Title
Unit 34 - The Polygon Overlay Operation

Authors
Unit 34, CC in GIS
White, Denis

Publication Date
1990

Peer reviewed

UNIT 34 - THE POLYGON OVERLAY OPERATION

UNIT 34 - THE POLYGON OVERLAY OPERATION

Compiled with assistance from Denis White, Environmental Protection Agency, Corvallis, OR

UNIT 34 - THE POLYGON OVERLAY OPERATION

Compiled with assistance from Denis White, Environmental Protection Agency, Corvallis, OR

A. INTRODUCTION

- the simple algorithms discussed previously form the basis for one of the most complex operations of vector GIS systems - polygon overlay

Traditions of polygon overlay use

- the three traditions of polygon overlay are:

1. Landscape planning

    - superimposing layers of geographical data (e.g. environmental and social factors) so that their spatial relationships can be used in making land use decisions
    - a key reference is McHarg, 1969, Design with Nature

2. Set theoretic

    - a polygon can be thought of as representing a set
    - when two sets (polygons) A and B are overlaid, we have a graphic interpretation of the set concepts of intersection and union

diagram

        - the area of overlap of A and B is A.AND.B (intersection)
        - the combined area is A.OR.B (union)

overhead/handout Sixteen combinations of Boolean operations

    - it is possible to identify 16 such combinations, or Boolean expressions including e.g.
        - A.AND.(NOT.B), i.e. all of the area of A which is not overlapped by B
        - NOT.(A.OR.B), i.e. the outside world (= (NOT.A).AND.(NOT.B)
    - in most polygon overlay operations it is the intersection which is of most interest, i.e. the area which is common to A and B

3. Area interpolation

Given: the population of area A and the fact that areas A and B overlap Estimate: the population of area B

diagram

- this problem can be solved by apportioning the population of A so that the amount in the area covered by B is proportional to the area of overlap

- this is a simple way of estimating the population of areas from statistics based on other areal units
    - e.g. estimating populations of voting districts from populations of census tracts

- this method assumes that density is uniform within A, but more realistic versions of the technique exist
    - see Unit 41

B. GENERAL CONCEPTS OF POLYGON OVERLAY OPERATIONS

- in GIS, the normal case of polygon overlay takes two map layers and overlays them
    - each map layer is covered with non-overlapping polygons

- if we think of one layer as "red" and the other as "blue", the task is to find all of the polygons on the combined "purple" layer

diagram

- attributes of a "purple" polygon will contain the attributes of the "red" and "blue" polygons which formed it
  - can think of this process as "concatenating" attributes
  - usually a new attribute table is constructed that consists of the combined old attributes, or new

    attributes formed by logical or mathematical operations on the old ones

- number of polygons formed in an overlay is difficult to predict
  - there may be many polygons formed from a pair of "red" and "blue" polygons, with the same "purple" attributes

- when two maps are overlaid, will result in a map with a mixture of 3 and 4 arc intersections
  - four arc intersections do not generally occur on simple polygon maps

Operations requiring polygon overlay

1. Windowing
   - the windowing operation, in which a window is superimposed on a map and everything outside the window is discarded, is a special case of polygon overlay

2. Buffering
   - buffering around points, lines and polygons is another case
     - buffers are generated around each point or straight line segment
     - the combined buffer is found by polygon overlay

diagram

3. Planar Enforcement

   - the process of building points, lines and areas from digitized "spaghetti" (see Unit 12)
     - wherever intersections occur between lines, the lines are broken and a point is inserted
     - the result is a set of points, lines and areas which obey specific rules:

C. OVERLAY ALGORITHMS

- to overlay polygons it is necessary first to find all intersections between "red" and "blue" boundary lines

- arcs are more efficient than polygons for this operation because:
  - intersections need to be found only once rather than four times
  - more information is available about labels (see below)

- example of overlay operation:

  overheads - Simple overlay example

Objective

- overlay two maps of different themes and determine the combined attributes of the new polygons
  - e.g. overlay a soils map on a vegetation map and create a new set of polygons with a new set of attributes

Given

  - two overlapping polygons as follows:
    - red map: a polygon with attribute A
    - blue map: a polygon with attribute 1
  - the outside world labelled 0 on both maps

  - two intersecting arcs defining the boundaries of the polygons:

    1. Red Map (light lines): (0,1) (0,3) (2,3) (2,1) (0,1) Polygons - Right: A Left: 0

    2. Blue Map (heavy lines): (1,0) (3,0) (3,2) (1,2) (1,0) Polygons - Right: 0 Left: 1

Procedure

- after intersections have been found, six new arcs are formed, three from arc 1 and three from arc 2:

  1. (0,1) (0,3) (2,3) (2,2) 2. (2,2) (2,1) (1,1) 3. (1,1) (0,1) 4. (1,0) (3,0) (3,2) (2,2) 5. (2,2) (1,2) (1,1) 6. (1,1) (1,0)

- because the right and left polygon labels are known for each input arc, we know the labels of the new polygons as soon as the intersections have been found
  - there are four new polygons
  - their attributes combine red and blue attributes: 00, A0, A1 and 01

- the arc right and left labels, deduced from the geometry of the intersections, are:

  Arc Right Left 1 A0 00 2 A1 01 3 A0 00 4 00 01 5 A0 A1 6 00 01

- a more complex example:

  overhead - Complex overlay example

- in this case the right and left polygon labels for arcs 1, 2, 4, 5 and 7 would be known from the geometry of the intersections:

  1R: A0 1L: 00 2R: A1 2L: 01 4R: A1 4L: 01 5R: 01 5L: 00 7R: A1 7L: A0

- the labels of the remaining arcs must be determined

- labels can be passed from one arc to another around a polygon:

  3R: must be the same as 2R and 4R 6L: from 2L, 4L

- arc 3 was part of the red network, so its soils labels are known, the remaining (blue) part of its left label must be the same as the blue part of its right label
  - 3R is A1 - thus 3L is B1
  - thus 6R is B1

- how to get the blue labels of arc 8?
  - use a point in polygon routine to find the enclosing blue polygon
  - use a data structure in which arcs on the inside of the polygon boundary "point" to arcs on the outside of any enclosed islands
    - e.g. 5R -> 4L -> 6L -> 8L -> 2L -> 5R
  - thus the labels of arc 8 are 8L: 01, 8R: C1

- the final step in the algorithm is to identify all new polygons by following around each polygon from one arc to the next until every right and left side of every arc has been identified with a uniquely numbered polygon

D. COMPUTATIONAL COMPLEXITY

- polygon overlay is numerically intensive and time consuming, therefore it is the most complex operation of most vector-based GIS programs

- notation: if computing time to process n objects is proportional to n, the computational complexity is "order n", or O(n)
  - if it is proportional to n2, we way it is "order n squared" or O(n2)

- it is important to know:
  - how long does it take to overlay a given number of polygons?

- what affects the amount of time taken?
  - obviously, the number of arcs and polygons affects the number of computations required
    - it is usually possible to determine the number of polygons being overlaid
    - the number of arcs is roughly 3 times the number of polygons
  - other factors, such as the wiggliness of boundaries, affect the time, but it is difficult to measure these

- if n1 polygons are overlaid on n2, how many polygons result? (assuming maps are different)
  - minimum is n1+n2, polygons on the two maps do not intersect at all
  - maximum is infinity, lines have infinite wiggliness

- typical is 3 or 4 times (n1+n2), discounting spurious polygons

- if every one of n1 "red" polygons has to be compared to every one of n2 "blue" polygons, the algorithm will be O(n1n2)

- if we could immediately find all "blue" arcs likely to intersect with a given "red" arc, we could build an algorithm which would be O(n1), which means it would be much more efficient for a given size of problem

- to find arcs in this way we would need an efficient method of spatial indexing
    - one of the most successful methods uses the moving band concept:
        - sort both "red" and "blue" arcs in ascending order of x
        - process the arcs beginning at the left, moving to the right, sweeping a "band" across the map
        - only those arcs falling in the band are examined
        - since the arcs are sorted, we can find those in the band on either red or blue maps quickly

- some of the best polygon overlay routines now available in the GIS market operate in close to O(n1+n2) time
    - a map with tens of thousands of polygons can be overlaid on another map with a similar number in an hour of computing time on a moderate-sized machine

E. INTERSECTION PROBLEMS

- because of computer precision, lines will be represented in the computer with great precision even though the accuracy of the representation is low

1. Adjacent lines

- the following diagram represents a case where two lines cross

  diagram

- the following diagram represents a similar case where the two lines do not actually cross

  diagram

- it is necessary to provide algorithms which distinguish between these very different conditions

2. Sliver polygons

- overlay algorithms compute the exact intersections between lines (see Unit 14)

- in any overlay operation, it is likely that there will be pairs of lines which should coincide, but do not because of differences in digitizing
    - these are called slivers, spurious polygons or "coastline weave"

  overhead - Overlay of two images

if an arc or polygon of n1 points is overlaid on one of n2 points, up to: (2n1n2/(n1+n2) - 3) slivers may be generated

- two possible approaches: a. delete during the overlay operation, or b. delete afterwards

Removing slivers during overlay

- this is the most common approach used in commercial GIS programs
- this approach deals with the line as if it were fuzzy

diagram

- define a tolerance limit for each line which indicates the amount of uncertainty that exists regarding the true position of the digitized line
  - provides a band of width "epsilon" around every line
  - for digitized lines this width may be 1 mm
- using epsilon, can then conclude that lines which have a difference in position less than epsilon are the same line
  - these two lines can then be collapsed to represent a single line

Problem

- it is easy to get into difficulty:
  - lines A and B within epsilon, therefore the same:
  - lines B and C within epsilon, therefore the same:
  - but lines A and C are not necessarily within epsilon of each other
- polygon overlay routines which do sliver removal "on the fly" must deal with this problem

Removing slivers after overlay

- need intelligent criteria to distinguish between slivers and real polygons
- criteria for removal include:
  - area: slivers are small
  - shape: slivers are long and thin
  - number of arcs: slivers generally have only 2 bounding arcs while real polygons rarely have only 2
  - alternating attributes: if a "red" arc between polygons A and B is overlaid on a "blue" arc between polygons 1 and 2, the slivers will alternate between A2 and B1

diagram

  - junctions: slivers terminate in 4 arc junctions, but 3 arc junctions are more common in real polygons

chaining: slivers tend to occur in chains
  - it is likely that the confidence with which it can be concluded that two arcs are forming slivers will increase steadily as we work along the arc
  - i.e. the attribute "sliver" is strongly correlated

- if a sliver is detected, it can be replaced by an arc along its center line

REFERENCES

McHarg, I.L., 1969. Design with Nature, Doubleday, New York

Goodchild, M.F., and N. S. Lam, 1980. "Areal Interpolation," Geoprocessing 1:297-312.

DISCUSSION AND EXAM QUESTIONS

1. McHarg described the overlay technique well before the advent of GIS and polygon overlay. Discuss the advantages and possible disadvantages of a computer implementation of the technique.

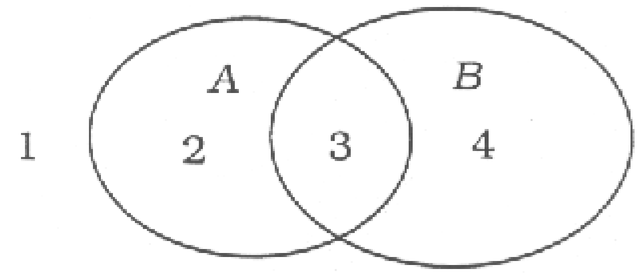
2. Write out and illustrate the 16 Boolean combinations of two polygons A and B.

3. Review and discuss the alternative forms of areal interpolation described by Goodchild and Lam, 1980.

4. Discuss the relative advantages of raster and vector approaches to polygon overlay. Identify the application areas likely to adopt each method given their advantages.

Last Updated: August 30, 1997.

UNIT 34 IMAGES

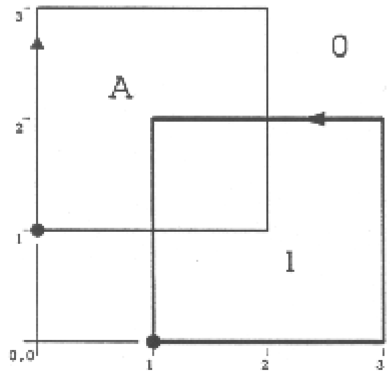


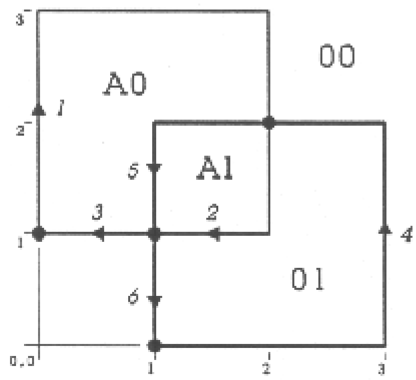
Intersection Concept

1	2	3	4	
1	1	1	0	A.OR.(NOT B)
1	1	1	1	NOT NULL
1	1	0	0	NOT B
1	1	0	1	(NOT A).OR.(NOT B)
1	0	1	0	(A.AND.B).OR.((NOT A).AND.(NOT B))
1	0	1	1	(NOT A).OR.B
1	0	0	0	(NOT A).AND.(NOT B)
1	0	0	1	NOT A
0	1	1	0	A
0	1	1	1	A.OR.B
0	1	0	0	A.AND.(NOT B)
0	1	0	1	(A.AND.(NOT B)).OR.((NOT A).AND.B)
0	0	1	0	A.AND.B
0	0	1	1	B
0	0	0	0	NULL
0	0	0	1	(NOT A).AND.B

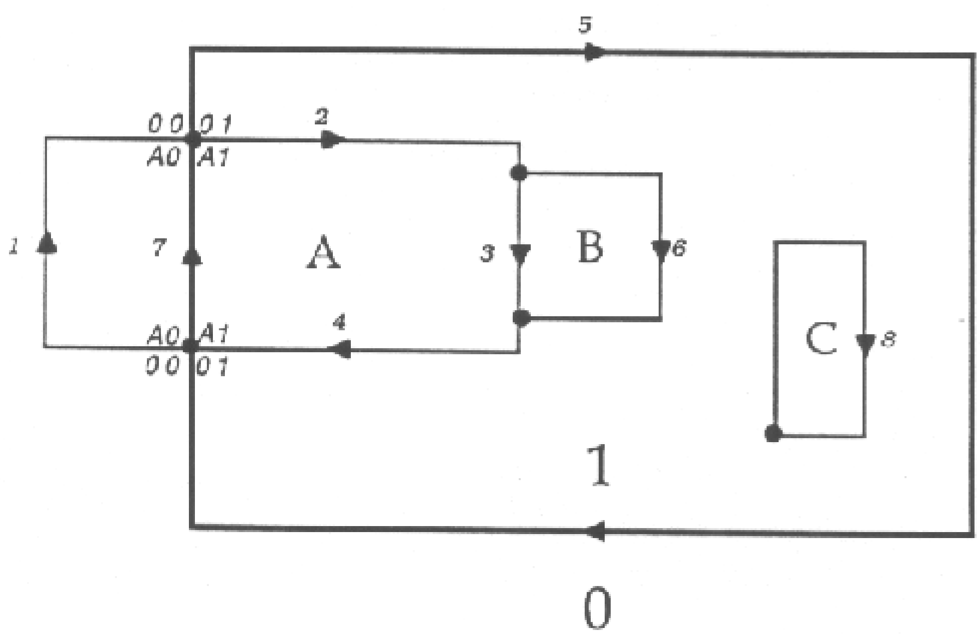
1 : YES

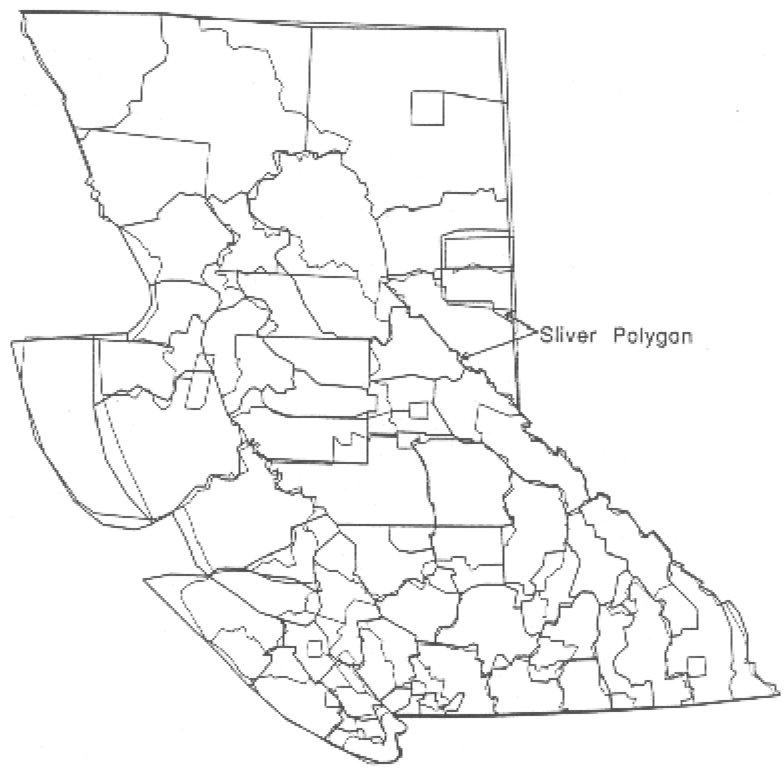
0 : NO





Arc	Right Poly	Left Poly
1	A0	00
2	A1	01
3	A0	00
4	00	01
5	A0	A1
6	00	01




Sliver Polygon