

遺伝的プログラミングによる 特徴的な TTSP グラフパターンの獲得

永井 聡^{†1} 宮原 哲浩^{†1}
鈴木 祐介^{†1} 内田 智之^{†1}

グラフ構造データからの機械学習やデータマイニングの研究が注目されている。TTSP グラフ (Two-Terminal Series Parallel graph) は、電気ネットワークやスケジューリングをコンピュータで扱う際にデータモデルとして用いられることが多い。TTSP グラフは並列操作、直列操作と呼ばれる操作を帰納的に繰り返して得られるサイクルを持たない多重辺を許す有向グラフである。本論文では、遺伝的プログラミングを用いて、正事例と負事例から特徴的な TTSP グラフパターンを獲得する進化的手法を提案する。

Acquisition of Characteristic TTSP Graph Patterns by Genetic Programming

SATOSHI NAGAI,^{†1} TETSUHIRO MIYAHARA,^{†1}
YUSUKE SUZUKI^{†1} and TOMOYUKI UCHIDA^{†1}

Machine learning and data mining from graph structured data are studied intensively. TTSP (Two-Terminal Series Parallel) graphs are used as data models for electric networks and scheduling. A TTSP graph is a directed acyclic graph obtained by recursively applying "series" and "parallel" operations. We propose an evolutionary method for acquiring characteristic TTSP graph patterns from positive and negative data by using Genetic Programming.

^{†1} 広島市立大学情報科学研究科知能工学専攻
Department of Intelligent Systems, Graduate School of Information Sciences, Hiroshima City University

1. はじめに

グラフ構造データからの機械学習やデータマイニングの研究が注目されている⁷⁾。本論文では、様々なグラフ構造データの中から TTSP グラフ (Two-Terminal Series Parallel graph) と言われるグラフ構造データに着目する。TTSP グラフとは、電気ネットワークやスケジューリングをコンピュータで扱う際にデータモデルとして用いられることが多い。TTSP グラフは並列操作、直列操作と呼ばれる操作を帰納的に繰り返して得られるサイクルを持たない多重辺を許す有向グラフである。

高味ら⁶⁾ は、TTSP グラフに変数の概念を持たせたグラフパターンである TTSP 項グラフ (TTSP グラフパターン) を導入し、TTSP 項グラフを定義する操作を木で表現する構文木を考案し、TTSP 項グラフと TTSP グラフの多項式時間マッチングアルゴリズムを与えている。さらに、正事例から極小一般化 TTSP 項グラフを効率的に学習する手法を提案している。また、河野ら²⁾ は正事例から極大頻出 TTSP 項グラフを枚挙する手法を提案している。

遺伝的プログラミング (Genetic Programming, GP) とは遺伝子型を主に配列で表現していた遺伝的アルゴリズム (Genetic Algorithm, GA) を拡張して、木構造やグラフ構造などの構造的表現を扱えるようにした進化的計算手法である⁵⁾。長嶺ら⁴⁾ は、タグ木パターン³⁾ と呼ばれる木構造パターンを用いて、木構造データとみなせる糖鎖データからの特徴抽出手法を提案している。正事例と負事例の木構造データから、正事例の特徴を表現する特徴的な木構造パターンを遺伝的プログラミングにより生成した。また、有向グラフ構造を扱う進化的計算手法に遺伝的ネットワークプログラミング (Genetic Network Programming, GNP) がある。遺伝的ネットワークプログラミングは動的な問題に対して有効であり、片桐ら¹⁾ は、遺伝的ネットワークプログラミングを用いて、知的エージェントの行動系列のモデル化と自動獲得を行った。

本論文では、正事例と負事例からなる TTSP グラフの集合から、特徴的な TTSP グラフパターンを獲得する進化的手法を提案する。進化的手法は遺伝的プログラミングを用い、遺伝操作は複製、逆位、交叉、突然変異を用いた。長嶺ら⁴⁾ とは異なり、木構造パターンに対して遺伝操作を適用するのではなく、TTSP 項グラフの構文木に対して遺伝操作を適用する。人工的に生成した TTSP グラフに対して実験を行い、提案する特徴的な TTSP グラフパターンを獲得する進化的手法の有効性を確認した。

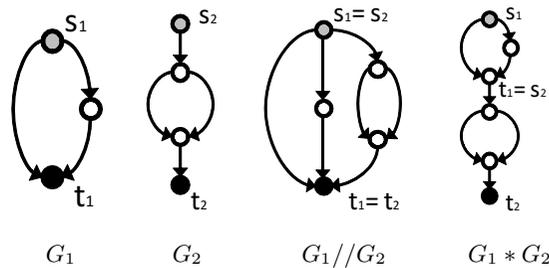


図 1 TTSP グラフ $G_1, G_2, G_1//G_2, G_1 * G_2$

2. TTSP 項グラフ

次の (1),(2) で帰納的に定義される多重辺を許すサイクルを持たない有向グラフを TTSP グラフという。

- (1) 2 つの頂点 u, v と u から v への 1 つの辺から成る有向グラフは、ソースとして u を持ちシンクとして v を持つ TTSP グラフである。
- (2) G_1, G_2 をソース s_1, s_2 とシンク t_1, t_2 をそれぞれ持つ TTSP グラフとする。このとき、 s_1 と s_2, t_1 と t_2 を同一視する操作を並列操作（得られるグラフは $G_1//G_2$ と表現する）という。 s_2 と t_1 を同一視する操作を直列操作（得られるグラフは $G_1 * G_2$ と表現する）という。並列操作、直列操作のどちらかによって得られるグラフは TTSP グラフである。この操作を図 1 に示す。

TTSP グラフのいくつかの辺を変数にしたグラフパターンを TTSP 項グラフという。TTSP 項グラフの変数は、TTSP グラフで置き換えることのできる構造的な変数である。TTSP 項グラフ g と TTSP グラフ G に対して、 g の変数を適切な TTSP グラフで置き換えることで G が得られるならば、 g と G はマッチするという。TTSP グラフや TTSP 項グラフを定義する操作を木で表現したものを構文木という。図では、TTSP 項グラフや構文木の変数は箱と線で表す。例えば、図 2 において、TTSP 項グラフ g の変数 x を TTSP グラフ G_1 で、変数 y を TTSP グラフ G_2 で置き換えると TTSP グラフ G が得られるので、 g と G はマッチする。 g の構文木は t_g である。構文木 t_g の x と y は変数を表し、 p は並列操作、 s は直列操作を表す。

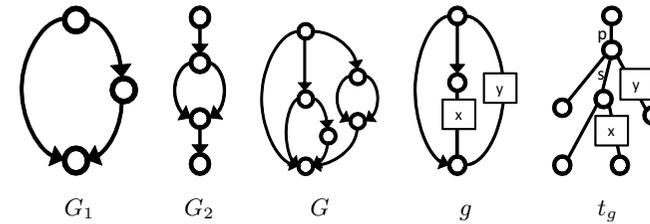


図 2 TTSP グラフ $G_1, G_2, G, TTSP$ 項グラフ g , 構文木 t_g

3. 進化的手法による特徴的な TTSP 項グラフの獲得

木構造表現に対する進化的手法である遺伝的プログラミング (GP) による TTSP 項グラフ発見問題を以下のように定義する。

GP による TTSP 項グラフ発見問題

入力：正事例と負事例からなる TTSP グラフの有限集合 D 。

問題：GP により、適合度の高い TTSP 項グラフを発見する。

ここで、TTSP 項グラフ g の適合度 $fitness_D(g)$ を (g が D の正事例にマッチする割合 + g が D の負事例にマッチしない割合) / 2 と定義する。

入力の TTSP 項グラフは構文木で与えられるものとし、GP の過程では TTSP 項グラフの構文木に対して遺伝操作を行う。適合度の高い TTSP 項グラフは、 D の多くの正事例にマッチし、 D の負事例にあまりマッチしないような、特徴的な TTSP 項グラフであると考えられる。TTSP 項グラフ発見問題に対する GP の手順を次のようにする。

TTSP 項グラフ発見問題に対する GP の手順

- (1) D から、TTSP 項グラフで使用するキーワードの有限集合 KW を求める。
- (2) KW に含まれるキーワードを辺ラベルとしてランダムに初期 TTSP 項グラフ集合を発生させる。
- (3) 集団の各 TTSP 項グラフ g の適合度 $fitness_D(g)$ を求める。
- (4) 適合度の大きさに比例した確率によって TTSP 項グラフの選択を行う。
- (5) 遺伝操作 (複製, 逆位, 交叉, 突然変異) により、次世代の集団を生成する。
- (6) 終了条件が満たされているときは終了する。そうでなければ (3) へ戻る。

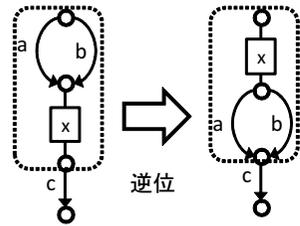


図 3 逆位の適用例

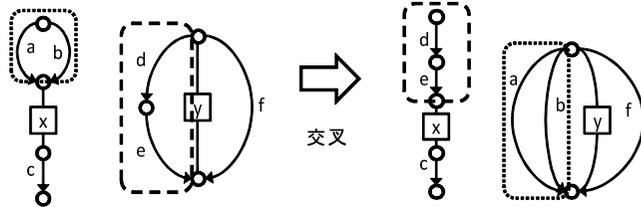


図 4 交叉の適用例

遺伝操作の逆位とは TTSP 項グラフの対象とする部分の順序を入れ替える操作であり、交叉とは 2 つの TTSP 項グラフの対象とする部分を入れ替える操作である。突然変異には以下の 4 つの操作がある。add-TTSP: TTSP 項グラフにランダムに生成した TTSP 項グラフを加える操作。del-TTSP: TTSP 項グラフの選択した部分を削除する操作。change-TTSP: TTSP 項グラフの選択した部分をランダムに生成した TTSP 項グラフと入れ替える操作。change-label: TTSP 項グラフの対象とする辺のラベルを替える操作。遺伝操作の逆位、交叉、突然変異の add-TTSP の適用例をそれぞれ、図 3, 4, 5 に示す。

4. 実験結果

3 節で提案した、正事例と負事例から特徴的な TTSP 項グラフを獲得する進化的手法を実現した。個体である TTSP 項グラフの適合度を計算するために、高味ら⁶⁾の TTSP 項グラフと TTSP グラフのマッチングアルゴリズムを用いた。実験環境として Intel(R) Xeon(R) CPU X5482 3.20GHz 3.19GHz (2 プロセッサ), RAM 32.0GB, OS Microsoft Windows 7 64 ビット, JAVA(jre1.6.0.29) を用いた。

実験に用いた GP のパラメータを表 1 に示す。ランダムに生成された TTSP グラフのう

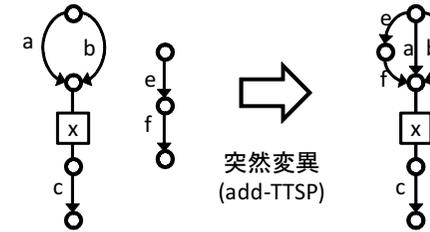


図 5 突然変異 (add-TTSP) の適用例

表 1 本実験における GP のパラメータ

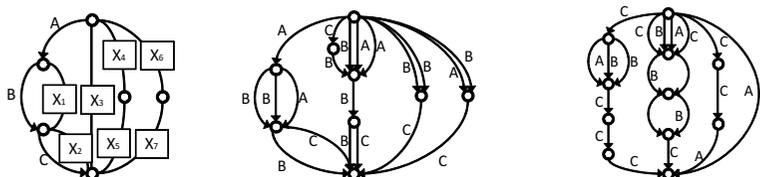
個体数	50	突然変異確率	0.45
複製確率	0.05	選択方法	ルーレット選択, トーナメント (サイズ 4)
逆位確率	0.05		エリート保存 (サイズ 3)
交叉確率	0.45	最大世代数	200

ちで、ある TTSP 項グラフ π_0 にマッチするもの 100 個を正事例、マッチしないもの 100 個を負事例とする (図 6 参照)。以下の 2 つの実験を 10 試行を行った。実験 a では遺伝操作に制限を設けず、実験 b では遺伝操作の際に、生成する TTSP 項グラフの構文木の大きさ (頂点数) をある定数以下にする制限を設ける。実験 a における最終世代の最も適合度が高い個体 (最良個体という) の例を図 7 に示す。

2 つの実験で行った 10 試行の平均値を図 8,9 に示す。図 8 は、各世代の適合度の最も高い個体の適合度 (1stfit)、各世代の集団における個体の適合度の平均値 (avefit) を示す。図 9 は、各世代の集団における個体の構文木の大きさの平均値 (avesize)、各世代の集団における最大構文木の大きさ (maxsize) を示す。実験 a に対して、図 8,9 の全ての値が収束していることから、GP による構文木で表された TTSP 項グラフの探索を行うと、最終世代で高い適合度を持つ個体が得られて、プロート現象が起きず正常に探索を終了していることがわかる。また、実行時間の平均値は実験 a が 1475 秒、実験 b が 1182 秒であった。よって、実験 b に対する図 8,9 の値より、遺伝操作に制限を設けても結果が変化せず実行時間が短くなり正常に探索を終了していることがわかる。

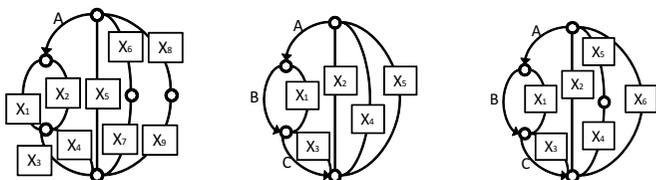
5. おわりに

本研究では、正事例と負事例から特徴的な TTSP 項グラフを獲得する進化的手法を提案



TTSP 頂グラフ π_0 正事例の TTSP グラフの例 負事例の TTSP グラフの例

図 6 データの生成に用いた TTSP 頂グラフ π_0 , 正事例と負事例の TTSP グラフの例



6 試行目の最良個体 8 試行目の最良個体 10 試行目の最良個体

図 7 実験 a で得られた最良個体である TTSP 頂グラフ

した。今後の課題として、より具体化された TTSP 頂グラフの獲得を図るために、より良い適合度の考案、遺伝操作による個体選択確率の変更などが考えられる。また、実データを入力とすることも考えられる。

参考文献

- 1) H.Katagiri, K.Hirasawa, J.Hu, Genetic Network Programming - Application to Intelligent Agents, Proc. IEEE Int. Conf. Systems, Man, and Cybernetics, pp.3829-3834, 2000
- 2) T.Kono, Y.Suzuki, T.Uchida, T.Miyahara, Enumerating Maximally Frequent TTSP Graph Patterns, Proc. 7th Workshop on Learning with Logics and Logics for Learning (LLL), pp.43-50, 2011
- 3) T.Miyahara, Y.Suzuki, T.Shoudai, T.Uchida, S.Hirokawa, K.Takahashi, H.Ueda, Extraction of Tag Tree Patterns with Contractible Variables from Irregular Semistructured Data, Springer-Verlag LNAI 2637, pp.430-436, 2003
- 4) M.Nagamine, T.Miyahara, T.Kuboyama, H.Ueda, K.Takahashi, A Genetic Programming Approach to Extraction of Glycan Motifs Using Tree Structured Patterns, Springer-Verlag LNAI 4830, pp.150-159, 2007

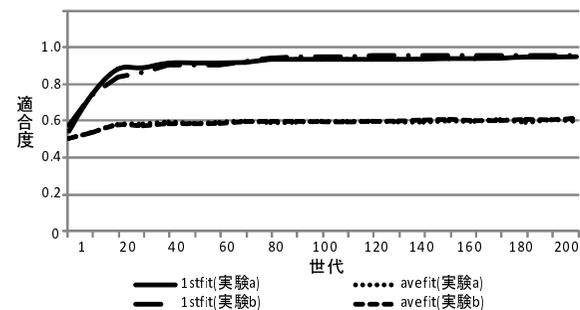


図 8 各世代の適合度の推移

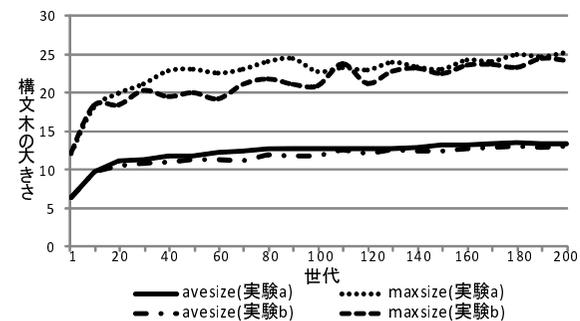


図 9 各世代の構文木の大きさの推移

- 5) R.Poli, W.Langdon, N.McPhee, A Field Guide to Genetic Programming, Lulu Press, 2008
- 6) R.Takami, Y.Suzuki, T.Uchida, T.Shoudai, Polynomial Time Inductive Inference of TTSP Graph Languages from Positive Data, IEICE Transactions, E92-D(2), pp.181-190, 2009
- 7) T.Washio, H.Motoda, State of the Art of Graph-based Data Mining, SIGKDD Explorations 5(1), pp.59-68, 2003