

平面図と断面図による軌跡可視化のための Line Simplification 手法

村上 司[†] 藤田 秀之[†] 大森 匡[†] 新谷 隆彦[†]

[†] 電気通信大学 大学院情報理工学研究科 〒182-8585 東京都調布市調布ヶ丘 1-5-1

E-mail: [†]tsukasa.murakami@uec.ac.jp, ^{††}{fujita,omori,shintani}@is.uec.ac.jp

あらまし ランニング、サイクリング、登山などの移動を伴う運動による軌跡データの収集・共有が身近になった。軌跡データは、緯度と経度の二次元平面で可視化した軌跡平面図と、起点からの距離と標高の二次元平面で可視化した軌跡断面図で可視化されることが多い。その際、形状を保ちながら軌跡の頂点を間引きデータサイズを削減する Line Simplification (LS) は重要な技術である。しかし、主要なアプリケーションは、軌跡平面図の LS で残った頂点から軌跡断面図を生成しており、この方法では、軌跡断面図上の山頂や谷といった顕著な頂点が消失してしまうことがある。そこで本研究では、3 次元軌跡データを入力とし、軌跡平面図と軌跡断面図の両者を適切に LS する手法を提案する。

キーワード 可視化、情報可視化、軌跡データ、地図、モバイルアプリケーション

1 背景と目的

ランニング、サイクリング、登山などの移動を伴うアクティビティによる軌跡データの収集・共有が身近になった。例えば、移動を伴う運動のフィットネスデータを収集・共有するサービスを提供する STRAVA は、2022 年時点でユーザー数 1 億人、アクティビティ数 70 億件を突破している [1]。

軌跡データは、緯度、経度、高度（標高）の 3 次元空間の位置のリストで構成され、地図と標高グラフにより可視化されることが多い。例えば、図 1 は収集した軌跡データの可視化と共有ができる、Garmin Connect というアプリケーションの画面である。東京都調布市から東京都檜原村までを往復でサイクリングしたときの軌跡であり、サイクルコンピュータにより収集した軌跡データである。図 1 には軌跡データを地図上に可視化したグラフと、軌跡上の起伏を可視化したグラフが表示されている。以下、前者のグラフを軌跡平面図、後者のグラフを軌跡断面図と呼ぶことにする。

移動を伴うアクティビティの中で特にサイクリングや登山では、標高に関する指標を重要視している。例えば、累積標高という概念は上った高さの総和、あるいは下った高さの総和を表す。累積標高は平坦な道では小さく、アップダウンの多い道では大きくなる。また、勾配という概念は坂道の傾きの程度を表す。勾配は上り坂の傾斜が緩い道では小さく、傾斜が急な道では大きくなる。

軌跡断面図からは、最高標高地点や最低標高地点、上り坂や下り坂の勾配、起伏が多いのか少ないのかなど標高に関する情報を読み取ることができる。標高に関する情報が重要になっている昨今、軌跡平面図だけでなく軌跡断面図による可視化も重要になっている。

軌跡データは、移動する物体の位置を定期的に収集することで生成される。軌跡データのサイズは、位置データの収集頻度や移動距離に依存する。すなわち、単位時間当たりの収集頻度が多いほど、または移動距離が長くなるほど巨大になりやすい。巨大なデータは時間的・空間的にコストがかかるため、形状を保ちながら軌跡の頂点を間引いてデータサイズを削減する Line Simplification (略称：LS) という技術が重要である。

主要なアプリケーションでは、軌跡平面図に LS を適用して得られる軌跡データを用いて、軌跡平面図と軌跡断面図を可視化している。このとき、各地点の緯度経度の情報のみで LS が適用されるため、標高の情報は考慮されない。よって、軌跡断面図において山頂や谷のような顕著な極値が消失してしまうことがある。加えて、軌跡断面図において適切に形状が維持されないことがある。

図 2 は軌跡平面図に LS を適用して得られた軌跡データを軌跡断面図として可視化したものである。LS 後の軌跡が 7km 付近にある最高標高地点を残していないのが読み取れる。

新たな LS 手法には、軌跡平面図と軌跡断面図それぞれの形状を維持しながら、軌跡断面図における山頂や谷のような顕著な極値を残す性質が必要である。本研究は、そのような性質を持つ、軌跡平面図と軌跡断面図の両者を適切に LS する手法を提案する。



図 1 主要アプリによる軌跡データの可視化 (出典: Garmin Connect)

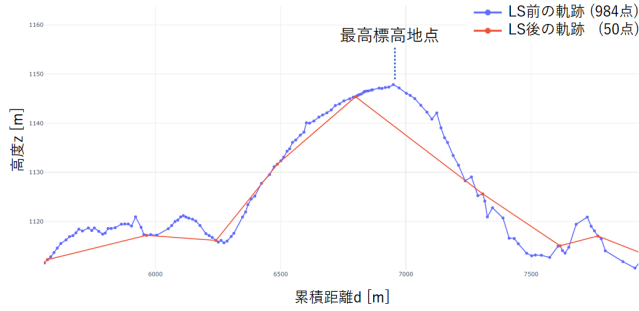


図 2 課題の実例（軌跡断面図表示）

本研究の貢献は以下のとおりである．

- 軌跡データを軌跡平面図と軌跡断面図で可視化する際の、現状の LS 適用方法の問題点を示す
- 軌跡平面図と軌跡断面図ともに、形状と特徴に関する誤差が閾値以内となる 3 次元軌跡データを出力する、新しい LS 手法を提案する
- 提案手法を含む複数の LS 手法を実データに適用して評価し、提案手法の有効性を示す

2 関連研究

Line Simplification は、地図学において、広域の縮尺の地図描画のために、省略や転移、誇張等により、読み取りやすさを考慮して図形を簡略化する、総描 [2] の技術のひとつとして提案され [3] [4]、計算幾何学分野で処理の高速化 [5] [6] が進み、主要な空間 DBMS [7] や GIS アプリケーションに実装され、近年では、サーベイ論文 [8] [9] がまとめるとおり、主にデータベース分野で、大規模データ処理における実用性向上が検討され (例えば [10]), 軌跡データマイニングにおける前処理の主要技術にも位置付けられ [11], さまざまな新しい性質を持たせた LS の手法が提案されている．一例として、軌跡間や軌跡と周辺の地物間のトポロジーを LS 前後で維持する手法 [12], ストリーミングデータとして与えられる軌跡データをリアルタイムに処理する LS 手法 [13], LS 後の軌跡上での時空間的な検索の性能を考慮し、軌跡間の同期距離に基づく LS 手法 [14], 移動体ごとの移動特性を考慮した LS 手法 (例えば船舶 [15]), 移動方向の誤差に基づく LS 手法 [16] 等があげられるが、すべて平面図のみを考慮した提案であり、断面図に特化した提案は、筆者らの知る限り存在しない．他方で、地形表現の単純化の手法は複数提案されており、例えば、2.5 次元的地形表現に用いられる TIN (Triangulated Irregular Networks) を対象に、3 次元 CG の単純化で一般的な Surface simplification を、地形の特徴となる地点を残すよう発展させた手法 [17] もあるが、断面図にしたに特化した手法ではない．また、本研究の主要な応用である自転車のルート計画に関しては、自転車を対象とした標高を考慮した経路探索のために、現状の Digital Elevation Model (DEM) のデータのエラーを、カルマンフィルタ等に基づき修正する手法 [18] も提案されている．

3 定義

3.1 軌跡

軌跡 T は、頂点 p の順序付きリストと定義できる．頂点 p は経度 x 、緯度 y 、高度 z 、累積距離 d を持つ．累積距離 d_i は、軌跡 T 上の頂点 p_1 から頂点 p_i までの、 $x-y$ 平面へ射影した軌跡沿いの距離である．

$$p = (x, y, z, d) \quad (1)$$

$$T = (p_1, p_2, \dots, p_n) \quad (2)$$

3.2 軌跡の可視化手法

単なる位置の羅列である軌跡データを人間が理解できるようにするためには、情報可視化が必要である．軌跡データの可視化手法は、連続する頂点同士を直線で結んだ折れ線グラフによる表現が一般的である．軌跡平面図は、軌跡データを $x-y$ 平面上に可視化した折れ線グラフであり、道路の線形を読み取ることができる．軌跡断面図は、軌跡データを $d-z$ 平面上に可視化した折れ線グラフであり、道路の起伏を読み取ることができる．

図 3 に東京都檜原村から東京都奥多摩町までを結ぶ全長 19.7km の奥多摩周遊道路の軌跡データを軌跡平面図と軌跡断面図で可視化したものを示す．図 3 の軌跡平面図からは、奥多摩周遊道路がほとんど直線のないカーブの多い道路であることが読み取れる．図 3 の軌跡断面図からは、奥多摩周遊道路が 7km 付近にある最高標高地点を境に上り坂から下り坂に転じていることが読み取れる．

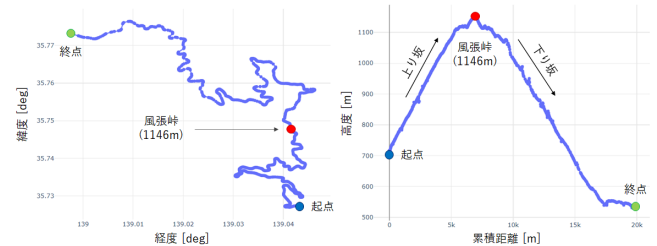


図 3 奥多摩周遊道路の軌跡平面図（左）と軌跡断面図（右）

3.3 Line Simplification (LS)

Line Simplification (LS) とは、もとの軌跡の形状を保ちながら点の数を削減する技術である．形状の維持に重要な点を残す、あるいは形状の維持に不要な点を削除することで点を間引く．制限として軌跡に新たに点を追加することや既存の点の位置を移動させることはできない．もとの軌跡 T と点の間引きの程度を決める閾値 ϵ を入力、単純化された軌跡 T' を出力とするとき、LS は次式で定義される．

$$T' \subseteq T \quad \text{かつ} \quad \text{Error}(T, T') < \epsilon \quad (3)$$

誤差測定手法 Error は軌跡 T と軌跡 T' の誤差を返す関数であり、定義の仕方によりさまざまなアルゴリズムが存在する．

軌跡 T の頂点数を n とすると、軌跡 T' の候補は n 個の各頂点について「残す」「残さない」の 2 通りの可能性があるため、 2^n 通りとなる。

図 4 に静岡県の境界データに対する LS の適用例を示す。頂点数が減少するにつれて細部の形状は簡略化されているが、静岡県全体の形状は維持されていることを読み取ることができる。

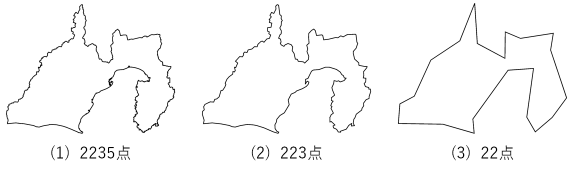


図 4 静岡県の境界データへの LS 適用例

3.4 Visvalingam のアルゴリズム

Visvalingam アルゴリズム [4] は、三角形の面積を利用した、代表的な LS 手法である。入力は軌跡 T と間引きの程度を決める三角形の面積の閾値 ϵ 、出力は単純化された軌跡 T' である。アルゴリズムの手順を Algorithm 1 に示す。

このアルゴリズムの時間計算量は $O(n^2)$ であるが、三角形の最小面積を求めるのにデータ構造の最小ヒープを用いると時間計算量は $O(n \log n)$ に改善される。軌跡上の点を 1 点ずつ間引いていくため、面積の閾値の代わりに軌跡の頂点数を指定することも可能である。

図 5 に頂点数 6 の軌跡と頂点数の閾値に 4 を入力したときの Visvalingam アルゴリズムの実行例を示す。

Algorithm 1 Visvalingam のアルゴリズム

Require: 軌跡 T , 面積の閾値 ϵ

Ensure: 軌跡 T'

```

1: while true do
2:   for all 頂点  $i \in T$  do
3:     面積  $S_i \leftarrow T$  の連続 3 頂点  $(i-1, i, i+1)$  による
       三角形の面積
4:   end for
5:   頂点  $i^* \leftarrow S_i$  最小となる  $i$ 
6:   if  $S_{i^*} < \epsilon$  then
7:      $i^*$  を  $T$  から取り除きループ継続
8:   else
9:      $T$  を  $T'$  として返す
10:  end if
11: end while

```

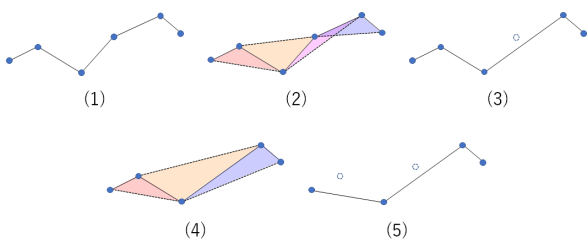


図 5 Visvalingam のアルゴリズムの実行例

4 提案する LS 手法

4.1 概要

3 次元軌跡データを入力とし、指定された閾値を満たすように、軌跡断面図において地形の特徴を提示するために重要な極値を残し、経路平面図と軌跡断面図の形状を維持しながら、点数をできるだけ削減した 3 次元軌跡を返す、新しい LS 手法を提案する。

軌跡断面図における重要な極値は、LS の際に無条件に残すものとする。ここで、平坦に見える場所でも、頂点の z 座標が完全に一致することは稀なため、前後の頂点より z 座標が高い、または、低いという定義に基づく極値は、軌跡断面図上に非常に多く存在する。そのため、地形の特徴を提示するために、LS 後も残すべき極値を抽出する方法は課題である。

本研究では、各極大値の顕著さの指標として、極値の相対的な突出度を表すプロミネンスを用いる。指定された極大値のプロミネンスは、自身より高い地点を含まない、自身の左右の範囲における各最小値のうち、小さい値からの自身の高さで定義される。プロミネンスが大きいほど極大値は突出している。プロミネンスは、信号処理や地誌学の分野で用いられる。例えば、富士山山腹に張り付いて見える宝永山の山頂は、標高 2,709m あるが、プロミネンスは 11m しかない。対して、蝦夷富士として知られる羊蹄山の山頂は、標高 1,898m だがプロミネンスは 1,878m と高く、広範囲から突出して見える。

軌跡断面図における山頂や谷は地形の重要な特徴である。これらは突出度の高い極値である。他方で、現地を移動中の利用者にとって、周辺が平坦な場所では低い丘でも地形の重要な特徴と考えられる。突出度の大きな極値のみを残すと、平坦な場所の極値は残らないという問題が生じる。そこで、軌跡断面図の任意の地点の周辺における、局所的に顕著な極値を抽出する手法を 4.2 節で提案する。

続いて、軌跡平面図・断面図の形状の維持に関しては、軌跡の形状を維持するため重要な地点は、軌跡平面図と軌跡断面図とで異なるという課題である。そこで本研究では、Visvalingam のアルゴリズムに即し、形状の維持において、平面図と断面図どちらにとっても重要でない地点から順に間引く手法を 4.3 節で提案する。

4.2 局所的に顕著な極値の検出

軌跡断面図 ($d-z$ 平面) における、 d 軸に平行な固定幅のウィンドウ内の局所的に顕著な極値を、 d 座標がウィンドウに含まれる部分軌跡上の、最大の極大値と、最小の極小値と定義する。ウィンドウ内に極値が存在しない場合は、局所的に顕著な極値も存在しない。提案する LS 手法において、無条件に残す頂点の集合を、 d 軸に沿ってスライドするウィンドウによる、任意の位置における局所的に顕著な極値となる頂点の集合とする。これは、局所的に顕著な極値となる条件を満たす頂点の集合として求めることができる。

軌跡 T 上の頂点 $p_m \in T$ が、軌跡断面図上で局所的に顕著な極値となる条件は次のとおりである．軌跡断面図上の頂点 p_m を中心とした、 d 軸に平行な固定幅 w のウィンドウ内に d 座標が含まれる頂点の集合を $P_{m,w} \subset T$ として、

$$z_m = \max_{p_i \in P_{m,w}} z_i \text{ を満たす.} \quad (4)$$

または

$$z_m = \min_{p_i \in P_{m,w}} z_i \text{ を満たす.} \quad (5)$$

ただし

$$z_{m-1} = z_m = z_{m+1} \text{ を満たさない.} \quad (6)$$

$$(p_{m-1}, p_m, p_{m+1} \in P_{m,w})$$

図 6、図 7 は頂点数 20 の軌跡に対する局所的に顕著な極値の検出例である．ウィンドウの幅が小さい図 6 では、頂点 p_5 はウィンドウ内において最大値をとるため、局所的に顕著な極値となる．一方、ウィンドウの幅が大きい図 7 では、頂点 p_5 はウィンドウ内において最大値も最小値もとらないため、局所的に顕著な極値とならない．このようにウィンドウの幅によって極値の検出数が異なる．

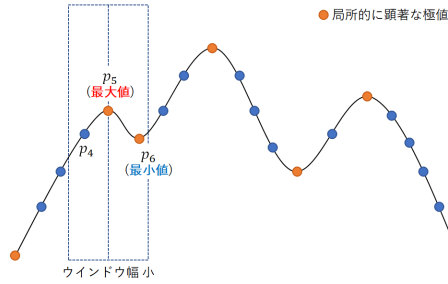


図 6 ウィンドウ幅が小さいときの局所的に顕著な極値の検出例

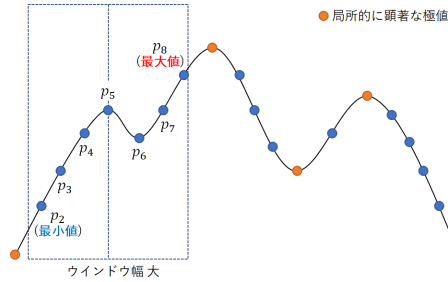


図 7 ウィンドウ幅が大きいときの局所的に顕著な極値の検出例

4.3 軌跡平面図と軌跡断面図に適した LS 手法

入力は軌跡 T 、面積の閾値 ϵ 、ウィンドウの幅 w 、面積の重み α 、出力は単純化された軌跡 T' である．ウィンドウの幅 w は極値を残す程度を決めるパラメータであり、大きいほど残る極値の数は減少する．面積の重み α は軌跡平面図と軌跡断面図のどちらの形状を優先的に維持するかを決めるパラメータであり、小さいほど軌跡平面図の形状を、大きいほど軌跡断面図の形状を優先的に維持する．アルゴリズムの手順を Algorithm 2 に示す．

Algorithm 2 提案する LS 手法

Require: 軌跡 T 、面積の閾値 ϵ 、ウィンドウの幅 w 、面積の重み α

Ensure: 軌跡 T'

```

1:  $E \leftarrow$  ウィンドウ幅  $w$  の局所的に顕著な極値群
2: while true do
3:   for all 頂点  $i \in T$  do
4:     面積  $S_i \leftarrow T$  の連続 3 頂点  $(i-1, i, i+1)$  による、軌跡平面図上と軌跡断面図上の各三角形の面積の重み  $\alpha$  付き線形和
5:   end for
6:   頂点  $i^* \leftarrow E$  に含まれない  $i$  のうち  $S_i$  最小となる  $i$ 
7:   if  $S_{i^*} < \epsilon$  then
8:      $i^*$  を  $T$  から取り除きループ継続
9:   else
10:     $T$  を  $T'$  として返す
11:   end if
12: end while

```

4.4 適用結果

図 8、図 9 はもとの軌跡に対して既存手法と提案手法を適用して得られた軌跡データを軌跡平面図と軌跡断面図で可視化したものである．軌跡断面図において既存手法は最高標高地点を残していないが、提案手法は残しているのが読み取れる．軌跡平面図では既存手法と比較して提案手法の形状が著しく悪化していないことが読み取れる．

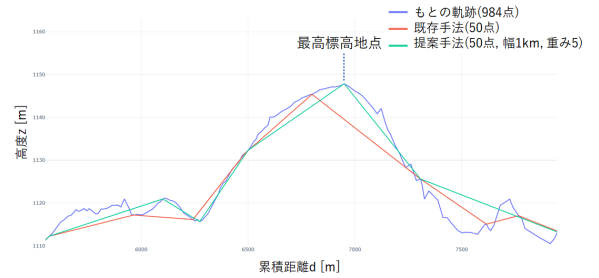


図 8 適用結果（軌跡断面図表示）

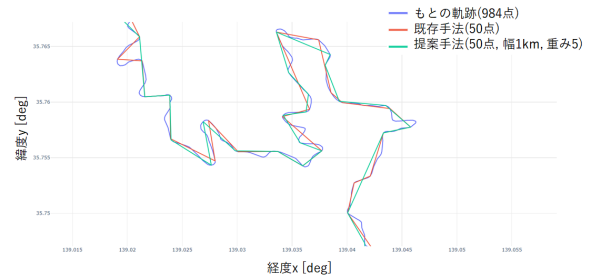


図 9 適用結果（軌跡平面図表示）

5 評価

5.1 評価実験の概要

既存手法、直観的手法、提案手法を対象に軌跡平面図と軌跡断面図を同時に生成する LS 手法の性能比較を行う。

LS 手法の性能比較実験の流れを説明する (図 10 参照)。入力 は 3 次元軌跡データである。比較対象とする 4 種類の LS 手法を入力データにそれぞれ適用し、同じ点数の 3 次元軌跡データを生成する。そして、生成された 3 次元軌跡データから、軌跡平面図と軌跡断面図をそれぞれ生成する。最後に、生成された軌跡平面図と軌跡断面図について、LS 前と比較した定量的な評価を行う。このとき、軌跡平面図では形状を維持しているかの観点、軌跡断面図では形状を維持しているかの観点と顕著な極値を維持しているかの観点で評価を行う。

比較対象となる 4 種類の LS 手法は以下の通りである。なお、各手法中の LS には Visvalingam のアルゴリズムを用いる。

1. 平面 LS (既存手法)

軌跡平面図上 ($x-y$ 座標による距離) で LS。

2. 断面 LS (既存手法)

軌跡断面図上 ($d-z$ 座標による距離) で LS。

3. 3 次元 LS (直観的手法)

3 次元空間上 ($x-y-z$ 座標による距離) で LS。

4. 提案手法

三角形の面積の重み α は 5 (経験的に決定)。

ウィンドウの幅 w は 1000 m (経験的に決定)。

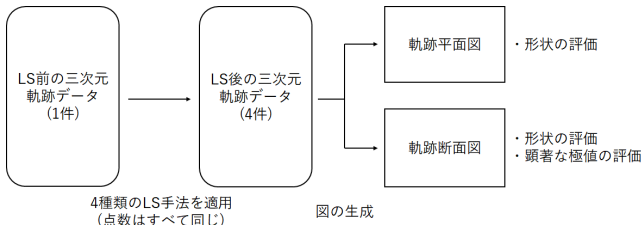


図 10 評価実験の流れ

5.2 軌跡データの作成

軌跡データの作成には、Mapbox 社の経路探索 API と国土地理院の標高タイルを利用する。経路探索 API に 2 地点を入力すると最適経路の軌跡データが得られる。この軌跡データの各地点には標高値が含まれないため、標高タイルから各地点の標高値を算出する。

標高タイルから得られる標高値は、地表面での標高値という制限がある。つまり、条件によっては道路上の標高値が取得できないことがある。例えば、トンネル区間ではトンネル上の山の標高値が取得されてしまう。

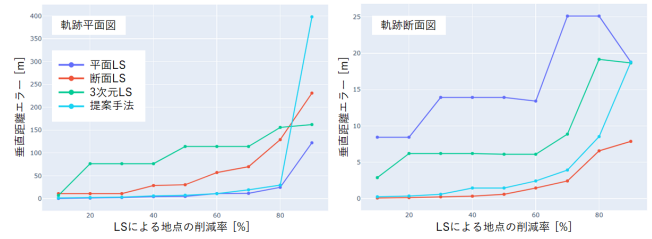


図 11 軌跡平面図と軌跡断面図における垂直距離エラー

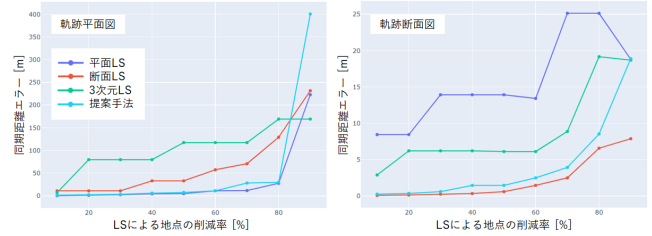


図 12 軌跡平面図と軌跡断面図における同期距離エラー

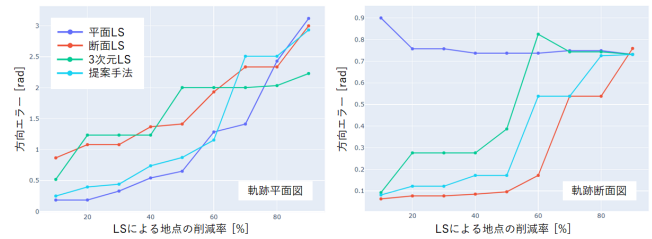


図 13 軌跡平面図と軌跡断面図における方向エラー

5.3 形状の維持に関する評価

形状の維持に関する評価では、垂直距離エラー [10]、同期距離エラー [10]、方向エラー [10] を用いる。これらは既存の LS 手法の評価で従来から用いられる誤差ベースの指標であり、小さいほどもとの形状を維持していることを表す。

図 11、図 12、図 13 は軌跡平面図と軌跡断面図における軌跡の頂点数の各削減率での垂直距離エラー、同期距離エラー、方向エラーのグラフである。どのグラフも削減率が大きくなるほど、もとの軌跡との誤差が大きくなっていることが読み取れる。

軌跡平面図のみの評価では、平面 LS、提案手法、断面 LS、3 次元 LS の順で良好な結果であることが読み取れる。軌跡断面図のみの評価では、断面 LS、提案手法、3 次元 LS、平面 LS の順で良好な結果であることが読み取れる。軌跡平面図と軌跡断面図を総合した評価では、提案手法が良好な結果であることが読み取れる。

平面 LS は軌跡平面図の形状の維持に重要な頂点のみを残していると考えられる。断面 LS は軌跡断面図の形状の維持に重要な頂点のみを残していると考えられる。提案手法は軌跡平面図の形状の維持に重要な頂点と、軌跡断面図の形状の維持に重要な頂点をバランスよく残していると考えられる。

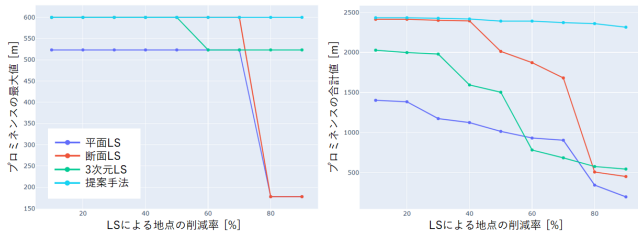


図 14 軌跡断面図における極値のプロミネンス

5.4 顕著な極値の維持に関する評価

顕著な極値の維持に関する評価では、極値のプロミネンスの最大値と合計値を用いる。

図 14 は軌跡断面図における軌跡の頂点数の各削減率での極値のプロミネンスの最大値と合計値のグラフである。どちらのグラフも、各削減率を通して評価値が最も大きくなっている、提案手法が良好な結果であることが読み取れる。

平面 LS、断面 LS、3 次元 LS は軌跡断面図における顕著な極値を考慮していないことが考えられる。一方、提案手法は軌跡断面図における顕著な極値を優先的に残していることが考えられる。

6 結 論

本研究では、軌跡平面図と軌跡断面図の両者に適した LS 手法の提案を目的として、3 次元軌跡データを入力とし、軌跡平面図と軌跡断面図それぞれの形状を維持しつつ軌跡断面図における顕著な極値を残す性質を持つ LS 手法を提案した。

提案手法は、軌跡断面図上の顕著な極値の検出に、固定幅のスライディングウィンドウによる局所的な極値の検出方法を用い、軌跡平面図と軌跡断面図の形状の維持に、軌跡平面図上と軌跡断面図上の各三角形の面積の重み付き線形和を用いるよう改変した、Visvalingam のアルゴリズムを用いた。

また、形状の維持と顕著な極値の維持に関する定量評価を行い、既存手法や直観的手法と比較して、提案手法が形状の維持と顕著な極値の維持を両立していることを示した。

今後の課題として、ウィンドウの幅や面積の重みのようなパラメータの自動調整が挙げられる。

文 献

- [1] Strava Inc., “Strava Releases Year In Sport Report, Showing Benefits of Community and Booming Popularity of International Travel Post-Pandemic”, <https://blog.strava.com/press/yis2022/> (accessed 2023-03-22)
- [2] Shea, K.S. and McMaster, R.B., “Cartographic Generalization in a Digital Environment: When and How to Generalize”, In Proc. Auto-Carto, 9, pp.56-67 (1989)
- [3] Douglas, D.H. and Peucker, T.K., “Algorithms for the reductions of the number of points required to represent a digitised line or its caricature”, The Canadian Cartographer, 10, 2, pp.112-122 (1973)
- [4] Visvalingam, M. and Whyatt, J. D., “Line generalisation by repeated elimination of points”, The Cartographic Journal, 30, 1, pp.46-51 (1993)
- [5] Hersherberger J. and Snoeyink, J., “An $O(n \log n)$ implementation of the douglas-peucker algorithm for line simplification”, In Proc. ACM Symposium on Computational geometry, pp.383-384 (1994)

- [6] Imai, H. and Iri, M., “Polygonal Approximations of a Curve - Formulations and Algorithms”, Machine Intelligence and Pattern Recognition, 6, pp.71-86 (1988)
- [7] Open Source Geospatial Foundation, PostGIS 3.3.1 Manual, pp.379-380 (2022)
- [8] Wang, S., Bao, Z., Culpepper, J. S. and Cong, G., “A Survey on Trajectory Data Management, Analytics, and Learning”, ACM Computing Surveys, 54, 2, Article 39, pp.1-36 (2021)
- [9] Amigo, D., Sánchez Pedroche, D., García, J. and Molina, J. M., “Review and classification of trajectory summarisation algorithms: From compression to segmentation”, International Journal of Distributed Sensor Networks. 17, 10, 15501477211 (2021)
- [10] Zhang, D., Ding, M., Yang, D., Liu, Y., Fan, Ju. and Shen, H. T., “Trajectory simplification: An experimental study and quality analysis”, In Proc. VLDB 11, 9, pp.934-946 (2018)
- [11] Zheng, Y., “Trajectory Data Mining: An Overview”, ACM Transactions on Intelligent Systems and Technology, 6, 3, Article 29, pp.1-41 (2015)
- [12] Saalfeld, A., “Topologically Consistent Line Simplification with the Douglas-Peucker Algorithm”, Cartography and Geographic Information Science, 26, 1, pp.7-18 (1999)
- [13] Lange, R., Farrell, T., Durr, F. and Rothermel, K., “Remote real-time trajectory simplification”, In Proc. IEEE International Conference on Pervasive Computing and Communications, pp.1-10 (2009)
- [14] Lin, X., Jiang, J., Ma, S., Zuo, Y. and Hu, C., “One-pass trajectory simplification using the synchronous Euclidean distance”, The VLDB Journal, 28, pp.897-921 (2019)
- [15] Wei, Z., Xie, X. and Zhang, X., “AIS trajectory simplification algorithm considering ship behaviours”, Ocean Engineering, 216, Article 108086 (2020)
- [16] Long C., Wong R. C. W. and Jagadish H. V., “Direction-preserving trajectory simplification”, In Proc. VLDB, 6, 10, pp.949-96 (2013)
- [17] Fugacci, U., Kerber, M. and Manet, H., “Topology-Preserving Terrain Simplification”, In Proc. ACM International Conference on Advances in Geographic Information Systems (SIGSPATIAL 2020). pp.36-47 (2020)
- [18] Silbernagl, D., Krismer, N., Malfertner, M. and Specht, G., “Optimization of Digital Elevation Models for Routing”, In Proc. 28th GI-Workshop on Foundations of Databases, pp.103-108 (2016)