

GISで用いるポリゴンデータのトポロジー修正システム

メタデータ	言語: Japanese 出版者: 独立行政法人農業工学研究所 公開日: 2024-02-28 キーワード (Ja): キーワード (En): 作成者: 飯嶋, 孝史, 石田, 憲治, 松森, 堅治, 嶺田, 拓也 メールアドレス: 所属:
URL	https://doi.org/10.24514/0002000055

GISで用いるポリゴンデータのトポロジー修正システム

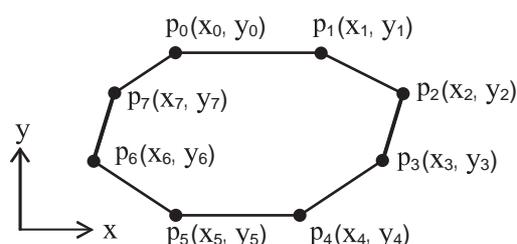
飯嶋孝史*・石田憲治*・松森堅治**・嶺田拓也*

目 次			
緒 言	61	頂点列方向判定処理	66
システムの概要	61	内点判定処理	67
自己交差多角形分割処理	62	結 言	68
交差多角形変換処理	64	参考文献	69
頂点列方向調整処理	66	Summary	70

緒 言

近年，さまざまな分野でGIS (Geographical Information System) の利用が普及しており，農業農村整備に係る計画策定や流域環境管理に関する調査，研究においてもGISが利用されることが多くなってきた。農業農村整備計画策定等に関するGISの利用にあたっては，行政区域，流域，植生，土地利用など，面的な広がりを持つ空間的要素を処理対象とすることが多いと考えられる。

面的な広がりを持つ空間的要素をGISで処理するには，その輪郭線を多角形とみなしたときの頂点座標値の連続としてとらえる方法 (ベクター方式) と，形と面積が等しい微小な単位区画の集合としてとらえる方法 (ラスター方式) がある。本報でいうポリゴンデータとは，面的な要素をベクター方式で処理するために，その輪郭線を近似的に表す



ポリゴンデータ (頂点座標の配列)
{ {X₀, Y₀}, {X₂, Y₂}, ..., {X₇, Y₇} }

Fig. 1 ポリゴンデータの例

Example of data structure describing a polygon

多角形の頂点座標値の集合を一定の規則に従った形式に整えたものである (Fig. 1)

ポリゴンデータによって表される空間的要素のうち，市区町村界などの汎用性の高いものはいくつかの機関・団体がデータを作成しており，入手も容易である。しかし，筆者らがこれまでに試用した既成のポリゴンデータの中には，システム上で領域の内外が正しく判別できないなど，GISソフトウェアで正常に処理できないものがあった。その原因は，平面領域の輪郭線を表すための頂点座標値のデータは揃っているものの，それらを結んでできる多角形のトポロジー^注の一部に，GISで正常に処理するための規則に従っていない欠陥が存在することにあつた。

GISソフトウェアの機能を活用するにはポリゴンデータが表す多角形のトポロジーが一定の規則に従っていることが前提条件である。データの作成者が厳密にその規則に従ったデータを作成・提供すれば問題はないのだが，デジタルイザ等を用いて手動操作により地図等から対象とする空間的要素の輪郭線の座標値を読み込んでデータを作成するような場合，細部において部分的な欠陥が生じる可能性は排除できない。また，多量のデータの中からそのような欠陥部分を検出して修正するには多大な労力を要する。

筆者らは，このような部分的な欠陥があるデータの有効利用を図るため，トポロジーの欠陥部分を自動的に修正し，GISで正常な処理ができるように変換するためのデータ処理システム (以下，「本システム」という) を開発した。本報は，本システムにおけるデータ処理方法について報告するものである。

システムの概要

1 ポリゴンデータのトポロジー規則

ポリゴンデータは，例えば市区町村の区域のような飛び地や穴のある平面領域を複数の多角形で近似的に表現している。そのデータ構造を一般化して言えば，個々の多角形をx座標値とy座標値のペアからなる頂点データの配列

注) トポロジー (topology) とは，本来は数学の一分野である位相幾何学のことであるが，地理情報処理の分野においては，位置的关系を含む空間的要素の幾何学的特性を意味する用語として用いられ，本報においてもその意味で用いた。

* 農村環境部環境評価研究室，** 農村環境部景域研究室

平成16年1月8日受理

キーワード：GIS，ポリゴン，トポロジー，データ処理

(以下「頂点列」という)で表し、複数の頂点列によって1個の平面領域データを構成したものである。そして、ポリゴンデータファイルは、異なる属性を持つ複数の平面領域データが格納されたものである。

ただし、ポリゴンデータファイルの具体的なフォーマットは標準化されておらず、GISソフトウェア開発者等によって定義されたものが複数存在し、また、GISソフトウェア内部の処理方法もブラックボックスである。そこで、本システムのデータ処理の要件としては、ESRI社が提唱し、データ構造の仕様が完全に公開されているShapeファイル形式のポリゴンデータが表す多角形群に求められるトポロジー規則を採用した。同規則は以下のとおりである(ESRI, 1998)。

- (1) 多角形が自己交差しないこと
- (2) ひとつの平面領域を構成する複数の多角形同士が交差しないこと(頂点で接しても良いが辺を共有してはならない)
- (3) 多角形の頂点を頂点列における配列順に辿ったときの方向の右側が平面領域の内側であること
- (4) 多角形が閉じていること。つまり、頂点列の始点と終点の座標値が同じであること

本システムは、ポリゴンデータファイルにおいて上記の規則に従わない欠陥部分があった場合にそれを自動的に修正し、欠陥のないファイルに変換する手段を提供する。

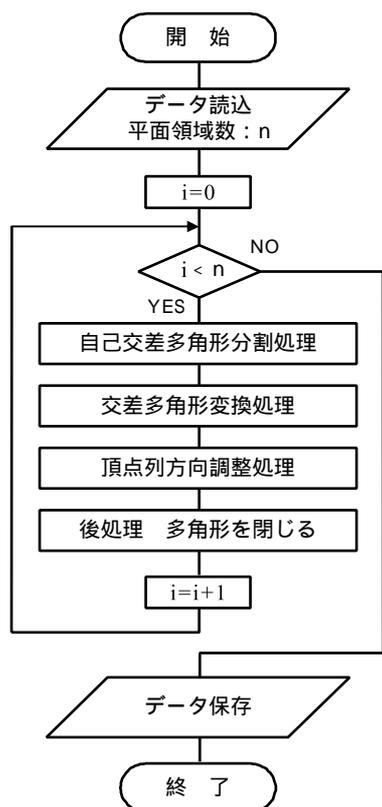


Fig.2 システム全体の処理フロー
Flow chart of the system

2 全体の処理フロー

Fig.2に本システムによるデータ処理の全体のフローを示す。本システムは、ファイルに格納されている平面領域データについて、当該平面領域データに含まれる多角形の内、自己交点、自己接点を有する多角形がある場合に、それらを自己交点等で分割することによって自己交差を解消し(以下「自己交差多角形分割処理」という)、当該平面領域データを構成する複数の多角形が交差(辺の一部を共有する場合を含む)する場合に、それらの多角形を結合するなどの変換を行うことによって交差を解消し(以下「交差多角形変換処理」という)、当該平面領域データを構成する多角形の頂点列の方向の右側が当該平面領域の内側となるように頂点列の方向を調整し(以下「頂点列方向調整処理」という)、後処理として、当該平面領域データを構成する多角形の頂点列の終点に始点と等しい頂点を追加することによって多角形を閉じる。

また、上記処理を行うにはいくつかのサブルーチンが必要であるが、中でも、頂点列の方向(時計回りか否か)を判定する処理(以下「頂点列方向判定処理」という)、任意の点が多角形の内部にあるか否かを判定する処理(以下「内点判定処理」という)は特に重要であり、かつ、汎用的な図形データ処理方法でもある。

以降で、上記 ~ の具体的なデータ処理方法について述べる。

3 汎用性について

本システムを構成するプログラムは、データ処理を行う部分と、記憶媒体からのデータ読み込み・記憶媒体へのデータ保存を行う部分を完全に切り離せるように構成した。また、データ処理部分では一般化したデータ構造(頂点列の配列で1個の平面領域を表す)を対象として処理を行うこととした。

以上のようにすることで、本システムの要件としてはShapeファイル形式ポリゴンデータのトポロジー規則を採用したが、システムのコアとなるデータ処理部分は特定のファイル形式に依存しない汎用的なものとなっている。

また、本システムを構成するプログラムのソースコードはC++言語(ISO/IEC 14882, JIS X 3014)によって記述したので、ソースコードレベルでの移植性、可読性が高く、また、カスタマイズも容易である。

自己交差多角形分割処理

1 概要

Fig.3(a)の多角形は、頂点 p_1 が辺 p_3p_4 と接しており、辺 p_3p_4 と辺 p_5p_0 が交差している。このように同一の多角形において頂点が辺と接している点を自己接点、辺同士が交差している場合の交点を自己交点と呼び、さらにそれら

を総称してカット点と呼ぶこととする。Fig. 3(a)では、カット点を記号 c_1, c_2 で示した。

まず、カット点 c_1, c_2 が存在する辺にそれらの座標値と等しい頂点を挿入する。すなわち、辺 p_3p_4 上には c_1 及び c_2 に等しい頂点、辺 p_5p_0 上に c_2 に等しい頂点の合計3個の頂点を挿入する。Fig. 3(b)は、そのように頂点を挿入した後に番号を振り直したものである。ここまでの処理を施した後の多角形には、1個のカット点につき、その座標値に等しい頂点が2個以上存在することとなる。

次に、この多角形について2個の頂点の座標値が等しいか否かを番号の小さい方から順番に調べると、まず、 p_1 と p_4 の座標値が等しいこと、すなわち、それらがカット点に等しいことが見出される。そこで、多角形を p_1 と p_4 によって2分割することにより、Fig. 3(c), (d)の2個の多角形が得られる。

さらに、Fig. 3(c)について、2個の頂点の組み合わせを調べると、 p_5 と p_8 の座標値が等しいことが見出されるので、当該頂点において多角形を2分割することにより、Fig. 3(e), (f)が得られる。

このように、本システムでは、自己交差する多角形にカット点に等しい頂点の挿入を行った後に、カット点の探索とカット点における多角形の2分割を繰り返すことにより、自己交差のない複数の多角形に分割する。

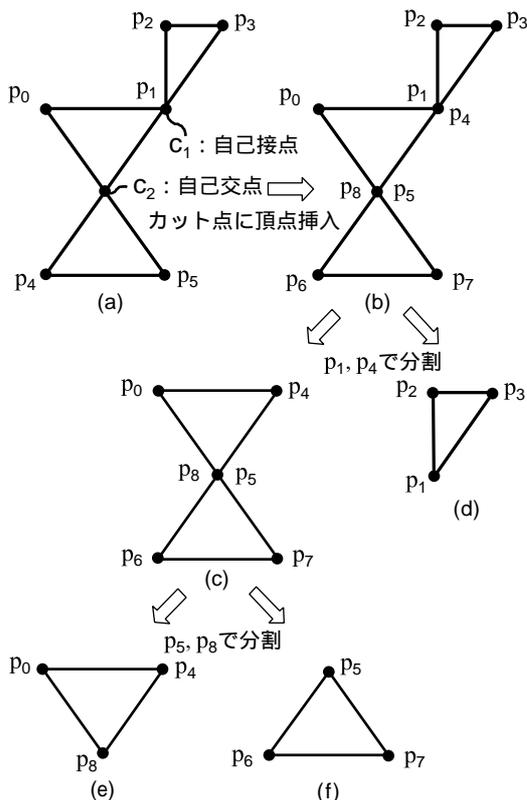


Fig. 3 自己交差多角形の分割方法
Method for dividing a self-intersected polygon

2 処理手順の詳細

(1) 前処理

処理の対象とする頂点列中の連続する複数の頂点（頂点列の終点と始点も連続しているものと見なす。以下同じ）の座標値が等しい場合、1個を残して他を削除する。これは、概要でも述べたように、処理手順中、カット点であるかどうかを2個の頂点の座標値が等しいか否かによって判定するため、連続する複数個の頂点の座標値が等しい場合にはカット点の判定が正しくできないからである。

(2) カット点の探索と頂点挿入

頂点列中の連続する2頂点を両端点とする線分（辺）の全ての組み合わせについて、当該2線分が交差するか否か、または一方の線分の端点が他方の線分上にあるか否かを調べることにより、カット点を探索する。線分上（両端点を除く）にカット点があった場合、そのカット点の座標値に等しい頂点データを頂点列に挿入する。

以上の処理により、多角形が自己交差する場合には、当該多角形を表す頂点列中にカット点に等しい頂点が2個以上存在することとなる。

(3) 自己交差多角形の分割

(2)の処理を施した後の頂点列を

$$V_0 = \{p_0, \dots, p_{i-1}, p_i, \dots, p_{j-1}, p_j, \dots, p_{n-1}\}$$

としたとき、当該頂点列中の2個の頂点の座標値が等しいか否かを番号の小さい方から順番に調べる。

その結果、 p_i と p_j の座標値が等しいこと、すなわち、それらがカット点に等しいことが見いだされた場合に、次の手順により V_0 を2分割する。

まず、 V_0 に基づいて、 p_0, p_i, p_j を始点とする3個の部分頂点列

$$V_1' = \{p_0, \dots, p_{i-1}\}$$

$$V_2 = \{p_i, \dots, p_{j-1}\}$$

$$V_1'' = \{p_j, \dots, p_{n-1}\}$$

を作成する。

次に、 V_1' と V_1'' を連結した頂点列

$$V_1 = \{p_0, \dots, p_{i-1}, p_j, \dots, p_{n-1}\}$$

を作成する。このようにして作成した V_1 と V_2 が、処理前の多角形を最初に見いだされたカット点において分割した2個の多角形を表す頂点列である。

ここで、Fig. 3の例のように、自己交差する多角形においてカット点は1カ所とは限らない。そのため、2分割後の頂点列 V_1, V_2 をそれぞれ V_0 に読み替えて、カット点がないことを終了条件として、上記の2分割処理を再帰的に繰り返す。これによって、処理前の多角形にカット点が複数あった場合でも、カット点のない複数の多角形に分割することができる。

なお、処理前の多角形において、2辺が同じ直線上で重なっている場合には、上記処理の過程において重なってい

る部分が分離され、その両端の2頂点だけからなる頂点列が作成される。これは面積のない線分であり、ポリゴンデータとしては不要なものなので排除する。

平面領域データを構成する全ての頂点列について上記(1)~(3)の処理を施し、カット点のある多角形の頂点列を分割後の複数の頂点列に更新することによって、当該平面領域を構成する全ての多角形の自己交差が解消される。

交差多角形変換処理

1 概要

Fig. 4(a)に示す2個の多角形A, Bについて、それらの頂点列の方向がどちらも時計回りであり、それらがFig. 4(b)のように交差している場合について考える。

まず、Fig. 4(b)に示すように、2個の多角形の交点 p_1, p_2, p_3, p_4 を求める。次に、任意の交点、例えば p_1 から出発し、Aの頂点列をその方向に従って順番に辿っていき、別の交点 p_4 に達したときにBの頂点列に乗り換えてBの頂点列を辿って p_1 まで戻ると、Fig. 4(c)に示すような多角形が得られる。

このように、ある交点を出発点としてAの頂点列を辿り、別の交点に達したらBの頂点列に乗り換える、というように、交点において2個の多角形の頂点列を乗り換えながら出発点に戻るまで頂点列を辿ることによって多角形を作成する手順を、全ての辺を一回ずつ辿り終わるまで行うことにより、交差する2個の多角形を、相互に交差しない複数の多角形に変換することができる。

変換後の多角形は、時計回りのもので面積が最大のものがAとBの領域を結合した領域(以下「結合領域」という)を囲む多角形(Fig. 4(c))、その他の時計回りのものがA

とBが重なり合う領域(以下「重複領域」という)を囲む多角形(Fig. 4(d))、反時計回りのものがAとBを結合してできた穴の領域(以下「穴領域」という)を囲む多角形(Fig. 4(e))に分類できる。

本システムでは、これらのうち、結合領域と穴領域を囲む多角形を変換後の多角形とする。もちろん、2個の多角形の位置関係によっては、穴領域は発生しないこともある。なお、変換前の多角形の頂点列の方向が反時計回りの場合には、変換後の多角形の分類における頂点列の方向をそれぞれ逆に読み替えれば同じ結果が得られる。

平面領域データが複数の多角形で構成されている場合には、多角形間の交差がないことを終了条件として上記変換を繰り返し行うことにより、交差しない複数の多角形に変換する。

2 処理手順の詳細

(1) 前処理

上記の方法により交差する2個の多角形を交差しない複数の多角形に変換するには、処理対象とする多角形の頂点列の方向が時計回りに統一されている必要がある。そのため、前処理として平面領域データを構成する全ての頂点列についてその方向を判定し、反時計回りである場合には頂点列における頂点データの格納順を反転させることにより、時計回りに統一する。頂点列の方向を判定する方法については 述べる。

(2) 2多角形間の交点等の探索と頂点挿入

平面領域データから2個の頂点列を取り出し、それらが表す2多角形間の辺の全ての組み合わせについて、交差するか否か、あるいは一方の辺の端点が他方の辺上にあるか否かを調べることにより、当該2個の多角形間の交点又は接点(以下「交点等」と総称する)を探索する。当該多角形において、辺上(両端の頂点上を除く)に交点等があった場合、当該交点等に等しい頂点を当該多角形を表す頂点列の該当力所に挿入する。

この処理を、平面領域データを構成する2個の頂点列の全ての組み合わせについて施すことにより、ある2個の多角形の組合せにおいて交点等がある場合、当該2個の多角形を表す頂点列の双方に必ず当該交点等に等しい頂点が存在することとなる。

(3) 2多角形の交差判定

以上の処理を施した平面領域データに含まれる2個の多角形間に交点等があるかどうかは、当該2個の多角形を表す頂点列の間に座標値が等しい頂点があるかどうかを調べることによって判定できる。

しかし、本システムの目的から、2個の多角形が頂点で接しているだけの場合は変換処理を行う必要はない。そのため、2個の頂点列間の頂点座標値の比較によって見出された交点等が交点と接点のいずれであるかを判別し、1個

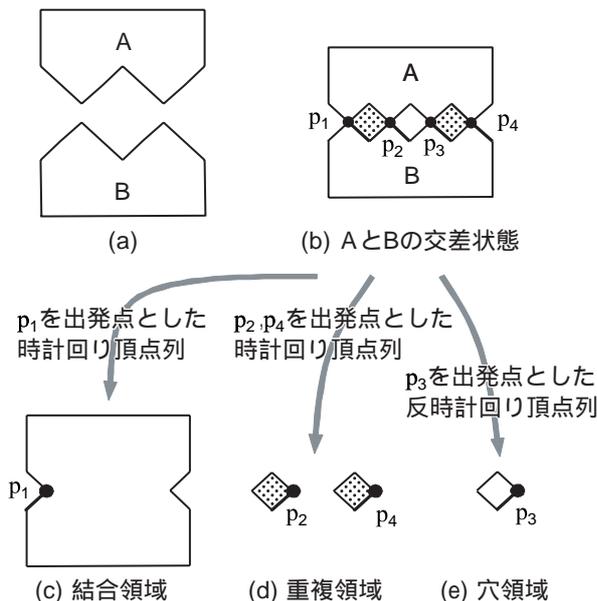


Fig. 4 交差する多角形の変換方法(1)

Method for transforming intersected polygons(1)

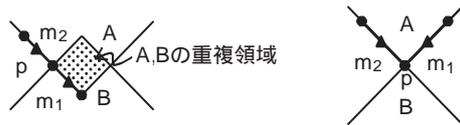
でも交点があった場合には当該2多角形が交差していると判定し、その場合のみ変換処理の対象とする。

2個の頂点列の間で座標値が等しい頂点があった場合、それが交点であるか接点であるかを判別する方法は次のとおりである。

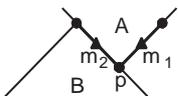
2個の多角形をA, Bとし、それらの間に交点等pがあった場合に、どちらか一方の頂点列（ここではAの頂点列とする）において、pに該当する頂点とその直前の頂点の間の辺上の任意の点 m_1 と、直後の頂点の間の辺上の任意の点 m_2 を設ける（実用上はそれぞれ中点とするなどの方法が簡便である）。Fig. 5(a)に示すように、 m_1 と m_2 のどちらか一方がBの内部、他方がBの外部にあればpはAとBの交点であり、Fig. 5(b)に示すように、 m_1 と m_2 の両方がBの外部か内部のいずれか一方にあれば、pはAとBの接点であると判定できる。

その他、想定されるケースとして、Fig. 5(c)に示すように、AとBの辺が重複し、 m_1 が m_2 のいずれかがBの辺上にある場合がある。本システムが採用したトポロジー規則により、同一の平面領域に属する多角形同士は辺を共有してはならないため、このケースも変換処理の対象とする。そのため、 m_1 、 m_2 の一方が多角形の周上にあり、他方が多角形の内部、あるいは外部にある場合もpはAとBの交点であると判定する。

m_1 、 m_2 が多角形の内部、外部、周上のいずれにあるかは、で述べる内点判定処理によって判定する。



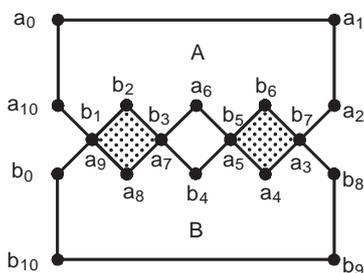
(a) 交点: m_1 がBの内部, m_2 が外部 (b) 接点: m_1, m_2 がBの外部



(c) 辺の重複: m_1 がBの外部, m_2 がBの辺上

Fig. 5 交点の判定方法

Method for deciding intersection



*網かけ部はAとBの重複領域を示す

Fig. 6 交差する多角形の変換方法(2)

Method for transforming intersected polygons(2)

(4) 交差する2多角形の変換

(1)及び(2)の処理後の2個の頂点列によって表される多角形が(3)の手順によって交差していると判定された場合に、当該2個の頂点列を、交差しない複数の多角形を表す頂点列に変換する。その手順について、Fig. 6の例を参照して述べる。

Fig. 6の多角形A, Bを表す頂点列は、それぞれ11個の頂点を持っており、次のように表す。

$$Va = \{a_0, a_1, a_2, a_3, a_4, a_5, a_6, a_7, a_8, a_9, a_{10}\}$$

$$Vb = \{b_0, b_1, b_2, b_3, b_4, b_5, b_6, b_7, b_8, b_9, b_{10}\}$$

ここで、 Va 、 Vb は上記(2)の処理後であり、双方の頂点列中には必ずAとBの交点等に等しい頂点があるので、双方の頂点座標値を比較することによって交点等を抽出できる。Fig. 6の例では、 Va と Vb の頂点の組合せのうち、 a_3 と b_7 、 a_5 と b_5 、 a_7 と b_3 、 a_9 と b_1 がそれぞれ等しいので、これらがAとBの交点等に等しい頂点として抽出される。

交点等に等しい頂点が抽出されたならば、頂点列 Va 、 Vb から、交点等に等しい頂点を始終点として、次のように部分頂点列の集合を作成する。

$$Sa = \{ Va_0 = \{a_3, a_4, a_5\},$$

$$Va_1 = \{a_5, a_6, a_7\},$$

$$Va_2 = \{a_7, a_8, a_9\},$$

$$Va_3 = \{a_9, a_{10}, a_0, a_1, a_2, a_3\} \}$$

$$Sb = \{ Vb_0 = \{b_1, b_2, b_3\},$$

$$Vb_1 = \{b_3, b_4, b_5\},$$

$$Vb_2 = \{b_5, b_6, b_7\},$$

$$Vb_3 = \{b_7, b_8, b_9, b_{10}, b_0, b_1\} \}$$

次に、 Sa の中から1個の要素、例えば Va_0 を選び、続いて Sb の中から、 Va_0 の終点 a_5 と等しい始点を持つ要素を選ぶ。ここでは $b_5 = a_5$ だから、 Vb_2 が選ばれる。このとき、 Vb_2 の終点 b_7 は、最初に選んだ Va_0 の始点 a_3 と等しいので要素の選択を終了し、 Va_0 と Vb_2 を、それぞれ終点を削除した上で連結することによって、頂点列

$$V_1 = \{a_3, a_4, b_5, b_6\}$$

を作成する。

以後、 Sa の中から未使用の要素を順次選んで同様の手順によって、

$$V_2 = \{a_5, a_6, b_3, b_4\}$$

$$V_3 = \{a_7, a_8, b_1, b_2\}$$

$$V_4 = \{a_9, a_{10}, a_0, a_1, a_2, b_7, b_8, b_9, b_{10}, b_0\}$$

が得られる。

このようにして得られた頂点列の方向は、で述べる頂点列方向判定処理によって V_1 、 V_3 、 V_4 が時計回り、 V_2 が反時計回り（穴領域を囲む多角形）であると判定でき、さらに、時計回りのものについては面積を比較することにより、 V_4 の面積が最大（結合領域を囲む多角形）であると判定できる。

以上の結果から、処理前の頂点列 V_a, V_b を V_4, V_2 に置き換えることによって、交差する2多角形 A, B が、それらの結合領域と穴領域を囲む多角形に変換される。

(5) 複数多角形の交差解消

平面領域データに含まれる複数の多角形の交差を全て解消するには、次の手順で処理する。

平面領域データに含まれる全ての頂点列について(1)前処理、及び、頂点列の全ての組み合わせについて(2)多角形間の交点等の探索と頂点挿入を施す。

(3)の交差判定方法を利用して平面領域データから交差する2個の頂点列を抽出し、それらを(4)の変換方法によって結合領域を囲む多角形等に変換し、平面領域データを更新する。

更新後の平面領域データについて、交差する頂点列の組み合わせがないことを終了条件として、の処理を再帰的に繰り返す。

頂点列方向調整処理

1 概要

平面領域を構成する複数の多角形が入れ子状になっている場合、最も外側の多角形の頂点列の方向を時計回りとし、内包される多角形の頂点列については、それぞれそのすぐ外側の多角形の頂点列の方向と逆回りになるように調整す

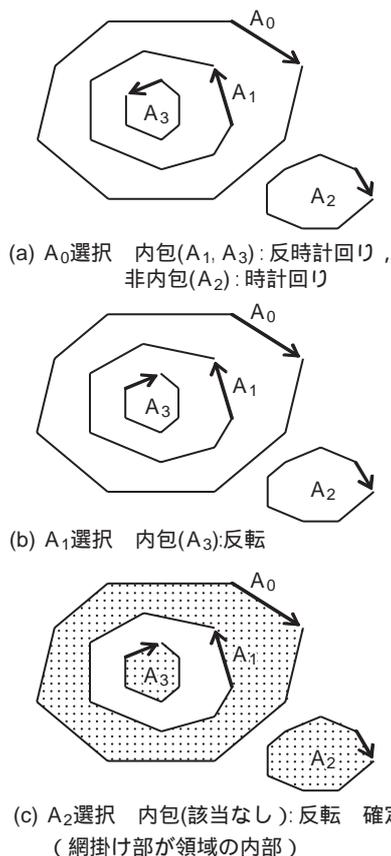


Fig. 7 頂点列方向の調整方法

Method for adjusting direction of polygon vertex arrays

る。これにより、全ての頂点列の方向の右側を平面領域の内側とすることができる。

2 処理手順の詳細

(1) 前処理

平面領域を構成する複数の多角形のうち、面積が最も大きい多角形 (A_0 とする) を表す頂点列の方向を時計回りとし、その他の多角形のうち、 A_0 に内包される多角形の頂点列を反時計回りに、内包されない多角形の頂点列を時計回りとする (Fig. 7(a))。

頂点列の方向は で述べる方法によって判定し、頂点列の方向を変更する必要がある場合には、当該頂点列における頂点データの格納順を反転させる。

また、多角形同士の内包関係の判定は、面積が小さい方の多角形の頂点が1個でも面積が大きい方の多角形の外部にあれば内包されないと判定し、そうでなければ、内包されると判定する。頂点が多角形の内部にあるか否かは、で述べる内点判定処理によって判定する。

(2) 頂点列方向調整

2番目に面積が大きい多角形の頂点列を選択し、それより面積が小さい多角形のうち、選択された多角形に内包されるものの頂点列の方向を反転させる (Fig. 7(b))。

以後、面積の降順により多角形を順次選択して同様の処理を行い、面積が2番目に小さい多角形を選択したときの処理が終了すると、全ての頂点列の方向が確定する (Fig. 7(c))。

頂点列方向判定処理

xy 座標平面上における多角形の頂点のうち、 x 座標値が最小または最大の頂点、あるいは、 y 座標値が最小または最大の頂点のうちいずれか1個を p_2 、当該頂点列における p_2 の直前の頂点を p_1 、直後の頂点を p_3 としたとき、式(1)により計算されるベクトル p_2p_1 とベクトル p_2p_3 の外積 G の正負によって、当該多角形の頂点列の方向を判定する方法が知られている。 x 軸の正の方向が右向き、 y 軸の正の方向が上向きの座標系においては、 G の値が正の場合は当該頂点列は時計回り、負の場合は反時計回りであると判定できる。

$$G = p_2p_1 \times p_2p_3 = (x_1 - x_2) \times (y_3 - y_2) - (y_1 - y_2) \times (x_3 - x_2) \dots (1)$$

x_n : 頂点 p_n の x 座標値 ($n=1, 2, 3$)

y_n : 頂点 p_n の y 座標値 ($n=1, 2, 3$)

ただし、この方法には、 G が0のときには判定できないという問題がある。

ここで、任意の点 p_1, p_2, p_3 が与えられた場合、式(1)の G が0になるのは、これらの3点が同一直線上にあ

る場合である。一般論としては、 p_1, p_2, p_3 のうちいずれか2点あるいは3点とも等しい場合も G は0になるが、 p_1, p_2, p_3 がで述べた自己交差多角形分割処理後の頂点列における連続する3頂点であるならば、それらはいずれも等しくなく、かつ、線分(辺) p_1p_2, p_2p_3 が重なることもない。

また、頂点列の方向を判定するうえで、 x 座標値が最小の頂点を p_2 とし、 p_1, p_3 をそれぞれ p_2 の直前、直後の頂点とした場合には、 p_2 の x 座標値は p_1 及び p_3 の x 座標値より大きくなることはないから、 G が0になる場合としては、 p_1, p_2, p_3 が y 軸に平行な直線上に並ぶ場合だけを考慮すれば良い。

以上の考察に基づき、本システムにおける頂点列の方向の判定は、以下の手順で行うこととした。

- (1) 頂点列の内 x 座標値が最小の頂点 p_2 と、その直前、直後の頂点 p_1, p_3 を選択する。
- (2) ベクトル p_2p_1 とベクトル p_2p_3 の外積 G を計算する。
- (3) $G > 0$ なら時計回り、 $G < 0$ なら反時計回りであると判定する。
- (4) $G = 0$ の場合(p_1, p_2, p_3 が y 軸に平行な直線上に並んでいる場合)には、 p_2 と p_3 の y 座標値 y_2 と y_3 を比較し、 $y_2 < y_3$ なら時計回り、 $y_2 > y_3$ なら反時計回りであると判定する。

内点判定処理

任意の点が多角形の内部にあるか否かを判定する方法として、鉛直線アルゴリズム(今井, 1994)を応用した内点判定法が知られている。この内点判定法は、判定の対象とする点(以下「判定対象点」という)を端点として任意の方向に半直線を引き、当該半直線が多角形の周と交差する回数を計数し、交差回数が偶数(0を含む)ならば当該判定対象点は多角形の外部、奇数ならば内部にあると判定するものである。例えば、Fig. 8において、点 p_a を端点とする半直線 L_a は2辺(辺 p_0p_1 及び p_5p_0)と交差する(交差回数が偶数)ので、点 p_a は多角形の外部にあると判定でき、点 p_b を端点とする半直線 L_b は1辺(辺 p_0p_1)のみ

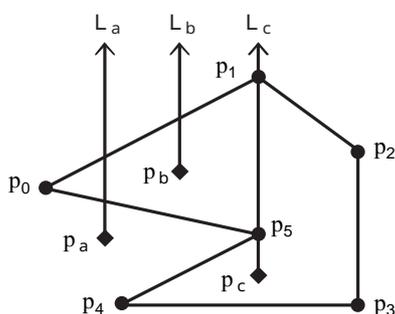


Fig. 8 鉛直線アルゴリズムによる内点判定法

Method for deciding inside point by vertical line algorithm

と交差する(交差回数が奇数)ので、点 p_b は多角形の内部にあると判定できる。

このように、一般的には、多角形の周の一部区間として、辺(頂点列中の連続する2頂点を両端点とする線分)を順次選択し、各辺が半直線と交差するか否かを判定することにより、半直線が多角形の周と交差する回数を計数することができる。

しかし、多角形の頂点が半直線上にある場合には、半直線と辺の交差の有無を調べるだけでは、半直線が当該頂点において多角形の周と交差するのか、接するのかを判別することはできない。例えば、Fig. 8において点 p_c を端点とする半直線 L_c は、頂点 p_1 において多角形の周と交差し、頂点 p_5 において多角形の周と接している(交差しない)が、半直線 L_c と辺の交差の有無を調べるだけではこれらを判別することができない。

本システムにおける内点判定処理は、鉛直線アルゴリズムによる内点判定法の原理を応用するが、半直線との交差判定を行う多角形の周の一部区間(以下「部分周」という)の選定方法を改良することにより上記問題を解決するとともに、多角形の周上(頂点上及び辺上)の点についての判定も可能とした。以下、具体的な手順を述べる。

(1) 半直線の方向

判定対象点を $p_t(x_t, y_t)$ とし、鉛直線アルゴリズムによる内点判定に用いる半直線 L_t を、 p_t を端点として y 軸の正の方向に引いた状態について考える。

ここで、例外的なケースとして、判定対象点 p_t が多角形の頂点上にある場合、あるいは、多角形に y 軸に平行な辺があり、かつ、 p_t が当該辺上にある場合については、以下の(2)~(4)で述べる手順では判定できない。そこで、(2)以降の処理を開始する前に、上記例外的なケースに該当するか否かの判定を行う。その結果、該当する場合には、 p_t は多角形の周上にあると判定し、その時点で内点判定処理を終了し、該当しない場合に(2)以降の処理を行う。

(2) 多角形の周と半直線の交差回数の計数

まず、交差回数を記憶するデータを0で初期化する。

次に、(3)で述べる方法によって部分周を選定し、(4)で述べる方法によって当該部分周と半直線 L_t の交差判定を行い、交差すると判定された場合には交差回数データを1増加させる。

上記の処理を多角形の全周について一通り施すまで繰り返すことによって、多角形の周と L_t の交差回数が計数される。

ただし、上記処理中、 p_t が辺上にあると判定された場合には、 p_t は多角形の周上にあると判定し、その時点で内点判定処理を終了する。

(3) 部分周の選定

まず、 L_t を含む直線上にない(x 座標値が x_t に等しく

ない)頂点を, 部分周の始点 p_s に選ぶ。具体的には, 初回にあっては, まず, 頂点列の終点 p_{n-1} を部分周の始点候補と考え, その x 座標値が x_t と等しくない場合には p_{n-1} を初回の p_s とし, 等しい場合には, 頂点列の順番と逆に1個ずつ始点候補を後退させていき, 最初に x 座標値が x_t と等しくなくなった頂点を初回の p_s とする。2回目以降にあっては, 直前に半直線との交差判定処理を終えた部分周の終点(後述のように, L_t を含む直線上にない)を p_s とする。

次に, 始点 p_s の次の頂点以降の頂点で L_t を含む直線上にないもののうち, 頂点列上の順番が p_s に最も近いものを部分周の終点 p_e に選ぶ。具体的には, まず, 頂点列における p_s の直後の頂点(初回にあっては頂点列の始点 p_0)を終点候補と考え, その x 座標値が x_t と等しくない場合には当該終点候補を p_e とし, 等しい場合には, 頂点列の順番に従って1個ずつ終点候補を前進させていき, 最初に x 座標値が x_t と等しくなくなった頂点を p_e とする。

以上のように選んだ部分周の始点 p_s , 終点 p_e は, いずれも L_t を含む直線上にはない。

ここで, p_s と p_e が頂点列において連続する2頂点であ

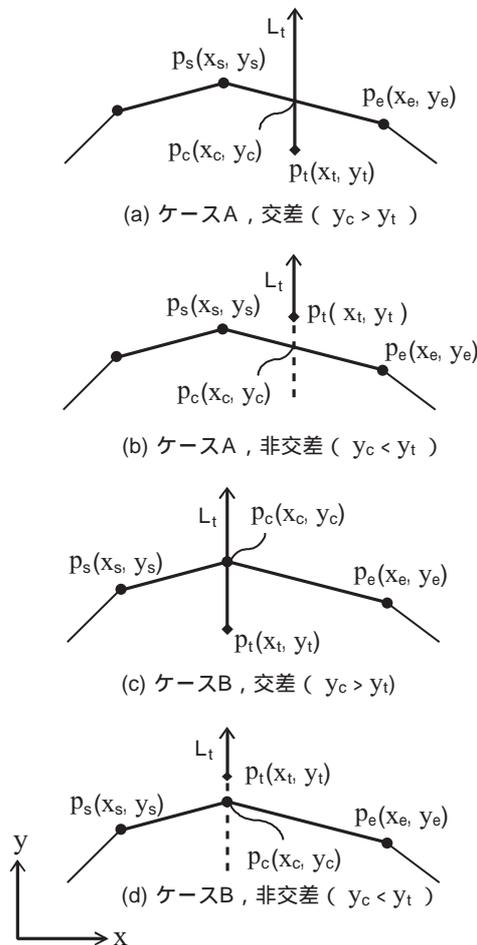


Fig. 9 半直線と部分周との交差判定
Method for deciding intersection of half line
and partial perimeter

る場合をケースAとする。当該部分周の選定において, 始点候補の後退(初回のみ), 及び, 終点候補の前進のいずれも全く行わなかった場合がこれに該当する。このとき, p_s と p_e を両端点とする線分は, 多角形の1辺である(Fig. 9(a),(b))。

また, p_s と p_e の間に1個以上の頂点がある場合をケースBとする。当該部分周の選定において, 始点候補の後退(初回のみ), あるいは, 終点候補の前進のいずれかを1回以上行った場合がこれに該当する。このとき, p_s と p_e の間にある全ての頂点の x 座標値は x_t と等しい(Fig. 9(c),(d))。

(4) 部分周と半直線の交差判定

まず, 判定対象点の x 座標値 x_t が, p_s の x 座標値 x_s と p_e の x 座標値 x_e の範囲内にあるか否かを判定する。

x_t が x_s と x_e の範囲内になければ, L_t が部分周 $p_s p_e$ と交差することはないので, 交差回数データは変更せず, 当該部分周についての交差判定を終了して次の部分周の選定に移る。

x_t が x_s と x_e の範囲内にある場合で,

ケースA, つまり, 部分周 $p_s p_e$ が多角形の1辺である場合には, L_t を含む直線($x = x_t$)と辺 $p_s p_e$ の交点の y 座標値を y_c とし,

ケースB, つまり, 部分周 $p_s p_e$ が連続する複数の辺であり, p_s と p_e の間に1個以上の頂点(直線 $x = x_t$ 上にある)がある場合には, p_s と p_e の間の頂点のいずれか1個(例えば, p_s の次の頂点)の y 座標値を y_c とする。

このように y_c を求めれば, ケースA, Bいずれの場合でも, y_c と y_t の大小関係を調べることにより, L_t が部分周 $p_s p_e$ と交差するか否かを判定することができる。すなわち, $y_c > y_t$ ならば L_t は部分周 $p_s p_e$ と交差するので(Fig. 9(a),(c)), 交差回数データを1増加させ, $y_c < y_t$ ならば L_t は部分周 $p_s p_e$ と交差しないので(Fig. 9(b),(d)), 交差回数データは変更せず, 当該部分周についての交差判定を終了して次の部分周の選定に移る。

ここで, ケースAの場合には, $y_c = y_t$ となる可能性もあるが, その場合は p_t は辺 $p_s p_e$ 上にあるので, p_t は多角形の周上にあると判定し, その時点で内点判定処理を終了する。

結 言

ある空間的要素を対象としてGISを用いて何らかの処理を行おうとするならば, その地理情報をデジタル化する必要がある, それには多くの労力, 時間, 資金の投入を要する。GISを用いた調査研究を効率的に行う上で, 既存のデータが利用可能な場面では可能な限りそれを活用すべきであろう。本報で述べたデータ処理方法とそれに基づくトポロジー修正システムは, 従来は部分的欠陥があるために利

用できなかったデータを利用可能なものに変換するものであり、調査研究の効率化に貢献するものである。また、地図等に基づいて新たにポリゴンデータを作成する際には、

で述べたようなトポロジー規則に従ったデータとなっているか否かを確認し、不規則な部分を検出して修正する工程が必要であるが、本報のデータ処理方法を応用することにより、そのためのシステムを開発することは容易である。さらに、本報で述べた各種処理方法は、単独で、あるいはいくつかを組み合わせることで様々な図形データ処理に応用することができる。関係方面で活用されることを期待するものである。

本報で述べたデータ処理方法の検討とシステムの開発は、農林水産省委託プロジェクト「流域圏における水循環・農林水産生態系の自然共生型管理技術の開発」の実行

課題「自然共生型環境管理に資する流域環境情報データベースの構築」の一環として行ったものである。なお、同データ処理方法は独立行政法人農業工学研究所より特許出願中である。

参考文献

- 1) ESRI (Environmental Systems Research Institute) (1998): ESRI Shapefile Technical Description, ESRI, p.9 (邦訳: シェープファイルの技術情報, ESRIジャパン株式会社)
- 2) 今井浩・今井桂子 (1994): 情報数学講座12 計算幾何学, 共立出版, p.188

Topology normalization system for GIS polygon data

IIJIMA Takashi, ISHIDA Kenji, MATSUMORI Kenji and MINETA Takuya

Summary

Authors developed the data processing system to normalize topology for GIS polygon data. The system gives the following management to the polygon data.

1. When a polygon has self-intersection(s), the system divides the polygon at the self-intersection(s).
2. When polygons intersect to each other, the system transforms them into the polygons which don't intersect to each other.
3. The system adjusts the direction of the vertex arrays of polygons so that the right side of the array direction becomes the inside of the polygon region.

By using the system, it becomes available the polygon data which weren't available before because there were partial topological defects in the data. And the system will contribute to the increase in efficiency of investigation / research which GIS is used for.

Keywords : GIS, Polygon, Topology, Data processing