

# 空間索引木を用いた範囲問合せ手法

蒲原 智也<sup>1</sup> 上島 紳一<sup>1,a)</sup>

受付日 2011年9月20日, 採録日 2012年1月3日

**概要:** 近年, 地理情報システムや道路網応用の研究分野で, 最短経路問題 (SPP), 範囲問合せ (DRQ),  $k$  最近傍探索問題 ( $k$ NN) などに有効な空間索引の研究が注目を集めている. 我々の先行研究で, ネットワーク上の距離を基準に, 道路網データから領域の分割と統合を繰り返して, ボトムアップに構成する空間索引木を提案した. さらに, この空間索引木を用いて最短経路問題を効率的に実行できることを確認した. 本稿では, 問合せ点を指定する範囲問合せについて議論する. SPP が始点と終点の 1 対 1 の計算になるのに対して, DRQ は, 問合せ点とオブジェクト集合の 1 対多の計算が必要となる. さらに, 先行研究の距離指標のみを用いて問合せ処理を実行すると, オブジェクトが存在すると推定される領域が冗長に見積もられ, 空間索引木の刈込みが有効に行えない. これは, 距離指標が, 母点領域の形状や位置関係を十分に表現できないことに起因する. また, DRQ は, グラフ上の問合せ点とオブジェクト集合を用いた問合せであるため, グラフとオブジェクトとの関係を利用しなければ, 冗長な領域を計算する必要がある. これらの問題の解決のために, 空間の拡がりを示す指標として MBR (最小外接長方形) を導入し, オブジェクトに関する指標を新たに追加する. MBR は, 空間索引木の構成時に付加し, オブジェクト指標については, 空間索引木構成後に, 様々なオブジェクト種別ごとに, 同一の空間索引木に組み込む. これら 2 つの指標と距離指標の 3 つの指標を用いた再帰的問合せ手順を新たに提案し, その効果について議論する. 提案手法の有効性について確認するため, 国土地理院発行の数値地図を利用して, 新規追加の指標のサイズと組み込み処理時間, 距離指標のみを用いたナイーブな手順と提案手法の刈込みの比較, また, 参照点埋込み法と提案手法の距離見積りの精度の比較, ならびに, DRQ において Kriegel らの手法を発展させた階層型参照点埋込み法と提案アルゴリズムの展開数と実行時間の比較を行う.

キーワード: 空間索引木, グラフ分割, 距離索引, MBR, 範囲問合せ

## Distance Range Queries in Spatial Index Tree

TOMOYA KAMBARA<sup>1</sup> SHINICHI UESHIMA<sup>1,a)</sup>

Received: September 20, 2011, Accepted: January 3, 2012

**Abstract:** Recently much attention has been focused on spatial query processing for road networks. In the previous work, the authors have proposed spatial index tree that can be constructed in a bottom-up fashion by generating Network Voronoi Diagram from a given base graph (=road network) using network distance. The tree possesses distance indices for each subregion that work efficiently to prune spatial index tree in processing one-to-one distance-based queries, including Shortest Path Problems (SPP). In this paper, the authors discuss spatial query processing for Distance Range Queries (DRQ). In SPP, distance index is used to extract heuristic path using one-to-one distance computation, while one-to-many distance computation is required for a given query point and target object set for DRQ. Only by use of the distance index, redundant search regions may be extracted from the tree, since distance estimation errors are accumulated regarding the size of generator regions that exist between query point and target objects. This is due to the fact that distance index is not sufficient for DRQ in pruning redundant subregions in the spatial index tree. Further, in DRQ, discussion on spatial relationships of query circles and generator regions become necessary. To cope with these problems, two new indices, namely Minimum Bounding Rectangles (MBR), as spatial extent index, and object index, are introduced in the spatial index tree in the paper. These spatial indexes can be built in the spatial index tree in a bottom up fashion, with preserving inclusion relationships among both generator regions and MBR's, respectively. Also, object indices are utilized to prune redundant subregions in DRQ. Then, using these 3 indices, a new recursive algorithm for DRQ is provided. The algorithm is based on the incremental identification of target subtrees, which corresponds to tree pruning. Numerical evaluations for real map data issued by Geographical Survey Institute, are provided to show effectiveness of the proposed approach, including newly added index size and processing time, and accuracies of distance estimation of our previous work and the proposed framework. Also the proposed algorithm is compared with the extended hierarchical embedding method proposed by Kriegel et al in DRQ.

**Keywords:** spatial index tree, graph partitioning, distance index, MBR, Distance Range Queries

## 1. はじめに

近年、地理情報システムや道路網応用などの空間データベースの研究分野で、最短経路問題 (SPP), 範囲問合せ (DRQ),  $k$  最近傍探索問題 ( $k$ NN) などに有効な空間索引の研究が注目を集めている [1], [2], [3], [4], [5], [6], [7].

我々の先行研究 [8] で、道路網をグラフと見なして、グラフをネットワークボロノイ分割し、分割で得られた各母点領域を統合していくことで、木をボトムアップに階層化した空間索引木を提案した。この空間索引木の生成では、グラフ分割の基準となる点をランダムに選択し、その点を中心にグラフを分割することで、ノード分布の疎密によらない平衡木を生成している。各母点領域は、領域内・隣接領域間の経路と距離に関する指標 (距離指標) を索引として持ち、木を階層化することで遠方領域に対する処理の効率化を図っている。

本稿では、空間索引木の DRQ への適用方法について議論する。SPP においては、2.3.4 項に概略を示すように、まず、母点領域の隣接関係を用いて目的地点から始点に向かって逆向きに経路と距離を見積もることで、木の刈込み方向を得ることができ、始点から目的地点に向かって境界点間経路をつなぐことで最短経路を求めた。

しかし、SPP がグラフ上の始点と目的地点の 2 点間の関係に基づく探索であるのに対して、DRQ は、問合せ点と全方位に存在するオブジェクトとの関係を調べることが必

要である。先行研究の母点領域の隣接関係と距離指標のみを用いた粗査を実行すると、問合せ点から遠方になるにつれて冗長な領域を抽出することになる。これは、母点領域の距離指標が、母点領域の形状や位置関係を十分に表現できないことに起因する。さらに、DRQ はグラフ上の問合せ点とオブジェクト集合を用いた問合せであるため、グラフとオブジェクトとの関係を用いなければ、冗長な領域を抽出することになり非効率的である。

そこで、本稿では、母点領域の拡がり指標として MBR (最小外接長方形) を空間索引木に組み込み、同時にオブジェクト指標も導入して、(i) 距離索引、(ii) 拡がり指標、(iii) オブジェクト指標を用いた再帰的問合せ手順を新たに提案し、その効果について検証する。2.2 節の関連研究で述べるように、 $k$ NN と DRQ などに対しては、空間分割による方法と参照点埋込み法の 2 種類のアプローチが代表的である。提案手法は前者であり、空間領域を刈り込んでからオブジェクトを操作する方法となっている。

2 章では、ネットワークと DRQ の定式化とネットワークボロノイ図に関する定義を行い、提案空間索引木について概説し、関連研究と比較する。3 章では距離指標のみをそのまま DRQ に適用した場合の問題点について議論する。空間索引木に対する拡がりの指標については 4 章で、探索対象となるオブジェクトの取扱いについては 5 章で議論する。6 章では、DRQ に対して MBR を組み込んだ再帰的範囲問合せアルゴリズムを与え、7 章では、国土院発行の数値地図を利用して、新指標 (ii), (iii) のサイズと処理量、参照点埋込み法との距離見積りの精度について比較する。さらに、先行研究の距離指標のみを用いた場合との比

<sup>1</sup> 関西大学大学院総合情報学研究科  
Graduate School of Infomatics, Kansai University,  
Takatsuki, Osaka 569-1095, Japan

a) ueshima@res.kutc.kansai-u.ac.jp

較, ならびに DRQ における Kriegel らの手法 [1] を発展させた階層型参照点埋込み法と提案手法を比較して提案手法の有効性について確認する.

## 2. 準備

### 2.1 定義

#### [ネットワーク]

道路網において, 交差点をノード, 交差点間の道路片をエッジと見なした無向グラフ  $G = (N, E)$  ( $N$ : ノード集合,  $E \subseteq N \times N$ : エッジ集合) を考える. さらに, 隣接する2つのノード  $P_i, P_j$  を結ぶエッジは, 非負値の重み  $w(P_i, P_j)$  を持つものとする.  $P_i, P_j \in N$  間のネットワーク上の距離は,  $P_i, P_j$  間の最短経路長とし,  $d_{net}(P_i, P_j)$  で表す.  $P_i, P_j$  が隣接する場合は,  $d_{net}(P_i, P_j) = w(P_i, P_j)$  である. また, 各ノード  $P$  は2次元平面での座標  $(P.x, P.y)$  を持ち, さらにエッジ上にも任意の点をとることができるものとする.

道路網を示すグラフ  $G$  上のオブジェクト集合を  $D$  とおく. オブジェクト  $o \in D$  はたとえば, 道路網上のガソリンスタンドやコンビニエンスの店舗などを指す.

#### [範囲問合せ (Distance Range Query)]

$G$  上に問合せ点  $q$  と問合せ半径  $r \in \mathbf{R}^+$  (ここで  $\mathbf{R}^+$  は正の実数を示す) が与えられたとき, ネットワーク上の距離  $r$  以内に存在するオブジェクト集合を返す問合せである. この問合せを  $DRQ(q, r)$  と記す. すなわち,

$$DRQ(q, r) = \{o \in D \mid d_{net}(q, o) \leq r\} \quad (1)$$

次に本稿で利用するグラフ分割法であるネットワークポロノイ図について定義する.

#### [ネットワークポロノイ図 (Network Voronoi Diagram)]

$N$  の部分集合  $\{P_1, P_2, \dots, P_n\}$  を母点集合とする.  $I_n = \{1, \dots, n\}$  とするとき,  $G$  上の任意の点  $p$  に対して,  $d_{net}(p, P_i)$  と  $d_{net}(p, P_j)$  を比較した場合,  $d_{net}(p, P_i)$  が小さければ,  $p$  に対して  $P_i$  は  $P_j$  より支配的 [9] であるという. このような支配的な点の集合を  $Dom(P_i, P_j)$  と書く. すなわち,

$$Dom(P_i, P_j) = \left\{ p \mid p \in \bigcup_{e_i \in E} e_i, d_{net}(p, P_i) \leq d_{net}(p, P_j), i \neq j \right\} \quad (2)$$

また,

$$b(P_i, P_j) = \left\{ p \mid p \in \bigcup_{e_i \in E} e_i, d_{net}(p, P_i) = d_{net}(p, P_j), i \neq j \right\} \quad (3)$$

は  $G$  上にあり  $P_i, P_j$  までの距離が等しい点 (境界点と呼ぶ) の集合である. このとき, 母点  $P_i$  のネットワークポ

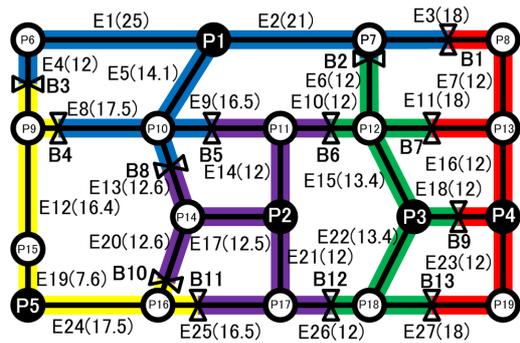


図 1 母点集合  $P_1, \dots, P_5$  に対するネットワークポロノイ図  
Fig. 1 Network Voronoi diagram for generators  $P_1, \dots, P_5$ .

ロノイ領域 ( $P_i$  の母点領域) は,

$$V(P_i) = \bigcap_{j \in I_n \setminus \{i\}} Dom(P_i, P_j) \quad (4)$$

となる.  $\mathbf{V} = \{V(P_1), \dots, V(P_n)\}$  をネットワークポロノイ図 (以後, NV 図) という.

ポロノイ図がユークリッド距離を用いて平面を分割した図であるのに対して, ネットワークポロノイ図は最短経路の距離を用いてネットワークを分割した図である. ノード NV 図はノード集合を所属する母点ごとに分割した図である. またエッジ NV 図はエッジ上の任意の点で分割を許し所属する母点ごとに分割した図であり, エッジの部分の集合から成り立っている [9]. ノード NV 分割は並列ダイクストラ法 [10] によって生成される. エッジ NV 分割は, エッジの持つ両端のノードの所属する母点を調べることで生成できる.

#### <母点領域 $V(P_i)$ >

式 (4) で定義した母点領域  $V(P_i)$  は, 母点  $P_i$  とその  $P_i$  に所属するノードとエッジを集めてできる領域であり, 以下の5つからなる.

- (1) 母点
- (2) 母点に所属するノード
- (3) 母点に所属する境界点を持たないエッジ
- (4) 母点領域の端を示す境界点
- (5) 境界点までの部分エッジ

#### 例:

図 1 は, 母点集合  $\{P_1, \dots, P_5\}$  に対して, 他のノードおよびエッジの所属する母点領域を示した図である. 図中での境界点の位置を  $\boxtimes$  で示している.

母点  $P_2$  に対する母点領域は, 図 1 の紫色の領域であり, (1)  $P_2$ , (2)  $\{P_{11}, P_{14}, P_{17}\}$ , (3)  $\{E_{14}, E_{17}, E_{21}\}$ , (4)  $\{B_5, B_6, B_8, B_{10}, B_{11}, B_{12}\}$ , (5)  $\{P_{11} - B_5, P_{11} - B_6, P_{14} - B_8, P_{14} - B_{10}, P_{17} - B_{11}, P_{17} - B_{12}\}$  から構成される.

表 1 関連研究比較

Table 1 Comparison of the proposed approach with related works.

提案者	手法	階層化	オブジェクト関連付け	対象問合せ
Kolahdouzan [6]	NV 分割・刈込み	×	構造生成時	$k$ NN
Hu [3]	distance signature	×	構造生成時	DRQ& $k$ NN
Kriegel [1]	参照点埋込み	○	問合せ実行時	DRQ& $k$ NN
Cheema [7]	矩形領域の刈込み	×	構造生成時	逆 $k$ NN
提案手法	NV 分割・刈込み	○	構造生成後	SPP&DRQ

2.2 関連研究

$k$ NN と DRQ などに対する問合せ処理には、対象空間を分割して領域を刈り込む方法と参照点（ランドマーク）を埋め込む方法の 2 種類のアプローチがある。

[基本的な考え方]

Papadias ら [2] は、空間ネットワークデータベース上での基本となる問合せ (DRQ,  $k$ NN, 最近点対探索, 距離結合) に対して、ユークリッド距離とネットワーク距離を用いた取扱いの違いを明らかにし、問合せの種類ごとにユークリッド空間を用いた制限を組み合わせることで直感的な問合せ方法を提案している。

[空間分割を用いた領域刈込み]

Kolahdouzan ら [6] は、NV 図による領域分割を用いて、 $k$ NN の効率的な計算法を提案している。同手法では、索引化の対象の位置を基準に NV 図を生成している。このため索引の対象の種類ごとにグラフを生成する必要があり、索引化の対象ごとのグラフは互いに異なる。また、階層化を適用する構造となっていないため、1 層のみの NV 分割では、隣接する領域の関係のみを保持することになり遠方領域の問合せの計算量が増大する。このため、 $k$ NN 以外の道路網応用へ容易に適用できない。

Cheema ら [7] は、オブジェクトと問合せ点がそれぞれ矩形領域内にあると仮定して、矩形の頂点座標間の関係を利用して半空間 (half-space) と呼ばれる領域を生成し、遠方にある候補となるオブジェクトを持つ矩形領域を刈り込むことで逆  $k$ NN に適用している。

[オブジェクトへの距離の埋込み]

上記に対して、Hu ら [3] は、事前にすべてのノードから各オブジェクトまでの距離を不均等なカテゴリ (distance signature) に分類し、実際にノードからオブジェクトまでの距離を求める場合の効率化を図るとともに索引の軽量化を行っている。また、Kriegel ら [4] は、参照点というオブジェクトまでの距離を正確に求めるノードをグラフに埋め込み、すべての索引化の対象オブジェクトからの距離をあらかじめ計算して、ベクトル空間に写像することで高速に DRQ と  $k$ NN を実行している。しかし、道路網が大きくなり、参照点の数が増えると、ベクトル空間の次元が上がり、検索の計算速度が落ちる。このため、Kriegel ら [1] では、重要な参照点を抽出し、それらのみでベクトル空間を低次

元化して、高速化する手法を提案している。

これによって、参照点までの距離を基準にオブジェクトまでの距離を見積もり、フィルタリングと精選の 2 段階探索により探索を高速化している。しかし重要な参照点のみを残し階層化すると、見積り時に誤差が増加するという問題点もある。同様に、Hjaltason ら [5] は、対象を距離空間に一般化して Lipschitz 埋込みと多次元ベクトルの低次元化の観点から、類似探索についてまとめている。

参照点埋込み法は、DRQ や  $k$ NN において、距離指標のみを用いてオブジェクトを直接取り出す方法であるが、領域を分割する手法は、索引木からまず領域を抽出し、その中のオブジェクトを確認する手順となっている。表 1 に提案手法と関連研究の比較を示す。

[提案手法]

これらの研究と比較して、提案手法は、空間をボロノイ分割する第 1 の方法にあたり、以下のような特徴を持つ。提案手法は、(i) 距離指標、拡がり指標、オブジェクト指標の 3 者を用いた、領域を効率的に刈り込む再帰的なアルゴリズムで、(ii) 索引化の対象となるオブジェクトは、種類ごとにグラフを生成する必要がない。また、(iii) 領域の包含関係に関する処理は、距離指標をもとに行う。このため (iv) 階層に沿った領域の統合・分割は、リスト構造の統合・分割に帰着できる。

提案手法と関連研究を比較した場合、 $D$  が多い場合は、各オブジェクトより距離を算出する必要のある Kriegel らや Hu らの手法では索引の大きさが大きくなり、提案手法や Kolahdouzan らの手法では、索引の大きさは変わらない。また探索範囲が遠い場合は、Kriegel らや Hu らの研究と提案手法では必要とするノード数がほぼ一定であるが、Kolahdouzan らでは必要となるノード数が多くなるなどの手法による差異がある。

2.3 先行研究にて提案した空間索引木とその利用法

2.3.1 ランダムな母点選択による領域分割と階層化

[確率的母点選択]

ノード集合  $N$  よりランダムに母点集合を選択し、グラフ  $G$  をノード NV 分割とエッジ NV 分割により分割し母点領域を得る。ランダムに母点選択することにより、 $G$  上にあるノード分布の疎密に依存せず、各母点に対する所属

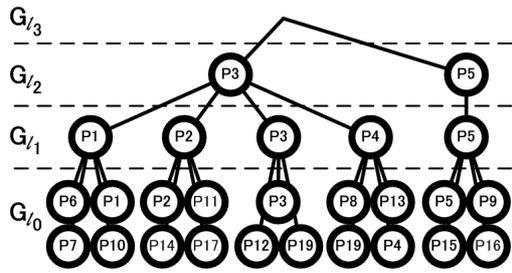


図 2 母点領域の包含関係を示す木構造表現

Fig. 2 Tree representation of inclusion relations among generator regions.

ノード数を一定に構成できる [8].

[階層化]

さらに、上記の分割により得られた母点領域をノードとして扱い、隣接する母点領域間を最短経路で結んだネットワークを作る。このネットワークに対しても同様に母点をランダムに選択し上位層でのノード NV 分割を行う。この手順を繰り返すことにより、図 2 に示すノードを表す領域の包含関係を持つ空間索引木を構成できる。

このように、空間索引木は与えられた道路網  $G$  に対して、ネットワーク上の距離に基づきノード NV 分割とエッジ NV 分割を用いて  $G$  を母点領域に分割し、さらに領域を統合してボトムアップに木構造を組み上げている。このとき上位層のノードは、すぐ下の層では母点領域となっている。

### 2.3.2 空間索引木の距離指標

第  $i$  階層  $l_i$  での母点領域  $P$  に対して、 $P$  に所属するノードとエッジのみで構成した  $P$  の持つ 2 つの境界点  $B_{l_i}^n, B_{l_i}^m$  を結ぶ最短経路を境界点間経路という。  $P$  の持つすべての境界点の組合せに対する境界点間経路の長さの中で最短値、最長値を  $P$  の距離指標として  $\delta_{l_i}^P, \Delta_{l_i}^P$  と書く。つまり

$$\Delta_{l_i}^P = \max_{n,m} d_{net}(B_{l_i}^n, B_{l_i}^m) \quad (n \neq m) \quad (5)$$

$$\delta_{l_i}^P = \min_{n,m} d_{net}(B_{l_i}^n, B_{l_i}^m) \quad (n \neq m) \quad (6)$$

と書ける。ただし、 $\delta_{l_i}^P, \Delta_{l_i}^P$  は領域  $P$  を渡る指標（最小距離、最大距離）として考えているため、同じ所属母点の組合せを持つ境界点間の経路が、より大きい値、小さい値を持っていても、それは距離指標としない。

また、母点から境界点までの最短経路と最短経路長も距離指標の一部として持たせる。

例：

図 3 は、図 1 の母点領域  $P2$  に着目した図である。同図で、 $B5-B11$  を結ぶ境界点間経路 ( $B5-P11-P2-P17-B11$ ) は  $P2$  内で最長であり、 $\Delta_{l_1}^{P2}$  となる。同様に、 $B8-B10$  を結ぶ境界点間経路 ( $B8-P14-B10$ ) が最小であり、 $\delta_{l_1}^{P2}$  となる。

仮に、 $P2-P14$  をつなぐ経路が長いと仮定すると、 $B10-P14-P2-P17-B11$  が  $B5-B11$  より長くて、

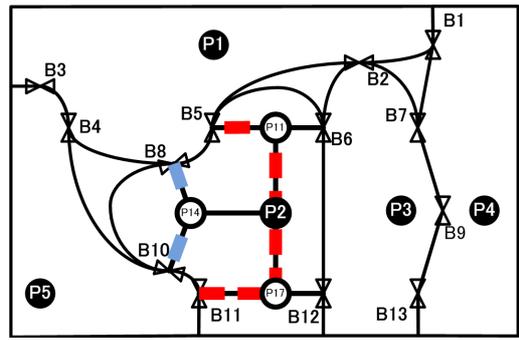


図 3  $P1, \dots, P5$  の母点領域の隣接関係と  $\Delta_{l_1}^{P2}$  (赤),  $\delta_{l_1}^{P2}$  (青)  
Fig. 3 Adjacency among generator regions for  $P1, \dots, P5$  and  $\Delta_{l_1}^{P2}$ :red,  $\delta_{l_1}^{P2}$ :blue.

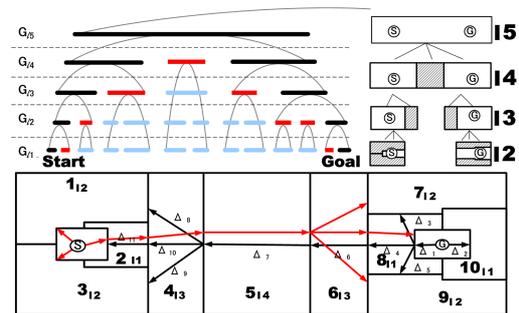


図 4 距離指標を用いた最短経路探索

Fig. 4 Search process of SPP by using spatial index tree.

$B10, B11$  は同じ母点の組合せであるので  $\Delta_{l_1}^{P2}$  とはしない。

### 2.3.3 空間索引木の特徴

空間索引木は以下の特徴を持つ。

- 道路網  $G (= G_{l_0})$  をノード NV 分割とエッジ NV 分割し、さらに母点領域ごとに統合して階層構造をボトムアップに構成している。
- 上位層でのノード NV 分割には、下位層での母点領域を 1 つのノードとし、母点領域間の隣接関係を用いてそれらを結ぶエッジを生成し、その距離を母点領域間の最短経路の距離としたグラフを用いる。
- 各母点領域の持つ距離指標としては、(i) 境界点間の最短経路と距離、(ii) 境界点間の最長経路と距離、(iii) 母点境界点間経路と距離を持つ。
- 空間索引木の生成段階では、道路網  $G$  のみを用いており、その上のオブジェクトについては考慮していない。オブジェクトとグラフの関連付けについては 5 章で説明する。

### 2.3.4 距離指標を用いた最短経路探索

距離指標の定義式 (5), (6) を用いて、与えられた始点  $s$  と目的地点  $g$  に対して、次の手順で最短経路を抽出できる。図 4 に空間索引木と平面図を用いて探索の様子を示す。

[部分木の抽出]

始点  $s$  と目的地点  $g$  が  $G_{l_0}$  上に与えられたとする。まず空間索引木において、 $s$  と  $g$  の両方を含む母点領域を発見

する。これは、探索に必要な部分木を取り出すことに相当する。このため、 $s$  と  $g$  それぞれを含む  $G_{l_1}$  の母点領域より、母点領域の包含関係を用いて順に上方に探索する。その結果、 $s$  と  $g$  が同一の母点領域となる階層（第  $i$  層とする）が判明する。図 4 上では第 5 層 ( $G_{l_5}$ ) がそれにあたるものと仮定する。

次に、発見した母点領域より、第  $i-1$  層 ( $G_{l_4}$ ) での母点の所属関係を抽出する。 $s$  と  $g$  を含まないノード（図 4 上右網掛け部）に対しては、母点領域を構成する情報とその距離指標に含まれる経路情報（母点・境界点間経路・母点境界点間経路）を取り出す。この作業を繰り返し、各階層での  $s$  と  $g$  を含まない母点領域（図 4 上左赤線）を取り出すことができる。取り出した母点領域と  $s$  と  $g$  を含む第 1 層 ( $G_{l_1}$ ) の母点領域は、それぞれの持つ境界点でつなぐことができ、1 枚の平面図（図 4 下）として見ることができる。

[ $g$  を持つ母点領域から抽出した母点領域すべてへの距離の見積り]

前段階で得られた平面図の母点領域（図 4 では  $\{S, l_{l_2}, \dots, 10l_1\}$ ）に、 $g$  を含む母点領域より距離指標 (5), (6) を用いてダイクストラ探索を実行し、 $g$  を含む母点領域を根とした木構造を構成する（図 4 下黒矢印）。この探索により各領域は、 $g$  を含む母点領域までの見積り経路と見積り距離（これを  $\Delta$ ,  $\delta$  として 7.3 節で使う）を得ることができる。

[見積り距離から実距離への置き換え]

$s$  から、 $s$  を含む母点領域が持つ境界点への経路探索を行い、境界点までの距離を算出する。同様に  $g$  を含む母点領域でも経路探索を行う。このとき、 $s$  と  $g$  が母点の場合、経路と距離は既知であるので探索を行う必要はない。

$s$  から境界点までの距離が判明すれば、各母点領域の境界点間経路をつないでいくことで、 $g$  を含む母点領域の持つ境界点まで経路をつなぐことができる。このときつなぐ境界点を持つ母点領域を決定するために、前段階で得た見積り距離を利用する。これにより、最短経路（図 4 下赤線）が得られる。

### 3. 距離指標のみを DRQ に用いた場合の問題点

#### 3.1 全数探索

先行研究 [8] で、2.3.4 項の方法を用いて、提案空間索引木が SPP に有効に利用できることを述べた。一方、 $DRQ(q, r)$  では、SPP の手法をそのまま適用すれば、 $G$  上のすべてのオブジェクトに対して、問合せ点  $q$  より探索を繰り返す全数探索が必要になる。オブジェクト集合  $D$  が大きければ、実行時間が大きくなり明らかに非効率である。

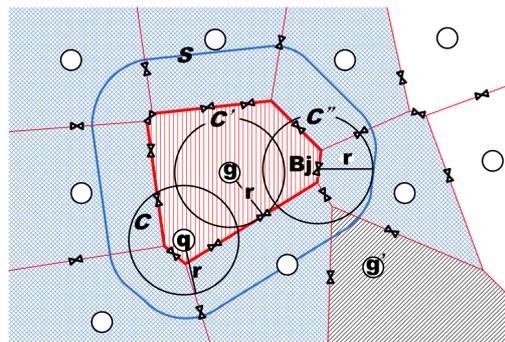


図 5 問合せ点  $q$  を含む探索対象領域の特定

Fig. 5 Identification of target generator regions for processing for given  $q$  and  $r$ .

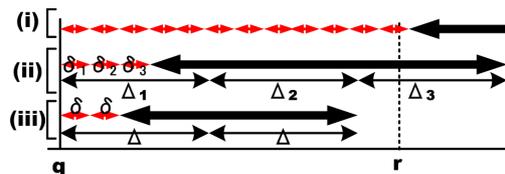


図 6 範囲問合せにおいてオブジェクトの推定位置

Fig. 6 Estimated positions by use of  $\delta$  and  $\Delta$ .

#### 3.2 距離指標のみを用いた部分木抽出による問題点

##### 3.2.1 ナイーブな部分木抽出手順

$DRQ(q, r)$  に対して、空間索引木を利用して、次のナイーブな方法に必要な部分木を抽出し、解候補となる  $D$  の部分集合を取り出すことを考える。

図 5 は、 $q$  が所属する母点領域  $g$ （赤領域）の持つ母点境界点間経路集合中の最も短い経路が、初めて  $r$  以上になる階層を取り出した結果を示している。つまり、 $g$  を中心とし半径  $r$  の円  $C'$  を考えていることになる。このとき、 $g$  内での  $q$  の位置を特定できないため、 $g$  内のオブジェクトはすべて解候補となる。次に、空間索引木では、 $g$  内での  $q$  の位置を特定できず、どの隣接領域に近いかわからないため、 $g$  の隣接する母点領域をすべて調べる必要がある。さらに、その隣接する母点領域の持つ  $\delta$  の値が  $r$  以下であれば、領域を拡大し、 $\delta$  の累計が初めて  $r$  以上になる領域まで特定し続ける必要がある。これらの処理の結果、図 5 青線  $S$  で囲まれた領域と重なる母点領域（網掛け部）が抽出対象となったものとしよう。このとき以下の 3.2.2 項と 3.2.3 項の問題がある。

##### 3.2.2 位置推定に関する累積誤差

図 5 で、母点領域  $g'$ （黒斜線部）は母点領域  $g$  と直接隣接していないが、 $g$  の円  $C''$ （境界点  $B_j$  からの半径  $r$  の円）と一部領域が重なっている。このような領域は、次の母点領域の位置推定に関する累積誤差の問題が発生する。

母点領域  $g$  から  $g'$  までの距離見積りの値は、経由する各領域の距離指標  $\delta$ （図 6 で  $\delta_1, \delta_2, \delta_3$ ）の合計により見積もられ実際より短く推定される。同様に距離指標  $\Delta$ （図 6  $\Delta_1, \Delta_2, \Delta_3$ ）の合計は実際の距離より長い。

図 6 は,  $DRQ(q, r)$  において,  $q$  から推定した  $g'$  の存在範囲を黒太矢印で示したものである. 左端が問合せ点  $q$  の位置を示し, 問合せ半径  $r$  の位置を  $r$  で示している.

この図 6 (i), (ii), (iii) の存在範囲は次のように算出する. 母点領域  $g$  より, 母点の隣接関係を用いて  $\delta$  と  $\Delta$  に関する 2 種類の最短経路木を構成する. このとき 2 つの木は同一構造と限らない. この生成した最短経路木で領域への距離に  $\delta$  を利用し合計した値を  $min$  とし,  $\Delta$  を利用し合計した値を  $max$  とする. 図 6 (ii) は 3 領域を経由したものを示し, このときの  $min = \delta_1 + \delta_2 + \delta_3$  となり,  $max = \Delta_1 + \Delta_2 + \Delta_3$  となる.

[(i) と (iii) の場合]

もし見積もった  $min$  が  $r$  を超えているとその領域  $t$  は対象に含まれない. したがって, この  $t$  を刈り込むことができる (図 6 (i)). また, 見積もった  $max$  が  $r$  以下ならば,  $t$  は問合せ円  $C''$  にある. したがってその領域内にあるオブジェクトはすべて解候補となる (図 6 (iii)).

[(ii) の場合]

$r$  の値が,  $min$  と  $max$  の間にあると探索対象領域となる (図 6 (ii)).

このように  $r$  が大きくなり経由する領域が増えると, 位置推定の誤差が累積し, この矢印の幅が広がり探索対象領域が増える結果となる. 先行研究 [8] では, 提案索引木は長距離を対象とした SPP で有効であることを確認したが, 反対に DRQ に対しては, 単純にナイーブな方法を用いると,  $r$  が大きい場合に有効に利用できないことになる.

3.2.3 空間索引木での刈込みへの累積誤差

前項は, 同階層での距離指標の累積誤差について述べたが, 下位層を利用した空間索引木での刈込みでも同様な問題が発生する.

空間索引木では, 上位層は複数の下位層の母点領域を統合して作られている. 上位層では, 図 6 (ii) に該当する領域であるが, 下位層でさらに展開して図 6 (i), (ii), (iii) の場合に分けることができる. そして判定を繰り返すことで, 下位層を利用して刈り込むことができる. 下位層で, 刈込みできる領域として考えられる部分は,  $q$  を持つ母点領域に直接隣接している領域の中で,  $\Delta$  の値が  $r$  以下となる領域を刈り込むことができる.

図 7 は, 図 5 に対応して母点領域を線分で表し, 包含関係は灰線を用いて表した図である. 同図の (a), (b), (c) において, 同じ位置にある線分は, 同じ母点領域を示す. (a) を実際の木とすると, (b) は  $\delta$  を用いて表している. 同様に (c) は,  $\Delta$  を用いている. この図を用いて,  $DRQ(q, r)$  において, 空間索引木の階層構造と距離指標のみを用いた場合の刈込みの問題点を示す.

たとえば, 図 7 の青線は, 図 5 での母点領域  $g$  を示す. 階層構造を用いて部分木を刈り込む際には, 抽出する部分木をできるだけ小さくしたい. そのため, 図 7 (a) で示す

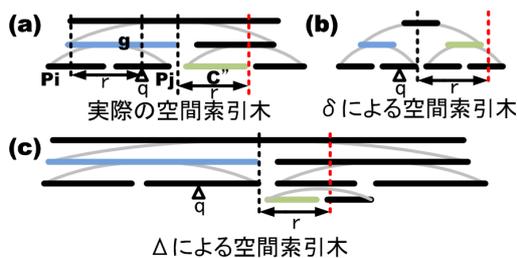


図 7  $\Delta, \delta$  より推定された空間索引木  
Fig. 7 Estimated subtrees by use of  $\Delta$  and  $\delta$ .

緑線の領域を円  $C''$  に完全に含まれる領域として刈り込む場合, 完全に含まれるか否かを, 距離の指標のみで判定できるかを考えてみよう. 図 7 (b) の場合, 同階層では, 前項と同様のことがいえる.  $\delta$  のみを用いていると遠くの領域が近くに見え, 必要でない母点領域まで抽出することになる. 図 7 (c) の場合, 円  $C''$  に完全に含まれる母点を示す線分は, 同階層よりたいていの場合, 下の階層に存在している. したがって (a) の緑線の領域よりもより下層の母点領域を取り出すことになり, 領域数が多くなって, 領域の抽出とその領域に至るための計算が増える.

このように DRQ や  $kNN$  に対して式 (5), (6) の距離指標が不十分である理由は, 2 点間の距離のみの指標であり, それらが 2 点間の位置関係と母点領域間の位置関係を保持しないことに起因していると考えられる. その結果, 領域を刈り込む段階で冗長な領域を残すことになる.  $DRQ(q, r)$  では全方位にノードを展開する必要があるため, 母点領域の位置と大きさを表す指標が新たに必要となる. このため次章で拡がりを示す指標を導入し, 空間問合せの効率化を図る.

4. 拡がり指標

4.1 母点領域の MBR の組込み

母点領域の拡がりを示す指標の要件は以下のように示すことができる. つまり,

- 母点領域に含まれるノードやエッジの位置をすべて内包している.
- 上位の階層を生成した場合, 所属母点領域間の包含関係をそのまま維持できる.
- 指標を構成するデータ量が少ない.

上記要件を満たす候補には, エリア NV 図 [9]・最小包含円や最小外接長方形 (MBR) が考えられる. しかし, エリア NV 図は領域の縁を複数の直線や曲線で持つためデータ量が大きくなり, また, 最小包含円は図形の包含関係を階層化した場合維持できるとは限らない. このため, 本稿では, 母点領域の MBR を計算し, 空間索引木の拡がり指標とすることを提案する.

MBR は, 母点領域から容易に計算でき, 階層化がしやすい [11]. また領域の重複を可能としており, ある矩形の

各母点の持つ母点領域の情報	<b>Node</b>	NodeID 母点nodeID 母点への 一步先のnodeID 母点への距離・経路 Level	<b>Edge</b>	EdgeID 母点nodeID-境界点ID or 境界点ID-境界点ID 距離・経路 母点nodeID Level	オブジェクト 情報 種類:個数 Level1では 識別子も持つ
	<b>Generator</b>	母点nodeID $\Delta$ $\delta$ MBRのXY座標 Level	<b>Boundary</b>	境界点ID 母点nodeID1 母点nodeID2 母点への距離 Level	
原地図	<b>Node</b>	NodeID X座標 Y座標	<b>Edge</b>	NodeID1 NodeID2 距離	NodeID or 座標 案種・店舗名等

図 8 空間索引木の MBR 付加後のデータ構造

Fig. 8 Data structure with MBR's in spatial index tree.

形は内部のノード・エッジ・境界点の位置をすべて包含できる。また、MBRの構造のデータとしては、2つの座標を持つだけでよい。

提案手法の母点領域の MBR の特徴は以下にまとめることができる。

- (1) 上位の母点領域の持つ MBR は、下位の母点領域の MBR を完全に包含する。
- (2) 同じ層での MBR は、一部重複する。
- (3) MBR は上位層になるほど広くなり、最上位層ではグラフ全体を覆う。
- (4) 範囲問合せ時に範囲を粗く絞ることができる。
- (5) 空間索引木において、MBR は母点領域に対して道路網からボトムアップに生成する。

#### 4.2 空間索引木の MBR 付加後の構造

図 8 にノード・エッジ・境界点の保持データを示す。下部が与えられた道路網の情報であり、上部が空間索引木の構造部分を示す構造化情報である。以下で、Node・Boundary・Edge・Generator はデータ構造を指し、ノード・エッジ・境界点は実体を示している。Node の情報は、ノード NV 図によって生成し、Boundary は、エッジ NV 図から生成する。Boundary は、母点の隣接関係と母点と境界点間の距離を持つ。上段の Edge は、1つ下の階層での経路集合を示す。Generator は母点領域の概要を示す。

### 5. オブジェクト指標

2.3 節で述べたように、提案空間索引木は道路網データのみを用いて構成されおり、オブジェクトの情報が付加されていない。ここでは、オブジェクト指標として、種別に分類したオブジェクト数を母点領域に与え、空間索引木に関連付ける方法を述べる。

#### 5.1 オブジェクト指標の組込みタイミング

本手法において、空間索引木とオブジェクト集合  $D$  の関連付けを行うタイミングは、木の確定後にオブジェクトの種類ごとに行う。これにより、様々な種類のオブジェクト集合を1つの木に組み込むことができる。

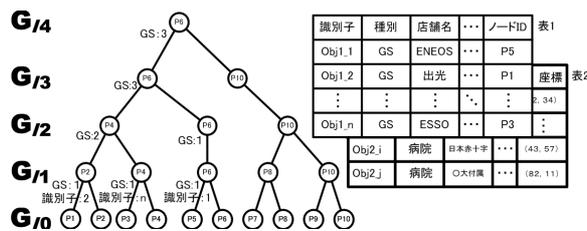


図 9 オブジェクト指標を組込んだ空間索引木とオブジェクト集合  $D$

Fig. 9 Spatial index tree with object indices and object tables.

#### 5.2 空間索引木へのオブジェクト指標の組込み手順例:

図 9 表 1 各オブジェクトに対してノード ID やエッジ ID が記述されている。表 2 はオブジェクトの位置座標のみを持つ場合である。

図 9 左に、道路網  $G_{l_0}$  とオブジェクト集合  $D$  より、空間索引木の階層  $G_{l_1}$ ,  $G_{l_2}$ , ... にオブジェクトに関連付けた情報を示している。たとえば種別 GS について考えると、 $G_{l_1}$  の母点  $P2$  では、所属ノードとして  $P1$ ,  $P2$  を持つため、表 1 を参照して、所属ノードの持つ Obj1.2 を検索し、種別の GS と個数の 1 をオブジェクト指標として持つ。

すべての階層の母点領域にオブジェクトの関連付けを行うと、同一オブジェクトが多重の出現することになるため、オブジェクト識別子は  $G_{l_1}$  にのみ配置している。また図 8 に点線でオブジェクトに対する情報を示す。

オブジェクト指標の組込み手順は以下のとおりである。

#### [A: オブジェクトの存在するノード・エッジが既知の場合]

道路網  $G_{l_0}$  から  $G_{l_1}$  の構成情報と図 9 の表を用いてオブジェクトと母点領域の関連付けを行う。すなわち、所属する母点が明確であるので、まず母点領域にオブジェクト識別子を持たせる。次に、母点は自身の母点領域の持つオブジェクト数を数え、オブジェクト指標に付加する。境界点上にあるオブジェクトは、2つの母点に所属することにする。

第 2 層以上では、オブジェクト指標は、その母点領域の統合情報から合計することができる。すなわち、上位層での母点領域に対するオブジェクトの個数は、下位層の母点領域にあるオブジェクトの個数の合計を繰り返す。この繰返しにより、空間索引木にオブジェクト情報を付加することができる。(Algorithm1)

#### [B: オブジェクトの位置情報のみが与えられている場合]

A の場合と同様に、第 1 層 ( $G_{l_1}$ ) でオブジェクトと母点領域を次のように関連付ける。オブジェクトの位置情報を用いて、空間索引木の上位層より、MBR の包含関係を用いて階層を下り  $G_{l_1}$  に至ってそのオブジェクトを含む MBR を見つける。MBR が重複した領域にオブジェクトが属する場合、それぞれの母点領域のノードとエッジを展開し、オブジェクトがどの母点領域に存在するかを確定する。そ

の後 Algorithm1 を実行する。

注) Algorithm1 では、図 8 の Node の情報に対して空間索引木の下位層より上位層に向かってノード ID と母点ノード ID を調べる。1 行目より 8 行目は、 $G_{l_1}$  への組込み、9 行目から 20 行目までは、 $G_{l_2}$  以降への組込み手順である。Algorithm1 中では記述の簡略化のためオブジェクト ID とノード ID は同一にしている。また各ノードは、右のようなクラス構造を持っている。2・10・12 行目の N (Level) はそのレベルに存在するノード集合を示している。2・12 行目の p.get (条件) は、条件に一致するノード p を N (Level) の中から取り出す作業である。3-7 行目と 13-17 行目は、p が母点と所属ノードの差異に応じ数え上げている。7 章の評価においては、処理を簡単にするためにノード上にオブジェクトがあると仮定している。

### 5.3 更新方法

オブジェクト集合  $D$  に対する更新は、以下のように空間索引木に反映する。

新規追加：

オブジェクトが新規にデータ集合  $D$  に追加された場合、5.2 節の組込みで述べた手順で、オブジェクト指標の関連付けを実行する。

削除：

オブジェクトがデータ集合  $D$  より削除された場合、追加と同様に、下位層より順に母点領域のオブジェクト指標を更新することで、削除を行うことができる。

編集：

オブジェクトの店舗名などの空間索引木に含まれない情報が更新された場合は、空間索引木内にオブジェクトの情報は存在しないため、空間索引木を変更する必要がない。これに対して、空間索引木に含まれるオブジェクトの位置や識別子が更新された場合は、削除と新規追加の手順を用いて、空間索引木の更新を行う。

### 5.4 確定部分でのオブジェクト情報への展開

6 章で説明する DRQ の問合せ手順での最終段階は以下のようなになる。第  $i$  階層 ( $G_{l_i}$ ) において、問合せ円に含まれる母点領域を持つオブジェクトの個数が判明した場合に

**Algorithm 1** Procedure Add object index to spatial index tree

```

1. for each o ∈ D
2.   p.get(p's nodeID==o) ∈ N(Level 1)
3.   if p's nodeID == p's GeneratorID
4.     p.addTree(p's nodeID)
5.   else
6.     p's GeneratorID.addTree(p's nodeID)
7.   end if
8. end for
9. for each i ≠ 1 Level
10.  for each p ∈ N(Level i-1)
11.   if p's nodeID == p's GeneratorID
12.     p.get(p's nodeID==p's GeneratorID) ∈ N(Level i)
13.     if p's nodeID == p's GeneratorID
14.       RaddTree(p's objectcount)
15.     else
16.       p's GeneratorID.addTree(p's objectcount)
17.     end if
18.   end if
19. end for
20. end for

```

Node class{  
 Int nodeID  
 Int Level  
 Int GeneratorID  
 Int objectcount  
 Double MaxX,MaxY,MinX,MinY  
 list objectList  
 void addTree(int i){  
   if(level==1){  
     objectcount++  
     objectList.add(i)  
   } else  
     objectcount = objectcount + i  
   }  
 }  
 objectList [int objectID, ...,int objectID]  
 objectID = nodeID

は次の手順でオブジェクトの詳細情報を得る。

空間索引木では、 $G_{l_1}$  がオブジェクト識別子を持つため、そこまで木構造をたどる。 $G_{l_i}$  である母点領域に含まれているため、 $G_{l_{i-1}}$  での母点ノード ID は容易に分かる。このときオブジェクトを持つ  $G_{l_{i-1}}$  での母点領域のみを次に展開する。 $G_{l_{i-1}}$  でオブジェクトを持つ領域が判明する。同様に  $G_{l_{i-1}}$  でも展開できるので、 $G_{l_1}$  に至るまで繰り返す、オブジェクト識別子が判明する。

### 5.5 オブジェクト指標の特徴

オブジェクト指標は以下の特徴を持つ。

- 各母点領域が、オブジェクトの個数を所持する。
- 図 9 のように、オブジェクトの詳細な情報 (たとえば、ガソリンスタンドの事業者名など) は、データ集合  $D$  中に記述しても、空間索引木には持たせない。
- オブジェクトの識別子は、 $G_{l_1}$  の母点領域を持つ。
- オブジェクト指標は空間索引木の生成後に別途に付加するために、道路網の更新とオブジェクトの更新を別々に行うことができる。
- オブジェクト指標は種別ごとに生成でき、多種類のオブジェクトに対しても生成できる。これにより、1 つの空間索引木に対して、多種類のオブジェクトを検索する複合探査にも適用できる。多種類オブジェクトに対する複合探査に対する手法の議論と評価は別稿に譲り、本稿では単一のオブジェクトに対する手法の提案と評価を行う。

## 6. DRQ の問合せ手順

本章では提案している各指標を用いた  $DRQ(q, r)$  に対する問合せ手順を述べる。手順は、部分木の特定・粗査・精査・オブジェクト情報の展開の 4 段階に分けることができる。図 10 に、DRQ での処理の段階を色別で示している。

#### 第 1 段階：MBR を用いた部分木の特定

$DRQ(q, r)$  に対して、問合せ点  $q$  を中心にユークリッド距離  $r$  で問合せ円  $C$  を描く。問合せ点  $q$  からネットワーク上の距離  $r$  で到達可能な地点は、円  $C$  内に存在する。この円  $C$  を用いて空間索引木より探索対象部分木を特定する処

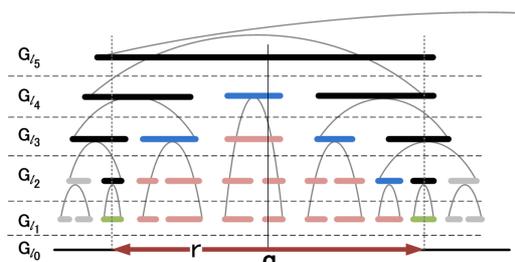


図 10 DRQ に対する母点領域の包含関係に着目した部分木  
 Fig. 10 Subtrees by use of inclusion relationships among generator regions (DRQ).

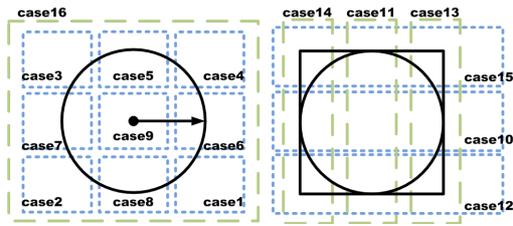


図 11 円 C と MBR の位置関係の場合分け

Fig. 11 Detailed categorization of Spatial relationships of MBR and query circle C.

理が第 1 段階である。

ここでは最上位層より、円 C と MBR の位置関係に基づく処理と MBR の包含関係に基づく処理を交互に繰り返すことで範囲を刈り込んでいくことができる。

[同一階層での円 C と MBR の位置関係に基づく処理] :

図 11 に示すように、問合せ円 C と MBR の位置関係は 16 通りに分類でき、大きく 4 つに分けることができる。

状況 1 円 C が MBR を完全に包含する (図 11 (case9)).

状況 2 円 C に一部のみ MBR が重なっている (図 11 (case1-case8)).

状況 3 円 C より MBR が大きく、その MBR の一部に円 C が重なる (図 11 (case10-case16)).

状況 4 円 C と MBR は、まったく重ならない (図 11 (その他)).

状況 1 では、問合せ円 C が MBR を完全に包含するため、この母点領域の部分木のより下位の層でも完全に包含されるため、これ以上、下位の層では、母点領域の抽出計算を行う必要がない。図 10 においては、青の領域にあたる。

状況 4 では、より下位の層の母点領域では問合せ円 C に重ならないため、下位の層では、抽出のための計算を行わない。図 10 では  $G_{l_2}$  でのグレーの領域にあたる。

その他の母点領域は基本的に状況 2 または状況 3 にあたり、これらの領域に対する木構造を下位の層で展開すると、状況 1~4 に再び分類できる。これを再帰的に繰り返す。このため以下の 2 つの処理を繰り返せばよい。

(1) 円 C と MBR の位置関係を用いた関係の判定 (Algorithm2)

(2) 円 C の外接正方形と MBR の位置関係を用いた関係の判定 (Algorithm3)

Algorithm2 では、状況 1 と状況 2 のみを判定できる。この判定は、問合せ円 C の中心  $q$  から MBR の頂点まで距離と半径  $r$  を比較する。状況 1 の場合は、2-5 行目までの条件式で判定できる。状況 2 は、MBR の頂点までの距離が  $r$  以下になる (6-11 行目)。

状況 3 と状況 4 は、 $q$  から MBR の各頂点までの距離が  $r$  より大きく距離のみの判定では判別できない。そこで、円 C の外接正方形を作成し、その正方形と MBR の位置関係で状況 3 と状況 4 を判定するために Algorithm3 を必要

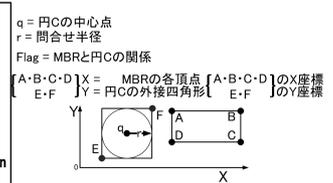
```

Algorithm 2
Procedure Distance Range Query Part 1
1. for each  $p \in N$ 
2. if  $\sqrt{(AX-qX)^2+(AY-qY)^2} < r$  then
3.   if  $\sqrt{(BX-qX)^2+(BY-qY)^2} < r$  then
4.     if  $\sqrt{(CX-qX)^2+(CY-qY)^2} < r$  then
5.       if  $\sqrt{(DX-qX)^2+(DY-qY)^2} < r$  then Flag  $\leftarrow$  i
6.     else Flag  $\leftarrow$  ii : end if
7.   else Flag  $\leftarrow$  ii : end if
8. else Flag  $\leftarrow$  ii : end if
9. else if  $\sqrt{(BX-qX)^2+(BY-qY)^2} < r$  then Flag  $\leftarrow$  ii
10. else if  $\sqrt{(CX-qX)^2+(CY-qY)^2} < r$  then Flag  $\leftarrow$  ii
11. else if  $\sqrt{(DX-qX)^2+(DY-qY)^2} < r$  then Flag  $\leftarrow$  ii
12. else Flag  $\leftarrow$  iii or iv : end if
13. end for
    
```



```

Algorithm 3
Procedure Distance Range Query Part 2
1. for each Flag iii or iv  $\in N$ 
2. if  $EX < DX$  &  $FY < FY$  then
3.   if  $DX < EX$  &  $FX < BX$  then
4.     Flag  $\leftarrow$  iii
5.   else if  $EX < BY$  &  $BY < FY$  then
6.     Flag  $\leftarrow$  iii
7.   else if  $DX < EX$  &  $FX < BX$  then
8.     Flag  $\leftarrow$  iii
9.   else if  $DY < EY$  &  $FY < BY$  then
10.    if  $EX < DX$  &  $DX < FX$  then
11.      Flag  $\leftarrow$  iii
12.    else if  $EX < BX$  &  $BX < FX$  then
13.      Flag  $\leftarrow$  iii
14.    else if  $DY < EY$  &  $FY < BY$  &  $DX < EX$  &  $FX < BX$  then
15.      Flag  $\leftarrow$  iv
16.   else
17.     Flag  $\leftarrow$  iv
18. end for
    
```



とする。このとき、MBR と円 C の外接正方形の位置関係は、Algorithm3 の右下に示す記号を使い、詳細を次に列記する。

case10 MBR の辺 AB・DC が点 F より下、点 E より上にある。

case11 MBR の辺 AD・BC が点 F より右、点 E より左にある。

case12 MBR の辺 AB が点 F より下、点 E より上にある。

case13 MBR の辺 AD が点 F より右、点 E より左にある。

case14 MBR の辺 BC が点 F より右、点 E より左にある。

case15 MBR の辺 DC が点 F より下、点 E より上にある。

case16 辺 BC が点 F より右、かつ辺 AB が点 F より上、かつ辺 DC が点 E より下、辺 AD が点 E より左にある。

上記の位置関係と Algorithm3 の対応は以下のとおりである。2 行目より 4 行目までの判定文で図 11 の case15 にあたる。次の 5・6・7 行目は case12 にあたる。また、8・9・10 行目は case13 に、11・12 行目は case14 にあたる。13・14 行目は、case16 である。case10 は、case12・15 の判定文に含まれ、case11 は同様に case13・14 の判定文に含まれる。これらすべての判定文で満足しなかったものが、状況 4 にあたる。

この処理を木構造の展開を行いながら再帰的に実行した結果、 $G_{l_1}$  でも状況 2 に存在する母点領域を抽出できる。この領域は、2次元平面上において円 C の最も外側にかかる領域である。この領域とそれより内側の状況 1 の領域を粗査にかけると、図 10 での次の粗査の対象領域は、青・緑の領域にあたる。また、桃色の領域は、より上位の層 (青領域) で粗査対象となっているため、その子孫領域は粗査を実行する必要はない。

第 2 段階：距離指標を用いた粗査

この段階では以下の 3 つを確認する。

- (1) 母点領域の隣接関係を用いた道路網上での接続の確認
- (2) 距離指標  $\Delta$  を用いた近接領域の確認

表 2 利用した道路網とそのサイズ  
Table 2 Road networks and their sizes.

所在地	ノード数	エッジ数	道路網ノードサイズ	道路網エッジサイズ
みなべ町	5,601	7,509	135,168 バイト	1,495,040 バイト
北部和歌山市周辺市町村	40,323	54,234	1,011,712 バイト	9,474,048 バイト

(3) 距離指標  $\delta$  を用いた遠方領域の確認

すなわち、第 1 段階で取り出した母点領域に対して、(1) 母点領域の隣接関係からその母点領域が問合せ点  $q$  から抽出した母点領域のみを經由して到達できるかを確認する。同時に、(2)  $\Delta$  を用いて各領域までの距離が  $r$  以下であるかを見積もる。このとき  $r$  以下であれば、確実にその領域は距離  $r$  以内に完全に含まれている。

また同時に (3)  $\delta$  を用いて、母点領域が距離  $r$  より大きくなるか否かを見積もる。その結果、距離  $r$  よりも遠い位置にある領域は、実際に遠い位置にあることが分かり、精査する必要がない。

これらにより、完全に含まれる領域である部分と、粗査の段階でも距離  $r$  内か不明な領域と、明らかに遠い領域か否かを判定できる。

第 3 段階：精査

第 3 段階では、詳細に位置を調べる必要がある領域内のオブジェクトに対して探索を行う。

- (1) 見積り距離から実距離への置換え
- (2) 対象オブジェクトに対する探査

精査段階では、先行研究 [8] で提案した見積り型経路探査を行う。このときのオブジェクトまでの経路探査での対象は、粗査の結果、距離  $r$  内か否か不明な領域にあるオブジェクトであり、経路探査の始点  $s$  は  $q$  の地点である。

この段階では、まず  $q$  から  $q$  の所属する母点領域の持つ境界点間までの経路探査を行い、精査対象領域に存在するオブジェクトから境界点までの経路探査を行って正確な距離を算出する必要がある。最後に境界点での距離計算を行う。また、境界点での実距離への置換えの段階で境界点を展開し、その結果、実距離が距離  $r$  以上になる経路は、以降展開する必要がない。ここで必要な部分は距離  $r$  以内の経路で到達できるオブジェクトである。

第 4 段階：オブジェクト情報の展開

前段階の結果と粗査により得られる解候補となるオブジェクト集合が  $DRQ(q, r)$  の解のオブジェクトであり、それらのオブジェクトに対して、5.4 節で述べたオブジェクト情報の展開を行う。

第 1 段階から第 4 段階までを行い、空間索引木の探査対象領域を刈り込み、最終的に解となるオブジェクト集合を得ることができる。

7. 評価

提案手法の有効性を確認するためにシミュレーショ

ンを実行した。実行環境は Windows7 64bit 版、CPU は Xeon2 G×2 でメモリ 8 ギガバイトである。7.2 節で、Kriegel ら [1] の手法を発展させた階層型参照点埋込み法について説明し、7.3 節で提案手法で距離指標の評価を行う。7.4 節では、ナイーブな部分木抽出手順と MBR を用いた手法の刈込みの比較を行い、7.5 節では、空間索引木へのオブジェクト指標の組込み処理時間と組み込んだファイルサイズについて調べる。また、7.6 節では、DRQ に対して、Kriegel らの手法の階層化あり（階層型参照点埋込み法）と階層化なし、提案手法を比較する。本章では、階層とレベルは同意で用いる。

7.1 数値地図の利用

国土地理院数値地図 2,500（空間データ基盤、1/2,500 縮尺）近畿-1 に含まれる和歌山県のデータを用いる。表 2 にその所在地と構成ノード数を示す。

[空間索引木の生成に関する評価]

提案手法で新たに追加される MBR の計算は、境界点の座標の算出と母点領域の XY 座標の計算であり、生成時間は先行研究 [8] での空間索引木の生成時間と同じであるため、ここでは割愛する。

図 12 は、2 種類の道路網を用いて、生成した空間索引木の各構成要素のファイルサイズならびにオブジェクト指標サイズと生成時間を示している。MBR の追加による増加ファイルサイズは、横軸のオブジェクト設定確率（なし）の場合と等しく、5,601 ノードの空間索引木で 311 キロバイトとなっており、40,323 ノードの空間索引木では 2,371 キロバイトとなっている。これは、空間索引木の Node, Generator, Edge, Boundary の合計ファイルサイズの 2%前後であり、MBR の付加による影響は小さくとどまることが分かる。また、オブジェクト指標の組込み時間やサイズについては、7.5 節で説明する。

7.2 階層型参照点埋込み手法

次に、比較対象として空間へのオブジェクト情報の埋め込み手法について考える。Kriegel ら [1] の参照点埋込み手法の原理を生かして拡張した階層型参照点埋込み法について説明する。彼らの提案する手法では、階層化時のノード選択方法が必ずしも明確でなく、階層間の参照点の距離関数の持ち方も明確でない。このためここでは、学習データを与え、見積り算出時に利用頻度の高い参照点を選択し上位の参照点とする階層化を行う方法について述べる。

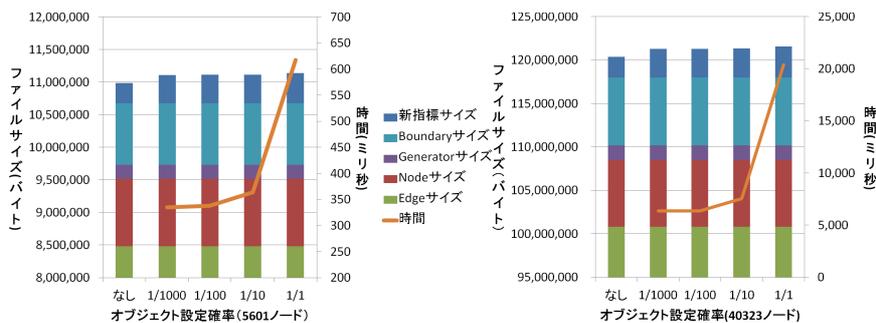


図 12 オブジェクト指標の組み込みによるファイルサイズの変化と組み込み時間

Fig. 12 Comparison of file size of spatial index tree and processing time for generation of object index.

[距離見積りの定義]

ここでは, Kriegel ら [1] の手法で定義された距離見積りを利用する. 第  $i$  層での参照点を  $R_1^i, R_2^i, \dots, R_j^i, \dots$  とする. 始点, 目的地点をそれぞれ  $s, g$  とするとき, 第  $i$  層の  $s - g$  間の距離見積り  $\Delta_{kr}^i, \delta_{kr}^i$  は以下のとおりである.

$$\Delta_{kr}^i = \max_j (d_{net}(R_j^i, s) + d_{net}(R_j^i, g)) \quad (7)$$

$$\delta_{kr}^i = \min_j (|d_{net}(R_j^i, s) - d_{net}(R_j^i, g)|) \quad (8)$$

参照点  $R_j^i$  を利用して  $s - g$  間距離を見積もる場合,  $s - g$  間の距離が最も距離が大きくなる並びは  $s - R_j^i - g$  の並びであり, 最も距離が小さくなる並びは  $s - g - R_j^i$  か  $g - s - R_j^i$  である. この場合,  $d_{net}(R_j^i, s) - d_{net}(R_j^i, g)$  の値が負になる場合があるので, 式 (8) で絶対値をとっている.  $d_{net}(R_j^i, s) + d_{net}(R_j^i, g)$  は, つねに正の値をとる.

[階層型参照点埋込み法での索引生成手順]

- (1) 道路網データ上にランダムに参照点を選択する. 各参照点より他のすべてのノードに対して, 最短経路木をそれぞれ生成する. これを第 1 層とする.
- (2) 第 2 層の参照点を決定するため, 以下の学習アルゴリズムを用いる. すなわち, 複数の  $(s, g)$  の組を与え, 式 (7), (8) に従い,  $\Delta_{kr}^2$  と  $\delta_{kr}^2$  をそれぞれ算出する. その際の  $\Delta_{kr}^2$  と  $\delta_{kr}^2$  を与える参照点に得点を与える. これを繰り返して得られた得点の高い上位半分の参照点を次の第 2 層の参照点とする. 第 2 層の参照点の持つ最短経路木は, 第 1 層での最短経路木と同じものとする.
- (3) 参照点数が一定数になるまで階層化を繰り返す.

注 1) ここでは (3) の一定数は, 提案手法の最上位層と揃えるために 4 としている.

注 2) 学習データにおける  $(s, g)$  の選択はランダムに行うものとする. ただし, 7.6 節で利用する索引では, DRQ に沿った学習データとするために,  $g$  はオブジェクト集合  $D$  より選択している.

7.3 提案手法と階層型参照点埋込み手法の距離見積り精度の比較

まず,  $G$  上の任意の点  $(s, g)$  の組を与え, 1 対 1 での見積りの距離と実際の距離の比をとることにより, 両者での距離見積り精度の比較を行う.

提案手法の距離指標 (2 章式 (5), (6)) は母点領域の大きさを示し, 比較手法の距離見積り (7 章式 (7), (8)) は 2 点間の距離見積りである. このため, 両者の見積り精度を比較するため, 2.3.4 項で述べた  $\delta, \Delta$  と上記の  $\delta_{kr}^i, \Delta_{kr}^i$  を比較する. 提案手法は, 領域ごとに距離指標を持つため,  $g$  の母点領域より  $s$  の母点領域までの距離見積りとなっている. このとき利用するグラフは  $(s, g)$  の組に対し複数の階層での領域を用いた 1 枚のグラフになる. そのため提案手法の見積り結果は 1 つとなり, 比較手法は各階層ごとに  $\delta_{kr}^i, \Delta_{kr}^i$  が求められる.

図 13 は, 真値を 1 としたときの  $\delta$  と  $\delta_{kr}^i$  の見積り精度を比較した際の平均 (左) と分散 (右) を示している. 同様に, 図 14 は  $\Delta$  と  $\Delta_{kr}^i$  の見積り精度の平均 (左) と分散 (右) を示している. また, それぞれにおいて  $\Delta_{kr}^i, \delta_{kr}^i$  の学習データ数の影響も見ている.

図 13 を見るとレベル 1・2 では提案手法の  $\delta$  が勝るが, その他のレベルでは, 比較対象の手法の  $\delta_{kr}^i$  の精度が高い. また, 図 13 右は, 比較手法では, レベルが高くなるほど分散が大きく, 提案手法は分散が 0 に近くばらつきが少ない. 一方,  $\Delta, \Delta_{kr}^i$  を比較すると平均の値では, 提案手法が勝るが, 分散では比較手法が勝っている. この結果は以下の理由による.

階層型参照点埋込み法は, 2 点間に対する距離見積りであり, 参照点の持つ最短経路木より算出されるため精度が高いが, 提案手法の  $\Delta, \delta$  は, 母点領域ごとの持つ距離指標を用いて算出した見積りのため, 精度の誤差が大きくなることに起因すると考えられる.

7.4 MBR の刈込みノード数縮減に対する効果

次に, 母点領域の MBR による刈込みノード数の効果を確認するため, 3.2.1 項で述べたナイーブな部分木抽出法

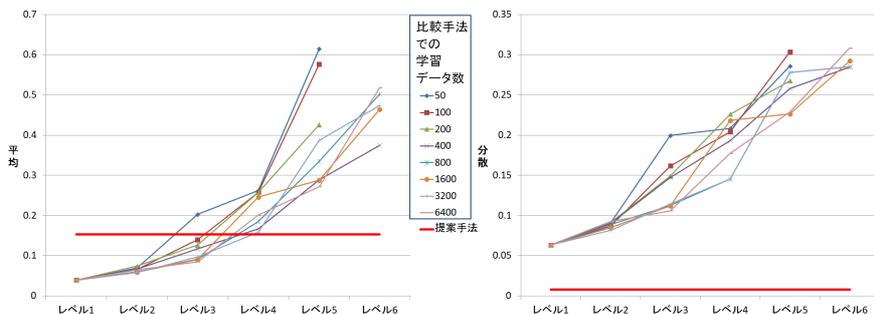


図 13 提案手法と階層型参照点埋込み手法での見積り精度の比較 ( $\delta$ )  
 Fig. 13 Accuracy of estimated distances  $\delta$  and  $\delta_{kr}^i$  (real distance = 1).

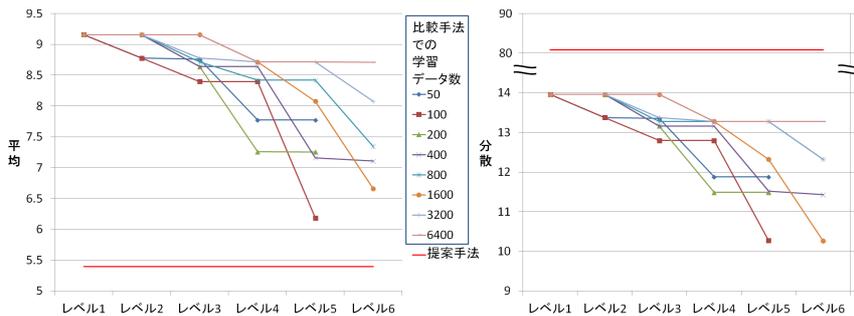


図 14 提案手法と階層型参照点埋込み手法での見積り精度の比較 ( $\Delta$ )  
 Fig. 14 Accuracy of estimated distances  $\Delta$  and  $\Delta_{kr}^i$  (real distance = 1).

と比較する。利用した地図はノード数 40,323 である。

図 15 に各抽出法で抽出された部分木の持つ道路網  $G_{l_0}$  上のノード数を縦軸 (対数軸) で示し、横軸にナイーブな手法で抽出した場合の部分木の最上位のレベルを表している。横軸にこのような設定をしたのは、以下の理由による。

道路網のノード分布の疎密によっては、 $q$  が密な部分にある場合は、取り出される部分木の最高位のレベルが高くなり、部分木に含まれるノード数が多くなる。一方、 $q$  が疎な部分にある場合には、同じ  $r$  でも最高位のレベルが低くなり、部分木に含まれるノード数が少なくなる。刈込み精度を精密に見るためには、部分木に含まれるノード数を一定にするため階層ごとに比較している。つまり、レベルが低い (図 15 レベル 3) とときは、ノード分布が疎な場合か  $r$  が小さい場合を示しており、反対にレベルが高い (図 15 レベル 9) とときは、ノード分布が密な場合か  $r$  が大きい場合に対応している。

また、図 16 には図 5 に対して、各抽出法で抽出される部分木の対応する部分を示している。ナイーブな部分木抽出法で取り出されたノードは、青楕円で、提案手法で取り出されたノードは、緑楕円となっている。また下に同じ母点領域名を付加している。同図では  $G_{l_2}$  で母点からの境界点までの最小の距離が  $r$  以上の距離になった場合を示している。そのときナイーブな手法ではその隣人領域と  $\Delta$  の値より範囲内か粗査対象になった領域 (図 5 の  $g'$ ) となる。また提案手法では、MBR での  $q$  より描ける円  $C$  を用いた

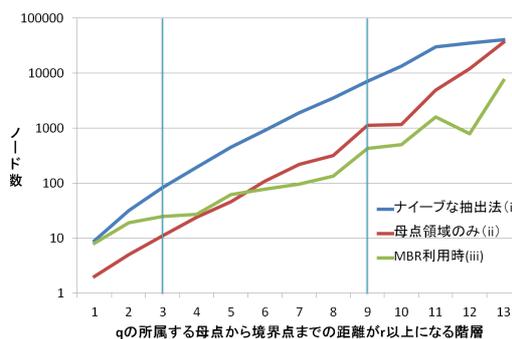


図 15 母点領域のナイーブな抽出法と MBR を利用した抽出法の比較  
 Fig. 15 Comparison of the counts of list expansions by naive approach and proposed approach using MBR.

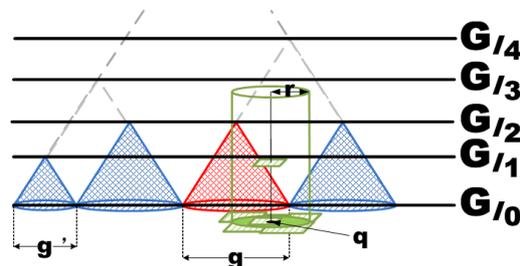


図 16  $DRQ(q, r)$  に対する抽出で対象となる部分木  
 Fig. 16 Target sub trees for extraction in process  $DRQ(q, r)$ .

判定を行い、 $G_{l_1}$  で 1 領域が完全に包含され、周辺の領域がいくつか  $C$  上にある状況を示している。

[折れ線の種類]

(i) ナイーブな抽出法 3章で述べたナイーブな部分木抽出法にあたる。

(図5・図16:赤領域)

(ii) 母点領域のみ 問合せ中心となる母点領域  $g$  に存在するノード数。

(図5・図16:青領域+斜線領域)

(iii) 提案手法 第1段階の [MBR を用いた部分木の特定] しか実行しておらず、ネットワーク的に  $r$  以内に到達不可能な領域に存在するノードも値の中に含まれている。

(図10・図16:緑領域)

[評価]

(i) と (iii) に対して、階層の低いレベルでの抽出は、部分木の持つノード数に差異が見られないが、上位のレベルになると明らかに抽出されるノード数が少なくなっている。このことから、MBR を用いた抽出法 (iii) が (i) と比較して、刈込みの効率が良く、提案手法が有効であると確認できる。

また (ii) と (iii) では、抽出される部分木の最上位のレベルを比較している。レベル6で (ii) よりも (iii) が少なくなっているため、MBR で抽出される部分木の最上位のレベルがナイーブな抽出法よりも下のレベルになっていることが分かり、提案手法に刈込みノードの縮減効果が見られる。

7.5 空間索引木に対するオブジェクト指標の組み込み

空間索引木に対するオブジェクト指標の組み込み処理時間と組み込まれた母点領域情報のファイルサイズについて確認する。図12は、オブジェクト指標を組み込んだ母点領域を示す部分 (図8のオブジェクト情報) のファイルサイズと生成時間を棒グラフと折れ線で示している。

[オブジェクトの設定]

組み込むオブジェクト数は、 $G$  のノード数に対する割合で設定している。ここでは、割合を  $1/1,000 \cdot 1/100 \cdot 1/10$  の場合と、全ノードと同じ数のオブジェクトが存在する場合の4パターンとした。またこの割合を図中ではオブジェクト設定確率としている。1/1の場合では、オブジェクトが複数あるノードもあれば、1つも存在しないノードもある。組み込み時間は、母点領域情報とオブジェクト集合  $D$  を読み込み、Node情報のリストを調査し、ファイルに出力するまでの時間を計測している。

[評価]

図12左では、存在確率が1/1の場合では、650ミリ秒かかっており、右では20,000ミリ秒である。Nodeファイルサイズに応じた、組み込み処理時間となっていることが分かる。また、オブジェクト指標のサイズは、最も多い情報の組み込まれている1/1のときに、空間索引木のNode、

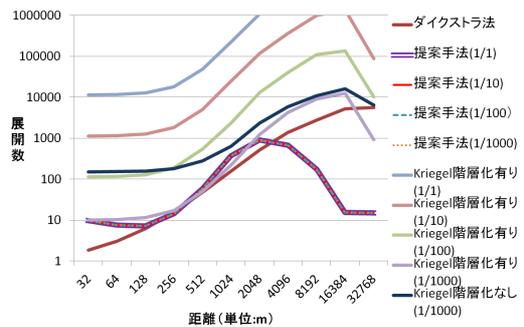


図17 DRQに対する提案手法・ダイクストラ法とKriegelらの手法での展開数

Fig. 17 Comparison of the counts of list expansions by the proposed approach, Dijkstra's algorithm and extended hierarchical embedding method for DRQ.

Generator, Edge, Boundaryの合計ファイルサイズサイズの4%程度である。この値には、母点領域に対するMBRも合計されているのでオブジェクト指標とMBRのサイズはほぼ同じであることが分かる。

7.6  $DRQ(q, r)$  に対する比較

[オブジェクトの設定とパラメータ]

DRQに対して (i) ダイクストラ法 (ii) Kriegelらの手法の階層化あり (階層型参照点埋込み法) と階層化なし (iii) 提案手法の3者を比較する。ダイクストラ法は、距離が  $r$  に到達した時点で探索を停止させるアルゴリズムとした。提案手法とKriegelらの手法の階層化ありは、前節と同様にオブジェクト数が6個 (1/1,000)・56個 (1/100)・560個 (1/10)の場合と、ノードと同じだけのオブジェクト数がある場合の4パターンについて計算した。利用した地図は、5,601ノードからなる。 $q$ の地点をランダムに200地点を与え、ノードの展開数と実行時間の平均値を比較した。また、Kriegelらの手法の階層化ありの有効性を確認するために、オブジェクト数がノード数に対して1/1,000でKriegelらの手法の階層化なしを用いて探索した結果を示す。階層化することで長距離の問合せに対してKriegelらの手法が階層化しない [4] と比較してより良い性能をあたえるものであることを確認する。

図17に、距離  $r$  を横軸で表し、縦軸を200カ所の問合せ点  $q$  に対する展開数の平均値を示す。ここでの展開数はノードの情報にアクセスした回数を指している。図18に、図17と同じ条件での次の問合せが可能となる時間を示している。

[Kriegelらの手法の階層化ありでの処理]

Kriegelらの手法の階層化ありでは、 $q$ から参照点を使ってオブジェクトまでの距離見積りを実行するために、オブジェクト数が直接展開数に影響する。このため、7.3節で評価として良かった階層4での学習データ数6,400を利用した。距離見積り後、 $r$ 以内か不明なオブジェクトには、

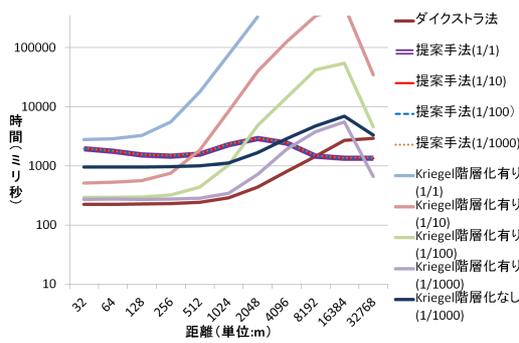


図 18 DRQ に対する提案手法・ダイクストラ法と Kriegel らの手法での実行時間

Fig. 18 Comparison of computation time by the proposed approach, Dijkstra's algorithm and extended hierarchical embedding method for DRQ.

A\*経路探索アルゴリズムのヒューリスティクス値として  $\delta_{kr}^A$  を利用した計算を行った。Kriegel らの手法では、階層化ありとなしでは、 $D$  の数によらず階層化ありの方が高速となっている。1/1,000 のみの表示であるが他のオブジェクト設定確率でも同様の結果が確認されている。これに対して、提案手法では、DRQ を実行する際に、領域を抽出し、その中の母点領域を順に展開させるため、原理的にオブジェクト設定確率によらず同じ値になる。また計測誤差が存在するが、ほぼ同一線上に表示されるため、線種へ変更して表示している。

【展開数に対する評価 (図 17)】

問合せ半径  $r$  が 2,000 m 以下 (含まれるノード数が約 1,000) であれば、ダイクストラ法が優位であるが、 $r$  がそれ以上になれば、提案手法は、オブジェクト設定数にかかわらず、展開数が頭打ちになり有効であることが分かる。問合せ範囲が地図全体をほぼ覆うような場合、ダイクストラ法ではほとんどすべてのノードを展開する必要があるが、提案手法では、 $q$  の所属する母点領域と他の母点領域との接続判定のみでよいため、展開数は少なくなる。これに対し、Kriegel らの手法の階層化ありでは、オブジェクト数に依存せず、 $r$  が大きくなれば一様に増大することが分かる。このことから提案手法は、 $D$  の個数によらず一定の展開数で終了しているが、参照点埋込み法が階層化有無によらず、 $D$  の個数で展開数が増えることが分かる。

【実行時間に対する評価 (図 18)】

ダイクストラ法は、展開するノード数が少ない階層では高速であるが多くなるにつれて時間が必要となる。それに対して提案手法では、空間索引木のファイルを読む時間が必要だがダイクストラ法と比較して大きく変化しない。提案手法と Kriegel らの手法の階層化ありを比較すると、オブジェクト数が少なく、 $r$  が短い場合、Kriegel らの手法の階層化ありが優位であるが、提案手法は、 $r$  が大きかったり、オブジェクト数が多い場合に有効に実行できる。

以上の評価より提案手法は、Kriegel らの手法の階層化あ

りとダイクストラ法と比べて、問合せ半径  $r$  の大きい問合せや  $G$  に対してオブジェクト集合  $D$  が多いときに有効であることが確認できる。これは、提案手法での  $r$  に完全に含まれる母点領域の階層が上がり、1 度に多くのオブジェクトを同時に刈り込む利点を持ち、問合せ円  $C$  に含まれるか否にある領域のレベルはつねに 1 で、精査の段階で道路網を用いて展開を必要とするノード数が少なくなるからであると考えられる。

先行研究と提案手法の違いは、部分木の抽出に対して、SPP では、母点領域の包含関係を用いて始点と目的地点を同時に含む母点領域を最上位のレベルとしていた。それに対して、DRQ では、木の全体を含む頂点より MBR に DRQ の円が絡んでいる領域を上位層より順に取り出している。また領域ごとに見ると、SPP では、始点と目的地点の間で中ほどにある領域のレベルが最も高かった。しかし DRQ では、問合せ点を含む領域のレベルが高い。先行研究 [8] と提案手法で同様の部分は、部分木の抽出後に、境界点で母点領域をつなぎ 1 枚のグラフとして見ている点であり、処理では、部分木抽出後に粗査と精査を行い、結果を導出している点である。

これらのことから空間索引木は SPP でも DRQ でも有効に実行できることが確認された。

8. おわりに

本稿では、空間索引木の DRQ への適用について議論した。まず空間索引木を直接的に DRQ に適用する場合の問題点について議論し、それらを解決するために、母点領域の拡がり指標母点領域に組み込むためのオブジェクト指標について提案した。DRQ に対して、(i) 距離指標、(ii) 拡がり指標、(iii) オブジェクト指標の 3 つの指標を用いた再帰的問合せ手順を提案した。

最後に、国土地理院発行の数値地図を用いて、新しい指標のサイズと処理量、関連研究である階層型参照点埋込み法との見積り距離に対する精度比較、刈込みにおける先行研究提案手法の比較、DRQ に対して階層型参照点埋込み法と提案手法の比較などを行い、提案手法の有効性を確認した。

今後の課題として、空間索引木の並列生成法や複数の探索対象オブジェクトに対する複合問合せの処理方法の検討があげられる。提案手法は  $k$ NN に対しても同様の手順で実行できると考えられるので、詳細に検討を行いたい。

謝辞 本研究の一部は、平成 23 年度関西大学重点領域研究助成ならびに先端科学技術推進機構 (社会基盤情報学応用研究グループ) による支援を受けた。ここに記して感謝する。

参考文献

- [1] Kriegel, H.-P., Kroger, P., Renz, M. and Schmidt, T.: Hierarchical Graph Embedding for Efficient Query Processing in Very Large Traffic Networks, *Proc. 20th International Conference on Scientific and Statistical Database Management*, pp.150-167 (2008).
- [2] Papadias, D., Zhang, J., Mamoulis, N. and Tao, Y.: Query Processing in Spatial Network Databases, *Proc. 29th International Conference on Very Large Data Bases*, pp.802-813 (2003).
- [3] Hu, H., Lee, D.L. and Lee, V.C.: Distance Indexing on Road Networks, *Proc. 32nd International Conference on Very Large Data Bases*, pp.894-905 (2006).
- [4] Kriegel, H.-P., Kroger, P., Kunath, P., Renz, M. and Schmidt, T.: Proximity Queries in Large Traffic Networks, *Proc. 15th ACM International Symposium on Advances in Geographic Information Systems*, pp.21:1-21:8 (2007).
- [5] Hjaltason, G.R. and Samet, H.: Properties of Embedding Methods for Similarity Searching in Metric Spaces, *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol.25, No.5, pp.530-549 (2003).
- [6] Kolahdouzan, M. and Shahabi, C.: Voronoi-based K Nearest Neighbor Search for Spatial Network Databases, *Proc. 30th International Conference on Very Large Data Bases*, pp.849-851 (2004).
- [7] Cheema, M.A., Lin, X., Zhang, Y., Wang, W. and Zhang, W.: Lazy Updates: An Efficient Technique to Continuously Monitoring Reverse kNN, *Proc. 35th International Conference on Very Large Data Bases*, pp.1138-1149 (2009).
- [8] 蒲原智也, 上島紳一: 道路網応用のための空間索引木の提案と最短経路探索への応用, *情報処理学会論文誌データベース*, Vol.2, No.2, pp.10-28 (2009).
- [9] Okabe, A., Boots, B., Sugihara, K. and Chin, S.N.: *Spatial Tessellations: Concepts and Applications of Voronoi Diagrams*, pp.218-224 (2000).
- [10] Erwig, M.: The Graph Voronoi Diagram with Applications, *Networks*, Vol.36, pp.156-163 (2000).
- [11] Guttman, A.: R-Trees: A Dynamic Index Structure for Spatial Searching, *Proc. 1984 ACM SIGMOD International Conference on Management of Data*, pp.47-57 (1984).



上島 紳一 (正会員)

昭和 30 年生。昭和 53 年京都大学工学部数理工学科卒業。工学博士。現在、関西大学総合情報学部教授。マルチメディア情報システム、柔軟な情報ベースに関する研究に従事。電子情報通信学会、ACM、IEEE 等の会員。

(担当編集委員 上善 恒雄)



蒲原 智也 (学生会員)

昭和 57 年生。平成 18 年阪南大学経営情報学部卒業。平成 20 年関西大学総合情報学研究科博士前期課程修了。現在、博士後期課程在学中。経路探索手法に関する研究に従事。日本データベース学会学生会員。