

ネットワークボロノイ図を用いたグラフ分割木の提案と最短経路探索への適用

Proposal of Graph-partitioning Tree using Network Voronoi Diagram and its Application to Shortest Path Problems

蒲原 智也[▼]

上島 紳一[◆]

Tomoya KAMBARA

Shinichi UESHIMA

本稿では、道路網をエッジ重み付きグラフと捉え、ネットワークボロノイ図を用いたグラフ分割木の生成法とそれを用いた経路探索法を提案する。提案手法では、道路網を表すグラフに対して、ネットワークボロノイ分割を行うことで、部分グラフに分割し、全体領域を部分領域に分割する。次に、各部分グラフをノードと見て、隣接する部分グラフ同士を融合することにより、各領域を拡張しながら、より広い部分領域のグラフを構成する。これにより、原地図と構造化情報として領域の隣接関係と包含関係を持つ階層型グラフ分割木を生成する。最後に、提案グラフ分割木に、再帰的な経路探索アルゴリズムを与え、その有効性を確認するため、国土地理院数値地図の道路網データに実際に適用して、提案アルゴリズムを用いたグラフ分割木の定量的な評価を行う。

The authors propose a new balanced space partitioning tree utilizing Network Voronoi Diagram and layering procedures, by considering road maps as planar graphs with edge costs. In our approach, we partition a given graph and generate subgraphs, which represents subregions of the target space. Next, setting each subgraph as a node, we merge with adjacent subgraphs, and extend the subgraphs to represent a wider subregion. By repeating these two steps successively, we generate a balanced space partitioning tree for original maps and structuring information for region adjacency and inclusion relation independently of road density spread over the space. Next, using the proposed tree, we provide a reflexive shortest path search algorithm. We perform numerical simulation to investigate the geometric characteristics of graph partitioning tree, and verify the efficiency of search method.

[▼] 学生会員 関西大学大学院総合情報学研究科
fa8d001@edu.kansai-u.ac.jp

[◆] 正会員 関西大学大学院総合情報学研究科
ueshima@edu.kansai-u.ac.jp

1. はじめに

近年、道路網を利用した応用のための経路探索、 k -最近傍探索問題などに用いられる有効なデータ構造の研究が注目を集めている [1, 2]. 本稿では、道路網において交差点をノード、交差点間をつなぐ道路をエッジとしたエッジ重み付き平面グラフとみなし、道路網を分割して得られる部分グラフ群をノードとして階層化した木構造を提案する。部分グラフの生成には、2種類のネットワークボロノイ図を用い [3, 4, 5], 更にランダムに選択した隣接領域間を統合することで、木の上位ノードを逐次的に生成して、グラフ分割木をボトムアップに生成する。提案木構造は、高さ一定やノード度数に関して平衡木の性質を持ち、ノードの上位・下位関係が領域の包含関係を持つ。つまり、より上位のノードが表す領域が、より広い範囲の領域を表す特徴がある。提案手法は、エッジの重みに応じた平面グラフの分割手法と、ノードの選択によるグラフの階層化手法の組合せであり、ノード間の位置関係と距離関係を保持した複数の詳細度別のグラフから成る階層型データ構造となっている。

提案手法の有効性を示すため、国土地理院数値地図(空間データ基盤, 1/2500 縮尺) [6] を用いて、道路網上の最短経路探索問題に提案木構造を適用する。それにより、道路網上の経路探索において、詳細に探索しなければならない領域は、出発点、目的地点を含む領域となり、計算時間が短縮されるとともに、任意の2地点間の探索において、ほぼ均等なノードの展開数と計算時間で最短経路を得ることができる。また提案木構造の生成法は、漸進的に平衡型木構造を構築する手法であり、領域分割と領域統合を繰り返すことで、道路網に対する領域間の位置関係と距離を計算していることに相当する。提案データ構造は、道路網を基盤としているため、様々な道路応用問題に利用できるものと考えられる。

2. 研究動機と関連研究

[動機と要件] 地図管理や平面データ管理の分野では、4分木、 R 木、領域2分木 [1] などを用いたデータ管理手法がよく知られている。これらの平面分割木は、対象領域を再帰的に分割して作成するため、部分平面の中にある道路やオブジェクトなどの存在状況を反映しない構成となっている。そのため、それらの存在状況によっては、これらの分割木にはデータ構造の操作に不要なノードや部分木が存在し、道路応用を考慮したデータ管理においては重要な問題となっている。

本稿では、上記の観点を整理して、道路網応用を考慮したデータ構造のために次の要件を考える。

- 領域分割木が平衡型である。つまり、木構造においてノード、度数がほぼ一定であり、各ノードが表す部分領域内の道路片数がほぼ一定である。
- ノードの上位・下位の関係に応じて、領域の包含関係が満たされている。
- 道路網の更新に対して、更新が局所的に留まり、計算コストが小さい。

[関連研究] Shahabiら [7] は、ネットワークボロノイ図による領域分割を用いて、 k -NN問題の効率的計算法を提案している。同手法では、索引対象毎にグラフ領域分割が行われているが、複数の部分グラフ間で関連付けがなされておらず、階層化などの手法

を適用した構造となっていない。

また、平面上のデータに対する木構造の冗長性を除去する目的で、Eppstein ら [8] は、木構造上でオブジェクトの存在しないノードを削除して、データに到達するための冗長なパスを短縮した圧縮 4 分木を提案している。また、計算幾何の分野では、de Berg ら [11] は平面上の直線群を格納するデータ構造を提案しているが、必ずしも道路網応用を目的とするものではない。

これらに対して、提案手法では、領域の分割に、ネットワークポロノイ図を利用しながら、平面グラフに SkipList[9] の考え方を応用して、ノードのランダム選択と、領域分割・統合を組み合わせたボトムアップな階層化となっている。このため、道路網の密度に依存しない平衡木を構成でき、同時に、種々の平面分割木の持つ領域包含関係を考慮した階層構造を構成できる。また、構造生成時の補助データを用いて、グラフの探索時間の圧縮を達成することもできる。

提案木構造は、ネットワークポロノイ図を用いて平面グラフのノードの位置関係に基づいて近傍ノードを分割している。つまり、原道路ネットワークのエッジに付けられた距離情報をもとに、大まかに抽象化した地図を階層的に構成することに相当する。提案手法は、グラフを対象とした階層化手法であるが、連続平面のポロノイ図の階層化手法としては、CG の分野で LOD(Level Of Detail) 表現を可能にした詳細度別表示に用いられている [10]。

次章では、領域分割に用いるネットワークポロノイ図を定義し、提案木構造の基本的な考え方について述べる。

3. 提案グラフ分割木

3.1 ネットワークポロノイ図によるグラフ分割

本節では、平面グラフに対して、提案グラフ分割木の生成法の考え方について述べる。道路網において、交叉点をノード、道路片をエッジとみなした無向グラフ $G = (N, E)$ (N : ノード集合, $E \subseteq N \times N$: エッジ集合) を考える。更に、各エッジは重みを持ち、各ノード n は 2 次元での座標 $(n.x, n.y)$ を持つものとする。図 1 では、ノードを P_i , エッジを E_i で示し、各エッジの重みを括弧内に示す。例えば、 $P_1 - P_{10}$ 間は、 E_5 で接続され重み 14.1 を持つ。

G に対して G のノードの部分集合を母点集合として与えると、次の 2 種類のネットワークポロノイ図を定義できる [3, 4, 5]。
 [ノードネットワークポロノイ図] (以後、ノード NV 図) ノード NV 図は、母点以外の各ノードに対して最も近い母点 (ノードの所属母点という) を決定し、母点に関して G の全ノード集合を分割した図をいう。

図 1 で、母点集合は、 P_1, P_2, P_3, P_4, P_5 である。 P_1 を所属母点とするノードは、 P_{10}, P_7, P_6 である。特に P_6 について見ると、母点 P_1 に重み 25 で最も近く、他の母点 P_2 へ重み 54.6, P_3 へ重み 71.4, P_4 へ重み 83.4, P_5 へ重み 36 となっていることが確認できる。他のノードに対しても同様である。また、2 つ以上の母点に等距離である場合については、所属母点は一意に決定できないため、ここではどちらか一方の母点に所属するものとして扱うものとする。

[エッジネットワークポロノイ図] (以後、エッジ NV 図) G のエッジに関しても、エッジ上の任意の地点から各母点への重みを考えれば、各地点の所属母点を決定することができる。この場合、1 つのエッジすべての地点が同一母点に所属する場合と、1 つのエッジの途中で所属母点が変わる場合がある。途中でエッジ

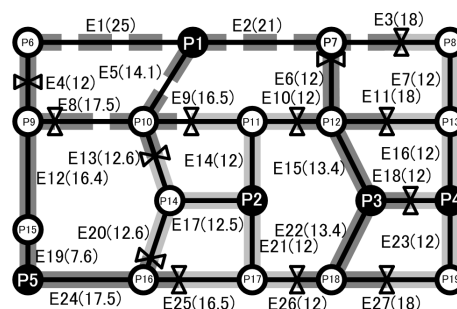


図 1 母点集合 P_1, \dots, P_5 に対するネットワークポロノイ図
 Fig. 1 Network voronoi diagram for generators P_1, \dots, P_5

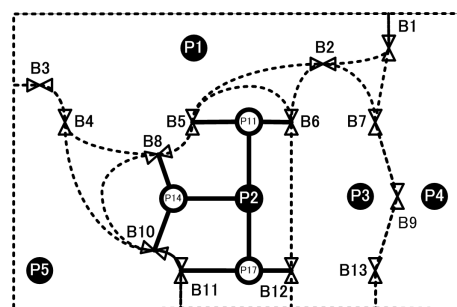


図 2 P_1, \dots, P_5 の母点領域の隣接関係図
 Fig. 2 Adjacency among generator regions for P_1, \dots, P_5

の所属母点が変わる地点を境界点といい、そのようなエッジを境界エッジという。図 1 では、 \times が境界点を示す。 G のすべてエッジ集合をエッジの所属母点に関して分割して得られた図をエッジ NV 図という。

図 1 で、 E_{12} 上の任意の地点でも最も近い母点は、 P_5 である。従って、 E_{12} は母点に所属する。境界点を持つエッジは、途中のどこかで所属母点が変わるエッジである。例えば、 E_4 の両端のノード P_6, P_9 の所属母点とその重みをみると、 P_6 は重み 25 で P_1 に所属し、 P_9 は重み 24 で P_5 に所属する。所属する母点異なるため、 E_4 上に境界点があることが確認できる。このときの境界点の位置は、 P_6 から P_9 までの E_4 の重み 5.5 の地点となっている。よって、 E_4 上の境界点は、母点 P_1 と P_5 より重み 30.5 の地点である。

[母点領域] 母点とその母点に所属するノードとエッジを集めてできる領域を母点領域と呼ぶ。つまり、各母点領域は、(i) 1 つの母点、(ii) 母点に所属するノード、(iii) 母点に所属するエッジ、(iv) 領域の端を示す境界点、(v) 境界点をもつエッジの境界点までの部分から構成される。ここで、母点領域間は、それぞれ有限個の境界点で、他の母点領域に隣接していることに注意する。

図 2 は、母点領域の隣接関係を示す。 P_2 の母点領域は、境界点 B_5, B_8 で P_1 に隣接し、境界点 B_6, B_{12} で P_3 に隣接し、境界点 B_{10}, B_{11} で P_5 の母点領域に隣接している。

各母点領域において、母点は、その母点領域の中心の役割を持ち、母点領域を代表する。提案するグラフ分割木においては、各母点が木の上位ノードを構成する。

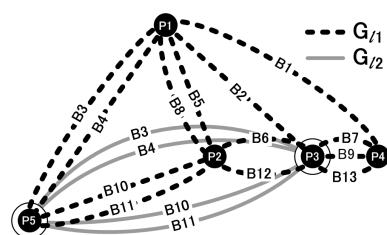


図3 母点領域の隣接関係を示すグラフ
Fig. 3 Graph of adjacency among generator regions

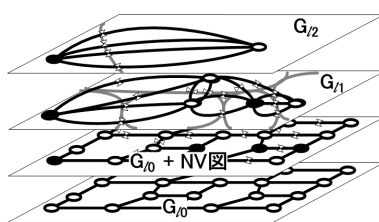


図4 グラフ分割木の階層構造
Fig. 4 Hierarchical structure of graph-partitioning tree

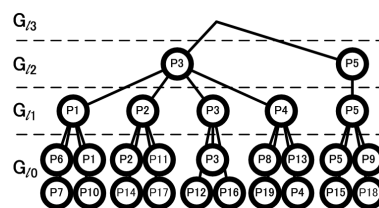


図5 領域の包含関係を示す木構造表現
Fig. 5 Tree representation for inclusion relations of region

3.2 領域の統合による木の生成

母点領域を単位と考えて1つのノードとみなし、互いに隣接する2つの母点領域間の境界点を接続部分として、2つの領域を統合することを考える。この統合により、より広い領域を生成する。ここでは、 G に対して、新たに母点集合を与えてノード NV 図を生成し、直上の階層を構成する手順から説明する。

[ノードと領域の同一視] 図2の領域を代表する母点 $P1..P5$ とその隣接関係のみを示した図が図3である。図2で、母点領域間の隣接関係は、境界点によりわかる。つまり、母点領域 $P1$ と母点領域 $P5$ は、境界点 $B3, B4$ を通して結ばれ、同様に、母点領域 $P1$ と母点領域 $P2$ は、境界点 $B5, B8$ を通して結ばれている。

この隣接する2つの母点領域間の境界点を接続部分として、ノードの隣接関係を結ぶエッジを構成することができる境界点から2つの母点までの重みは、それぞれ等しいので、このエッジの重みを境界点から母点への重みの2倍として与えて、 G_{11} に示すグラフを定義することができる。 G_{11} は、母点領域の隣接関係を示し、隣接領域間の代表点である母点間の重みを与えている。これを図3の点線で示し、このグラフを G_{11} とする。母点間を結ぶ経路は複数存在するため、 G_{11} は多重グラフとなっている。

[母点領域の統合] G_{11} において、各ノードは領域を示すため、隣接する領域に相当する2つ以上のノードを統合すれば、より広い領域を示すノードを生成することができる。そこで、 G_{11} に対してノードの部分集合を新たに母点集合として与えると、 G_{11} に対して、ノード NV 図を生成して、ノード集合を分割ことができ、このノード NV 図により、 G_{11} における新たな母点集合に対するノードの被所属関係を決定できる。

これにより、 G_{11} において、各ノードが領域を示すため、隣接する領域に相当する2つ以上のノードを統合すれば、より広い領域を示すノードを生成することができる。従って、上位のノードが示す領域が、下位のノードが示す複数の領域を完全に包含する木構造を生成することができる。

図3で、 G_{11} の全ノード集合 $P1, P2, P3, P4, P5$ の部分集合として、新たな母点集合として $P3, P5$ を与えたと仮定する。既に、 G_{11} では、母点間の経路の長さが求められているため、ノード NV 図を生成することができる。

実際に計算すると、ノードの被所属関係により、ノード NV 図を生成できる。 G_{11} のノード $P1, P2, P4$ は母点 $P3$ に所属し、母点領域は、そのままである。これにより、母点領域 $P1, P2, P3$ は、母点領域 $P5$ に統合される(図3の G_{12})。

この時点で、 G_{12} は、2つのノード $P3, P5$ を持つ。 $P3, P5$ を

結ぶ経路は、境界点 $B3, B4, B10, B11$ を通ることがわかる。

同様の手順を踏むことにより、より上位の層を逐次的に生成できる。このように、グラフを再帰的に統合することで、グラフ分割木の階層構造を得ることができる。

[階層構造] 図4に階層毎に作成されたグラフを示す。 G_{10} (原地図) から G_{11} を生成するときの母点のランダム選択をおこない、ネットワークボロノイ分割により、境界点が発生している。 G_{11} は、分割された領域間の経路を示し、これが上位層を生成する場合、分割対象となるグラフである。 G_{12} は、 G_{11} を上位層を生成した結果、得られたグラフである。

[母点による木構造] 図5に、図4での母点領域の統合関係を表す提案木構造の構造化データの部分を示す。同図の G_{10} と G_{11} の部分についてみると、図1での所属母点に応じて、親ノードと子ノードとなっている。 G_{11} と G_{12} についても同様に言える。

4. グラフ分割木の生成と利用

4.1 グラフ分割木の生成手順

図6にグラフ分割木の生成手順を示す。以下、 $Lv.i$ は提案階層構造の第 i 層を示す。 $Lv.0$ から $Lv.1$ つまり G_{10} から G_{11} を生成する場合、 G_{10} 上のノードすべてより、ランダムに母点を選択し、ノードネットワークボロノイ分割、エッジネットワークボロノイ分割を行う。このとき算出された各母点領域において、母点と境界点の間をつなぐ経路(母点-境界点間経路)と、母点領域の端にある複数の境界点の中から2つの境界点の組み合わせを行い、間をつなぐ経路(境界点間経路)を算出する。

$Lv.i-1$ から $Lv.i$ は、ランダムに母点を選択し、ノードネットワークボロノイ分割を行い各ノードの母点所属を決定する。これにより、各ノードの合併する領域は、一意に決定する。さらに、このノードは、下位層では、部分グラフである領域であるので、さらに領域同士の合併処理が必要である。

合併処理は、母点-境界点間経路の算出と境界点間経路の算出からなる。

[ノード・エッジ・境界点の保持データ] 原地図には、ノードとエッジは、それぞれ図7に示すデータを保持している。原地図に対して、提案手法により図7の構造化情報として生成することができる。Node の情報は、ノード NV 図によって生成できる。Boundary は、エッジ NV 図から、Edge については、3.2節 [ノードと領域の同一視] に示す方法で生成することができる。図8は、図7を図示したものである。Boundary と Edge の一部が重複しているが、Edge 側は、境界点と母点間の経路を詳細

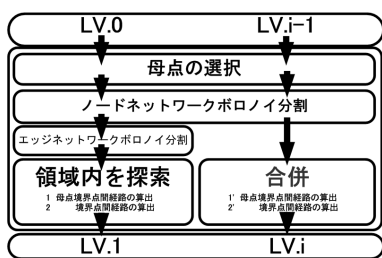


図6 グラフ分割木の生成手順

Fig. 6 Generation procedure of graph-partitioning tree

構造化情報	Node	NodeID 母点nodeID 母点への 一步先のnodeID 母点への距離 LV	Edge	EdgeID 母点nodeID-境界点ID or 境界点ID-境界点ID 距離 所属母点 LV
	Boundary	境界点ID 母点nodeID1 母点nodeID2 母点への距離 LV		
原地図	Node	NodeID X座標 Y座標	Edge	NodeID1 NodeID2 距離

図7 ノード・エッジ・境界点の保持データ

Fig. 7 Attributes of nodes, edges, boundaries (1)

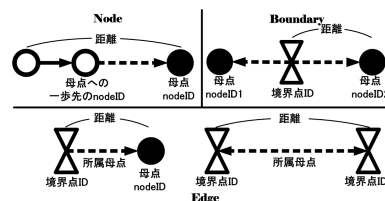


図8 構造化情報のデータの図

Fig. 8 Attributes of nodes, edges, boundaries (2)

に保持し、Boundary 側は、母点の隣接関係を示すためである。Edge の所属母点とは、その Edge がどの母点領域内に存在するかを示している。

4.1.1 第1層のグラフ生成

ノード集合 N より、ネットワークポロノイ分割における母点の選択を各ノードにおいて、確率 p で選択する。

1.[母点-境界点間経路の算出] エッジネットワークポロノイ分割において、各境界点に対して母点までの距離は算出されているので、境界点と母点間の距離を再度求める必要はなく、境界点を持つノードから母点への経路をノードネットワークポロノイ分割で求められた経路をたどる。図9で B_5 から P_2 の経路は B_5 を持つノード P_{11} から P_2 までの経路である。 P_2 からダイクストラ展開され構成された木の親ノードをたどり、 P_{11} から P_2 までの経路を構成する。

2.[境界点間経路の算出] 境界点を持つノードから同じ母点に所属するノードに対して、最短経路を求める。図10で境界点 B_{12} より $B_{12}-B_5$ 間の B_{12} を持つ P_{17} と B_5 を持つ P_{11} 間の経路を求めればよい。したがって、 P_{17} から P_{11} にダイクストラ法で経路を求める。同様に B_6, B_8, B_{10}, B_{11} に対しても境界点間の経路を求める。

4.1.2 第2層以上の上位層(第*i*層)のグラフ生成

図6での合併においては、次の二つの処理を行う。つまり、母点-境界点間経路の算出(1')と境界点間経路の算出(2')である。
1'. [母点-境界点間経路の算出] 各境界点から母点までの経路を算出する。図11で P_3 を主とし P_2 が合併するときについて述べる。 B_5, B_8, B_{10}, B_{11} が新たに P_3 の領域に追加され、 B_2, B_7, B_9, B_{13} は、そのまま維持し、 B_6, B_{12} は、境界点ではなくなる。新規境界点 B_5 についてみると、母点 P_3 までの経路を求めるには、領域 P_2 と P_3 の境界点 B_6, B_{12} を取り出し、それらを経由する母点 P_3 までの最小の距離をもつ経路を採用する。この場合、直下層で求めた経路 B_5-B_6 と、経路 B_6-P_3 をあわせて経路 $B_5-B_6-P_3$ とし、同様に経路 $B_5-B_{12}-P_3$ も生成し、この二つの経路長を比較して、前者を採用している。

合併において、 P_3 が直下層で持つ所属母点が変わらない境界点 B_2, B_{13} などについては、新たに母点-境界点間経路を計算せず、そのまま当該層での境界点とする。

2'. [境界点間経路の算出] すべての境界点間の経路を求める必要がある。主たる領域 P_3 の新規境界点 B_5, B_8, B_{10}, B_{11} から、 P_3 が合併前に元々もつ境界点 B_2, B_7, B_9, B_{13} への経路を出す計算が必要である。

新規境界点 B_5, B_8, B_{10}, B_{11} から、主たる領域 P_3 と主でない領域 P_2 がとも持つ境界点 B_6, B_{12} を経由し、 P_3 が合併前に持つ境界点 B_2, B_7, B_9, B_{13} をつないで、それぞれの経路長を比較して、最短の経路を求める。図13は、図12での B_5 と B_7 をつなぐエッジの処理を示している¹。

直下層と当該層で同じ境界点の組み合わせ間の経路は計算しなくてよい。図12で、左の境界点 B_5, B_8, B_{10}, B_{11} 相互の境界点間経路(左の破線部)と、右の境界点 B_2, B_7, B_9, B_{13} 相互の境界点間経路(右の破線部)にあたる。合併により得られた領域 P_3 の境界点間 B_5, B_8, B_{10}, B_{11} について計算する。

これら上記の処理は新たに合併された領域を上位レイヤーで1つの領域として扱うための処理・準備である。また3つ以上の領域が合併する場合は、母点となる母点領域に接している領域から、順に母点領域を拡大していく。

4.2 グラフ分割木を用いた最短経路探索

提案するグラフ分割木を用いた最短経路探索は、大きく分けて4つの段階に分けることができる。

1.[始点と終点を共に含む母点領域の決定] グラフ分割木 T において、始点と終点を共に同一となるノード(母点領域)を根から順に下方に探索する。これにより、 T の探索するグラフ分割木の枝を限定する。

図14は、 G_{i4} 層のあるグラフが指定した始点と終点を共に含む領域 V_0 であることを示す。

2.[始点終点の領域以外の探索] V_0 は G_{i3} で複数のノード(母点領域) V_1, V_2, V_3 から構成されている。その内、始点と終点を含む母点領域 V_1, V_3 以外の部分 V_2 を合併する。図14では V_2 の領域は、一つしか存在しないが複数ある場合でも一意に合併することができる。このとき境界点間経路の中でも始点の母点領域の境界点から、終点の母点領域の境界点に結ぶ複数の経路を候補経路と呼ぶ。

3.[逐次的差分探索] レベルを一つ下げ、始点と終点を含む母点領域のサイズを縮小し、始点と終点を含まない母点領域の部分を拡大する。上位レベルと下位レベルでの、始点と終点を含む母点領域の差の部分(図14 G_{i2} 濃灰線)を上位レベルで合併した領域(図14 G_{i2} 薄灰線)と合併させ、下位レベルでの経路候補を生成する(差分探索)。逐次的にレベルを下げ、レベル1まで解経路候補を生成する。

¹ 図13では、境界点間経路について示しているが、母点-境界点間経路でも同様である。

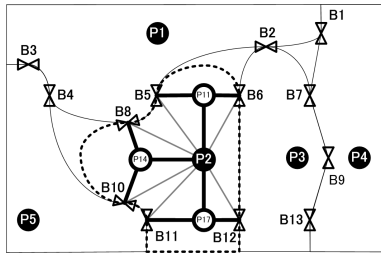


図 9 Level1: 母点-境界点間経路の算出
Fig. 9 Level1: Route generation between generator P2 and boundaries

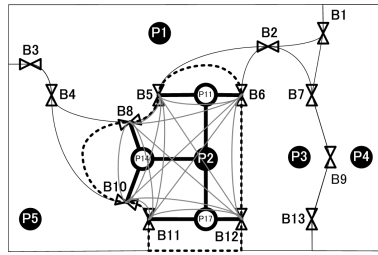


図 10 Level1: 境界点経路の算出
Fig. 10 Level1: Route generation between boundaries

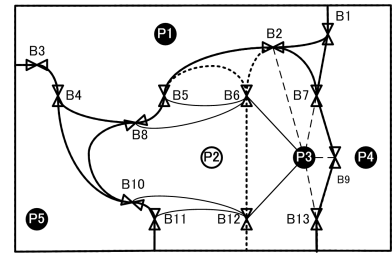


図 11 Level2: 母点領域の合併 (1')
Fig. 11 Level2: Region merge route generation from P3 to new boundaries

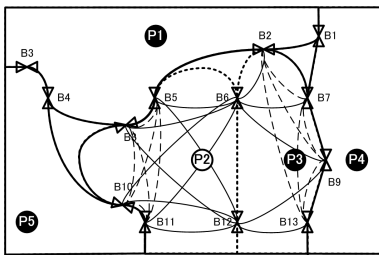


図 12 Level2: 母点領域の合併 (2')
Fig. 12 Level2: Region merge route generation among new combination of boundaries

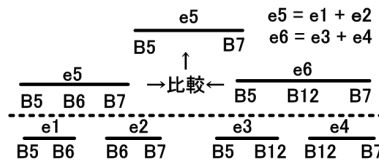


図 13 B5-B7をつなぐ経路の生成
Fig. 13 Compare costs routes in Lv.1 for generating a route in Lv.2

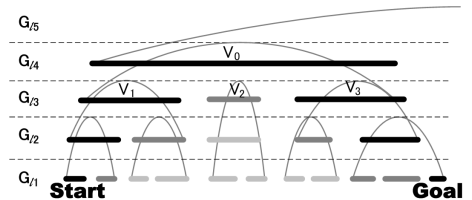


図 14 エッジの包含関係のみを抽出した木構造
Fig. 14 Tree for inclusion relations of regions

4. [G₁₀での探索] レベル1での経路候補の生成が終了したとき、始点と終点を含む母点領域の境界点まで、始点と終点から、G₁₀(原地図)上で経路探索を行う。(図14G₁₁黒線)

[結果の合成] G₁₀での探索の結果とレベル1での経路候補を合計し、その組み合わせの中から、重みが最小となる経路の組み合わせを選択することで、最短経路を求めることができる。

[例外] 始点終点のいずれかが母点となった場合、母点となった母点領域では、レベルを一つ下げ、母点領域を縮小させ、できた差分の部分を上位レベルでの経路候補と合併することはしない。

なぜならば、各母点領域は、母点-境界点間経路を所持し、この経路は、生成時の条件付けより最短であるからである。ともに母点となった場合、母点-境界点間経路と、間にある経路候補を合計し、重みが最小となる経路を最短経路とする。

4.3 提案グラフ分割木の特徴

前述の手順で得られたグラフ分割木は、以下の性質を持つ。

- 各層では、部分グラフ間の接続によって、領域の隣接関係を示している。
- 木におけるノードの接続関係が、上位の領域が下位の領域を含むことを示す。このため、R-木やR⁺木などに見られる領域の包含関係が成立している。従って、領域の持つ属性を領域から包含された領域へ継承させることが可能になり、地図応用などに有効である。
- また、原地図から、ノードを選択してボトムアップに木を生成しているため、平面上の道路の分布の歪みに依存せず、平衡木の性質を持つ(図5)。すなわち、木は、(i) 一定の高さ

O(log_p N), (ii) 根ノードから葉ノードまで同じ経路長, (iii) 各ノードが持つ枝数はほぼ一定, (iv) 葉ノードの部分グラフに含まれるノード数とエッジ数は一定などの性質を持つ。このため、任意の2地点間に対して、経路を探索する際、グラフ分割木における展開ノード数はほぼ均等となる。各ノードの持つ構造化データにより、経路探索の時間が大幅に短縮されるなどの利点が得られる。

5. 評価

5.1 グラフ分割木の性質

シミュレータによりグラフ分割木の生成処理について評価する。生成する原地図として国土地理院数値地図2500(空間データ基盤, 1/2500縮尺)の高槻市街中心部を利用する。利用したマシンは、WindowsXP, Xeon 3.72Ghz × 2, メモリ2GBである。すべてのノードが同一の領域に含まれる階層までの生成を1回とし、その試行を各確率において100回試行した。確率pを1/2から1/512まで変化させた。

生成時間: グラフ分割木の生成時間に関する確率pの影響を評価するため、生成時間を計測した。

計測結果の平均値と最大値を図15に示す。図より、平均生成時間は1612ノードに対して、500秒 당りに抑えられることがわかる。最大の生成時間を持つ、p=1/16のときの最大生成時間は、2055秒であった。この生成時間の変動は、各ノードにおいて、上位レベルでも存在するか否かを、確率で選択していること

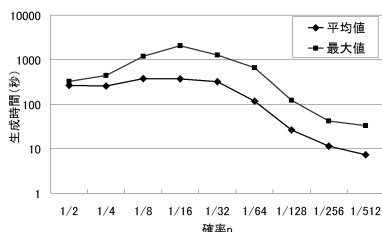


図 15 1612 ノードに対する確率 p における生成時間

Fig. 15 Generation time for graph-partitioning tree T vs p (1612 nodes)

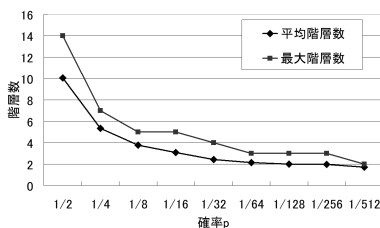


図 16 1612 ノードに対する確率 p における階層数

Fig. 16 Number of layers for Graph-partitioning Tree T vs p (1612 nodes)

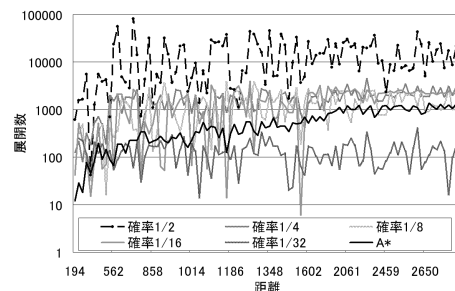


図 17 経路探索時の組み合わせの数

Fig. 17 Number of edges expanded for route generation proposed method A*algorithm

に由来すると考えられる。

グラフ分割木の高さ：グラフ分割木の階層数に関する確率 p の影響を評価するために、グラフ分割木の階層数を計測した。

計測結果の平均値と最大値を図 16 に示す。図から見て取れるように、階層数は、 $\log_p N$ に平均して収まっていることがわかる。また最大階層数も平均と比較しても大きく変化しない。

5.2 最短経路探索

確率 p を変化させ、提案グラフ分割木を一度ずつ構成する。構成できた木構造において原地図にあるノードからランダムに選択した始点と終点の組み合わせ 100 通り経路を求めた。組み合わせによる経路探索のために、確率 p の変動は $1/2$ から $1/32$ までとした。このとき、全ての階層で、始点の所属する母点の持つ境界点と、終点の所属する母点の境界点の組み合わせの合計を、与えられた 100 通りを距離順にならべたものを図 17 に示した。各確率において距離の長さにかかわらず、組み合わせの合計がそれほど変化しないことがわかる。A*経路探索アルゴリズムを利用して探索した場合と比較して、A*経路探索アルゴリズムでは、距離が遠くなるにつれ、展開ノード数が増大しているが、提案探索手法では増大しない。提案探索手法は長距離になるにつれ、A*探索アルゴリズムよりも効率的であると考えられる。

6. おわりに

本稿では、原地図に対してグラフ分割木の構成方法を提案し、最短経路探索への応用について議論した。提案手法は、境界点を結ぶエッジの包含関係と長さに基づいて生成される特徴を持つため、木の生成と最短経路探索が境界点エッジの合併操作に帰着されることを示した。同時にシミュレーションにおいて提案手法の評価を行った。今後の課題として、詳細な評価、グラフ分割木の並列生成法の検討、アドホックな環境を想定したグラフ分割木の生成と利用、最短経路探索以外の様々な局面での利用などが考えられる。

[文献]

- [1] Hanan Samet, "Foundations of Multidimensional And Metric Data Structures", Morgan Kaufmann (2006)
- [2] Jagan Sankaranarayanan, Houman Alborzi, and Hanan Samet, "Efficient Query Processing on Spatial Networks" ACM GIS'05, November 4 (2005)

- [3] M.Erwig, "The Graph Voronoi Diagram with Applications", Networks, 36(3), 156-163 (2000)
- [4] Atsuyuki Okabe, Barry Boots, Kokichi Sugihara, Sung Nok Chin, "Spatial Tessellations: Concepts and Applications of Voronoi Diagrams" "2nd edition", John Wiley & Sons Ltd, pp.218-224 (2000)
- [5] Margot Graf, Stephan Winter, "Network Voronoi Diagrams" Österreichische Zeitschrift für Vermessung und Geoinformation, 91(3), pp.166-174 (2003)
- [6] 国土地理院, 数値地図 (空間データベース), <http://sdf.gsi.go.jp/>
- [7] Mohammad Kolahdouzan, Cyrus Shahabi, "Voronoi-based K Nearest Neighbor Search for Spatial Network Databases" Proceeding of 20th VLDBconference Toronto, Canada 2004, pp.849-851 (2004)
- [8] David Eppstein, Michael T. Goodrich, Jonathan Z. Sun, "The Skip Quadtree: A Simple Dynamic Data Structure for Multidimensional Data", Proceedings of the 21st ACM Symposium on Computational Geometry 2005, pp.296-305 (2005)
- [9] William Pugh. "Skip lists: A probabilistic alternative to balanced trees." Communications of the ACM, 33(6), pp.668-676 (1990)
- [10] Christopher M. Gold, Paul Angel "Voronoi Hierarchies." GIScience 2006, pp.99-111 (2006)
- [11] Mark de Berg, Micha Streppel, "Approximate Range Searching Using Binary Space Partitions" Computational Geometry Vol.33 No.3, pp.139-151 (2006)

蒲原 智也 Tomoya KAMBARA

関西大学大学院総合情報学研究科在学中・日本データベース学会、ゲーム学会などの学生会員

上島 紳一 Shinichi UESHIMA

関西大学大学院総合情報学研究科教授・1983 京都大学大学院工学研究科博士課程単位取得退学、工学博士・マルチメディア情報システムの研究・開発に従事。情報処理学会、日本データベース学会などの会員