

# 地形的特性により探索領域を限定した 日本全国道路網における経路探索手法

飯村 伊智郎 加藤 誠巳  
上智大学 理工学部

近年車載ナビゲーション・システムの普及が著しいが、目的点までの最適経路を提供するものは少ない。最適経路を算出するためには道路ネットワーク・データを必要とするが、道路ネットワークが膨大になるとディスクへのアクセス・読み込み時間、探索を実行するために必要な RAM 領域、並びに計算所要時間が増大し実行に耐えなくなることがある。本論文では日本全国の道路ネットワークを地形的特性を近似的に表現すると考えられる複数個の郡レベルの行政区域に相当する閉領域に分割し、オフラインで作成した閉領域集合限定テーブルを利用することにより上述の欠点を解決したものである。郡レベルの行政区域を閉領域として採用した理由の一つは、一般に行政区界は河川や山脈等の地形的特性と密接な関係を有していることが多く、このような行政区界を横切る道路は少ないと考えられ、そのため閉領域集合限定テーブルを作る時間が短縮されることである。ここで提案する手法の他の特徴は、得られた経路の最適性が保証されていることである。

## A Route Finding Method using the Search Areas restricted through the Topographical Properties

Ichiro IIMURA Masami KATO  
Faculty of Science and Technology, Sophia University  
7-1, Kioi-cho, Chiyoda-ku, Tokyo, 102 Japan

In recent years, a number of onboard navigation systems have become commercially available. Few of them, however, can calculate the optimal route to the destination. Moreover, as the size of the road networks becomes larger, the access and read time of disk storage devices, the capacity of RAM required for calculation and the CPU time increases proportionally, which results in an impractical system. In this paper, the nationwide road networks of Japan are divided into a plurality of closed areas, each of which corresponds to a county. A table showing which closed areas are necessary and sufficient in order to calculate the optimal route was constructed offline. By using this table, the above mentioned problems can be solved. The reason why a county is selected as a closed area is that the border of a county is closely related to the topographical properties such as rivers and mountain ranges. Because only a few roads intersect such borders, the time required to obtain the table can be reduced. Another feature of the proposed method is that the route obtained is guaranteed to be optimal.

## 1 まえがき

近時、車載ナビゲーション・システムが急速に普及しつつある。しかし、GPS、マップマッチング技術等を用いて自車位置、走行軌跡を表示するものが主体であり、目的点への最短距離あるいは最短時間経路を提供するものは少ない。特に道路ネットワーク・データが膨大になると、例えば Dijkstra 法<sup>[1]</sup>等従来から良く知られている手法をそのまま適用すると、ネットワーク・データの記憶されているディスクあるいは CD-ROM 等へのアクセスおよび読み込み時間が長大となるだけでなく、データをロードし探索アルゴリズムを実行するために必要な RAM 領域が膨大になるとともに、計算実行時間も実用に耐えなくなる場合がある。このため出発点および目的点が指定されたとき、それらを囲む矩形または楕円の領域<sup>[2]</sup>に含まれる道路ネットワーク・データを RAM にロードして探索を実行する方法が提案されているが、経路が求まらない場合もあり、経路が求まっても最適である保証がない。また、A\*アルゴリズム<sup>[3]</sup>として知られている手法では、推定コストなる概念を用いて探索する道路ネットワーク・データを限定しているが、推定コストを算出するために計算時間がかかる欠点があった。膨大なネットワーク・データを RAM にロードすることを回避し、計算時間を短縮する他の方策として道路ネットワークを階層化し、出発点および目的点の近辺のみ下位の階層の道路ネットワークも使用し、それ以外では上位の階層のネットワークのみを使用することが行なわれている<sup>[4][5]</sup>が、道路階層間の遷移に制約があるため、最適な経路が求められる保証がない。

本論文では、道路ネットワークのノードは、抽象的なネットワークのノードとは異なり、固有の  $(x, y)$  座標を有していることに着目し、膨大な道路ネットワークを互いに素な複数個の閉領域に分割し、任意の2つの閉領域間の最適経路を算出するために RAM にロードすべき閉領域を予めすべて求めて、これをテーブルとしている。経路探索実行時に出発点と目的点が指定されると、出発点および目的点がそれぞれ属する閉領域を求め、これら2つの閉領域間の最適経路を算出するのに必要とされる閉領域をテーブルを参照して求め、これら閉領域に属する道路ネットワーク・データのみを RAM にロードして計算を実行することにより、ディスクへのアクセス・読み込み時間の短縮、必要とされる RAM の容量の減少ならびに計算時間の高速化を図ることが可能となる<sup>[6]</sup>。ここで提案する手法の他の利点は、求めた経路の最適性が保証されることである。

道路ネットワークを分割する閉領域としては、格子状直線により矩形領域に分割することも考えられるが、ここでは任意の2つの閉領域間の最適経路を算出するのに必要とされる閉領域を記憶している閉領域集合限定テーブル・データの算出時間の短縮を図るため郡レベルの行政区域を採用した。

以下で提案する手法、データの生成法および評価結果等について詳細に述べる。

## 2 従来の探索領域限定法

道路ネットワークにおける経路探索において、道路ネットワーク・データの格納されているディスクへのアクセス・読み込み時間を短縮させ、またロードして探索を実行するための RAM 容量を削減し、経路算出時間を高速化するため、意味がないと考えられるネットワーク・データを経路探索の対象から除外することは従来から行なわれてきた<sup>[2][3][4][5]</sup>。例えば、出発点および目的点が指定されたとき、図 1(a) および図 1(b) に示すようにそれらを囲む矩形または楕円の内部に相当する道路ネットワークのみに探索領域を限定することが考えられる<sup>[2]</sup>。ほとんどの場合これで十分な結果が得られるが、例えば図 1(a) あるいは図 1(b) に示すように出発点と目的点の間の湾や川、山脈等によって経路が見い出せない場合が生じ得るし、たとえ得られたとしても最適な経路である保証はない。

探索領域を自動的に限定し、しかも最適な経路を得ることが可能な方法として A\*アルゴリズムが知られている<sup>[3]</sup>。良く知られている最適経路探索法である Dijkstra 法では、出発点からの累積コストが最小である未探索のノードから未探索のリンクを探索延伸することにより実行されるのに対し、A\*アルゴリズムでは出発点からの累積コストとそのノードから目的点への推定コストの和が最小である未探索のノードから

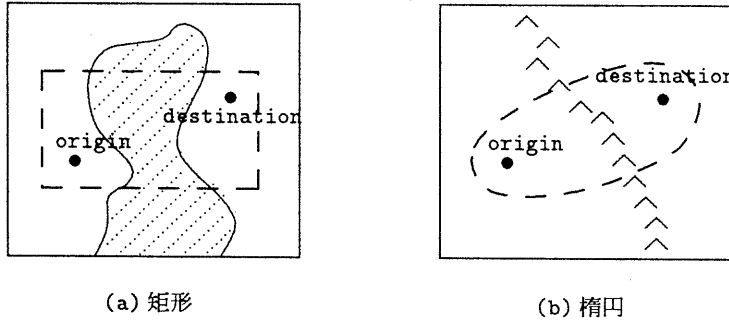


図 1 矩形・楕円による探索領域限定

未探索のリンクを探索延伸することにより実行される。対象とするネットワークが道路ネットワークのようにノードが 2 次元の座標を有する場合（さらに標高を考慮して 3 次元の場合でも），推定コストとしてそのノードと目的点との直線距離に相当するコストを採用すれば求めた経路は最適であることが保証される。この A\* アルゴリズムの欠点は推定コストを算出するのに時間がかかることと，例えば日本の道路ネットワークのように細長い形状をしたネットワークを対象とした場合，九州から北海道に至る経路を探索しようとするとはほとんどのネットワーク・データを探索の対象としなければならないことである。

RAM に読み込むネットワーク・データ量および探索に要する CPU 時間を減少させる他の方法として階層化経路探索が知られている[4][5]。これは出発点および目的点の近辺でのみデータ量の多い下位の階層の道路ネットワークも使用して最寄りの上位階層の道路ネットワークへのアクセス・ポイントを見出し，それ以外では上位の階層の道路ネットワークのみを使用して経路探索を行なうのであるが，道路階層間の遷移に制約があるため，最適な経路が求められる保証がない。

### 3 提案する探索領域限定法

前述の通り従来の探索領域限定法には問題点があった。ここで提案する探索領域限定法は，以下に述べるようなテーブルを参照することにより，必要にして十分な領域のみの道路ネットワークをディスクから RAM にロードすることにより，ディスクへのアクセス・読み込み時間を短縮し，必要な RAM 容量を削減し，経路算出時間を高速化するとともに得られた経路の最適性を保証するものである。

道路ネットワークの各データにはそれぞれ固有の  $(x, y)$  座標が対応しているので，この道路ネットワークを互いに素な複数の閉領域に容易に分割することができる。例えば，図 2 に示すような  $N$  個の閉領域に道路ネットワークを分割した場合を考える。このとき閉領域を形成する閉曲線と道路ネットワークのリンクとの交点ノードを境界ノードと呼ぶことにする。例えば閉領域  $\#i$  に属する出発点  $O$  から，閉領域  $\#j$  に属する目的点  $D$  に至る最適経路を探索することを考える。閉領域  $\#i, \#j$  の境界ノード数をそれぞれ  $k_i, k_j$  とし，閉領域  $\#i, \#j$  の第  $i'$  および第  $j'$  境界ノードをそれぞれ  $n(i, i'), n(j, j')$  とする。出発点  $O$  と目的点  $D$  が異なる閉領域に属する場合，すなわち  $i \neq j$  のとき，出発点  $O$  から目的点  $D$  に至る最適経路は閉領域  $\#i$  の境界ノード  $n(i, i'), (i' = 1 \sim k_i)$  のいずれかから出て，閉領域  $\#j$  の境界ノード  $n(j, j'), (j' = 1 \sim k_j)$  のいずれかから入ることは明らかである。例えば境界ノード  $n(i, i')$  から  $n(j, j')$  に至る最適経路が閉領域  $\#p, \#q, \#r$  を通るならば，これらの閉領域は閉領域  $\#i$  に属する出発点  $O$  から閉領域  $\#j$  に属する目的点  $D$  に至る最適経路を探索する場合に，閉領域  $\#p, \#q, \#r$  に属する道路ネットワーク・データは必須であることがわかる。このようにして閉領域  $\#i$  の  $k_i$  個の境界ノード  $n(i, i'), (i' = 1 \sim k_i)$  をそれぞれ出発点，閉領域  $\#j$  の  $k_j$  個の境界ノ

$n(j, j')$ , ( $j' = 1 \sim k_j$ ) をそれぞれ目的点とする  $k_i \times k_j$  通りの最適経路を算出し、それらすべての経路が通過する閉領域の集合が、閉領域  $\#i$  に属する任意の出発点  $O$  から閉領域  $\#j$  に属する任意の目的点  $D$  に至る最適経路を算出する場合に必要な十分な閉領域の集合となる。道路ネットワークが双方向性のときは、出発点が閉領域  $\#i$ 、目的点が閉領域  $\#j$  に属する場合に必要な十分な閉領域の集合と、出発点が閉領域  $\#j$ 、目的点が閉領域  $\#i$  に属する場合に必要な十分な閉領域の集合は同一であるが、双方向性が成り立たないときは2つの集合は異なるがその差異は僅少であると考えられるので記憶容量を削減するため、ここでは両集合の和集合を用いることにした。

また出発点  $O$  と目的点  $D$  が同一の閉集合  $\#i$  に属する場合にも同様にして最適経路を求めるのに必要にして十分な閉領域の集合が求められるが、図3に示すように閉領域  $\#i$  のみのデータでは十分でない場合が多いことに注意する必要がある。

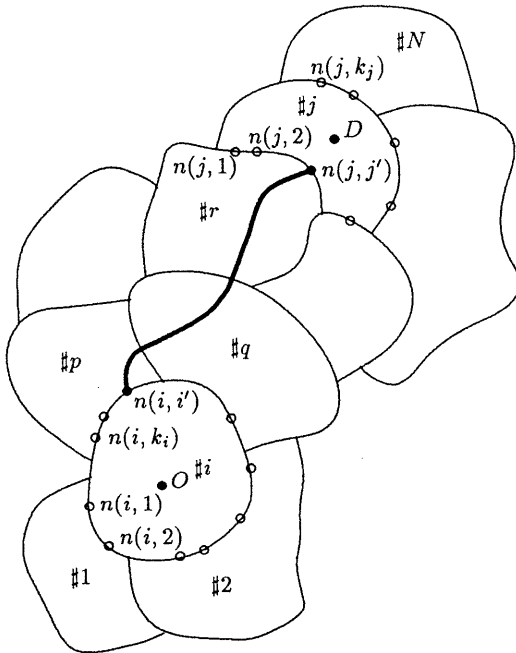


図2 閉領域への分割

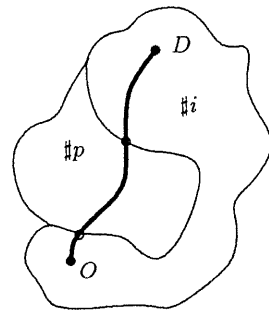


図3 特殊な例

このようにして表1に示すような閉領域  $\#i$  と閉領域  $\#j$  の間の最適経路を探索するのに必要にして十分な閉領域の集合を表現するテーブル（これを閉領域集合限定テーブルと呼ぶ）が形成される。このテーブルのサイズは行方向  $N$ 、列方向  $N(N+1)/2 (= \sum_{i=1}^N i)$  であり、テーブルの0または1を1ビットで表現することにすると、

$$\frac{1}{2} N^2(N+1) \text{ ビット} \dots\dots\dots (1)$$

の記憶容量を必要とすることになる。

$N$  が大きいほど、一般に一つの閉領域の面積は小さくなるから、RAMにロードすべきデータは減少するが、テーブル・データが膨大となり適当な大きさの  $N$  を選定する必要がある。

ところで、この閉領域集合限定テーブルを作成するために、各境界ノードから他のすべての境界ノードに至る最適経路をオフラインで求める必要があるが、その計算時間はほぼ境界ノード数に比例するので境界ノード数が少ないほど有利となる。道路ネットワークを分割する閉領域のとり方として、格子状直線により矩形領域に分割することも考えられるが、ここでは上述の境界ノード数を少なくすることを目的として郡レベルの行政区画を閉領域として用いることにした。これは郡レベルの行政区画は河川や山脈等の地形的特性と密接な関係を有していることが多く、このような行政区界と交差する道路ネットワークのリンクの数、すなわち境界ノード数は少ないと考えられるからである。

ここではこのような地形的特性により境界ノードを減少させるため、昭和23年当時の地図<sup>[7]</sup>を参照して、現在市制を布いているところも郡（北海道の場合は支庁）に編入して全国を513個の閉領域に分割した。例えば“東京都”は“23区”、“北多摩郡”、“西多摩郡”、“南多摩郡”の4つの閉領域に分割したが、“23区”の境界の一部を成す多摩川、江戸川、荒川などには数えるほどの橋しかないこと、また“西多摩郡”の埼玉県、山梨県との境界はほぼ山の尾根であってほとんど道がないことは明らかである。

この場合は、 $N = 513$ であるので閉領域集合限定テーブルの容量は式(1)より約8.45MBとなる。

表1 閉領域集合限定テーブル

出発 閉 領域	目的 閉 領域	#i ↔ #j 最適経路探索に 必要十分な閉領域の集合を表すテーブル								
		#1	#2	#3	...	...	...	...	...	#N
#1	#1	1	*	*	...	...	...	...	...	*
#1	#2	1	1	*	...	...	...	...	...	*
#1	#3	1	*	1	...	...	...	...	...	*
⋮	⋮	⋮								
#1	#N	1	*	*	...	...	...	...	...	1
#2	#2	*	1	*	...	...	...	...	...	*
#2	#3	*	1	1	...	...	...	...	...	*
⋮	⋮	⋮								
#2	#N	*	1	*	...	...	...	...	...	1
#3	#3	*	*	1	...	...	...	...	...	*
⋮	⋮	⋮								
#3	#N	*	*	1	...	...	...	...	...	1
⋮	⋮	⋮								
#N	#N	*	*	*	...	...	...	...	...	1

1: 必要  
0: 不要  
\*: 0 または 1

## 4 データ生成の方法

本論文で対象とした道路ネットワークは、日本デジタル道路地図協会の作成した日本全国の基本道路、すなわち都道府県道以上の道路、都道府県道以下の道路幅員が5.5m以上の道路およびこれらの道路間を連結する道路で構成されており、ノード数は226,737、リンク本数は521,190である。この日本デジタル道路地図協会のデータには行政区画の変化点に相当するノードが存在し、またリンクの属性として行政区画コードが付与されているので、前述した513の郡レベルの行政区画を一つの閉領域としてまとめることは容易にできる。このようにして形成された513の閉領域の境界ノード総数は7,425であり、一つの閉領域当たりの平均境界ノード数は境界ノードが2つの閉領域に共有されることを考えると約29となる。

このようにして得られた閉領域を単位とするネットワーク・データは閉領域毎に1から順にノード番号が付与され、各リンクの始点ノード番号、終点ノード番号、そのコストならびに属性等を一括して読み込めるようにした。この場合、これら513の領域より成るネットワーク・データは全体としてディスク中で約21.8MBの容量を占めており、したがって閉領域一つ当たりの平均ネットワーク・データ量は約42KBである。

ところで3で述べた閉領域集合限定テーブルは日本全国の道路ネットワーク・データに対し、7,425回出発点をかえて他の7,424点に至る最適経路を求め、求めた経路を構成するリンクがどの閉領域に属するかを調べる必要があるため、ワークステーション SPARCstation 10 を用いて約80時間を要した。

## 5 最適経路探索の手順

本論文で提案する地形的特性により探索領域を限定した経路探索の処理の手順は次の通りである。

**ステップ1:** 出発点、目的点が属する閉領域 $i, j$ を求める。

**ステップ2:** ディスク中にある閉領域集合限定テーブルの $(i, j)$ 行( $i \leq j$ のとき)、または $(j, i)$ 行( $i > j$ のとき)を読み込む。

**ステップ3:** 読み込まれた閉領域集合限定データに応じて、必要十分な閉領域ネットワーク・データをディスクから読み込み、ノードおよびリンクに連続番号を付与する。その際、隣接する閉領域が存在する場合には、相応する境界ノード間をコスト0の仮想リンクにより相互接続する。

**ステップ4:** 得られたネットワーク・データに対してDijkstra法等通常の方法を用いて最適経路を探索する。

## 6 経路探索実行例と評価

ここで提案した手法の利点を明らかにするために、日本全国の道路ネットワーク・データをRAMに読み込んだ後、通常のDijkstra法あるいはA\*アルゴリズムを適用した場合との比較を行なった。評価項目としてはディスク中の道路ネットワーク・データへのアクセス・読み込み時間、必要とされるRAM容量に比例すると考えられるRAM中に読み込まれたリンク本数、ならびに最適経路を探索するのに要したCPU時間を用いた。また探索に要するCPU時間にほぼ比例すると考えられる探索したリンク本数も参考のため示した。例1は出発点=能登半島先端、目的点=伊豆半島先端、例2は出発点=山口県西部、目的点=高知県中央、例3は出発点=九州佐多岬、目的点=北海道知床岬であり、時間をコストとして最適経路の探索を行なった結果を表2に示す。なお探索の実行にはSPARCstation 370を使用した。

表2 評価結果

	探索手法	ディスクアクセス 読み込み時間 (sec)	読み込まれた リンク本数 (本)	探索した リンク本数 (本)	探索に要した CPU時間 (sec)
例1	本手法	5.77	54,804	54,465	1.36
	通常 Dijkstra	136.42	536,040	353,997	7.41
	A*アルゴリズム	同上	同上	230,924	19.15
例2	本手法	3.00	25,556	23,254	0.72
	通常 Dijkstra	136.42	536,040	177,995	3.72
	A*アルゴリズム	同上	同上	92,126	7.47
例3	本手法	24.37	209,522	208,510	5.28
	通常 Dijkstra	136.42	536,040	529,576	12.72
	A*アルゴリズム	同上	同上	528,688	45.40

図4, 図5および図6に, それぞれ例1, 例2および例3に対し本手法により RAM 上にロードされたリンクと算出された最適経路を示す.

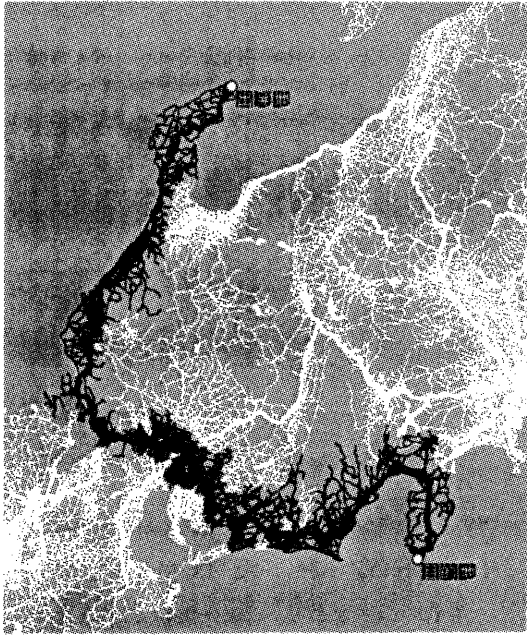


図4 例1

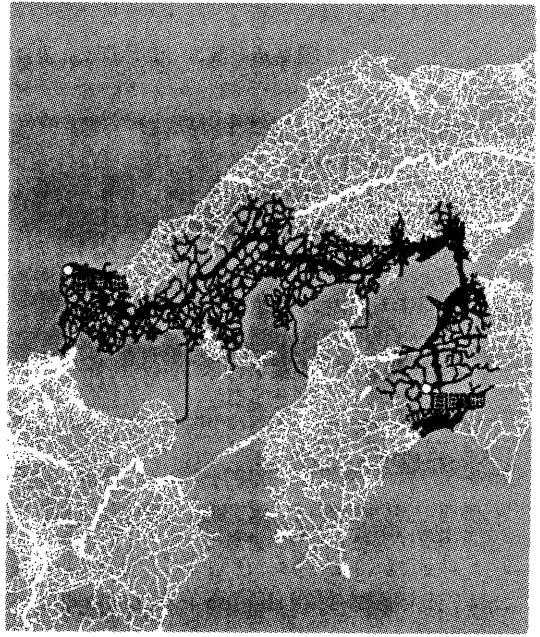


図5 例2

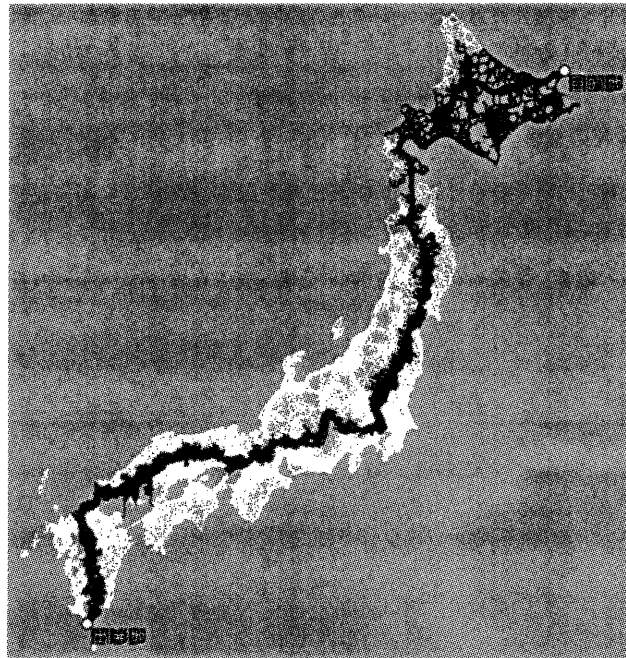


図6 例3

以上より本手法の利点は明らかである.

## 7 むすび

本論文では道路ネットワークを地形的特性を近似的に表現すると考えられる郡レベルの行政区域に相当する複数の閉領域に分割し、オフラインで作成した閉領域集合限定テーブルを利用することにより、ディスクへのアクセス・読み込み時間、RAMの必要容量および計算所要時間を大幅に削減させることができることを示した。

本手法の問題点は閉領域集合限定テーブルを作成するのにかなりの時間を有すること、渋滞や事故等の時間的変動を考慮に入れられないこと、新しい道路ができたときテーブルを作り直さなければならないこと等があげられる。またここでは郡レベル行政区域に相当する513の閉領域に分割したが、さらに望ましい分割法について今後検討を行なう予定である。

## 謝辞

最後に有益な御討論戴いた本学マルチメディア・ラボの諸氏に謝意を表する。

## 参考文献

- [1] 伊理 正夫, 中森 真理雄: 算法の最近の進歩, 信学誌, Vol.58, No.4, pp.433-445(1975).
- [2] 小林 祥延, 平野 和夫, 出川 裕久, 橋本 武夫, 名倉 充彦: 推奨経路表示機能付きナビゲーションシステム, 住友電気, 第141号, pp.151-160(1992).
- [3] Nils J.Nilsson 著, 白井 良明, 辻井 潤一, 佐藤 泰介 訳: 人工知能の原理, 日本コンピュータ協会(1983).
- [4] 加藤 誠己, 大西 啓介: 階層化されたデジタル地図データベースに基づく都心部自動車用経路案内システム, 信学技報, DE89-24(1989).
- [5] 丹羽 寿男, 吉田 雄二, 福村 晃夫: 道路網の階層的表現にもとづく経路探索アルゴリズムと地図情報システムへの応用, 情報処理学会論文誌, Vol.31, No.5, pp.659-666(1990).
- [6] 加藤 誠己, 飯村 伊智郎: 地形的特性により探索領域を限定した日本全国道路網における経路探索手法, 情処学会第48回全大, 2T-6(平06-03).
- [7] 日本地方行政研究会 編纂: 全国市町村便覧 附・分縣地圖, 日本観光出版株式会社(1948).