

# 歩行者自律航法測位結果のマップマッチングアルゴリズム検討と評価

大向雅歌<sup>†</sup> 篠宮聖彦<sup>†</sup> 小川泰嗣<sup>†</sup>

**概要**：近年、ナビゲーションや従業員の業務効率分析・位置把握のため、屋内測位技術が注目されている。代表的な実現方法として、歩行者自律航法があるが、歩行するにつれ誤差が蓄積されるため、マップマッチングによる補正を行うことがある。しかし、マップマッチングによる補正には、1度補正を間違えるとその後も継続的に間違えてしまうという問題がある。そこで、リアルタイムで補正可能なアルゴリズムの検討を行った。歩行した経路の候補リストを保持し、新しく測位結果が得られる毎に、測位結果に類似性の高い経路候補を選択することで、過去の補正結果に依存しない補正が可能になった。評価した結果、途中で間違った補正行ったうちの6割で最終的に正しい経路に復帰することができた。

**キーワード**：屋内測位，マップマッチング，曲がり，歩行経路

## 1. はじめに

近年、ナビゲーションや従業員の業務効率分析・位置把握のため、屋内測位技術が注目されている。ここで、複数の店舗が入っている商業施設内で、ある利用者がスマートフォンで時々現在位置を把握しながら目当ての店舗へ向かうシーンを例に、人の歩行について考える。利用者にとって興味を引く店舗が、目当ての店舗以外にも、いくつかあると、店舗に近づくため、道の方向に対し横方向に右へ左へ歩行することがある。また、目当ての店舗がある道以外に興味を引く店舗があり、そこへ行ってから進んできた道に戻り目当ての店舗に行くということもある。

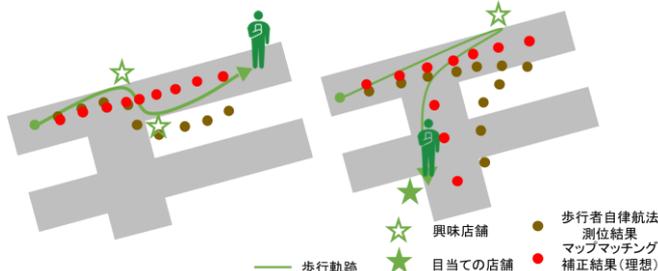


図 1 人の歩行例

人の現在位置を表示するには、道の方向に対して横方向に歩行する場合や進んできた道に戻る場合でも、リアルタイムに位置を求める必要がある。また、算出する位置は、人が地図上にプロットした位置で、現在位置が分かるよう、実際にいる道の上に算出できることが望ましい。

屋内測位の代表的な実現方法として、歩行者自律航法がある。歩行者自律航法とは、歩行者が加速度センサ・角速度センサ等が搭載されているデバイスを持って歩き、センサの出力値から歩行速度を算出し積算していくことで相

対位置を求める。歩行者自律航法は、センサの出力値の誤差により、歩行するにつれ誤差が蓄積されるため、補正を行う必要がある。マップマッチングとは、マップ情報を使って、測位結果を道の上に補正する技術である。マップ情報には、交差点をノードとした座標情報と道をリンクとした両端ノード情報で表現するベクターマップがある。

## 2. 提案手法の要件

本研究では、用途をナビゲーションや従業員の位置把握のための現在位置表示とし、歩行者自律航法の結果を補正するマップマッチングのアルゴリズムを提案する。人の歩行には、道なりに進む意外に、

- 道の方向に対して横方向に歩行する
- 進んできた道に戻る

歩行者自律航法の結果は、

- 歩行を重ねるにつれ、誤差を蓄積するという特徴がある。現在位置表示用の位置算出には、
- リアルタイムに算出できること
- 歩行者が実際にいる道の上に補正できることが必要である。

## 3. 関連研究

マップマッチングはカーナビゲーションで広く利用されている。車の位置を特定するGPSに対するマップマッチングとしては、GPS測位結果を用いて算出した絶対位置と進行方向から道との評価値を算出し、評価値が最大となるリンクを選定し補正する手法がある。田川らは、歩行者自律航法にこの補正を応用して、選定した道に現在位置を補正し、進行方向を一致させることで、蓄積誤差を除去している[1]。しかし、人が道の方向に対して横方向に歩行する場合、人の進行方向と道の方向が異なるため、現在位置を

<sup>†</sup>株式会社リコー  
Ricoh Institute of Information and Communication Technology, RICOH COMPANY, LTD.

人が実際にいる道に補正できない。また、補正した道でセンサの蓄積誤差を除去するので、一度間違えるとその後も間違えてしまう。

大倉らは、ベクターマップマッチングを用いて高精度に歩行経路を推定するシステムを研究開発したが、ある程度まとまった量の測定データを解析しており、リアルタイムの補正には、対応していない[2]。

また、ビットマップデータを用いて、歩行可能エリアと不可能エリアを色分けし、不可能エリアに入った場合、歩行可能エリアに補正する技術がある。しかし、補正した歩行可能エリアが間違っていると、その後も間違った補正をしてしまうという問題がある。

## 4. 提案手法

### 4.1 全体像

本提案では、経路候補のリストを保持し、歩行者自律航法の測位結果（以降測位結果とする）が得られる毎に、経路候補のリストの中から測位結果の軌跡との類似性が高い（本提案では類似性を、類似性が高いほど小さい値になる類似度で表す）経路候補を選択し、その経路候補上に現在位置を補正（以降補正した位置を補正結果とする）する。この手法により、過去の補正結果にとらわれない補正が可能となり、1度実際に歩いていない経路候補で補正しても、歩いていくうち実際に歩いている経路候補と測位結果の軌跡の類似度が小さくなる（類似性が高くなる）ため、実際に歩いている経路候補で補正が出来る。

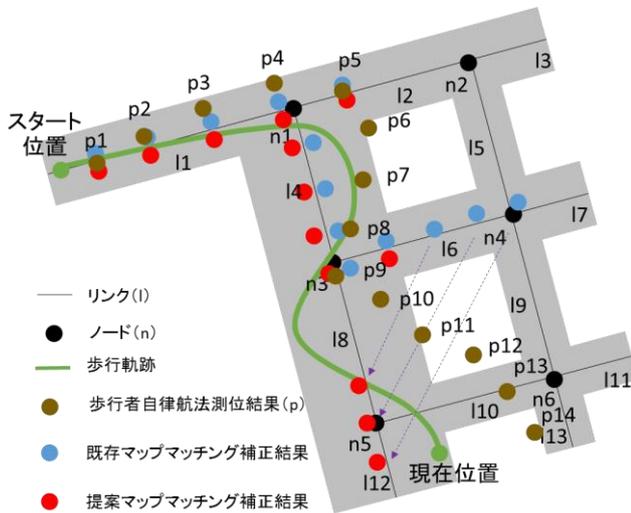


図 2 提案手法の補正結果

経路候補の概念は、通った道と進んでいる道を結んだもので、測位結果が得られる毎に、進んでいる道から交差点で曲がり、直線で繋がっていない道に進んだ（以降曲がったと表現する）可能性がないか確認する。曲がった可能性があれば、元の経路候補を保持したまま経路候補のリストに新たな経路候補を追加する。

図 2 は、スタート位置から歩き始め、n1 の交差点で曲がり、現在位置まで歩いた例である。図 3 で経路候補を追加する過程を説明する。経路候補のリストには初期状態では、スタート位置がある道を進んでいる経路候補 1 がある。歩行していき、p6 の測位結果が得られたときに経路候補 1 から p5 で n1 を曲がった経路候補 2 と n2 を曲がった経路候補 3 を追加する。さらに歩行して行き、p10 が得られたとき経路候補 2 から p9 で n3 を曲がった経路候補 4 と n5 を曲がった経路候補 5 を、経路候補 3 から p9 で n4 を曲がった経路候補 6 と n6 を曲がった経路候補 7 を追加する。図 4 は現在位置まで歩いたときの全てで 7 つの経路候補を示している。

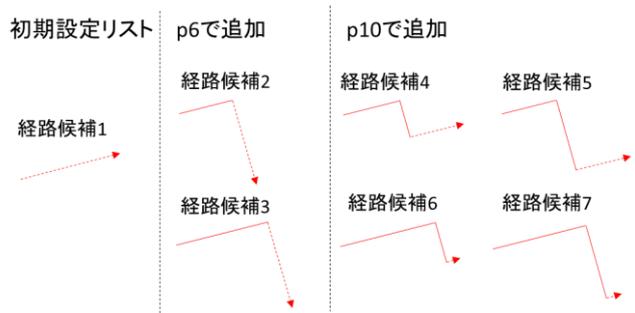


図 3 経路候補の追加過程

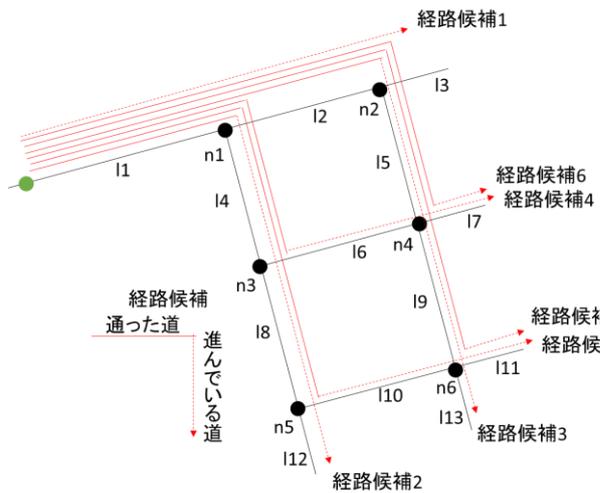


図 4 現在位置時点の経路候補

経路候補と測位結果の軌跡の類似度は、曲がったノード間と曲がった測位結果間の距離の比、曲がったノード前後と曲がった測位結果前後の角度差に加え、進行方向に対する左右のぶれ幅を使って求める。本提案では、曲がった可能性があり経路候補を追加しても、元の経路候補を保持している。図 5 のようスタート位置から現在位置まで歩いた場合、経路候補 B が実際に歩いた経路である。しかし、距離の比・角度差のみで比較すると、経路候補 A の類似度が小さくなる可能性がある（類似性が高いほど類似度が小さい）。そこで、歩行者自律航法では歩行の軌跡を表現できるため、そこから算出できる、進行方向に対する左右のぶれ幅を考え、曲がった回数が少ない候補が優位にならないよ

う、類似度の算出を行う。

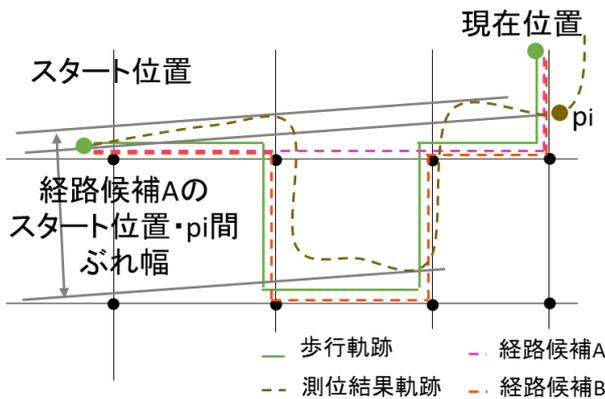


図 5 ぶれ幅の考え方

全体のブロック図を図 6 に示す。測位結果は、歩行者自律航法の結果で、時刻・座標からなる。マップマッチングでは、測位結果の取得を行うごとに、このとき取得した測位結果を補正した補正結果の算出を行う。

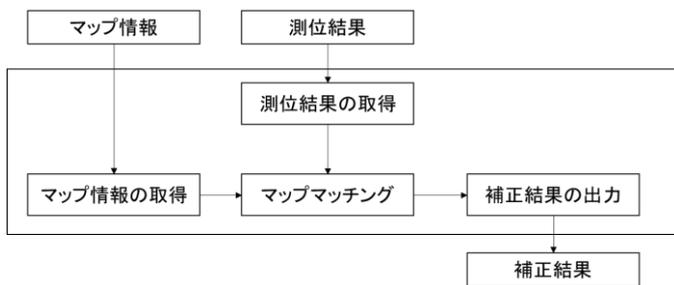


図 6 ブロック図

マップマッチングの処理の流れを図 7 に示す。人は進み始めた方向に進み続けるとは限らず、進んできた道に戻る場合もある。進んできた道に戻る場合に対応するため、進み始めた方向の最も遠い測位結果（以降遠い位置とする）を保持し、新たに得た測位位置（以降得た測位結果とする）が遠いか判断し、結果により以降の処理を変えている。通常は、ここで遠い位置と判断し、遠い位置を得た測位結果に更新する。遠い位置でないと判断した場合は、各処理で遠い位置を使うことで対応する。以降、通常の、遠い位置と判断する場合を先に説明し、遠い位置かの判断と遠い位置でないと判断した場合の処理については、4.6 で説明する。ノード曲がり可能性判断で、リストに保持してある経路候補から曲がった可能性があるか判断し、曲がった可能性があれば、経路候補追加を行う。経路候補を無駄に増やさないため、保持してある経路候補（追加した経路候補を除く）と測位結果の軌跡に乖離がないか確認し、乖離があれば経路候補を削除する。経路候補ごとに測位結果の軌跡との類似度を求め、類似度が小さい順に並び替え、先頭の経路候補で補正結果を算出する。また、経路候補を無駄に増やさないため、閾値を設け、閾値以降を削除する。

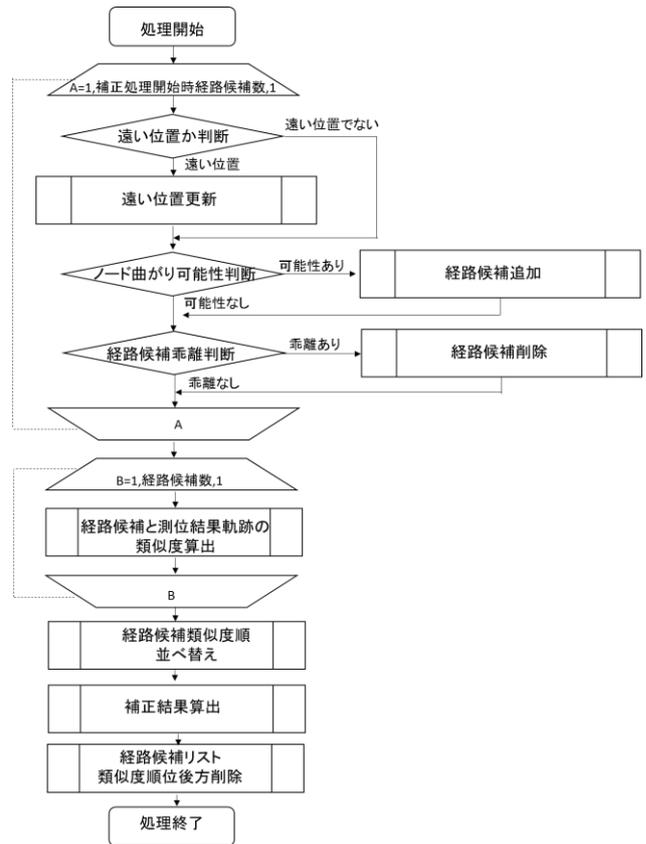


図 7 マップマッチングの処理フロー

経路候補はそれぞれ、下記の情報を保持する。

- 曲がった最終ノード：CN
- CN の前に曲がったノード：BCN
- 進行方向ノード：FN
- 曲がった最終測位結果：CP
- CP の前に曲がった測位結果：BCP
- 遠い測位結果：FP
- 曲がった回数：N
- 曲がり間距離比平均・二乗和：AL・SL
- 曲がり前後角度差平均・二乗和： $A\alpha \cdot S\alpha$
- 幅×距離の和：TW
- 曲がり間の距離の和：TL

曲がった最終ノード・曲がった最終測位結果は、経路候補の最後に曲がったノードと測位結果である。図 3・図 4 の経路候補 2 を例にすると、CN は n1、CP は p5 である。進行方向ノードは、曲がった最終ノードから進み始めたリンクの曲がった最終ノードと反対側の端ノードで、図 4 の経路候補 2 を例にすると、FN は n3 である。来た道に戻って進行方向が変わっても、FN は変更しない。曲がり間距離比平均・二乗和、曲がり前後角度差平均・二乗和、幅×距離の和は、経路候補と測位結果の軌跡の類似度算出に使用する値である。詳細は、4.3 で説明する。初期状態では、CN・CP にスタート位置を設定する。

## 4.2 ノード曲がり可能性判断と経路候補追加

ノード曲がり可能性判断は、曲がりから進んだ距離、変化した進行方向の角度が、ノード間・測位結果間で同程度の場合に曲がった可能性があるとして判断する。

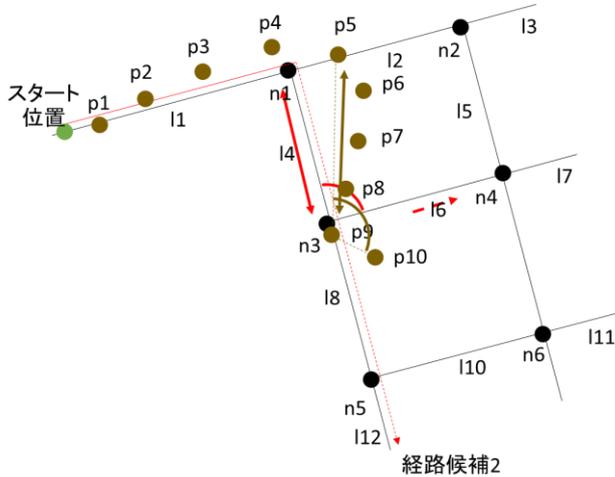


図 8 曲がりの判断指標

図 8 で、p10 を得たときに経路候補 2 から曲がった可能性があるか判断する場合を例に説明する。経路候補 2 は、スタート位置から歩き始め、p5 で n1 を曲がり、n3 の方向へ進み始めたとする経路候補である。このとき、CN が n1、BCN がスタート位置 (曲がった回数が 1 回のみの場合スタート位置を入れる)、CP が p5、BCP がスタート位置、FP が p10、FN が n3、N が 1 となる。

n3 から、n4 の方向へ曲がった可能性があるかは、距離が同程度かを式(1)で、角度が同程度かは式(2)で判断し、共に満たす場合、一つ前に得た p9 で n3 を曲がった可能性があるとして判断する。pi は p10、pi-1 は p9、JN は、n3、JFN は n4 となる。TLmin・TLmax・Tα は閾値である。

$$TLmin < \frac{\text{length}(CN, JN)}{\text{length}(CP, p_{i-1})} < TLmax \quad (1)$$

$$\left| \frac{\text{Angle}(CP, p_{i-1}, p_i) - \text{Angle}(CN, JN, JFN)}{\text{Angle}(CN, JN, JFN)} \right| < T\alpha \quad (2)$$

ただし、曲がり途中で何度も近くの測位結果で曲がったとする候補を増やさないため、前の点でも曲がったと判断された場合は、だぶりを防ぐために追加しない。この判断では、pi は p9、pi-1 は p8 となる。

曲がり可能性は、進み始めたリンクと直線で繋がるリンクのノード毎に、判断する。直線で繋がるかを、リンクとリンクの角度差が、微小な値以下かを式(3)で判断し、満たす場合直線で繋がると判断する。図 8 の例では、14・18 が直線で繋がるか、ni は n1、nj は n3、nk は n5 で確認する。このときは、直線なので、さらに ni は n3、nj は n5、とスライドして確認していく。TαS は閾値である。

$$|\text{Angle}(n_i, n_j, n_k)| < T\alpha S \quad (3)$$

14 と 18 は直線でつながっているため、n3 同様 n5 につい

ても、曲がった可能性があるか判断する。この場合、JN は、n5、JFN は n6 となる。

曲がり可能性のあるノードが見つかるごとに、保持する値を設定し、経路候補を追加する。図 8 の例で経路候補 2 から n3 を p9 で曲がった、経路候補を追加する場合、CN が n3、BCN が n1、CP が p9、BCP が p5、FP が p10、FN が n4、BN が n3、N が 2 の経路候補 4 を追加する。

## 4.3 経路候補乖離判断と経路候補削除

直線で繋がっているリンクがないのに進み続けている場合や、曲がった最終測位結果から得た位置まで進んだ方向が曲がった最終ノードから進んだリンクの方向と大きく異なる場合は、経路候補と測位結果の軌跡に乖離が生じたと考え、経路候補を保持していても意味がないため削除する。進み続けているかは式(4)、方向が大きく異なるかは式(5)で判断し、いずれか 1 つでも満たさない場合、乖離していると判断する。LFN は、曲がった後に進んだリンクと直線でつながる最も遠いリンクの遠い端のノード、TαD は閾値である。

$$TLmin < \frac{\text{length}(CN, LFN)}{\text{length}(CP, p_i)} \quad (4)$$

$$|\text{Angle}(BCP, CP, p_i) - \text{Angle}(BCN, CN, FN)| < T\alpha D \quad (5)$$

## 4.4 経路候補と測位結果軌跡の類似度算出

経路候補と測位結果軌跡の類似度は、曲がったノード間と測位結果間の距離の比、曲がったノード前後と曲がった測位結果前後の角度差、横に移動した幅から DL, Dα, DW を求め、足し合わせることで算出する。

$$D = DL + D\alpha + DW \quad (6)$$

曲がり間の距離の比と曲がり前後の角度差は、平均値がそれぞれ 1,0 に近く、標準偏差が小さければ、経路候補と測位結果の軌跡の類似性が高いといえる。そこで、DL を式(7)で、Dα を式(8)で算出する。CLi は経路候補の i 番目と i-1 番目の曲がったノード間の距離を i 番目と i-1 番目の曲がった測位結果間の距離で割った値である。Cαi は経路候補間の i+1 番目と i 番目と i-1 番目の曲がったノードの角度を i+1 番目と i 番目と i-1 番目の曲がった測位結果の角度から引いた値である。

$$D\alpha = \left| \frac{\sum_{i=1}^N C\alpha_i}{N} \right| + \sqrt{\frac{\sum_{i=1}^N C\alpha_i^2}{N} - \left( \frac{\sum_{i=1}^N C\alpha_i}{N} \right)^2} \quad (7)$$

$$DL = \left| \frac{\sum_{i=1}^N CL_i}{N} - 1 \right| + \sqrt{\frac{\sum_{i=1}^N CL_i^2}{N} - \left( \frac{\sum_{i=1}^N CL_i}{N} \right)^2} \quad (8)$$

曲がった測位結果・曲がったノードを全て保持しなくても DL, Dα を算出できるようにするため、CLi, Cαi の平均値・2乗和を保持している。

$$AL = \frac{\sum_{i=1}^N CL_i}{N} \quad (9)$$

$$SL = \sum_{i=1}^N CL_i^2 \quad (10)$$

$$A\alpha = \frac{\sum_{i=1}^{N-1} C\alpha_i}{N} \quad (11)$$

$$S\alpha = \sum_{i=1}^{N-1} C\alpha_i^2 \quad (12)$$

DL, Dαは, AL, SL, Aα, Sαを用いて, 式(13)・式(15)で算出する.

$$DL = |AL - 1| + \sqrt{\frac{(SL - AL^2)}{N}} \quad (13)$$

$$C_N = (\text{Angle}(BCP, CP, p_i) - \text{Angle}(BCN, CN, FN)) \quad (14)$$

$$D\alpha = \left| \frac{A\alpha \times (N - 1) + C_N}{N} \right| + \sqrt{\frac{S\alpha + C_N^2}{N} - \left( \frac{A\alpha \times (N - 1) + C_N}{N} \right)^2} \quad (15)$$

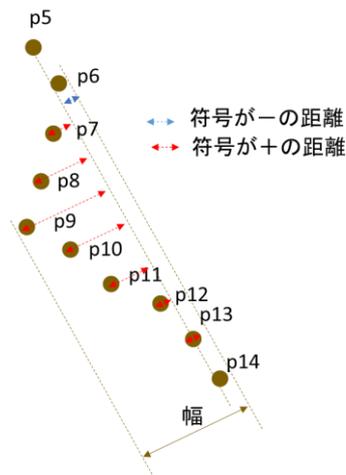


図 9 幅の求め方

横に移動した幅が小さければ, 経路候補と測位結果の軌跡の類似性が高い. p5 から p14 までの測位結果の横に移動した幅を求める場合, 直線 p5, p14 と p6 から p13 の距離をそれぞれ求める. このとき, 進行方向に対して右側はプラス, 左側はマイナスとする. それらの最大値から最小値を引いた値が幅である. 幅の項 DW は式(16)で算出する.

$$DW = \frac{TW + W \times L}{TL + L} \quad (16)$$

#### 4.5 経路候補並べ替えと補正結果算出と経路候補リスト後方削除

このようにして, 経路候補毎に類似度を算出した後, リストの経路候補を類似度が小さい順に並び替え, 先頭の経路候補で補正結果を算出する. また, 経路候補を無駄に増やさないために, 閾値を設け, 経路候補が追加されていき, リストの経路候補の総数が閾値以上になった場合は, 閾値以降の経路候補を削除する.

補正結果の算出は, 過去の測位結果の距離の傾向を反映するため, 曲がったノード間と曲がった測位結果間の距離の比を, 曲がった最終測位結果と得た測位結果の距離に掛

けることで, 進んだ距離 RL を式(17)で求める. 曲がった最終ノードから進んだリンクと直線に繋がるリンク上の, 進んだ距離だけ離れた位置を補正結果として算出する. 図 10 の経路候補 2 で p14 の補正結果を求める場合, CP は p5, pi は p14 である.

$$RL = \text{length}(CP, p_i) \times AL \quad (17)$$

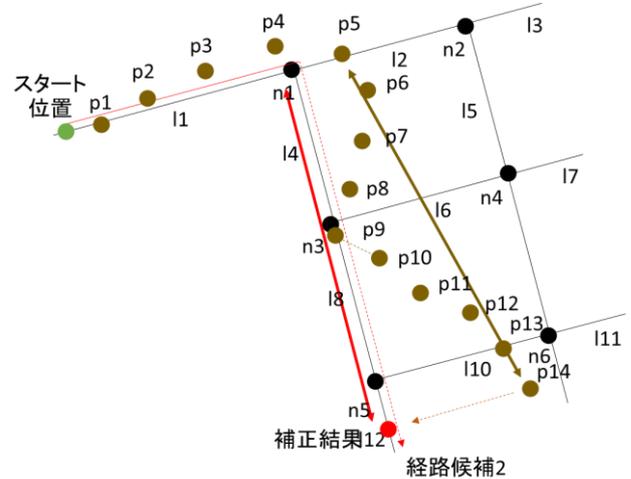


図 10 補正結果の算出方法

#### 4.6 遠い位置か判断と遠い位置でない場合の処理

4.2・4.3・4.4 と, 曲がった後進み始めた方向に進み続ける場合について説明してきた. しかし, 曲がった後進んだ方向に進み続けるとは限らない. 得た測位結果が遠い位置か判断し, 遠い位置でない場合, 各処理で遠い位置 FP を使うことで, 曲がった後進んできた道に戻る場合に対応する.

測位結果が遠い位置かの判断は, 曲がった測位結果・遠い位置・得た測位結果の角度で判断し, 式(18)を満たす場合, 得た測位結果が遠い位置と判断する.

$$|\text{Angle}(CP, FP, p_i)| > \frac{\pi}{2} \quad (18)$$

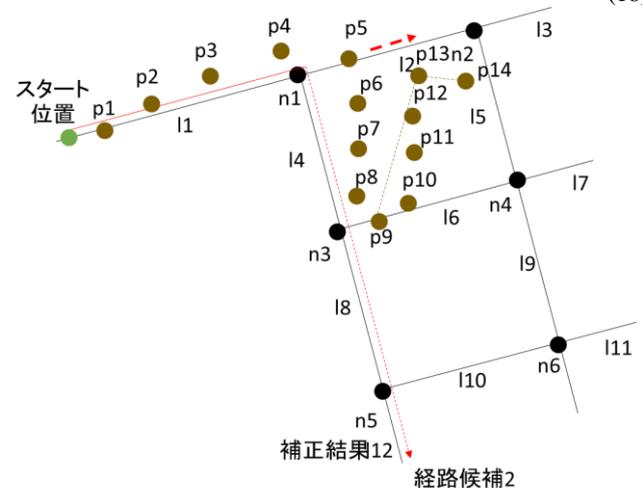


図 11 来た道に戻る例

図 11 で, n1 を p5 で曲がった経路候補 2 を例にすると,

p6～p9 まで、各測位結果を得たときに、式(18)を満たすため、FP を得た測位結果に更新する。p10 以降は、得た測位結果が式(18)を満たさないため、FP は p9 のままになる。

FP が、得た測位結果でない場合、ノード曲がり可能性判断の角度が同程度かの判断では、FP を使い式(19)で判断する。図 11 の p14 を得たとき経路候補 2 から、n1 を n2 の方向へ曲がったかの判断では、pi は p14、pi-1 は p13、FN は p9、JN は n1、JFN は n2 となる。

$$\left| \frac{\text{Angle}(FP, p_{i-1}, p_i) - \text{Angle}(FN, JN, JFN)}{\text{Angle}(FN, JN, JFN)} \right| < TA \quad (19)$$

## 5. 評価

提案手法の有効性を確認するため、道の方向に対して横方向に歩行する場合（横移動評価）と進んできた道に戻る場合（戻る評価）に、実際に歩いた経路に補正されているか検証した。

### 5.1 評価方法

歩行者自律航法のアプリケーションが入ったデバイスを保持して歩いた後、PC 上で測位結果を入力し検証を行った。歩行者自律航法の測位結果は、弊社で開発した[3]を用いて算出する。本方式では、一歩ごとではなく一定時間ごとに測位結果が算出される。補正結果は測位結果毎に提案手法を適用して算出する。また、評価指標を算出するための、正解位置は、アプリ上のボタンを押すことで記録し、その間は等速で歩いたものとして位置を計算する。マップマッチングからは、補正に使用した経路候補が正しいか確認するため、補正結果に加え、一定時間ごとに算出された補正結果補正それぞれの、結果算出に使った経路候補の曲がったノードを曲がった回数分出力する。

### 5.2 評価指標

精度・経路正解率・最終経路正誤を算出する。一定時間ごとに算出された測位結果・補正結果と正解位置の距離をそれぞれ算出して平均した値とする。経路正解率は、曲がった時刻の記録から、一定時間ごとの正解経路の曲がったノードを定め、各時刻の出力した曲がったノードと正解経路の曲がったノードが全て同じであれば正解とし、正解率を求める。母数は、一定時間ごとに算出した補正結果の総数である。最終経路正誤は、最後の補正結果算出に使った経路候補の曲がったノードと、正解経路の曲がったノードが全て同じであれば正解とする。

### 5.3 評価コース

評価コースはリコー中央研究所内に設定した。

#### 5.3.1 横移動評価コース

横移動評価では、曲がった回数が多い経路候補・曲がっ

た回数が少ない経路候補のどちらかが優位になっていないことを確認するコース 1・コース 2 を検証した上で、道の方向に対して横方向に歩行するコース 3 を検証する。

コース 1 では図 12 の緑の点線に沿って時計回り・反時計回り (A→B→C→D→A→B・B→A→D→C→B→D) に歩行する。コース 2 では、コース 1 の E・F 間は並行する道に図 13 のように曲がり青の点線に沿って時計回り・反時計回り (A→B→C→E→G→H→F→D→A→B) に歩行する。コース 3 は、歩く道はコース 1 と同じだが、F 地点で並行する道の方へ図 14 のように移動する。各コース時計回り 5 回・反時計回り 5 回、の計 30 回の歩行を行った。コース 3 では、横方向に進んでいる部分は、F・H を進んでいるともいえるので、F 地点で曲がり進んで戻って F を曲がり元の道を進む経路も、正解経路とする。

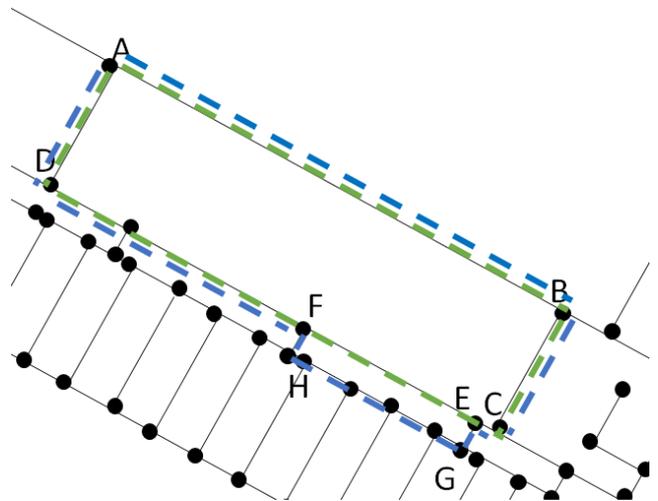


図 12 横方向評価コース

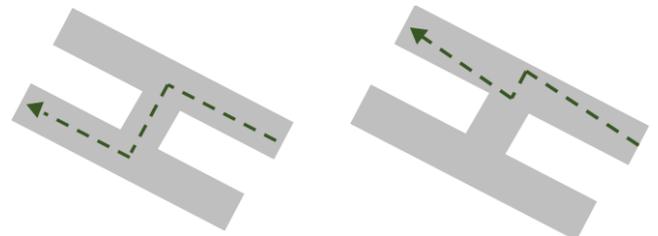


図 14 コース 2 の曲がり 図 13 コース 3 の横移動

#### 5.3.2 戻る評価コース

戻る評価では、歩いて、曲がり、歩き、戻って曲がるという歩行を、戻った後に曲がった交差点より手前で曲がる・同じ交差点で曲がる・奥で曲がるという 3 パターンを、右折・左折の組み合わせ 4 パターンの計 12 パターンで各 5 回ずつ計 60 回の歩行を行った。

##### 右折・右折

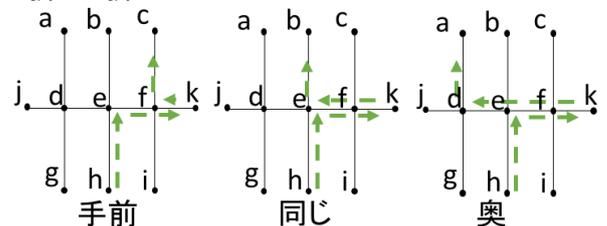


図 15 戻る評価コースパターン\_1

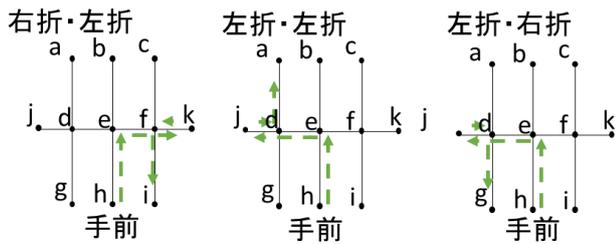


図 16 戻る評価のコースパターン\_2

5.4 結果

5.4.1 横移動評価結果

いずれのコースも測位結果と比べ補正結果精度が向上した。コース3は、経路正解率平均は64%で、最後の測位結果を6回(10回中)正しい経路候補で補正することができた。歩行途中に関しては、H・Gの道に入っている経路候補で補正しているか確認したところ、9回(10回中)がH・Gの道に入っていた。(9回には、F・H以外で曲がった場合も含む)。

表 1 横移動評価結果\_精度

	測位結果 精度平均	補正結果 精度平均
コース 1	4.91(m)	3.74(m)
コース 2	7.74(m)	6.26(m)
コース 3	5.95(m)	4.65(m)

表 2 横移動評価結果\_経路正誤

	経路正解率 平均	最終経路候補 正解数
コース 1	81.7(%)	8 回 (10 回中)
コース 2	64.3(%)	6 回 (10 回中)
コース 3	64.0(%)	6 回 (10 回中)

5.4.2 戻る評価結果

手前・同じ・奥いずれのパターンも、測位結果と比べ補正結果精度が向上した。経路正解率平均は、手前・同じ・奥いずれも60%以上で、最後の測位結果を9回(20回中)以上正しい経路候補で補正することができた。

表 3 戻る評価結果\_精度

	測位結果 精度平均	補正結果 精度平均
手前	4.20(m)	3.80(m)
同じ	7.13(m)	6.84(m)
奥	3.59(m)	3.51(m)

表 4 戻る評価\_経路正誤

	経路正解率 平均	最終経路候補正 解数
手前	77.0(%)	9 回 (20 回中)
同じ	61.4(%)	16 回 (20 回中)
奥	72.9(%)	13 回(20 回中)

5.5 考察

5.5.1 横移動評価考察

曲がった回数に違いがある、コース1とコース2を比較すると経路正解率がコース2の方が低い。コース2で補正結果を算出する補正経路のノードを確認すると、曲がった回数が少ない経路候補で補正したのは2回のみだった。この結果から、コース2はコース1より経路が複雑なため、経路正解率が落ちているが、曲がった回数が多い・少ないどちらかが優位になることなく補正結果が求められたと考える。

その上で、コース3では道の中で横方向に動き、途中経路を間違えた9回中5回で、最後の測位結果で実際の経路で補正でき、道の中で左右に動き間違った経路で補正を行っても、歩行するにつれ実際の経路で補正されることが確認できた。しかし、コース2・3の経路正解率は60%程度で、高いとはいえない。今後、リンクの幅情報の取得手段と、交差点曲がり可能性判断・経路候補と測位結果軌跡の類似度算出にリンクの幅を使う新たな手法を検討し、経路正解率の向上、補正結果精度の向上を目指す。

5.5.2 戻る評価考察

来た道に戻る場合でも、測位結果に比べ精度高く補正結果を求めることが出来き、60%以上で実際に歩いた経路で補正することが出来た。しかし、経路正解率が高いとはいえない。本提案手法では、交差点曲がり可能性判断・経路候補と測位結果軌跡の類似度算出で、戻った距離分離

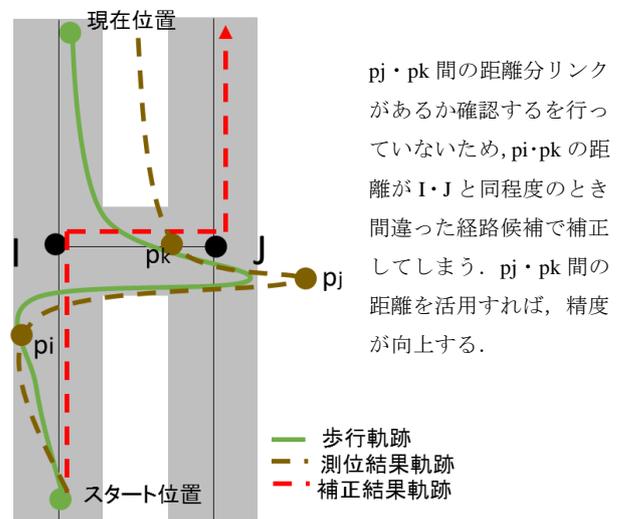


図 17 戻る場合の処理改善点

リンクが続いているかの判断を行っていない。今後、交差点曲がり可能性判断・経路候補と測位結果軌跡の類似度算出を改善していくことで、経路正解率の向上、補正結果精度の向上が出来ると考える。

## 6. まとめ

本稿では、道の方向に対して横方向に歩行する場合・進んできた道に戻る場合に考慮した、過去の補正結果に依存しないリアルタイムの歩行者自律航法測位結果のマップマッチングアルゴリズムの提案を行い、実際に提案手法の有用性を確かめる検証を行った。その結果、道の方向に対して横方向に歩行する場合・進んできた道に戻る場合共に、測位結果に比べ精度高く補正結果を求めることが出来た。また、60%以上で実際に歩いた経路で補正することができ出来た。今後、リンクの幅情報を使うなど更なるアルゴリズムの検討を行い、精度を向上していきたい。

## 参考文献

- [1] 田川達司, 内匠逸, 打矢隆弘, カルマンフィルタとマップマッチングを用いた歩行者経路の推定精度向上に関する研究, 第75回全国大会講演論文集, 2013(1), pp.207-208 (2013-03-06).
- [2] 大倉輝, 山本寛, 山崎克之, Android による屋内歩行経路推定システムの開発と評価 (インターネット計測, ネットワーク監視, ネットワークセキュリティ, トラヒック理論及び一般), 電子情報通信学会技術研究報告. IA, インターネットアーキテクチャ 111.347 (2011): 49-54.
- [3] 松下裕介, 吉澤史男, 金崎克己, 歩行者自律航法をベースとしたハイブリット屋内測位手法の提案, 第77回全国大会講演論文集 2015.1 (2015): 3-4.
- [4] 吉見駿, 村尾和哉, 望月祐洋, 西尾信彦, マップマッチングを用いた PDR 軌跡補正, 研究報告ユビキタスコンピューティングシステム (UBI), 2014(20), 1-8.
- [5] 北澤桂, 小西勇介, 柴崎亮介, Personal Positioning System におけるマップマッチング法の提案, 全国測量技術大会 2001, 学生フォーラム(2001)