

名古屋大学工学部	正 員	河上 省吾
名古屋大学工学部	正 員	林 良嗣
名古屋大学大学院	学生員	○杯 由紀夫
名古屋大学大学院	学生員	野口 零一

1. はじめに

現在使われている最短経路探索法には、LP法・DP法・Moore法・Dijkstra法・Warshall-Floyd法などがある。これらの方法の中で能率的であるといわれているのがDijkstra法である。一般に、Dijkstra法は1ノードから出るリンク数が一定のとき、最短経路樹を作るのに要する計算時間は、ノード数の2乗程度に比例するといわれる¹⁾。したがって交通需要予測を行う際、対象となるネットワークが大規模にあれば、計算時間の多くはこの最短経路探索に費やされることになる。そこで本発表では、大規模ネットワークを小ネットワークに分割することにより、計算時間を短縮し、また、分割に応じて所望の精度で近似解を求める方法を提案しようとするものである。なおネットワークを階層的に分割する方法は、実際にはいくつか試みられているが、それらのほとんどはかなり恣意的な方法であり、精度の保障が与えられるものはないといえる。

2. 階層的経路探索法の手順

- (1)対象となるネットワーク(図1-a)を、境界上になるべく多くのセントロイドが位置するように小ネットワーク(図1-b)に分ける。
- (2)各小ネットワーク毎に、内部のセントロイドとスクリーンノード(ここで、小ネットワークの境界となるノードをスクリーンノードと呼ぶことにする)の間の最短経路探索をDijkstra法により求める。(図1-c)
- (3)各小ネットワーク内で求めた最短経路を、新たにリンクと置き、セントロイドとスクリーンノードのみをノードとしたネットワークを構成する。そして、セントロイド間の最短経路を求める。(図1-d)

3. 計算作業量の従来の方法との比較

計算作業量は、従来の方法では式(1)、本方法では式(2)によつて概算される。

$$F_1 = CN^{k_1} \quad \text{—— (1)}$$

$$F_2 = \sum_{i=1}^m \{ (C_i + S_i) N_i^{k_1} \} + C(C + S)^{k_2} \quad \text{—— (2)}$$

ここに、 C : セントロイド数, N : ノード数,
 N_i, C_i, S_i : ゾーン i のノード数 セントロイド数、スクリーンノード数(省セントロイド),

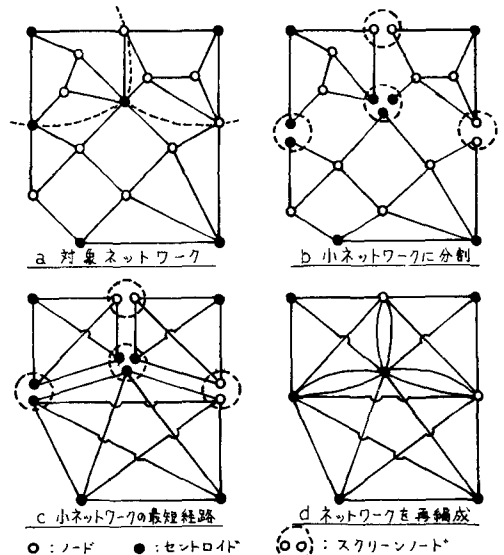


図1 階層的探索法の手順

m : ゾーン数, $1 < k_1 < 2$, $1 < k_2 < 2$

図2の名古屋市道路網について最短経路を求める。ネットワークは125ノード、224リンク、16セントロイドで、1ノードから出る平均リンク数は3.6本である。このネットワーク全体をDijkstra法で解く場合をケース0とする。次に階層的探索法で厳密な最短経路を求めるのをケースAとする。二の結果を表1に示す。表中のノード数 a とCPU時間 c の関係より、CPU時間は a^k ($1 < k < 2$) に比例していることがわかる。その結果、16本の最短経路樹を作るのに要する計算時間は195ms→127msと35%程短縮されているのがわかる。

4. 計算の精度と計算時間の関係

表2は、図2のネットワークのスクリーンノードのうち、

セントロイドと重なり、ていない幾つかのノードを、スクリーンノードとせずに最短経路を求めたときの近似計算例である。ここでは、各ケースについて誤差が得られており、これをを用いれば分析の目的に応じて、短時間で近似解を求めることができる。なおケースAの平均トリップ長は33分である。

5. おわりに

大規模ネットワークに階層的探索法を用いることによって、経路探索に要する時間を短縮することができることをここに示した。さらに、道路交通量がその容量に比して小さい場合にネットワークの改修があった場合、関連する小ネットワークのみ再計算すればよい場合がほい、多数の代替案を比較する場合には各代替案ごとに全ネットワークについて計算するのに比して、大幅な計算の節約がなされる。厳密には、道路交通流はflow-dependentであるので、影響はネットワーク全体に波及すると考えられるが、改修リンクからの距離によって影響量は小さくなると思われるので、このような近似計算が可能になる。また、本表では名古屋の市街地のネットワークを例に示したが、本方法は、その解法からわかるように、本来はより大規模で、ネットワークの何箇所かにネック（例えば地形上の制約）の存在する広域道路網に対してさらに有効であると言える。最後に、階層的探索法はflow-independentな鉄道ネットワークの場合には、より有効であることは既に確かめられている^{22,3)}。

〈参考文献〉

- 1) 伊理正天他編：ネットワーク構造を有するオペレーションズリサーチ問題の電算機処理に関する基礎研究、日本オペレーションズリサーチ学会、1973.3
- 2) 中村英夫、林良嗣他：大規模鉄道ネットワークにおける経路探索の簡略化に関する研究、エネ学会論文報告集（投稿中）
- 3) 林良嗣：大規模交通ネットワークモデル、第2回地域計画と地域データベースシンポジウム、1981.11

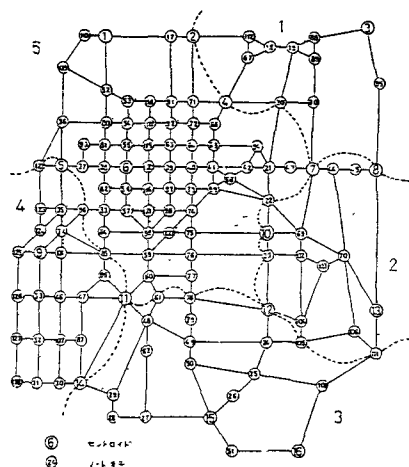


図2 名古屋市道路ネットワーク

表1 ノード数によるCPU時間

	ノード数 a (個)	リンク数 b (本)	セントロイド スクリーン ノード数 c (個)	1ノードが 出るリンク数 d=2b/a (本)	スクリーンノードを 作るのに要する CPU (ms)	CPU時間 f=c×e (ms)
ケース 0	125	224	16	3.6	12	195
ケース A	ゾーン1	14	17	6	2.4	0.1
	ゾーン2	17	24	9	2.8	0.1
	ゾーン3	22	28	9	2.5	1
	ゾーン4	25	41	7	3.3	1
	ゾーン5	69	114	17	3.3	4
	結合ネット	26	224	16	18.8	2.3

表2 誤差平均とCPU時間

	誤差平均	標準偏差	最大誤差	CPU (ms)
ケース B	0.38	2.06	15.77	110
ケース C	0.88	1.73	8.46	94
ケース D	0.54	2.10	15.77	93
ケース E	1.73	3.39	20.38	72