

多層ストロークネットワークの構築と道なり優先経路探索への応用

新帯 里奈[†] 山本 大介[†] 高橋 直久[†]

[†] 名古屋大学大学院工学研究科情報工学専攻 〒466-8555 愛知県名古屋市昭和区御器所町

E-mail: [†]shintai@moss.elcom.nitech.ac.jp, ^{††}{yamamoto.daisuke, naohisa}@nitech.ac.jp

あらまし 道なり経路は、右左折が少なく運転がしやすい、道案内が容易であるなどの利点がある。だが、道なり道路(ストローク)の判定や道なり経路の探索に多大な時間を要す等の課題がある。本稿では、道路の形状からストロークを求め、ストローク数最少の中で最短の経路を求める手法(SPMS法)を提案し、以下の特徴を持つ経路探索システムの実現法について述べる。(1)事前に複数の基準長以上のストロークからなるネットワーク(ストロークネットワーク(SN))を作成し、SNDBを構築。これにより、ストロークの判定時間を短縮とともに適切なネットワークを選択可能にする。(2)基準長の異なる複数のSNを合成し、多層ストロークネットワーク(MSN)を構築。また高密度のSNの範囲を適応的に変化。これにより、ノード数が少なくコネクティビティの高いMSNを作る。(3)MSN上で、幅優先探索によりSPMS法を実現。また本稿では、MSNと従来の階層的ネットワーク(MCN)及び全ての道路からなるネットワーク(ベースライン)を探索グラフとし、SPMS法を適用し、探索時間を比較した。MSNは、ベースラインより100分の1、MCNより10分の1の探索時間で探索できることがわかった。

キーワード 経路探索, 道路ネットワーク, 道なり, ナビゲーションシステム

1. はじめに

経路には最短経路や有料道路優先経路、道なり優先経路など様々な経路が存在する。しかし、「道なり」という概念はあいまいで、道なり経路を実現するには以下の課題がある。

問題1 道なり道路(ストローク)の判定

道路形状に沿って忠実に「道なり道路」を選択しようとする、と、交差点ごとに各リンクがストロークに含まれるか判定するのに時間を要する。

問題2 探索グラフの作成

全ての交差点を頂点とするグラフでは頂点数が膨大になり、経路探索に多大な時間を要する。一方、長いストロークを抽出し、それらの交点を頂点とするグラフでは、出発地と目的地を結ぶ経路を探索できないことがある。

問題3 ストローク数最少経路の探索

出発地から目的地に向かう経路の中でストローク数が最少の経路を求めるとき、幅優先探索では計算が進むにつれて候補経路が大量に発生し途中結果のデータ量も膨大になる。

我々はこれらの問題を解決するため、道路の形状に沿ってストロークを求め、基準長以上のストロークからなるグラフ(ストロークネットワーク(SN))を作成する。問題2の解決のため、異なる基準長の複数のSNを合成し多層ストロークネットワーク(MSN)を生成する。また、高密度のSNの範囲を適応的に変化させる。これにより、ノード数が少なくコネクティビティの高いMSNを実現させる。

経路探索において、文献[3]ではできるだけ道なり経路を探索する手法を実現しているが、最も道なりに従う経路を求めているわけではない。また、ノードを隣接する道路の数だけ分割して経路探索を行うため、経路探索に時間を多く要する問題がある。

本稿では、道路の形状を表すジオメトリに従ってストローク

を求め、ストロークデータベース(SDB)を作成する。さらに、ある長さ以上のストロークを抽出し、それらのストロークからなるストロークネットワークを求め、ストロークネットワークデータベース(SNDB)を構築する。そして、基準長の異なる複数のストロークネットワークを合成することで、多層ストロークネットワーク(MSN)を生成する。MSN上で最少ストローク数最短経路探索法(SPMS法)で最少ストローク数の経路の中で最短の経路を求めるシステムを提案する。これにより、ストロークネットワーク上での最少ストローク数の経路の中で最短の経路を実現する。

2. 関連研究

経路探索に関しては、道路の特徴をコストにして渋滞を回避する経路を探索する研究[4]、ダイクストラの最短経路探索から高速化アルゴリズムを開発する研究[5]などがある。

文献[3]では、右左折等のコストを考慮した道路網ネットワークのデータ構造を提案し、そのデータ構造を用いて出発点から任意の到着点までの最小コスト経路を探索する手法を提案している。全ての交差点の右左折リンクのコストを直進リンクのコストより大きく設定すると右左折の回数が少ない経路が得られる。しかし、この経路の右左折の回数が最少となるとは限らない。

文献[6]では、道路網を道路の役割を基準に階層化して表現し、階層化により生成されたブロック単位で最短経路を探索する手法を提案している。道路網の階層は上位層の道路が幹線道路や高速道路などの広く長い道路で長距離の移動に適した道路、下位層は上位層が表現する領域を上位層の道路で区切り、それから得られる多角形領域内の道路網としている。出発点を含む下位層から2点を含む上位層、到着点を含む下位層へ順に最短経路探索を行うことで、探索するノードを減らし、長距離の移動

ほど上位層の道路が含まれる経路が得られる。しかし、出発点・到着点のいずれかを含む下位層と2点を含む上位層に階層の差があるとその分さらに時間がかかってしまう。本研究は、上位層にあるような道路からなるストロークネットワークと、下位層にある道路からなるストロークネットワークを重ね合わせ1つのストロークネットワークにし、その上でストロークに基づいて道なり優先である経路を求めることを目的とするため、これらとは大きく異なる。

3. 提案システムの概要

3.1 提案システムの特徴

本研究では、道路の形状を表すアークというジオメトリの成す角に従ってストロークを求め、図1に示すような基準長以上の長さがあるストロークからなるグラフ(ストロークネットワーク(SN))を構築する。ストロークネットワーク内に存在するストロークの割合を密度とする。基準長が長いSNは、基準長の短いSNに比べて、短いストローク上にある道路が間引かれているため、道路の密度が低くなる。

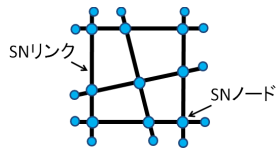


図1 ストロークネットワーク

本研究で提案するシステムの特徴を以下に挙げる。

特徴1 SN作成機能

”道なり”を形成するストロークが満たすべきルールを定める。このルールに基づいて、複数の基準長に対してSNを作りデータベース(SNDB)化する。これによりストローク判定時間を短縮させ、適切なネットワークを選択可能にする。

特徴2 MSN生成機能

地理的範囲(以下、単に範囲という)と基準長の異なる複数のSNを合成し、MSNを構築する。また、高密度のSNの範囲を適応的に変化させる。これにより、ノード数が少なくコネクティビティの高いMSNを作る。

特徴3 MSN探索機能

MSNを探索グラフとして、幅優先探索により出発地の頂点からエッジを辿りストローク数最少の中で最短の経路を選択する経路探索法(SPMS法)を実現する。各頂点で方向別に経路を求めストローク数最少の経路を選択し候補経路を絞り込む。

3.2 提案システムで使用するデータ

提案システムにおいて使用するデータは、道路データとストロークデータ、ストロークネットワークのデータである。

3.2.1 道路データ構造

道路のデータ構造を図2に示す。道路データには、ポイント、ノード、アークの3つのデータがある。ポイントは、道路上の点を表し、位置座標(緯度、経度)からなる。道路の形状は図2のように道路上の隣接ポイントを順番につないでできる折れ線で近似的に表現される。ノードは、道路ネットワークの交差点であり、アークは始点ノードから終点ノードに至る道路で、道

路上のポイントの系列で表す。リンクはアークとノードをまとめたものであり、始点ノードと終点ノードの組をエッジと呼ぶ。

3.2.2 道路データベース

提案システムでは、道路データに対し、リンクごとに対応するノードとアークをまとめて、表1に示す形式のリンクテーブルを作成する。このとき、リンクは表2に示す9種類の道路クラスに分類し、アークは、線を表現するデータ型 LineString の形式[8]で表現する。

表1 リンクテーブル

カラム名	データ型	説明
ID	int	リンクを一意に識別するリンクID
clazz	int	道路のクラス
source	int	リンクの始点ノードのID
target	int	リンクの終点ノードのID
x1	double	リンクの始点ノードの経度
y1	double	リンクの始点ノードの緯度
x2	double	リンクの終点ノードの経度
y2	double	リンクの終点ノードの緯度
km	double	ノード間の道路の長さ(単位はkm)
arc	LineString	道路の形状を表す

表2 道路クラスについて

clazz	kmh	説明
11	120	高速道路
12	30	高速道路の出入り口
13	90	国道
14	30	国道への連絡路
15	70	主要地方道
16	30	主要地方道への連絡路
21	60	一般地方道
31	40	2車線以上の一般道
41	30	2車線未満の一般道

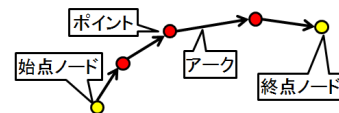


図2 道路データ構造

3.2.3 ストロークのルール

ストロークとは認知心理学に基づいて道路ネットワークをグルーピングしたものであり、道なりに続く道路に対応する。しかし、道なりという概念は形状や人の感覚によって異なり、何をストロークと定めるかにより違いが出る[9]。本研究では、道路の形状を表現するアークを道なりの判定に用いる。以下のルールに従って、連続したリンクを同一のストロークと定める。また、ストロークの例を図4に示す。

ルール1 図3の(a)のように注目するリンクLの端点に接続するリンクがリンク L_i の1つのみの場合、リンク L_i とリンクLは同じストロークに属する。

ルール2 図3の(b)のように注目するリンクLの端点に接続するリンクが複数の場合、注目するリンクと同じ道路のクラスで、隣接するアークの成す角が45度以下であり、かつ最小となるアークを持つリンク L_i をリンクLの系列の対象とする。

ルール3 大きな交差点内において交差点内リンクが存在する場合、交差点内リンクを挟むリンクを全て接続するものとみなし、条件2を満たすリンクをリンクLの系列の対象とする。

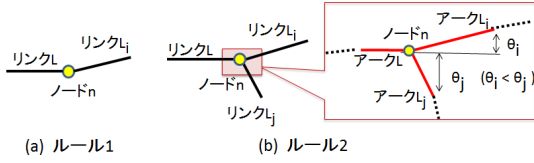


図3 ストロークルール

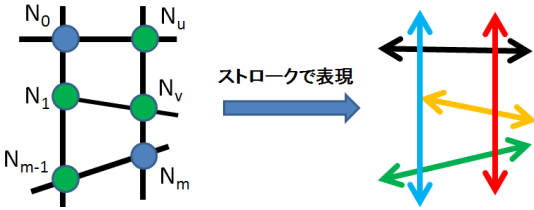


図4 ストローク例

3.2.4 ストロークデータベース (SDB)

ストロークデータベースにはストロークテーブルが存在する。ストロークテーブルのデータ構造を表3に示す。

表3 ストロークテーブル

カラム名	データ型	説明
stroke_ID	int	ストロークの ID
stroke_line	geometry(MultiLineString)	ストロークの形状
stroke_length	double	ストロークの長さ

3.3 単層ストロークネットワーク (SN(α , R))

単層ストロークネットワークは、範囲 R に存在する αm 以上であるストロークからなる、ストロークの交点または端点を SN ノード、SN ノード間を結ぶリンクの系列を SN リンクとする無向グラフである。SDB から出発点と到着点の2点を含む範囲 R 内で長さ αm 以上のストロークを抽出して単層ストロークネットワークを構築する。構築した単層ストロークネットワークは SNDB に格納する。

3.3.1 ストロークネットワークのデータ構造

ストロークネットワークのデータ構造を図5に示す。ストロークネットワークのデータには、SN ノードと SN リンクの2つのデータがある。SN ノードは、ストロークの交点または端点である。SN リンクは、SN ノード間を結ぶリンクの系列である。

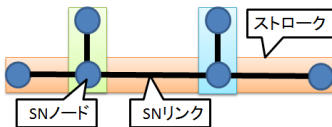


図5 SN ノードと SN リンク

3.3.2 ストロークネットワークデータベース (SNDB)

ストロークネットワークデータベースには SN テーブルと SN リンクテーブルが存在する。SN テーブルのデータ構造を表4に、SN リンクテーブルのデータ構造を表5に示す。

表4 SN テーブル

カラム名	データ型	説明
SN_ID	int	SN の ID
Base_length	double	基準長
SN_shape	MultilineString	SN の形状

表5 SN リンクテーブル

カラム名	データ型	説明
SN_linkID	int	SN リンクの ID
SN_ID	int	SN の ID
stroke_id	int	SN リンクのストローク ID
source_nodeID	int	SN リンクの始点ノード ID
source_x	double	始点ノードの経度
source_y	double	始点ノードの緯度
target_nodeID	int	SN リンクの終点ノード ID
target_x	double	終点ノードの経度
target_y	double	終点ノードの緯度
link_length	double	SN リンクの長さ
SN_shape	MultilineString	SN リンクの形状

3.4 多層ストロークネットワーク (MSN(SN_N, SN_L, SN_R))

単層ストロークネットワークだけでは出発点や到着点を通するストロークが含まれるとは限らないという問題がある。出発点や到着点を含めるために短いストロークからなるストロークネットワークではほとんどの道路が含まれ、SN ノード数が多くなる問題がある。これらの問題を解決するために、コネクティビティを保つように、出発点や到着点の周辺の範囲ではより短いストロークから、それ以外の場所ではより長いストロークからなるネットワークを構築する。出発点や到着点を含める広範囲でストロークの長さ α 以上であるストロークネットワーク (SN1) を、出発点や到着点の周辺で、出発点または到着点を通るストロークの長さ $\beta (\alpha \geq \beta)$ 以上であるストロークネットワーク (SN2, SN3) を SNDB から抽出する。SN1, SN2, SN3 を合成し、図6に示す多層ストロークネットワークを構築する。多層ストロークネットワークは SN ノード集合 (SN_N), SN リンク集合 (SN_L), 領域とその領域での基準長を対とした制約条件の集合 (SN_R) の3つの集合で表現される。

3.5 提案システムの構成

提案システムの構成を図7に示す。提案システムは、ユーザが指定した地図と MSN の範囲を描画し、MSN を構築する。経路探索における探索範囲を SN の範囲内に属する SN ノードとし、探索範囲内で最少ストローク最短経路探索法 (SPMS 法) により経路探索をする。SPMS 法はストロークネットワークデータベース (SNDB) より取得した各 SN リンクのストローク ID 用いて経路探索をし、道なり優先経路を実現する。提案システムは、ユーザが出発点と到着点、そして MSN を構築する範囲

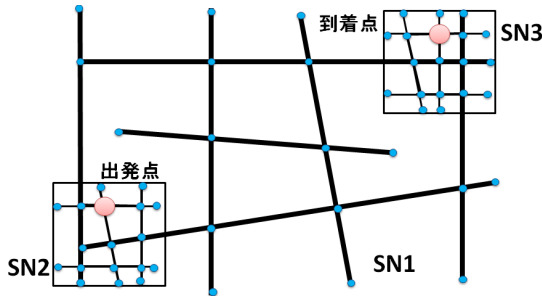


図 6 多層ストロークネットワーク

と基準長を指定すると、地図サーバから地図画像をストロークネットワークデータベース (SNDB) から範囲内の SN データを取得し、MSN を構築する。構築した MSN 上で SPMS 手法により経路を探索して地図画面に描画する。さらに、経路長、ストローク数、探索時間、SN 構築時間を地図外に表示する。

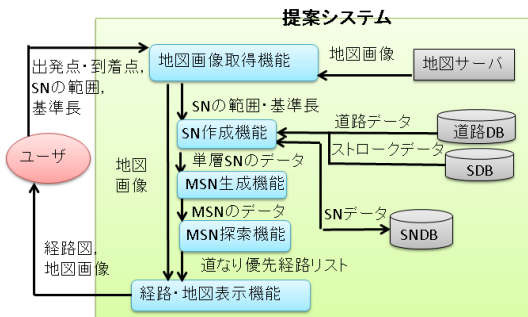


図 7 システム構成図

4. 提案システムの実現法

提案システムは SN 作成機能、MSN 生成機能、MSN 探索機能の 3 つからなる。それぞれの実現法を以下に示す。

4.1 SN 作成機能

SN 作成機能は SNDB 作成機能と SN 取得機能からなる。SNDB 作成機能は、以下の手順で SN を構築し、SNDB へ格納する。SN 取得機能は、SNDB より指定した範囲とストロークの基準長の SN を抽出する。

● SN 作成手法

- Step1 指定した基準長以上のストロークを抽出する。
- Step2 ストロークの交点および端点 (SN ノード) を求める。
- Step3 各ストローク上の SN ノードの位置でストロークを分割し、ストロークの断片を SN リンクと呼ぶ。

4.2 MSN 生成機能

MSN 生成機能には拡大機能と合成機能からなる。拡大機能は、合成する 2 つの SN 間のコネクティビティを保つように SN の範囲を調整する。合成機能は、2 つの SN の合成を繰り返して、多層ストロークネットワーク (MSN) を生成する。

4.2.1 合成機能

基準長の異なる複数の SN 内のストロークの中には共通のストロークが含まれることがあり、そのストローク上で SN リンクが異なることがある。共通するストローク内の SN リンクの

重なり方に着目すると、図 9 に示すような両端点が同一なもの (パターン 1)、一端のみが同一なもの (パターン 2)、また、ある SN リンクが他の SN リンクを覆い被さるもの (パターン 3) がある。このような重なりを解消するために、重なりがある SN リンクを分解し、重なりをもたない SN リンクを生成する必要がある。本研究では短い基準長のストロークネットワーク (SN2, SN3) を基にし、そのなかに長い基準長のストロークネットワーク (SN1) を加え、SN1 の SN リンクを重なりを持たない SN リンクへ変換させる。これにより、SN リンクが重なりが発生するのは SN2, SN3 の境界線上となる。

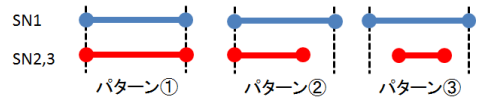


図 8 SN リンクの重なり

差分 SN リンク生成手法 差分 SN リンク生成手法では、重なる SN1 の SN リンクから差分 SN リンクを生成し、重なる SN1 の SN リンクを差分 SN リンクと置き換えることで、重なりを持たない SN リンクへ変換することが目的である。

パターン 1 では、まったく同一の SN リンクが 2 つあることになるので、差分 SN リンクは生成せず、SN1 の SN リンクを削除する。

パターン 2 では、2 つの SN リンクで共有していない端点をつないだリンクを生成し、ストローク ID をその SN リンクのストローク ID、長さを 2 つの SN リンクの長さの差とする。このとき、2 つの SN リンクで共有していない端点をつないだリンクを差分 SN リンクとし、SN1 の SN リンクと置き換える。

パターン 3 では、範囲を SN1 の SN リンクの長さとし基準長を SN2, SN3 の基準長とした SN を SNDB より取得し、SN1 の SN リンクの端点から SN2,3 の SN リンクの端点までに存在するリンクの系列を 1 つの差リンクにする。このとき、長さはこのリンクの系列の長さの合計であり、ストローク ID はこのリンクの系列のストローク ID とする。そして、SN2, SN3 の SN リンクの両側に存在する差リンク 2 つをそれぞれ差分 SN リンクとし、SN1 の SN リンクと置き換える。

アルゴリズム 短い基準長のストロークネットワーク (SN2, SN3) と長い基準長のストロークネットワーク (SN1) で重なりを持たない SN リンクと上記の手法より得た差分 SN リンクからなる多層ストロークネットワーク (MSN) を構築する。MSN の SN リンクを MSN.L, その SN リンクのフラグを $Flug_k$ として、以下に疑似コードを示す。

```

Step1 SN2,3 を MSN へ
for each SNlink  $L_i$  in SN2, SN3
    MSN.L ←  $L_i$ 
    MSN.Flugk = 0
Step2 SN1 を MSN へ
for each SNlink  $L_j$  in SN1
    if SN1. $L_j$  が SN2, SN3 の境界線上に存在する
        MSN.L ←  $L_j$ 
        MSN.Flugk = 1
    
```



```

else then
  MSN.L ←  $L_i$ 
  MSN.Flugin = 0
Step3 重なり補正
for each SNlink  $L_k$  in MSN
  if MSN.Flugin > 0
    link = 差分 SN リンク生成手法
    MSN.L ← link
    delete(MSN.Lk)

```

4.2.2 拡大機能

合成する基準長の異なる SN の範囲内に共通するストロークが存在せず、接続できないことがある。この問題を解決するために、合成前に、長い基準長のストロークネットワーク (SN1) と短い基準長ストロークネットワーク (SN2, SN3) に共通するストロークが含まれるまで短い基準長のストロークネットワーク (SN2, SN3) の範囲を一定の幅で拡大する。アルゴリズムとしては、合成前に基準長の異なる SN 内に共通するストロークが存在するか確かめる。存在しない場合には共通するストロークが存在するまで短い基準長のストロークネットワーク (SN2, SN3) の範囲の拡大を繰り返す。

4.3 MSN 探索機能

MSN 探索機能では、MSN 上で後述の最少ストローク数最短経路探索法により道なり優先経路を探索する。

4.3.1 最少ストローク数最短経路探索法 (SPMS 法)

最少ストローク数最短経路探索法とは、ストローク数が最少の中で最短距離である経路を探索する手法である。ストロークを考慮した経路探索をする上では、ある頂点を通過する場合、どの頂点から通過するかにより経路長とストローク数に違いが出る。このため、頂点を接続するエッジごとに分解し、分解したエッジデータから道なり経路探索を行う。

4.3.2 構造体

頂点、エッジデータ、エッジの関係を図 9 に示す。また、頂点とエッジデータ、エッジのそれぞれに構造体を持たせる。頂点とエッジは SNDB に格納された一意に識別する ID を持つ。またエッジデータはそれを所持する頂点のノード ID を用いて一意に識別した ID を持つ。

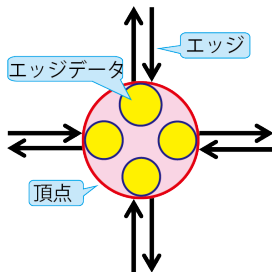


図 9 頂点とエッジデータ、エッジの関係図

頂点 各頂点には接続するエッジの数だけエッジデータを持ち、そのエッジデータ ID を持つ。また、各エッジデータはそれを所持する頂点を持つエッジ ID より取り出すことができることとする。各エッジデータが持つ構造体を表 6 に示す。初期状態

では経路長とストローク数は無限大の値とし、着目頂点の接続エッジのストローク ID と、直前の頂点の接続エッジのリンク ID とそのストローク ID は未登録とする。着目頂点の接続エッジのストローク ID は SNDB より取得する。

表 6 エッジデータの構造体

カラム名	データ型	説明
EdgeData_ID	int	着目頂点のエッジデータ ID
TotalLength	double	着目頂点の経路長
TotalStroke	int	着目頂点のストローク数
After_linkID	int	着目頂点の接続エッジのリンク ID
After_strokeID	int	着目頂点の接続エッジのストローク ID
Bofore_nodeID	int	直前頂点のノード ID
Before_linkID	int	直前頂点の接続エッジのリンク ID
Before_strokeID	int	直前頂点の接続エッジのストローク ID

エッジ 各エッジが持つ構造体を表 7 に示す。ストローク ID は SNDB より取得する。

表 7 エッジの構造体

カラム名	データ型	説明
Link_ID	int	エッジのリンク ID
Length	double	エッジの距離数
Stroke_ID	int	エッジのストローク ID
startNode_ID	int	エッジの始点のノード ID
nextNode_ID	int	エッジの終点のノード ID

4.3.3 アルゴリズム

経路探索するグラフを $G=(V,E)$ とし、頂点集合 V とエッジ集合 E でアルゴリズムを表現する。

ストローク数計算手法 ストローク数計算では、直前の頂点のエッジデータ $v_j.e_{ij}$ の接続エッジ e_{ij} のストローク ID と、着目頂点 v_k と直前頂点 v_j に接続するエッジ e_{jk} のストローク ID を比較し、比較した結果よりストローク数を計算する。比較した 2 つのエッジのストローク ID が同一であれば、着目頂点のエッジデータのストローク数は直前の頂点のエッジデータのストローク数となる。また異なれば、直前の頂点のエッジデータのストローク数に 1 加えたものとなる。計算した結果を着目頂点のエッジデータ $v_k.e_{jk}$ に登録する。直前の頂点のエッジデータ $v_j.e_{ij}$ 、 v_j に接続するエッジ e_{ij} 、着目頂点のエッジデータ $v_k.e_{jk}$ 、 v_k に接続するエッジ e_{jk} の関係図を図 10 に示す。

ストローク数最適解算手法 ストローク数最適解算手法では、ある頂点のエッジデータの中で以下のコスト比較条件を満たすエッジデータを探索する。

- ストローク数が最少
- ストローク数が同数なら、経路長が最小

このときコストはストローク数と経路長の 2 つであり、コスト

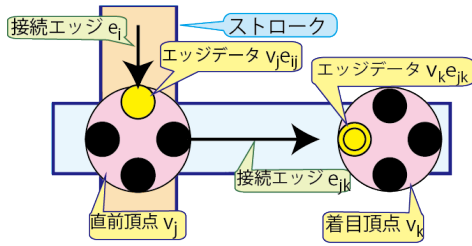


図 10 エッジデータと接続エッジ

比較の際にはストローク数のみを用いて、ストローク数が同数となる場合のみ経路長を用いる。着目するエッジに隣接する 2 つの頂点に着目し、着目頂点を持つエッジデータの内の着目するエッジのリンク ID を持つエッジデータ (エッジデータ 1) と直前の頂点を持つ複数のエッジデータからストローク数計算手法に従って、着目頂点のエッジデータ 1 のストローク数と経路長を計算し、コスト比較を行う。コスト比較条件を満たせばそのストローク数と経路長を着目頂点のエッジデータ 1 に上書き登録する。これを直前の頂点をもつエッジデータ全てに対して行い、着目頂点のエッジデータ 1 に最適解となるストローク数とその経路長とその時の直前の頂点や直前の頂点の接続エッジのリンク ID・ストローク ID を登録させる。着目エッジを e_{jk} とし、直前の頂点を v_j 、そのエッジデータを $v_k.e_{ij}$ 、着目頂点のエッジデータを $v_k.e_{jk}$ として、以下にストローク数最適解算手法 ($v_j, v_k.e_{jk}$) の擬似コードを示す。また、着目エッジ e_{jk} と直前の頂点 v_j とそのエッジデータ、着目頂点のエッジデータ $v_k.e_{jk}$ の関係図を図 11 に示す。

- ストローク数最適解算手法 ($v_j, v_k.e_{jk}$)

for v_j . エッジデータ数

ストローク数=ストローク数計算 ($v_j.e_{ij}, v_k.e_{jk}$)

if ($v_k.e_{jk}.TotalStroke > \text{ストローク数}$) or

{ ($v_k.e_{jk}.TotalStroke = \text{ストローク数}$) and
($v_k.e_{jk}.TotalLength > v_j.e_{ij}.TotalLength + e_{jk}.Length$) }

then $v_k.e_{jk}.TotalStroke = \text{ストローク数}$

$v_k.e_{jk}.TotalLength = v_j.e_{ij}.TotalLength + e_{jk}.Length$

$v_k.e_{jk}.Before_nodeID = v_j.e_{ij}.Node_ID$

$v_k.e_{jk}.Before_linkID = v_j.e_{ij}.After_linkID$

$v_k.e_{jk}.Before_strokeID = v_j.e_{ij}.After_strokeID$

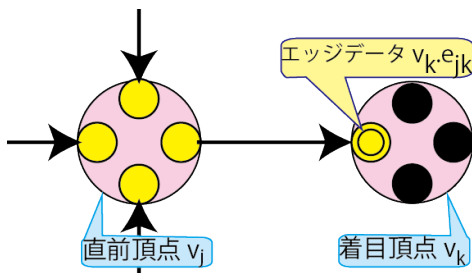


図 11 ストローク数最適解算手法

道なり経路探索のアルゴリズム SPMS 法のアルゴリズムは上記 2 つの手法を用いて出発点を持つエッジデータと隣接する頂

点を持つエッジデータから順に、ストローク数最少の中で最短距離となる経路を確定させ、経路探索するグラフ $G=(V,E)$ 内に存在する全ての頂点に対して経路探索を行う。ストローク数最少の中で最短距離であるエッジデータを持つ頂点に着目し、その頂点に隣接する頂点を持つエッジデータへ 2 つの手法よりストローク数最少の中で最短距離となるストローク数と経路長、直前の頂点を登録する。これを全てのエッジデータにストローク数最少の中で最短距離となるストローク数と経路長、直前頂点が登録されるまで繰り返し行う。探索経路は到着点を持つエッジデータの中で、ストローク数最少の中で最短経路長が登録されたエッジデータから、そのエッジデータに登録された直前頂点と直前頂点の接続エッジのリンク ID より直前頂点を持つエッジデータを辿ることと求める。 v_j は頂点、 $v_j.e_{ij}$ は v_j のエッジデータ、 $v_k.e_{jk}$ は v_j に隣接する頂点 v_k のエッジデータ、 v_s は出発点、TC は終了条件 ($G=(V,E)$ 上の全ての頂点を持つエッジデータに対するストローク数最適解を算出すること) として以下の擬似コードに示す。

Step1 初期化

for each v_j in $G(V,E)$

if $v_j = v_s$

then for each $v_j.e_{ij}$ in v_j

$v_j.e_{ij}.TotalStroke = 1$

$v_j.e_{ij}.TotalLength = 0$

else for each $v_j.e_{ij}$ in v_j

$v_j.e_{ij}.TotalStroke = \text{無限大}$

$v_j.e_{ij}.TotalLength = \text{無限大}$

Step2 本計算

While (TC=False) Do

$v_j = \text{ストローク数最少の内で最短経路長の}$

エッジデータの頂点

for v_j に隣接する頂点のエッジデータ $v_k.e_{jk}$

ストローク数最適解算手法 ($v_j, v_k.e_{jk}$)

5. プロトタイプシステム

提案システムを実現するために、以下のリソースを用いてプロトタイプシステムを実装した。

- JAVA 言語 [10]
- PostgreSQL [11]
- PostGIS [12]

5.1 プロトタイプシステムの概要

プロトタイプシステムの構成は提案システムそのものをプロトタイプシステムで実現したため、提案システムと同一である。地図データは OpenStreetMap[17] の地図データが利用された地図サーバから取得する。地図画像取得機能では、入力した緯度経度より地図サーバから地図画像を取得し、画像中心の緯度経度の座標を算出する。入力したストロークの基準長と緯度経度座標を用いて SNDB から範囲内に存在する SN の道路データを取得する。取得したストロークネットワークの道路データを用いて MSN を生成する。生成した MSN 上で SPMS 法より指定した 2 点間の最少ストローク数最短経路の探索を行う。

経路・地図表示機能で、SPMS 法における経路長、ストローク数、計算時間の結果および経路図と地図画像を表示する。また経路・地図表示機能には、SN と MSN のデータの表示機能を有する。

5.2 プロトタイプシステムの動作

プロトタイプシステムの操作画面を図 12 に示す。左部に地図画像を、右上部に入力インターフェースを、右下部に出力部を配置している。地図画像内で SN 範囲にする長方形の左上の角・右下の角の順に左クリックをすると地図内に枠が表示され、出発点・到着点の順に右クリックをすると地図内にマークが表示される。出力部には出発点と到着点の緯度経度が表示される。枠とマークが表示され、広域 SN と狭域 SN の各ストローク基準長を入力した状態でボタン「MSN 上で SPMS 法」を押すと、MSN 上で SPMS 法による経路探索を実行し、地図画像内に探索経路を、出力部に経路長とストローク数および計算時間が表示される。探索範囲は枠の範囲内である。



図 12 プロトタイプシステムの外観



図 13 MSN 上で SPMS 法経路探索

6. 評価実験

6.1 評価対象

・MSN(提案システム) 層数は 2 層で、各層の範囲は MCN と同一である。ストローク基準長は、狭域では出発点到着点に

隣接する道路のストローク長とし、広域は 1km, 3km, 5km, 10km とする。

・MCN(従来ネットワーク) MCN は、国道、県道など管理範囲の広さにより順序付けられた道路クラスに基づく階層的ネットワークである。層内の道路は基準の道路クラス以上の道路からなる。構築する範囲は MSN と同一で、狭域の基準道路クラスは出発点到着点に隣接する道路のものとする。広域の道路クラスは国道のみと国道～県道の 2 種とし、ストロークデータを付加する。

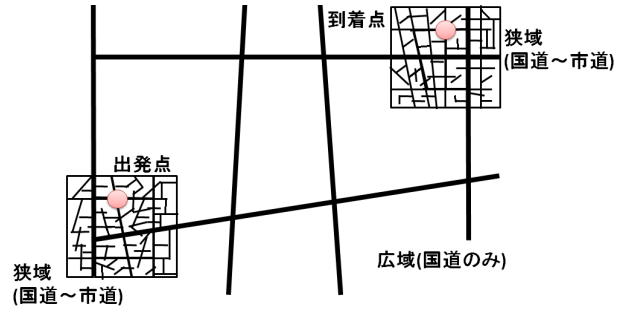


図 14 MCN

・ベースライン (全道路ネットワーク) ベースラインは、単層の全道路からなるネットワークである。構築する範囲は MSN および MCN の広域の範囲とする。

6.2 評価尺度

ネットワーク内のノード数、経路探索時間、ストローク数、経路長の差を評価尺度として用いる。

6.3 実験方法

直線距離の異なる 9 組の出発点と到着点と範囲を定め、上記 3 つのネットワークを構築し、ノード数を求める。構築した各ネットワーク上で SPMS 法により経路を探索する。

6.4 結果と考察

6.4.1 各ネットワーク内のノード数

MSN と MCN とベースラインのノード数を図 15 に示す。MSN はベースラインより約 10 分の 1 のノード数で構成されることがわかった。また、提案システムの MSN は MCN と同等以下のノード数で構成されることがわかった。これは、MSN の広域で指定した長大なストロークの道路が MCN の広域で指定した道路クラスの道路と同じであったためと考えられる。

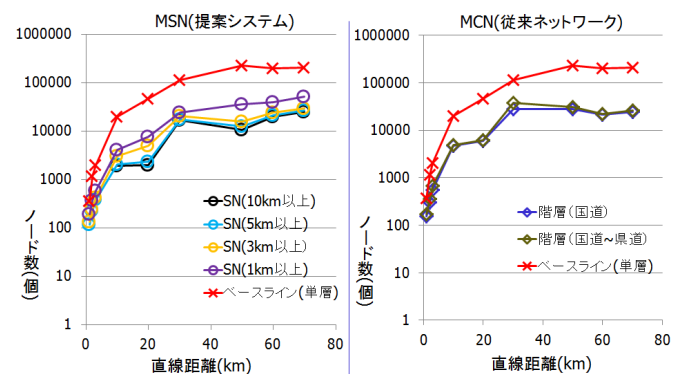


図 15 各ネットワーク内のノード数

6.4.2 経路探索時間

MSN と MCN とベースラインでの経路探索時間の結果を図 16 に示す。提案システムでは約 1000 秒以下の時間で探索でき、ベースラインより探索時間を約 100 分の 1 に抑えられることがわかった。また、提案システムはノード数が同等以下であった MCN より約 10 分の 1 に抑えられることがわかった。ノード数を減らすことで探索時間を大いに抑えられることがわかった。

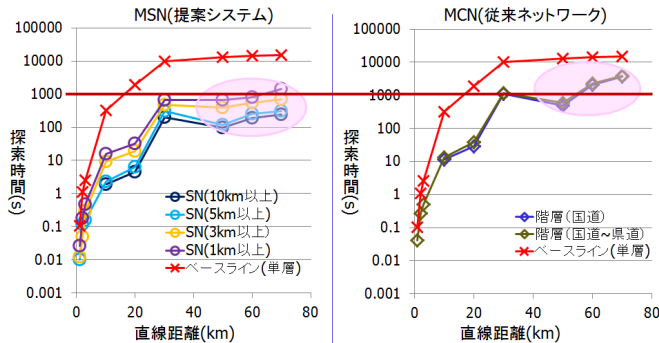


図 16 経路探索時間

6.4.3 経路上のストローク数

MSN と MCN とベースラインでの経路上のストローク数の結果を図 17 に示す。MSN(提案システム) はベースラインと全て同じになり、ストローク数の最適解が得られた。MCN は、ベースラインと半分の割合で同じになり、MSN は MCN より、最適解を得やすいことがわかった。

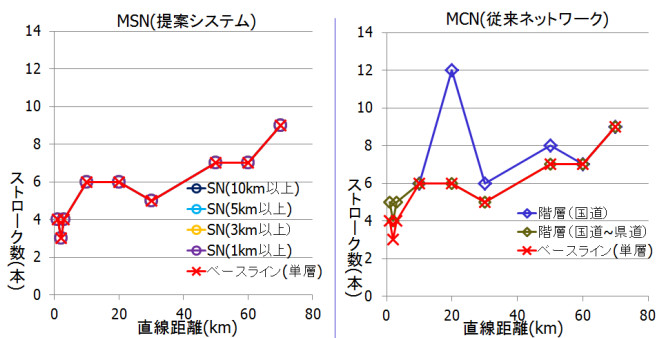


図 17 経路上のストローク数

6.4.4 経路長

MSN と MCN とベースラインでの経路長の結果を図 18 に示す。MSN(提案システム) はベースラインと全て同じになり、最適解が得られた。MCN は、ベースラインと半分の割合で同じになり、MSN は MCN より、最適解を得やすいことがわかった。

7. おわりに

本稿では、複数の SN を合成して構築した MSN 上でストローク数最少の経路の中から最短経路を探索するシステムを提案し、その実現法について述べた。また、提案システムに基づいてプロトタイプシステムを実装し、比較評価実験を行った。実験結果より、ネットワークにおいては MSN は MCN と同等以下のノード数で構成された。また、構築範囲が広がると広

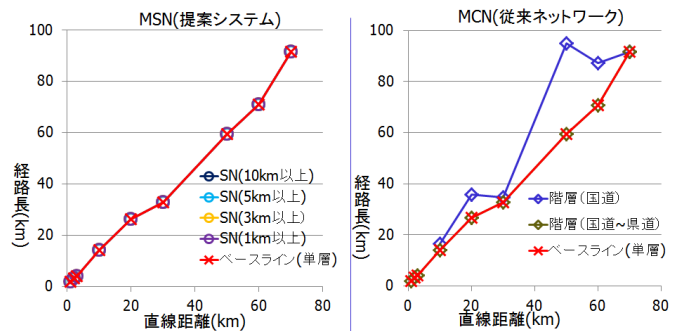


図 18 経路長

域のストローク基準長が大きくなるほど MCN のノード数に近いノード数で構成されることがわかった。経路探索においては、提案システムはベースラインと比較して約 100 分の 1 の探索時間で最少ストローク数最短経路が探索できることがわかった。また、提案システムは MCN と比較して約 10 分の 1 の探索時間でストローク数を約 10 % 抑えられることがわかった。MSN や MCN の広域と狭域のノード数の評価、層の数を増やしてノード数をさらに減らすことなどが今後の課題である。

文 献

- [1] Pablo MARTINEZ LERIN, Daisuke YAMAMOTO, Naohisa TAKAHASHI, "Encoding network-constrained travel trajectories using routing algorithms," International Journal of Knowledge and Web Intelligence, Vol.4, No.1, pp.34-49, May.2013.
- [2] D.Yamamoto, S.Ozeki, N.Takahashi, Focus+Glue+Context: An Improved Fisheye Approach for Web Map Services, Proceedings of the ACM SIGSPATIAL GIS 2009, pp.101-110, 2009.
- [3] 大西, 加藤, "交差点内コストを考慮した道路網における経路探索の手法とそのマルチメディア型経路案内システムへの応用", 情報処理学会論文誌, 33(7), 970-979, 1992
- [4] 原, 塚原, 狩野, "多目的遺伝的アルゴリズムを用いたカーナビゲーションのための予測経路探索", 情報処理学会高度交通システム研究会, ITS-24(5), pp.31-38, 2006
- [5] P.Sanders, D.Schultes, Engineering Fast Route Planning Algorithms, Experimental Algorithms (WEA), pp.23-36, 2007
- [6] 丹羽, 吉田, 福田, "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 31(5), 659-666, 1990
- [7] 新帯, 山本, 高橋, "道路ネットワークにおけるストロークに基づく経路探索法の実現と評価", DICOMO2015 シンポジウム論文集, 2015.
- [8] PostGIS 入門第 8 章
<http://workshops.boundlessgeo.com/postgis-intro-jp/geometries.html>
- [9] Robert C.Thomson, Dianne E.Richardson, The 'Good Continuation' Principle of Perceptual Organization applied to the Generalization of Road Networks, the 19th International Cartographic Conference, Ottawa, Canada, pp.1215-1223, 1999
- [10] Java
<https://www.java.com/ja/>
- [11] PostgreSQL
<http://www.postgresql.jp/>
- [12] PostGIS
<http://postgis.refractory.net/>