた提案手法を C++ で実装し，人工データ及び実データへ適用させて実験した結果を比較し，その効果について述べる．実験は，Intel Xeon CPU W3565 3.20GHz のプロセッサ，12GB 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．

4.1　人工データ

人工データは次のように作成した．はじめに，平均頂点数 3, 2 頂点間の辺存在確率 15% でグラフ系列の 1 つ目の外部状態 $g^{(1)}$ を生成する．次に，確率 80% で頂点（あるいは辺）を追加する変換規則を系列に付加し，平均 6 個のユニーク ID を持つ変換系列 d を $|D|$ 個生成した．この処理では，連続する二つの外部状態の間で，平均 2 個の変換規則を付加した．また，各外部状態のグラフが疎グラフであることを保ちつつ，各変換系列 $seq(d)$ の和グラフが連結になるまで変換規則の付加を続けた．同様にして，頂点や辺の追加の変換規則の選択確率 50%，ユニーク ID の平均 3 個で 10 個の rFTS を生成し，各系列 $d \in D$ に rFTS の 1 つを上書きした．外部状態のグラフは 5 種の頂点ラベルと 1 種の辺ラベルを持つよう人工データを生成した．

表 2 は，$|D| = 1000$ で最小支持度 σ' を 30% から 5% 刻みに 5% まで減少させたときの計算時間及び取り出された rFTS の数を表したものである．最小支持度を下げると，計算時間は指数関数的に増加する．また，大きな計算時間を要する低い最小支持度において，GTRACE の内部で呼び出される PrefixSpan を CloSpan に置き換えることによる効果が大きくなることが示されている．これは，最小支持度を低くすると，rFTS の数が増える一方で探索空間の枝刈りが出来るからである．

図 4 は，$\sigma' = 20\%, 30\%$ で $|D|$ を 1000 個から 1000 個刻みに 10000 個まで増加させたときの計算時間を表したものであ

表 2　人工データに対して最小支持度の変化による計算時間と発見された rFTS の変化（データ数 1000 個）

	GTRACE(CloSpan)		GTRACE	
σ'(%)	time（秒）	# of rFTSs	time（秒）	# of rFTSs
30	75.913	1862	83.907	2049
25	131.213	3431	144.754	3749
20	241.704	6841	268.007	7518
15	539.061	16361	592.145	18099
10	1511.015	55258	1707.814	62115
5	8547.865	338335	13006.728	893467

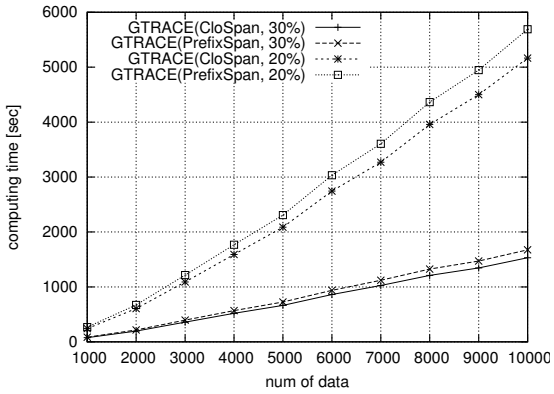

図 4　人工データに対してデータ数 $|D|$ と計算時間の関係

表 3　実データに対して最小支持度の変化による計算時間と発見された rFTS の変化

	GTRACE(CloSpan)		GTRACE	
σ'(%)	time（秒）	# of rFTSs	time（秒）	# of rFTSs
50	114.397	1371	135.691	1687
49	176.677	1551	215.487	1905
48	221.305	1750	258.602	2142

る．計算時間は $|D|$ に比例して長くなっている．最小支持度が 30% の時，CloSpan 利用による枝刈りの効果によって全領域において約 10% の速度向上が見られる．表 2 で示されたように，最小支持度を下げると，枝刈りの効果はさらに大きくなる．

4.2　実データ

提案手法の性能を評価する実験を実データを用いて行った．用いた実データはエンロン社電子メールデータ [Enr] である．このデータは，1998 年 11 月 16 日（月）から 2001 年 3 月 25 日（日）までの 123 週，182 人の間での電子メールやりとりを表したデータである．182 人それぞれが固有の名前，すなわち，ユニーク ID を持ち，ある二人が 1 日の間に電子メールコミュニケーションを取ると 2 頂点間に辺を張り，その 1 日に対応するグラフ $g^{(j)}$ を生成した．また，各頂点には，“CEO”, “Director”, “Employee”, “Lawyer”, “Manager”, “President”, “Trader”, “Vice President” のいずれかが頂点ラベルとして割り当てられている．各週を単位として各々一つのグラフ系列を生成した．すなわち，入力となるグラフ系列数は 123 である．

表 3 は，最小支持度 σ' を 50% から 1% 刻みに 48% まで減少させたときの計算時間及び取り出された rFTS の数を表したものである．最小支持度を下げると，発見される rFTS の数が増加するため計算時間は長くなるが，提案手法により探索空間の枝刈りが行われるため，GTRACE の計算時間が削減されている．

表 4　実データに対してユニーク ID 数の変化による計算時間と発見された rFTS の変化（最小支持度 50%）

	GTRACE(CloSpan)		GTRACE	
人数	time（秒）	# of rFTSs	time（秒）	# of rFTSs
100	0.02	38	0.023	40
120	0.137	378	0.163	467
140	0.296	419	0.364	515
160	14.805	886	17.425	1095
182	114.397	1371	135.691	1687

表 4 は，グラフ系列に含まれる人数を 100 人から 20 人刻みに 160 人まで及び 182 人に増加させた時の計算時間，GTRACE によって取り出された rFTS の数を表したものである．グラフ系列を生成するための人は 182 人からランダムに選択した．人数を増加させると，GTRACE の入力であるグラフ系列のユニーク ID 数が多くなるので，そこに含まれる rFTS の数が指数関数的に増加する．従って，計算時間も指数関数的に長くなる．

5.　まとめ

本稿では，既存の GTRACE を改良し，計算時間を削減する新たな手法について述べ，その性能評価の実験を行った．本稿で提案した GTRACE(CloSpan) の手法では，既存の GTRACE の中で呼び出されている PrefixSpan をその改良手法である CloSpan に置き換えた結果，CloSpan の探索空間削減の効果による性能向上が確かめられた．これにより，提案手法の GTRACE(CloSpan) を用いることで，既存のグラフ系列マイニングを適用できたグラフ系列よりも系列数が多く，系列長が長く，外部状態の頂点数が多いグラフ系列に対しても，より高速にグラフ系列マイニングを適用できるようになった．

参考文献

[Enr] Enron Email Dataset, http://www.cs.cmu.edu/~enron/

[Han 06] Han, J. and Kamber, M.: *Data Mining: Concepts and Techniques*, Morgan Kaufmann (2006)

[Inokuchi 02] Inokuchi, A., Washio, T., Nishimura, Y., and Motoda, H.: A Fast Algorithm for Mining Frequent Connected Graphs, *IBM Research Report* (2002)

[Inokuchi 08] Inokuchi, A. and Washio, T.: A Fast Method to Mine Frequent Subsequences from Graph Sequence Data, in *Proc. of Eighth IEEE International Conference on Data Mining, 2008. ICDM'08*, pp. 303–312 (2008)

[Pei 01] Pei, J., Han, J., Mortazavi-Asl, B., Pinto, H., Chen, Q., Dayal, U., and Hsu, M.: Prefixspan: Mining Sequential Patterns by Prefix-Projected Pattern Growth, in *Proc. of the 17th International Conf. on Data Engineering*, pp. 2–6 (2001)

[Yan 03] Yan, X., Han, J., and Afshar, R.: CloSpan: Mining Closed Sequential Patterns in Large Datasets, in *Proc. of 2003 SIAM International Conf. on Data Mining (SDM'03)* (2003)

[元田 99] 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)