

Thuật toán tìm đường đi trong mê cung

Bách khoa toàn thư mở Wikipedia

Các **thuật toán tìm đường đi trong mê cung** là những phương pháp được tự động hóa để giải một mê cung. Các thuật toán chọn đường ngẫu nhiên, bám theo tường, Pledge, và Trémaux được xây dựng để một đối tượng sử dụng chạy bên trong mê cung mà hoàn toàn không có biết trước về mê cung, còn các thuật toán lấp kín đường cụt và đường đi ngắn nhất được thiết kế để sử dụng khi đã biết trước toàn bộ mê cung.

Mê cung không chứa các vòng lặp được gọi là mê cung "tiêu chuẩn" hoặc "hoàn hảo", và nó tương đương với một cây trong lý thuyết đồ thị. Vì vậy, nhiều thuật toán tìm đường đi trong mê cung có liên quan chặt chẽ với lý thuyết đồ thị. Một cách trực quan, nếu ta kéo dài các đường trong mê cung ra một cách thích hợp, kết quả thu được có thể trông giống như một cây.^[1]

Mục lục

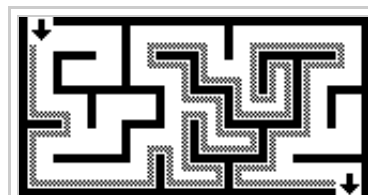
- 1 Thuật toán chọn đường ngẫu nhiên
- 2 Bám theo tường
- 3 Thuật toán Pledge
- 4 Thuật toán Trémaux
- 5 Lấp kín đường cụt
- 6 Xem thêm
- 7 Chú thích
- 8 Liên kết ngoài

Thuật toán chọn đường ngẫu nhiên

Đây là một phương pháp đơn giản có thể được thực hiện bởi một robot rất không thông minh hoặc thậm chí là một con chuột (còn gọi là thuật toán random mouse). Nó chỉ đơn giản là chạy theo một đường thẳng cho đến khi gặp một đường giao nhau thì đưa ra quyết định ngẫu nhiên về hướng tiếp theo để chạy. Mặc dù phương pháp này cuối cùng luôn luôn tìm ra giải pháp đúng, nhưng thuật toán này có thể cực kỳ chậm.

Bám theo tường

Thuật toán bám theo tường (wall follower) là một quy tắc nổi tiếng nhất để vượt qua mê cung, còn được gọi là quy tắc tay trái hoặc quy tắc tay phải. Nếu mê cung chỉ liên thông đơn giản nghĩa là tất cả các bức tường của nó được kết nối với nhau hoặc kết nối với đường bao quanh mê cung, thì bằng cách dò một tay lên một bức tường của mê cung thì người đi đảm bảo không bị lạc và tìm được lối ra nếu có một lối ra trên đường bao; hoặc nếu không có lối ra thì sẽ quay trở lại lối vào và sẽ đi qua tất cả các đường của mê cung ít nhất 1 lần.



Tìm đường theo quy tắc tay phải

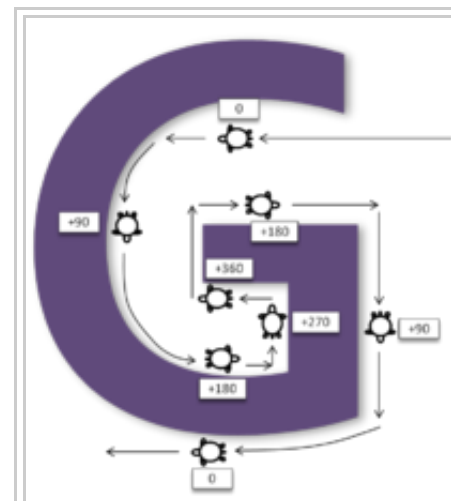
Đây là một khía cạnh khác cho thấy lý do vì sao việc bám theo tường là một topo. Nếu các bức tường được kết nối, thì có thể được kéo giãn biến dạng thành một vòng lặp hoặc vòng tròn.^[2] Do đó, bức tường buộc người đi theo xung quanh một vòng tròn từ điểm đầu đến cuối.

Thuật toán Pledge

Nếu mê cung có các tường rời nhau, đồng thời lối vào lối ra của mê cung nằm trên tường bao của mê cung, thì thuật toán bám tường vẫn có thể tìm được đường ra. Tuy nhiên, nếu điểm vào nằm bên trong mê cung và tách rời khỏi lối ra, thì bám theo tường sẽ chỉ có thể đi thành 1 vòng cục bộ và không tìm được lối ra. Đối với trường hợp này, thuật toán Pledge (được đặt theo tên của Jon Pledge ở Exeter) có thể giải quyết được vấn đề.^[3]

Giải thuật Pledge được thiết kế để vượt qua vật cản, bằng cách chọn một hướng đi bất kỳ. Khi gặp vật cản, thì chuyển sang đi chuyển bám theo tường dọc theo vật cản, kết hợp với đếm góc quay. Sau khi bám tường và quay mà trở về lại đúng hướng đi ban đầu đồng thời tổng góc đã quay bằng '0' (zero) thì tức là đã vượt qua khỏi vật cản, thì không cần bám tường nữa và tiếp tục đi chuyển theo hướng ban đầu đã chọn.

Chỉ thoát khỏi bám theo tường khi thỏa mãn cả 2 điều kiện: "hướng hiện tại trùng với hướng ban đầu" và "tổng số góc đã quay bằng '0'". Điều này là cần thiết để giúp đi chuyển vòng qua vật cản phức tạp ví dụ như vật cản hình chữ "G" như trong hình. Ở vị trí +360°, robot có hướng đi chuyển cùng hướng ban đầu, nếu bỏ bám tường bên phải mà tiếp tục đi chuyển thẳng thì sẽ đi vào 1 vòng lặp mà không thoát ra được. Thay vào đó, nếu tiếp tục bám vào tường phải và thực hiện quay cho đến khi tổng góc quay là '0' thì sẽ thoát ra khỏi vật cản.



Ví dụ thuật toán giúp vượt qua vật cản hình chữ "G": quay ngược kim đồng hồ là chiều dương, quay cùng chiều kim đồng hồ là chiều âm. Các số liệu tương ứng với từng lần quay.

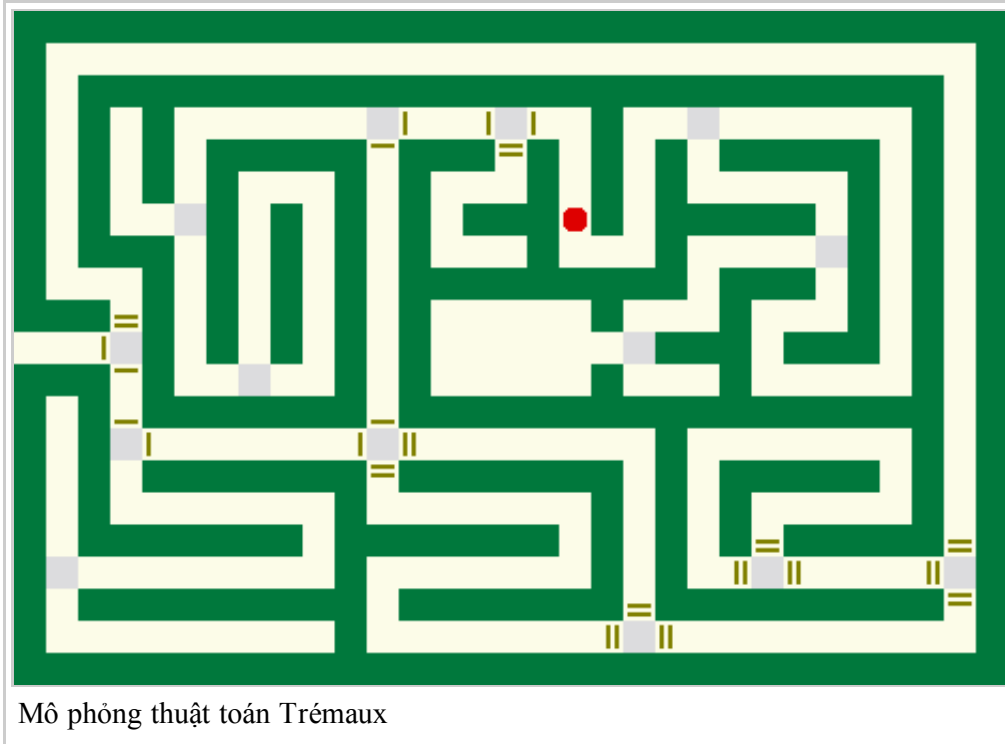
Thuật toán Trémaux

Thuật toán Trémaux được Charles Pierre Trémaux^[4] phát minh, sử dụng các dấu hiệu để ghi nhớ đường đi, ví dụ đánh dấu trên mặt sàn, là một phương pháp hiệu quả để tìm lối ra của một mê cung. Thuật toán có thể giải tất cả các mê cung có đường đi rõ ràng.^[5]

Một đường trong mê cung sẽ được ghi nhớ bằng cách đánh dấu bởi 1 trong 3 trạng thái: chưa qua, đã qua 1 lần hoặc qua 2 lần. Một đường được chọn để đi sẽ luôn được đánh dấu bằng 1 vạch dưới sàn (từ ngã giao này đến ngã giao kia). Tại điểm bắt đầu có thể chọn một hướng bất kỳ (nếu có nhiều hơn một hướng). Khi đến một ngã giao, nếu các đường rẽ đều chưa qua, thì chọn ngẫu nhiên 1 đường để đi và đánh dấu đường ấy 1 vạch. Khi gặp một ngã giao mà đường trước mặt theo hướng đi hiện tại đã có 1 dấu, và đường đang đi hiện tại chỉ mới đánh dấu 1 lần, thì quay trở lại và đánh dấu đường ấy 2 vạch. Nếu đến 1 ngã giao mà không rơi vào 2 trường hợp trên, thì chọn đường đi có ít vạch nhất, và nhớ đánh dấu đường ấy luôn. Khi đến đích, thì những con đường chỉ đánh dấu 1 vạch là đường dẫn trở về điểm xuất phát.

Nếu không có ngã ra, thì phương pháp này sẽ dẫn người đi trở về lại điểm xuất phát, và khi ấy tất cả con đường sẽ đánh dấu 2 vạch, mỗi vạch tương ứng với 1 hướng đi. Kết quả được gọi là vạch đôi 2 chiều.^[6]

Về cơ bản, thuật toán này được phát hiện từ thế kỷ 19 đã được sử dụng khoảng hàng trăm năm sau như một phương pháp tìm kiếm ưu tiên chiều sâu.^{[7][8]}



Lấp kín đường cụt

Lấp kín đường cụt (*dead-end filling*) là một thuật toán để giải mê cung bằng cách lấp kín tất cả các ngã cụt, chỉ để lại một đường chính xác không bị lấp. Nó có thể được sử dụng để giải các mê cung trên giấy hoặc với một chương trình máy tính, nhưng nó không hữu dụng nếu mê cung chưa biết bởi vì phương pháp này phải biết trước toàn bộ mê cung. Các bước của phương pháp lấp kín đường cụt là:

1. Tìm tất cả điểm cụt trong mê cung
2. "Lấp kín" các con đường từ mỗi điểm cụt cho đến ngã giao đầu tiên.

Xem đoạn phim mô phỏng giải thuật này tại: [1] (<http://www.youtube.com/watch?v=yqZDYcpCGAI>)[2] (<http://www.youtube.com/watch?v=FkueaIT6RSU&NR=1>).

Xem thêm

- Mê cung
- Thuật toán tạo mê cung

Chú thích

- ¹ ^ Maze to Tree (<https://www.youtube.com/watch?v=k1tSK5V1pds>) trên YouTube
- ² ^ Maze Transformed (<https://www.youtube.com/watch?v=IIBwiGrUgzc>) trên YouTube
- ³ ^ Abelson; diSessa (1980), *Turtle Geometry: the computer as a medium for exploring mathematics*
- ⁴ ^ Public conference, December 2, 2010 - by professor Jean Pelletier-Thibert in Academie de Macon (Burgundy -

France) - (Abstract published in the Annals academic, March 2011 - ISSN: 0980-6032)

Charles Tremaux (° 1859 - † 1882) Ecole Polytechnique of Paris (X:1876), French engineer of the telegraph

5. ^ Édouard Lucas: *Récréations Mathématiques* Volume I, 1882.

6. ^ H. Fleischner: *Eulerian Graphs and related Topics*. In: *Annals of Discrete Mathematics* No. 50 Part 1 Volume 2, 1991, page X20.

7. ^ Even, Shimon (2011), *Graph Algorithms* (<http://books.google.com/books?id=m3QTSMYm5rkC&pg=PA46>) (ấn bản 2), Cambridge University Press, tr. 46–48, ISBN 978-0-521-73653-4.

8. ^ Sedgewick, Robert (2002), *Algorithms in C++: Graph Algorithms* (ấn bản 3), Pearson Education, ISBN 978-0-201-36118-6.

Liên kết ngoài

- Think Labyrinth: Maze algorithms (<http://www.astrolog.org/labyrnth/algrithm.htm#solve>) (details on these and other maze solving algorithms)
- MazeBlog: Solving mazes using image analysis (<http://www.cb.uu.se/~cris/blog/index.php/archives/277>)
- Maze generation and solving Java applet (<http://www.mazeworks.com/mazegen/>)
- Video: Maze solving simulation (<http://www.youtube.com/watch?v=jhL8uELbVIM>)

Lấy từ “http://vi.wikipedia.org/w/index.php?title=Thuật_toán_tìm_đường_đi_trong_mê_cung&oldid=16272618”

Thể loại: Thuật toán | Mê cung | Rô bốt học

-
- Trang này được sửa đổi lần cuối lúc 19:55, ngày 18 tháng 5 năm 2014.
 - Văn bản được phát hành theo Giấy phép Creative Commons Ghi công/Chia sẻ tương tự; có thể áp dụng điều khoản bổ sung. Xem Điều khoản Sử dụng để biết thêm chi tiết.
- Wikipedia® là thương hiệu đã đăng ký của Wikimedia Foundation, Inc., một tổ chức phi lợi nhuận.