

曲線道路を滑らかに接続する道路ネットワーク整形手法

折原 照崇^{1,a)} 柳澤 政生¹ 戸川 望^{1,b)} 新田 知之² 進藤 大介² 田中 清貴²

概要: 近年, スマートフォンの普及により携帯電話に様々な利用形態が生まれている. よく利用される携帯電話機能の1つとして地図サービスがある. 携帯電話のような表示画面の狭い端末には地図に略地図を使用することで視認性を高められる. 略地図とは地理情報の中から余分な情報を取り除き, ユーザの必要な情報を分かりやすく提示するよう簡略化した地図である. 本稿では略地図に表示する曲線道路を滑らかに接続するための整形手法を提案する. 滑らかな接続は道路を一本の長いストリートとして抽出すること, ベジエ曲線による曲線化により実現する. 実験結果から提案手法は実データの道路ネットワークに適用可能であり, 曲線道路を滑らかに接続した略地図が生成可能であることを確認した.

キーワード: 略地図, 道路ネットワーク, 滑らかな接続, ベジエ曲線

A Curved Road-network Shaping Algorithm for Smoothly-connected Deformed Map Generation

TERUTAKA ORIHARA^{1,a)} MASAO YANAGISAWA¹ NOZOMU TOGAWA^{1,b)} TOMOYUKI NITTA²
DAISUKE SHINDOU² KIYOTAKA TANAKA²

Abstract: In recent years, mobile phones and smart phones are used in wide-variety of situations. Map service is one of the most useful mobile phone services, where a deformed map improves the visibility on a small screen. In this paper, we propose a curved road-network shaping algorithm for generating smoothly-connected deformed maps. Our algorithm is based on (1) extracting a long street and (2) generating curves using Bezier-curve parameters. Several experimental results show that our algorithm can be successfully applied to practical road network and obtains smoothly-connected deformed maps.

Keywords: deformed map, road network, smooth connection, bezier curve

1. はじめに

携帯電話は小型化・高性能化が進み, 音楽や動画の視聴, パソコン向けウェブサイトの閲覧等の機能が搭載され, 様々な利用形態が生まれている [1]. また近年, スマートフォンの登場により, 電話以外の機能の利用が広がっている. よく利用される携帯電話機能の1つとして地図サービスがある. 地図サービスを利用する上で, 携帯電話はパソコンに比べると, 表示画面が狭い. 表示する地図内に表示要素が多く存在すると, 狭い表示画面に視認性の悪い地図が表示される. この問題を解決する方法の1つに略地図生成がある. 略地図とは, 地理情報の中から余分な

情報を取り除き, ユーザに対して必要な情報を分かりやすく提示するよう簡略化した地図である. 略地図の多くはデザイナーにより手作業で作成されているが, GPS や GIS といった位置情報や地理情報を計算機上で扱うための技術の発達に伴い, 略地図生成自動化の手法が注目されてきた [2], [3], [4], [5], [6], [7].

略地図は2種類に分類できる. エリア全体の道路ネットワークの略地図と, 2地点間の経路を中心とした略地図である. 前者をエリア略地図, 後者を2地点間略地図と言う. エリア略地図は広告, パンフレット, 雑誌等に使用され, 1つの目的地の周辺エリア, またはエリア内の複数の目的地を表示したり, エリア内の道路の位置関係を把握したりする目的がある. 2地点間略地図は, 道案内のために使用され, 経路に関する移動に必要な情報のみを表示する. エリア略地図は2地点間略地図と比べて, 複数の道路が複雑に絡み合うために作成が難しい. 本稿ではエリア略地図に着目

¹ 早稲田大学

Waseda University

² ゼンリンデータコム

ZENRIN DataCom Co., Ltd.

a) terutaka.orihara@togawa.cs.waseda.ac.jp

b) togawa@togawa.cs.waseda.ac.jp

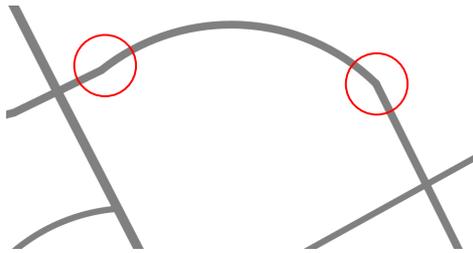


図 1 半円形置き換えによる問題.

Fig. 1 The problem with replacing the semi-circular.

する.

既存のエリア略地図生成手法には、道路形状正規化モデルによる簡略化手法 [2], [3], 道路ネットワーク分割による簡略化手法 [6] が提案されている. 道路形状正規化モデルによる簡略化は、ノードやリンク間に埋め込んだばねの作用により、道路形状を正規化する手法である. 全体のばねエネルギーが収束条件を満たすまで処理を繰り返す. 碁盤目のようなデザイン性の高い略地図が生成可能だが、収束のための繰り返し処理に時間がかかる. ノード数 100 程度のエリアの略地図生成にかかる時間は約 25 秒である. 道路ネットワーク分割による簡略化は、ノードを分割条件に基づき分割する. 分割条件に当てはまらないノードを間引くことで、ノードを削除して直線化されるリンクを増やし視認性の向上を図っている. ノード数 100 程度のエリア略地図生成にかかる時間は約 60 ミリ秒である. 略地図生成に何か収束を待つ繰り返し処理がなく、高速な略地図生成が可能である. しかし、分岐のない道路でしか曲線化できないという問題があるため、分岐の多く出現するエリア略地図では適用可能箇所が制限される. また、エリア略地図として全体にまとまりを与えることができていなかった.

以上の既存研究を踏まえてエリア略地図の高速生成が可能かつ視認性向上を図る手法として、特定形状を考慮したエリア略地図生成手法 [8] を提案した. [8] では、碁盤目、楕円形、半円形の 3 つの特定形状を考慮して変形したエリア略地図を生成した. アンケート調査の結果、碁盤目、楕円形、半円形の 3 つの特定形状の考慮による視認性の向上を確認した. しかし、一部出力結果ではリンク同士の接続部が滑らかに接続できず、視認性の悪い変形が起きた. 特に図 1 のように、半円形への置き換え箇所と直線の接続部の滑らかな接続が不可能であった.

本稿では [8] の問題点を説明した後、直線と曲線、曲線同士を滑らかに接続するための整形手法を提案する. 滑らかな接続は道路を一本の長いストリートとして抽出することと、ベジェ曲線による曲線化により実現する.

2. 略地図の自動生成

計算機で略地図を自動生成するには、道路ネットワークを構成するノード、リンク情報と、目印となるランドマーク情報を入力する. 道路ネットワークは無向グラフであり、

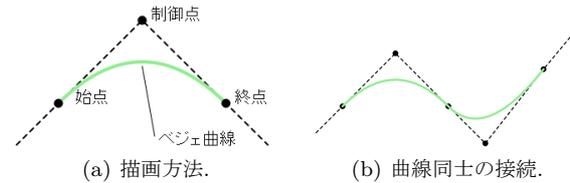


図 2 ベジェ曲線.

Fig. 2 Bezier curve.

それを構成するノードは座標情報を持ち、リンクはノードの接続関係情報を持つ. 出力は一部のノードとリンクを省略してデフォルメ化した道路ネットワークと、そこにランドマークを重ね合わせたものである. また、全ての道路ネットワークとランドマークを表示せずユーザにとって必要な道路とランドマークを選択する.

本研究は、ノードとリンクを省略してデフォルメ化することで、人間にとって見やすい道路ネットワークを高速で自動生成することを目指すものである. 略地図はユーザにとって必要な情報を見やすく分かりやすく提示する必要がある. これを複数の道路をまとまりあるものとして捉え整形することで見やすさの向上を図っている. また略地図が使用されるのは、ユーザがどこかへ向かおうとしている状況であると考えられる. 今すぐ目的地に向かいたい状況で、略地図を生成し提示するするのに多く時間を要することはできない. そのため、高速な略地図生成が求められる.

3. 曲線道路を滑らかに接続する道路ネットワーク整形手法

本章では、既存の略地図生成手法の問題点を述べた後、その問題を解決するための手法を提案する. 提案手法はストリート抽出、ストリート単位変形、構成ノード移動の 3 ステップで構成している.

3.1 既存研究 [8] の問題点とその解決方法

既存研究 [8] では道路ネットワークの曲線と判断した道路を半円形へ置き換えている. しかし、半円形への置き換えでは図 1 のように直線と半円形の道路が滑らかに接続せず折れ曲がっている. 探索により曲線と判定した箇所に半円形への置き換えは起こるが、ここで置き換える半円形は真円の弧である. しかし実際の道路ネットワークに存在する曲線は真円の弧と一致するとは限らないため、図 1 のように半円形と直線の接続がうまくいかない.

この問題点を解決するには、曲線の描き方を改善する必要がある. そこで、曲線と曲線、曲線と直線を滑らかに接続する方法としてベジェ曲線の性質を利用する. ベジェ曲線は始点 (x_1, y_1) 、制御点 (x_2, y_2) 、終点 (x_3, y_3) の 3 点と、以下の媒介変数表示で定義される.

$$x(t) = t^2 x_3 + 2t(1-t)x_2 + (1-t)^2 x_1 \quad (1)$$

$$y(t) = t^2 y_3 + 2t(1-t)y_2 + (1-t)^2 y_1 \quad (2)$$

ベジェ曲線は始点や終点に直線を滑らかに接続できる. 図 2(a) のようにベジェ曲線の始点と制御点を結んだ直線

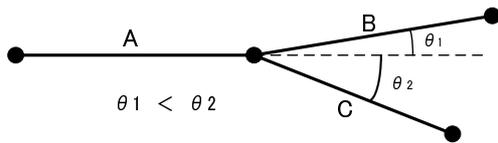


図 3 ストリート抽出.
Fig. 3 Street extraction.

の延長線は必ずベジェ曲線から滑らかに伸びる。ベジェ曲線は 3D のコンピュータグラフィクスに使用される [9] など、組み合わせ次第で様々な曲線を表現できる。また、図 2(b) のようにベジェ曲線同士も滑らかに接続可能で、S 字カーブのような道路も表現できる。

[6] の手法ではベジェ曲線により道路を曲線で表現している。しかし曲線化可能箇所は分岐のない一本道のみである。ストリート抽出により分岐のある道路を一本道と見なすことができれば、ベジェ曲線によって表現できる。そこで分岐のある道路においても曲線化可能にするために、ストリートの概念を導入する。ストリートとは、リンクの接続角が閾値以内で直線的に繋がっているリンク列を指す。ストリートの定義は文献 [4] の定義に従うと、以下の 3 つの条件を満たすリンク列をさす。

- (1) リンク間の外角が閾値以内である。
- (2) 分岐において直線性が最も優れているリンク間である。
- (3) リンクを共有しない。

3.2 節から 3.4 節では、ベジェ曲線とストリートを利用した提案手法を説明する。提案手法は、ストリート抽出、ストリート単位変形、構成ノード移動からなる。

3.2 ストリート抽出

ストリート抽出では、一本道の道路を抽出する。ストリートの 3 つの条件に加えて、提案手法では道路幅を考慮して、1 つのストリート内のリンクは全て道路幅が等しいとする。ストリート抽出方法を図 3 で説明する。まず、リンク A から見て外角が閾値以内で、かつ最小であるリンク B をストリート構成するリンクの候補に選ぶ。次に、リンク B から見てもリンク A は外角が最小であるため、リンク A とリンク B はストリート構成する要素とする。リンク A の左側、リンク B の右側のノードにおいて繋がるリンクが存在するならば、同様に条件を満たすリンクを抽出し、1 つのストリートが求まる。リンク C はリンク A をストリート構成するリンクの候補と選ぶが、リンク A はリンク B を候補と選ぶため、リンク C とリンク A は同じストリートにはならない。

また 1 つのストリート内は同じ道路幅を持つリンクのみで構成されるため、例えば図 3 の例でリンク A はリンク B と道路幅が異なるがリンク C とは道路幅が等しい場合、リンク A とリンク C が同じストリートになる。

3.3 ストリート単位変形

ストリート単位変形では、抽出したストリートごとに整

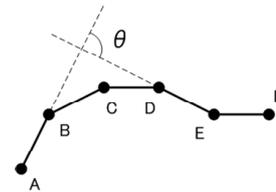


図 4 回転角度.
Fig. 4 Angle of Rotation.

形する。ストリート内のノードの位置関係から、リンクの回転方向と回転角度を求め、直線化する箇所と、ベジェ曲線で曲線化する箇所を決定する。

以下の手順でストリート内の直線と曲線を決定する。

- 手順 1. 始点ノードから順に回転方向が逆転する、または終点ノードに着くまでノードを巡る。
- 手順 2. 図 4 のように角度 θ を求め、 θ が閾値未満ならば直線、閾値以上なら曲線にする。
- 手順 3. 次の始点ノードを設定し手順 1 に戻る。

- 終点ノードに着いたならば手順 1 に戻らず終了する。
- 終点ノードに着いていないならば、初めて回転方向が逆転したノードの 2 個前のノードを始点とする。

ストリート内の直線と曲線の並び順によって、滑らかな接続のための変形を実行する。組み合わせは (1) 曲線 → 曲線, (2) 曲線 → 直線, (3) 直線 → 曲線, (4) 直線 → 直線の 4 つである。本手法では曲線の接続を考慮した手法であるため、(4) について考慮していない。よって 3.3.1 項 ~ 3.3.3 項で (4) 以外の 3 つの組み合わせごとの変形手法を説明する。

3.3.1 曲線 → 曲線

「曲線 → 曲線」の場合を図 5 で説明する。ストリートの端であるノード A を始点としてノード A からノード B、ノード B からノード C と順番に回転方向を調べる。ノード E からノード F で初めて回転方向が逆転する。ここで角度 θ_1 を求める。 θ_1 が閾値より大きければ曲線にし、閾値より小さければ直線にする。 θ_1 は閾値より大きいとするとノード A からノード E は曲線である。この曲線の制御点を図 5 のように 2 つの延長線の交わる点 CP_1 とする。

次に始点を変更して同様に回転方向を調べる。共有リンクを考慮してノード D を始点としてノード D からノード E、ノード E からノード F と順番に回転方向を調べる。ノード G からノード H で終点に着くため、角度 θ_2 を求める。 θ_2 は閾値より大きいとするとノード D からノード H は曲線である。この曲線の制御点を 2 つの延長線の交わる点 CP_2 とする。2 つの制御点 CP_1, CP_2 によって曲線と曲線を滑らかに接続できる。

3.3.2 曲線 → 直線

「曲線 → 直線」の場合を図 6 で説明する。「曲線 → 曲線」と同様に、ノード A からノード E は曲線となり、この曲線の制御点を図 6(a) のように 2 つの延長線の交わる点

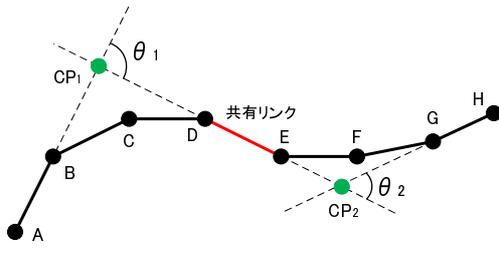
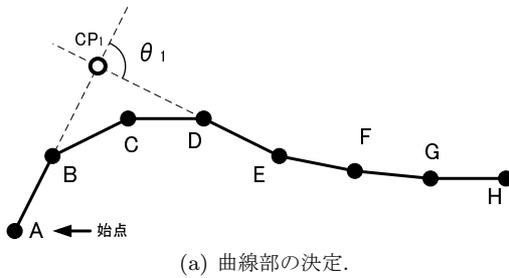
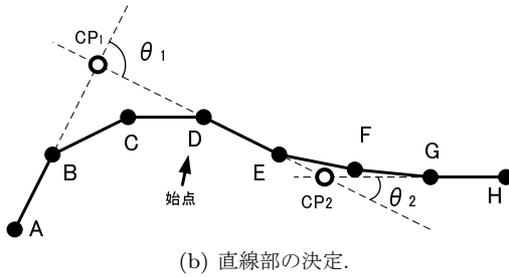


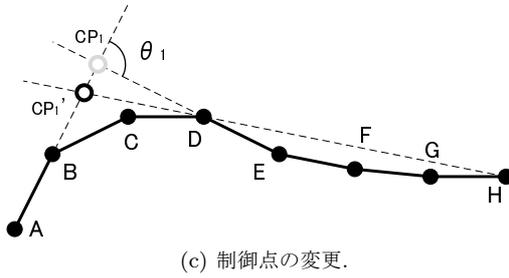
図 5 曲線から曲線の接続.
Fig. 5 Connection to the curve from the curve.



(a) 曲線部の決定.



(b) 直線部の決定.



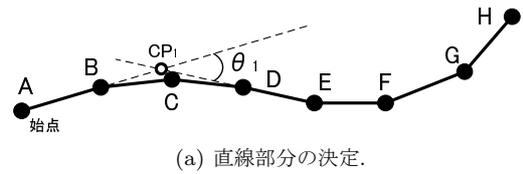
(c) 制御点の変更.

図 6 曲線から直線の接続.
Fig. 6 Connection to the straight line from the curve.

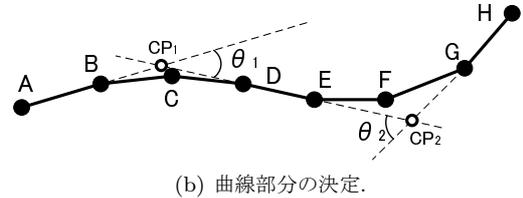
CP_1 とする. 次に図 6(b) のようにノード D を始点としてノード D からノード E, ノード E からノード F と順番に回転方向を調べる. ノード G からノード H で終点に着くため角度 θ_2 を求める. θ_2 は閾値より小さいとするとノード D からノード H は直線である. ノード D とノード H を繋ぎ直線にする. このままではノード D において曲線と直線の接続が滑らかでないため, 制御点を変更する. 新たな制御点はノード A, B の延長線とノード D, H の延長線の交わる点 CP'_1 とする. CP'_1 の位置を図 6(c) に示す. 制御点 CP'_1 によって曲線と直線を滑らかに接続できる.

3.3.3 直線 → 曲線

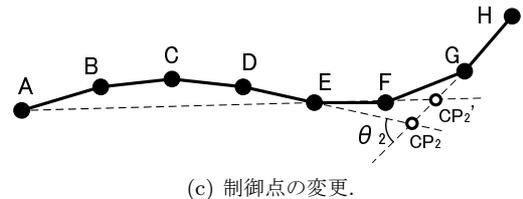
「直線 → 曲線」の場合を図 7 で説明する. ストリートの端であるノード A を始点としてノード A からノード B, ノード B からノード C と順番に回転方向を調べる. ノード E からノード F で初めて回転方向が逆転するため, 図 7(a) のように角度 θ_1 を求める. θ_1 は閾値より小さいとすると



(a) 直線部分の決定.



(b) 曲線部分の決定.



(c) 制御点の変更.

図 7 直線から曲線の接続.

Fig. 7 Connection to the curve from the straight line.

ノード A からノード E は直線である. 次に図 7(b) のようにノード D を始点としてノード D からノード E, ノード E からノード F と順番に回転方向を調べる. ノード G からノード H で終点に着くため角度 θ_2 を求める. θ_2 は閾値より大きいとするとノード D からノード G は曲線である.

滑らかに接続するには, 直線の延長線上に曲線の制御点がある必要があるため, 図 7(c) のように新たな制御点を CP'_2 とすると CP'_2 によって直線と曲線を滑らかに接続できる.

3.4 構成ノード移動

ストリート構成ノードの移動では, ノード位置を算出し移動する. ストリート単位変形後, ストリートとノード位置は一致しないため, ノード位置をストリート上へ移動し一致させる. 対象のノードは丁字路のノードである. 元の地図では丁字路になっている部分が 3.3 節の変形によって十字路になったり, 途切れてしまった場合の対処である.

丁字路には 2 つのストリートがある. 図 8 のように赤いストリートをストリート A, 青いストリートをストリート B とする. 移動するノードはストリート A の途中ノードであり, ストリート B の先頭ノードである. 以下のような場合分けでノードを移動する.

- ストリート A は直線
 - ストリート B は直線
 - A と B が交わるならば B を縮める (図 8(a)).
 - A と B が交わらないならば B を伸ばす (図 8(b)).
 - ストリート B は曲線
 - A と B が交わるならば B を縮める (図 8(c)).
 - A と B が交わらないならば B の先頭ノードと制御点を繋ぐ直線を伸ばす (図 8(d)).

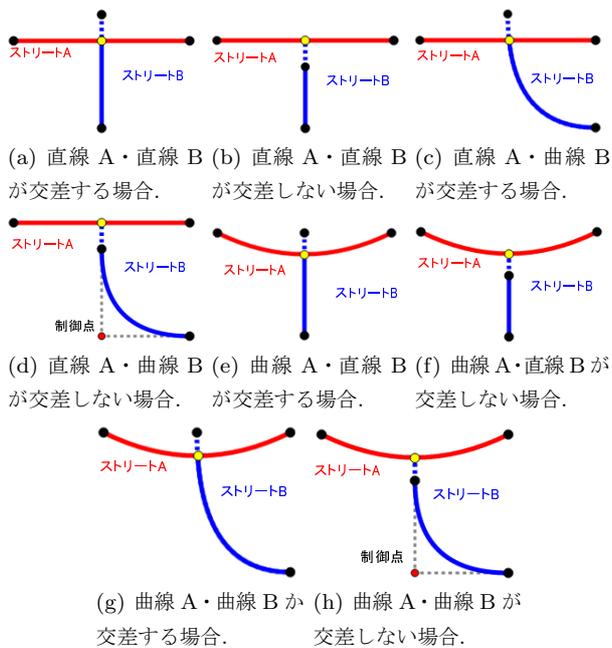


Fig. 8 Movement of the T-shaped path node.

- ストリート A は曲線
 - ストリート B は直線
A と B が交わるならば B を縮める (図 8(e)).
A と B が交わらないならば B を伸ばす (図 8(f)).
 - ストリート B は曲線
A と B が交わるならば B を縮める (図 8(g)).
A と B が交わらないならば B の先頭ノードと制御点を繋ぐ直線を伸ばす (図 8(h)).

丁字路はどの場合もストリート同士が交わるならば縮め、交わらないならば伸ばす。図 8(g) のようなベジェ曲線同士の交点は Bezier Clipping 法 [9] で求める。

4. 適用実験

3章で提案した手法を C 言語で実装して略地図の生成実験を行った。[8] では入力データに手作業で作成したノード・リンク情報を使用した。本研究では実データを使用して実験を行った。手作業で作成した入力データと実データによる入力データの違いは、地図上に打たれたノードの細かさや道路の本数である。手作業では無意識にノードを打つ点の間隔が一定になりがちであるのに対し、実データはノード間隔はまちまちであり、特に曲線の道路には数多くのノードが並んで敷き詰められている。また、実データは細かなどんな道でも拾ってデータ化している。

必要な範囲の実データから道路ネットワーク情報として、位置座標をノード情報とし、ノード同士の接続関係とその間の幅をリンク情報として抽出して CSV ファイルを用意した。この CSV ファイルを入力データとした。[10] よりストリート単位変形の閾値は 45 度とした。また出力は、曲線化処理後の道路ネットワークを描くのに必要とな

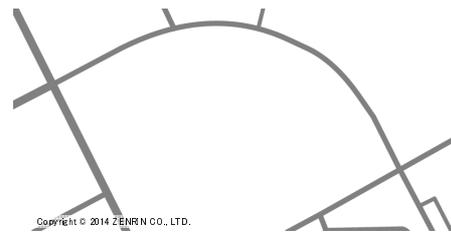


図 9 提案手法の適用結果.
Fig. 9 The result of applying the proposed method.

るノード、リンク情報の SVG ファイルである。

1章で問題とした地点での提案手法による略地図生成結果を図 9 に示す。提案手法の適用結果を主観的に目視で判断して、問題として挙げた曲線と直線の接続部が滑らかに接続できていることが分かる。表示されている道路の違いは、入力データが手作業のデータから実データに変わったこと、幅の狭い道路を取り除いて表示したことによるものである。略地図生成結果の別の例を図 10~12 に示す。それぞれ実データによる入力データと提案手法の適用結果である。本研究の目的である曲線の滑らかな接続が可能であることが分かる。変形処理にかかる時間は、図 10 の例で既存手法が約 250 ミリ秒、提案手法が約 60 ミリ秒、図 11 の例で既存手法が約 170 ミリ秒、提案手法が約 90 ミリ秒、図 12 の例で既存手法が約 140 ミリ秒、提案手法が約 200 ミリ秒であった。この結果から高速な生成を損なっていないことが分かる。

[10] より閾値は 45 度としてストリート抽出を行ったが、この閾値をより小さくすることで抽出するストリートは長くなる。また、変形では少しの曲がりやで曲線化が起こるようになる。

5. おわりに

本稿では曲線道路の滑らかな接続のための道路ネットワーク整形手法を提案した。ストリートの概念による長い道路の抽出と、ストリート単位でのベジェ曲線による曲線化により道路間の滑らかな接続を可能にする。適用実験では提案手法の実データへの適用が可能であることを確認した。一方、実データのノードの細かさゆえに思うような適用結果の表れていない箇所も見られたため、今後改善に生かせる結果を得られた。

謝辞 本研究を進めるにあたり貴重な議論及びデータを提供いただいた川上佳也氏をはじめとする株式会社ゼンリンデータコムのみなさま、ならびに株式会社ゼンリンのみなさまに感謝いたします。

参考文献

- [1] 岡葉子, “歩行者ナビゲーションのための高齢者と青年に対する歩行感覚実地調査,” 2011 年度卒業論文, 2011.
- [2] 丸山貴志子, 谷崎正明, 嶋田茂, “デフォルメマップ生成のための道路形状正規化モデルとそのシステム評価,” 電子情報通信学会論文誌, vol. J87-A, no. 1, pp. 108-119, 2004.

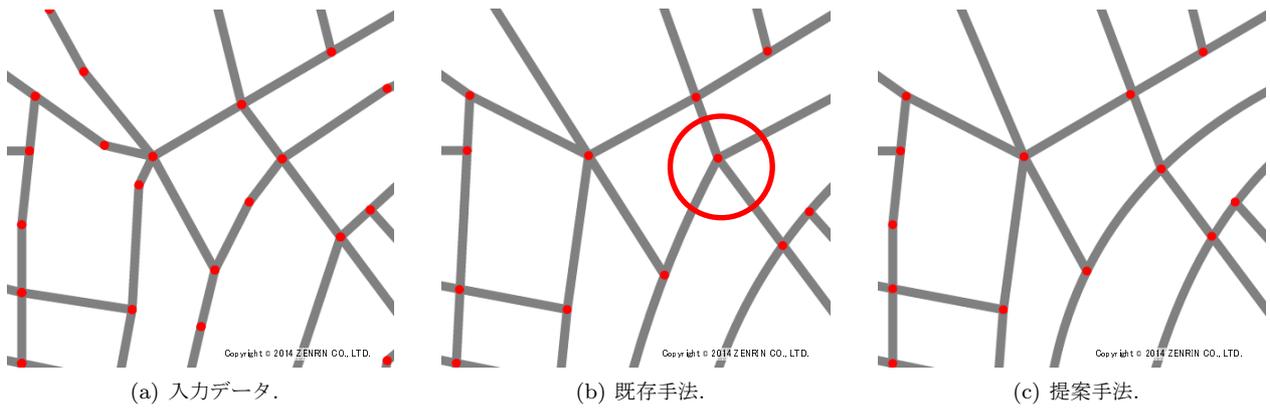


図 10 提案手法適用結果 (北千住).
Fig. 10 The result of applying the proposed method (Kitasenju).

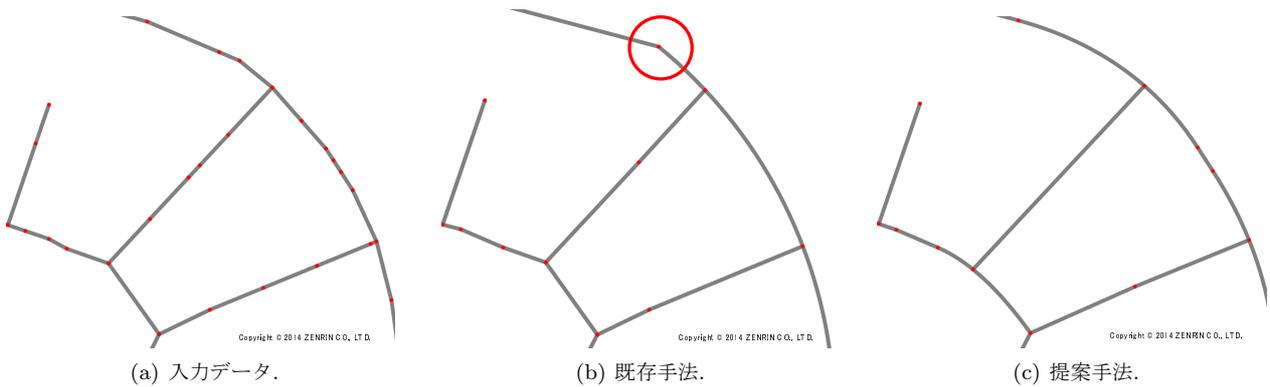


図 11 提案手法適用結果 (川崎).
Fig. 11 The result of applying the proposed method (Kawasaki).

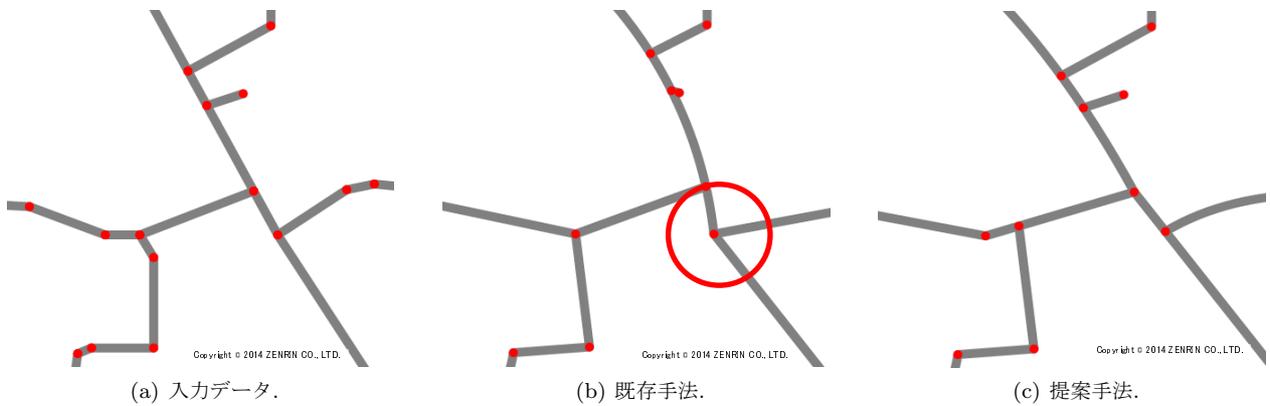


図 12 提案手法適用結果 (早稲田).
Fig. 12 The result of applying the proposed method (Waseda).

[3] 丸山貴志子, 谷崎正明, 嶋田茂, “デフォルメマップ生成のための道路変形モデルとそのシステム評価,” 情処研報, 2008-ITS-12, vol. 102, no. 695, pp. 77-84, 2003.

[4] 山守一徳, 本田宏, 長谷川純一, “ストリート単位の変形に基づく道路網の整形手法,” 電子情報通信学会論文誌, vol. J84-D-II, no. 9, pp. 2058-2069, 2001.

[5] 二宮直也, 戸川望, 柳澤政生, 大附辰夫, “歩行者ナビゲーションにおける微小画面での視認性とユーザの迷いにくさを考慮した略地図生成手法,” 信学技報, ITS2006-34, vol. 106, no. 266, pp. 53-58, 2006.

[6] 松本和也, 戸川望, 柳澤政生, 大附辰夫, “道路ネットワーク分割に基づく高速エリア略地図生成手法,” 信学技報, ITS2008-12, vol. 108, no. 171, pp. 25-30, 2008.

[7] 永田剛彦, 前田義信, “エージェントを用いた略地図にお

る地理情報選択モデルの提案,” ヒューマンインタフェースシンポジウム 2006, pp. 479-484, 2006.

[8] 折原照崇, 柳澤政生, 戸川望, “特定形状を考慮した視認性の良いエリア略地図生成手法,” マルチメディア, 分散, 協調とモバイル (DICOMO2013) シンポジウム, pp. 2036-2043, 2013.

[9] Tomoyuki Nishita, Thomas W. Sederberg, Masanori Kakimoto, “Ray tracing trimmed rational surface patches,” ACM SIGGRAPH Computer Graphics, vol. 24, no. 4, pp. 337-345, Aug. 1990.

[10] 山本直英, 岡部篤行, “曲がり角が一つある通路における定性的方向推論についての実験による分析,” CICS(Center for Spatial Information Science, University of Tokyo) Discussion Paper, no.38, 2001.