【土木計画学研究・講演集 No.24 (2) 2001年11月】

予算制約をもつネットワークデザイン問題の貪欲解法*

Greedy Algorithms for the Network Design Problem with a Budget Constraint*

片山 直登**・百合本 茂***
By Naoto KATAYAMA**・Shigeru YURIMOTO***

1. はじめに

予算制約をもつネットワークデザイン問題(*BND*)は，与えられたデザイン費用の予算の範囲内で，ルーティング費用を最小にするようなアークを選択し，ネットワークの形状を定める問題である．この問題は，道路ネットワーク設計，輸送路線計画や通信ネットワーク設計などに現れる基礎的な問題である．

*BND*に対して，多くの厳密解法，緩和問題や近似解法が提案されている．Gallo[1]，Ahuja-Murty[2]などは下界平面とよばれる下界関数を用いたナップサック型の緩和問題を示した．片山直登-春日井[3]は，Lagrange緩和問題とその解法を示した．Scott[4]，西村昂-日野 泰雄[5]や森津秀夫[6]は，貪欲解法であるフォワード法やバックワード法とその改良法などを提案した．一方，固定費用をもつネットワークデザイン問題(*FND*)に対して，Minoux[7]や片山[8]は高速な貪欲解法を示した．

*BND*は混合整数計画問題となるため，ノード数が少ない小規模な問題であれば一般の汎用パッケージで最適解が得られる可能性がある．しかし，ネットワークが密の場合では，ノード数が30程度であっても，0-1変数であるアーク変数が約400個，フロー変数は数十万個となり，一般的な汎用パッケージを用いてそのまま解くことは困難である．一方，Lagrange緩和法では良い下界値を求めることができるが，ノード数が100，アーク数が5000程度を超えると膨大な計算時間が必要となる．したがって，規模の大きな問題では，実行可能解を短時間で求める近似解法が必要となる．

本研究では，*FND*に対するMinoux法とその改良法を*BND*に適用した貪欲解法を提案する．

2. 問題の定式化

需要の発生・集中点，施設，集線装置などのように点で表すことのできるものをノード，道路，路線，通信回線などのように点と点を結ぶ線をアークで表す．車両，貨物，通信などのように，ネットワーク上には決められたノード間を移動するものが存在する．ここでは同一の始点・終点をもつものを同一の品種と定義する．単位時間当たりの移動量を需要量とよぶ．

アークには，デザイン費用とルーティング費用が与えられている．デザイン費用は，道路建設費用，回線設置などのようにアークを設置・選択するときに発生する固定費用を表す．ルーティング費用は，走行費用(時間)，輸送費用(時間)，通信費用(時間)などのようなフロー量に依存する変動費用(時間)を表す．また，使用できるデザイン費用に上限が与えられており，これを予算とよぶ．

ノード集合をN，アーク候補集合をA，選択するアーク集合をA'，品種の集合をKとする．ネットワーク$G(N,A')$上のルーティング費用の総和を$\phi(A')$，アーク(i,j)上の品種kのフロー量を表すアークフロー変数をx_{ij}^kとする．アーク(i,j)を選択することによって発生するデザイン費用をf_{ij}，品種kの単位当たりのルーティング費用をc_{ij}^k，デザイン費用のための予算をBとする．品種kの需要量をd^kとし，ノードnが品種kの始点であれば$-d^k$，終点であればd^k，それ以外では0である定数をd_n^kと表す．また，$N^{(n)}=\{j\in N|(j,n)\in A'\}$，$N^{+(n)}=\{j\in N|(n,j)\in A'\}$とおく．

*キーワード：交通網計画，道路計画
**正員，工修，流通経済大学流通情報学部
(茨城県竜ケ崎市平畑120
TEL0297-64-001，FAX0297-64-0011)
***正員，工修，流通経済大学経済学部

*BND*をアーク候補集合*A*から*A'*を選択する問題とみると，*BND* は次の2段階の最適化問題として定式化することができる．

$$min \quad _{A' \in A} \phi(A') \tag{1}$$

$$st \sum_{(i,j) \in A'} f_{ij} \le B \tag{2}$$

$$\phi(A') = min \sum_{(i,j) \in A'} \left(c_{ij}^k x_{ij}^k + c_{ji}^k x_{ji}^k \right) \tag{3}$$

$$st \sum_{i \in N^-(n)} x_{in}^k - \sum_{i \in N^+(n)} x_{nj}^k = d_n^k \quad {}^\forall n \in N, k \in K \tag{4}$$

$$0 \le x_{ij}^k \le d^k, 0 \le x_{ji}^k \le d^k \quad {}^\forall (i,j) \in A', k \in K \tag{5}$$

(1)式は，ルーティング費用の総和を最小化するアーク集合*A'*上を選択することを表す．(2)式は選択したアークのデザイン費用の合計が予算以下であることを表す．(3)式はルーティング費用の総和の最小化を表す．(4)式は，始点から出たフローが必ず終点に到着することを表すフロー保存式である．(5)式は，フローが非負で需要量以下となる条件である．

2．従来の貪欲解法

基本的な貪欲解法として，次のようなバックワード法がある．すべてのアークを含む最大ネットワークを初期解とする．各アークをネットワークから削除したときの総ルーティング費用の増加量を計算する．この増加費用が最小であるアークを求め，このアークをネットワークから削除する．デザイン費用の合計が予算以下になるまで，この操作を繰り返す．最後に，残余予算に対するフォワード法を実施する．

バックワード法は良い解を算出する貪欲解法ではあるが，アークを削除したネットワーク上における最短路問題を繰り返し解くことが必要となる．Murchland法[9]を用いると，アークを削除した全ノード間の最短路を高速に求めることができる．しかし，この場合でもバックワード法の計算量は$O(|A|^2|N|^3)$であり，密で規模の大きな問題に適用することは困難が伴う．

一方，*FND*に対して，Minouxは高速な貪欲解法を示している．*FND*と*BND*の相違点はデザイン費用を目的関数に含むか予算として制約条件にもつかの違いだけであるので，*FND*に対するMinoux法を*BND*に適用すれば，効果的な解法となることが期待できる．

*FND*に対するMinoux法を示しておく．各アーク*(i,j)*に対して，このアークを削除したときのノード*i*, *j*間の最小ルーティング費用パスに，現在アーク*(i,j)*上を流れているフローを流し代えたときの目的関数の減少量を求め，これをアークを削除したときの目的関数値を減少量の評価値として，バックワードタイプの解法を行う．ただし，この評価値の大きい順にアークの削除リストを作成しておき，実際に削除の順になったときに，減少量の評価値を再計算する．これがリスト中で最大であるときに限りアークを削除し，そうでなければ評価値にしたがって，このアークを削除リストの適切な位置に格納する．減少量の評価値が正である間，この操作を繰り返す．

3．*BND*の貪欲解法

フローを最短(費用)路に流せば，*BND*の実行可能性は，ネットワークがアークによって連結し，かつデザイン費用が予算以下であることで判定できる．*FND*とは異なり，アークを取り除いたときに総ルーティング費用は増加(非減少)するため，総ルーティング費用の増加量を評価する必要がある．そこで，この値の小さい順の削除リストを作成し，予算以下なるまで，この順にアークの削除を検討すれば良い．

あるアーク*(i,j)*を削除したときのノード*i*, *j*間の最小ルーティング費用パスに，現在アーク*(i,j)*上を流れているフローを流し代えたときの総ルーティング費用の増加量ϕ_{ij}を求め，これをアーク*(i,j)*を削除したときの総ルーティング費用の増加量の評価値とする．

*BND*に対する貪欲解法のアルゴリズムを示す．

貪欲解法1

[1] $G(N,A)$を初期ネットワークとし，多品種の最短(費用)路問題を解く．$A' := A$とする．

[2] ${}^\forall (i,j) \in A'$について，$G(N,A')$上における評価値ϕ_{ij}を求める．

[3] 評価値の小さい順に，該当するアークと増加量の評価値をリストに格納する．

[4] リスト内の評価値が最小のアーク(i^*,j^*)を選ぶ．

[5] アーク(i^*,j^*)に対して，$G(N,A')$上における評価値$\phi_{i^*j^*}$を再計算する．この値がリスト内で最小であれば[6]へ行く．そうでなければ，評価値ϕ_{ij}にしたがって，このアークと評価値をリスト内の適切な位置に移動し，[4]へ戻る．

[6] アーク(i^*,j^*)と評価値$\phi_{i^*j^*}$をリストから削除し，$A':=A'\setminus(i^*,j^*)$とする．$G(N,A')$に含まれているアークのデザイン費用の合計が予算以下になれば[7]へ，そうでなければ[4]へ戻る．

[7] 残余予算に対するフォワード法を実施し，終了する．

[7]を除けば，この貪欲解法ではアーク(i,j)が実際に削除の対象となる場合に限って増加量の近似的な評価値を計算するため，計算量は$O(|A|^2|N|^2)$であるが，平均的には$O(|A||N|^2)$である．また，[7]では，数本のアークの付加を行うだけであるので，計算時間への影響は少ない．

貪欲解法1では，Minoux法と同様に総ルーティング費用の増加量の評価値を近似値として求めている．一方，1本のアークが削除されたネットワーク上の最短路問題は，Murchland法を用いると$O(|N|^2)$で求めることができる．したがって，Murchland法を用いてアークを削除したときの総ルーティング費用の増加量を厳密に評価しても，計算量は同じである．

貪欲解法1における総ルーティング費用の増加量を厳密に評価するように改良した方法を貪欲解法2とする．

4．数値実験

数値実験によって，Scottのバックワード法(Scott法)，提案した貪欲解法1および貪欲解法2を比較する．100×100の平面上にランダムにノードを発生し，ノード間のユークリッド距離を整数化した値をデザイン費用とする．ルーティング費用はすべての品種で同一とし，デザイン費用に比例するものとする．予算制約は，デザイン費用をアークの重みとした最小木のデザイン費用の2～6倍に設定する．すべての2ノード対間のアークを集合Aとし，すべての2ノード対間に品種が流れ，すべての需要量を同一とする．

ノード数は10から100までとし，同一ノード数では各10組のデータを使用する．使用計算機・言語はIBM PC互換機，CPU Pentium1.7GHz，メモリ256Mb，OS Windows2000，Microsoft Fortran Power Station Ver.4である．近似解の誤差を定めるために，Lagrange緩和法[3]を用いて下界値を算出する．

各解法を用いて，“予算=最小木の2倍”の場合に得られた近似解の目的関数値の平均誤差と1問当りの平均計算時間を表1に示す．誤差は，“100×(近似解の目的関数値-Lagrange緩和法による下界値)/Lagrange緩和法による下界値”の平均値である．“予算=最小木の4倍”の結果を表2に，“予算=最小木の6倍”の結果を表3に示す．また，貪欲解法1を用いて，予算=最小木の2倍，100ノード，4950アークのあるデータに対して求めた解のネットワークを図1に示す．

貪欲解法1・2において，“予算=最小木の2倍”では誤差が最大2.52%，“予算=最小木の4倍”では最大0.53%，“最小木の6倍”では最大0.21%となっている．また，問題の規模にしたがって，誤差は増加している．貪欲解法1と2では，誤差に顕著な差は見られない．Scott法と比べると，貪欲解法1・2は，予算＝最小木の2倍のノード数60で約0.3%，それ以外では0.1%程度の悪い解となっている．

計算時間において，Scott法はノード数100を超えると膨大な時間を必要とし，さらに大規模な問題に対してそのまま適用することは困難である．一方，貪欲解法1や貪欲解法2では，ノード数100においても2.5秒以内で近似解を算出しており，大規模な問題に対しても有効な解法であることが分かる．また，貪欲解法2の方が，貪欲解法1よりも10～20%程度計算時間は短くなっている．これは，厳密な目的関数値の増加量を求めることによって，アルゴリズムの[5]においてリスト中で最小となることが多くなり，再計算の回数が減少するためと考えられる．

5．おわりに

本研究では，予算制約をもつネットワークデザイン問題に対する2種類の貪欲解法を提案した．提案した解法では，予算が最小木の2倍の場合は2.5%以下，予算が最小木の4倍では0.5%以下，最小木の6倍では0.2%以下の解を求めることができ，Scott

表-1　近似解の比較：予算=最小木の2倍

ノード数	Scott法		貪欲解法1		貪欲解法2	
	誤差(%)	計算時間(s)	誤差(%)	計算時間(s)	誤差(%)	計算時間(s)
10	0.22	0.0	0.30	0.0	0.30	0.0
20	0.49	0.1	0.53	0.0	0.53	0.0
30	0.69	1.1	0.64	0.0	0.64	0.0
40	0.98	6.3	1.10	0.0	1.09	0.1
50	1.48	23.7	1.53	0.2	1.53	0.2
60	1.34	69.7	1.66	0.4	1.62	0.3
70	1.75	175.5	1.80	0.7	1.83	0.6
80	1.84	390.8	1.87	1.1	1.92	0.9
90	2.06	805.6	2.20	1.7	2.18	1.4
100	2.47	1500.7	2.48	2.5	2.52	2.1

表-2　近似解の比較：予算=最小木の4倍

ノード数	Scott法		貪欲解法1		貪欲解法2	
	誤差(%)	計算時間(s)	誤差(%)	計算時間(s)	誤差(%)	計算時間(s)
10	0.01	0.0	0.01	0.0	0.01	0.0
20	0.09	0.1	0.10	0.0	0.10	0.0
30	0.11	1.2	0.11	0.0	0.11	0.0
40	0.15	6.2	0.18	0.1	0.17	0.1
50	0.26	23.5	0.28	0.2	0.28	0.1
60	0.27	69.2	0.29	0.4	0.30	0.3
70	0.31	174.6	0.34	0.6	0.33	0.6
80	0.31	388.9	0.36	1.0	0.37	0.9
90	0.35	802.5	0.40	1.6	0.38	1.4
100	0.48	1495.7	0.53	2.5	0.53	2.1

法と0.3%以内の差の解を算出できることを示した．また，ノード100までの問題に対して2.5秒以内で近似解を求めることができることを示し，大規模な問題に対しても有効な解法であることを示した．

表-3　近似解の比較：予算=最小木の6倍

ノード数	Scott法		貪欲解法1		貪欲解法2	
	誤差(%)	計算時間(s)	誤差(%)	計算時間(s)	誤差(%)	計算時間(s)
10	0.01	0.0	0.01	0.0	0.01	0.0
20	0.03	0.1	0.03	0.0	0.03	0.0
30	0.04	1.1	0.06	0.1	0.06	0.0
40	0.07	6.1	0.07	0.0	0.07	0.1
50	0.09	23.2	0.10	0.1	0.10	0.1
60	0.08	68.7	0.10	0.3	0.10	0.3
70	0.11	173.6	0.11	0.6	0.11	0.5
80	0.10	387.4	0.12	1.0	0.12	0.9
90	0.17	799.6	0.19	1.6	0.19	1.3
100	0.18	1491.2	0.21	2.4	0.21	2.0

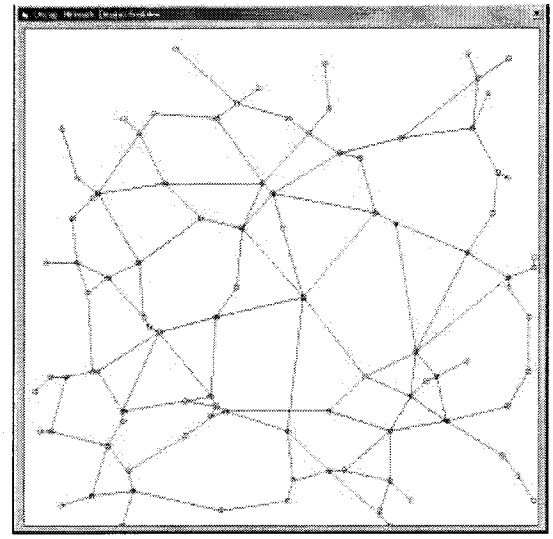

図1. 貪欲解法１による解のネットワーク

（予算=最小木の2倍，100ノード，4950アーク）

参考文献

1) Gallo, G. : lower planes for the network design problem, Networks, Vol. 13, pp. 411-425, 1983.

2) Ahuja, R. and Murty, V. : new lower planes for the network design problem, Networks, Vol. 17, pp. 113-127, 1987.

3) 片山直登，春日井博：ラグランジュ緩和法を用いた予算制約をもつネットワークデザイン問題の解法，日本経営工学会誌，Vol. 46, pp. 21-27, 1995.

4) Scott, A. : the optimal network problem: Some computational procedures, Transportation research, Vol. 3, pp. 201-210, 1969.

5) 西村昂，日野泰雄：最適ネットワーク構成に関する一考察，土木学会論文報告集，Vol. 250, pp. 85-97, 1976.

6) 森津秀夫：最適交通網構成手法に関する基礎的手法，神戸大学，1984.

7) Minoux, M. : network synthesis and optimum network design problems: models, solution methods and applications, Networks, Vol. 19, pp. 313-360, 1989.

8) 片山直登：ネットワークデザイン問題の近似解法，流通経済大学流通情報学部開校記念論文集，pp. 171--191, 1997.

9) Steenbrink, P. : Optimization of Transport Networks, John Wiley & Sons, 1974.