

経路選択履歴を用いたモンテカルロ木探索による 推薦経路探索手法

百瀬 凌也^{1,a)} 石川 和明² 高山 敏典² 柳澤 政生¹ 戸川 望^{1,b)}

概要：現在、インターネット上で提供されている経路推薦・案内サービスは、経路長や所要時間で最適化した経路を推薦する不特定多数向けサービスが主であるが、道路は交通量・幅員・勾配・舗装状態といった特徴があり、例えば高齢者であれば勾配が急な坂や幅員が広い横断歩道の横断を避けた経路、未成年であれば夜は明かりが多く人通りの多い安全な経路、雨天時はできるだけ地下を歩く経路を選択したいといったニーズがある。そこで、現在の状況やユーザの嗜好に応じて経路を選択することでより個々のユーザにパーソナライズした最適経路を推薦することが考えられる。個々のユーザにパーソナライズした経路を推薦するためには、ユーザにとってどれだけ好ましいかをなんらかの方法で評価する必要があるが、好みは曖昧な指標であり個人の主観が大きく影響するため評価が難しい。そこで本稿では、経路選択履歴を用いたモンテカルロ木探索による MCTR-RR 推薦経路探索手法を提案する。提案手法は、過去にユーザが判定した経路（経路選択履歴）の特徴量を特徴ベクトルとして機械学習を用いてユーザの潜在的な経路選択傾向を学習することで、特定のユーザにとって好ましい経路かどうかを正しく判定し、モンテカルロ木探索を組み合わせることで途中経過状態の評価関数を必要としない最適経路推薦を実現する。その結果、新宿・高田馬場の2箇所で行った最適経路推薦実験では、2箇所中2箇所ユーザの好みに近い経路を推薦することに成功した。

1. はじめに

現在、インターネット上で提供されている経路推薦・案内サービスは、経路長や所要時間で最適化した経路を推薦する不特定多数向けサービスが主である。しかし、道路は交通量・幅員・勾配・舗装状態といった特徴があり、例えば高齢者であれば勾配が急な坂や幅員が広い横断歩道を避けた経路、未成年であれば夜は明かりが多く人通りの多い安全な経路、雨天時はできるだけ地下を歩く経路を選択したいといったニーズがある。そこで、現在の状況やユーザの嗜好に応じて経路を選択することでより個々のユーザにパーソナライズした最適経路を推薦することが考えられる。個々のユーザにパーソナライズした経路を推薦するためには、ユーザにとってどれだけ好ましいかをなんらかの方法で評価する必要があるが、好みは曖昧な指標であり個人の主観が大きく影響するため評価が難しい。また、ある地点で最適な評価関数を設計できたとしても、すべての場

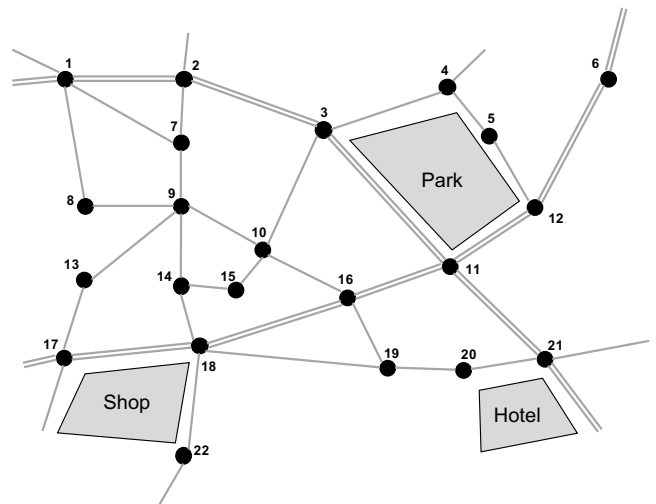


図 1: 地点 A での地図。

所で最適な評価関数であるとは限らない。

図 1 に示す地点 A の地図でユーザの経路選択の例を示す。一本線は通常の道路を示し、二重線は広い道路を示す。例えば様々な嗜好を持つユーザが地点 A で 1 番のノードから 19 番のノードに向かう経路を選択することを考える。このとき、早く着く・距離が短い経路を好むユーザは図 2(a) のように、 $1 \rightarrow 7 \rightarrow 9 \rightarrow 10 \rightarrow 16 \rightarrow 19$ と辿る経

¹ 早稲田大学
Waseda University

² ゼンリンデータコム
ZENRIN Datacom

a) ryoya.momose@togawa.cs.waseda.ac.jp

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

路を選択すると考えられる。安全な道幅が広い経路を好むユーザは図 2(b) のように、 $1 \rightarrow 2 \rightarrow 3 \rightarrow 11 \rightarrow 16 \rightarrow 19$ と辿る広い道路を通る経路を選択すると考えられる。覚えやすい経路を好むユーザは図 2(c) のように、 $1 \rightarrow 2 \rightarrow 3 \rightarrow 11 \rightarrow 21 \rightarrow 20 \rightarrow 19$ と辿るやや大回りになるが Park, Hotel が目印となる経路を選択すると考えられる。このように、同じ目的地へ向かう経路についてもユーザの嗜好によって様々な経路を選択することが予想される。

このようなユーザの好みの経路を推薦する手法として、アンケート調査により決定した探索パラメータを探索に用いる手法 [6, 8, 16] が挙げられる。

アンケート調査により探索パラメータを決定する手法は、アンケートの結果を t 検定によりパラメータを決定する手法 [16]、ファジィ測度・積分によりパラメータを決定する手法 [6, 8] などがある。文献 [16] では、ユーザの属性 (性別・年齢) と経路の嗜好に関するアンケートの相関から、信号やガードレールなどの道路の特徴に対して重みを定義することで、ユーザ固有の好みをダイクストラ法の探索コストに反映し推薦する経路を探索する。文献 [6] では、事前アンケートにより算出したファジィ測度・積分モデルにより、ユーザの好み が反映された経路を推薦する手法を提案している。経路を評価する属性間の相乗効果や相殺効果といった相互作用を考慮することでユーザごとに異なる好みの属性の重みを算出している。好みの相互関係を非加法的に算出しているが、評価関数では加重和を用いて計算している。事前アンケートでは、探索対象である地図データ中の各道路に対し歩行したときに受ける印象や状況に応じた満足度を細かく評価する必要がある。文献 [8] では、カーナビゲーションから提示されるドライバ向けの経路を対象として、人間の主観的判断とシステムアプローチの両面から問題を解決する意思決定手法である AHP (Analytic Hierarchy Process) とファジィ測度・積分を併用してアンケート結果から算出したユーザ固有の好みをダイクストラ法の探索コストに反映し推薦する経路を作成する手法を提案している。ダイクストラ法 [13] や A*法 [14] を用いて逐次的に経路を探索する手法 [6, 8, 16] においては、道路にコストを与える必要があるためユーザの好みを客観的に数値化し評価することが大きな課題となる。アンケートによる調査方法では、質問項目に現れないような潜在的なユーザの傾向を掴むことはできない上、アンケートなどにより道路のコストを決定しても全ての道路の形状に対応できるとは限らない。

そこで本稿では、経路選択履歴を用いたモンテカルロ木探索をベースにユーザに最適経路を推薦する MCTS-RR 手法を提案する。モンテカルロ木探索を用いるにあたり、探索した経路がユーザにとって好ましい経路かどうかをいかに評価するかが鍵となる。提案手法は、過去にユーザが判定した経路 (経路選択履歴) の特徴量を特徴ベクトルとし

て機械学習を用いてユーザの潜在的な経路選択傾向を学習することで、特定のユーザにとって好ましい経路かどうかを正しく判定し、モンテカルロ木探索を組み合わせることでユーザにとってより好ましい経路を推薦する。

本稿の貢献は以下の通りである。

- (1) 過去にユーザが判定した経路 (経路選択履歴) の特徴量を特徴ベクトルとして機械学習を用いてユーザの潜在的な経路選択傾向を学習することで、特定のユーザにとって好ましい経路かどうかを正しく判定する。
- (2) さらに、モンテカルロ木探索を組み合わせることで途中経過状態の評価関数を必要としない最適経路推薦を実現する。
- (3) その結果、新宿・高田馬場の 2 箇所で行った最適経路推薦実験では、2 箇所中 2 箇所ユーザの好みに近い経路を推薦することに成功した。

2. 個々のユーザに対する最適経路推薦問題

本章では、経路推薦に用いる地図データを説明する。その後、本稿で扱う最適経路推薦問題を定義する。

2.1 地図データ

地図データは道路リンクの集合 $R = \{r_1, r_2, \dots, r_{N_r}\}$ と交差点ノードの集合 $C = \{c_1, c_2, \dots, c_{N_c}\}$ 、ランドマークオブジェクトの集合 $L = \{l_1, l_2, \dots, l_{N_l}\}$ によって構成される。各道路リンク $r_s \in R$ は、道路の種類 *type*、道路の構造 *structure*、勾配種別 *slope*、横断有無 *cross*、屋根の有無 *roof*、幹線道路かどうか *wide*、道路の始点 $c_{begin} \in C$ 、終点 $c_{end} \in C$ 、を持つ。表 1 に道路リンクの要素と変数の対応をまとめる。各交差点ノード $c_t \in C$ は、緯度 *lat*、経度 *lng*、高度 *z*、可視ランドマークの集合 $L_{visible} \subseteq L$ を持つ。可視ランドマークとは、その交差点ノードから見ることが可能なランドマークのことである [9]。表 2 に交差点ノードの要素と変数の対応をまとめる。また、交差点ノードは図 1 の交差点ノード 8 のように曲がり角の頂点にも用いられる。このような分かれ道のない交差点ノードを一本道ノードと定義する。各ランドマークオブジェクト $l_u \in L$ は、カテゴリ *ct*、高さ *h*、形状座標の集合 *pp* を持つ。表 3 にランドマークオブジェクトの要素と変数の対応をまとめる。

図 3 に地図データの例を示す。図 3(a) をみると、地点 B では 3 本の道路が図のように交差しており、3 つの交差点が存在している。道路の側にはガソリンスタンド・スーパーマーケット・ホテルが建っている。このような地点 B を地図データで表現すると、図 3(b) のように道路リンク $R = \{r_1, r_2, \dots, r_7\}$ 、交差点ノード $C = \{c_1, c_2, c_3\}$ 、ランドマークオブジェクト $L = \{l_1, l_2, l_3\}$ から構成される。形状座標 *pp* はそれぞれ $\{pp_1, pp_2, pp_3, pp_4\}$ は l_2 に、 $\{pp_5, pp_6, pp_7, pp_8\}$ は l_1 に、 $\{pp_9, pp_{10}, pp_{11}, pp_{12}\}$ は l_3 に

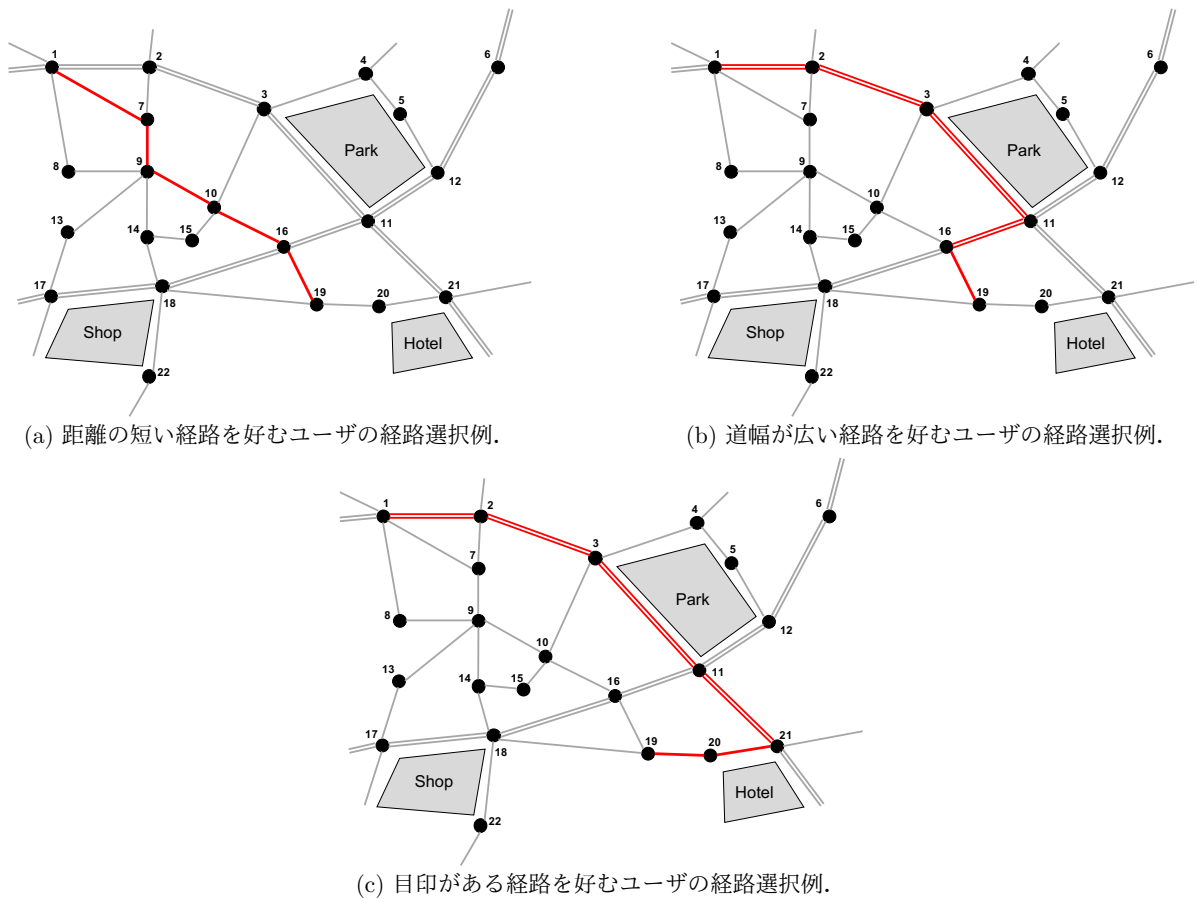


図 2: ユーザによる経路選択の例.

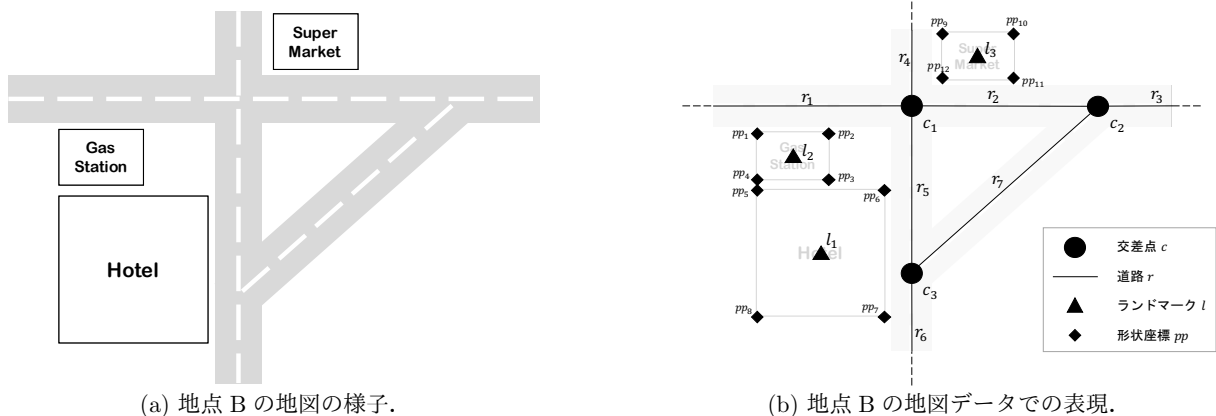


図 3: 地図データの例.

紐付き, ランドマークオブジェクトの形状を構成する.

2.2 最適経路推薦問題

道路やランドマークの情報を含む地図データは事前に与えられるとする. このとき, 本稿で定義するユーザ最適経路推薦問題とは, 出発地点の交差点 c_{start} と目的地点の交差点 c_{goal} を入力とし, 出発地点から目的地点を結ぶユーザ A にとって好ましい経路 RR (Recommended Route) を出力とするものである.

3. MCTS-RR 手法 (Monte Carlo Tree Search for Recommended Route)

モンテカルロ木探索は, モンテカルロシミュレーションと木探索を組み合わせた探索アルゴリズムで, ゲームなどの逐次決定木やプランニング問題において注目されている手法である [7, 10, 12]. モンテカルロシミュレーション (以下, モンテカルロシミュレーションをプレイアウトと呼ぶ) を多数実行し, 準最適解を探索する手法である. プレイアウトで探索を進めるため途中まで探索された経路に対する

表 1: 道路リンク $r \in R$ の要素と変数の対応.

変数名	説明	格納されているデータ例
$type$	道路種別	通常/横断歩道/横断道路/歩道橋/踏切内通路/連絡通路/建物内通路/敷地内通路/橋
$structure$	道路構造	通常/階段/エスカレータ/エレベータ/スロープ/動く歩道/段差
$slope$	勾配種別	上り/平坦/下り
$cross$	横断有無	1/0
$roof$	屋根有無	1/0
$wide$	幹線道路かどうか	1/0
c_{begin}	道路の始点	交差点ノード $c_{begin} \in C$
c_{end}	道路の終点	交差点ノード $c_{end} \in C$

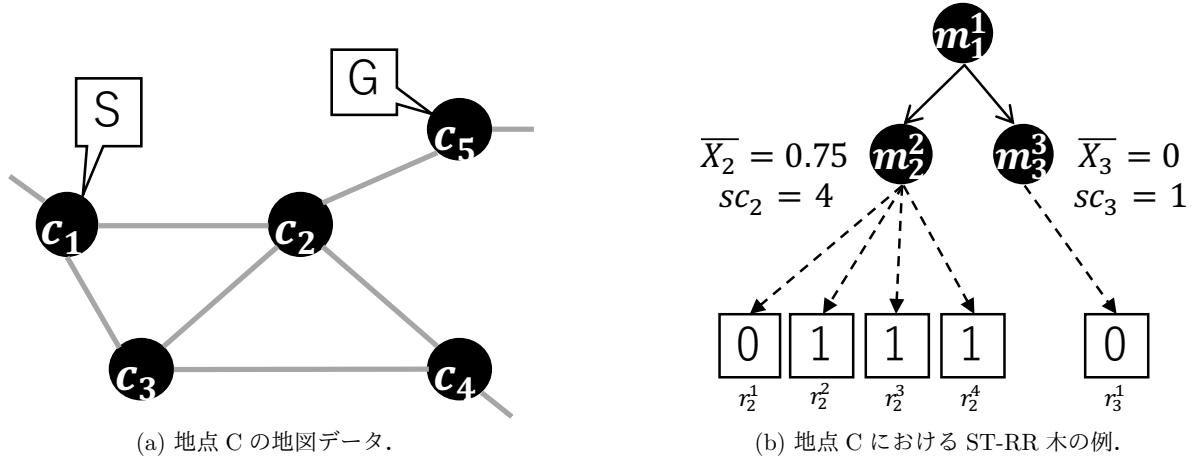


図 4: UCB1-RR 値を用いた探索の様子.

表 2: 交差点ノード $c \in C$ の要素と変数の対応.

変数名	説明	格納されているデータ例
lat	緯度	35.491441
lng	経度	139.64337
z	階層	.../-1/0/1/...
$L_{visible}$	可視ランドマーク	ランドマークデータ l の集合

表 3: ランドマークオブジェクト $l \in L$ の要素と変数の対応.

変数名	説明	格納されているデータ例
ct	カテゴリ	公官庁施設/公官庁建物/教育施設/学校/病院 レジャー/交通/宿泊建物/商業建物/目標建物 その他目標物/一般建物/駅舎/駅建物
h	高さ [m]	3
pp	形状データ	建物の頂点の緯度経度の集合

評価が不要である. また, ユーザに最適な経路を探索するとき, ユーザの好みは非明示的情報であり客観的指標として扱えず, ユーザの経路に関する好みを考慮した経路探索をモデル化することは困難である. モンテカルロ木探索ではプレイアウトを多数実行することにより, 経路を生成しユーザの好みである経路の探索が可能となる. そこで, 本稿ではモンテカルロ木探索をベースにユーザに最適経路を推薦する MCTS-RR 手法を提案する.

本章では, まず MCTS-RR 手法を提案するにあたり

MCTS-RR 手法の鍵となるデータ構造として ST-RR 木と, モンテカルロ探索における評価値 UCB1-RR 値を提案する.

3.1 ST-RR 木

ST-RR 木は, ノード集合 M とエッジ集合 E を用いて $ST-RR = (M, E)$ と表される. 本稿では, 交差点ノードと区別するためノード $m \in M$ をモンテカルロノード (MC ノード) と呼ぶ. 各 MC ノードは, 地図上の 1 つの交差点ノードと対応付けられる. ST-RR 木において j 番目の MC ノードが交差点ノード c_i と対応付けられているとき, この MC ノードを $m_j^i \in M$ とかく. このとき, ST-RR 木は, 出発地点を表す交差点 c_{start} と対応付けられる m_1^{start} を根とする木である.

今, 地図上で交差点ノード c_{i_1} と c_{i_2} が隣接しているとす. c_{i_2} と c_{i_2} に対応する MC ノードを $m_j^{i_1}$ と $m_k^{i_2}$ とする. このとき, ST-RR 木において $m_j^{i_1}$ は $m_k^{i_2}$ を子として持つことができる (あるいは, $m_k^{i_2}$ は, $m_j^{i_1}$ を子として持つことができる). 推薦経路 RR は, 複数個の MC ノードをシーケンスに並べることで表現できる.

3.2 UCB1-RR 値

モンテカルロシミュレーションを ST-RR 木に単純に適用し, 出発地点から目的地点までの経路を探索すると, プレイアウトによって出発地点の交差点ノードを起点に隣接

する交差点ノードの中からランダムに次の交差点ノードへ移動することを繰り返すことで目的地の交差点ノードへの経路を探索できる。しかし、解空間が大きい経路探索では、明らかにユーザの好みが高くなる要素をもつ道路リンクが判明した上で、その道路リンクを含む経路を探索することは好ましくない。そこで、MCTS-RR手法はUCB1-RR値なる評価値を導入しST-RR木を探索することでより効率的なプレイアウトを行う。

UCB1-RR値は、UCB1値 [15] をベースとした探索することの良しさを表す評価値である。ST-RR木のもつMCノードはそれぞれUCB1-RR値を持ち、UCB1-RR値が大きいノードは探索する上で価値が大きいことを示す。4章で提案するMCTS-RR手法は、SS-RR木の根から葉に向かって、UCB1-RR値が大きい子を順に探索する。

各MCノード m_j^i は、探索回数 sc_j と k 回目のプレイアウトにより生成された経路の良しさを rw_j^k を記録する。このとき、MCノード m_j^i のUCB1-RR値 $UCB1_j^{RR}$ は、 sc_j, rw_j^k をもとに式 (1) によって算出する。

$$UCB1_j^{RR} = \bar{X}_j + W \sqrt{\frac{2 \log N_{po}}{sc_j}}. \quad (1)$$

ここで \bar{X}_j はMCノード m_j^i を sc_j 回探索した時点での経路の良しさを期待値、 N_{po} は行ったプレイアウトの総数、 W は探索に対する重みを表現する定数を示す。

sc_j 回プレイアウトを行ったMCノード m_j^i の期待値 \bar{X}_j は、 sc_j, rw_j^k を用いて、式 (2) で算出する。

$$\bar{X}_j = \frac{\sum_{k=1}^{sc_j} rw_j^k}{sc_j}. \quad (2)$$

式 (1) の第1項 \bar{X}_j で期待値の高いMCノードに対する重み、第2項 $W \sqrt{\frac{2 \log N_{po}}{sc_j}}$ で期待値が収束していない未発見のMCノードに対する重みが大きくなることで、良さそうな経路を探索することが可能となる。

例 1 図 4 を用いて UCB1-RR 値を利用した ST-RR 木を探索する動作を説明する。図 4(a) は交差点ノード 5 つ c_1, c_2, \dots, c_5 とそれをつなぐ 9 つの道路リンクからなる地図データである。出発地点を c_1 、目的地点を c_5 とし、経路を探索することを考える。

図 4(b) に、 c_1 から c_5 への経路を探索する途中過程の ST-RR 木を表す。この ST-RR 木は 3 つの MC ノード m_1^1, m_2^2, m_3^3 からなる。出発地点の交差点ノード c_1 と対応付けられている MC ノード m_1^1 が根ノードの ST-RR 木であり、 c_1 と隣接している c_2, c_3 と対応付けられている MC ノード m_2^2, m_3^3 が m_1^1 の子ノードとして生成されている。ここで、出発地点 m_1^1 からプレイアウトを行い 5 つの経路を生成したとする。その結果、生成され

た 5 つの経路のうち、出発地点 m_1^1 の次に m_2^2 を通った経路が 4 つ、 m_3^3 を通った経路が 1 つ生成され、プレイアウトにより生成された経路の良しさはそれぞれ $rw_2^1 = 0, rw_2^2 = 1, rw_3^1 = 1, rw_3^2 = 1, rw_3^3 = 0$ となったと仮定する。このとき、各 MC ノードの期待値 \bar{X}_j 及び探索回数 sc_j はそれぞれ $\bar{X}_2 = 0.75, \bar{X}_3 = 0, sc_2 = 4, sc_3 = 1$ となる。探索に対する重み $W = 1.5$ とすると、

$$\begin{aligned} UCB1_2^{RR} &= 0.75 + 1.5 \times \sqrt{\frac{2 \log 5}{4}} \\ &= 0.75 + 0.88 \\ &= 1.63 \end{aligned} \quad (3)$$

となり、 $UCB1_2^{RR}$ の値は 1.63 となる。同様に

$$\begin{aligned} UCB1_3^{RR} &= 0 + 1.5 \times \sqrt{\frac{2 \log 5}{1}} \\ &= 0 + 1.77 \\ &= 1.77 \end{aligned} \quad (4)$$

となり、 $UCB1_3^{RR}$ の値は 1.77 となる。

$UCB1_2^{RR} < UCB1_3^{RR}$ であるため、6 回目のプレイアウトは MC ノード m_3^3 を通るよう実行される。これは UCB1-RR 値が、「MC ノード m_2^2 のほうが期待値は高いが、MC ノード m_3^3 は探索回数が少なく、MC ノード m_3^3 に進むとよりよい結果が得られる可能性がある」という意味を示しているからである。

もし探索の重み $W = 1$ とすると、 $UCB1_2^{RR} = 1.34, UCB1_3^{RR} = 1.18$ となる。このように、UCB1-RR 値により未発見の経路の探索と探索済みの経路の利用を両立でき、プレイアウト数を少なく効率の良い探索が可能となる。■

4. 経路選択履歴を用いた MCTS-RR 経路探索手法

本章では、ユーザの経路選択履歴を経路評価に利用することでユーザに最適な経路を推薦する MCTS-RR 経路探索手法を提案する。

ユーザにとって好ましい経路とはどのようなことか考える。ユーザは目的地に向かうための経路を例えば「道路を横断したくないため、できるだけ横断歩道を避けたい」、「迷う可能性があるため分かれ道が少ない道を通りたい」、「目印となるランドマークができるだけ多い経路を通りたい」、「遠回りになるが覚えやすく迷いにくい大通りを通りたい」といった様々な要因によって経路を選択する。そのため、ユーザにとって良い経路に対し、それぞれの要素に重みを定義し足し合わせることで評価関数を設計することが難しい。道路の特徴によってコストを設定し、最小コストの経路を逐次探索的に求める探索では、局所最適解となる可能性がある。また、アンケートによる調査方法では、

質問項目に現れないような潜在的なユーザの傾向を掴むことはできない上、ユーザ自身が経路に関する良いらしさの認識があるとは限らず実際の経路に反映されない可能性もある。

そこで、ユーザにとって好ましい経路の評価指標に過去にユーザが判定した経路(経路選択履歴)を用いることで、ユーザの潜在的な経路選択傾向を評価に反映させ、推薦経路のような途中経過状態の評価関数の設計が難しい問題に用いると良い結果を得られることで知られているモンテカルロ木探索を組み合わせることでユーザにとって好ましい経路を探索する。

4.1 提案手法の概要

提案手法の概要を図5に示す。ユーザは提案手法に出発地点の交差点ノード c_{start} と目的地の交差点ノード c_{goal} を入力する。提案手法は予め、学習フローによって経路選択履歴を用いてユーザの好みらしさを評価する評価器を生成する。まず、経路選択履歴の経路データから20個の特徴量を抽出する(Step L1)、次に抽出した特徴量をSVM[11]を用いて学習させ、評価器を生成する(Step L2)。探索フローでは、ユーザの入力を受け付けるとMCTS-RR手法によってユーザの好みらしい経路を探索する。まず、UCB1-RR値を用いた探索により最良子MCノードを探索する(Step S1)。最良子MCノードからプレイアウトを行いプレイアウト経路を得る(Step S2)。プレイアウト経路の経路データから20個の特徴量を抽出する(Step S3)。次に、プレイアウト経路の特徴量を学習フローで生成した評価器に適用してプレイアウト経路の好みらしさを推論する(Step S4)。評価器により推論された結果をST-RR木に逆伝搬する(Step S5)。探索フローが収束するとMCTS-RR手法はユーザに最適経路を出力する。

以下、本章では本稿で学習に用いる経路の特徴量、特徴量を用いて学習器を生成する学習フロー、生成器とMCTS-RR手法を用いて経路探索を行う推論フローについて説明する。

4.2 経路の特徴量

本節では、経路を評価する際に用いる特徴量を提案する。

文献[1]では、経路を選択する際に安全性に重点を置いて経路を選択する歩行者は(A)幅員が広い、(B)自動車の交通量が多い、(C)舗装されている、(D)商店がある、(E)歩道が整備されている、(F)自動車に対する不安感が少ないといった経路、男性や高齢者は(G)最短となる経路を選択する特性をもつ結果を示している。文献[2]では、方向感覚がよく迷いにくい人は(H)経路中に含まれるランドマークを頼りに経路を覚える傾向にある結果を示している。文献[16]では、乳幼児を連れている歩行者や高齢者は(I)横断歩道、(J)信号のある安全な経路、(K)階段・坂を避けエ

レベータ・エスカレータを使う経路を選択する結果を示している。文献[5]では、(L)ユーザは直進方向と目的地方向の狭角が大きくなならないうちは直進を優先するという結果を示している。

以上からユーザが経路の好みだと判定するとき、その経路には以下のような何らかの特徴があると考えられる。

- 距離が短く早く到着する経路 (G)
- 歩道が整備されている大通りを通る覚えやすい経路 (A)(B)(C)(E)
- 右左折が少ない迷いにくい経路 (L)
- 信号や横断歩道など整備された交差点で道路を横断する経路 (I)(J)
- 幹線道路ではなく車通りが少ない経路 (F)
- 知っている目印(ランドマーク)が途中に建っている経路 (D)(H)
- 階段や段差のない荷物を持っていても歩きやすい経路 (K)

提案手法では、経路に関してこれらの要素を表現できるような以下の20個の特徴量を抽出し、その特徴を用いてユーザの好みをモデリングすることを考える。出発地点の交差点ノード c_{start} から目的地の交差点ノード c_{end} に至る交差点ノード N_p 個からなるある経路を p とする。 p の経路長を RL_p とする。

- 1. 経路長 (rRL)

経路の最大の特徴として経路長があげられる。大通りを通る経路を好むユーザでも大回りすることをどれだけ許容するかはユーザにより異なる。 c_{start} から c_{end} に至る最短経路長を RL_{min} とする。経路長比の特徴量を $rRL = \frac{RL_p}{RL_{min}}$ とする。

- 2. 分かれ道数 (aPC)

分かれ道が多い経路は途中で確認しつつ移動しなくては迷う可能性がある。目的地まで迷わずに正しい経路を進むために分かれ道が重要な指標となり得る。経路 p に含まれる各交差点ノード c_j が隣接する交差点ノードの総数を分かれ道と定義する。経路中に含まれる分かれ道の総数を PC_p とすると、分かれ道数の特徴量は $aPC = \frac{PC_p}{N_p}$ となる。

- 3. 右左折数 (rTC)

右左折が多い経路は、途中のどこで曲がるかを記憶するのが難しく、逐一提案された経路を確認しながら目的地に向かうこととなる。右左折が少ない経路であれば曲がる交差点を記憶することで目的地にたどり着けるため、経路選択の指標となり得る。文献[3,4]により、人間は8方向の参照軸を基に方向を認識していることから22.5度以上の方向転換を右左折と定義する。経路に含まれる右左折数を TC_p とすると、右左折数の特徴量は $rTC = \frac{TC_p}{N_p}$ となる。

- 経路タイプ

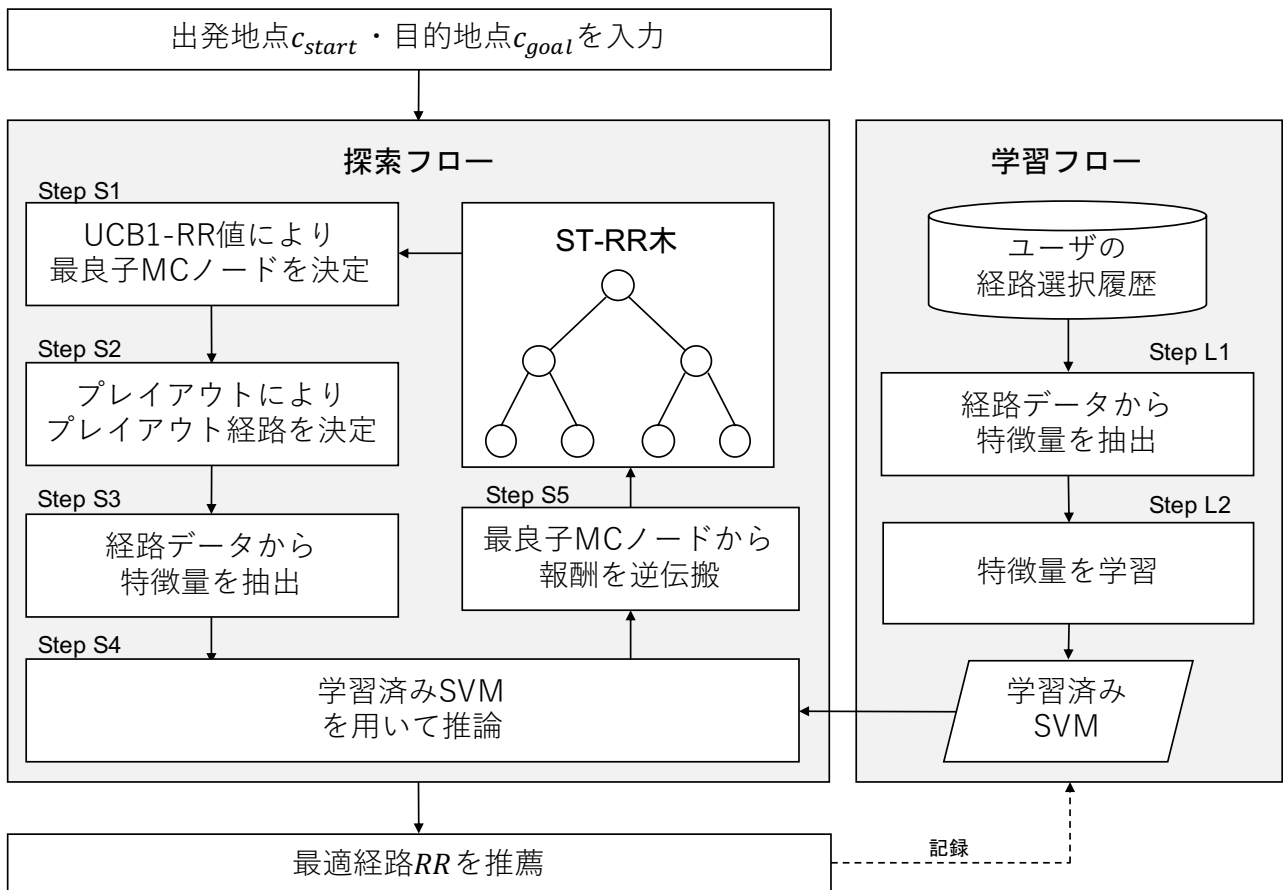


図 5: MCTS-RR 手法の概要.

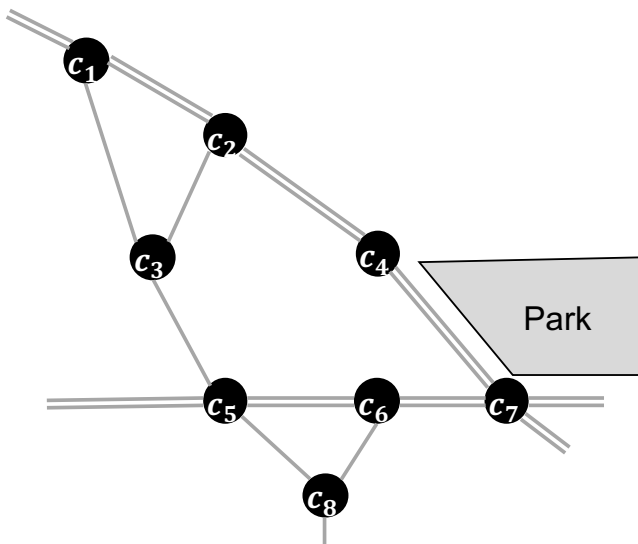


図 6: 地点 D の地図データ.

横断歩道をできるだけ避けるユーザ、大通りを歩きたいユーザ、歩道がある安全な道を歩きたいユーザなどが考えられる。例えば、経路 p に含まれる横断歩道の経路長を TCL_p 、横断歩道ノードの数を TCC_p とすると、横断歩道の経路長の特徴量は $rTCL = \frac{TCL_p}{RL_p}$ 、横断歩道のノード数を $rTCC = \frac{TCC_p}{N_p}$ となる。

- 4. 横断歩道の経路長 (rTCL)

- 5. 横断歩道のノード数 (rTCC)
- 6. 幹線道路の経路長 (rTWL)
- 7. 幹線道路のノード数 (rTWC)
- 8. 歩道付き道路の経路長 (rTSL)
- 9. 歩道付き道路のノード数 (rTSC)

● 経路種別

大きな荷物を持っているユーザは階段や上り坂を避けることが考えられる。例えば、経路 p に含まれる階段の経路長を $CStL_p$ 、階段ノードの数を $CStC_p$ とする。階段の経路長の特徴量を $rCStL = \frac{CStL_p}{RL_p}$ 、階段のノード数を $rCStC = \frac{CStC_p}{N_p}$ となる。

- 10. 階段の経路長 (rCStL)
- 11. 階段のノード数 (rCStC)
- 12. 坂の経路長 (rCSIL)
- 13. 坂のノード数 (rCSIC)
- 14. 段差の経路長 (rCCLL)
- 15. 段差のノード数 (rCLC)
- 16. 平坦の経路長 (rCNL)
- 17. 平坦のノード数 (rCNC)

● ランドマーク

経路の途中にコンビニエンスストアや公園、ホテルなど目印となるランドマークがあると間違えずに経路を歩けると考えられる。文献 [9] の手法を用いて各ノ

ドからの可視ランドマークを求めた。推薦経路の1ノードあたりの有名可視ランドマーク数、無名可視ランドマーク数をランドマークに関する特徴量とする。有名ランドマークとは公官庁施設/公官庁建物/教育施設/学校/病院/レジャー/交通/宿泊建物/商業建物/目標建物/一般建物/駅舎/駅建物のカテゴリをもつランドマークを指し、それ以外のランドマークを無名ランドマークとする。例えば、経路 p 中に存在する可視ランドマーク数を RmC_p とすると、可視ランドマーク数の特徴量は $aRmC = \frac{RmC_p}{N_p}$ となる。

- 18. 可視ランドマーク数 (aRmC)
- 19. 有名可視ランドマーク数 (aRmFC)
- 20. 無名可視ランドマーク数 (aRmNC)

4.3 学習フロー

ユーザの経路に関する好みを過去にそのユーザが判定した経路 (経路選択履歴) から 20 個の特徴量を抽出する (Step L1)。抽出した特徴量を特徴ベクトルとして機械学習で学習する (Step L2)。

経路選択履歴は、2 地点間の経路情報と好みかどうかのラベルにより構成される。機械学習の評価器にはサポートベクタマシン (SVM) [11] を用いる。学習し作成した評価器 (featureSVM) に対してある経路の特徴量を入力することで、その経路がユーザにとってどれだけ好ましいかを判定し、ユーザにとって好みである経路である確率 $[0,1]$ を出力する。学習に用いたパラメータは、グリッドサーチにより最適化を行い $C = 1, degree = 2, gamma = 10, kernel = poly$ と決定した。

4.4 推論フロー

我々の提案する MCTS-RR 手法は、地図データから ST-RR 木を作成しながら最適な経路を探索する。MCTS-RR 手法は、出発地点の交差点ノード c_{start} ・到着地点の交差点ノード c_{goal} を入力し、推薦経路 RR を出力する。

例 2 図 2(a) を例にとると、入力は出発地点・到着地点として交差点ノード 1・交差点ノード 19 を入力したときに、 $RR = [1, 7, 9, 10, 16, 19]$ を出力する。 ■

以下に MCTS-RR 手法による経路探索アルゴリズムを提案する。MCTS-RR 手法は探索ステップ (Step S1)、プレイアウトステップ (Step S2)、特徴量抽出ステップ (Step S3)、推論ステップ (Step S4)、逆伝搬ステップ (Step S5) を多量に実行することで ST-RR 木を更新していく。この探索ステップ (Step S1) から逆伝搬ステップ (Step S5) を 1 回ずつ実行することを 1 サイクルと呼ぶ。出発地点に対応する MC ノードを起点に 1 サイクルごとに、次に進む MC ノードを決定し推薦経路に追加していくことで出発地

点からの推薦経路 RR を生成する。

例 3 図 6 に示す地点 D の地図データを用いて、交差点ノード 1 から交差点ノード 8 に向かう推薦経路を探索するときの提案手法の動作例を合わせて説明する。一本線は通常の道路を示し、二重線は広い道路を示す。ユーザ A は広い道路を好むユーザと仮定する。このとき、ユーザ A の選択経路履歴を学習した評価器は広い道路を通る経路を好みと判定する評価器であると仮定すると、交差点ノード $c_1, c_2, c_4, c_5, c_6, c_7$ を通るような経路は経路の良いらしさが高くなる。 ■

MCTS-RR 手法により探索を行う際、開始する前に初期化を行う。出発地点の交差点ノードから MC ノード m_1^{start} を生成し、ST-RR 木の根ノードとする。推薦経路 RR の先頭に m_1^{start} を加える ($RR = [m_1^{start}]$)。探索レベル $level = 1$ とする。 $level$ は ST-RR 木において深さ $level$ の MC ノードまでを決定していることを示す。

例 4 図 7(a) に初期化の様子を示す。出発地点の MC ノード m_1^1 を生成し、ST-RR 木の根ノードとする。 $level = 1, RR = [m_1^1]$ とする。 ■

以下では、Step S1~Step S5 の各ステップを提案する。**探索ステップ (Step S1)**

決定済みの推薦経路 RR のうち $level$ 番目の MC ノードを着目 MC ノードとする。着目 MC ノードを起点として葉に向かって UCB1-RR 値がより大きい子ノードを選択し木の根から葉に向かって進める。進める途中、MC ノードが持つことができる子ノードのうち 1 つでも生成されていない子ノードがある MC ノード $m_j^{i_1}$ に到達した場合は、交差点ノード c_{i_1} を参照し $m_j^{i_1}$ が持つことができる子ノードの中からランダムに $m_{child}^{i_2}$ を生成する。生成した MC ノード $m_{child}^{i_2}$ を $m_j^{i_1}$ の子ノードとして追加する。このとき、生成した MC ノード $m_{child}^{i_2}$ を最良子 MC ノードとする。また、進める途中、目的地点と対応する MC ノード m_j^{goal} に到達した場合は、 m_j^{goal} を最良子 MC ノードとする。

例 5 図 7(b) に探索ステップ (Step S1) の様子を示す。まず、 $level = 1$ であるため、 RR の先頭にある MC ノード m_1^1 を着目 MC ノードとする。 m_1^1 を起点に UCB1-RR 値を用いた探索を進める。図 6 を見ると、交差点ノード c_1 は交差点ノード c_2, c_3 と隣接しているため、 m_1^1 は、 c_2, c_3 と対応付けられている MC ノード m_j^2, m_j^3 を子ノードに持つことができる。しかし、 m_1^1 はこれらの子を持っていない。 c_2, c_3 のうち、ランダムに c_2 を選択し、対応する MC ノード m_2^2 を生成し m_1^1 の子ノードとして追加する。 m_2^2 を最良子

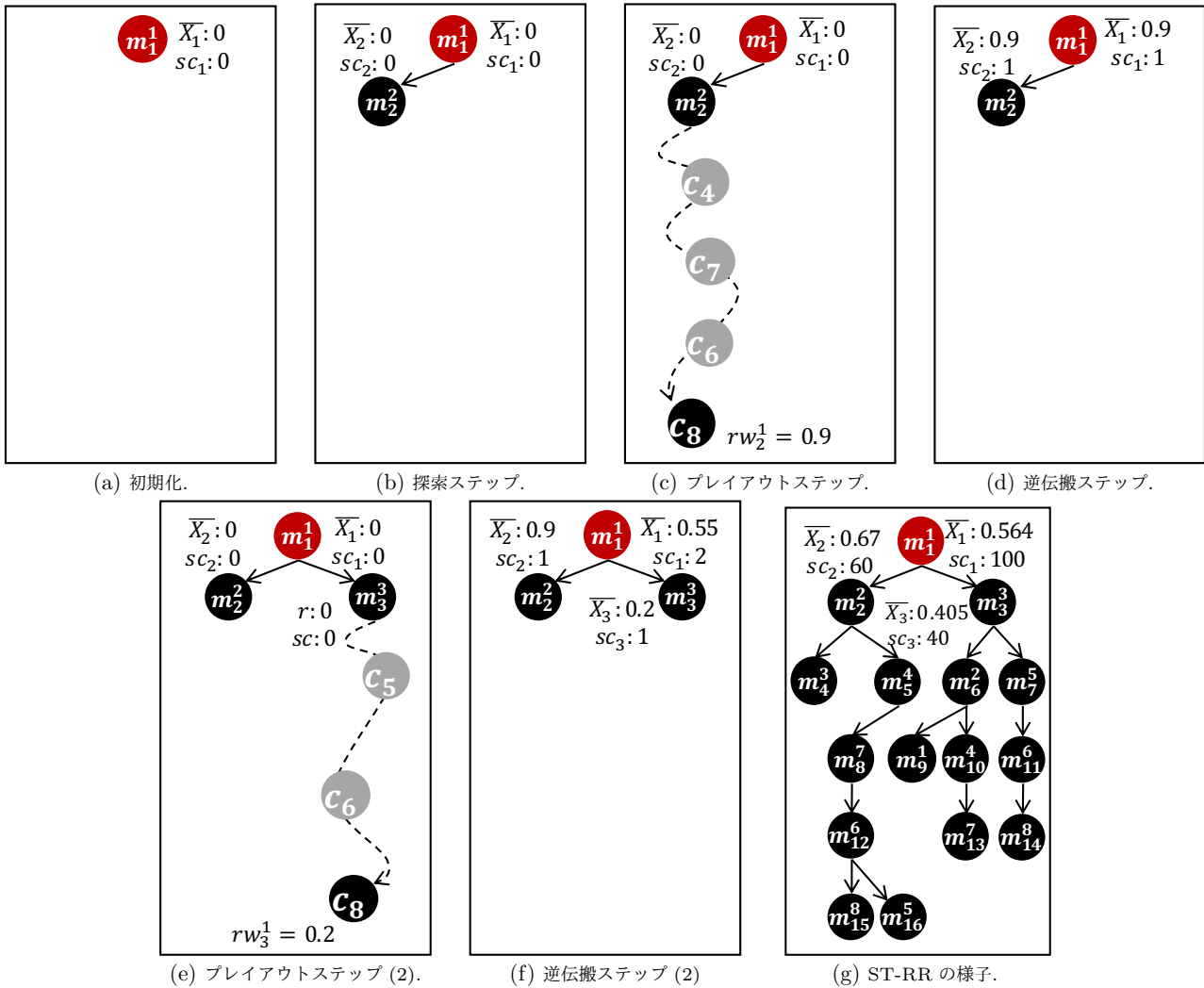


図 7: 提案手法の動作の様子.

MC ノードにする (図 7(b)).

プレイアウトステップ (Step S2)

最良子 MC ノードに対応する交差点ノードを起点にプレイアウト経路を生成する。プレイアウト経路は、交差点データを参照し隣接する交差点ノードをランダムに選択し移動することで生成される。目的地点に到達したとき、探索ステップ (Step S1) で得た着目 MC ノードから最良子 MC ノードまでの軌跡とプレイアウトによる移動の軌跡を合わせてプレイアウト経路 R_{po} とする。

なお、もし探索ステップ (Step S1) で最良子 MC ノードが m_j^{goal} である場合、プレイアウトによる経路生成はスキップされる。

例 6 図 7(c) に最良子 MC ノード m_2^2 からプレイアウトを行う様子を示す。 m_2^2 に対応する交差点ノード c_2 からランダムに経路を生成する。 c_2 に隣接する交差点の中からランダムで c_4 に移動、 c_4 に隣接する交差点の中からランダムで c_7 に移動、のようにプレ

イアウトを行う。例ではプレイアウトによる移動の軌跡は $[c_2, c_4, c_7, c_6, c_8]$ となった。探索ステップにより着目 MC ノードから最良子 MC ノードまでに得た移動の軌跡は $[c_1, c_2]$ であるため、プレイアウト経路 $R_{po} = [c_1, c_2, c_4, c_7, c_6, c_8]$ となる。

特徴量抽出ステップ (Step S3)

プレイアウト経路 R_{po} から 20 個の特徴量を抽出する。

推論ステップ (Step S4)

Step S3 で抽出したプレイアウト経路 R_{po} の特徴量を 4.3 章で作成した評価器に適用することで、プレイアウト経路 R_{po} が好ましいかを推論する。評価器が出力する入力した経路のユーザの好みである確率を経路の良いらしきとする。

例 7 例の評価器は広い道路を通る経路を好みと判定するため、 $c_1, c_2, c_4, c_7, c_6, c_8$ を通るような経路は経路の良いらしきが高くなる。図 7(c) の例では、 $R_{po} = [c_1, c_2, c_4, c_7, c_6, c_8]$ から抽出した特徴量から経路の良いらしき rw_2^1 は 0.9 となった。

逆伝搬ステップ (Step S5)

評価器により推論された経路の良しさを最良子 MC ノードから親ノード m_p を辿り着目 MC ノードまで逆伝搬させる。 rw_i を用いて \overline{X}_i を更新する。探索回数も $sc_i = sc_i + 1$ と更新する。

例 8 図 7(d) に逆伝搬の様子を示す。プレイアウト経路 $R_{po} = [1, 2, 4, 7, 6, 8]$ の良しさを $rw_2^1 = 0.9$ を $ST-RR$ 木の最良子 MC ノード m_2^2 から着目 MC ノード m_1^1 までの MC ノードに対して逆伝搬する。 $\overline{X}_2 = 0.9/1 = 0.9$, $sc_2 = 0+1 = 1$, $\overline{X}_1 = 0.9/1 = 0.9$, $sc_1 = 0+1 = 1$ とする。 ■

この探索ステップ (Step S1) から逆伝搬ステップ (Step S5) までの処理を事前に定めたサイクル数 N_{MC} だけ繰返し処理する。

例 9 次のサイクルでは、探索ステップ (Step S1) で m_3^3 を生成する。プレイアウトステップ (Step S2) で図 7(e) に同様に m_3^3 からプレイアウトの様子を示す。 $R_{po} = [1, 3, 5, 6, 8]$ となった。推論ステップ (Step S4) で $R_{po} = [1, 3, 5, 6, 8]$ の経路の良しさを $rw_3^1 = 0.2$ となったとする。逆伝搬ステップ (Step S5) では図 7(f) のように m_3^3, m_1^1 の期待値 $\overline{X}_3, \overline{X}_1$ と探索回数 sc_3, sc_1 をそれぞれ更新する。 ■

探索ステップ (Step S1) から逆伝搬ステップ (Step S5) のサイクルを N_{MC} 回処理したあと、着目 MC ノードの子の中で期待値が一番大きい MC ノード $m_{max}^{i_{max}}$ を推薦経路 RR に追加する。これは推薦経路のうち、 m_1^{start} から $m_{max}^{i_{max}}$ までの経路が決定したことを意味する。加えた MC ノード $m_{max}^{i_{max}}$ に対応する交差点ノード $c_{i_{max}}$ が目的地点の交差点ノード c_{goal} と同一の場合は、探索を終了し推薦経路 RR を出力する。目的地点の交差点ノードでない場合は、 $level = level + 1$ として再度探索ステップ (Step S1) から逆伝搬ステップ (Step S5) までの処理を N_{MC} 回行う。

例 10 図 7(g) に $M = 100$ として探索ステップ (Step S1) から逆伝搬ステップ (Step S5) までの処理を終えた $ST-RR$ 木の様子を示す。各 MC ノード m_j^i は良しさを期待値 \overline{X}_j 、探索回数 sc_j を保持しているが、図中では省略している。着目 MC ノード m_1^1 の子ノードの期待値はそれぞれ $\overline{X}_2 = 0.67$ 、MC ノード 3 が $\overline{X}_3 = 0.405$ となった。 $\overline{X}_2 > \overline{X}_3$ より m_2^2 に対応付けられている c_2 を推薦経路 RR に加える。 $RR = [c_1, c_2]$ となる。 c_2 は目的地点 c_8 ではないため、 $level = 1 + 1 = 2$ として探索ステップ (Step S1) に戻る。 ■

5. 評価実験

提案手法によるユーザ最適経路推薦アプリケーションを Python で実装した。地図の描画には OpenStreetMap [17] を用いた。実験のため用意した地図データを図 8 に示す。ユーザの経路選択履歴は図 8 に示した 4 都市から好み経路 24 個、好みでない経路 29 個の計 53 個を用意し事前にアプリケーションに読み込ませた。今回実験を行ったユーザは、「歩いているときに推薦された経路を確認することなく目的地にたどり着きたい」という好みがあり、覚えやすい大通りを通ること・あまり右左折数がないことなどを重視していた。

5.1 比較実験

最適経路推薦アプリケーションに出発地点と目的地点を入力し、経路を推薦する実験を行った。新宿では直線距離約 0.73km の 2 地点間、高田馬場では直線距離約 0.53m の 2 地点間の最適経路を探索する実験を行った。モンテカルロ木探索のプレイアウトの回数 N は 1000 とした。

比較実験として以下の 4 つの手法で推薦する経路を探索した。

DIJKSTRA-distance 距離をコストとし、ダイクストラ法で最短経路を探索する手法。

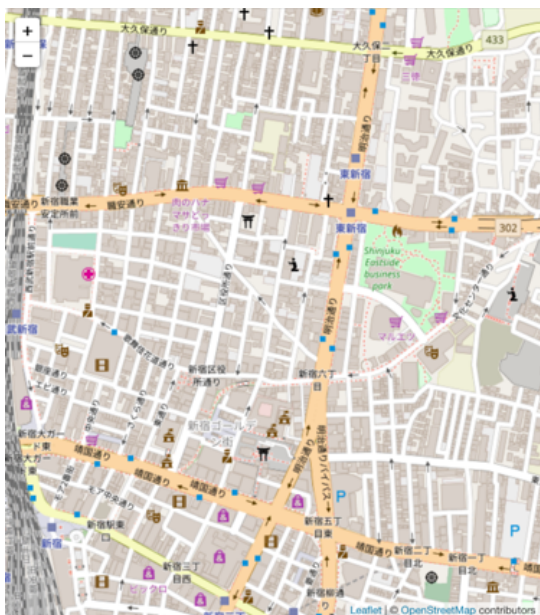
DIJKSTRA-preference [16] アンケートを基にユーザ最適な経路に関するコストを算出し、道路の特徴により重み付けした経路長をコストとしてダイクストラ法を用いた最小コスト探索により経路探索を行う手法。「安全や快適」を選択したユーザであったため、文献 [16] を参考に信号 (0.674) / ガードレール (0.695) / 横断歩道 (0.673) / 歩道 (0.700) / エスカレータ (0.693) / 坂 (1.418) / 階段 (1.446) に対してそれぞれコストを設定した。

MCTS-distance 評価を距離の短い経路を良いとする評価関数により判定し MCTS-RR により経路を探索する手法。

MCTS-featureSVM 評価を featureSVM によって経路の良しさを判定し MCTS-RR により経路を探索する提案手法。

5.2 実験結果

図 9 に新宿周辺で設定した出発地点から目的地点への推薦経路を探索した結果を示す。図 9(a) に示す DIJKSTRA-distance による経路は出発地点から目的地点までほぼ直線で到達する経路を推薦している。図 9(b) に示す DIJKSTRA-preference による経路はコストの距離に対して道路の特徴による重み付けをしているが、今回の探索では DIJKSTRA-distance と同様の経路を推薦している。図 9(c) に示す



(a) 新宿周辺 (交差点ノード数: 4881, ランドマークオブジェクト数: 66) .



(b) 横浜周辺 (交差点ノード数: 3919, ランドマークオブジェクト数: 24) .



(c) 恵比寿周辺 (交差点ノード数: 3980, ランドマークオブジェクト数: 43) .



(d) 高田馬場周辺 (交差点ノード数 10614 個, ランドマークオブジェクト数 130 個) .

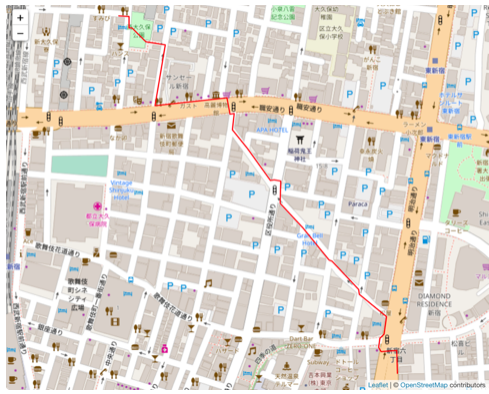
図 8: 実験に用いた地図データ.

MCTS-distance による経路も DIJKSTRA-distance と同様の傾向を示しているが、少し速回りをする経路を推薦している。図 9(d) に示す MCTS-featureSVM による経路は、他 3 つの手法と比べるとやや遠回りとなる経路であるが、大通りを通り目的地に到達する経路を推薦している。

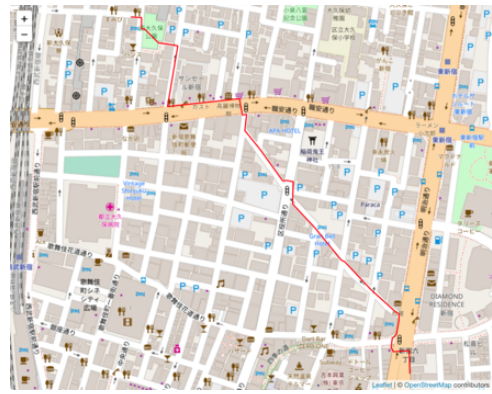
図 10 に高田馬場周辺で設定した出発地点から目的地点への推薦経路を探索した結果を示す。図 10(a) に示す DIJKSTRA-distance による経路は出発地点から目的地点までほぼ直線で到達する経路を推薦している。図 10(b) に示す DIJKSTRA-preference による経路はコストの距離に対して道路の特徴による重み付けをしていることにより、DIJKSTRA-distance の経路とは少し異なり大通りを通る経路を推薦している。図 10(c) に示す MCTS-distance による経路は DIJKSTRA-distance とほぼ同様の傾向の経路

を推薦している。図 10(d) に示す MCTS-featureSVM による経路は、他 3 つの手法と比べるとやや遠回りとなる経路であるが、大通りを通り目的地に到達する経路を推薦している。

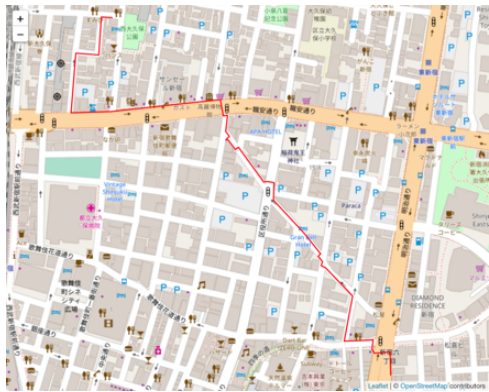
表 4 に評価実験の結果をまとめる。○はユーザが好みであったことを表す。×はユーザの好みでなかったことを表す。本実験に参加したユーザは、「歩いているときに推薦された経路を確認することなく目的地にたどり着きたい」という好みがあり、覚えやすい大通りを通ること・あまり右左折数がないことなどを重視していた。ユーザによる判定でこれを満たす経路を推薦したものは提案手法 (MCTS-featureSVM) のみであった。



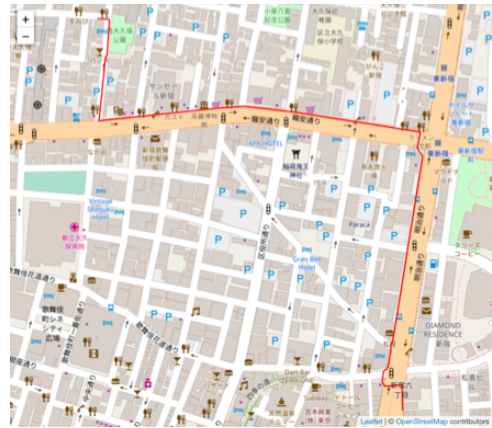
(a) DIJKSTRA-distance.



(b) DIJKSTRA-preference [16].



(c) MCTS-distance.



(d) MCTS-featureSVM.

図 9: 新宿周辺での経路推薦実験の結果.



(a) DIJKSTRA-distance.



(b) DIJKSTRA-preference [16].



(c) MCTS-distance.



(d) MCTS-featureSVM.

図 10: 高田馬場周辺での経路推薦実験の結果.

表 4: 実験結果のまとめ.

手法	新宿	高田馬場
DIJKSTRA-distance	×	×
DIJKSTRA-preference [16]	×	×
MCTS-distance	×	×
MCTS-featureSVM	○	○

6. おわりに

本稿では、モンテカルロ木探索による経路選択履歴を考慮した推薦経路探索手法を提案した。提案手法では、(a) 過去にユーザが判定した経路 (経路選択履歴) の特徴量を特徴ベクトルとして機械学習を用いてユーザの潜在的な経路選択傾向を学習することで、特定のユーザにとって好ましい経路かどうかを正しく判定し、(b) モンテカルロ木探索を組み合わせることで途中経過状態の評価関数を必要としない最適経路推薦を実現することで、単に事前アンケートを用いてコスト関数を設計し最小コストを探索する経路推薦手法と比較してユーザの潜在的な好みをモデリングし、ユーザの好みに近い経路を推薦することを可能とした。新宿・高田馬場の2箇所で行った最適経路推薦実験では、2箇所中2箇所ユーザの好みに近い経路を推薦することに成功した。

提案手法では、以下のような課題が挙げられる。

- ユーザの健康状態、時刻、天気といった動的な要素が加味されていない
- 出発地点と目的地点の距離が離れるとプレイアウトが収束しにくい
- 提案手法を複数のユーザーに対して未適用

今後は、以上に挙げた点を改善することでよりユーザに最適な経路推薦を実現することを目指す。

参考文献

- [1] 毛利正光, 塚口博司, “歩行者の経路選択特性について,” 土木学会関西支部年次学術講演会講演梗概集, vol. IV, no. 28, 1979.
- [2] 新垣紀子, “なぜ人は道に迷うのか?: 一度訪れた目的地に再度訪れる場面での認知プロセスの特徴,” 認知科学, vol. 5, no. 4, pp. 4.108–4.121, 1998.
- [3] 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,” *Center for Spatial Information Science, University of Tokyo (CSIS), Discussion Paper*, no. 41, 2001.
- [4] 山本直英, 岡部篤行, “曲線通路における方向判断についての実験による分析,” 日本建築学会計画系論文集, vol. 67, no. 559, pp. 165–170, 2002.
- [5] 塚口博司, 松田浩一郎, “歩行者の経路選択行動分析,” 土木学会論文集, no. 709, pp. 117–126, 2002.
- [6] 赤坂優太, 鬼沢武久, “ファジィ測度・積分によって個人の好みを反映した歩行者ナビゲーション,” 日本知能情報ファジィ学会 ファジィ システム シンポジウム 講演論文集 第21回ファジィ システム シンポジウム, p. 51, 2005.
- [7] 美添一樹, “モンテカルロ木探索-コンピュータ囲碁に革命を起こした新手法,” 情報処理, vol. 49, no. 6, pp. 686–693, 2008.
- [8] 藤琳, 泉隆, 魯曉鋒, 涌井文雄, “ファジィ ahp を応用した最適経路探索の一手法,” 電気学会論文誌 C (電子・情報・システム部門誌), vol. 133, no. 6, pp. 1269–1276, 2013.
- [9] 竹田健吾, 柳澤政生, 戸川望, 新田知之, 進藤大介, 田中清貴, “迷いにくい可視ランドマークに基づく屋外歩行者ナビゲーションシステム,” 組込みシステムシンポジウム 2014 論文集, vol. 2014, pp. 102–107, 2014.
- [10] B. Brüggmann, “Monte carlo go,” Technical report, Physics Department, Syracuse University Syracuse, NY, Tech. Rep., 1993.
- [11] C. Cortes and V. Vapnik, “Support-vector networks,” *Machine learning*, vol. 20, no. 3, pp. 273–297, 1995.
- [12] R. Coulom, “Efficient selectivity and backup operators in monte-carlo tree search,” in *International conference on computers and games*, pp. 72–83. Springer, 2006.
- [13] E. W. Dijkstra, “A note on two problems in connexion with graphs,” *Numerische mathematik*, vol. 1, no. 1, pp. 269–271, 1959.
- [14] P. E. Hart, N. J. Nilsson and B. Raphael, “A formal basis for the heuristic determination of minimum cost paths,” *IEEE transactions on Systems Science and Cybernetics*, vol. 4, no. 2, pp. 100–107, 1968.
- [15] L. Kocsis and C. Szepesvári, “Bandit based monte-carlo planning,” in *European conference on machine learning*, pp. 282–293. Springer, 2006.
- [16] M. Matsuda, H. Sugiyama and M. Doi, “A personalized route guidance system for pedestrians,” *The Transactions of the Institute of Electronics, Information and Communication Engineers A*, vol. 87, no. 1, pp. 132–139, 2004.
- [17] OpenStreetMap contributors. (2017) Planet dump retrieved from <https://planet.osm.org>. <https://www.openstreetmap.org>