

Top-k 最適旅行路検索

押野 泰雅

Htoo Htoo

大沢 裕

埼玉大学大学院理工学研究科

1 はじめに

道路網距離に基づく旅行計画路決定のための Top-k 検索方式を提案する。ここで扱う旅行計画は、現在地と最終目的地、及び途中で訪れる M 種類のカテゴリーが与えられたとき、旅行路長 (D) と途中に経由するデータ点の評価値 (V) から算出される総合評価値 (E) が高いものから k 個の旅行路を提案するものである。 E は以下の式で求められる。

$$E = \alpha D + (1 - \alpha)V \quad (1)$$

ここでは、 D 、 E の範囲は $[0, 1]$ であり、高い値の方が好ましいものとする。 α は重要度のパラメータであり、こちらの範囲も $[0, 1]$ である。

本稿では、Top-k 旅行路決定のためのアルゴリズムを提案し、実験的に評価する。

2 関連研究

この研究は Top-k 検索と旅行経路検索に密接に関係する。以下では 2 つの関連項目でまとめる。

複数の尺度に基づき評価関数によって順位付けし、総合評価値の高いものから k 個求める検索を Top-k 検索という。 k は検索時に指定される数である。空間キーワード Top-k 検索のための検索方式 [1] が提案されている。この検索はキーワードの関連性と空間類似性の 2 つの尺度によって評価された Top-k を求める。この研究は研究の早期段階であり空間類似性の評価にユークリッド距離を利用している。

ユーザーが移動するとき、道路網上での距離や旅行時間での最適化を望む。したがって、道路網距離での検索手法 [2] もまた後に提案された。

しかしながら、これらの検索は単純な種類の検索に制限される。我々の知る限り、旅行路検索のような複雑な検索の種類をターゲットとした検索手法の提案はまだ研究されていない。

一方で、いくつかの種類の旅行経路検索が提案されている。trip planing query (TPQ) は [3] で最初に提案された。この手法では、TPQ は、指定されたデータカテゴリーの集合から選択された各データ点を順次訪問することによって、出発点から目的地までの最短経路を見つける。この検索では訪問順序は指定されていない。

データ点カテゴリーの訪問順序の制限がなく、訪問されるデータ点カテゴリーの数が増えると、莫大な処理時間が要求される。

optimal sequenced route (OSR) と呼ばれる同様のアプローチが [4] で提案された。OSR 検索では、訪問順序が一意に指定され、TPQ 検索よりも処理時間が明らかに減少する。

3 提案方式

Top-k sequence route (Top-kSR) 検索は旅行路長と訪問したデータ点の評価値から算出される評価値によって Top-k 旅行路を求める。

Top-k 検索での順位付けは式 (1) に基づいて行われる。検索条件を満たす順序付き経路の最短距離を d_{min} 、検索中の旅行路の距離を d_p とするとき、 D は $D = d_{min}/d_p$ で評価される。一方で、訪問するデータ点の評価値の総最大スコアを v_{max} 、経路内のデータ点の総評価値を v_p とするとき、 V は $V = v_p/v_{max}$ で評価される。したがって E_p は以下の式で表される。

$$E_p = \alpha \frac{d_{min}}{d_p} + (1 - \alpha) \frac{v_p}{v_{max}} \quad (2)$$

順序付き経路が k 番目に大きい総合評価値 E_k を持つとする。 d_{min} が決定し、少なくとも k 個の経路が得られたとき、全ての見つかった経路の総合評価値は式 (2) によって計算される。距離が d_x の順序付き経路について以下の不等式が成り立つ間、順序付き経路の探索をインクリメンタルに続ける必要がある。

$$\alpha \frac{d_{min}}{d_x} + (1 - \alpha) \geq E_k \quad (3)$$

この不等式は、評価値が v_{max} である経路が存在する可能性があるために成り立つ。

もし $1 - \alpha \geq E_k$ ならば、式 (3) は $\alpha \cdot d_{min}/d_x$ から独立して成り立つ。これは全ての順序付き経路が距離とは無関係に Top-k の集合になる可能性があるということの意味する。結果として処理時間が長くなる。これを避けるために、非常に長い経路を排除するための閾値を使う。探索される最長の順序付き経路は以下の条件によって制限される。

$$\Theta_L = \begin{cases} \theta \cdot d_{min} & \text{if } (1 - \alpha) \geq E_k \\ \min(\theta \cdot d_{min}, \frac{\alpha \cdot d_{min}}{E_k + \alpha - 1}) & \text{else} \end{cases}$$

ここで、 θ は事前に定義する閾値である。

Top- k SR 検索は経路距離の昇順でインクリメンタルな順序付き経路検索を必要とし、この検索は高速に行われることが好ましい。このインクリメンタルな検索では、道路網距離の検索より2~3桁高速であるユークリッド距離での順序付き経路検索 [5] を拡張し Top- k SR 検索に適用する。

一連の訪問点が見つけれられた後、実際の経路や道路網距離は A^* アルゴリズムなどで得ることができる。Top- k SR 検索は多くの順序付き経路を検索し、同じ点のペアは Top- k 検索の間に何回も現れる。重複した最短経路探索を回避するために、一度得られた2つの隣接点間の最短経路はハッシュテーブルに記録される。ハッシュテーブルに既に存在する最短経路が再度要求された場合、その経路は直ちに返される。これは処理時間を大幅に向上させる。

4 性能評価

提案したアルゴリズムを評価するための実験を行った。このアルゴリズムは Java 言語で実装し、実験は Intel Core i7 4770(3.4 GHz) で実行した。168km² の範囲で 16,284 ノード、24,914 リンクで構成された実際の道路網を使用した。

データ点集合はいくつかの密度の下で擬似乱数によって道路網のリンク上に生成した。各データ点は図1に示す4種類の確率分布関数に従って、疑似乱数で生成された1~5の整数の評価値を持つ。

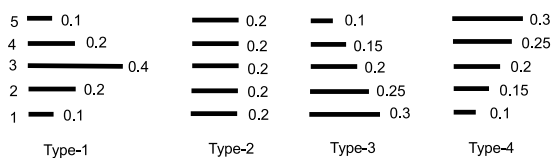


図1: 確率分布関数

図2は type-1 の分布での Top- k SR 検索の処理時間を示している。 M の値は3~5で変化させた。この図が示すように、データ点密度が低いとき処理時間は短い、密度が0.05のとき処理時間は長くなる。

図3は $M = 3$ で処理時間に対する k の値の影響を示している。この図が示すように、処理時間は k の値の増加に従って少しずつ増加しているが、処理時間に対する k の値の影響は小さい。

図4は α の値と処理時間の関係を示している。この図が示すように、 α の値が小さいとき経路の検索範囲が大きくなる。

図5は $M = 3, k = 5$ で確率分布関数の影響を比較し

ている。Type-3 と Type-4 が処理時間の面で両極端となり、他はその間に位置している。

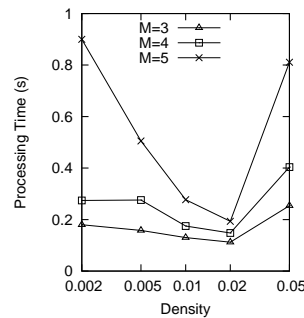


図2: $k=5$, Type-1

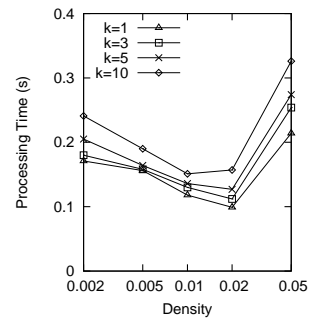


図3: $M=3$, Type-1

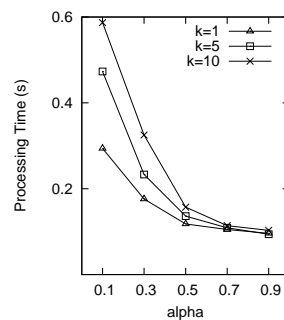


図4: α 値と処理時間の関係

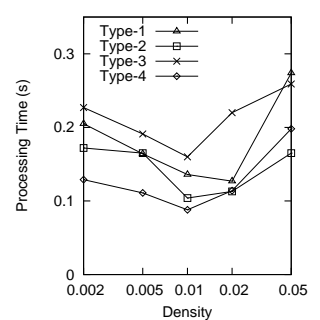


図5: データ分布と処理時間の関係

5 まとめ

本稿では経路距離と訪問したデータ点の評価値の2種類の尺度で評価された k 個の順序付き経路を求める Top- k SR 検索アルゴリズムを提案した。提案したアルゴリズムはユークリッド距離で順序付き経路の候補を検索し、その後、最短経路探索を用いて道路網距離での距離を求める。これらの段階によって、処理時間を低く抑えている。

参考文献

- [1] J. B Rocha-Junior et al., "Top- k spatial keyword queries on road networks", in *EDBT2012*, 2012.
- [2] J. Zhang et al., "Evaluation of spatial keyword queries with partial result support on spatial networks", in *MDM*, 2013.
- [3] F. Li et al., "On trip planning queries in spatial databases", in *Proc.SSTD2005*, 2005, pp. 273–290.
- [4] M. Sharifzadeh et al., "The optimal sequenced route query", Computer Science Department, University of Southern California, Tech. Rep., 2005.
- [5] Y. Ohsawa et al., "Sequenced route query in road network distance based on incremental Euclidean restriction", in *DEXA2012, LNCS7446*, 2012, pp. 484–491.