

大規模鉄道ネットワークにおける経路探索の
簡略化手法に関する研究A STUDY ON ALGORITHM FOR FINDING SHORTEST PATHS
IN LARGE-SCALE RAILWAY NETWORKS

林 良 嗣*・榎谷博光**・大島邦彦***・中村英夫****

By Yoshitsugu HAYASHI, Hiromitsu MAKITANI, Kunihiko OSHIMA
and Hideo NAKAMURA

1. はじめに

鉄道新線の整備あるいは既設路線における速度や頻度の改善等、1つの路線の交通条件の変化は、その沿線地域のみならず、鉄道路線網全体の交通に影響を及ぼす。その結果、この影響の及ぶ広い範囲にわたって住宅や産業等の立地条件が変化する。

このように、鉄道ネットワークの整備は交通や土地利用に対して広域的な影響をもたらすものである。しかしながら、従来広域における大規模ネットワークの解法が十分に実用的なものではなかったために、これらの影響の分析の大きな障壁となっていた。その理由としては、次に示すような広域土地利用交通分析の性格を十分に考慮した鉄道ネットワークモデルが開発されていなかったことが挙げられる。

① 鉄道ネットワークにおける交通の特徴として、「乗り換え」が存在し、また同一路線を異なる等級の列車が運行していること。

② 圏域内の各地区（駅）から、特定少数個の従業地（駅）までの所要時間を求める必要はあるが、必ずしもすべての地区（駅）間の所要時間を求める必要はない場合が多い。

③ 広域土地利用交通分析全体の精度から考えて、必ずしも厳密な地区（駅）間所要時間は要求されない場合が多い。

④ 土地利用あるいは交通計画等においては、一般に1回のネットワーク計算のための効率性よりも、多数の、しかも現状のネットワークに対して一部分変更を加えた程度のネットワーク代替案の比較の容易さが要求さ

* 正会員 工博 名古屋大学助教授 工学部土木工学科
** 正会員 工修 東京工業大学助手 工学部社会工学科
*** 正会員 工修 熊谷組(株)技術研究所第2部
**** 正会員 工博 東京大学教授 工学部土木工学科

れる。

本研究では以上のような特性を踏まえ、大規模鉄道ネットワークにおける駅間所要時間を必要十分な精度で効率的に見出すための方法をつくり上げるものである。

2. 従来の最短経路探索手法

最短経路探索は最大流問題と並んでネットワーク問題の基礎であり、従来から多くの手法が開発されている。

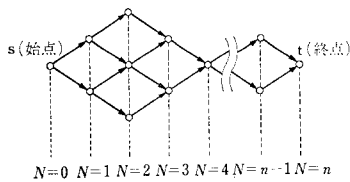
これらは表-1に示すように、その理論的根拠を数理計画法におくものとネットワーク理論におくものとの2つに大別することができる¹⁾²⁾。ここでは、これら従来の最短経路探索手法を計算効率の点から比較してみる。

一般に、あるネットワークについて最短経路探索を行う場合の計算量は、用いる手法のアルゴリズムから大略推測が可能である。この場合、ネットワーク自体の特性をネットワーク規模（ノード数）とネットワーク構造（1つのノードから出るリンク数）で表現すると、計算量がおもにネットワーク規模に依存する手法（Dijkstra法、Warshal-Floyd法など）、およびおもにネットワーク構造に依存する手法（LP法、DP法、Moore法など）とに分けることができる。ここで、後者についてはネットワーク構造を指標化することは難しいので図-1に示

表-1 従来開発されてきた最短経路探索手法とその必要計算回数

	手 法	計 算 回 数
数 理 計 画 法	Linear Programming	pML^{N-1}
	Dynamic Programming	$p(N-1)ML$
ネ ッ ト ワ ー ク 理 論	Dijkstra 法	$pq \log p$ 程度
	Moore 法	$p(N-1)ML$ 程度
	Warshall-Floyed 法	p^3 に比例

N: 段階数, M: 各段階におけるノード数, L: 各ノードから出るリンク数, p: 全ノード数, q: 全リンク数



図一 経路探索説明図

すような構造を1つの標準形として計算量を見積ることとする。同図において、 s (始点) から t (終点) までの最短所要時間を求めるものとする、経路探索は s (第0段階) から順次第1段階、第2段階、……と行われる。 N はこの段階数を示す。なお、いずれの手法においてもその計算プログラムは単純な加算と大小比較よりなっている、これらの演算を計算量の単位としている³⁾。

表一は文献 1), 3), 7) 等に示された計算量をまとめたものであるが、この表より、一般に LP 法による計算は DP 法によるものより多くの計算を要することがわかる。また Moore 法はそのアルゴリズムが DP 法に似ているため、これらの計算量は同程度である。また、図一1のような細長い梯子状のネットワークでは、表一の N は p に比例するが L および M がそれぞれ 2~3 程度の値となるため、DP 法あるいは Moore 法による計算回数は p^2 に比例する程度となり、他の方法に比べて有利であるといえる。逆に各ノードから出るリンク数が多く、ネットワークの形状が複雑な場合には Dijkstra 法が有利となることわかる。

これらのことからわかるように、現実のネットワークで最短経路探索を行う場合、そのネットワークの構造特性および上述のような各手法の特徴を十分に検討したうえで用いるべき手法を選択することが必要である。

3. 従来の鉄道ネットワークモデル

前述のような各種の最短経路探索手法を実際の鉄道ネットワークの分析に適用した研究例はいくつかあるが、大規模なネットワークでは相当な計算時間を要しており、少数のネットワーク代替案間の比較にとどまっている。

本項ではまず鉄道ネットワークの特徴について述べ、従来の鉄道ネットワークモデルがどのようなネットワーク構成法を用いているかについての概観をした後、大規模鉄道ネットワークへの適用に際しての問題点を明らかにする。

(1) 鉄道ネットワークの特徴

種々の交通ネットワークの中にあって、鉄道ネットワークは次のような特徴を有する。

- ① 通常1つの路線内に等級の異なる複数の列車(普通, 準急, 急行など)が運行されている。
- ② 同一路線, 同一等級であっても上りと下りで運行間隔が異なる。
- ③ 異なる路線の交差駅における別線への乗り換えのみならず, 同一路線内での異等級列車への乗り換えが存在し, しかも乗り換えに要する時間がそれぞれの場合で異なる。

(2) 一般的なネットワーク構成法

(1) で述べたような鉄道ネットワークの特徴を組み入れて最短経路探索を行うには、次に述べるようにして現実のネットワークから仮想的なネットワークを構成することが必要になる。

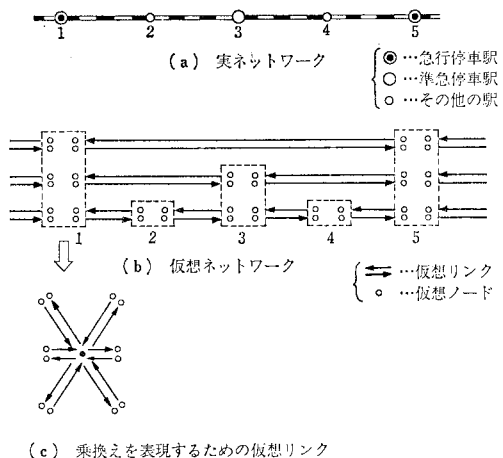
① 仮想ノードの設定

各駅(実ノード)について、そこを通る路線およびその列車等級ごとに対応する仮想ノードを設定する。

② 仮想リンクの設定

各駅間を結ぶリンク(実リンク)を列車等級ごとに分解し、それぞれに対応する仮想リンクを設定するとともに、各駅について乗り換えを表現するための仮想リンクを設定する。

上記のように仮想ノードおよび仮想リンクを設定することにより、仮想ネットワークが得られる。図二はその概念図を示すもので、図二(a)のような実ネットワークにおいて上述のような仮想ノードおよび仮想リンクを設定し、図二(b)に示すような仮想ネットワークを得、さらに各駅(たとえば同図の駅1)について異なる路線あるいは異なる等級の列車への乗り換えを表現するための仮想リンクを設定する(図二(c))。なお、図二(c)で駅中心の黒丸は各駅コードに対応した代表ノードである。また同図において、駅内の仮想リンクには路



図二 仮想ネットワークの構成法

線識別のためのコード番号を与えてあり、これによって通過と乗り換えを区別して扱うことができる。

(3) 従来の鉄道ネットワーク計算の限界

前項で述べた仮想ネットワーク構成法から容易に理解されるように、最短経路探索を行うためには実際は幾何形状が比較的簡単な鉄道ネットワークであってもきわめて複雑な仮想ネットワークを構成する必要がある。ここでは実際の鉄道ネットワークから仮想ネットワークを構成する際にネットワークがどの程度大規模・複雑化するかを、図-3に示す簡単なネットワークを例として定式化し、従来の鉄道ネットワークモデルを大規模鉄道ネットワークに適用する際の限界について考察する。

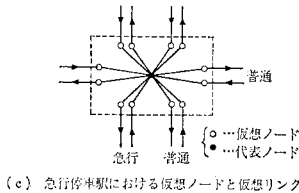
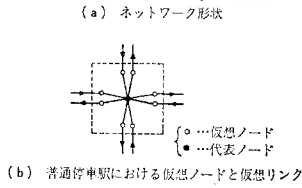
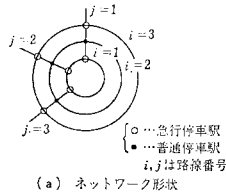


図-3 単純なネットワークの例

a) 仮想ネットワークの規模

簡単のため、図-3(a)の環状方向の路線 ($i=1, 2, \dots, I$) には普通列車のみが、放射方向の路線 ($j=1, 2, \dots, J$) には普通、急行列車のいずれもが運行し、停車駅は図に示すとおりとする。

図-3(a)において、実ノード数を N とすると、明らかに

$$N=IJ \dots \dots \dots (1)$$

であり、各駅における仮想ノード数は、図-3(b)、図-3(c)に示すとおり、普通停車駅では9個、急行停車駅では13個となる。ここで、放射方向の急行停車駅数を K とし、 $I=2K-1$ の場合について考えると、普通停車駅は $(I-K)J$ 個、急行停車駅は KJ 個存在する。したがって、仮想ノードの総数を N_v とすれば、一般に、

$$N_v=9(I-K)J+13KJ=(11I+2)J \dots \dots \dots (2)$$

またこのとき、実リンクは環状方向に IJ 本、放射方向に $(I-1)J$ 本存在するから、実リンク総数 L は

$$L=IJ+(I-1)J=(2I-1)J \dots \dots \dots (3)$$

となる。次に仮想リンクについて考える。まず、環状方向の仮想リンク数は、

$$I \times J \times 1 (\text{列車等級}) \times 2 (\text{上下方向}) = 2IJ \dots (4)$$

また、放射方向の仮想リンク数は、急行列車用の仮想リンクとして (急行停車駅区間数) \times (上下方向) $= (K-1)J \times 2$ (本)、普通列車用の仮想リンクとして (普通停車駅区間数) \times (上下方向) $= (I-1)J \times 2$ (本) となるから、放射方向の仮想リンク総数は、

$$(K-1)J \times 2 + (I-1)J \times 2 = 3(I-1)J \dots \dots \dots (5)$$

さらに各駅内での乗り換えを表現するための仮想リンクが必要である。これは 図-3(b)、図-3(c) からわかるように、急行停車駅で12本、普通停車駅で8本存在するから、乗り換えのための仮想リンク総数は、

$$KJ \times 12 + (I-K)J \times 8 = 2(5I+1)J \dots \dots \dots (6)$$

したがって仮想リンク総数を L_v とすると、(4)~(6)を加えて

$$L_v = 2IJ + 3(I-1)J + 2(5I+1)J = (15I-1)J \dots \dots \dots (7)$$

b) 経路探索の計算回数

図-3のように4方に広がりをもつネットワークに対しては、2.で述べたように、Dijkstra法が現在最も優れているといわれていることから、これを用いることにする。Dijkstra法では、近年の研究では一般にあるノードから他のすべてのノードに至る最短経路を求めるための計算量は、(総ノード数) \times (総リンク数) $\times \log$ (総ノード数) のオーダー程度まで縮小されるようになってきた。これを用いると、全駅間の所要時間を求めるのに要する計算回数 F は、1回の加算あるいは大小比較を単位として、

$$F \approx N_v L_v \log N_v = (11I+2)(15I-1)J^2 \log \{(11I+2)J\} \dots \dots \dots (8)$$

上述の N_v, L_v, F などの値を I, J の値ごとに実際に計算してみた例が表-2である。この表をみると、計算に際して記憶しておくべき仮想ノード数 N_v 、仮想リンク数 L_v は、実ノード数 N 、実リンク数 L に比べてはるかに大きく、図-3(a)程度の簡単な鉄道ネットワークにおいてすら、このように複雑な仮想ネットワークとなるのがわかる。したがって、大規模な鉄道ネットワークでは膨大な計算機容量を要することが推察される。また、計算回数も駅数の増加につれて急激に増加する。後で示すように駅数200程度のネットワークについてもかなりの計算時間を要している。したがって、たとえば首都圏の鉄道ネットワークの場合1000個以上の駅が存在することを考えると、膨大な計算時間を要し、実際への適用は容易でないことが理解されよう。このように、

表-2 ネットワーク規模の増大に伴う計算回数の増加

I	J	N	N_v	L	L_v	F
3	4	12	140	20	176	5.29×10^4
5	5	25	285	45	370	2.59×10^5
7	10	70	790	130	1040	2.38×10^6
⋮	⋮	⋮	⋮	⋮	⋮	⋮

従来の手法ではネットワークの規模が大きくなるにつれて適用性は急激に低下する。

4. 鉄道ネットワーク階層モデル

従来の鉄道ネットワークモデルを首都圏全域のような大規模ネットワークに適用することの困難性については上述のとおりである。

本研究においては、ネットワークを階層的に再構成することにより、大規模鉄道ネットワークに対して適用可能なネットワーク計算モデルを構築する。これは、ネットワークを大きく2種類のネットワークに分解することにより、全体のネットワーク計算を時間、容量の両面から効率化しようとするものである。すなわち、階層化によって得られたこれら2種類のネットワークに対し、おのおのの構造上の特徴に適した最短経路探索手法を適用して後に示すような最短所要時間テーブルを作成し、これを用いて全ネットワーク上の任意の2駅間の最短経路が見出される。以下に本モデルの考え方および従来の鉄道ネットワークモデルとの比較について述べる。

(1) 階層化の考え方

本研究では、駅を2階層に分けることによってネットワークを2つに階層化する。以下にその定義を示す。

a) 駅の階層化

図-4に示すような対象地域内すべての駅(実ノード)の集合を $\{N\}$ とし、これを次のような2つの集合に階層分けする。

$\{N_1\}$: 異なる路線の交差駅および始発・終着駅(これらをまとめて分岐駅とよぶ)よりなる集合(図-5に示すノード)。

$\{N_2\}$: 上記以外の駅(普通駅とよぶ)よりなる集合。

b) ネットワークの階層化

上のように定義された分岐駅、普通駅を用いて実ネットワーク(図-4)を次のような2階層に分ける。

分岐駅ネットワーク: 分岐駅間を仮想リンクで結ぶことにより得られるネットワーク(図-5)。

分岐駅へのアクセスネットワーク(以下アクセスネットワークと称する): 隣り合う分岐駅を両端とし、普通駅および実リンクで構成されるネットワーク。

(2) 経路探索の方法

いま、図-4のネットワーク上の i 駅より j 駅に至る最短経路を見出すことを考える。本研究では、 i, j よりそれぞれ最寄りの分岐駅に至るアクセスネットワーク内の所要時間、および分岐駅間の所要時間を別々に求め、その組合せによって i, j 間の所要時間を得る方法

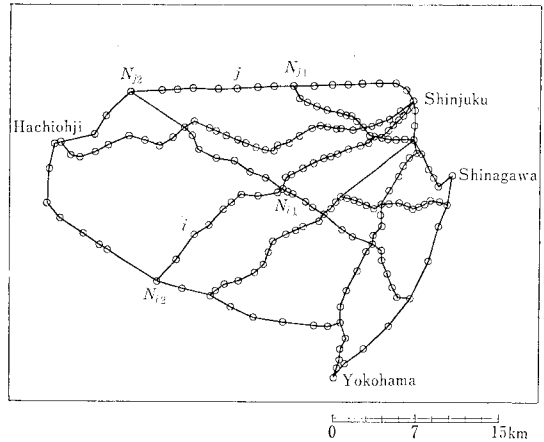


図-4 実ネットワークの例

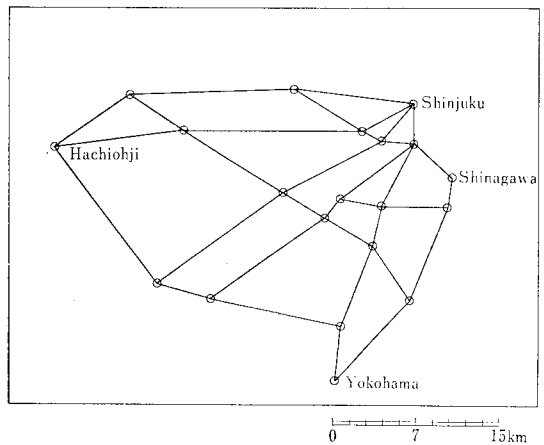


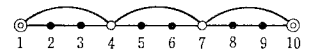
図-5 分岐駅ネットワーク

を提案する。以下にこの方法を説明する。

a) 分岐駅ネットワーク上での経路探索

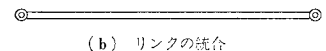
① 仮想リンク、仮想ノードの設定

分岐駅間を結ぶリンクは、たとえば図-6(a)の

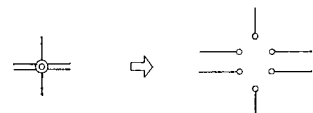


(a) 仮想ネットワーク

◎ ……分岐駅
○ ……急行停車駅
● ……普通停車駅



(b) リンクの統合



(c) 仮想ノードの設定

図-6 リンクの統合および仮想ノードの設定

を 図-6(b) に示すような統合された2本の仮想リンクで表わす。また、すべての分岐駅に接続する列車等級ごとに仮想ノードを設ける(図-6(c))。

② 最短経路の厳密性の保証

分岐駅間のみに対してならば前記の分岐駅ネットワークについて従来の最短経路探索手法を適用すればよいのであるが、後述のアクセスネットワークとの接続を考えると、出発側の分岐駅 N_i (図-4 で i をはさむ2つの分岐駅) および到着側の分岐駅 N_j において、それぞれの等級の列車でアクセスネットワークに接続するかによって N_i, N_j 間の最短経路が異なる場合が生ずる。

そこで、以下の条件に適合する経路を最短経路候補として考慮することにより、あらゆる駅間の最小所要時間の厳密性を保証する。すなわち、 N_i, N_j における任意の路線間および列車等級間の乗り換え時間のうちで最大のものをそれぞれ T_{Ni}, T_{Nj} とし、分岐駅ネットワーク上での N_i, N_j 間の最小所要時間を $T_{\min}(N_i, N_j)$ とするとき、両駅間の所要時間 $T(N_i, N_j)$ に対して、

$$T(N_i, N_j) \leq T_{\min}(N_i, N_j) + T_{Ni} + T_{Nj} \dots\dots\dots (9)$$

を満足するすべての経路が、任意の駅 i, j 間の最短経路を求める場合の通過経路の候補として考慮すべき必要十分な組合せである。

このような複数の経路の所要時間を比較するのに適した経路探索手法として、本研究では Moore 法⁹⁾を採用する。

③ 分岐駅間所要時間テーブル

上のようにして選び出された各最短経路候補に対し、 N_i からの出発リンク、 N_j への到着リンク別に 表-3 に示すようなテーブル形式で所要時間の情報を蓄えておく。

b) アクセスネットワーク内の探索

表-3 分岐駅間所要時間テーブル

発 駅 コード	着 駅 コード	所要時間	発リンク コード	着リンク コード
N_1	N_2	10	l_1	l_2
N_1	N_2	16	l_3	l_4
⋮	⋮	⋮	⋮	⋮
	N_3	12	l_5	l_6
	N_3	8	l_7	l_8
	⋮	⋮	⋮	⋮

表-4 アクセスネットワーク内所要時間テーブル

発 駅 コード	着 駅 コード	N_{i1} (または N_{j1})			1	2	⋯	N_{i2} (または N_{j2})		
		急行	準急	普通				急行	準急	普通
N_{i1} (または N_{j1})	急行									
	準急									
	普通									
1										
2										
⋮										

各アクセスネットワーク (首都圏全体で約 300 個) はそれぞれ小規模なネットワークであるから、従来の最短経路探索手法 (ネットワーク構造の特徴から DP 法を採用した) をそのまま適用して計算し、両端の分岐駅に対しては出発、到着リンク別にして 表-4 のようなテーブル形式で所要時間の情報を蓄える。

e) 任意駅間の最小所要時間

先述のごとく任意の 2 駅 i, j 間の最短経路候補としては、分岐駅ネットワークおよびアクセスネットワーク上での経路の組合せがいくつか考えられるが、それらの候補についてのみ以下の手順を施すことにより、 i, j 間の最小所要時間を求めることができる。

(i) i, j の属するアクセスネットワークの所要時間テーブルを用い、それぞれ両端の分岐駅 N_{i1}, N_{i2} および N_{j1}, N_{j2} までの所要時間 $T(i, N_{i1}), T(i, N_{i2})$ および $T(j, N_{j1}), T(j, N_{j2})$ を読み取る。

(ii) 分岐駅間所要時間テーブルで上記の 4 つの分岐駅間の所要時間 $T(N_{i1}, N_{j1}), T(N_{i1}, N_{j2}), T(N_{i2}, N_{j1}), T(N_{i2}, N_{j2})$ を読み取る。この中には先述の最短経路候補も含む。

(iii) 4 つの分岐駅における乗り換え時間を $T_{Ni1}, T_{Ni2}, T_{Nj1}, T_{Nj2}$ (これらは乗り換え先路線の平均運行間隔の 1/2 と考える注1)) とするとき、たとえば $i \rightarrow N_{i1} \rightarrow N_{j2} \rightarrow j$ なる経路に対する所要時間は、

$T(i, N_{i1}) + T_{Ni1} + T(N_{i1}, N_{j1}) + T_{Nj1} + T(N_{j1}, j)$ となる。 i より j に至る経路に含まれる分岐駅は他に 3 つが考えられる。すなわち経路としては $i \rightarrow N_{i1} \rightarrow N_{j2} \rightarrow j, i \rightarrow N_{i2} \rightarrow N_{j1} \rightarrow j, i \rightarrow N_{i2} \rightarrow N_{j2} \rightarrow j$ が考えられるが、上と同様にして所要時間が求まる。これらの最小値が $i \rightarrow j$ の最小所要時間である。

(3) 大規模鉄道ネットワークへの適用

以上のような鉄道ネットワークモデルを実際の鉄道ネットワークに適用し、従来の手法との比較を行う。

a) 対象ネットワーク

ここでは首都圏の鉄道ネットワークのうち東京および神奈川に位置する 11 路線を選んで構成したものをサンプルネットワークとする。このネットワークに含まれる実ノード数 (駅数) は 188, 実リンク数は 202 であり、首都圏全域のほぼ 5 分の 1 の規模を有し、これを先述のように仮想ノード、仮想リンクに分解した場合、仮想ノード数 710, 仮想リンク数 2 048 とかなり大規模なネットワークとなる。

b) モデルの適用

注 1) 各分岐駅においては、乗り換え前後の路線番号および列車種別についてのチェックを行い、両者が同じ (すなわち乗り換えなしの直通列車) 場合には乗り換え時間をゼロとしている。

表—5 従来の手法との計算時間の比較

	手 法	計算時間 (min : s)
全 駅 間	Dijkstra 法	9 : 09.21
	階層モデル	6 : 47.24
全駅より特定の1駅	Dijkstra 法	9 : 09.21
	階層モデル	0 : 05.24

(IBM 東京サイエンティフィックセンター system 370/158 を使用)

表—6 近似解法と従来の手法との比較 (全駅間)

手 法	所要時間 (min : s)	誤 差
Dijkstra 法	9 : 09.21	
近似解法	Case-1 (許容誤差 5 分)	$\mu=1.9 \text{ min}$ $\sigma=2.66 \text{ min}$
	Case-2 (誤容誤差 15 分)	$\mu=2.1 \text{ min}$ $\sigma=2.73 \text{ min}$

μ : 誤差の絶対値の平均, σ : 誤差の標準偏差

表—7 特定駅間の解法 (特定 33 駅間) と従来の手法との比較

手 法	計算時間 (min : s)
Dijkstra 法	1 : 28.76
特定駅間の解法	0 : 11.10

(IBM 東京サイエンティフィックセンター system 370/158 を使用)

上記のサンプルネットワークの形状を画像出力したものが先の 図—4 であるが, 3. (2) で述べたような方法でこれより仮想ネットワークを構成し, 従来の最短経路探索手法の中でも最も優れた手法の1つである Dijkstra 法をそのまま適用して計算を行った。

一方, 先述の階層化の考え方に従い, 図—5 に示すような分岐駅ネットワーク, さらに個々のアクセスネットワークに分けて計算を行った。

なお, 今回の計算では「全駅→全駅」, 「全駅→特定の1駅」の2ケースについて 厳密な最短経路を求めている。

c) 計算結果

結果は表—5 に示すとおりであり, 本研究で開発したモデルは従来の手法に比べて計算時間をかなり短縮できることがわかる。ここで, 階層モデルの計算時間には各段階の所要時間テーブル作成のための計算時間も含まれている。このことは表—6 および表—7 についても同様である。特に「全駅→特定の1駅」の場合, Dijkstra 法³⁾では始点駅よりすべての駅までの経路探索が行われ, その結果として特定駅までの所要時間が求まるため, 結局全駅間の探索が行われることになるのに対し, 階層モデルでは各段階で作製された所要時間テーブルを用いることにより効率的に処理され, 計算時間に格段の差が生じていることがわかる。なお, 両手法によって得られた各駅間の最小所要時間が完全に一致することも確認されて

いる。

5. 鉄道ネットワーク計算の簡略化手法

本研究で開発した鉄道 ネット ワーク階層 モデルにより, 従来の手法に比べて計算時間が大幅に節約されることが示された。しかしながら, 首都圏全域といった規模の鉄道ネットワークにおいてすべての駅間の厳密な所要時間を求めようとする, 相当な計算機容量および計算時間が必要であると考えられる。

一般に地域計画の分析においては, 全体の斉合性からみて所定の誤差範囲内でネットワーク計算を簡略化することが許される場合や, 必要とされる最短経路情報もゾーン中心駅のような特定の駅についてのみでよい場合が多い。

そこでこのような観点から, 利用目的に応じて所定の誤差範囲内で計算の簡略化を図るための手法, およびネットワーク上の特定駅間の最短経路のみを求めるための簡略解法について提案する。

この解法は, 先述の階層モデルにおいて最短経路探索の過程で種々の仮定を設け, 結果の厳密性は多少損なわれるが計算時間の大幅な短縮を図るものである。

(1) 近似解法

4. (2) で述べたように, 分岐駅ネットワーク, アクセスネットワークそれぞれの中での最短経路をつなぎ合わせたものが必ずしも元のネットワークでの最短経路であるとは限らない。

この解法は式 (9) を満足する最短経路候補の数を減らすことによって計算時間の短縮を図るものである。したがって得られる結果の厳密性は多少損なわれるが, 以下に述べるように, 生ずる誤差を一定の許容範囲内に抑えることが可能である。すなわち, 許容誤差を ϵ とするとき, 式 (9) の代わりに次の式 (10) を満足する経路のみを最短経路候補として考慮する。

$$T(N_i, N_j) \leq T_{\min}(N_i, N_j) + T_{N_i} + T_{N_j} - \epsilon \dots\dots\dots(10)$$

この種のモデルの例として本研究では次の2つの場合について, 図—4 のネットワークを対象として計算を行った。

- Case-1 : 許容誤差 (ϵ) = 5 分
- Case-2 : 許容誤差 (ϵ) = 15 分

結果は表—6 に示すとおりであり, 従来の手法に比べて計算時間を飛躍的に短縮している。また, Case-1, Case-2 ともに誤差の平均は 2 分程度であり, 通常の広域的な計画分析においては十分実用に耐え得るものと考えられる。

(2) 特定駅間の解法

これは、特定駅間の所要時間のみが必要とされる場合に用いる解法で、階層モデルで定義された分岐駅の概念を特定駅まで拡張し、分岐駅と特定駅のみで構成されるネットワーク上での経路探索のみにより、特定駅間の所要時間を求めるものである。

この解法はネットワーク構成自体を簡略化しているため、表一7に示すように計算時間は従来の手法に比べ圧倒的に短く、しかも得られる結果は厳密である。したがってゾーン中心駅等の特定駅間についての所要時間のみが必要とされるような分析に対しては有利な手法であるといえる。

(3) 多数のネットワーク代替案の比較

土地利用交通計画等の代替案として分析されるネットワークは、現状のネットワークに一部追加した程度のものが通常である。これに対して本研究では、4.(2)で示したように所要時間情報を分岐駅ネットワーク、アクセスネットワークのそれぞれに分けて蓄えておくという方法を用いているため、ネットワーク変更に対しても代替ネットワーク全体を計算し直す必要はなく、変更の生じた路線についてのアクセスネットワークの計算および分岐駅ネットワークの再計算のみで対処できる。

本研究による階層モデルが1つのネットワークに対して有効であることは示したが、上述の理由から多数のネットワーク代替案の比較に対しても従来の手法よりはるかに柔軟に対処することが可能であり、総計算時間の節約はさらに大きい。

6. 結 論

以上本研究において新たに開発した鉄道ネットワーク階層モデル、およびこれを利用目的に応じて簡略化したモデルについて、その基本的考え方と実際の鉄道ネットワークへの適用結果について述べたが、本研究における成果を要約すると以下のようになる。

(1) ネットワークを階層的に再構成することにより、従来の最短経路探索手法をそのまま適用する場合に比べ計算量を大幅に縮小し、首都圏におけるような大規模鉄道ネットワークに対しても適用可能なモデルを構築した。

(2) 種々の分析レベルにおける目的や必要精度に応じてネットワーク構成あるいは経路探索アルゴリズムを

簡略化することにより、計算時間を飛躍的に短縮することができた。

(3) 階層的なネットワークの構成・計算により、多数のネットワーク代替案の比較に要する総計算時間を大幅に節約することができた。

なお、本論文は広域的な土地利用交通モデルの開発に際しての必要性から着手されたものである。このネットワーク問題の研究にあたっては、基本的構想は主として中村によるものであるが、実際の研究開発は林、楨谷、大島が行い、特にモデルの構成および論文のとりまとめは主として林と楨谷が、計算処理はすべて大島が担当して行ったものである。

最後に、本研究を進めるにあたり日本 IBM 東京サイエンティフィックセンターの松家英雄、杉本和敏の両氏にはプログラム整備およびデータ処理に関してお世話いただいた。また、東京理科大学 内山久雄助教授、東京大学 宮本和明講師には、土地利用交通モデルの開発に関連して貴重なご意見をいただいた。ここに記してこれらの方々に感謝の意を表するものである。

参 考 文 献

- 1) 高橋啓郎・五百井清右衛門：ネットワークプログラミング，森北出版数学ライブラリー 42，pp. 1~34，昭和 51 年。
- 2) 伊理正夫・古林 隆：ネットワーク理論，日科技連 OR ライブラリー 12，pp. 18~52，昭和 51 年。
- 3) 伊理正夫ほか：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，上級情報処理技術者教育のための電算機による問題解決手法の調査研究報告書，情報処理研究センター，日本オペレーションズリサーチ学会，pp. 15~20，昭和 48 年。
- 4) Aho, A.V., J.E. Hopcroft and J.D. Ullman: Data Structures and Algorithms, Addison Wesley, pp. 206~208, 1982.
- 5) アメリカ市町村協会(新谷洋二・黒川 洗・中川三郎・山形耕一・涌井哲夫 訳)都市交通計画のたて方，鹿島出版会，pp. 98~109，昭和 47 年。
- 6) 松田正一・洲之内治男・杉山昌平・出居 茂：OR のための基礎数学 4，丸善，pp. 183~186，昭和 49 年。
- 7) 菅原 操ほか：土木工学大系 12 計画論，彰国社，pp. 225~239，昭和 52 年。
- 8) 白川 功・尾崎 弘：グラフとネットワークの理論，コロナ社，昭和 48 年。
- 9) 枝村俊郎・森津秀夫：最適交通ネットワーク問題の厳密解法と近似解法，土木学会論文報告集，No. 262，昭和 52 年。
- 10) Ishii, H.: A New Method Finding the K-TH Best Path in a Graph, Journal of the Operations Research, Vol. 21, No. 4, December 1981.

(1982. 4. 19・受付)