Vol.2017-DPS-172 No.6
Vol.2017-SPT-26 No.6
Vol.2017-EIP-78 No.6
2017/11/29

キーワードと評価値を付与した道路ネットワークにおける経路提示手法に関する研究

小林 由真[1,a)]　王 家宏[1]　児玉 英一郎[1]　高田 豊雄[1]

概要：近年デジタル地図を用いたサービスが普及しており，店舗検索や目的地までの経路探索に役立てられている．さらにデジタル地図へ情報を付与し，より質の高いサービス提供に役立てようとする動きもみられる．しかし，付与された情報は経路探索に役立てられず，最短経路のみ提示する手法が多い．本研究では，情報を付与した道路ネットワークにおける評価値を用いた経路探索と車の移動に伴って動的に経路を提示する手法の提案を行う．

Answering Top-k Queries on Road Networks with Keywords and Evaluation Values

Yuma Kobayashi[1,a)]　Jiahong Wang[1]　Eiichiro Kodama[1]　Toyoo Takata[1]

Abstract: In recent years, digital map-based services have become widely used in, for examples, searching for shops and routes to some destinations. It is also noted that other kinds of information than the traditional ones for route search have been being added to digital maps to provide higher quality services. However, most of the existing route search services could not effectively and efficiently utilize these kinds of information, but only provide the shortest route-oriented services. This paper presents a new route search method that could utilize the keywords and their evaluation values attached to road nodes in a road network, and provide automobile users with higher quality.

1. 背景

OpenStreetMap や GoogleMap を代表としたデジタル地図を用いたサービスが普及している．このようなサービスは，地点情報の閲覧や店舗検索，目的地までの経路探索に用いられる．

デジタル地図の形式の一つに道路ネットワークデータがある．道路ネットワークデータは, 店舗や曲がり角といった地点の特徴を表すものをノードとみなし，ノード同士をエッジで結んで作られた無向グラフで構成される．無数のノードで構成することで，曲がる地点や速度制限など目的地までの経路案内でユーザへの詳細な指示が可能になる．

表 1 はカリフォルニア州の道路ネットワークデータの一部である．ノードは，ノード ID，緯度，経度で構成される．エッジは，エッジ ID，始点ノード ID，終点ノード ID，始点ノードと終点ノード間の距離で構成される．

表 1　カリフォルニア州のデジタル地図データ

ノード情報：
ノードID, 緯度, 経度

```
0 -121.904167 41.974556
1 -121.902153 41.974766
2 -121.896790 41.988075
3 -121.889603 41.998032
4 -121.886681 42.008739
5 -121.915062 41.970314
6 -121.910088 41.973942
7 -121.916199 41.969482
⋮
```

エッジ情報 : エッジID, 始点ノード, 終点ノード, 距離

```
0 0 1 0.002025
1 0 6 0.005952
2 1 2 0.014350
3 2 3 0.012279
4 3 4 0.011099
5 5 6 0.006157
6 5 7 0.001408
7 5 8 0.012008
⋮
```

近年では，デジタル地図に情報を付与した，より高度なサービスが見られるようになった．例として，セーフティマップ [1] や HowLoud[2] が挙げられる．セーフティマップは，車両から収集した急ブレーキを行った地点やユーザが投稿した危険地点をデジタル地図に付与し，交通事故

[1] 岩手県立大学ソフトウェア情報学研究科
[a)] g231o010@s.iwate-pu.ac.jp

ⓒ 2017 Information Processing Society of Japan

Vol.2017-DPS-172 No.6
Vol.2017-SPT-26 No.6
Vol.2017-EIP-78 No.6
2017/11/29

の起こりやすさを可視化して閲覧できるサービスである．HowLoud は，騒音の大きさを地図上に付与し，閲覧できるサービスである．

また，旅行者のブログや SNS をデジタル地図と連携することで，SNS からの投稿を地図上に付与する試みもある．ユーザ投稿からデジタル地図に情報を付与するサービスとして，Logo[3] などが挙げられる．Logo は，ユーザが投稿した観光スポットを地図上に付与し，閲覧することができる．

本研究では，以上の例で説明したデジタル地図データに付与された情報をキーワードとキーワードの評価値で表せるものとする．情報を付与したデジタル地図も道路ネットワークデータで表すことができる．このような情報を付与した道路ネットワークデータの例を表 2 に示す．各ノードは，ノード ID，緯度，経度に加えて，キーワードと評価値のペアを複数付与することで表される．エッジは，エッジ ID，始点ノード，終点ノード，ノード間の距離で表される．

表 2　情報を付与したカリフォルニア州のデジタル地図データ

ノード情報 : ノードID, 緯度, 経度 + キーワードと評価値の場合

0	-121.9041 41.9745 <“hotel”, 0.7>, …
1	-121.9021 41.9747 <“cafe”, 0.4>, …
2	-121.8967 41.9880 <“restaurant”, 0.6>, …
3	-121.8896 41.9980 <“park” , 0.6>, …
4	-121.8866 42.0087 <“park” , 0.5>, …
5	-121.9150 41.9703 <“café” , 0.3>, …
6	-121.9100 41.9739 <“park” , 0.2>, …
7	-121.9161 41.9694 <“hotel”, 0.8>, …
	⋮

エッジ情報 : エッジID, 始点ノード, 終点ノード, 距離

0	0	1	0.002025
1	0	6	0.005952
2	1	2	0.014350
3	2	3	0.012279
4	3	4	0.011099
5	5	6	0.006157
6	5	7	0.001408
7	5	8	0.012008
		⋮	

店舗や曲がり角をノードと見なし，各ノードにはその場所の特徴を表すキーワードと評価値のペアが複数付与されている．ノード同士を線 (エッジ) で結んで作られたグラフを情報が付与された道路ネットワークデータとして扱う (図 1)．

現状では，情報を付与した道路ネットワークデータを用いた経路探索はいくつか課題が存在しており，３つの例を以下に示す．１つ目は，キーワード“cofe”と“hotel”を共に含む場所を経由する経路を車の移動に伴って連続的に提示したいという要求がある (図 2)．この要求に関しては，関連研究 [4] で実現している．

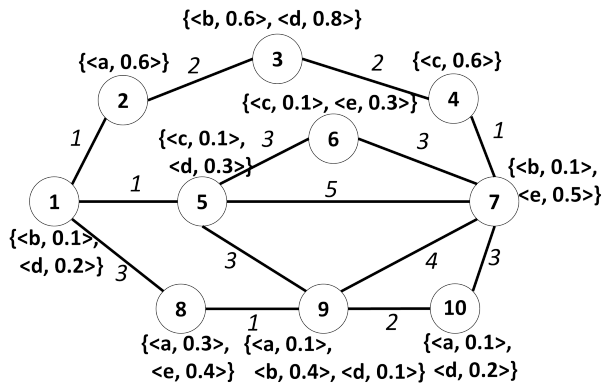


図 1　デジタル地図にキーワードと評価値を付与したシステムモデル

２つ目は，キーワード“cafe”と“hotel”に関係する地点を経路全体に含む経路を車の移動に伴って連続的に提示したいという要求である (図 3)．この課題に関しては，調べた限りいずれの関連研究でも実現されていない．

３つ目は，キーワード“cafe”と“hotel”に関連する地点を経路全体に含み，人気の高い場所を通る経路を優先して車の移動に伴って連続的に提示したいという要求である．

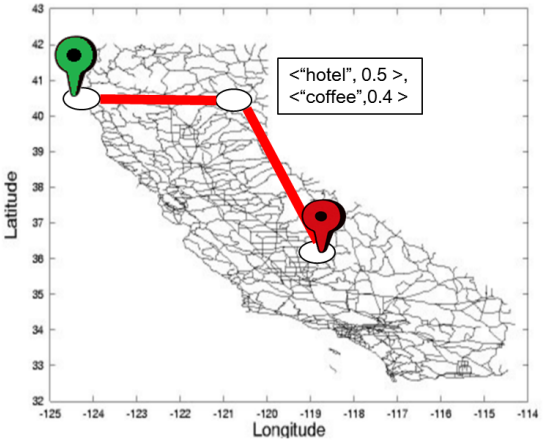


図 2　複数キーワードを持つノードを経由する経路の提示の要求

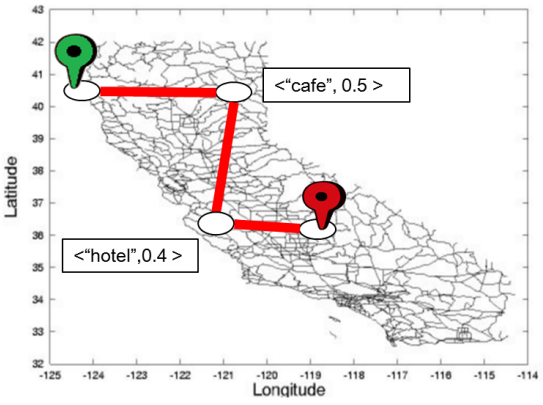


図 3　経路全体にキーワードを含む経路の提示の要求

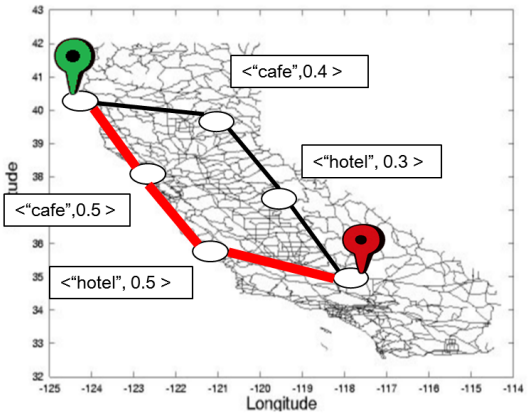


図 4　複数の経路提示の要求

ⓒ 2017 Information Processing Society of Japan

Vol.2017-DPS-172 No.6
Vol.2017-SPT-26 No.6
Vol.2017-EIP-78 No.6
2017/11/29

例として，図 4 に示す，キーワード“cafe”と“hotel”を通る 2 つの経路の場合，“cafe”と“hotel”の評価値が高い下端の経路を優先して表示，次に上端の経路をユーザ提示したい．この課題に関しては，他の関連研究では最短経路のみ表示することを目的としているため，実現されていない．

そこで，本研究では，情報が付与された道路ネットワークにおいて，次の 3 点を満たす複数の経路を車の移動に伴って連続的に提示する手法の提案を目的とする．1 つ目は，ユーザの求める複数のキーワードを同時に含むノードを経由する経路を提示する．2 つ目は，経路全体でユーザの求める複数のキーワードを含む経路を提示する．3 つ目は，複数経路が存在する場合，評価値の高い経路を優先して提示する．

本論文では，第 2 節にて情報が付与された道路ネットワークでの経路探索に関する研究と課題について述べる．第 3 節では，本提案手法である道路ネットワークに付与された情報を活用した経路探索と提示手法について述べる．第 4 節では，ランダムな値で作成したデジタル地図データに対して本提案手法を用いた経路抽出を行い，本提案アルゴリズムの性能に関する考察を行う．最後に，まとめとして結論を述べる．

2. 関連研究

情報が付与された道路ネットワークデータを用いて，目的地までの経路を車の移動に伴って提示し続けることを目的とした研究として，Zheng らの研究 [4] がある．この研究では，道路ネットワークデータの各ノードにキーワードのみ付与したものを用いている (図 5)．この研究では，ユーザの求めるキーワードを同時に含むノードを経由する経路の提示を実現している．手法として，現在地と目的地と同距離にあるノードを起点として，再帰的に中間のノードが持つキーワードと隣接ノードと距離を探索していくことによって，低い探索コストで経由するノードを発見している．また，探索しない区間を計算によって設けることによって，車による動的な移動に対しても検索コストを削減している．しかし，ユーザの求めるキーワードを含むノードを経路全体に含む経路の提示は実現していない．また，用いた道路ネットワークデータには，キーワードに関する評価値が各ノードに付与されていないため，最短経路の提示のみで，キーワードに重きをおいた経路の優先表示ができない．

SNS などから収集される大量の経路情報と検索に用いたキーワードを活用して経路を提示する研究として Wang らの研究 [5] が挙げられる．この研究では，ユーザの求めるキーワードを含むノードを経路全体に含む経路の提示を実現している．しかし，Zheng らの研究同様，用いた道路ネットワークデータにはキーワードに関する評価値が各ノードに付与されていないため，キーワードに重きをおいた経路の優先表示ができない．この研究では，検索キーワードと関連した経路の数が多い経路を提示する．そのため，一般的によくユーザが通る道が提示されやすく，キーワードの関係性や距離が無視される可能性がある．

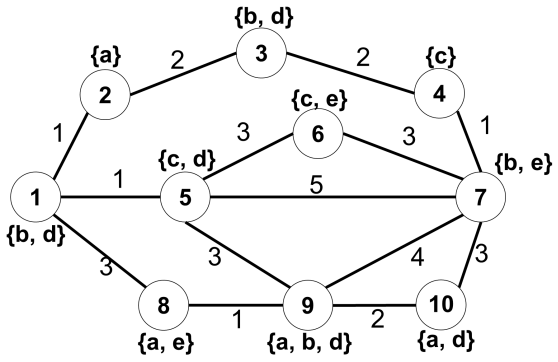


図 5　関連研究 [4] のシステムモデル

道路ネットワークデータに付与されたキーワードと評価値を活用した経路の提示手法は調べた限り存在せず，最短経路の提示を主とした目的の研究がほとんどであった．

3. 情報を付与した道路ネットワークにおける評価値を用いた経路の動的提示手法

本研究では，道路ネットワークに付与されたキーワードと評価値を用いて，ユーザの求めるキーワードの評価値の高いノードを経由する経路を優先して，車の移動に伴って動的に複数の経路を提示する手法を提案する．

3.1 システムモデル

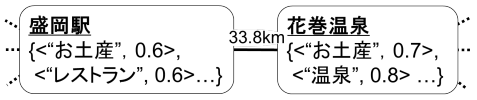
ユーザは車の移動しながらシステムを利用することとし，現在地と目的地，経由したい地点に関する複数のキーワードを入力することで，評価値を考慮した順番で複数の経路を車の移動に伴ってユーザへ動的に提示するシステムを考える．店舗や曲がり角といった地点の特徴を表すものをノードとみなし，ノード同士をエッジで結んで作られた道路ネットワークを用いる．また，各ノードには複数のキーワードと評価値のペアが付与されている．評価値は，“その地点にあるホテルの評価は 0.6 である”のようにノードがそのキーワードで評価された値である．しかし，“その地点にあるホテルの静かさの評価は 0.6 であり，ホテルの広さは 0.4 である”のようにキーワードの細かな指標で評価することは考慮しない．

情報が付与された道路ネットワークデータを G = (V, E) でモデル化する．ノードの集合を V とし，V = {NodeID，Longitude，Latitude，{<Keyword, FeatureValue$_i$ >}$_{i=1,2,\cdots,K_{NodeID}}$} で表す．エッジの集合を E とし，E = {<EdgeID, StartNodeID, EndNodeID, Distance>} で表す．本提案手法では，事前に G を現在地を中心とし，現在地と目的地を半径とした円で範囲を限定

Vol.2017-DPS-172 No.6
Vol.2017-SPT-26 No.6
Vol.2017-EIP-78 No.6
2017/11/29

しているものとする．

例として，盛岡駅と花巻温泉のノードが存在する道路ネットワークデータを図 6 に示す．盛岡駅のノードは，$V_{盛岡駅}$ = {0，39.701164，141.133516，<“お土産”，0.6>，<“レストラン”，0.6>，…} と表される．花巻温泉のノードは，$V_{花巻温泉}$ = {1, 39.451619, 141.06967, <“お土産”, 0.7>，<“温泉”，0.8>，…} と表される．盛岡駅と花巻温泉を結ぶエッジは，e_0 = {0，$V_{盛岡駅}$，$V_{花巻温泉}$，33.8km} と表される．



$V_{盛岡駅}$ = {0, 39.701164, 141.133516,
<“お土産”, 0.6>,<“レストラン”, 0.6> …}
$V_{花巻温泉}$ = {1, 39.451619, 141.06967,
<“お土産”, 0.7>, <”温泉”, 0.8>, …}
e_0 = {0, $V_{盛岡駅}$, $V_{花巻温泉}$, 33.8km}

図 **6**　ノードとエッジの例

3.2　評価値を用いた経路の動的提示アルゴリズム

本提案手法では，道路ネットワークに付与された複数のキーワードと評価値を活用して，ユーザが求めるキーワードを含めた経路を r 個求め，キーワードの評価値を考慮した順番で車の移動に伴ってユーザへ経路を動的に提示する．具体的には，目的地までの経路探索の際に，キーワードを含むノードが目的地までの経路にあるかどうかを随時計算を行い，ユーザの求めるキーワードを多く含むように目的地までの経路を計算する．

車の移動に伴って動的に目的地までの複数の経路を提示するアルゴリズムを図 7 に示す．アルゴリズムは，経由地選出部，経路計算部，経路提示部の 3 部で構成される．処理の流れは，はじめに 2 行目の経由地選出部にて目的地まで経由するノードの組み合わせを複数生成する．その後，3 行目の経路計算部にてノードの組み合わせから最短経路探索アルゴリズムを用いて詳細な経路を複数生成し，4 行目の経路提示部で評価の高い経路から順番にユーザへ提示する．以下，各部の詳しい処理について述べる．

3.2.1　経由地選出部について

経由地選出部では，ユーザの求めるキーワードを含むノードを経由するノードとして選出し，つなげることで簡易的な経路を k 個生成する (r < k)．後述する経路計算部で簡易的な経路から詳細な経路を生成時に，別々の簡易的な経路から同一の経路が生成される可能性が存在する．また経由するノードの組み合わせを列挙しているに過ぎないので，別の道路ネットワーク上にあって辿り着けないノードが含まれる可能性がある．そのため，簡易的な経路から詳細な経路が生成できない場合も存在する．そのため，経由地選出部では詳細な経路を r 個抽出させるために，ユーザへ提示する経路数 r よりも多い k 個の簡易的な経路を用意する．はじめから詳細な経路を求めずに，キーワードを含むノードのみに着目して簡易的な経路を作成する理由は，キーワードを含まないノードに関する探索の処理を省くためである．経由地選出部のアルゴリズムを図 8 に示す．

ユーザの求めるキーワードを含んだノードのみで構成された簡易的な経路を SimpleRoute とし，SimpleRoute= {{ ノード 1，ノード 2，…}，…} と表す．また，計算途中の SimpleRoute のリストを SearchList とし，SearchList= {{ ノード 1，ノード 2，…}，…} と表す．経由地選出部の処理は，3 行目までの処理で，探索の起点となる経路を複数作り上げるために，現在地を中心，現在地から目的地までの距離を半径とした円を探索範囲として計算する．その後，探索範囲内にある，ユーザの求めるキーワードを含んだ道路ネットワーク上の任意のノード v を対象として，経

Algorithm 1 : 道路ネットワークにおける複数キーワードを用いた経路検索アルゴリズム

```
Input.  現在地
        目的地
        道路ネットワーク
        提示する経路数r
        ユーザの求めるキーワードのリストKList
Output. 目的地までの経路r個
1 while 現在地 != 目的地   //目的地に着くまで結果を計算する
    // 探索範囲からKList を含む
    // 経由するノードの組み合わせをk個求める (r < k)
2   SimpleRoute = CalcSimpleRoute(現在地, 目的地, KList, k)

    // SimpleRouteから最短経路アルゴリズムを用いて
    // 詳細な経路をr個生成する
3   DetailRoute = CalcDetailRoute(現在地, 目的地, SimpleRoute, r)

    // 詳細な経路をそれぞれ評価し,
    // 評価の高い順序でユーザへ提示する
4   DisplayRoute(DetailRoute, KList)
```

図 **7**　車の移動に伴って複数の経路を提示するアルゴリズム

CalcSimpleRoute : キーワードを含む目的地までの経路の候補を列挙するアルゴリズム

```
Input.  現在地
        目的地
        道路ネットワーク
        提示する経路数k
        ユーザの求めるキーワードのリスト
Output. SimpleRoute(目的地まで経由するノードの組み合わせ)k個

1 探索途中の経路のリストSearchListを初期化する
2 現在地から目的地までの探索範囲SearchRangeを計算する
3 SearchRange内のユーザの求めるキーワードを含んだ
  すべてのノードvから経路{現在地, v}をSearchListに追加する
4 while 抽出したSimpleRouteの数 < k
5   SearchListから, ユーザの求めるキーワードの種類数,
    キーワードの評価値の高さ, 距離の短さの順で評価し,
    最も評価の高い経路をBaseRouteとし,
    その末尾ノードをBaseNodeとして選出する
6   BaseNodeから目的地までの探索範囲SearchRangeを計算する
7   if 探索範囲内にBaseRouteに
      含まれないキーワードを持つノードがない場合
8       経路{BaseRoute, 目的地}をSimpleRouteとして抽出する
    else
9       探索範囲内にあるBaseRouteに
        含まれないキーワードを持つすべてのノードvから
        経路{BaseRoute, v}をSearchListに追加する
```

図 **8**　簡易的な経路を生成するアルゴリズム

ⓒ 2017 Information Processing Society of Japan

Vol.2017-DPS-172 No.6
Vol.2017-SPT-26 No.6
Vol.2017-EIP-78 No.6
2017/11/29

路 { 現在地，v} を SearchList へ追加する．

その後，5 行目の処理で，SearchList に含まれるユーザの求めるキーワードの種類が多いこと，経路上に含まれるユーザの求めるキーワードの評価値が高いこと，距離が短いことの順で評価が最も高い経路 (BaseRoute) の末尾にあるノード (BaseNode) から探索を進める．評価は，BaseRoute 上のノードと BaseNode から目的地までの距離を直径とした円の範囲内にあるすべてのノードを対象とする．経路上に含まれるユーザの求めるキーワードの評価値の計算は，対象のすべてのノードからユーザの求める各キーワードの最も高い評価値の総和で求める．6 行目の処理では，BaseNode から目的地までの距離を直径とした円の範囲を探索範囲として計算する．

7 行目と 8 行目の処理では，選出された BaseNode を基点とした探索範囲内にキーワードを含むノードがない場合，選出された経路は探索の余地がなく完成された経路であると判断し，SimpleRoute として抽出する．9 行目と 10 行目では，探索の余地があるとして，探索範囲内にある BaseRoute 上のノードに含まれないキーワードを持つすべてのノードを BaseRoute の末尾にそれぞれ追加し，SearchList へ追加して経路を伸ばす．

5 行目以降の処理を繰り返すことで，k 個の経路を抽出する．抽出した経路の数が k 個を満たさなかった場合，最短経路 { 現在地，目的地 } を抽出して経路計算部の処理に移る．

以下，図 9 に示す例を用いて，図 8 に示したアルゴリズム CalcSimpleRoute の処理の流れを説明する．ここでは，抽出する経路数を 2 つ，ユーザの求めるキーワードを “A”，“B” とする．はじめに，始点ノード N_{Start} から終点ノード N_{End} までの距離を直径とした円を探索範囲 SearchRange1 として計算する．次に，SearchRange1 の範囲内にあるユーザの求めるキーワードを含んだ任意のノード v から N_{End} までの距離を直径とした円を探索範囲としてそれぞれ計算する．計算した探索範囲から，ユーザの求めるキーワードの種類が探索範囲内で最も多いノード $v = N_{12}$ を BaseNode と設定する．このとき，N_{Start} から N_{12} までの経路 $\{N_{Start}, N_{12}\}$ が BaseRoute となる．N_{12} を BaseNode として選出後，N_{12} から N_{End} を直径とした円を探索範囲 SearchRange2 として計算する．SearchRange2 の範囲内にある BaseRoute に含まれないキーワード “B” を持つ任意のノードを BaseRoute の末尾に追加し，探索途中の経由地点の組み合わせを格納するリスト SearchList にそれぞれ追加する．SearchRange2 より，SearchList は $\{\{N_{Start}, N_{12}, N_{20}\}, \{N_{Start}, N_{12}, N_{28}\}\}$ となる．

その後，以上の処理と同様に，SearchList からの各組み合わせを対象として末尾のノードからそれぞれ探索範囲を計算し，含まれるキーワードが多く，キーワードの評価が高く，かつ N_{End} までの推定距離が短いノードを BaseNode として選出し，探索を続ける．推定距離は，BaseRoute 上の各ノード間の直線距離を計算した総和で求める．BaseNode から計算した探索範囲内にユーザの求めるキーワードを含むノードがない場合，探索が終了したものとして，BaseRoute の末尾に N_{End} を追加したものを簡易的な経路 SimpleRoute として抽出する．SearchList 内の N_{20} と N_{28} から計算される探索範囲には，ユーザの求めるキーワードを含むノードが存在しないため，$\{N_{Start}, N_{12}, N_{20}, N_{End}\}$ と $\{N_{Start}, N_{12}, N_{28}, N_{End}\}$ をそれぞれ SimpleRoute として抽出し，処理を終了する．

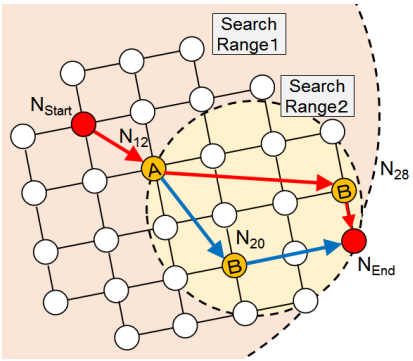


図 9　探索範囲と経路の探索

3.2.2　経路計算部について

経路計算部では，経由地点選出部で抽出された，現在地から目的地までのキーワードを持つ経由ノードのみで構成された簡易的な経路を基に，詳細な経路を生成する．経由地選出部で生成された経路 {{ ノード 1，ノード 2，ノード 3，…}，…} を 2 点間の区間 {{{ ノード 1，ノード 2}，{ ノード 2，ノード 3}}，…} に分解する．その後，各区間ごとに A∗ アルゴリズム [6] を代表とする最短経路探索アルゴリズムを用いて区間の詳細な経路を求め，求めた経路をつなぎ合わせることで，現在地から目的地までの詳細な経路を計算する．その後，計算した複数の詳細な経路から r 個の経路を抽出する．

車の移動に伴う経路探索は，ノードを移動するごとに探索を行う必要がある．そのため，以前の処理で計算した区間を移動の度に計算してしまう恐れがある．この問題を解決するために，2 点間の詳細な経路を保存するデータベースを設け，経路は { 始点ノード，終点ノード，{ 詳細な経路 }} の形式で保存する．経路計算部で既に計算済みである区間を求める際にデータベースにある経路を用いることで，再計算の処理を省く．

3.2.3　経路の提示部について

経路提示部では，経路計算部で抽出した経路をそれぞれ評価し，評価の高い順でユーザへ複数の経路を提示する．経路の評価では，経路上に含まれるユーザの求めるキーワードの種類が多いこと，経路上に含まれるユーザの求めるキーワードの評価値が高いこと，距離が短いことの順で

ⓒ 2017 Information Processing Society of Japan

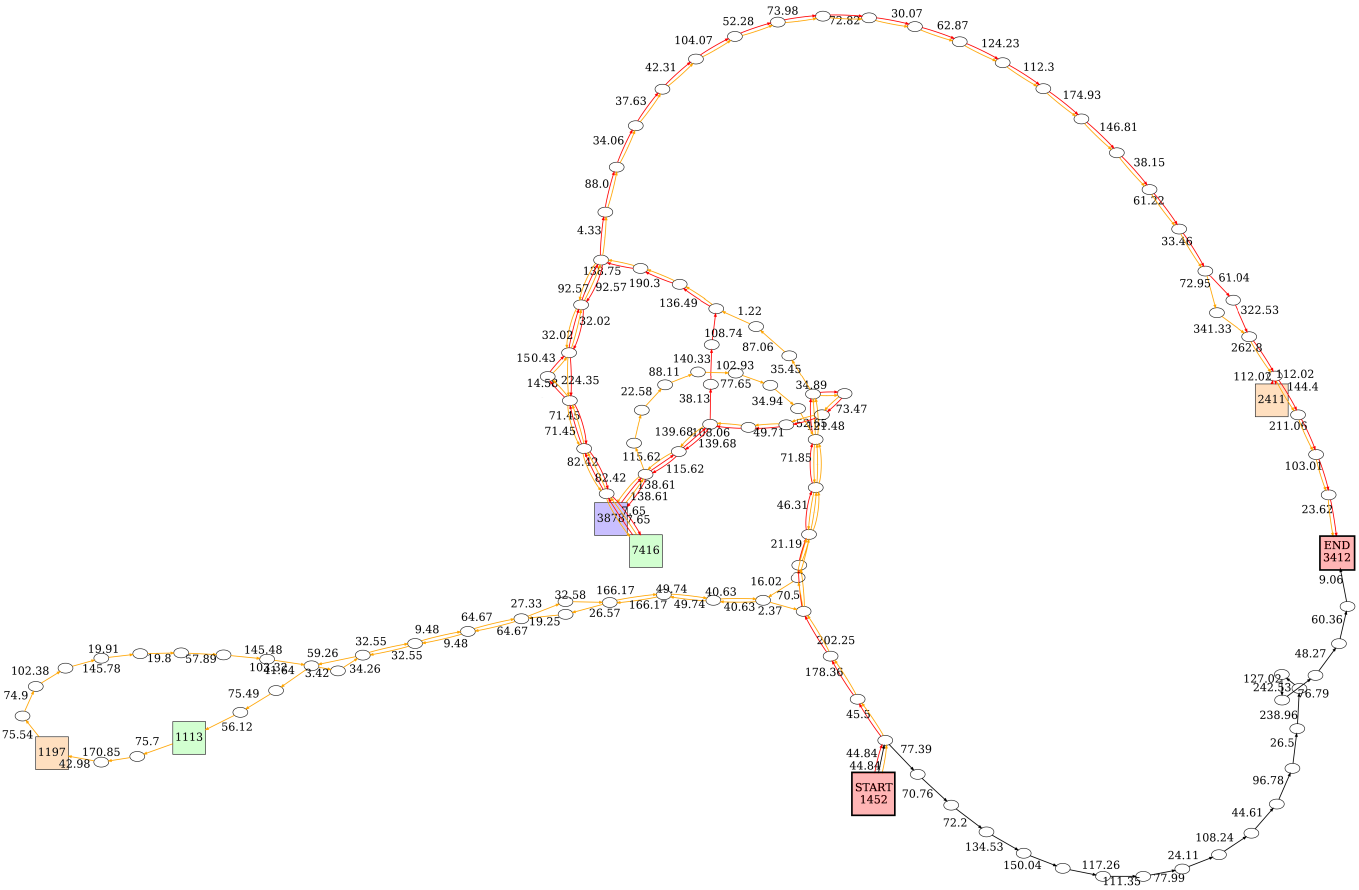


図 10 本提案手法によって抽出した経路

優先して評価する．この処理で，キーワードを多く含んだ上で高い評価値を持つノードを経由する経路が優先され，距離だけを優先した経路は低く評価がされる．

以上の経路計算から経路提示までの処理を車で移動するユーザがノード間を移動する度に行うことで，車に移動に伴って経路を提示する．

4. シミュレーション結果と考察

本節では，ランダムな値で生成したデジタル地図データに対して本提案手法を実行することで，本提案手法の特徴を分析する．デジタルデータ生成に用いた値として，ノード数を 10k，キーワードの種類を 1k，1 ノードあたりに付与されるキーワード数を 3 つ，道路ネットワークの広さは 1 辺 1000km の正方形として，ランダムなデジタル地図データを生成した．生成したデジタル地図データの一部を表 3 に示す．

生成したデジタル地図データに対して，表 4 に示す実行環境にて本提案手法を実行した．実行結果を図 10 と図 11 に示す．実行には“AAA”，“AAB”，“AAC”の 3 種類のキーワードを入力として指定し，抽出した複数の経路を評価の高い順番で提示させた．実行結果として，キーワードを含む経路 2 つとキーワードを含まない最短経路 1 つを抽出させた．

表 3 ランダムな値で生成したノードデータ (左) とエッジデータ (右)

```
1 176.28 733.6 ARR,0.75 BCW,0.43 AGY,0.2
2 260.23 286.36 BJC,0.05 AEU,0.55 BFB,0.59
3 364.71 694.44 AHG,0.7 BES,0.62 BKF,0.09
4 481.87 138.23 ALV,0.35 ANZ,0.44 AWP,0.29
5 298.14 811.76 AZK,1.0 AOY,0.88 AZM,0.7
6 259.78 473.26 BHS,0.26 ANM,0.91 AGG,0.24
7 120.96 349.57 ASZ,0.05 AZV,0.71 AAC,0.3
8 944.69 537.24 AAD,0.44 APC,0.62 BIT,0.26
9 294.39 570.46 BFI,0.05 AWR,0.25 AML,0.66
10 987.24 292.57 ASQ,0.69 BDW,0.48 BLU,0.99
⋮
```

```
1 1 5919 38.98
2 1 4292 5.17
3 1 6237 11.73
4 2 8268 11.59
5 2 646 67.06
6 2 9415 17.69
7 3 6387 25.74
8 3 6324 29.96
9 3 4185 23.77
10 4 8017 68.34
⋮
```

表 4 実行環境

開発言語	Python3
CPU	Intel Core i5-440 3.10 GHz
OS	Arch Linux
メモリ	4GB

実行結果より，キーワード“AAA”を含む N_{1197} と N_{2411}，キーワード“AAB”を含む N_{1113} と N_{7416}，キーワード“AAC”を含む N_{3878} をそれぞれ経由する経路が抽出されていることがわかる．また，現在地から目的地までキーワードを持つノードを経由した上で最短の経路を抽出されていることが確認できた．

以上の実行結果を踏まえて，本提案アルゴリズムの性能に関する考察を行う．実行結果から，キーワード分布が偏っていない道路ネットワークでは期待された結果が返っ

Vol.2017-DPS-172 No.6
Vol.2017-SPT-26 No.6
Vol.2017-EIP-78 No.6
2017/11/29

['1452', '2811', '7740', '82', '3582', '8139', '6212', '3266', '444', '9945', '8837', '2886', '348', '6666', '2305', '2347', '433', '3878', '433', '2347', '2305', '7326', '2776', '779', '6223', '9716', '546', '7995', '4945', '3299', '6574', '3843', '7416', '3843', '6574', '3299', '8549', '4945', '7995', '546', '5493', '3002', '1680', '5428', '4658', '6743', '4430', '6851', '1282', '9199', '1338', '48', '7478', '4768', '5988', '8447', '4757', '6986', '1105', '7719', '61', '2411', '61', '6955', '6789', '8015', '3412']
EvaluationValue: 1.97, Keyword : ['AAA', 'AAB', 'AAC'] , Distance : 6269.22

['1452', '2811', '3907', '7740', '82', '3582', '4724', '8987', '3366', '3381', '7499', '2168', '4145', '130', '8009', '765', '6972', '1113', '5254', '3176', '1197', '3176', '5254', '5284', '6293', '4796', '7972', '8246', '4145', '2168', '7499', '9768', '3381', '3366', '8987', '9715', '6770', '6212', '3266', '444', '9945', '8837', '2886', '348', '6666', '2305', '2347', '433', '3878', '433', '2347', '2305', '7326', '2776', '779', '6223', '9716', '546', '7995', '4945', '3299', '6574', '3843', '7416', '3843', '6574', '3299', '8549', '4945', '7995', '546', '5493', '3002', '1680', '5428', '4658', '6743', '4430', '6851', '1282', '9199', '1338', '48', '7478', '4768', '5988', '8447', '4757', '6986', '5311', '7719', '61', '6955', '6789', '8015', '3412']
EvaluationValue : 1.73, Keyword : ['AAA', 'AAB', 'AAC'] , Distance : 8023.34

['1452', '2811', '6643', '2432', '777', '3267', '6419', '3428', '9090', '4767', '8705', '7363', '8324', '7640', '2044', '5003', '9306', '4086', '983', '3316', '8620', '3412']
EvaluationValue : 0.0, Keyword : [] , Distance : 1959.58

図 11 本提案手法によって抽出した経路の詳細

てくると考えられる．しかし，ユーザの求めるキーワード数が多くなるほど，経路計算部で探索する組み合わせが爆発的に増えるため，実行時間が増える可能性が存在する．その場合，車の移動速度やノード間の距離によっては探索の実行時間が車の移動にかかる時間より大きくなってしまう可能性が存在する．

また，キーワードを持つノードの分布が大きく偏っている場合，ユーザの求めるキーワードを含まない経路が生成される恐れがある．図 12 を例とする．この例では，N_{14} と N_{27} を経由地点とした経路を抽出することができれば理想的である．しかし，N_{14} を基点とした SearchRange1 からは N_{27} を含めることができない．同様に N_{27} を基点とした SearchRange2 からは N_{14} を含めることができない．このため，N_{14} と N_{27} の片方のみで構成された経由地点の組み合わせがそれぞれ生成されてしまう．これは，探索が進むごとに探索範囲が限定され，キーワードを含むノードが探索範囲から外れる可能性が高くなるためである．

しかし，このような地図からユーザの求めるすべてのキーワードを含む経路を生成する場合，非常に遠回りな経路が生成されてしまう可能性が高いため，移動距離の長い経路が生成されてしまう恐れがある．複数の経路の提示を前提とする本研究では，キーワードの有無を考慮しない最短経路，キーワードと距離のバランスを考慮した本提案手法，距離の長さを無視してユーザの求めるすべてのキーワードを含む経路をそれぞれ提示し，複数経路からユーザが判断して経路を選択することも改善案として考えられる．

次に目的地までの経路を生成できない場合について，2 つの例を用いて述べる．1 つ目は，ユーザの求めるキーワードを持つノードが探索範囲外にある場合である (図 13)．図 13 では，キーワード “A” を持つノード N_{30} が探索範囲外にあるため，経由するノードとして検出できない．この例では，探索範囲を広げることで解決できる可能性がある．しかし，探索範囲を広げることでより多くのノードが探索の対象となり，計算コストが増大する恐れがある．また，

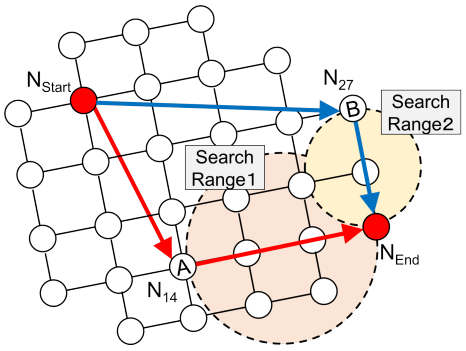


図 12 キーワードを含んだノードをつなぐことができない例

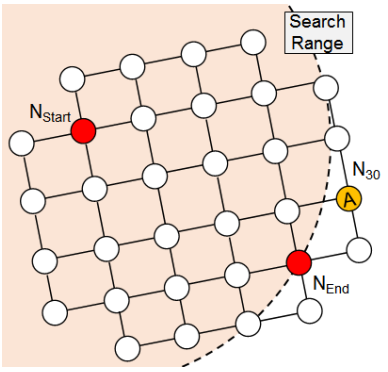


図 13 ユーザの求めるキーワードを持つノードが探索範囲外にある例

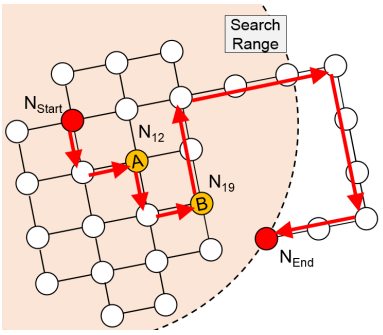


図 14 目的地までの経路が探索範囲外にある場合

探索範囲をどんな広げたとしても，目的地までの経路や探索範囲外のキーワードを含むノードを見逃してしまう可能性が存在するため，探索範囲については検討する必要がある．

2 つ目は，始点ノードから終点までの経路が探索範囲外にある場合である (図 14)．この場合，簡易的な経路から詳細な経路を生成する経路計算部にて，探索範囲外のノードも対象とするように修正することで，最短経路探索アルゴリズムで経路を生成することができる．

5. まとめ

情報が付与されたデジタル地図が普及している．しかし，付与された情報は経路探索に活用されておらず，最短経路のみの提示を行う手法が多数であった．そこで情報を付与した道路ネットワークにおける評価値を用いた経路探

ⓒ 2017 Information Processing Society of Japan

索と経路提示の手法を提案した．ランダムに生成したデジタルデータに対して本提案手法を実行することで，本提案手法の特徴を分析した．結果として，ユーザの求めるキーワードを持つノードが極端に配置されてない場合に，キーワードを多く含み，評価値の高い複数の経路を評価値の高い順番で提示することができた．

参考文献

[1] Honda : セーフティマップ，入手先〈https://safetymap.jp/〉(参照 2017-07-31).
[2] HowLoud : HowLoud, 入手先〈http://www.gizmodo.jp/2015/07/post_17608.html〉(参照 2017-07-31).
[3] Logo : Logo, 入手先〈http://van-app.com/logo〉(参照 2017-07-31).
[4] Bolong Zheng, Kai Zheng, Xiaokui Xiao, Han Su, Hongzhi Yin, Xiaofang Zhou, Guohui Li : Keyword-Aware Continuous kNN Query on Road Networks, IEEE International Conference on Data Engineering, pp.871-882, (ICDE-2016).
[5] Sheng Wang, Zhifeng Bao, J. Shane Culpepper , Timos Sellis, Mark Sanderson, Xiaolin Qin: Answering Top-k Exemplar Trajectory Queries, IEEE International Conference on Data Engineering, pp.597-608, (ICDE-2017).
[6] Peter E. Hart, Nils J. Nilsson, Bertram Raphael : A Formal Basis for the Heuristic Determination of Minimum Cost Paths, IEEE Transactions on Systems Science and Cybernetics, Vol.4, pp.100-107, (1968).

ⓒ 2017 Information Processing Society of Japan