最大隣接順序を用いた最大流アルゴリズムの改良と実装

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

1. はじめに

最大流問題とは，与えられたネットワークにおける最大フローを探す問題のことで，様々な最適化問題に適用できることからこれまでに活発な研究かなされている．1988年にGoldberg-Tarjanによってプッシュ・リラベル法が提案され，ネットワークの点の数をn，辺の数をm，最大容量をUとすると，$O(mn\log(n^2/m))$の計算量で最大流問題が解けるようになった．このGoldberg-Tarjanのアルゴリズムが現在のところ実用的にも最速といわれている．

そうした流れの中，2003年にFujishige[1]が最大隣接順序に基つく新しい枠組みの最大流アルゴリズムを提案した．Fujishigeのアルゴリズムは理論的には$O(n(m+n\log n)\log nU)$の計算量であるが，強多項式時間ではないことが示されている．実用的にも，Goldberg-Tarjanのアルゴリズムと比較すると大きく劣っていた．

そこで本論文では，Fujishigeのアルゴリズムに対する改良として，プリフローを用いたアルゴリズムを提案する．計算機実験を行ったところ，実行時間が改良前のアルゴリズムに比べて10分の1程度と大きく減っており，実用的な速度においてGoldberg-Tarjanのアルゴリズムに大きく近ついたといえる．

2. ネットワークに関する諸定義

ネットワークは$\mathcal{N}=(G=(V,A),s^+,s^-,c)$と書かれ，$G=(V,A)$は$V$を点集合，$A$を枝集合とする有向グラフ，$s^+\in V$は入口，$s^-\in V$は出口，$c$ $A\to\mathbf{Z}_+$は容量関数をそれぞれ表している．また，$|V|=n, |A|=m$とおく．

ネットワーク$\mathcal{N}$が与えられたとき，入口s^+から出口s^-へのフローφ　$A\to\mathrm{R}$とは容量制約，流量保存則を満たす実数値関数である．また，入口から流出する量をフローφの流量という．

ネットワーク$\mathcal{N}$にフローφが流れているとき，あとどれぐらい流せるかを考えたネットワークを残余ネットワークといい，$\mathcal{N}_\varphi$で表す．

入口s^+から出口s^-へのプリフローφ: $A\to\mathrm{R}$とは次の条件(1)，(2)を満たす関数である．

(1) 容量制約：各枝$a\in A$に対して，$0\le\varphi(a)\le c(a)$.

(2) フロー境界条件：各点$v\in V\setminus\{s^+,s^-\}$に対して，

$$\partial\varphi(v)=\sum_{a\ vから出る枝}\varphi(a)-\sum_{a\ vに入る枝}\varphi(a)\le 0$$

つまり，フローと異なり流量が保存されていなくてよい．$-\partial\varphi(v)(\ge 0)$をプリフローの残存量といい，残存量が正の点を活性点という．また，出口に流入する量$-\partial\varphi(s^-)$をプリフローφの流量$\hat{v}(\varphi)$と定義する．

3. プリフローを用いた改良案

改良したアルゴリズムは次のとおりである．

Step 0：入り口から出る枝$a=(s^+,u)\in A$に対して，$\varphi(a)\leftarrow c(a)$．それ以外の枝$a$に対しては，$\varphi(a)\leftarrow 0$.

Step 1-1：MA-Ordering$(\mathcal{N}_\varphi,s^-)$を行い，出口からの順序$(v_0(=s^-),v_1,\cdots,v_k)$を得る．もしすべての$i=1,\cdots,k$に対して$\partial\varphi(v_i)=0$を満たすならばStep 2-1へすすむ．

Step 1-2：$i=k,k-1,\cdots,1$について，すべてのv_iから出る枝$(v_i,u)\in L_{v_i}$に対して，push(v_i,u):

$(v_i,u)\in A^+_\varphi$ならば，

$\varphi(v_i,u)\leftarrow\varphi(v_i,u)+\min\{-\partial\varphi(v_i),c_\varphi(v_i,u)\}$,

$(v_i,u)\in A^-_\varphi$ならば，

$\varphi(u,v_i)\leftarrow\varphi(u,v_i)-\min\{-\partial\varphi(v_i),c_\varphi(v_i,u)\}$

Step 1-1にもどる．

Step 2-1：MA-Ordering$(\mathcal{N}_\varphi,s^+)$を行い，入口からの順序$(v_0(=s^+),v_1,\cdots,v_k)$を得る．もしすべての$v\in V\setminus\{s^+,s^-\}$に対して$\partial\varphi(v)=0$ならば終了．

Step 2-2：$i=k,k-1,\cdots,1$について，すべてのv_iから出る枝$(v_i,u)\in L_{v_i}$に対して，push(v_i,u):

$(v_i,u)\in A^+_\varphi$ならば，

$\varphi(v_i,u)\leftarrow\varphi(v_i,u)+\min\{-\partial\varphi(v_i),c_\varphi(v_i,u)\}$,

© 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．

$(v_i, u) \in A_\varphi^-$ ならば，

$\varphi(u, v_i) \leftarrow \varphi(u, v_i) - \min\{-\partial\varphi(v_i), c_\varphi(v_i, u)\}$

Step 2-1 にもどる．

手続き MA-Ordering($\mathcal{N}_\varphi, s$) は，点 s から始めて，すでに選んだ点の集合 W への流出量が大きい順に点を選んでいき，選ぶ点がなくなれば終了する．文献[1]において提案された手続きと異なり，辺の向きを逆にたどることに注意する．

理論的には，次の補題から少なくとも改良前の計算量 $O(n(m+n\log n)\log nU)$ でおさえられることが示せた．

補題 3 1 Step 1-2 の実行後にプリフローの残存している点があったとする．$\tilde{\varphi}$ を Step 1-2 の実行前のプリフロー φ とし，$\bar{\varphi}$ を Step 1-2 の実行後に得られたプリフロー φ とする．このとき，$\bar{v}(\bar{\varphi}) - \bar{v}(\tilde{\varphi}) \geq (\bar{v}^* - \bar{v}(\tilde{\varphi}))/n$ が成り立つ．ただし，$\bar{v}^*$ は $\mathcal{N}$ における最大流量を表す．

4. 計算機実験の結果

改良したアルゴリズムをプログラム FMAP に実装した．プログラム FMA は改良前のアルゴリズムを実装している．プログラム HI_PR は Goldberg-Tarjan の最遠点選択基準を用いたアルゴリズムを実装しており，Goldberg のサイトからダウンロードした．

プログラムの入力となるネットワークを作るために，DIMACS Core Experiment で提示されているジェネレータ GENRMF を用いた．各サイズについて 5 種類のランダムな入力データを作り，その平均をとっている．

結果をまとめると，改良したプログラム (FMAP) は，すべてのデータにおいて改良前 (FMA) に比べて大きく実行時間が減っている．また，Goldberg-Tarjan のアルゴリズム (HI_PR) と比較すると，実用的に大きく近づいたことが分かる (図 1〜3)．

5. まとめ

本論文では，Fujishige のアルゴリズムに対する改良としてプリフローを用いたアルゴリズムを提案した．計算機実験の結果をみると，すべてのデータにおいて改良前より速くなっており，実用的に有意な改良であることが確認できた．さらに，Goldberg-Tarjan のアルゴリズムと比較したところ，実用的に大きく近づいたことが分かった．

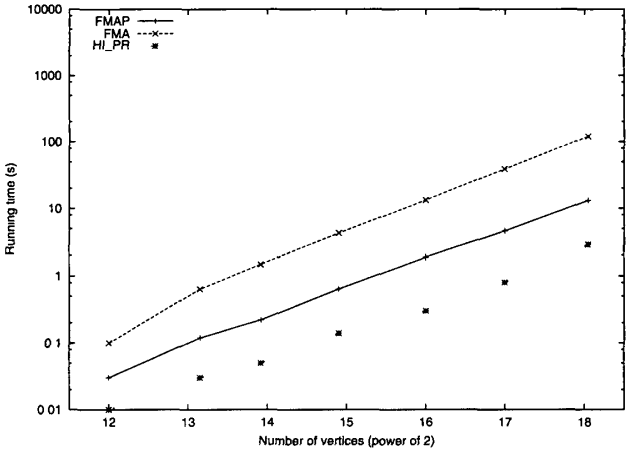

図 1 GENRMF-LONG family に対する結果

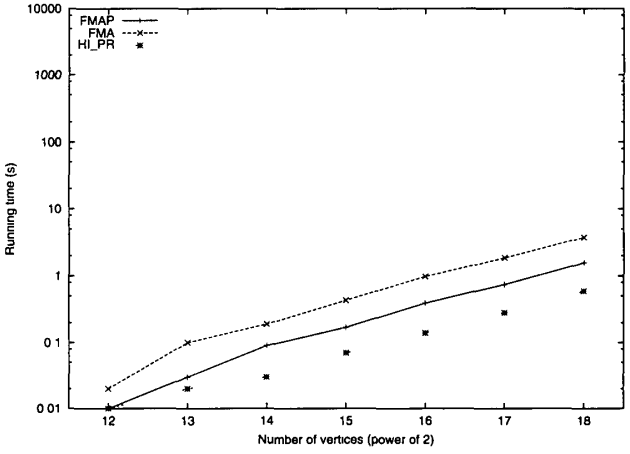

図 2 GENRMF-LONGER family に対する結果

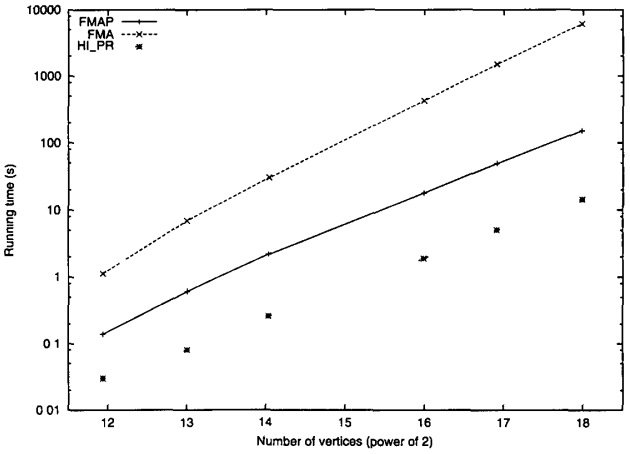

図 3 GENRMF-WIDE family に対する結果

参考文献

[1] S Fujishige. A maximum flow algorithm using MA orderings *Operations Research Letters*, vol 31, pp 176-178, 2003

© 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.