グラフの向き付けに関する最適化問題の解法

小林　佑輔
（東京大学工学部計数工学科　現所属・同大学院情報理工学系研究科数理情報学専攻）
指導教員　岩田　覚 助教授

1. はじめに

$2k$ 辺連結な無向グラフが与えられたときに，k 枝連結となるような最小費用の向き付けを求める問題は，最小費用 k 枝連結向き付け問題と呼ばれ，これは劣モジュラ流問題として定式化される問題である．この問題に対し Frank[2] は劣モジュラ流問題としての定式化を用いて計算時間 $O(k^2n^3m)$ のアルゴリズムを構成している．ただし，頂点数を n，辺の数を m とする．また，Gabow[3] は Frank のアルゴリズムに複雑なデータ構造を導入することにより，$O(kn^2m)$ の計算時間によるアルゴリズムを実現している．本研究では，複雑なデータ構造を用いることなく $O(k^3n^3+kn^2m)$ の計算時間で最小費用 k 枝連結向き付け問題を解くアルゴリズムを示した．また，この応用として有向カット被覆を $O(n^2m)$ の計算時間で見出すアルゴリズムを示した．

2. 最小費用 k 枝連結向き付け問題

無向グラフ $\bar{G}=(V,E)$ は，任意の点集合 $X\neq\emptyset, V$ に対して $\delta(X)\geq k$ となるとき，k 辺連結であるという．また，有向グラフ $G=(V,A)$ が k 枝連結であるとは，任意の頂点集合 $X\neq\emptyset, V$ に対して $\delta^+(X)\geq k$ となることである．ここで $\delta(X)$ は X と $V\backslash X$ とを結ぶ辺の数を表し，$\delta^+(X)$ は X から出て $V\backslash X$ に入る枝の数を表す．すると，以下の定理が成立することが知られている．

定理1 (Nash-Williams[4])．無向グラフ $\bar{G}=(V,E)$ が $2k$ 辺連結であるとき，かつそのときに限り，各辺に適当な向きを付けることにより，k 枝連結であるような有向グラフ $G=(V,A)$ を得ることができる．

定理1に関連して以下の最適化問題が自然に考えられる．

問題. $2k$ 辺連結な無向グラフに対し，各辺 $e=uv$ の両方の向きにそれぞれ向き付けの費用 $c(uv), c(vu)$ が与えられている．このとき，k 枝連結であるような有向グラフが得られる向き付けのうち費用が最小となるものを求めよ．

これを最小費用 k 枝連結向き付け問題という．最小費用 k 枝連結向き付け問題は，最小費用有向カット被覆問題なども含んだ広い枠組みである劣モジュラ流問題として捉えることにより，多項式時間解法が得られている．しかし，実際に効率的なアルゴリズムを構成する際には，各問題に特有の性質をうまく使うことが重要となる．

最小費用 k 枝連結向き付け問題に対して，Frank はこの問題が劣モジュラ流問題に帰着されることを用いて計算時間 $O(k^2n^3m)$ の解法を構成している．また，Gabow は Frank のアルゴリズムを基本とし，セントロイド木（centroid tree）と呼ばれる複雑なデータ構造を導入することにより，$O(kn^2m)$ の計算時間によるアルゴリズムを実現している．本研究では，Frank のアルゴリズムを基本とし，複雑なデータ構造を用いることなく $O(k^3n^3+kn^2m)$ の計算時間で最小費用 k 枝連結向き付け問題を解くアルゴリズムを開発した．これは，$k^2n=O(m)$ となるときには Gabow のアルゴリズムと同じ計算時間であり，しかもより単純なアルゴリズムとなっている．これらのアルゴリズムの比較を表1に示す．なお，表中の $M(n)$ は n 行 n 列の行列同士の積の計算にかかる時間を表す．

Frank のアルゴリズムは，始めに k 枝連結な向き付けを求め，そこから，新たな k 枝連結な向き付けへの更新を繰り返していくものである．ある k 枝連結な向き付けが与えられているとき，$X\subseteq V$ が臨界集合であるとは，$\delta^+(X)=k$ であることを表す．v を

表1　最小費用 k 枝連結向き付け問題の解法の比較

	時間計算量	領域計算量
Frank [2]	$O(k^2n^3m)$	$O(n^2)$
Gabow [3]	$O(kn^2m)$	$O(m)$
	$O(knM(n))$	$O(n^2)$
本論文	$O(k^3n^3+kn^2m)$	$O(n^2)$

© 日本オペレーションズ・リサーチ学会．無断複写・複製・転載を禁ず．

含むすべての臨界集合の積集合を $R(v)$ とすると，最小費用 k 枝連結向き付け問題を解くアルゴリズムの更新手続きの中では，すべての頂点 u，$v\in V$ に対して $u\in R(v)$ であるか否かを判定することが必要となる．Gabowは，複雑なデータ構造を用いてすべての臨界集合を表現することにより，$u\in R(v)$ であるか否かを判定したが，本研究では，$u\in R(v)$ であるか否かの判定に，必ずしもすべての臨界集合が必要なわけではないことに注目して，複雑なデータ構造を必要とせずに，$u\in R(v)$ であるか否かを以下のように判定した．

ある固定した頂点 r を含まず，v を含む臨界集合全体を $\mathcal{F}_v^+$ とし，r を含み，u を含まない臨界集合全体を $\mathcal{F}_u^-$ とおく．すると，$\mathcal{F}_v^+, \mathcal{F}_u^-$ は和集合と積集合に関して閉じている．よって，$\mathcal{F}_v^+$ の中で最小の集合が存在するのでこれを R_v^+ で表す．ただし，$\mathcal{F}_v^+=\emptyset$ のときには $R_v^+=V$ とする．また，$\mathcal{F}_u^-$ の中で最大の集合が存在するのでこれを R_u^- で表す．ただし，$\mathcal{F}_u^-=\emptyset$ のときには $R_u^-=\emptyset$ とする．すると $u\in R(v)$ であることは，$u\in R_v^+$ かつ $v\notin R_u^-$ であることと同値である．そこで，すべての頂点について R_v^+ と R_u^- を求めることにより $u\in R(v)$ であるか否かを判定することができる．

3. 有向カット被覆への応用

弱連結な有向グラフが与えられたときに，いくつかの枝を縮約して強連結なグラフを作ることを考える．ただし，有向グラフが弱連結であるとは，枝の向きを無視した無向グラフが連結であることをいう．各枝を縮約する費用が与えられているときに，強連結なグラフを得る最小費用の縮約を求める問題を最小費用有向カット被覆問題という．本研究では，最小費用有向カット被覆問題が最小費用 k 枝連結向き付け問題の $k=1$ の場合に帰着されることを用いた，$O(n^2m)$ の計算時間による単純な解法を示した．

最小費用有向カット被覆問題に対しては表2のような計算時間で解くアルゴリズムが知られている．

Frankのアルゴリズムは劣モジュラ流問題の最適性条件に基づいた主双対アルゴリズムであり，GabowはFrankのアルゴリズムに，最小費用 k 枝連結向き付け問題に用いたものと同じ複雑なデータ構造を導入することにより，$O(n^2m)$ の計算量を実現している．それに対しShepherd-Vettaは，劣モジュラ流問題の持つ性質に加えてこの問題特有の性質をうまく用いることにより，複雑なデータ構造を用いる必要のない主アルゴリズムでGabowのアルゴリズムと同じ計算時間を持つものを構成した．しかし，Shepherd-Vettaのアルゴリズムは前処理を多く施すために，グラフを n 個分保持する必要があり，領域計算量が $O(nm)$ となっている．本研究のアルゴリズムは，複雑なデータ構造を用いることのない計算時間 $O(n^2m)$ のアルゴリズムで，しかも領域計算量が $O(n^2)$ となっている．このアルゴリズムを実装したところ，CPUがIntelのPentium 4 1.60 GHz，メモリが1 GBである計算機で，$n=50$，$m=100$ の問題を1秒以内で，$n=200$，$m=400$ の問題を1分程度で解くことができた．また，n，m の値を変化させて，計算時間がおよそ n^2m に比例することを確認した．

表2　最小費用有向カット被覆問題の解法の比較

	時間計算量	領域計算量
Frank [1]	$O(n^3m)$	$O(n^2)$
Gabow [3]	$O(n^2m)$	$O(m)$
	$O(nM(n))$	$O(n^2)$
Shepherd-Vetta [5]	$O(n^2m)$	$O(nm)$
本論文	$O(n^2m)$	$O(n^2)$

参考文献

[1] A. Frank: How to make a digraph strongly connected. *Combinatorica*, 1 (1981), pp. 145-153.

[2] A. Frank: An algorithm for submodular functions on graphs. *Annals of Discrete Mathematics*, 16 (1982), pp. 97-120.

[3] H. N. Gabow: Centroids, representations, and submodular flows. *J. Algorithms*, 18 (1995), pp. 586-628.

[4] C. St. J. A. Nash-Williams: On orientations, connectivity and odd-vertex-pairings in finite graphs. *Canadian J. Mathematics*, 12 (1960), pp. 555-567.

[5] F. B. Shepherd, A. Vetta: Visualizing, finding and packing dijoins. *Graph Theory and Combinatorial Optimization*, D. Avis, ed., Springer-Verlag, 2005, pp. 219-254.

© 日本オペレーションズ・リサーチ学会．無断複写・複製・転載を禁ず．