

空間検索を重視した幾何情報処理環境の開発

5B-7

林 英明†, 大澤 裕‡, 坂内正夫†

†東京大学生産技術研究所, ‡埼玉大学工学部

1. はじめに

地理情報処理や図面自動入力などの分野では、点や多角形、折れ線、円等様々な図形(以降、これらを図形データと呼ぶ)を、それらの位置関係を保存して管理する必要がある。処理に際して範囲検索や、最近点検索など、図形相互の位置関係に依存した検索が必要となるためである。筆者らは従来より、空間検索機能を重視した、幾何情報管理のためのデータ構造の研究を行っており、それを各種図形処理に応用して来ている。ここで問題となるのは、各応用毎にデータ型が異なるため、それらに合わせて演算も少しずつ変える必要があることである。しかし、図形処理で必要となる基本的な演算は似通っており、オブジェクト指向の枠組みで整理することにより、これらを共有できる。応用に依存した違いは、子のクラスとして導出することにより、容易に吸収できる。

本稿では、GBD木⁽¹⁾という多次元データ管理構造を核にして開発中の幾何情報処理のためのデータ管理方式について述べる。現在本システムを用いて、地理情報処理と図面自動入力の各システムを開発中であるが、ここでは主として地理情報処理への応用について述べる。

2. 図形情報の管理

本システムで管理される全ての図形データは、リスト(entity-list)に登録される。これは図形データを参入順に繋いだものであり、このリストで指し示される実体部には図形の属性情報や形状自身(頂点座標等)のデータが置かれる。GBD木の葉ノードには、このリスト上に存在する実体へのポイントが置かれている。このようにGBD木とentity-listの2つの構造に分けて管理する理由は、第1には図形データの実体が、図形種や頂点数等により可変長となることから、GBD木のノードような固定長のブロックでの管理には不向きであり、第2には応用分野により、全ての図形データにおいて必ずしも空間検索の機能が必要ないという理由によるものである。entity-listに登録されている図形の内、空間検索が必要なものがGBD木に登録される。システムへのデータの登録は、entity-listへ登録することによりなされ、逆にデータの抹消は、entity-listからその図形を外すことにより行なわれる。GBD木への登録や削除は自動的に行なわれる。

全ての図形データは、以下に示す情報を持つクラスentityから導出される。

(1) entityの種類 kind (2) 図形の中心座標点 xc, yc (3) entity-id

ここで(1)のkindはentityの種類を表すもので、各図形種毎に、また導出クラス毎にユニークな番号である。(2)の図形中心点座標(xc, yc)はGBD木で管理する際に必要となる付加情報である。また(3)のentity-idは個々のentityを区別するためのものであり、システムへの参入順に番号が付けられる。

このentityから、図形の基本クラス、点、線分、ポリゴン、ポリラインが導出される。次に、それら基本クラスから、ネットワークや領域のクラス等が導出され、更に順次応用毎に特化したクラスが導出される。例えば地理情報処理への応用では道路、河川、鉄道網、家屋など、応用別のクラスが必要となるが、オブジェクト指向の持つ利点により、これらのクラスの定義は、必要な差分のみで行なうことができる。

前述のように、全ての図形はentity-listに登録されて管理される。本システムを図形データベース管理システムとして見た場合、空間的な位置関係による空間検索の他、図形データの属性値(例えば道路名、道路幅員、建物名等)による検索も必要になる。従って、これらの検索を高速に実行するための構造も必要になるが、現在は図形データの処理に際して特に重要となるentity-idによる検索にたいしてのみ、ハッシングで対応している。

3. 地理情報処理への応用

地理情報システムでは、膨大な幾何情報を管理する必要があり、データ本体及びGBD木に要するデータ量も膨大なものになる。従って、entity-list、GBD木共にディスク上に置いている。本システムを中核にして、現在構築されている地理情報システムはまだ極小規模なものであり、実行可能な演算の種類も限られているが、鉄道網や、道路網等のネットワークデータを対象として、最短経路や、コスト最小経路を求める機能や領域決定問題を中心に機能拡張を行なっている。

An Impremtation of a Geometrical Information Environment for Spatial Retrievals

LIN Ying Ming†, Yutaka OHSAWA‡, Masao SAKAUCHI†

†University of Tokyo, ‡Saitama University

3. 1 図形種を限定した検索

地理情報システムへの応用では、道路、鉄道、建物、河川、行政界、設備ネットワーク等、属性の異なる様々な図形が対象データとなる。空間検索が必要な図形はGBD木に登録されるが、検索や種々の処理においてこれらの内一部のクラスのみを対象としたいことが多い。大量のデータの中で比較的少数のデータしか存在しないクラスを対象にした範囲検索を考えた場合、多数のノードを無駄に調べなければならない。

このような検索を効率良く実行するために、GBD木のノードに、そのノードより下位の部分木に含まれる図形の種類を表す情報を付与している⁽³⁾。具体的には、各ノードに固定長のビット列を置き、各ビットを図形の一つ一つの属性に対応させ、GBD木への登録時に下位の構造にその属性の図形が存在するノードは対応するビットの値を1にする。検索に際しては、探索パス上のノードでまずこのビット列を調べ、検索対象の図形に対応するビットが0であれば、そのノードを含めて下位に置かれている部分木をたどらない。これにより実際にアクセスされるノード数を低く押えている。

3. 2 ネットワークの表現

ネットワークはノード(net-node)とパス(poly-link)により構成される。ノードは点(point)を親として導出され、パスへのポイントを含んでいる。地理情報処理や図面入力等の応用で、パスは経路の形状(例えば道路)も合わせて表現できることが望まれるため、クラスpolylineへのポイントを持っている。以下に、net-nodeとpoly-linkの構造を示す。

<p>[net-node] 交差点 poly-linkへのポイント</p>	<p>[poly-link] 道路 from-node: リンクの始点ノード to-node: リンクの終点ノード shape: 形状を表すpolylineへのポイント</p>
---	--

これらを基本データとして構成した道路上の最短経路を求める応用例を図1に示す。図中の×印がネットワークのノード(道路交差点)に対応する。それ等を結んでいる線が道路である。これらが表現されている平面上で2点(P,Q)が指定された時、それらの点から最も近い交差点を探し、そこからスタートして、距離最小の経路を求める。道路の長さは単純にそれを表現している折れ線の長さで求めている。最短経路探索は次に行なわれる。まず、PとQに最も近い交差点CpとCqをGBD木上で最近点検索により求める。このCpからDijkstra法を適用することによりCqまでの最短経路を求めている。図1の太線が求めた最短経路である。

3. 3 領域決定

地理情報処理では、行政界や土地の使用用途界のように領域分けされた地図を扱うことも多い(図2)。このようなデータの表現としてDIME構造がある。領域境界線を交点から交点迄を結ぶ有向経路に分割し、各経路に対して、その始点Pと終点Q、及びPからQに至る時その右側の領域(right-polygon)Rと左側の領域(left-polygon)Lの情報を与える。DIME構造で表現された地図上で、任意の一点Gが与えられる時、Gが含まれる領域はGBD木上での最近接線分の探索により決定される⁽²⁾。

文 献

- (1)Y. Ohsawa, M. Sakauchi, "A New Type Data Structure with Homogeneous Nodes Suitable for a Very Large Spatial Database", Proc. of 6-th Int. Conf. on Data Eng., pp.296-303, Feb. 1990
- (2)大沢, 林, 坂内, "空間検索を重視した幾何情報処理環境", TV学技報, ICS92-35, pp.13-18, 1992.5
- (3)中村, 阿部, 大沢, 坂内, "多次元データの平衡木による管理-MD木の提案-", 信学論, J71-D, No.9, pp.1745-1752, 1988.9

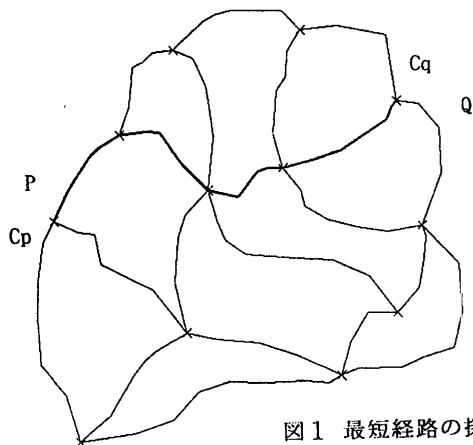


図1 最短経路の探索

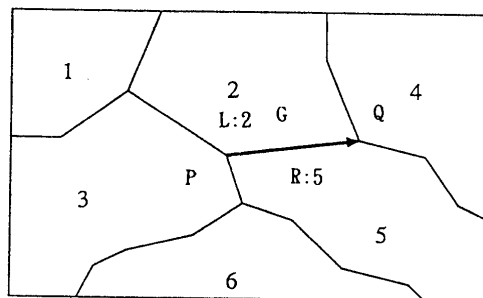


図2 領域の決定