

POIを考慮した経路長指定の複数巡回経路探索手法

西村 天晴^{1,a)} 石川 和明² 高山 敏典^{2,b)} 柳澤 政生¹ 戸川 望^{1,c)}

概要: 地図アプリケーションの普及により, 経路探索が身近な機能となっている. それに伴い出発地から目的地までの迅速な移動経路以外に, ユーザの嗜好を考慮した経路や, 出発地に戻る巡回経路といった, 観光や散策での経路探索が注目されている. 特に散策での巡回経路ではユーザの検索興味対象であるPOI (Point of Interest) を寄り道したり, ユーザが満足する経路を選択できるように複数候補の巡回経路を提示することが望ましい. 体調や空き時間に応じて望まれる経路長も変わるため, 散策での巡回経路では, ユーザが経路長を指定したり, 同一地点の通過を避ける必要もある. これに対し, ユーザの嗜好を考慮した経路探索では非巡回経路の探索が多く, 巡回経路に対応していない. 加えて, 従来の巡回経路探索手法は, 指定経路長と同一地点の通過の回避の両方を満たさない. 本稿では, 同一地点の通過を避け多くのPOIを経由し, 指定経路長で巡回経路を複数個探索する手法を提案する. 提案手法では, 指定経路長に基づき巡回の基準点を複数個生成する. 基準点を配置することによって指定経路長かつ複数個の巡回経路を得られ, しかも迂回が少なく経由可能なPOIを多く経由する経路を探索できる. 加えて, 基準点の間の非巡回経路のつなぎ合わせとして巡回経路を得られ, 適当な重み付けで同一地点の通過も避けられる. 提案手法では, 従来手法と比較して, 経路長を指定経路長に近づけたうえで, 同一地点の通過回数を80%以上削減し, POI 経由数も約1.24倍に増加した.

1. はじめに

地図アプリケーションの普及により, 経路探索が身近な機能となっている. 経路探索では時間や距離に基づく迅速な移動経路の探索が主流であるが, 歩行者は安全性や快適性も重視することが指摘されている [1]. 特に, 散策では迅速な移動経路は適さず, 散策が行われる市街地や観光地では, 「楽しい」, 「にぎやか」などの主観的要望を満たす経路を選択することが望まれ [2], ユーザの嗜好を経路に反映することが重要となる. 加えて, 散策は宿泊地や駅, 集合場所を拠点とし, 出発地である拠点へ戻ることが多く, 巡回経路を探索する必要もある.

散策での歩行距離は性別や年齢などのユーザの属性や, 同伴者の有無, 天候, 時間帯, 散策目的などの多くの要素によって変化する [3]. また, 体調や空き時間などの状況に応じて, ユーザが望む歩行距離は大きく変化する. そのため, 散策での巡回経路ではユーザが任意に経路長を指定できる機能が必要となる. また, 既に通過した道をただ単に

戻る経路の重複や, 同一地点の繰り返しの通過は, ユーザに無意味な経路と認識される恐れがある. ユーザは経路の新規性も重視して歩行経路を選ぶ傾向にあり [4], 散策での巡回経路では同一地点の通過を避ける必要もある. さらに「楽しい」経路とするには, できるだけ多くのPOI (Point of Interest) を経由することが望まれる [5]. 加えて, 探索経路が1つだけでは, ユーザが満足する経路が提示されない可能性がある [6]. ユーザが複数の提示経路の中から満足する経路を選択できるように, 複数個の経路を提示する必要もある. 以上の点から, 散策での巡回経路では (a) ユーザが指定する経路長で, (b) 同一地点の通過を避け, (c) POI を多く経由する経路を, (d) 複数個提示する必要がある.

非巡回経路の探索では条件 (c) や条件 (d) を目的とした手法 [5, 7, 8] が多く提案されている. [5] ではPOIを1つ寄り道して目的地に至る経路のうち, 経路長が最短のものから k 番目に最短のものまでの経路を探索する. 経由するPOIは1つだけであるものの, 同一地点の通過を避け, POIを経由する経路を複数個探索でき, 条件 (b)~(d) を満たす. しかし, 指定経路長の経路を探索できず, 条件 (a) を満たさない. [7] では特定の複数POIをすべて経由する経路を探索する. そのため, 多くのPOIを経由する経路を生成でき, 条件 (c) を満たすが, 指定経路長の経路を探索できず, 同一地点の通過も避けられないうえ, 探索経路は

¹ 早稲田大学
Waseda University

² ゼンリンデータコム
ZENRIN DataCom

a) tensei.nishimura@togawa.cs.waseda.ac.jp

b) takayama@zenrin-datacom.net

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

1つのみで条件(c)以外を満たさない。[8]ではPOI候補の中から複数のPOIを経由する経路を時間制約とユーザ指定の重要度に基づいて探索するため、指定経路長でPOIを多く経由する経路を探索でき、条件(a),(c)を満たす。しかし、同一地点の通過を許容し、経路も1つのみしか探索しないため、条件(b),(d)を満たさない。また、これらは非巡回経路用の探索で、巡回経路の探索に対応しておらず、散策での巡回経路に適さない。

巡回経路の探索では、組合せ最適化問題に関する研究[9]が多く、実経路の探索を目的とする研究は少ない。実経路の探索を目的とする従来の巡回経路探索手法には、与えられた地点すべてを巡回する手法(全地点型)[10,11]と与えられた地点の中から複数地点に限定して巡回する手法(全候補列挙型)[12]の2種類がある。

全地点型：

[10,11]では、地図上の配置を利用して巡回順番を決定する。巡回地点をPOIにすることで多くのPOIを経由でき、条件(c)を満たし、巡回順番の決定後に経路を探索するため、同一地点の通過を避けられ、条件(b)が満たせる。しかし、与えられた全地点を巡回するため、指定経路長の経路にできず、巡回順番は一意に決定するため複数の巡回経路も探索できなく、条件(a),(d)を満たさない。

全候補列挙型：

[12]ではパーソナルナビゲーションシステムP-tourを提案している。P-tourではユーザがPOIに重要度の割り振りと時間制約を設定する。P-tourは予め確定しているPOI間の経路長をもとに、与えられた重要度と時間制約に基づく評価値を最大にする準最適解を遺伝的アルゴリズム[13]で導出する。POIを経由する指定経路長の経路を探索でき、導出の過程で複数の巡回経路も得られ、条件(a),(c),(d)を満たす。しかし、POIの個数が少ないと、指定経路長となる巡回地点の組合せが存在しない場合があり、条件(a)を満たさない可能性がある。探索時間もPOIの個数に依存し、探索対象の地図上にPOIが30個程度ある場合でも1つの経路の探索に約5秒かかり、複数の経路を探索するには実用的な時間ではない。また、予めPOI間の経路を確定しているため、同一地点の通過が避けられず、条件(b)を満たさない。

以上、従来手法で条件(a)~(d)をすべて満足するものは筆者らが知る限り存在せず、これらの条件を満足する巡回経路探索手法の構築が望まれる。

本稿では、ユーザ指定の経路長で、同一地点を避けながら、多くのPOIを経由する巡回経路を複数個探索する手法を提案する。提案手法は、まず指定経路長をもとに、出発地を頂点の1つとする正方形を考え、正方形の各頂点を基準点とする。このような正方形を複数生成し、各基準点

を通る経路を生成することで、指定経路長に近い複数個の巡回経路が探索でき、条件(a),(d)を満たす。さらに、2つの隣接する基準点において、迂回を抑えたうえで経由可能なPOIを探索する。POIを多く経由できる経路が探索可能となり、条件(c)を満たす。重複した経路を避けるため、経路に適当な重み付けをし、基準点やPOIを結ぶ最小重み経路を探索することで、同一地点の通過を避けられ、条件(b)を満たす。その結果、提案手法は指定経路長に近い経路長で、同一地点の通過を避け、多くのPOIを経由する巡回経路を複数個探索でき、条件(a)~(d)すべてを満たす。

本稿の貢献は以下の通りである。

- (1) 地図上の配置を利用して複数の基準点集合を生成することで、指定経路長の巡回経路を複数個探索する。また、隣接する基準点の間の経路を探索することで、巡回経路を非巡回経路のつなぎ合わせとして扱え、同一地点の通過を避けられる。
- (2) 2つの隣接する基準点の間の経路探索において、大きな迂回をすることなく経由可能なPOIを追加することで、POI密度が疎な場所でも多くのPOIを経由できる。
- (3) 従来手法と比較して、経路長を指定経路長に近づけたうえで、同一地点の通過回数を80%以上削減し、POI経由数も約1.24倍に増加した。その結果、ユーザの嗜好を反映する散策経路を自動生成することを可能とした。

2. 複数巡回経路探索問題

本章ではまず、複数巡回経路探索問題を定義する。 V を交差点を表すノードの集合、 E を道路を表すエッジの集合とし、道路ネットワーク N を $N = (V, E)$ で表す。出発ノード $s \in V$ を道路ネットワーク上のノードとし、経路探索の出発地とする。POIを学校や公園、寺院、レストランなどの場所や建物とし、代表点の位置を緯度経度で表す。あるPOI p に対して、 p の座標に最も近いノードを p に対するPOIノード $n(p) \in V$ とし、 $n(p)$ の通過を p の通過とみなす。POIノードの集合を $P \subseteq V$ で表す。

例1 図1に道路ネットワークの例を示す。点がノード、線がエッジ、長方形が建物を表し、赤色の点が出発ノード s を表す。黄色の印がPOIの代表点の緯度経度の位置を表し、直近の青色の点に対応するPOIノードを表す。□

このとき複数巡回経路探索問題を次のように定義する。

定義1 (複数巡回経路探索問題) 複数巡回経路探索問題とは道路ネットワーク $N = (V, E)$ 、出発ノード $s \in V$ 、POIノードの集合 $P \subseteq V$ 、指定経路長 l_R が与えられたとき、 s から s へ戻る複数個の巡回経路を探索する問題である。このとき、 l_R に近い経路長で、同一地点の通過回数は少なく、POIノードを多く経由する経路を探索する。□

例2 図2に巡回経路の例を示す。点がノードを表し、

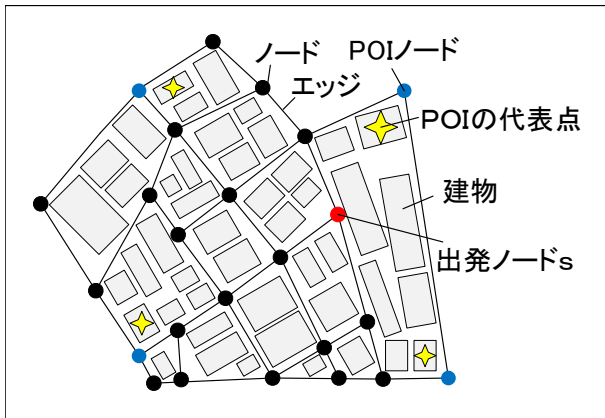


図 1 道路ネットワーク N の例.

赤色が出発ノード, 青色が POI ノードを表す. 線がエッジを表し, 緑色の線が経路を表す. 各エッジの長さは 100m とし, 指定経路長 $l_R = 800\text{m}$ とする. このとき, 図 2(a) は POI ノードを 3 つ経由し, 同一地点の通過もなく, 経路長は指定経路長で条件 (a)~(c) すべてを満たす. 一方で, 図 2(b) は経路長が指定経路長でなく, 図 2(c) は同一地点の通過が 1 回生じており, 図 2(d) は POI ノードの経由数が少なく, 条件 (a)~(c) すべてを満たしていない. □

複数巡回経路探索問題では, 図 2(a) のような条件 (a)~(c) すべてを満たす経路を複数個探索し, 条件 (a)~(d) すべてを満たすことを目的とする.

3. POI を考慮した経路長指定の複数巡回経路探索手法

3.1 方針

1 章で議論したように, 散策での巡回経路では (a) ユーザが指定する経路長で, (b) 同一地点の通過を避け, (c) POI を多く経由する経路を, (d) 複数個提示する必要がある.

条件 (a) は実際の利用場面で最も大きな制約条件となる. 散策はユーザの属性や目的, 時間帯, 天気などによって歩きたい距離が大きく変わり [3], 体調や空き時間, 予定に応じて散策可能な経路長に上限が生じる. 一方で, 条件 (b)~(d) はユーザの嗜好を配慮した条件である. 体力の限界やスケジュールなどに繋がる条件 (a) と比較して, 条件 (b)~(d) の優先度は低く, 条件 (a) が散策での巡回経路で最も重要となる. そこで, まず条件 (a) に着目する.

条件 (a) を満たす巡回経路を探索する方法として, 1 章で議論したように複数の巡回地点候補をすべて列挙し, このうちいくつかを経由する方法 (全候補列挙型) と, これとは異なるアプローチとして, 最初に巡回の基準点を設定し, 基準点を必ず経由する方法 (基準点設定型) の 2 つが考えられる.

全候補列挙型:

従来手法 [12] のように, このアプローチではまず, すべ

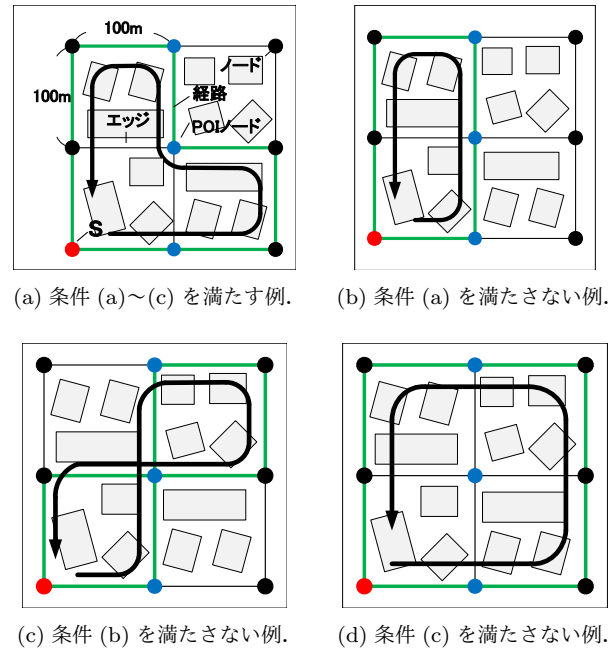


図 2 経路の例.

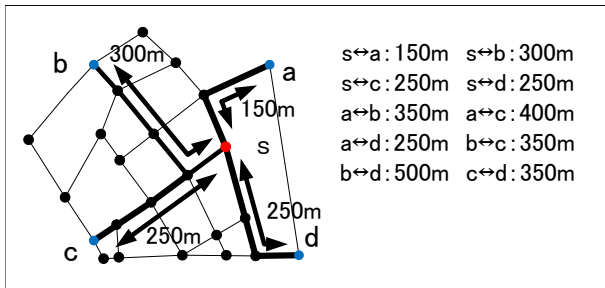
ての巡回地点の候補を列挙する. 次に, すべての候補地点間の経路を探索し, 各候補地点間の経路長を確定する. そして, 確定した経路と経路長を用い, 指定経路長で出発地に戻る巡回順番を導出する. 導出した巡回順番と確定している経路から指定距離の巡回経路を得る.

このアプローチは予めすべての候補地点間の経路を確定するため, これらを接続することで, 指定経路長の経路を作ることができる. しかし, 候補地点が多数あると多くの時間を要する. また, 予め経路を確定しているため, 巡回順番によっては同一地点の通過が生じ, 条件 (b) を満たさない可能性がある. POI を候補地点とすることで POI を経由できるが, POI 数が多いと, 探索時間が実用的な時間を超え, POI 数が少ないと, 条件 (a) を満たす経路が複数個存在しない可能性があり, 条件 (a),(d) を同時に満たすことは難しい.

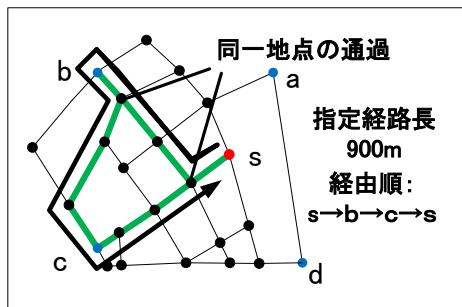
例 3 図 3(a) は全候補列挙型の巡回経路探索において, 候補地点としてすべての POI ノード a, b, c, d と出発ノード s を列挙した. その上で候補地点間の経路を確定した. s から各候補地点への経路を太線で表している. 他の候補地点間の経路も同様に探索し, すべての候補地点間の経路と経路長を確定している. 図 3(b) に指定経路長を 900m としたときの全候補列挙型の探索例を示す. 図 3(a) で確定した候補地点間の経路と経路長から, 指定経路長となる候補地点の巡回順番を求め, 巡回経路を生成している. 予め経路を確定しているため, 指定経路長に近い経路を探索できるが, 図 3(b) のような同一地点の通過が生じやすい. □

基準点設定型:

一方, 全候補列挙型とは逆のアプローチとして, 指定距離に基づいて基準点と基準点を通過する順序を決定するア



(a) 候補地点間の経路と経路長の確定.



(b) 指定距離 900m の全候補列挙型の探索例.

図 3 全候補列挙型の巡回経路探索.

アプローチを考える．基準点とその順序が決まれば，順番に基準点間の経路を探索できる．最後に経路をつなぎ合わせることで，指定距離の巡回経路を得る．

一般に 2 地点間の直線距離が増加すると 2 地点間の実経路長も増加するため，直線距離と経路長には相関がある．そこで隣接する 2 つの基準点の間隔を指定距離に基づいて決定することで，経路長が指定距離に近づくことが見込まれる．また，基準点の決定後に経路を探索するため，探索時に同一地点の通過を避けながら，迂回することなく経路可能な POI を経由することで条件 (b),(c) を満たす．加えて，最初の基準点の選び方を変えることで複数個の巡回経路が得られ，条件 (d) を満たす．

例 4 図 4 に基準点設定型の探索例を示す．まず，大きい点で表している基準点を指定経路長をもとに決定している．次に，基準点の間の経路を順番に探索する．緑色の線が経路を表し，同一地点の通過を避け，迂回を抑えて経路可能な POI を経由する経路を探索している． □

以上のように，全候補列挙型は同一地点の通過が生じやすいという必ずしも巡回経路に適した経路の生成が困難なのに対し，基準点設定型は基準点間の経路探索を工夫することで同一地点を避けながら多くの POI を経由できる巡回経路を生成できると考えられる．そのため，本稿では基準点設定型を採用し，複数個の基準点を設ける巡回経路探索手法を提案する．

3.2 提案手法の流れ

まず，提案手法の流れを示す．

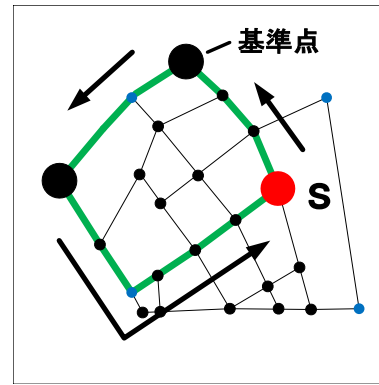


図 4 基準点設定型の巡回経路探索.

Step 1 道路ネットワーク $N = (V, E)$ と POI ノードの集合 $P \subseteq V$ からすべてのエッジ $e \in E$ に重みを定める．

Step 2 出発ノード $s \in V$ を基準点 p_0 とし，指定経路長 l_R をもとに p_0 を通る円を描く． p_0 を含む円周上の n 個のノードを基準点集合 $S = \{p_0, p_1, \dots, p_{n-1}\}$ とし，隣接する基準点の間を区間を $S_0 = p_0 \rightarrow p_1, S_1 = p_1 \rightarrow p_2, \dots, S_{n-1} = p_{n-1} \rightarrow p_0$ とする．

Step 3 区間 S_0 から順にすべての区間において，迂回を避けながら経路可能な POI ノードを経由し，同一地点の通過を避ける経路を探索する．

Step 4 Step 3 と逆順にすべての区間を探索し比較することで，経路を改善する．

Step 5 基準点を変えて Step 2～Step 4 を繰り返すことで複数の経路を生成する．

提案手法の流れを図 5 に示す．図 5(a) は Step 1 を表す． Step 1 ではすべてのエッジ e に， e の長さを重み $w(e)$ として与え， POI ノードに接続するエッジの重みを小さくする．図 5(a) では POI ノードに接続するエッジを細くして表している．図 5(b) は Step 2 を表す． Step 2 ではまず，出発ノード s を通る円を描き，円周上の s を含む n 個のノードを基準点とする．基準点と隣接する基準点の間の区間は s を基準とし，図 5(b) のように反時計回りで定める．図 5(c) は Step 3 を表す． Step 3 では迂回を避けて経路可能な POI ノードを経由し，同一地点の通過を避ける経路を探索する．各区間の経路を巡回経路の部分経路とし， s から反時計回りに探索する．図 5(c) では S_0, S_2 で POI ノード (青いノード) を経由し， S_1, S_3 で同一地点の通過を避ける経路が探索されている．図 5(d) は Step 4 を表す．図 5(d) では Step 3 と逆順の時計回りで経路を探索している．図 5(c) と図 5(d) を比較し，探索結果の良い経路を採用することで，経路全体を改善する．図 5(e) は Step 5 を表す． Step 5 では基準点を変えて Step 2～Step 4 を繰り返すことで，異なる巡回経路が得られる．

Step 5 は Step 2 から Step 4 の繰り返しのため，以下では Step 1 から Step 4 の各処理を提案する．

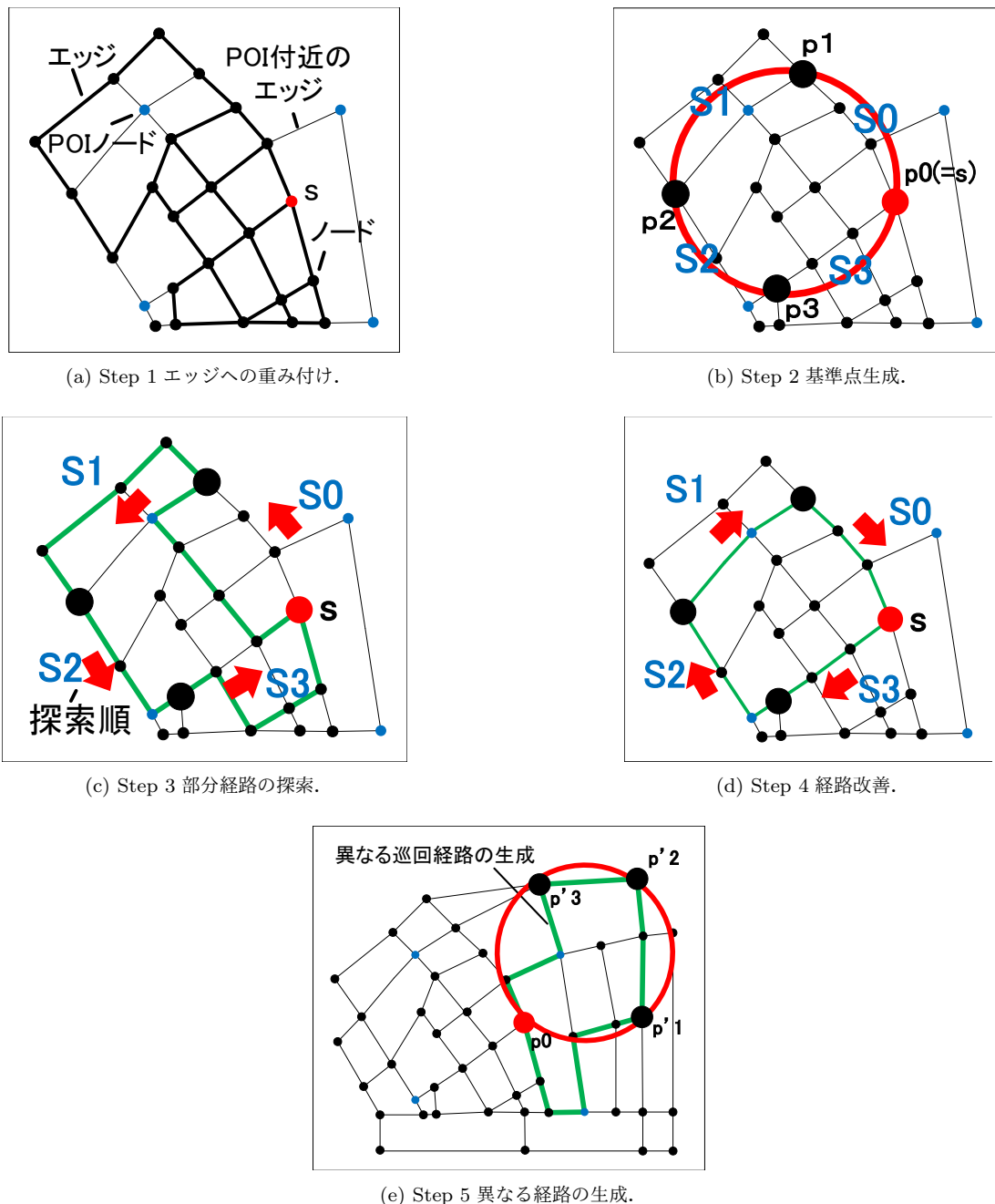


図 5 提案手法の流れ.

3.3 エッジへの重み付け (Step 1)

提案手法では、基準点間の最小重み経路を探索するため、最初にすべてのエッジ $e \in E$ に重みを付ける。 e の重み $w(e)$ は e の長さを用いる。 加えて、POI ノード付近を通過するとき、POI ノードを経由するように、POI ノード付近のエッジの重み $w(e)$ を小さくする。

提案手法では指定経路長と POI 経由数の 2 点から、予備実験を踏まえ、POI ノードに接続するエッジの重みを $0.2 \times w(e)$ 、POI ノードと隣接するノードに接続しかつ POI ノードに接続しないエッジの重みを $0.4 \times w(e)$ とした。

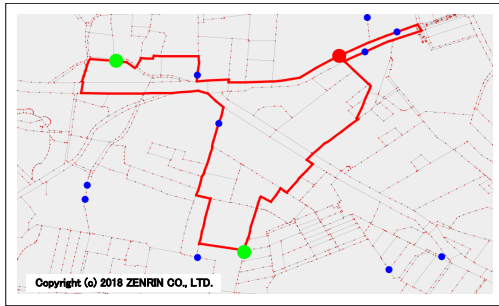
3.4 基準点生成 (Step 2)

提案手法は基準点を導入することで条件 (a)~(d) すべてを満たす経路が探索可能となる。

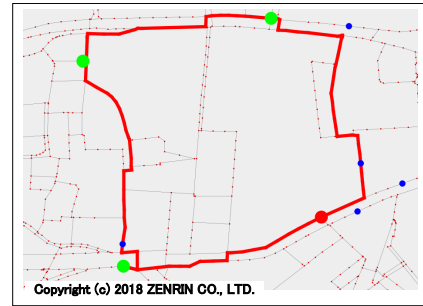
3.4.1 基準点数

提案手法では基準点数が重要となるため、基準点数 n を決めるため予備実験を実施した。 予備実験では n を 2~5 に設定し経路を探索した。 Step 1 は 3.3 節で述べたもの、Step 3 と Step 4 は以下に述べるものと同じのもの、Step 5 は 3.2 節で述べたものとし、Step 2 の基準点数 n のみ変化した。

表 1 に予備実験の結果を示す。 表 1 は指定経路長 l_R を 2000m とし、100 個の巡回経路を生成した時の平均の結果



(a) $n = 3$ での探索経路例.



(b) $n = 4$ での探索経路例.

図 6 $n = 3, 4$ での探索経路例.

表 1 基準点数 n を変更したときの探索結果.

探索場所	n	同一地点の通過回数	POI 経由数
新宿	2	3.72	5.19
	3	2.55	5.78
	4	1.39	6.13
	5	3.96	5.57
大阪	2	3.03	4.81
	3	1.21	5.95
	4	1.14	5.28
	5	1.27	3.94

である。新宿では、 $n = 4$ のとき、同一地点の通過回数が最小、POI 経由数は最大となり、 $n = 4$ が最適であった。大阪では、同一地点の通過回数は $n = 4$ のとき最小、POI 経由数は $n = 3$ のとき最大となり、 $n = 3, 4$ が適していた。新宿、大阪以外の日本各地でも、同様に予備実験を行った結果、 $n = 3, 4$ が適当であることが分かった。

$n = 3, 4$ のときの探索経路の例を図 6 に示す。点がノード、線がエッジを表し、赤色の太線が経路を表す。大きい点のうち赤色が出発ノード、青色が POI ノード、緑色が出発ノード以外の基準点ノードを表す。 $n = 3$ の場合、探索経路は三角形状に近く、探索領域が狭い。 $n = 4$ の場合、探索領域は四角形状となって、経路が円状に近づき、探索領域も広がる。探索領域が狭いと、近場のみを巡回する経路となりやすいため、提案手法では円状の巡回経路を生成しやすい $n = 4$ とした。

3.4.2 基準点生成手法

前項の議論のもと、基準点生成では 4 個の基準点の集合を複数生成する。まず、出発ノード s を通過し、指定経路長 l_R に対し円周の長さが $k_1 \times l_R$ の円 C を考える。 k_1 の値を適当に設定し、円 C 上に 4 つの基準点をとることで、結果的に経路長が l_R に近づくことが期待される*1。基準点の間隔が狭いと経由可能な POI ノードが見つかりにくくなるため、4 つの基準点は円 C の円周上に等間隔で位置することが望ましい。そこで円 C に内接する正方形で頂点の 1 つが s となるものを考えると、正方形の一辺の長さ r

*1 k_1 の値は予備実験を踏まえて $k_1 = 0.75$ としている。

は式 (1) となる。

$$r = k_1 \frac{l_R}{\sqrt{2}\pi} \quad (1)$$

図 7(a) に円 C と円 C に内接し s を 1 つの頂点とする一辺 r の正方形を示す。この正方形の 4 つの頂点を基準点として、基準点の間の経路を求めれば、その経路長は l_R に近づくものと期待される。

今、 s を中心とする半径 r の円 C' 上の任意のノード p_1 を考え、 sp_1 を一辺とする正方形を描けば、この正方形は s を通り円周 l_R の円 C の内接正方形となる。図 7(b) に円 C' と sp_1 を一辺とし円 C に内接する正方形を示す。円 C' 上のノード p'_1 においても同様に正方形を描いた様子を図 7(c) に示す。円 C' 上のノードの選び方で、円周の長さが $k_1 \times l_R$ の円に内接しかつ s を 1 つの頂点とする正方形が複数生成できる。これらの正方形の 4 つの頂点に最も近いノードを基準点とすれば、複数個の指定経路長 l_R の巡回経路を生成でき、条件 (a),(d) を満たす。また、基準点によって巡回経路を非巡回経路の集合として扱えるため、Step 3 を適用でき、条件 (b),(c) も満たすことが可能となる。

以下に基準点生成のアルゴリズムを示す。

Step 2.1 出発ノード s を基準点 p_0 とし、 p_0 から距離 r^{*2} のノードの集合を P_1 とする。

Step 2.2 P_1 からランダムでノードを 1 つ選び、基準点 p_1 とする。

Step 2.3 p_0p_1 を一辺とする正方形の各頂点に最も近いノードを基準点集合 $S = \{p_0, p_1, p_2, p_3\}$ とし、隣接する基準点の間の区間を S_0, S_1, S_2, S_3 とする。ただし、基準点と区間は p_0 から反時計回りで定める。

3.5 部分経路の探索 (Step 3)

まず、隣接する 2 つの基準点 p_0 と p_1 の間の区間 S_0 の経路を探索する。このとき、迂回を避けて経由可能な POI ノードを探して経由し、同一地点の通過を避ける経路を探索する。

*2 許容誤差を 20m としており、 P_1 は複数個のノードの集合となる。

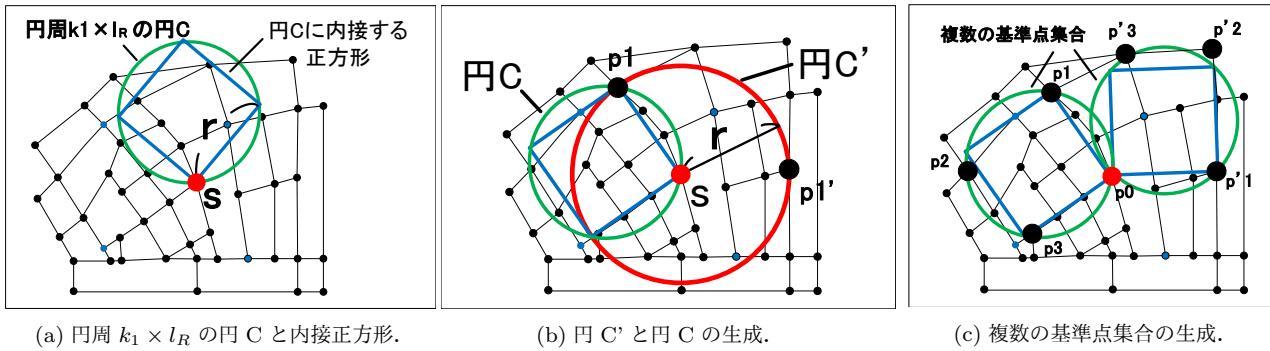


図 7 複数の基準点集合の生成例.

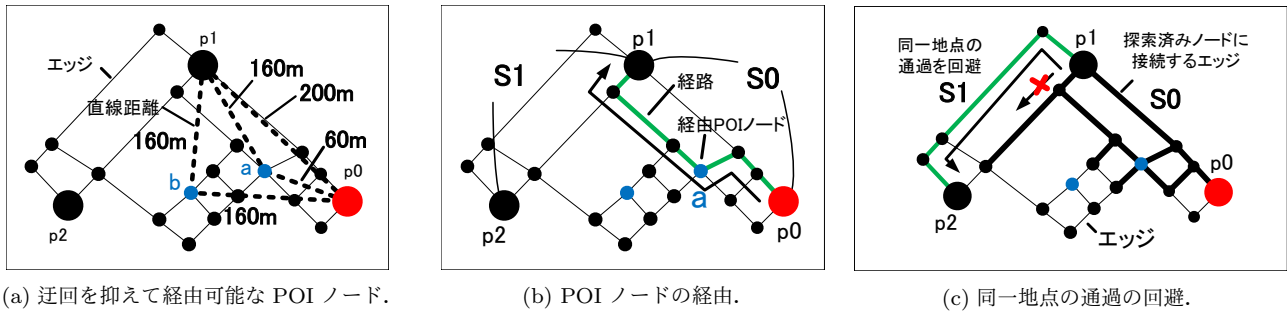


図 8 部分経路の探索.

2 地点 a, b の直線距離を $l(a, b)$ で表す. このとき, $i = 0$ において以下の式を満足し, かつ左辺の値が最小となる未経路の POI ノードを $p \in P$ とする.

$$l(p_i, p) + l(p, p_{i+1}) < k_2 \times l(p_i, p_{i+1}) \quad (2)$$

ここで, k_2 の値は POI ノードを通過することで, どの程度経路が迂回するかを表す係数で, 予備実験を踏まえ, $k_2 = 1.2$ とした. そして, p_0 から p , p から p_1 に至る最小重み経路を探索することで, 区間 S_0 の経路を求める. このとき, Step 1 にて, POI ノードに近いエッジには比較的小さい重みを与えているため, POI ノード p 以外にも, 複数の POI ノードを通過することが期待される.

また, 一旦経路を生成した後, 探索済みノードに接続するエッジの重みを α ($\alpha > 1$) 倍にすることで同一地点の通過を避ける. α の値が大きいと, 同一地点の通過を避けられるが大きな迂回が生じやすくなる. 一方で, α が 1 に近いと, 大きな迂回が生じにくいが同一地点の通過が生じやすくなる. 同一地点の通過と大きな迂回の生じにくさの 2 点から, 提案手法では予備実験を踏まえ, $\alpha = 10$ とした.

同様に, 区間 S_1, S_2, S_3 についても順番に経路を探索することで, 同一地点の通過を避け, POI を多く経由する経路が得られ, 条件 (b), (c) を満たす.

例 5 図 8 に部分経路の探索の例を示す. 図 8(a) は区間 $S_0 = p_0 \rightarrow p_1$ において, 直線距離を用い, 迂回を抑えて経路可能な POI ノードを探している. 線がエッジ, 点線が直線距離を表す. この場合, POI ノードを a あるいは b と

したとき, 式 (2) の左辺はそれぞれ $60 + 160 = 220m$, $160 + 160 = 320m$ となる. 式 (2) の右辺は両方とも $k_2 \times 200 = 1.2 \times 200 = 240m$ となる. POI ノードを a としたときのみ式 (2) を満たす. 図 8(b) では a を経由し, p_0 から p_1 に至る最小重み経路を生成している. 図 8(c) では図 8(b) に示した S_0 を対象とした経路探索後に S_1 を対象とした経路探索を行っている. S_0 で探索済みのノードに接続するエッジの重みが α 倍となり, 黒い太線で表している. エッジに α 倍の重み付けをすることによって S_1 では同一地点の通過を避ける経路が探索される. □

以下に p_i, p_{i+1} の間の区間 S_i における経路の探索アルゴリズムを示す.

- Step 3.1** 式 (2) を満足し, 式 (2) の左辺が最小となる未経路の POI ノード p を求める.
- Step 3.2** p が存在する場合は p_i から p を経由して p_{i+1} に至る最小重み経路を, 存在しない場合は p_i から p_{i+1} に至る最小重み経路を探索する.
- Step 3.3** 探索済みのノードに接続するすべてのエッジ e の重み $w(e)$ を $\alpha \times w(e)$ に変更する.

3.6 経路改善 (Step 4)

Step 3 で順番に探索される部分経路を最後につなぎ合わせることで, 巡回経路を得る. しかし, 各部分経路の探索終了時に, 探索済みノードに接続するエッジに対して重みを変更するため, 部分経路の探索順番によって, 探索結果が変わる.

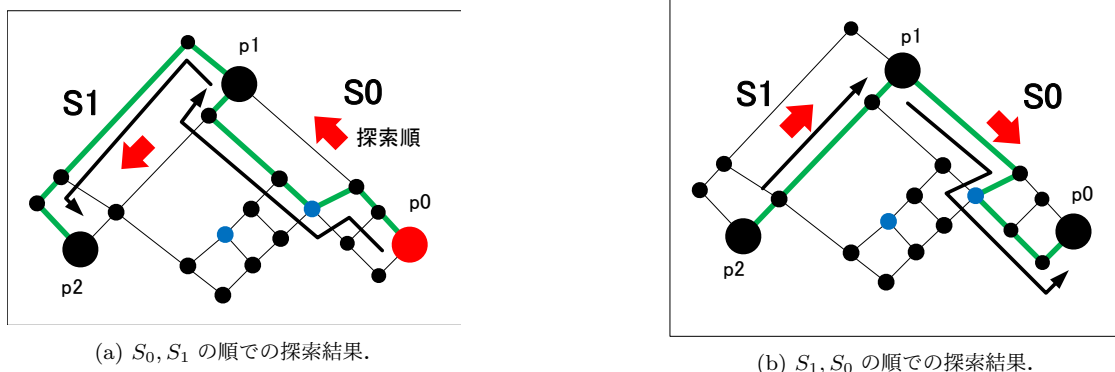


図 9 探索順による探索結果の変化.

表 2 探索条件.

探索場所	全ノード数	全 POI 数	全エッジ数	広さ [km ²]
大阪	16542	157	20672	2.32 × 2.84
練馬	8006	65	9384	2.32 × 2.84

図 9(a) に区間 S_0, S_1 の順, 図 9(b) に図 9(a) と逆順の区間 S_1, S_0 の順で探索して得られる経路の例を示す. 図 9(a) では S_0 で探索済みのノードを避けるために, S_1 で大きな迂回が生じている. 図 9(b) でも S_1 で探索済みのノードを避けるために S_0 で迂回が生じているが, 図 9(a) より迂回による経路長の増加が小さく, 探索順によって探索結果が変化している.

探索順により, 連続 2 区間の経路は変化するため, Step 3 の反時計回りでの探索経路と, 逆順の時計回りでの探索経路を比較し, 探索結果の良い経路を採用する.

経路改善手法ではまず, 連続 2 区間 S_0 と S_1 に着目し, 以下のアルゴリズムを適用する.

Step 4.1 S_0 と S_1 の経路情報を削除する.

Step 4.2 Step 3 と逆順の S_1, S_0 の順で探索する.

Step 4.3 もとの経路と比較し, 指定経路長から遠ざからず, 同一地点の通過回数も増えず, POI 経由数が減らない場合, 逆順の経路を採用し, それ以外はもとの経路を採用する.

同様に S_1 と S_2 , S_2 と S_3 , S_3 と S_0 においても実行し, 巡回経路全体を改善する.

4. 評価実験

4.1 実験条件

提案手法を CPU が intel CORE i7, メモリが 16GB の PC 端末に Java 言語で実装し, 評価実験を行った. POI 密度による探索結果の変化を調べるため, 探索場所は, POI 密度が密な大阪の市街地と POI 密度が疎な練馬の市街地から十分離れた場所とした. 表 2 に探索場所の条件を示す. 指定経路長を 2000m とし, 出発ノードを 1 つ指定し, 100 個の巡回経路を生成した.

提案手法を評価するため, 2 つの手法を比較した.

(1) 基準点間の最短経路探索手法 (比較手法 1)

提案手法の Step 2 と同様にして基準点を 4 つ設定し, 区間 S_0, S_1, S_2, S_3 を定める. 区間 S_0 から順番に最短経路探索し, 最後にすべての区間の探索経路をつなぎ合わせる.

(2) 基準点間の最短寄り道経路探索手法 [5] (比較手法 2)

提案手法の Step 2 と同様にして基準点を 4 つ設定し, 区間 S_0, S_1, S_2, S_3 を定める. 区間 S_0 から順番に最短寄り道経路探索 [5] を適用し, 最後にすべての区間の探索経路をつなぎ合わせる.

なお, 指定距離の巡回経路を探索する従来手法に P-tour [12] がある. しかし, P-tour は 1 つの経路を探索するのに約 5 秒かかり, 複数個の経路を探索するには実用的な時間ではない. 加えて, 実際の経路よりも POI の経由スケジュールの作成を目的としているため, 同一地点の通過が多く生じる経路が探索される. さらに, POI の個数によって指定経路長の巡回経路を探索できない可能性もある. 以上の点から, 従来手法との比較では, P-tour との比較はせず, 基準点間の最短経路探索と最短寄り道経路探索 [5] を採用している.

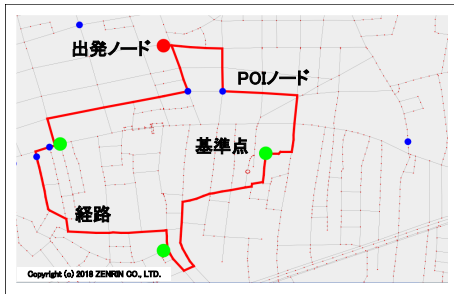
4.2 比較結果

表 3 に提案手法と比較手法の探索結果を示す. ここで, 経路長, 同一地点の通過回数, POI 経由数, 探索時間は生成された 100 個の巡回経路の長さ, 同一地点の通過回数, POI 経由数, 探索時間のそれぞれの平均を表す.

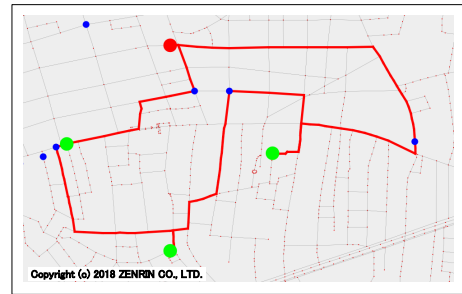
まず, 提案手法と比較手法 1 を比較する. POI 密度が密な大阪では, 提案手法は比較手法 1 と比べ, 経路長を指定経路長に近づけたうえで, 同一地点の通過回数を約 60% 削減し, POI 経由数も約 6.00 倍に増加した. また, POI 密度が疎な練馬でも同様に経路長, 同一地点の通過回数, POI 経由数すべてにおいて, 提案手法は比較手法 1 より良い結果が得られた. 提案手法の探索時間は比較手法 1 より長いですが, 提案手法も探索時間は十分に短い. 提案手法は条件 (a)

表 3 提案手法と比較手法の探索結果.

探索場所	手法	指定経路長 [m]	生成された経路数	経路長 [m]	同一地点の通過回数	POI 経由数	探索時間 [s]
大阪	提案手法	2000	100	2001.7	1.14	5.28	0.15
	比較手法 1	2000	100	1711.2	2.76	0.88	0.08
	比較手法 2 [5]	2000	100	1939.7	7.66	4.24	0.20
練馬	提案手法	2000	100	2039.4	0.98	2.81	0.05
	比較手法 1	2000	100	1677.4	6.54	1.39	0.02
	比較手法 2 [5]	2000	100	2540.2	17.77	4.14	0.07



(a) 提案手法の探索経路例.



(b) 比較手法 2 の探索経路例.

図 10 練馬における探索経路の比較.

~(c) すべてにおいて比較手法 1 よりも優れる.

次に提案手法と比較手法 2 を比較する. POI 密度が密な大阪では, 提案手法の方が指定経路長に近づけたうえで, 同一地点の通過回数を 80%以上削減し, POI 経由数も約 1.24 倍に増加した. POI 密度が疎な練馬では, 提案手法の POI 経由数は比較手法 2 の約 70%と少なくなったが, 比較手法 2 の経路長は指定経路長から 500m 以上離れ, 同一地点の通過回数は提案手法の約 18 倍となった. また, 提案手法の探索時間は比較手法 2 よりも短かった.

図 10 に同一の基準点集合を与えたとき, 練馬における提案手法と比較手法 2 の探索経路の 1 つを示す. 図 10 において POI 経由数は同じであるが, 比較手法 2 では, 基準点間で必ず 1 つの POI を経由するため, 同一地点の通過回数と経路長が大きくなっている. 一方で提案手法は, 基準点間で大きな迂回を避けて経由できる POI のみを経由している. 加えて, POI 付近のエッジの重みを小さくしているため, POI 付近を通過する経路の生成時, 未経由の POI を小さい迂回で経由し, 多くの POI を経由する.

大阪と練馬の POI 密度を考える. 大阪の POI 密度は $157/(2.32 \times 2.84) = 23.8\text{km}^{-2}$ であるのに対し, 練馬の POI 密度は $65/(2.32 \times 2.84) = 9.9\text{km}^{-2}$ で, 練馬は大阪の POI 密度の約 40%である. 練馬における提案手法の POI 経由数は約 2.8 個で比較手法 2 と比べると少ない. しかし, 練馬は大阪の約 40%の POI 密度であるのに対し, 練馬における提案手法の POI 経由数は大阪における提案手法の POI 経由数の約 50%となっている. 探索場所の POI 密度を考慮すると, 練馬における提案手法の POI 経由数は十分である. 提案手法は POI 密度に応じて経路を探索できており, 探索場所の POI 密度に依らず, 条件 (a)~(d) すべてを満

たす.

5. おわりに

本稿では, POI を考慮した経路長指定の複数巡回経路探索手法を提案した. 提案手法では同一地点の通過を避け, 多くの POI を経由し, 指定経路長に近い経路長で, 巡回経路を複数個探索する. 提案手法では, 従来手法と比較して, 経路長を指定経路長に近づけたうえで, 同一地点の通過回数を 80%以上削減し, POI 経由数も約 1.24 倍に増加した. 加えて, 提案手法は POI 密度に依存せず, 様々な場所で良い探索結果が得られることが分かった. 今後は, POI を抽出して特定の POI を優先する経路や, 道幅や曲がる回数を考慮した道に迷いにくい経路も探索できるようにする予定である.

参考文献

- [1] 松田美恵子, 杉山博史, 土井美和子, “歩行者の経路への嗜好を反映した経路生成,” 電子情報通信学会論文誌, vol. J87-A, no. 1, pp. 132-139, 2004.
- [2] 赤坂優太, 鬼沢武久, “経路選択に個人の好みを反映させた歩行者ナビゲーション-ファジィ測度・積分モデルの妥当性評価-,” 日本知能情報ファジィ学会誌, vol. 18, no. 6, pp. 900-910, 2006.
- [3] 外井哲志, 坂本紘二, 井上信昭, 中村宏, 根本敏則, “散歩経路の道路特性に関する分析,” 土木計画研究・論文集, no. 14, pp. 791-798, 1997.
- [4] 木村一裕, 清水浩志郎, 土田裕子, “散歩に利用された道からみた快適な歩行環境に関する考察,” 土木計画学研究・講演集, no. 18, pp. 289-292, 1995.
- [5] 大沢裕, 藤野和久, “前処理を必要としない道路ネットワーク上での最短寄り道経路探索アルゴリズム,” 電子情報通信学会論文誌, vol. J93-D, no. 3, pp. 203-210, 2010.
- [6] 稲垣潤, 長谷山美紀, 北島秀夫, “遺伝的アルゴリズムを用

- いた経路探索における複数経路候補の決定法,” 電子情報通信学会論文誌, vol. J82-D-I, no. 8, pp. 1102–1111, 1999.
- [7] 稲垣潤, 長谷山美紀, 北島秀夫, “遺伝的アルゴリズムを用いた複数経由点を伴う経路探索,” 電子情報通信学会論文誌 D-I, vol. J83-D-I, no.5, pp. 504–507, 2000.
- [8] Atsushi Maruyama, Naoko Shibata, Yoshihiro Murata, Keiichi Yasumoto, Minoru Ito, “A personal tourism navigation system to support traveling multiple destinations with time restrictions,” Proceedings of the 18th International Conference on Advanced Information Networking and Application (AINA’04), pp. 18–21, 2004.
- [9] 山本芳嗣, 久保幹雄, 巡回セールスマン問題への招待, オペレーションズ・リサーチ, pp. 25–31, 1994.
- [10] 吉江一成, 石原伸一, 松永俊雄, 吉瀬譲二, 川辺秀樹, “オンライン型道路経路・巡回路探索法の検討,” 情報処理学会全国大会, vol. 6S-04, pp. 355–356, 1997.
- [11] 坂内伸也, 松坂卓司, 松永俊雄, 吉瀬譲二, 川辺秀樹, “道路案内用巡回路探索システムの検討,” 情報処理学会全国大会, vol. 3U-02, pp. 359–360, 1998.
- [12] 丸山敦史, 柴田直樹, 村田佳洋, 安本慶一, 伊藤実, “P-tour: 観光スケジュール作成支援とスケジュールに沿った経路案内を行うパーソナルナビゲーションシステム,” 情報処理学会論文誌, vol. 45, no. 12, pp. 2678–2687, 2004.
- [13] 北野宏明, “遺伝的アルゴリズム,” 人工知能学会誌, vol. 7, no. 1, pp. 26–37, 1992.