動的ネットワークフロー

神山　直之

Ford & Fulkerson によって体系化されたネットワークフロー理論は，そのモデルの持つ表現力が強力であるがゆえ，理論的に深く研究され，多くの現実問題に応用されてきた．しかし，このネットワークフローモデルにも１つの弱点がある．それは，時間の要素が欠けているということである．その欠点を補うために開発されたモデルが動的フローである．本稿では動的フローのモデルおよび基本的な結果を紹介する．

キーワード：動的フロー，最大動的フロー問題，最速フロー問題

1. 序論

ネットワーク上に何かしらのものを流す，という問題はオペレーションズ・リサーチの分野においては様々な場面で現れる．そのような問題を解決するための有用な数学的道具としてネットワークフロー理論がある．このネットワークフロー理論はFord & Fulkerson[8]によって体系化され，今日に至るまで多くの発展を遂げてきた．しかし，このネットワークフローモデルにも１つ弱点がある．それは時間的要素が考慮されていないことである．言うまでもなく，実際のネットワーク上の輸送問題等においては，ものがある地点から別の地点に移動する際にかかる時間というものが非常に重要な要素となる．この点を解決するために生み出されたものが**動的フロー**である．つまり，動的フローとは時間的要素をもったネットワークフローモデルである．今後，動的フローと区別するために，通常のネットワークフローを静的フローと呼ぶこととする．

実はこの動的フローは最近新しく生み出されたものではなく，すでにFord & Fulkerson[8]によって静的フローとともに紹介されていた．しかし，動的フローは静的フローと異なり時間的要素を持っているため，解析することが非常に困難であり，長い間大きな理論的進展が得られることはなかった．しかし，近年Hoppe & Tardos[11]による最速フロー問題に対する多項式アルゴリズムを中心として，FOCSやSODA，IPCOといったアルゴリズム・計算量理論に関する最難関国際会議においてもいくつかの成果が発表されるようになってきた[3]～[7][10][16][18]．このように，動的フローの理論的解析が進み，現実問題への応用のための土壌が整ってきつつある現状をかんがみ，本稿ではこの動的フローの普及のため，そのモデルおよび基本的な結果を紹介することを目的としている．

本稿は以下のように構成されている．まず第2節においては，動的フローのモデルを紹介する．そして続く２つの節においては，動的フローのモデルにおける代表的な２つの問題に関する結果を紹介する．まず第3節で扱う問題は**最大動的フロー問題**である．この問題は通常のネットワークフロー，つまり静的フローに対する最大フロー問題の動的フロー版である．具体的には，ある制限時間 T が与えられたとき，時刻 T までに可能な限り供給点から需要点へフローを流すことを目的とする問題である．一方，第4節で扱う問題は，前節の最大動的フロー問題のある意味双対とみなすことのできる**最速フロー問題**である．最速フロー問題においては，供給点に存在するサプライを，可能な限り早く需要点まで流すことが目標となる．そして，第5節では我々のグループがどのようにしてこの分野で最先端を目指しているかを軽く述べさせていただき，最後の第6節においては，本稿で取り上げることができない他の動的フローの問題を紹介し，本稿をまとめる．

2. 定義

本節では，動的フローのモデルの説明と形式的な定義を行う．そのために，まず静的フローと動的フローのモデルの違いを直感的に説明しよう．共通していることは，どちらも与えられたネットワーク上を何かしらのもの，例えば水のようなものが流れるモデルであるということである．しかし，静的フローにおいては

かみやま　なおゆき
中央大学 理工学部情報工学科
〒112-8551 文京区春日 1-13-27

ネットワーク上の各辺に対するフローの量は常に一定，つまり同じ量の水が常に流れ続けているような状況を表している（図 1）．一方，動的フローにおいては，ある時刻にある辺に入った水は一定のスピードで，その辺を移動し続ける（図 2）．このフローの時間的要素が静的フローと動的フローの決定的な違いである．

では，動的フローを形式的に定義しよう．そのためにまず**動的ネットワーク**を定義する必要がある（図 3 参照）．本稿では，$\mathbb{R}_+$ と $\mathbb{Z}_+$ を用いて非負の実数の集合と非負の整数の集合を表す．また，$D=(V,A)$ を点集合 V と辺集合 A で構成される有向グラフとする．このとき，**ネットワーク**とは点や辺にある値（例えば容量など）が与えられたり，V のいくつかの点が供給点や需要点として指定されているものである．動的ネットワークにおいては，各辺 $a \in A$ に容量 $u_a \in \mathbb{R}_+$ と移動時間 $\tau_a \in \mathbb{Z}_+$ が与えられている．また特定の点 $t \in V$ が需要点として与えられ，各点 $v \in V \backslash \{t\}$ に対してサプライ $b_v \in \mathbb{R}_+$ が与えられる．D はネットワークのトポロジーを表し，u_a は各辺の幅を，τ_a は各辺を移動するために必要な時間を，b_v は各点に存在するものの量を，そして t は目的地をモデル化している．また，一般性を失うことなく t を始点とする辺は存在しないとする．なぜならば，一度目的地にたどり着いたらそこから離れる必要がないためである．

このとき，動的フローとは関数 $f: A \times \mathbb{Z}_+ \to \mathbb{R}_+$ である（図 1 参照）．各辺 $a \in A$ と時刻 $\theta \in \mathbb{Z}_+$ に対して，$f(a,\theta)$ は時刻 θ に辺 a に流れ込むフローの量を表している．そして時刻 θ に辺 a に流れ込んだフローは時刻 $\theta+\tau(a)$ に a の終点に到着する．動的フロー f が，各辺 $a \in A$ と時刻 $\theta \in \mathbb{Z}_+$ に対して，

$$f(a,\theta) \leq u_a,$$

各時刻 $\Theta \in \mathbb{Z}_+$ と点 $v \in V \backslash \{t\}$ に対して

$$\sum_{a \in \delta_v} \sum_{\theta=0}^{\Theta} f(a,\theta) - \sum_{a \in \rho_v} \sum_{\theta=0}^{\Theta-\tau_a} f(a,\theta) \leq b_v,$$

ある時刻 $\Theta \in \mathbb{Z}_+$ に対して

$$\sum_{a \in \delta_t} \sum_{\theta=0}^{\Theta-\tau_a} f(a,\theta) = \sum_{v \in V \backslash \{t\}} b_v \tag{1}$$

を満たすとき実行可能であるという．ただし，各 $v \in V$ に対して δ_v と ρ_v はそれぞれ始点が v と終点が v であるような辺の集合を表している．1 つ目の条件は，ある時刻に辺に流れ込むフローの量が，容量を超えないことを意味している．また，2 つ目の条件はある時刻においてそれまでに流れ込んだフローの量と元々その点にあったサプライの合計以下しか流れ出すことができないことを意味している．ただし，この定義においてはサプライが点に滞留することを許していることに注意する．最後の条件はすべてのサプライが需要点に流れ込むことを意味している．また(1)を満たす最小の $\Theta \in \mathbb{Z}_+$ を f の**避難完了時間**とよぶ．

また，紙面の都合上詳しい説明は省くが，動的ネットワーク上の問題を静的ネットワーク上の問題に帰着するための手法として，Ford & Fulkerson[8] によって提案された**時間拡大ネットワーク**というものがある．この時間拡大ネットワークのサイズは，入力サイズの多項式で押さえることができないため，直接多項式時間アルゴリズムを導くことはないが，多くの動的フローの問題を静的フローの問題に帰着し，既存の手法で解くことが可能であるため，実用上は非常に有益である．興味を持たれた読者は Ford & Fulkerson[8] 等を参照していただきたい．

図 1　静的フローのイメージ，左の点から右の点へ常に水が流れ続けている．

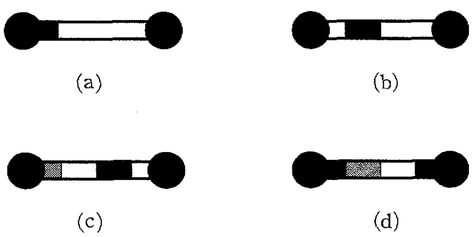

図 2　動的フローのイメージ．左の点から右の点への流れを考える．図は(a)，(b)，(c)，(d)の順に時刻が経過している．まず，(a)の段階で辺の中に存在する水が(b)の段階では少し前に進んでいる．そして，(c)の段階では一番最初に流れ込んだ水がさらに右に進み，さらに新しい水が辺に流れ込んでいる．(d)の段階では一番最初に流れ込んだ水が右の点に到達している．

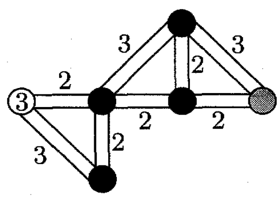

図 3　動的ネットワークの例．白い点にのみサプライ 3 があり，灰色の点が需要点となっている．各辺の容量は 1 で，各辺に付けられている数字は移動時間を表している．

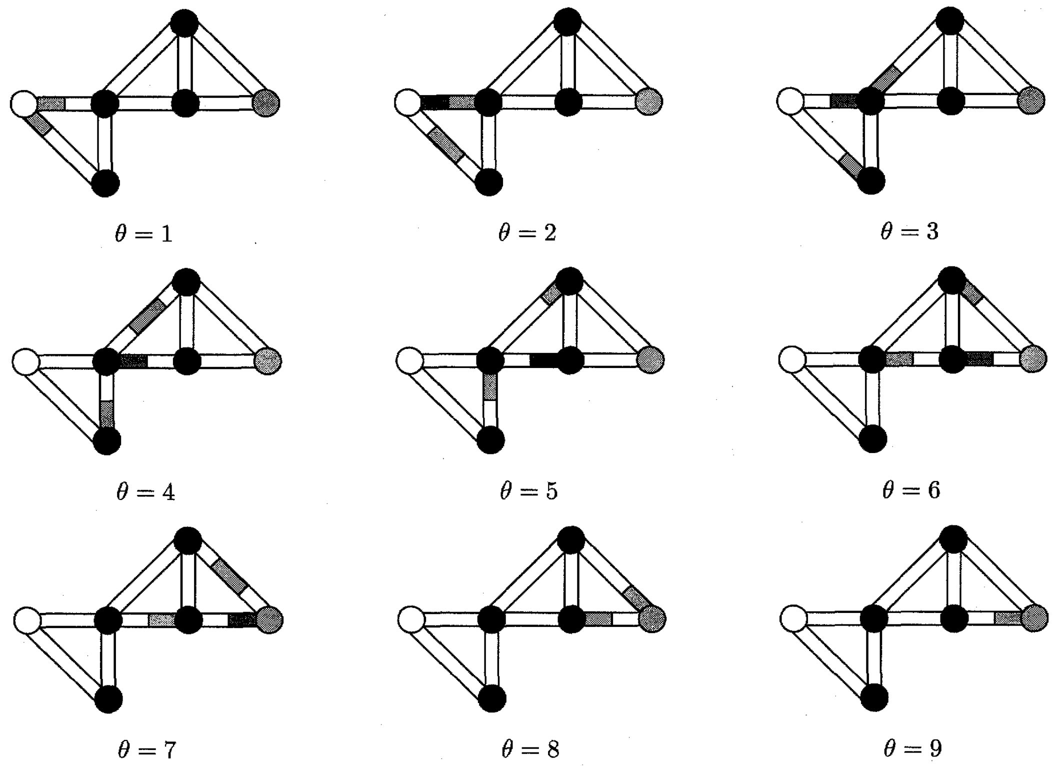

図4　図3の動的ネットワーク上の実行可能な動的フローの例．この動的フローの避難完了時間は9である．

3.　最大動的フロー問題

この節では最大動的フロー問題を扱う．最大動的フロー問題とは，ある制限時間 T が与えられたとき，時刻 T までに可能な限り供給点から需要点へフローを流すことを目的とする問題である．この問題を形式的に定義すると，ある供給点 $s\in V$ が与えられ，$b_s=\alpha$，そして各点 $v\in V\backslash\{s, t\}$ に対して $b_v=0$ となる動的ネットワークが与えられたとき，避難完了時間が T 以下の実行可能な動的フローが存在する最大の α を求める問題である．この最大動的フロー問題は，時間拡大ネットワーク上の静的な最大フロー問題へ帰着することで解くことができることが知られているが，前述の通り時間拡大ネットワークのサイズは入力のサイズの多項式で押さえることができないため，多項式時間アルゴリズムとはならない．そこで，Ford & Fulkerson[8]はこの問題を動的ネットワーク上の静的フロー（動的フローではない！）に関する最小費用流問題へと帰着することにより解決した．本節では，このアルゴリズムを紹介する．

Ford & Fulkerson[8]のアルゴリズムを紹介するためには，まず動的ネットワーク上の静的フローの定義を行わなければならばい．そのために，ひとまず動的ネットワーク上の移動時間関数 τ の存在を忘れていただきたい．このとき，静的フローとは辺集合 A から $\mathbb{R}_+$ への関数 $\xi: A\to\mathbb{R}_+$ である．静的フロー ξ が，各辺 $a\in A$ に対して

$$\xi(a)\leq u_a,$$

各点 $v\in V\backslash\{s, t\}$ に対して

$$\sum_{a\in\delta_v}\xi(a)-\sum_{a\in\rho_v}\xi(a)=0,$$

を満たすとき，ξ は実行可能であるという．

ξ を動的ネットワーク上の実行可能な静的フローとしたとき，$|\xi|$ で ξ の流量，つまり $\sum_{a\in\rho_t}\xi(a)$ を表すとする．ここで，静的フロー ξ と各辺 a に対して，辺 a の移動時間 τ_a を a の費用とみなそう．このとき，動的ネットワーク上の実行可能な静的フロー ξ の中で

$$(T+1)|\xi|-\sum_{a\in A}\tau_a\cdot\xi(a)$$

を最小化するものを ξ^* とする．このような ξ^* は多項式時間で見つけることができることが知られている[1]．Ford & Fulkerson[8]は ξ^* を用いて最大動的フロー問題に対する最適解が多項式時間で求めることができることを示した．

では，Ford & Fulkerson[8]は具体的にどのようにして ξ^* から最大動的フローを求めたのであろうか．そのために，まず ξ^* をパス分解する必要がある．ξ^* のパス分解とは，D 上の s から t までのパスの集合

$\mathcal{P}$と関数$\lambda:\mathcal{P}\to\mathbb{R}_+$の組であり，各辺$a\in A$に対して

$$\xi^*(a)=\sum_{P\in\mathcal{P}:a\in P}\lambda(P)$$

を満たすものである．このパス分解も多項式時間で求めることができることが知られている．このとき，ξ^*のパス分解から以下のようにして最大動的フローを求めることができる．各パス$P\in\mathcal{P}$上に時刻0から$T-\tau_P$まで$\lambda(P)$だけフローを流すような動的フローを考える．ただし，τ_PはP上のすべての辺の移動時間の合計である．このようにして構成された動的フローが実行可能であることは容易にわかる．Ford & Fulkerson[8]はこのようにして構成されたものが最大動的フローであることを時間拡大ネットワーク上の最大フロー最小カット定理[1]を用いることにより証明した．このアルゴリズムは，もちろん最大動的フロー問題を多項式時間で解くことができるという点で素晴らしいのだが，それだけではなく最大動的フローがある種のフローの繰り返しによって構成されている洞察も与えている点も非常に興味深い．

4. 最速フロー問題

続いて，この節では最速フロー問題を扱おう．最速フロー問題においては，すべてのサプライを，可能な限り早く需要点まで流すこと，つまり避難完了時間が最も小さい実行可能な動的フローを求めることが目標となる．最速フローの避難完了時間を最速避難完了時間と呼ぶ．

まず簡単な場合として，ある特定の点$s\in V$にのみサプライが存在する場合を考えよう．この場合，ある制限時間T以内にsのサプライb_sをtまで流せるか否かは，sからtへの制限時間Tに関する最大動的フローを求めれば判定することができる．つまり，供給点が1つの場合は2分探索を用いることにより，最速フロー問題は多項式時間で解くことができる．

では供給点が2つ以上ある場合はどうなるであろう．この場合は供給点が一つの場合のように単純に最大動的フロー問題へと帰着することはできない．この問題に対してHoppe & Tardos[11]は以下のようにして多項式時間アルゴリズムを与えた．まずKlinzによる重要な定理を紹介しよう[1]．各点集合$S\subseteq V\setminus\{t\}$と時刻$\theta\in\mathbb{Z}_+$に対して，$o_S^\theta$でサプライを無視し，$S$を供給点の集合とみなしたときの$\theta$に関する最大動的フローの流量を表すとする．$S$の点を1点に縮約することによりこの問題は最大動的フロー問題へと帰着することができるので，o_S^θは多項式時間で求めることができる．このときKlinzは最速避難完了時間がθ以下である必要十分条件は，すべての$S\subseteq V\setminus\{t\}$に対して，

$$o_S^\theta\geq\sum_{v\in S}b_v \tag{2}$$

が成り立つことであることを証明した．もしこの定理を使い，最速避難完了時間がθ以下であるか否かを多項式時間で判定することができれば，最速避難完了時間は2分探索を用いることにより多項式時間で求めることができる．では，あるθを固定したとき，すべてのSに対して(2)が成り立つか否かはどのように判定すればよいであろうか．ここで各$S\subseteq V\setminus\{t\}$に対して

$$g(S)=o_S^\theta-\sum_{v\in S}b_v$$

と定義される関数gを考えよう．もしgの最小値が0以上であれば(2)がすべてのSに関して成り立つことがわかる．では，gの最小値が0以上であるか否かは多項式時間で判定することができるのであろうか．実はできるのである．このためのgの重要な性質として，各$X, Y\subseteq V\setminus\{t\}$に対して

$$g(X)+g(Y)\geq g(X\cup Y)+g(X\cap Y)$$

が成り立つことが容易にわかる．この性質は劣モジュラ性といわれ，劣モジュラ性を持つ関数の最小値は多項式回の関数値の評価で求めることができることが知られている[12][21]．よってこの事実より最速避難完了時間を多項式時間で求めることができることがわかった．では，最速フローそのものはどのようにして求めればよいのであろうか．この問題をHoppe & Tardos[11]は辞書式最大動的フロー問題というものへ帰着することで解決しているのだが，具体的な内容に関しては本稿の範疇を超えているために割愛させていただく．

本節の最後に，最速フロー問題をさらに一般化した**普遍的最速フロー問題**を紹介しよう．最速フロー問題とは，いわば最後に避難する人の時刻を最小にしようとする問題であった．しかし，例えば大規模災害時等において建物からの避難を考えると，ただすべての人を可能な限り早く避難させるだけではなく，さらに任意の時刻において可能な限り多くの人が避難すること

[1] この結果は最速フロー問題において非常に重要な結果なのだが，論文にはなっておらずHoppe & Tardos[11]の中で“personal communication”として引用されているのみである．

ができる方が好ましい．このような制約を最速フロー問題に課したものが，普遍的最速フロー問題である．この問題は形式的には以下のように定義される．$\mathcal{F}$で最速フローの集合を表す．このとき，すべての$f\in\mathcal{F}$と$\Theta\in\mathbb{Z}_+$に対して，

$$\sum_{a\in\rho_t}\sum_{\theta=0}^{\Theta-\tau_a} f(a,\theta)\leq\sum_{a\in\rho_t}\sum_{\theta=0}^{\Theta-\tau_a} f^*(a,\theta)$$

を満たす実行可能な動的フローf^*を求める問題である．このような動的フローが存在するか否かは自明ではないのだが，Minieka[20]は任意の動的ネットワークに対して普遍的最速フローが存在することを証明した．しかし，普遍的最速フローを効率的に求めるアルゴリズムの存在は長い間未解決であったのだが，Baumann & Skutell[3]によって肯定的に解決された．

5．最先端を目指して

ではこの分野で我々のグループはどのようにして最先端を目指そうとしているのか．細かいところには目をつぶり，ここでは雰囲気を感じとっていただきたい．

動的フローの問題において，最大動的フロー問題はFord & Fulkerson[8]によって動的フローのモデルが提唱された段階で，多項式時間アルゴリズムが提案されていたので，中心的な問題は最速フロー問題が多項式時間で解くことができるかどうかという問いであった．この未解決問題は長い間解かれることはなかったのだが，Hoppe & Tardos[11]によって肯定的に解決された．しかし，このアルゴリズムは最速フロー問題を劣モジュラ最小化[12][21]という別の問題に帰着している．そこで，最速フロー問題の多項式時間可解性が明らかとなった今，次なる挑戦は劣モジュラ関数最小化に帰着しないネットワークベースの多項式時間アルゴリズムが存在するかどうかという問題となった．現在のところ，この問題の解決への糸口は全くないというのが正直なところであるが，我々のグループでは時間拡大ネットワークの縮小[9]という手法が有益ではないかと考えている．時間拡大ネットワークとはFord & Fulkerson[8]によって提案されたもので，動的フローの問題を静的フローの問題へと帰着することができるものであるが，そのサイズが入力サイズの多項式で抑えられないため直接多項式時間アルゴリズムを導くことができなかった．しかし，入力のネットワークがある条件を満たす際にはそのサイズを多項式サイズに縮小することができ，最速フロー問題に対する多項式時間アルゴリズムが開発できることが知られている[9][13]～[15]．我々のグループでは，この手法が適用できる範囲を広げることによりネットワークベースの最速フロー問題への多項式時間アルゴリズムが提案できないかと模索している．

6．結論

本稿では，動的フローに関する代表的な問題である最大動的フロー問題と最速フロー問題を扱ったが，ここでは扱うことのできなかった問題をいくつか紹介しよう．通常の静的ネットワークにおける問題として最大フロー問題と並ぶ重要な問題の一つとして最小費用フロー問題がある．もちろん，この最小費用フロー問題の動的フロー版も研究されているのだが，実は最大動的フロー問題が多項式時間で解くことができるのとは異なり，この最小費用動的フロー問題はNP困難であることが知られている．この問題に関しては文献[7][16]を参照していただきたい．静的フローとの関係という点では，多品種フローの動的フロー版も考えることも自然であろう．もちろん，静的ネットワークにおいても多品種フローは非常に困難な問題であるため，その動的フロー版はさらに困難となることは容易に想像できるであろう．この問題に関しては文献[6][9]を参照していただきたい．また，静的フローとゲーム理論の重要な融合として均衡フローというものがある．この均衡フローとはネットワーク上を移動するものが，それぞれ自分勝手に動いたらどのような状態になるかを解析するためのものであり，静的ネットワーク上では非常に多く研究されてきた．近年，この枠組みを動的フローにも拡張しようとする試みがなされており[2][17]，まだ萌芽的な段階であることは否めないが，発展を続ければ非常に興味深いモデルが完成するのではないかと思われる．動的フローのモデル自体の興味深い変種として，Melkonian[19]によって提案されたモデルにおいては，辺の容量がある時刻に辺に入る人数を制限するのではなく，ある時刻に同時に辺上に存在することのできる人数を制限するものとなっているようなものがある．

動的フローはその理論面での奥深さと応用面でのモデルのもつ表現力の強力さにもかかわらず，現状では残念ながらそれほどメジャーであるとは言い切れない．私としては，本稿を読んでくださってこの分野に少しでも興味を持っていただいたら幸いである．

参考文献

[1] R.K. Ahuja, T.L. Magnanti and J.B. Orlin, *Network Flows-Theory, Algorithm, and Applications*, Prentice Hall, 1993.

[2] E. Anshelevich and S. Ukkusuri, Equilibria in dynamic selfish routing, In *SAGT*, 171-182, 2009.

[3] N. Baumann and M. Skutella, Solving evacuation problems efficiently-earliest arrival flows with multiple sources, In *FOCS*, 399-410, 2006.

[4] L. Fleischer, Faster algorithms for the quickest transshipment problem with zero transit times, In *SODA*, 147-156, 1998.

[5] L. Fleischer, Universally maximum flow with piecewise-constant capacities., In *IPCO*, 151-165, 1999.

[6] L. Fleischer and M. Skutella, The quickest multicommodity flow problem, In *IPCO*, 36-53, 2002.

[7] L. Fleischer and M. Skutella, Minimum cost flows over time without intermediate storage, In *SODA*, 66-75, 2003.

[8] L.R. Ford Jr and D.R. Fulkerson, *Flows in Networks*, Princeton University Press, 1962.

[9] A. Hall, S. Hippler and M. Skutella, Multicommodity flows over time: Efficient algorithms and complexity, *Theoretical Computer Science*, 379 (3) : 387-404, 2007.

[10] B. Hoppe and É. Tardos, Polynomial time algorithms for some evacuation problems, In *SODA*, 433-441, 1994.

[11] B. Hoppe and É. Tardos, The quickest transshipment problem, In *SODA*, 512-521, 1995.

[12] S. Iwata, L. Fleischer and S. Fujishige, A combinatorial strongly polynomial algorithm for minimizing submodular functions, *J. ACM*, 48 (4): 761-777, 2001.

[13] N. Kamiyama and N. Katoh, A polynomial-time algorithm for the universally quickest transshipment problem in a certain class of dynamic networks with uniform path-lengths, In *ISAAC*, 802-811, 2009.

[14] N. Kamiyama, N. Katoh and A. Takizawa, An efficient algorithm for evacuation problem in dynamic network flows with uniform arc capacity, *IEICE Transactions*, 89-D (8): 2372-2379, 2006.

[15] N. Kamiyama, N. Katoh and A. Takizawa, An efficient algorithm for the evacuation problem in a certain class of networks with uniform path-lengths, *Discrete Applied Mathematics*, 157 (17) : 3665-3677, 2009.

[16] B. Klinz and G.J. Woeginger, Minimum cost dynamic flows: The series-parallel case, In *IPCO*, 329-343, 1995.

[17] R. Koch and M. Skutella, Nash equilibria and the price of anarchy for flows over time, In *SAGT*, 323-334, 2009.

[18] E. Köhler and M. Skutella, Flows over time with load-dependent transit times, In *SODA*, 174-183, 2002.

[19] V. Melkonian, Flows in dynamic networks with aggregate arc capacities, *Inf. Process. Lett.*, 101 (1): 30-35, 2007.

[20] E. Minieka, Maximal, lexicographic, and dynamic network flows, *Operations Research*, 21: 517-527, 1973.

[21] A. Schrijver, A combinatorial algorithm minimizing submodular functions in strongly polynomial time, *J. Comb. Theory, Ser. B*, 80 (2): 346-355, 2000.