日本オペレーションズ・リサーチ学会
2004年秋季研究発表会

木構造動的ネットワークにおける複数個の施設配置問題

02602474　大阪大学　＊間々田 聡子　＊MAMADA Satoko
01013640　国立情報学研究所　宇野 毅明　UNO Takeaki
01605984　大阪大学　牧野 和久　MAKINO Kazuhisa
01502254　京都大学　藤重 悟　FUJISHIGE Satoru

1 序論

動的ネットワーク $\mathcal{N}$ は，各枝 $a \in A$ が容量 $u(a)$ と移動時間 $\tau(a)$ をもつ有向グラフ $G=(V,A)$ によって定義される．動的ネットワーク中のいくつかの供給点から需要点へ定まった量を最短時間で送り出す問題については，B. Hoppe–É. Tardos [1] によって多項式時間アルゴリズムが提案されているが，それは高次の多項式時間アルゴリズムである．そこで我々は [3] において，対象となるネットワークが木構造であり，各点に与えられた供給量を最速に送り出すような最適な出口を見出す問題を考え，$\mathrm{O}(n\log^2 n)$ 時間アルゴリズムを提案した (ただし，n は点数である)．また，[2] においては，対象となるネットワークが木構造で，複数個の出口が与えられているときに各点に与えられた供給量を最速に送り出すようなフローを求める問題に対して，$\mathrm{O}(kn^2\log^2 n)$ 時間アルゴリズムを提案した．ただし，各点を経由するフローは，その点に接続する同一の枝を通ると仮定している．この仮定は，木構造動的ネットワークとして実際の交通網をみたとき，その災害時において住民 (各点の供給量に相当する) を公平に，かつ混乱なく避難誘導を行うためのひとつの解決策として妥当なものと考えることが出来る．

本研究では，与えられた供給量 $d(v)$ $(v \in V\setminus S)$ を (1) 指定された時間 Θ 以内に送り出すことのできる最小個数の出口を配置する問題と (2) 指定された個数 k の出口を，最速にフローを送り出すことができるように配置する問題を考える．ただし，本研究でも，上述と同様の仮定をする．このとき，我々は (1) を $\mathrm{O}(n^2\log^2 n)$ 時間で，さらに (2) を $\mathrm{O}(k^3n^3\log^2 n)$ 時間で解けることを示す．

2 問題の定義とアルゴリズム

木構造の動的ネットワーク $\mathcal{N}=(T=(V,A),u,\tau,d)$ を考え，V は点集合，A は枝集合とする．このとき，$u: A\to \mathbf{R}_+$ は各枝に非負の実数を与える容量関数，$\tau: A\to\mathbf{R}_+$ は各枝の一方の端点から他方の端点への移動時間を与える関数であり，さらに，供給量 $d: V\to\mathbf{R}_+$ が与えられている．各枝 $a\in A$ と任意の $\theta\in\mathbf{R}_+$ に対して，$f_a(\theta)$ を枝 a に時刻 θ で入り込み，時刻 $\theta+\tau(a)$ に枝 a の終点に到着する (単位時間当たりの) フロー量とし，$f_a(\theta)$ $(a\in A,\ \theta\in\mathbf{R}_+)$ が (1) 容量制約, (2) 流量保存則, (3) 供給量制約を満たすとき，連続時間動的フローと呼ぶ．ただし，各点でフローの滞留は可能であるとする．

本研究では，各点 v を経由するフローは，v を始点とする同一の枝を通ると仮定する．また，T をある点 r を根とする根付き木とみなす．このとき，$T_v=(V_{T_v},A_{T_v})$ を v を根とする T の部分木とし，$V_{T_v}\subseteq V$，$A_{T_v}\subseteq A$ は，それぞれ T_v の点集合，枝集合である．ここでは簡単のため，V_{T_v} を T_v と表記する．

2.1 Θ 時間以内の避難を可能にする施設配置問題

本節では，与えられた供給量 $d(v)$ $(v\in V\setminus S)$ を (指定された) 時間 Θ 以内に送り込むことのできる最小個数の出口を配置する問題に対する多項式時間アルゴリズムを与える．

このアルゴリズム中の S は，アルゴリズム実行中の任意の時点で，それまでに確定した出口集合を示し，アルゴリズム終了時には最適解を与える．また，S_{tmp} は T_v に限定した施設配置問題において現在の S に加えて v を施設としたときに，Θ 以内で避難可能であるような v の集合である．W は，現在の S のなかで，自分の子孫である点で要求が満たされることの確定している点集合である．また，$L(T)$ は T の葉集合，$T[X]$ は X によって誘導される T の部分木，T^W はどんな $w\in W$ の子孫ではない T の点集合，P_{sv} は s と v とを結ぶパスの点集合とする．

我々のアルゴリズムは，直感的には点 v に対する部分木 T_v の完了時間をしらべ，Θ 以内であれば，さらに内側へフローを送れる可能性があるので，$v\in S_{tmp}$ とする．Θ より大きければ，$v\in S$，すなわち v を出口とする．この操作を葉から順に内側へ向かって繰り返す．

Algorithm LOCATION
Input: 木構造動的ネットワーク $\mathcal{N}=(T=(V,E),u,\tau,d)$, と完了時間 Θ.
Output: 最小個数の出口集合 S.

Step 0: $S = S' = W := \emptyset$, $S_{tmp} := L(T)$, $X := V \setminus L(T)$ とする.

Step 1: 点 $v \in L(T[X])$ の子のうち, S_{tmp} に含まれる点 w に対して, 次の操作を行う.

$S_{tmp} := S_{tmp} \setminus \{w\}$ とし, $T' := (T_w \cup \{v\})^W$ に対して, v を出口としたフロー問題を解く.

$C(T') > \Theta$ ならば, $S := S \cup \{w\}$, $W := W \cup \{w\}$, $S' := S' \cup \{w\}$.

Step 2: 各 $s \in S' \cap T_v$ に対して, $T' := T_v^{W \setminus P_{sv}}$ で, s を出口としてフロー問題を解く.

$C(T') \leq \Theta$ ならば, $W := W \cup \{v\}$ とし, Step 4 へ.

そうでなければ, $S' := S' \setminus \{s\}$.

Step 3: v が根でないならば, $S_{tmp} := S_{tmp} \cup \{v\}$. そうでなければ, $S := S \cup \{v\}$.

Step 4: $X := X \setminus \{v\}$ として, $X \neq \emptyset$ ならば, Step 1 へもどる. $X = \emptyset$ ならば, S を出力して終了. □

ここで, 木 T と T 中の点 v が与えられたとき, T において最速に供給量を v に送り出す問題をフロー問題と呼び, その完了時間を $C(T)$ と記す.

紙面の都合上アルゴリズムの正当性, 計算時間の詳細は省略するが, 直感的には, Step 1 と Step 2 において, [3] で提案されたフロー問題に対する $O(n \log^2 n)$ 時間アルゴリズムを $O(n)$ 回用いることにより, 最適出口集合を求めることができる.

定理 1 木構造の動的ネットワークにおける施設配置問題は $O(n^2 \log^2 n)$ 時間で解くことができる. □

2.2 最速に避難可能な k 個の施設の配置問題

本節では, 指定された出口数 k に対して, 与えられた供給量 $d(v)$ $(v \in V \setminus S)$ を最速に送り出すことができるような k 個の出口の配置問題を考える. このアルゴリズムでは, まず, 2.1 節で述べた完了時間が指定されたときに最小個数の出口集合を求めるアルゴリズム (以下, アルゴリズム Location と呼ぶ) を利用して, 各時間での施設配置を求める. そして, 出口集合が与えられたときに最小な完了時間を求める [2] の $O(kn^2 \log^2 n)$ 時間アルゴリズム (以下, アルゴリズム Partition と呼ぶ) により, その出口集合での最小完了時間を求め, その求めた時間をわずかに小さくして, さらにアルゴリズム Location を適用し, 出口数が k より大きくなるまで続ける.

Algorithm k-Location

Input: 木構造動的ネットワーク $\mathcal{N} = (T = (V, E), u, \tau, d)$, 施設数 $k \in \mathbf{Z}_+$, 十分小さい正数 ϵ.

Output: 出口集合 S.

Step 0: $S := \emptyset$, $\Theta := M$ とする. ただし, M は十分大きな正数.

Step 1: アルゴリズム Location を使って, Θ のときの出口集合 S^* を求める. このとき, $|S^*| > k$ ならば, S を出力して終了.

Step 2: アルゴリズム Partition を使って, S^* に対する Θ^* を求める. $\Theta := \Theta^* - \epsilon$, $S := S^*$ として, Step 1 へ. □

このアルゴリズムの正当性も紙面の都合上省略する. 計算時間の評価の要点は次の通りである. Step 2 で求めた Θ を微少量小さくして, アルゴリズム Location に適用したとしても, 必ずしも施設数が増えるとは限らない. しかし, 高々(施設数 $\times n$) 回繰り返してやれば, 施設数は少なくとも 1 増加する. したがって, アルゴリズム Location と Partition を高々$k^2 n$ 回繰り返せば良い.

定理 2 木構造の動的ネットワーク上の k 施設配置問題は $O(k^3 n^3 \log^2 n)$ 時間で解くことができる. □

3 結論と今後の課題

我々は (1) Θ 時間以内の避難を可能にする施設配置問題が $O(n^2 \log^2 n)$ 時間で, (2) 最速に避難可能な k 個の施設の配置問題が $O(k^3 n^3 \log^2 n)$ 時間で解けることを示した (ただし, n は点数).

さらに, この問題の他に, より一般のネットワークを想定した問題, などが今後の課題として挙げられる.

謝辞 本研究は文部科学省の学術創成研究費 (課題番号 13GS0018) と笹川科学研究助成の支援による.

参考文献

[1] B. Hoppe and É. Tardos: The Quickest Transshipment Problem, *Mathematics of Operations Research*, **25** (2000) 36–62.

[2] 間々田聡子, 牧野和久, 藤重悟: 木構造動的ネットワークにおける複数の施設への避難誘導問題, 日本オペレーションズ・リサーチ学会 2004 年春季研究発表会アブストラクト集, (2004) 18–19.

[3] S. Mamada, T. Uno, K. Makino and S. Fujishige: An $O(n \log^2 n)$ Algorithm for a Sink Location Problem in Dynamic Tree Networks, *3rd IFIP International Conference on Theoretical Computer Science* (August 23–26, 2004, Toulouse, France).