



Tạp chí Khoa học Kỹ thuật Mỏ - Địa chất

Trang điện tử: <http://tapchi.humg.edu.vn>



Kỹ thuật nâng cao tìm sửa lỗi trong bài toán tạo vùng phục vụ công tác biên tập bản đồ

Phạm Thế Huynh *

Khoa Trắc địa – Bản đồ và Quản lý đất đai, Trường Đại học Mỏ - Địa chất, Việt Nam

THÔNG TIN BÀI BÁO

Quá trình:

Nhận bài 15/08/2017

Chấp nhận 18/10/2017

Đăng online 29/12/2017

Từ khóa:

Tạo vùng

Lỗi khép vùng

Bản đồ địa chính

Thành lập bản đồ

Khoảng hở lớn nhất

TÓM TẮT

Việc tạo vùng từ các đường liên kết bởi những đối tượng dạng đường được ứng dụng nhiều trong thực tế. Các đối tượng dạng đường có thể là đoạn thẳng, đa tuyến, cung tròn... Trong một số chương trình đồ họa, chức năng tạo vùng được sử dụng khi tính diện tích, tô màu, tạo đường ranh giới khép kín... Tuy nhiên, trước khi tạo vùng, các đối tượng đường biên phải được xử lý các lỗi khép vùng nhằm tạo ra các vùng khép kín. Bài báo này trình bày kỹ thuật tự động tìm, sửa lỗi tạo vùng phục vụ công tác biên tập bản đồ. Các lỗi trong bài toán tạo vùng như điểm cuối tự do, trùng đè, giao nhau được phát hiện tự động và thông báo đến người sử dụng.

© 2017 Trường Đại học Mỏ - Địa chất. Tất cả các quyền được bảo đảm.

1. Đặt vấn đề

Một số mô đun phần mềm biên tập bản đồ địa chính ở Việt Nam hiện nay chạy trên nền các phần mềm đồ họa nước ngoài như Famis, TMV.Map, VietMap XM... khi xác định lỗi khép vùng trước khi tạo vùng thường sử dụng mô đun MRFClean của hãng Bentley. Thuật toán tạo vùng đã được trình bày trong một số tài liệu (Robert Sedgewick, 1994; Trần Thùy Dương, 2006; Trần Thùy Dương, 2007; Đinh Hải Nam, 2009; Phạm Thế Huynh, 2015)..., tuy nhiên việc xác định lỗi khép vùng trước khi tạo vùng chưa được quan tâm xử lý triệt để trong các nghiên cứu này đối với các trường hợp biên.

Các lỗi khép vùng xảy ra có thể không tạo được vùng hoặc tạo ra những vùng không đúng theo thực tế và thường khó phát hiện khi số lượng

vùng khép kín cần tạo lớn dẫn đến những sai lầm không thể chấp nhận được. Những sai lầm không được phát hiện khi tạo vùng có thể tốn rất nhiều thời gian và công sức để sửa chữa. Đặc biệt, đối với công tác biên tập bản đồ địa chính: thửa đất là một vùng khép kín, là yếu tố quan trọng nhất trên bản đồ địa chính, được gắn nhiều thông tin thuộc tính, liên quan đến nhiều hồ sơ kèm theo; ranh giới thửa đất được kiểm tra chặt chẽ tới từng chủ sử dụng và trải qua nhiều cấp kiểm tra nghiệm thu.

Nhằm chủ động trong việc xây dựng chương trình đồ họa độc lập của Việt Nam phục vụ công tác biên tập bản đồ nói chung và biên tập bản đồ địa chính nói riêng, cần nghiên cứu kỹ thuật xác định lỗi khép vùng của các đối tượng đường biên một cách triệt để.

Kỹ thuật xác định lỗi khép kín vùng thường gặp trong công tác biên tập bản đồ được trình bày một cách chi tiết trong bài báo này.

*Tác giả liên hệ

E-mail: phamthehuynh@humg.edu.vn

2. Giải quyết vấn đề

2.1. Bài toán tạo vùng

Mục đích của bài toán tạo vùng là đi tìm các đoạn giới hạn biên của vùng, thường là các đoạn liên tiếp trên biên. Để thực hiện được mục đích này, tiến hành bắt đầu từ một điểm đầu mút của một đoạn trên biên một vùng, lần lượt xét các đoạn để tìm đoạn có đầu mút trùng với đầu mút thứ hai của đoạn xuất phát, quá trình này thực hiện liên tục cho đến khi quay được trở về điểm đầu mút ban đầu. Tiếp tục, tiến hành lần lượt lát kín mặt phẳng không gian cần tạo vùng bằng các vùng như vậy chung quanh điểm xét theo chiều kim đồng hồ (hoặc ngược lại).

Như vậy, đầu vào của bài toán tạo vùng sẽ là các đoạn. Đoạn ở đây không chỉ là đoạn thẳng mà có thể là một đối tượng dạng tuyến có hai đầu mút phân biệt như một đường đa giác, một cung tròn, một đoạn đường cong... Khi danh sách các đoạn chứa nhiều dạng đối tượng khác nhau thì việc tổ chức quản lý các đoạn cần được thực hiện dựa trên các cấu trúc đặc biệt từ đó định rõ chỉ số điểm đầu, điểm cuối, thuộc tính, kiểu đối tượng của đoạn cũng như phương pháp quản lý dữ liệu từng đoạn.

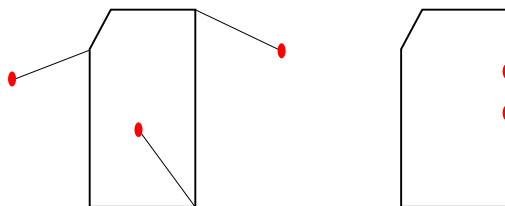
Mỗi đoạn là đường biên chung của hai vùng kề nhau, do đó mỗi đoạn cần quản lý với hai chiều ngược nhau nhằm mục đích khi quét qua các đoạn, mỗi đoạn chỉ được quét qua một lần.

2.2. Các lỗi khép vùng

Trong thực tế, một cách trực quan, nhìn vào bản vẽ đơn giản người ta có thể phát hiện ngay



Hình 1. Các đối tượng trùng nhau.



Hình 2. Các điểm cuối tự do.

các vùng khép kín tương đối chính xác bằng cách xuất phát từ một điểm, dò các đường biên gần nhất và lát toàn bộ mặt phẳng chung quanh điểm này. Tuy nhiên, vùng mà mắt thường nhìn thấy trên bản vẽ không thể xác định chính xác đó có phải là một vùng khép kín hay không do có những đoạn hở, những đoạn giao nhau, trùng nhau..., từ đó sẽ cho ra kết quả không đúng với thực tế. Đối với chương trình tự động tạo vùng, danh sách các đoạn đưa vào tạo vùng cần được tìm, sửa lỗi tự động hoặc đánh dấu những lỗi khép vùng để người biên tập sửa chữa triệt để. Sau đây là một số lỗi khép vùng thường gặp.

2.2.1. Các đối tượng trùng nhau:

Các đối tượng trùng nhau khi tạo vùng có thể tạo ra những vùng lặp hoặc có diện tích bằng 0.

2.2.2. Các điểm cuối tự do:

Các điểm cuối tự do sẽ làm cho việc tạo vùng cho ra kết quả là một vùng có diện tích bằng 0 hoặc vùng có chứa các đỉnh lặp (không phải là đỉnh xuất phát).

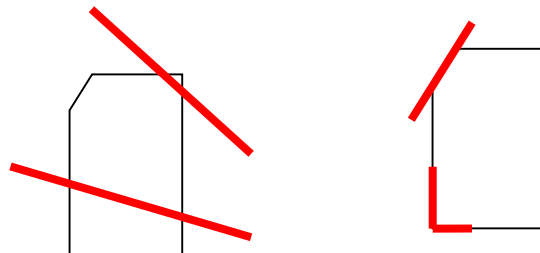
2.2.3. Các đối tượng giao nhau:

Các đối tượng giao nhau có thể do nối vùng sai hoặc bắt buộc phải thực hiện trong các bài toán chồng phủ các vùng nhằm tạo ra các vùng giao nhau. Quá trình xác định giao giữa các đoạn tương đối phức tạp, gặp nhiều trường hợp biên cần xử lý một cách triệt để (như Hình 3) nhằm tạo ra các đoạn rời nhau thì mới có thể tạo vùng một cách chính xác (Vũ Thị Minh Huyền, 2016).

2.3. Kỹ thuật xác định lỗi khép vùng

2.3.1. Các đối tượng trùng nhau

Đối với chương trình quản lý các đối tượng đồ họa, các đoạn đã được định rõ điểm đầu, điểm cuối, thuộc tính và kiểu đối tượng cùng với dữ liệu quản lý chúng một cách chặt chẽ. Như vậy, việc xác định các đối tượng trùng nhau tương đối dễ dàng



Hình 3. Các đối tượng giao nhau.

nhờ việc sắp xếp và so sánh các đoạn cùng kiểu dữ liệu, cùng thuộc tính, cùng điểm đầu và điểm cuối. Việc lựa chọn giải pháp xử lý sau khi xác định các đối tượng trùng nhau được cài đặt trước khi tìm lỗi nhằm xây dựng danh sách các đoạn tạo vùng không chứa các đoạn trùng nhau.

2.3.2. Các điểm cuối tự do

Tại mỗi đỉnh của danh sách đoạn sẽ quản lý số hướng xuất phát từ đỉnh này. Số hướng tại mỗi đỉnh luôn có giá trị lớn hơn hoặc bằng 1. Nếu số hướng bằng 1 thì đỉnh xét là điểm cuối tự do. Có thể sử dụng thuật toán rút đỉnh cho các điểm cuối tự do để tự động đánh dấu và loại bỏ các điểm này.

2.3.3. Các đối tượng giao nhau

a. Giao nhau của hai đoạn thẳng

Trước hết cần làm rõ khái niệm hai đoạn thẳng giao nhau. Thông thường, trong hình học phẳng, hai đường thẳng chỉ có các vị trí tương đối là trùng nhau, song song hoặc cắt nhau. Hai đoạn thẳng 12 và 34 giao nhau khác với hai đường thẳng giao nhau. Hai đoạn thẳng được coi là giao nhau nếu chúng không trùng nhau hoàn toàn và giao điểm của hai đường thẳng chứa chúng nằm trong cả hai đoạn thẳng hoặc một đầu mút của đoạn thẳng này nằm trong đoạn thẳng kia.

Dưới đây là tất cả các trường hợp thể hiện vị trí tương đối của hai đoạn thẳng 12 và 34 (Hình 5).

Qua 18 trường hợp đã chỉ ra cho thấy 2 đoạn thẳng có rất nhiều vị trí tương đối với nhau. Vậy kỹ thuật xử lý như thế nào để có thể nhanh chóng xác định được chính xác mối quan hệ giữa hai đoạn thẳng?

Kỹ thuật xử lý:

* Có thể xác định vị trí tương đối của 2 đầu mút đoạn thẳng này với đoạn thẳng kia bằng cách sử dụng hàm CCW (Robert Sedgewick, 1994), tuy nhiên kỹ thuật này không xác định được tọa độ điểm giao và trường hợp các đoạn thẳng nằm trên hai đường thẳng đồng phương.

* Sử dụng phương pháp đại số như sau:

- Xét hai đường thẳng $d1$, $d2$; $d1$ đi qua các điểm có tọa độ $(x1, y1)$ và $(x2, y2)$; $d2$ đi qua các điểm có tọa độ $(x3, y3)$ và $(x4, y4)$;

- Tính: $dx12 = x2 - x1$; $dy12 = y2 - y1$; $dx34 = x4 - x3$; $dy34 = y4 - y3$

$TS = dx34 * (x1 * y2 - x2 * y1) - dx12 * (x3 * y4 - x4 * y3)$

$MS = dy12 * dx34 - dy34 * dx12$

- Xét các trường hợp:

+ TS bằng 0 và MS bằng 0 thì $d1$, $d2$ đồng phương (13 vị trí tương đối đầu tiên)

+ TS khác 0; MS bằng 0 thì $d1$ song song với $d2$ (vị trí tương đối 18)

+ MS khác 0 thì $d1$ cắt $d2$ tại điểm M có tọa độ (vị trí tương đối 14 đến 16:

Nếu $dx12$ khác 0 thì: $x5 = TS/MS$; $y5 = y1 + (x5 - x1) * (dy12/dx12)$)

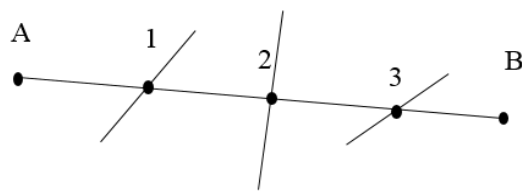
Nếu $dx34$ khác 0 thì: $x5 = TS/MS$; $y5 = y3 + (x5 - x3) * (dy34/dx34)$)

- Xét tiếp điểm M nếu nằm trên cả hai đoạn thẳng 12 và 34 thì M là giao của hai đoạn thẳng.

Với phương pháp đại số như trên tốc độ xử lý vẫn chưa đạt yêu cầu do phải xét quá nhiều trường hợp. Chính vì vậy, trong nghiên cứu này, chúng tôi đã cải tiến giải pháp xử lý nhằm tăng tốc độ xét bằng cách xét trước 15 vị trí tương đối đầu tiên thông qua xét vị trí tương đối 2 đầu mút của đoạn thẳng này với đoạn thẳng kia, nếu có một đầu mút đoạn này nằm trong đoạn kia thì hai đoạn thẳng được coi là giao nhau.

b. Một đoạn bị cắt bởi nhiều đoạn khác

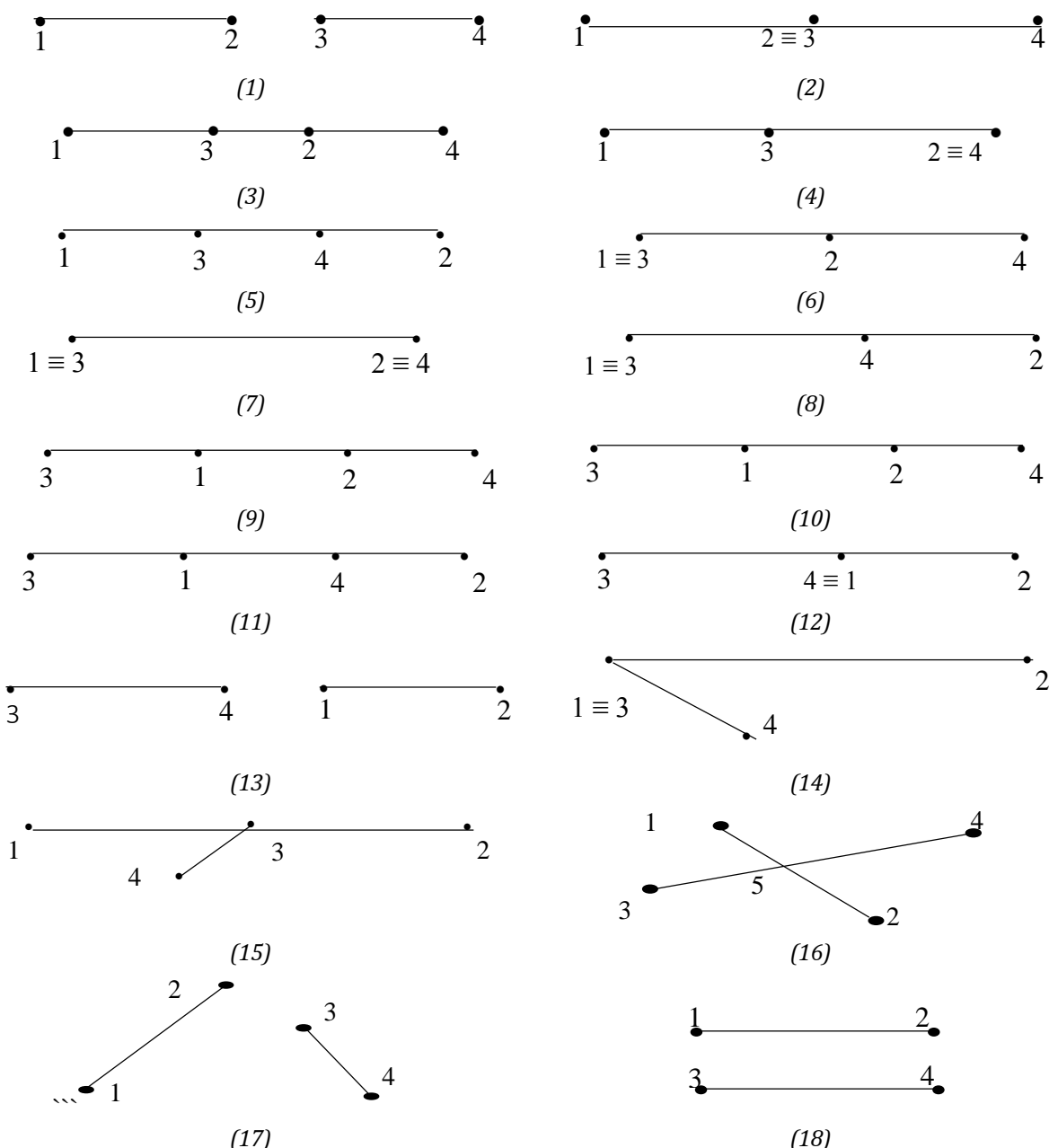
Khi một đoạn m bị cắt bởi k đoạn khác nhau ni , trên đoạn m lưu k vị trí giao. Sắp xếp các vị trí giao theo thứ tự từ điểm đầu đến điểm cuối của đoạn. Sau khi sắp xếp xong, tiến hành phân tách đoạn m theo k vị trí giao sẽ được $k+1$ đoạn thẳng mới. Chẳng hạn, đoạn AB có 3 vị trí giao cắt 1, 2, 3. Sắp xếp tăng dần theo khoảng cách được các đoạn $A1$, $A2$, $A3$. Đoạn AB được phân tách thành các đoạn 4 đoạn mới $A1$, 12, 23 và 3B (Hình 4).



Hình 4. Đoạn AB có 3 vị trí giao cắt 1, 2, 3.

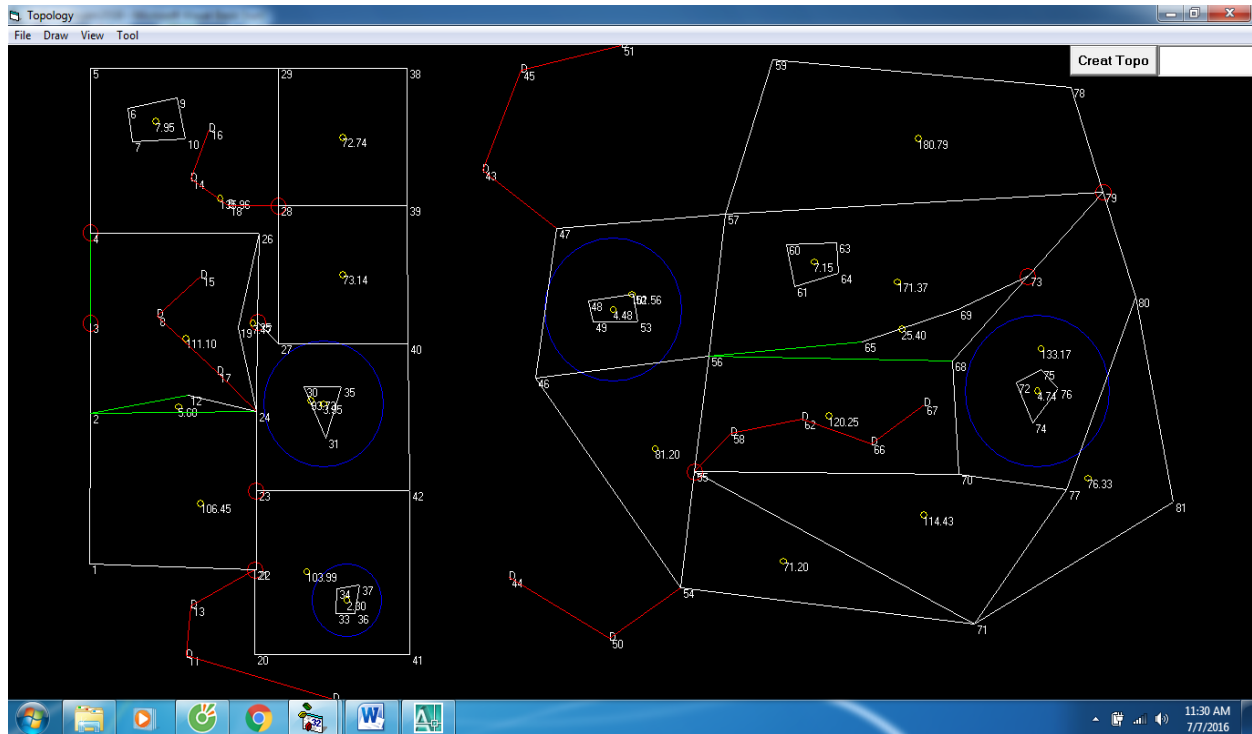
2.4. Chương trình thử nghiệm

Qua nghiên cứu, chúng tôi sử dụng ngôn ngữ lập trình Visual Basic 6.0 lập một chương trình thử nghiệm (Hình 6). Qua thử nghiệm cho thấy, kết quả nhận dạng các lỗi thường gặp với tốc độ nhanh, xử lý các lỗi tương đối triệt để. Trong giới hạn cho phép, bài báo không thể trình bày hết các trường hợp khép vùng, chương trình thử nghiệm



Hình 5. các trường hợp thể hiện vị trí tương đối của hai đoạn thẳng 12 và 34.

(1): 1,2 nằm bên trái 3. Hai đoạn thẳng không cắt nhau; (2): 1 nằm bên trái 3, 2 trùng 3. Cắt thành các đoạn thẳng 12 và 24; (3): 1 nằm bên trái 3, 2 nằm giữa 3 và 4. Cắt thành các đoạn thẳng 13, 32 và 24; (4): 1 nằm bên trái 3, 2 trùng 4. Cắt thành các đoạn thẳng 13 và 34; (5): 1 nằm bên trái 3, 2 nằm bên phải 4. Cắt thành các đoạn thẳng 13, 34 và 42; (6): 1 trùng 3, 2 nằm giữa 3 và 4. Cắt thành các đoạn thẳng 12 và 24; (7): 1 trùng 3, 2 trùng 4. Hai đoạn thẳng không cắt nhau; (8): 1 trùng 3, 2 nằm bên phải 4. Cắt thành các đoạn thẳng 34 và 42; (9): 1 và 2 nằm giữa 3 và 4. Cắt thành các đoạn thẳng 31, 12 và 24; (10): 1 nằm giữa 3 và 4, 2 trùng 4. Cắt thành các đoạn thẳng 31 và 12; (11): 1 nằm giữa 3 và 4, 2 nằm bên phải 4. Cắt thành các đoạn thẳng 31, 14 và 42; (12): 1 trùng 4, 2 nằm bên phải 4. Cắt thành các đoạn thẳng 34 và 42; (13): 1 và 2 nằm bên phải 4. Hai đoạn thẳng không cắt nhau; (14): 1 hoặc 2 trùng 3. Hai đoạn thẳng không cắt nhau; (15): 3 nằm giữa 1 và 2. Đoạn thẳng 12 bị cắt thành các đoạn 13 và 32; (16): Hai đoạn 12 và 34 giao nhau tại 5, cả 2 đoạn bị cắt thành các đoạn 15, 52 và 35, 54; (17): 2 đoạn thẳng không cắt nhau; (18): 2 đoạn thẳng song song.



Hình 6. Chương trình thử nghiệm.

còn nhận dạng thêm một số lỗi thường gặp khác như vùng có góc nhỏ, diện tích nhỏ hoặc chu vi nhỏ, vùng trong vùng... các lỗi thường gặp được đánh dấu trên bản vẽ bằng màu sắc riêng cho từng loại.

3. Kết luận

Kỹ thuật tìm, sửa lỗi tạo vùng trình bày trong bài báo đã xử lý được hầu hết các trường hợp biên của bài toán tạo vùng và xác định giao trong bài toán chồng phủ các vùng. Trong quá trình nghiên cứu, ngoài việc xem xét các trường hợp biên, việc tăng tốc độ tính toán và xử lý đã được tính đến. Kỹ thuật này là cơ sở để xây dựng phần mềm ứng dụng độc lập trong công tác biên tập bản đồ địa chính ở Việt Nam.

Tài liệu tham khảo

Trần Thùy Dương, 2006. Một giải pháp xử lý trường hợp biên trong bài toán tạo Topology.

Tạp chí khoa học kỹ thuật Mỏ - Địa chất 14, 88-91.

Trần Thùy Dương, 2007. Nghiên cứu xây dựng công nghệ thành lập bản đồ số độ cao trong điều kiện Việt Nam. Luận án tiến sĩ kỹ thuật, Đại học Mỏ - Địa chất, Hà Nội.

Phạm Thế Huỳnh, 2015. Nghiên cứu công nghệ thành lập và ứng dụng bản đồ số địa chính trong điều kiện Việt Nam. Luận án tiến sĩ kỹ thuật, Đại học Mỏ - Địa chất, Hà Nội.

Đình Hải Nam, 2009. Nghiên cứu cấu trúc dữ liệu và thuật toán tạo Topology phục vụ cho công tác xây dựng cơ sở dữ liệu và quản lý đất đai. Luận án tiến sĩ kỹ thuật, Đại học Mỏ - Địa chất, Hà Nội.

Robert Sedgewick, 1994. Cẩm nang thuật toán- Các thuật toán chuyên dùng. Tập 2, Nhà xuất bản Khoa học - Kỹ thuật, Hà Nội.

ABSTRACT

The advanced techniques of finding errors in Polybuiding for map editing

Huynh The Pham

Faculty of Geomatics and Land Administration, Hanoi University of Mining and Geology, Vietnam.

Polybuiding from links of linear objects is practically applied in practice. Linear objects can be Lines, Polylines, Arcs, etc. In certain graphic softwares, the selection function is used when calculating area, coloring, creating boundary lines. However, before selecting, boundary objects must be processed to enclose closed areas to create closed areas. This article presents the technique for automatically finding and correcting regions for map editing. Errors in the area problem such as free end point, overlap, intersection are automatically detected and notified to the user.