ĐẠI HỌC KHOA HỌC TỰ NHIÊN

ĐẠI HỌC QUỐC GIA HÀ NỘI

THIẾT KẾ VÀ ĐÁNH GIÁ THUẬT TOÁN

Báo cáo:

CÁC PHƯƠNG PHÁP GIẢI BÀI TOÁN MÃ ĐI TUẦN

Người hướng dẫn : Nguyễn Thị Hồng Minh

Thành viên : Nguyễn Đức Mạnh - 20001563

Lê Viết Thịnh - 20001584

Hoàng Nhật Minh - 20001564

Đoàn Đức Trung - 20001587

Hanoi - 05/2023

Mục lục

Tóm tắt hướng đi của báo cáo

Thách thức của bài toán mã đi tuần là khi kích thước ô bàn cờ lớn, không gian tìm kiếm tăng vô cùng nhanh. Nếu chúng ta sử dụng lớp các phương pháp thử sai, thời gian hoàn thành công việc của thuật toán sẽ rất tệ, dù vậy các phương pháp thử sai giúp ta khẳng định được bài toán có lời giải hay không (không phải kích thước ô bàn cờ nào cũng tìm được lời giải) và rất hiệu quả với bàn cờ kích thước nhỏ. Điều này được nêu ra ở chương 2, các phương pháp cơ sở giải bài toán mã đi tuần. Cũng trong chương 2, báo cáo đề xuất ra một giải pháp khác để giải bài toán mã đi tuần với kích thước bàn cờ trong khoảng cho phép.

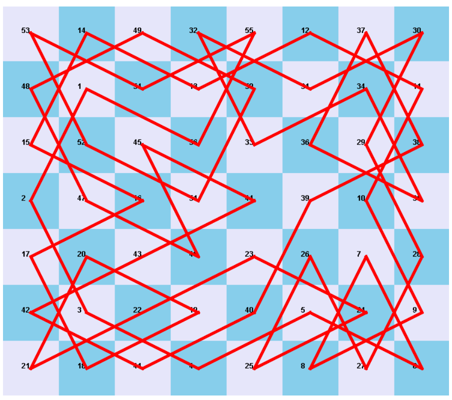
Từ các phương pháp cơ sở này, chúng tôi đã tham khảo và thực hiện chứng minh lại bằng thực nghiệm những định lý của nhà toán học Parberry I.An (tài liệu tham khảo) cho sự tồn tại đường đi đóng của bài toán mã đi tuần, được nêu ra ở chương 3.

m \ n	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
1	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
2	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
3	x	x	x	o	x	x	o	o	e	c	e	c	e	c	e	c	e	c	e	c
4	x	x	o	x	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o
5	x	x	x	o	e	c	e	c	e	c	e	c	e	c	e	c	e	c	e	c
6	x	x	x	o	c	c	c	c	c	c	c	c	c	c	c	c	c	c	c	c
7	x	x	o	o	e	c	e	c	e	c	e	c	e	c	e	c	e	c	e	c
8	x	x	o	o	c	c	c	c	c	c	c	c	c	c	c	c	c	c	c	c
9	x	x	e	o	e	c	e	c	e	c	e	c	e	c	e	c	e	c	e	c
10	x	x	c	o	c	c	c	c	c	c	c	c	c	c	c	c	c	c	c	c
11	x	x	e	o	e	c	e	c	e	c	e	c	e	c	e	c	e	c	e	c
12	x	x	c	o	c	c	c	c	c	c	c	c	c	c	c	c	c	c	c	c
13	x	x	e	o	e	c	e	c	e	c	e	c	e	c	e	c	e	c	e	c
14	x	x	c	o	c	c	c	c	c	c	c	c	c	c	c	c	c	c	c	c
15	x	x	e	o	e	c	e	c	e	c	e	c	e	c	e	c	e	c	e	c
16	x	x	c	o	c	c	c	c	c	c	c	c	c	c	c	c	c	c	c	c
17	x	x	e	o	e	c	e	c	e	c	e	c	e	c	e	c	e	c	e	c
18	x	x	c	o	c	c	c	c	c	c	c	c	c	c	c	c	c	c	c	c
19	x	x	e	o	e	c	e	c	e	c	e	c	e	c	e	c	e	c	e	c
20	x	x	c	o	c	c	c	c	c	c	c	c	c	c	c	c	c	c	c	c

x	không giải được
e	có thể giải được với một ô thiếu
o	có thể giải được hành trình mở
c	có thể giải được với hành trình đóng

Hình 1: Sự tồn tại đường đi của quân mã trong các bàn cờ có kích thước khác nhau

Sau khi đã có cái nhìn tổng quan về vấn đề, chúng tôi đi vào giải quyết bài toán trên bàn cờ cỡ lớn bằng phương pháp chia để trị ở chương 4 bằng các phương pháp cơ sở từ chương 2 và chia bàn cờ thành các bàn cờ nhỏ hơn sao cho chúng thỏa mãn điều kiện về kích thước ở chương 3.



Hình 2: Một lời giải chấp nhận được xây dựng từ các phương pháp cơ sở của chương 2

Chúng tôi thực hiện đánh giá độ phức tạp của thuật toán bằng các đường hồi quy thực nghiệm. Để có cái nhìn tổng quan hơn về vấn đề, báo cáo đã nêu ra những định nghĩa cơ bản ở chương mở đầu. Mã nguồn của dự án được viết bằng Java, vui lòng truy cập dự án tại đây.

Chương 1

Giới thiệu bài toán

Mã đi tuần (hay hành trình của quân mã) là bài toán về việc di chuyển một quân mã trên bàn cờ vua (8 x 8). Quân mã được đặt ở một ô trên một bàn cờ trống nó phải di chuyển theo quy tắc của cờ vua để đi qua mỗi ô trên bàn cờ đúng một lần. Trong bài này, chúng tôi xin giới thiệu về bài toán thú vị này và những điều có thể khai thác được qua bài toán.

Trong cờ vua, quân Mã là quân có cách đi phức tạp nhất. Xét một quân Mã đang đứng trên bàn cờ và tất cả các hình chữ nhật 2 x 3 nhận ô mà quân Mã đó đang đứng làm đỉnh. Quân Mã đó có thể đi tới các đỉnh khác màu với đỉnh nó đang đứng của bất kì hình chữ nhật 2 x 3 nào, miễn là đỉnh đó không nằm ngay cạnh đỉnh nó đang đứng.

Quân Mã có thể nhảy qua tất cả các quân khác để đến ô nó muốn, miễn là ô đó chưa bị ai chiếm giữ. Nói nôm na là quân Mã không bị cản. Khác với cờ tướng, nơi mà quân Mã có thể bị cản nếu có quân nào đứng ngay trước mặt nó, trong cờ vua, nước đi của quân Mã không có tính chất này. Khi một quân Mã đứng ở cạnh bàn cờ, số nước đi có thể của nó sẽ bị thu hẹp xuống còn nhiều nhất là một nửa số nước đi ban đầu. Đặc biệt, nếu nó đứng ở một trong bốn góc bàn cờ, nó chỉ đi được tối đa hai nước.

1.1 Xuất xứ bài toán

Bài toán quân Mã đi tuần là một dạng của bài toán tổng quát hơn là bài toán tìm đường đi Hamilton trong lý thuyết đồ thị, là một bài toán NP-đầy đủ. Bài toán tìm hành trình đóng của quân mã là một bài toán cụ thể của bài toán tìm chu trình Hamilton.

Nếu quân mã kết thúc hành trình của mình tại ô có thể đi tới ô xuất phát thì hành trình này được gọi là hành trình đóng. Bằng không nó là hành trình mở.

Các tài liệu tham khảo sớm nhất để giải quyết bài toán xuất hiện từ thế kỷ thứ IX. Nhưng giải thuật đầu tiên đầy đủ cho bài toán về hành trình của quân mã là “Giải thuật Warnsdorff” (Warndorff’s algorithm) công bố lần đầu tiên năm 1823 bởi H. C. Warnsdorff(Heinrich C. Warnsdorff- một nhà toán học người Đức). Thuật toán Warndorff được biết đến là phương pháp Heuristic của “Mã đi tuần”.

1.2 Mô tả bài toán

1. Xác định đầu vào(input)
- Một bảng vuông(bàn cờ) kích thước n*n
- Tọa độ xuất phát cảu quân mã N(x,y)
2. Xác định thông tin ra(Output)
- Một bàn cờ có đánh dấu vị trí từ 1(vị trí xuất phát) đến n*n(vị trí kết thúc) của quân mã.
- Từ vị trí thứ k sang vị trí thứ k+1 phải theo đúng luật di chuyển quân mã trong bàn cờ vua.
3. Xác định vị trí di chuyển từ 2 đến n*n cho quân mã
- Di chuyển đúng luật, mỗi ô chỉ được qua đúng một lần
- Vị trí cuối cùng (n*n) có thể nhảy tới vị trí xuất phát.

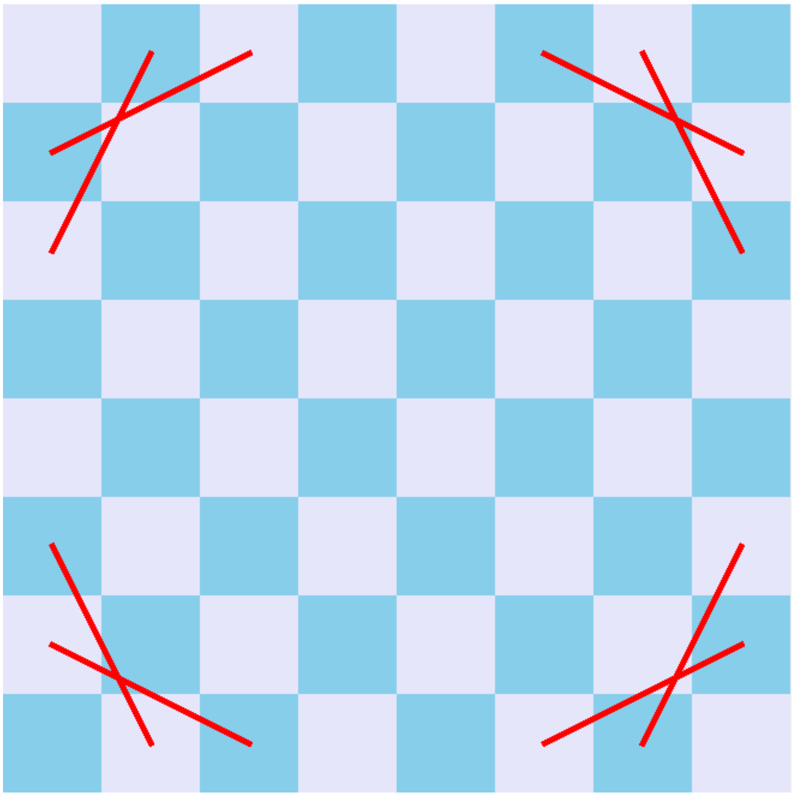
1.3 Các định nghĩa và thuật ngữ liên quan

1. Một đường đi của quân mã đi qua tất cả các ô được gọi là một hành trình của quân mã (Knight's Tours).

2. Nếu hành trình của quân mã kết thúc tại điểm mà tại điểm đó không thể đi trở lại điểm xuất phát thì đó được gọi là hành trình mở (Opened Knight's Tour). Ngược lại nếu ô kết thúc có thể đi trở lại điểm xuất phát thì đó được gọi là hành trình đóng (Closed Knight's Tour).

3. Các ô trong tập có thể các nước đi tiếp theo của quân mã được gọi là các hàng xóm (Neighbors), số lượng các hàng xóm được định nghĩa là bậc của đồ thị vô hướng tương ứng (Degree)

4. Một đường đi của quân mã mà ở đó tồn tại các đoạn đường như hình bên dưới được gọi là một đường đi có cấu trúc (Structured Path).



Hình 1.1: Đường đi có cấu trúc của quân mã trên bàn cờ 8x8

Chương 2

Các phương pháp cơ sở giải bài toán mã đi tuần

2.1 Thuật toán quay lui (backtracking)

2.1.1 Ý tưởng

Thuật toán quay lui thử tất cả các khả năng có thể của bài toán bằng cách xây dựng một cây tìm kiếm gồm tập tất cả các bước có thể xảy ra và duyệt qua tất cả các phần tử trong cây theo chiều sâu. Kiểm tra xem các giá trị đang duyệt đó có thỏa mãn điều kiện của bài toán hay không. Nếu thỏa mãn, tiếp tục thử các giá trị cho các biến tiếp theo. Nếu không thỏa mãn, quay lại trạng thái trước đó và thử các giá trị khác cho biến đó.

2.1.2 Áp dụng thuật toán để giải bài toán mã đi tuần

Ý tượng sử dụng phương pháp quay lui để giải bài toán mã đi tuần là:

1. Khởi tạo bàn cờ rỗng và danh sách các bước di chuyển hợp lệ của quân mã, được biểu diễn bằng tọa độ x và y.

2. Đặt quân mã vào một ô bất kỳ trên bàn cờ và đánh dấu ô đó là đã đi qua.

3. Tạo một hàm đệ quy để thực hiện việc di chuyển của quân mã. Hàm này nhận vào tọa độ hiện tại của quân mã và đếm số ô đã đi qua.

4. Trong hàm đệ quy:

Thử tối đa 8 ô xung quanh quân mã để tìm ra các bước di chuyển hợp lý, các bước này sẽ là nhánh của cây tìm kiếm trong thuật toán backtracking.

Lặp qua danh sách các bước di chuyển hợp lệ của quân mã. Kiểm tra từng bước di chuyển xem có thể di chuyển tới ô tiếp theo hay không:

Nếu ô tiếp theo chưa được đi qua, đánh dấu ô đó là đã đi qua và gọi đệ quy với ô tiếp theo và số ô đã đi qua tăng lên 1.

Nếu đệ quy trả về true, tức là đã tìm được lời giải, kết thúc thuật toán và trả về true.

Ngược lại, quay lại bước trước đó, đánh dấu ô đang xét là chưa đi qua và thử bước di chuyển khác.

Nếu số ô đã đi qua là bằng số ô trên bàn cờ, tức là đã tìm được lời giải. Kết thúc thuật toán và trả về true.

5. Nếu sau khi thử tất cả các bước di chuyển mà không tìm được lời giải, kết thúc thuật toán và trả về false.

2.1.3 Pseudocode

```
function solveKnightTour(board, currentRow, currentCol, moveCount):
    if moveCount = total number of cells on the board:
        return true

    for each valid move from the current position:
```

```
        if the move is valid and not visited before:
            mark the current cell as visited with moveCount
            if solveKnightTour(board, nextRow, nextCol, moveCount + 1) is true:
                return true
            unmark the current cell
    return false

function knightTour(boardSize, startRow, startCol):
    create an empty chessboard with size boardSize
    mark the start position as visited with 1
    if solveKnightTour(chessboard, startRow, startCol, 2) is true:
        print the chessboard
    else:
        print "No solution exists."
```

2.1.4 Tính khả thi của thuật toán

Sử dụng thuật toán quay lui để giải quyết bài toán mã đi tuần đơn giản là thử hết các trường hợp quân mã có thể đi. Nếu bài toán không tồn tại giải pháp thì quân mã sẽ đi hết tất cả các trường hợp và dừng lại. Nếu bài toán có giải pháp thì quân mã sẽ đi hết các ô trên bàn cờ và dừng lại. Trong trường hợp thuật toán không có lời giải, do ta đã thử qua tất cả các trường hợp có thể nên thuật toán là đúng.

2.1.5 Ưu điểm và hạn chế của thuật toán

Ưu điểm:

- Đơn giản và dễ hiểu: Thuật toán quay lui áp dụng giải bài toán mã đi tuần khá đơn giản và dễ hiểu. Nó sử dụng cách tiếp cận vét cạn (brute force) để thử tất cả các khả năng di chuyển cho mỗi ô trên bàn cờ.
- Tìm được lời giải nếu tồn tại: Nếu có lời giải cho bài toán mã đi tuần, thuật toán quay lui sẽ tìm ra một lời giải. Điều này đảm bảo tính đầy đủ của thuật toán.
- Hiệu quả cho bài toán không có lời giải: Thuật toán quay lui sẽ thử tất cả các khả năng di chuyển cho mỗi ô trên bàn cờ nên trong trường hợp không có lời giải, do thuật toán đã kiểm tra tất cả các khả năng nên kết quả nó trả lại là chính xác. Những bài toán không có lời giải đa phần có kích thước khá bé n = 1, 2, ... 5. Do vậy không gian tìm kiếm và thời gian thực thi của thuật toán trong các trường hợp này là chấp nhận được.

Nhược điểm

- Độ phức tạp thời gian lớn: Thuật toán quay lui áp dụng giải bài toán mã đi tuần có độ phức tạp thời gian lớn. Với bàn cờ kích thước N x N, độ phức tạp của thuật toán là $O(b^d)$ trong đó b(branch) là số lượng nhánh có thể được tạo ra tại mỗi bước và d(depth) là độ sâu của các bước. Cụ thể trong bài toán 8x8 độ phức tạp tính toán sẽ là $O(8^{64})$
- Thời gian thực thi tăng nhanh với kích thước: Khi kích thước bàn cờ tăng lên, thời gian thực thi của thuật toán quay lui cũng tăng nhanh chóng. Điều này là do số lượng các trạng thái trung gian cần kiểm tra tăng theo lũy thừa khi kích thước bàn cờ tăng.
- Không tối ưu hóa: Thuật toán quay lui không tối ưu hóa việc tìm kiếm, nghĩa là nó không thể loại bỏ các bước di chuyển không cần thiết. Do đó, thuật toán có thể thực hiện nhiều bước di chuyển không cần thiết và dẫn đến thời gian thực thi dài hơn.
- Rất khó khăn để tìm một hành trình đóng.

Đánh giá thuật toán quay lui trên thực nghiệm

Size	Time	Trials	Moves
36 (6x6)	27	43232	5422
42 (6x7)	84	6084478	760581
64 (8x8)	277	25936255	3242065
100(10x10)	>120000	...	...

- Thời gian chạy của thuật toán đối với bàn cờ kích cỡ nhỏ là rất nhanh, số bước thử (Trials) và số bước đi cho tới khi đi qua hết bàn cờ (Moves) cũng tăng rất nhanh.

- Thực nghiệm được thực hiện trên máy tính cá nhân cấu hình CPU intel core i3, ram 8Gb. Trên bàn cờ 10x10, việc chạy thuật toán là không khả thi.

2.2 Thuật toán Warnsdorff

Trong phần này chúng ta sẽ thảo luận về một thuật toán sử dụng phương pháp thiết kế Heuristic theo hướng tìm ra một lời giải có thể chấp nhận được trong thời gian ngắn. Thuật toán này giúp ta khắc phục được rất nhiều nhược điểm mà các phương pháp thử sai gặp phải.

2.2.1 Ý tưởng

Giải thuật Warnsdorff là một thuật toán giải quyết bài toán Mã đi tuần trên bàn cờ vua. Thuật toán này khá đơn giản với cách tiếp cận "lựa chọn đường đi ít lựa chọn nhất". Cụ thể, với mỗi ô trống trên bàn cờ, ta xét các ô kề cạnh chưa được đi qua và chọn ô kề cạnh đó có ít lựa chọn nhất để đi tiếp theo. Nếu có nhiều ô kề cạnh có số lựa chọn bằng nhau, chọn ô có tương lai ít lựa chọn nhất. Thuật toán sẽ lặp lại quá trình này cho đến khi đi hết các ô trống trên bàn cờ hoặc không thể đi tiếp

2.2.2 Áp dụng thuật toán để giải bài toán mã đi tuần

1. Tạo một bàn cờ với kích thước NxN, và đặt quân mã vào ô bất kỳ trên bàn cờ.
2. Tại vị trí hiện tại của quân mã, xác định tất cả các ô trống mà quân mã có thể di chuyển đến. Ghi nhớ số lượng ô trống mà quân mã có thể di chuyển đến từ mỗi ô đó.
3. Từ danh sách bước tiếp theo, chọn ô trống có bậc ít nhất làm mục tiêu di chuyển tiếp theo. Điều này giúp tăng khả năng thành công của thuật toán. Nếu có nhiều hơn 1 ô có bậc nhỏ nhất, ta chọn ngẫu nhiên một trong các ô đó. Điều này cũng rất quan trọng trong việc tạo ra các kết quả khác nhau tại những lần chạy khác nhau.
4. Di chuyển quân mã đến ô được chọn là bước đi tiếp theo. Đánh dấu ô đó đã được đi qua.
5. Tiếp tục lặp lại các bước 2-4 cho đến khi quân mã đã đi qua tất cả các ô trên bàn cờ, hoặc không còn bước đi nào khả dụng.
6. Kiểm tra xem quân mã đã đi qua tất cả các ô trên bàn cờ hay chưa. Nếu đã đi qua tất cả các ô, thuật toán kết thúc và trả về lời giải. Nếu không, thuật toán trả về không có lời giải.

2.2.3 Pseudocode

```
procedure Warnsdorff(N, start_x, start_y):
create an NxN chessboard and initialize all cells as unvisited
mark the starting cell (start_x, start_y) as visited with the value 1
set the current cell as the starting cell
set the move number as 2

while there are unvisited cells:
    generate a list of valid moves from the current cell
    initialize an empty list of neighbor counts

    for each valid move in the list of moves:
        count the number of unvisited neighbors for the current move
        add the count to the list of neighbor counts

    find the move with the minimum neighbor count from the list of moves
    make a move with the minimum neighbor count as the next move
    Mark the next move as visited with the current move number
    increment the move number

    set the current cell as the next move cell
```

```
if all cells are visited:
    print "The final Knight's Tour Path"
else:
    print "No solution exists "
```

2.2.4 Ưu điểm và hạn chế của thuật toán

Ưu điểm:

- Thuật toán Warnsdorff dễ hiểu và triển khai.
- Có thể giải được bài toán với bàn cờ cỡ lớn.
- Thời gian thực thi của thuật toán khá nhanh và gần như là xấp xỉ $O(n^2)$ với những bàn cờ cỡ nhỏ.
- Không đảm bảo tìm được lời giải ngay lần chạy đầu nhưng có thể tìm được lời giải thỏa mãn chấp nhận được sau chỉ vài bước thử.
- Do các cạnh ô có bậc thấp thường nằm ở viền của bàn cờ, vì vậy xác suất để tạo ra một hành trình có cấu trúc là rất cao. Chúng ta sẽ lợi dụng ưu điểm này để thực hiện một phương pháp khác được đề cập ở phần sau đó chính là chia để trị.

Hạn chế:

- Chỉ giải được bài toán trong phạm vi nhất định, đối với những bàn cờ đặc biệt mà không thể tìm ra lời giải trong thời gian ngắn, thuật toán trở lại thành lớp các phương pháp thử sai.
- Hiệu suất của thuật toán giảm đi đáng kể khi kích thước bàn cờ tăng lên, vì số lượng các ô lân cận cần kiểm tra cũng tăng lên đáng kể.

2.2.5 Tính khả thi của thuật toán

Nếu ta biểu diễn đường đi của quân mã dưới dạng một đồ thị vô hướng, những ô có nhiều lựa chọn di chuyển sẽ là những điểm có bậc của đồ thị lớn. Đường đi của thuật toán backtracking thực hiện phần lớn sẽ bị "cụt"khi đi vào một ô có bậc là 1. Thay vào đó thuật toán Warnsdorff đi vào những ô có bậc thấp nhất trước, những có bậc cao hơn sẽ được đi cuối cùng để đảm bảo không xảy ra tình trạng bị "cụt"đường như trong các thuật toán backtracking.

Thuật toán có độ phức tạp $O(n^2)$, trong trường hợp thuận lợi ta chỉ cần chạy thuật toán duy nhất một lần, nếu không tìm được nghiệm ta chạy lại thuật toán một lần nữa. Vì trong thuật toán có yếu tố ngẫu nhiên, nên ở lần chạy thứ 2 cho ra kết quả khác lần chạy thứ nhất.

Thực nghiệm

- Ta sẽ thực nghiệm chứng minh tính khả thi của thuật toán trên bàn cờ nhỏ kích cỡ dưới 20 ô bằng cách kiểm tra xem thuật toán có trả lại kết quả hay không, và kết quả đó được trả lại sau bao nhiêu lần thực hiện.
- Sau các lần thực nghiệm và tổng hợp từ các tài liệu tham khảo, chúng tôi thu được các kết quả là những trường hợp có thể xảy ra như sau:
1. Bài toán không thể giải được
2. Bài toán có thể giải được với một ô trống không được đi qua.
3. Bài toán có thể giải với hành trình mở.
4. Bài toán có thể giải được với hành trình đóng.
- Chi tiết của thực nghiệm sẽ được trình bày ở chương sau.

Chương 3

Định lý về sự tồn tại đường đi trong bài toán mã đi tuần

Nội dung định lý và chứng minh được tham khảo từ nghiên cứu của nhà toán học Parberry (phần tài liệu tham khảo) và đã được thực nghiệm lại bởi nhóm chúng tôi.

3.0.1 Định lý 3.1

Với mọi n chẵn sao cho $n >= 6$, luôn tồn tại một đường đi có cấu trúc trên bàn cờ kích thước n x n và n x (n+2).

3.0.2 Hệ quả 3.2

Với mọi $n >= 6$ luôn tồn tại một đường đi có cấu trúc trên bàn cờ n x (n + 1).

3.0.3 Định lý 3.3

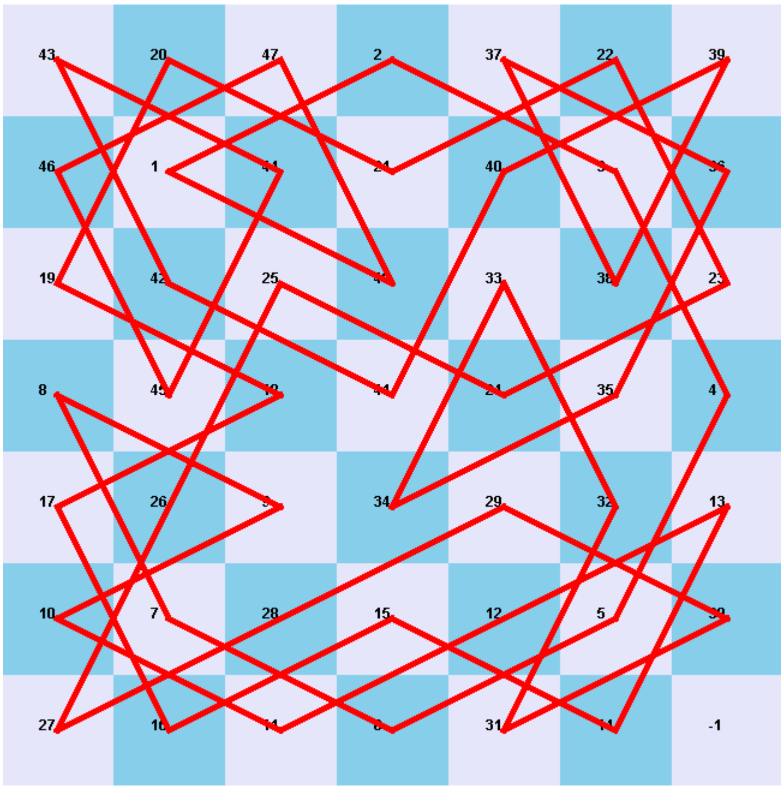
Với mọi n lẻ $n >= 5$ luôn tồn tại một đường đi có cấu trúc trên bàn cờ n x n sao cho chúng khuyết một góc.

Kết quả của định lý này được chúng tôi thử nghiệm lại trên kích cỡ khác nhau của bàn cờ.

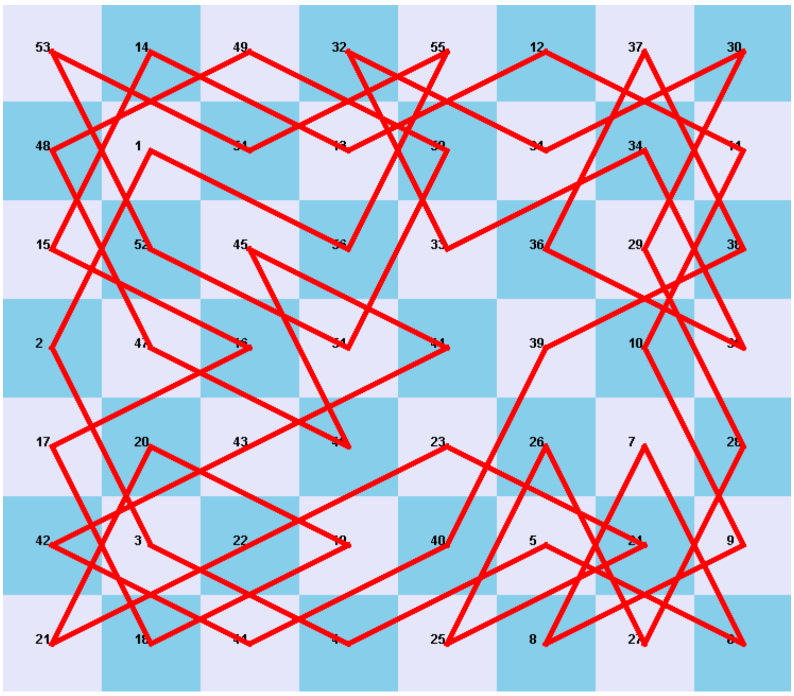
	n	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
m																					
1		x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
2		x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
3		x	x	x	o	x	x	o	o	e	c	e	c	e	c	e	c	e	c	e	c
4		x	x	o	x	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o	o
5		x	x	x	o	e	c	e	c	e	c	e	c	e	c	e	c	e	c	e	c
6		x	x	x	o	c	c	c	c	c	c	c	c	c	c	c	c	c	c	c	c
7		x	x	o	o	e	c	e	c	e	c	e	c	e	c	e	c	e	c	e	c
8		x	x	o	o	c	c	c	c	c	c	c	c	c	c	c	c	c	c	c	c
9		x	x	e	o	e	c	e	c	e	c	e	c	e	c	e	c	e	c	e	c
10		x	x	c	o	c	c	c	c	c	c	c	c	c	c	c	c	c	c	c	c
11		x	x	e	o	e	c	e	c	e	c	e	c	e	c	e	c	e	c	e	c
12		x	x	c	o	c	c	c	c	c	c	c	c	c	c	c	c	c	c	c	c
13		x	x	e	o	e	c	e	c	e	c	e	c	e	c	e	c	e	c	e	c
14		x	x	c	o	c	c	c	c	c	c	c	c	c	c	c	c	c	c	c	c
15		x	x	e	o	e	c	e	c	e	c	e	c	e	c	e	c	e	c	e	c
16		x	x	c	o	c	c	c	c	c	c	c	c	c	c	c	c	c	c	c	c
17		x	x	e	o	e	c	e	c	e	c	e	c	e	c	e	c	e	c	e	c
18		x	x	c	o	c	c	c	c	c	c	c	c	c	c	c	c	c	c	c	c
19		x	x	e	o	e	c	e	c	e	c	e	c	e	c	e	c	e	c	e	c
20		x	x	c	o	c	c	c	c	c	c	c	c	c	c	c	c	c	c	c	c

x	không giải được
e	có thể giải được với một ô thiếu
o	có thể giải được hành trình mở
c	có thể giải được với hành trình đóng

Hình 3.1: Sự tồn tại đường đi của quân mã trong các bàn cờ có kích thước khác nhau



Hình 3.2: Hành trình đóng có cấu trúc được mô tả ở định lý 3.3



Hình 3.3: Hành trình đóng có cấu trúc được mô tả ở hệ quả 3.2

Chương 4

Sử dụng thuật toán chia để trị giải bài toán mã đi tuần trên bàn cờ có kích thước lớn

4.1 Thuật toán chia để trị(Divide and Conquer)

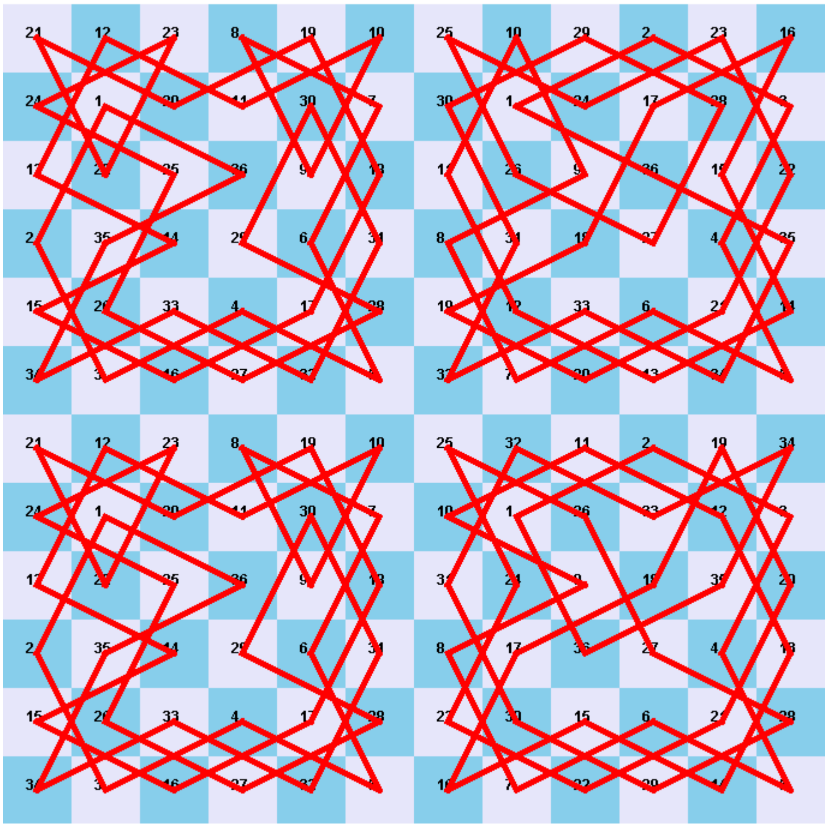
4.1.1 Ý tưởng

Ý tưởng của thuật toán chia để trị là chia chia bài toán cần giải quyết thành các bài toán con nhỏ hơn. Tiếp tục chia cho đến khi các bài toán nhỏ này không thể chia thêm nữa, khi đó ta sẽ giải quyết các bài toán nhỏ nhất này và cuối cùng kết hợp giải pháp của tất cả các bài toán nhỏ để tìm ra giải pháp của bài toán ban đầu. Trong khi đó các bài toán nhỏ được giải bằng các phương pháp cơ sở đã nêu ở các chương trên. Nhờ đó mà ta giải được bài toán trên bàn cờ với kích thước lớn.

4.1.2 Áp dụng thuật toán để giải bài toán mã đi tuần

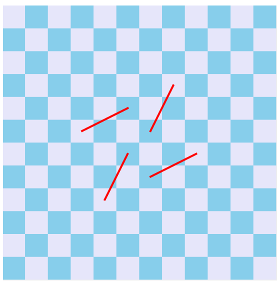
1. Đặt bài toán thành bài toán con nhỏ hơn: Bắt đầu bằng cách chia bàn cờ thành các ô nhỏ hơn.

2. Tìm giải pháp cho các bài toán con: Áp dụng thuật toán mã đi tuần cho các bài toán con nhỏ hơn. Nếu bài toán con nhỏ hơn có kích thước đủ nhỏ, có thể tìm một giải pháp trực tiếp. Nếu không, có thể tiếp tục chia bài toán con thành các bài toán con nhỏ hơn. Vì tính hợp lý của giải thuật Warndorff các bài toán con được chia nhỏ ra luôn tạo thành các đường đi đóng có cấu trúc.

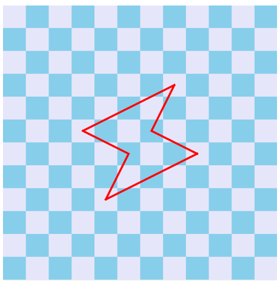


Hình 4.1: Chia bàn cờ 12x12 thành 4 bàn cờ 6x6

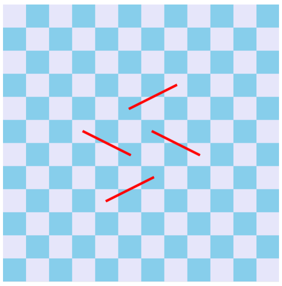
3. Kết hợp các giải pháp con: Sau khi có các giải pháp cho các bài toán con, cần kết hợp chúng để tạo ra một giải pháp cho bài toán ban đầu. Trong trường hợp của bài toán mã đi tuần, có thể sắp xếp các bước đi của mã từ các bài toán con để tạo thành một chuyến đi hoàn chỉnh cho mã đi tuần trên bàn cờ ban đầu.



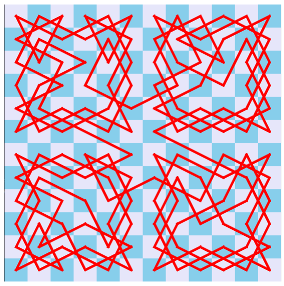
Hình 4.2: Do đường đi là có cấu trúc nên tồn tại các cạnh trên



Hình 4.3: Thêm các đường đi kết nối phần còn lại của ô bàn cờ



Hình 4.4: Xóa bỏ các cạnh thừa là các cạnh ban đầu (hình 4.2)



Hình 4.5: Đường đi mã đi tuần trên bàn cờ 12x12 hoàn chỉnh

Hình 4.6: Quá trình kết hợp các giải pháp con

4.1.3 Pseudocode

```
function isSafe(x, y, board):
    return (x >= 0 and x < N and y >= 0 and y < N and board[x][y] == -1)

function solveKT(board, x, y, moveX, moveY, moveCount):
```

```
    if moveCount == N^2:
        return true

    for i in range(8):
        nextX = x + moveX[i]
        nextY = y + moveY[i]
        if isSafe(nextX, nextY, board):
            board[nextX][nextY] = moveCount
            if solveKT(board, nextX, nextY, moveX, moveY, moveCount + 1):
                return true
            else:
                board[nextX][nextY] = -1

    return false

function solveKnightTour(N):
    board = createEmptyBoard(N)
    moveX = [2, 1, -1, -2, -2, -1, 1, 2]
    moveY = [1, 2, 2, 1, -1, -2, -2, -1]
    board[0][0] = 0

    if solveKT(board, 0, 0, moveX, moveY, 1):
        printBoard(board)
    else:
        print("No solution exists.")

function createEmptyBoard(N):
    return [[-1] * N for _ in range(N)]

function printBoard(board):
    for row in board:
        for cell in row:
            print(cell, end="\t")
        print()
```

Trong đoạn mã trên, chúng ta sử dụng hàm solveKnightTour(N) để giải bài toán mã đi tuần cho một bàn cờ có kích thước N x N. Thuật toán chia để trị được áp dụng trong hàm solveKT, trong đó chúng ta sử dụng đệ quy để thử tất cả các nước đi tiếp theo có thể từ vị trí hiện tại. Nếu tìm được giải pháp, chúng ta in ra bàn cờ kết quả bằng hàm printBoard, ngược lại, thông báo rằng không có giải pháp tồn tại.

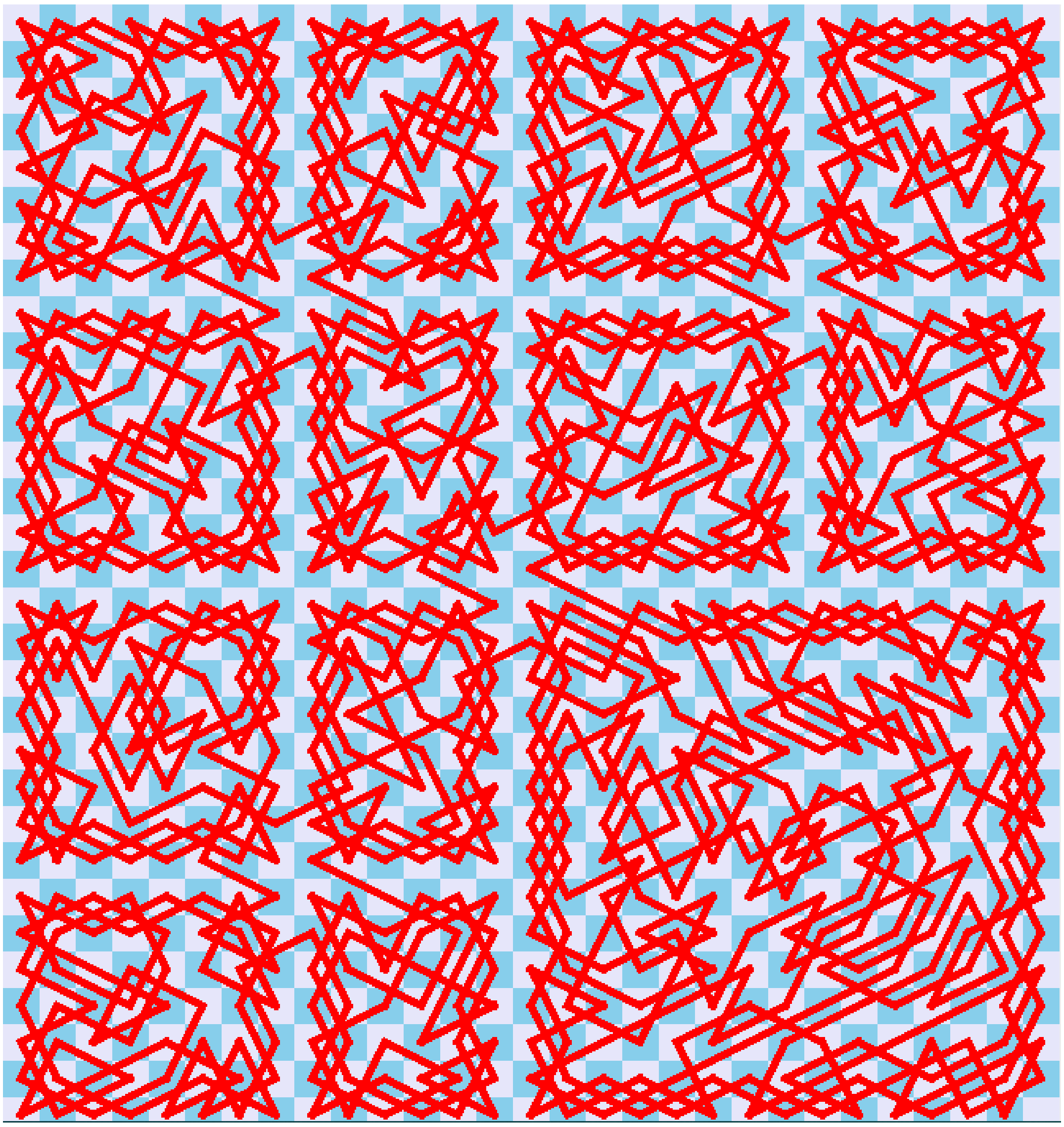
4.1.4 Chứng minh tính đúng

Ta sử dụng phương pháp quy nạp để chứng minh: 1. Ta chứng minh rằng thuật toán đúng với kích thước bàn cờ nhỏ (5->20), điều này đã được chỉ ra ở chương 3, bài toán được giải bằng giải thuật warndorffs và thực nghiệm cho thấy xác xuất để tìm được lời giải sau ít nhất 3 bước lặp là 98%. Sau số bước lặp nhất định, ta luôn luôn tìm được lời giải hợp lý cho bàn cờ cỡ (5->20).

2. Giả định rằng thuật toán đúng cho bàn cờ kích thước n x n với n chẵn.

3. Ta cần chứng minh rằng thuật toán cũng đúng cho bàn cờ kích thước (n + 2) x (n + 2), n x (n + 1), n x (n + m). Bước chứng minh này đã được nhà toán học Parberry trình bày trong nghiên cứu của ông.

Lưu ý: trong bước chia để trị, ta cần tránh việc chia bàn cờ ra thành các ô có kích thước đều lẻ như 7x7, 9x7,... Nếu gặp một ô có kích thước 14x14, thay vì chia thành 4 ô 7x7, ta chia bàn cờ thành 6x8, 6x6, 8x6, 8x8, các ô trên đều tồn tại đường đi và có thể giải được trong thời gian $O(n^2)$



Hình 4.7: Bàn cờ kích thước 31x29 áp dụng theo cách tránh chia bàn cờ thành các ô có kích thước lẻ

4.1.5 Ưu điểm và nhược điểm của thuật toán

Ưu điểm

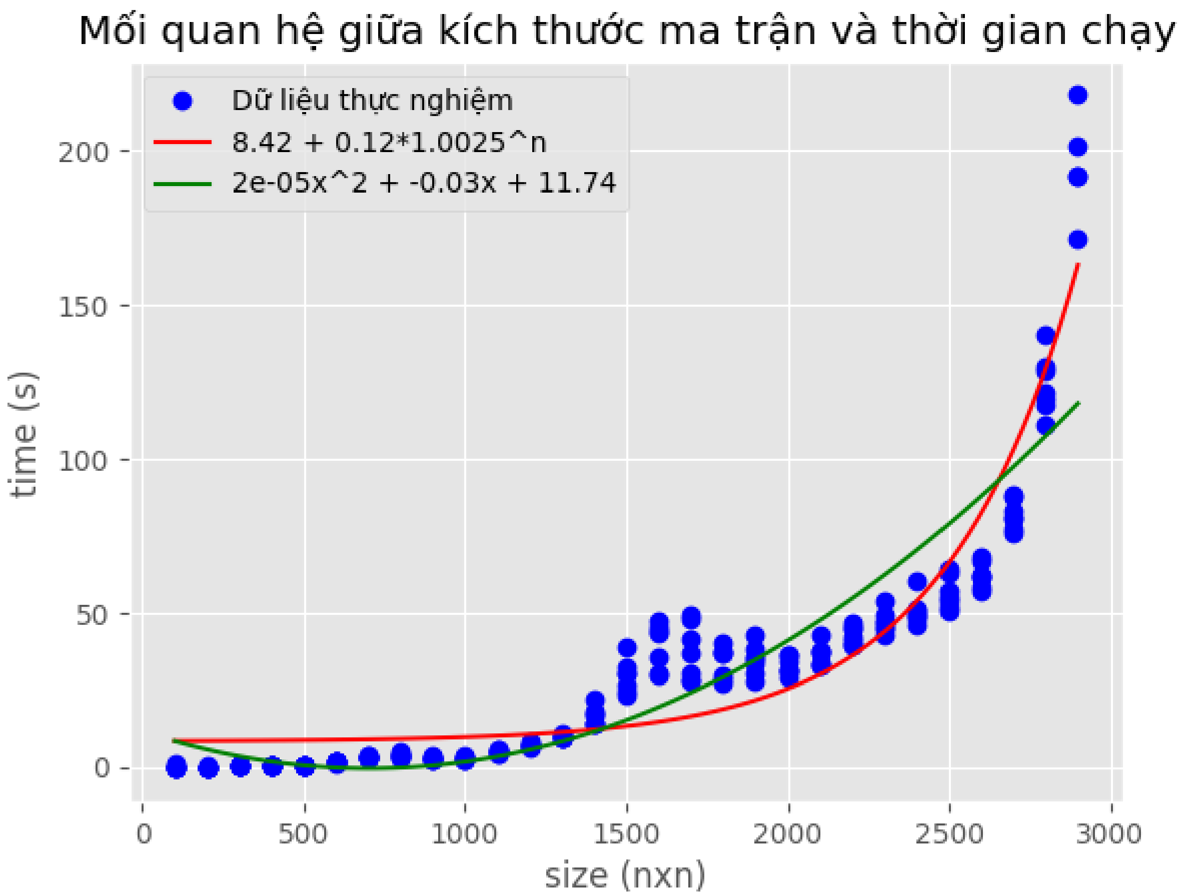
- Tính chất đệ quy: Thuật toán chia để trị thường được xây dựng dựa trên cách tiếp cận đệ quy, cho phép chia bài toán ban đầu thành các bài toán con nhỏ hơn. Trong trường hợp của bài toán mã đi tuần, thuật toán chia để trị cho phép chia bàn cờ lớn thành các bàn cờ con nhỏ hơn, giúp giảm độ phức tạp của bài toán.
- Tối ưu hóa tìm kiếm: Thuật toán chia để trị cho phép loại bỏ các khả năng không hợp lệ và tối ưu hóa tìm kiếm trong quá trình giải quyết bài toán mã đi tuần. Bằng cách thử từng bước di chuyển hợp lệ và loại bỏ những bước không thể tiếp tục, thuật toán giảm số lần thử nghiệm và tìm kiếm, giúp cải thiện hiệu suất của thuật toán.
- Độ phức tạp thời gian tương đối tốt: Thuật toán chia để trị cho bài toán mã đi tuần có độ phức tạp thời gian khá tốt trong trường hợp bàn cờ có kích thước bất kì. Trong các trường hợp này, thuật toán có thể tìm ra giải pháp trong một thời gian hợp lý.

Nhược điểm

- Khả năng không tìm ra giải pháp: Thuật toán chia để trị không đảm bảo tìm ra giải pháp cho tất cả các trường hợp của bài toán mã đi tuần. Điều này có thể xảy ra khi bàn cờ có kích thước quá lớn hoặc có các ràng buộc đặc biệt không thể đáp ứng. Điều này cũng có thể xảy ra khi kích cỡ bàn cờ vượt ra khỏi giới hạn xử lý của thuật toán warndorffs.
- Sử dụng không hiệu quả tài nguyên: Thuật toán chia để trị có thể yêu cầu lưu trữ và sử dụng tài nguyên lớn trong quá trình giải quyết bài toán mã đi tuần. Khi bàn cờ có kích thước lớn, việc lưu trữ và xử lý thông tin của các bước di chuyển trở nên phức tạp và yêu cầu nhiều tài nguyên hơn.

4.1.6 Thực nghiệm đánh giá độ phức tạp của thuật toán chia để trị

Chạy thuật toán 7 lần trên dữ liệu khác nhau, ta thu được biểu đồ scatter cho các điểm dữ liệu. Thoạt nhìn có thể thấy biểu đồ tăng trưởng bậc 2 hoặc tăng theo hàm số mũ. Chúng tôi thực hiện vẽ đường hồi quy dựa trên bộ dữu liệu đã cho trên hai mô hình hồi quy khác nhau. Có thể thấy mô hình tăng trưởng mũ khớp với thời gian chạy của thuật toán hơn mô hình bậc 2.



Chương 5

Kết luận

Nghiên cứu thực hiện phân tích, thiết kế và đánh giá các thuật toán giải quyết bài toán này. Ban đầu, thuật toán backtracking được sử dụng để tìm kiếm lời giải, tuy nhiên, độ phức tạp của thuật toán này tăng nhanh khi kích thước bàn cờ tăng lên. Để cải thiện hiệu suất, thuật toán Warnsdorff, một thuật toán heuristic, được sử dụng để đưa ra kết quả chấp nhận được trong thời gian hợp lý. Tuy nhiên, giới hạn của thuật toán này là chỉ áp dụng được cho các bàn cờ có kích thước nhỏ.

Để giải quyết bài toán mã đi tuần trên các bàn cờ có kích thước lớn, chúng tôi đã sử dụng phương pháp chia để trị kết hợp với thuật toán Warnsdorff. Kết quả thực nghiệm đã cho thấy rằng phương pháp này cho kết quả tốt và có thể áp dụng cho các bàn cờ có kích thước lớn. Tuy nhiên, do độ phức tạp của bài toán mã đi tuần, việc tìm kiếm lời giải chính xác vẫn đòi hỏi thời gian và tài nguyên đáng kể.

Phân tích độ phức tạp của thuật toán mã đi tuần là một thách thức. Độ phức tạp thời gian và không gian của thuật toán mã đi tuần phụ thuộc vào kích thước của bàn cờ. Vì số lượng ô trên bàn cờ tăng theo cấp số nhân, việc đánh giá độ phức tạp dựa trên master theorem hoặc các phương pháp phân tích thông thường khác trở nên khó khăn. Thay vào đó, chúng tôi đã thực hiện thử nghiệm và đánh giá thời gian chạy của thuật toán trên các bàn cờ có kích thước khác nhau.

Trong tương lai, chúng tôi đề xuất nghiên cứu và phát triển các phương pháp giải quyết bài toán mã đi tuần với độ phức tạp thấp hơn, nhằm tăng hiệu suất và mở rộng khả năng giải quyết của thuật toán trên các bàn cờ có kích thước lớn hơn. Đồng thời, chúng tôi cũng đề xuất áp dụng các phương pháp tối ưu hóa để cải thiện thuật toán hiện tại và tìm kiếm lời giải tốt nhất cho bài toán mã đi tuần.

Tổng kết lại, nghiên cứu về bài toán mã đi tuần đã mang lại những hiểu biết quan trọng về phân tích, thiết kế và đánh giá thuật toán. Mặc dù còn nhiều thách thức và giới hạn, nhưng bài toán mã đi tuần vẫn là một bài toán thú vị và có ứng dụng trong nhiều lĩnh vực.

Tài liệu tham khảo

[Parberry, I.An efficient algorithm for the Knight's tour problem. Discrete Applied Mathematics. (1997)]
.

[Gerlach, John R., and F. L. Cape Coral.The Knight's Tour in Chess–Implementing a Heuristic Solution.(2015)]
.

[Người, B. T. H. C. Ml. BÀI TOÁN "QUÂN MÃ ĐI TUẦN" VÀ NHỮNG ĐIỀU THÚ VỊ ẨN SAU NÓ.2017))]
Spiderum.

[Bài toán quân mã đi tuần và những điều thú vị ẩn sau nó | algorithm. (n.d.)]