

Cải tiến thuật toán Ant Colony giải quyết bài toán người bán hàng (TSP)

Improve the Ant Colony algorithm to solve travelling salesman problem

Lê Thị Ngọc Vân*, Nguyễn Dũng, Trần Huệ Chi
Thi Ngoc Van Le, Dung Nguyen, Hue Chi Tran

*Khoa Công nghệ Thông tin, Trường Đại học Duy Tân, Đà Nẵng, Việt Nam
Faculty of Information Technology, Duy Tan University, Da Nang, Vietnam*

(Ngày nhận bài: 09/09/2019, ngày phản biện xong: 03/12/2019, ngày chấp nhận đăng: 20/12/2019)

Tóm tắt

Bài báo đề xuất cách cải tiến thuật toán Ant Colony để hỗ trợ tìm ra đường đi ngắn hơn cho bài toán người bán hàng. Bài toán người bán hàng yêu cầu tìm ra đường đi ngắn nhất cho người bán hàng đi qua các thành phố và cuối cùng quay về lại thành phố xuất phát, mỗi thành phố chỉ được ghé thăm một lần, biết rằng tất cả các thành phố đều có đường đi đến với nhau và khoảng cách giữa các thành phố là biết trước. Có rất nhiều thuật toán giải quyết được bài toán này. Một trong những thuật toán được nghiên cứu nhiều và giải quyết khá hiệu quả cho bài toán này là thuật toán Ant Colony (thuật toán đàn kiến). Thuật toán Ant Colony có những hỗ trợ tìm kiếm rất mạnh mẽ và tỏ ra khá thích hợp với những bài toán có không gian tìm kiếm cực lớn. Tuy nhiên khi áp dụng thuật toán Ant Colony cho bài toán người bán hàng thì chi phí vẫn còn khá cao. Vì vậy nhóm chúng tôi thiết kế giải thuật cải tiến thuật toán Ant Colony để tìm lời giải tối ưu hơn cho bài toán người bán hàng. Chúng tôi đã tiến hành xây dựng thử nghiệm với nhiều bộ dữ liệu đầu vào để so sánh giữa thuật toán cải tiến với thuật toán Ant Colony nhằm đánh giá một cách chính xác và khách quan nhất. Kết quả cho thấy thuật toán cải tiến của nhóm chúng tôi đề xuất đã có cải thiện đáng kể về chi phí so với thuật toán Ant Colony.

Từ khóa: Ant Colony, thuật toán cải tiến Ant Colony, bài toán người bán hàng.

Abstract

This article proposes a way to improve the Ant Colony algorithm to assist in finding a shortest path for the Travelling Salesman Problem. The aim is to find the shortest path for sellers to go through cities and finally return to the starting point, each city is visited only once, given that all the cities are interconnected and the distances among the cities are given. There are many algorithms to solve this problem. One of the algorithms that has been studied and proved to be effective for this problem is the Ant Colony Optimization (ACO). ACO has very strong search support and seems suitable for problems with extremely large search space. However, when applying ACO to the salesman problem, it is still quite expensive. Therefore, our team designed an algorithm to improve ACO to find a better solution to the travelling salesman problem. We have built a test with multiple data sets to compare the proposed algorithm with the ACO to evaluate accurately and objectively. The results showed that our improved algorithm has significantly improved the cost compared to ACO.

Keywords: Ant Colony, improved Ant Colony algorithm, Travelling Salesman Problem.

1. Giới thiệu

Lý thuyết về thuật toán Ant Colony và ứng dụng thuật toán đã được nhiều tác giả trong và ngoài nước nghiên cứu. Các tác giả Marco Dorigo, Thomas Stützle ở trường đại học The MIT Press Cambridge, Massachusetts London đã đưa ra cách tiếp cận và lý thuyết để ứng dụng thuật toán Ant Colony vào bài toán người bán hàng (TSP) [1]. Trong bài báo của tác giả Phạm Hồng Luân và Dương Thành Nhân, Tạp chí Phát triển Khoa học và Công nghệ (Journal of Science and Technology Development), ISSN: 1859-0128 đã đưa ra thuật toán Ant Colony và ứng dụng thuật toán vào quản lý chi phí dự án xây dựng [2]. Có thể nói, nghiên cứu lý thuyết về thuật toán Ant Colony đã rất hoàn thiện. Bài báo này đề xuất một cách cải tiến thuật toán Ant Colony để có thể tìm được đường đi ngắn hơn đường đi mà thuật toán Ant Colony tìm ra.

Ngoài phần giới thiệu và lịch sử nghiên cứu của vấn đề ở phần 1, phần 2 sẽ trình bày về bài toán người bán hàng và các giải thuật giải quyết bài toán. Phần 3 trình cách cải tiến thuật toán Ant Colony. Kết quả nghiên cứu và kết luận sẽ là nội dung của phần 4.

2. Bài toán người bán hàng (TSP) và các giải thuật giải quyết bài toán

2.1 Bài toán người bán hàng (Traveling Salesman Problem)

Bài toán yêu cầu tìm đường đi ngắn nhất cho nhân viên bán hàng (traveling salesman), nhân viên bán hàng xuất phát từ một thành phố, đi qua lần lượt tất cả các thành phố có trong lộ trình duy nhất một lần và quay về thành phố ban đầu với chi phí thấp nhất.

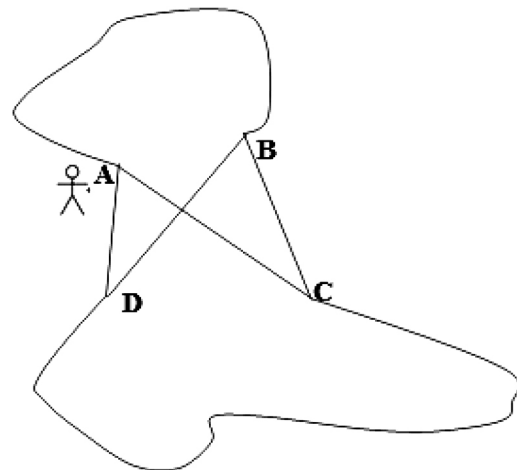
Ví dụ ở Hình 1 bài toán yêu cầu tìm đường đi ngắn nhất cho người bán hàng xuất phát từ thành phố A và cần đi qua tất cả các thành phố B,C,D sau đó quay về thành phố A, tất cả các thành phố đều có đường đi đến các thành phố còn lại.

Nếu người bán hàng xuất phát từ thành phố A,

và khoảng cách giữa hai thành phố bất kì đã biết trước thì đường đi ngắn nhất mà người bán hàng có thể thực hiện được sao cho đi hết tất cả các thành phố, mỗi thành phố một lần để quay về lại thành phố A ban đầu là như thế nào?

Bài toán này được đề xuất giải quyết vào thế kỷ thứ XVII bởi hai nhà toán học người Anh là Sir William Rowan Hamilton và Thomas Penyngton Kirkman, và được ghi chép trong giáo trình Lý thuyết đồ thị nổi tiếng của Oxford. Sau đó bài toán trở thành bài toán khó thách thức toàn thể các nhà toán học và tin học trên thế giới bởi độ phức tạp thuật toán tăng theo hàm số mũ (thuộc lớp bài toán NP-khó).

Các nhà khoa học bắt đầu tiếp cận bài toán, dùng máy tính thử nghiệm tính toán và công bố các kết quả giải bài toán này từ năm 1954 với số thành phố là 49, và năm 2004 bài toán giải được với số thành phố là 24.978, và số lượng thành phố càng ngày càng tăng cao [3].



Hình 1. Bài toán người bán hàng

2.2 Giải thuật Ant Colony

Bài toán người bán hàng hoàn toàn có thể giải bằng giải thuật Ant Colony, thuật toán này có thể áp dụng cho nhiều lớp bài toán. Để có thể hiểu hơn về thuật toán này chúng ta có thể tìm hiểu đặc điểm hoạt động trong tự nhiên của đàn kiến và từ đó có thể hiểu hơn về thuật toán Ant Colony.

Trong tự nhiên loại kiến hoạt động theo quy luật sau: Khi tìm kiếm thức ăn, trong quá trình

di chuyển kiến thường để lại mùi riêng của nó để các con kiến trong đàn có thể nhận diện ra lối di chuyển của chúng, mùi của kiến để lại trên đường đi đó được gọi là pheromone. Đây là yếu tố giúp đàn kiến đánh dấu và trao đổi thông tin với nhau khi chúng đi tìm nguồn thức ăn. Khi tìm thấy nguồn thức ăn, mỗi con kiến sẽ di chuyển từ tổ tới nơi có nguồn thức ăn theo đường đi đã tìm thấy. Từ tổ kiến đến nơi có thức ăn sẽ có nhiều đường đi. Vì thế ban đầu các chú kiến sẽ đi ngẫu nhiên theo các con đường từ điểm xuất phát đến đích. Tuy nhiên, sau một thời gian di chuyển nhất định, các chú kiến trong đàn sẽ tìm ra được đường đi ngắn nhất từ tổ tới nguồn thức ăn. Sau khi quan sát và nghiên cứu hoạt động của đàn kiến trong tự nhiên, chúng ta nhận thấy rằng quá trình tìm đường đi ngắn nhất từ tổ tới nguồn thức ăn dựa trên các quy luật sau:

Khi di chuyển kiến sẽ dùng pheromone để đánh dấu và trao đổi thông tin với nhau nhằm tìm ra đường đi ngắn nhất từ tổ đến nguồn thức ăn, đồng thời trong quá trình di chuyển đó chúng để lại một lượng mùi (pheromone) trên đường chúng di chuyển qua.

Các con kiến đi sau sẽ dựa vào lượng mùi mà các con kiến đi trước để lại để tiến hành tìm đường đi cho mình, khi trên đường đi chưa có mùi để lại thì chúng di chuyển một cách ngẫu nhiên, lượng mùi để lại trên đường đi của các con kiến di chuyển trước cũng bị bay hơi theo thời gian.

Đường đi nào có lượng thông tin mùi để lại càng nhiều thì xác suất được các con kiến chọn càng cao, ngược lại đường đi có lượng mùi thấp thì xác suất chọn càng thấp.

Qua một quá trình di chuyển đàn kiến sẽ chọn được cho mình một đường đi ngắn nhất từ tổ đến nguồn thức ăn.

Dựa vào cơ chế hoạt động của đàn kiến trong tự nhiên mà thuật toán Ant Colony đã ra đời. Thuật toán Ant Colony mô phỏng hoạt động của đàn kiến nhân tạo như sau: Thuật toán sử dụng

một đàn kiến nhân tạo (Artificial Ants) để mô phỏng lại các hoạt động trong tự nhiên của đàn kiến. Đàn kiến nhân tạo có một số điều chỉnh về mặt hoạt động so với đàn kiến tự nhiên nhằm tăng tính hiệu quả của thuật toán. Giải thuật này chủ yếu mô tả hoạt động tìm đường đi ngắn nhất của các con kiến nhân tạo dựa vào lượng mùi để lại trong quá trình di chuyển của đàn kiến. Cụ thể về hoạt động của đàn kiến nhân tạo được trình bày như sau: Cho một đồ thị đầy đủ gồm các đỉnh và cạnh trong đó đỉnh là nơi kiến sẽ đi qua và các cạnh là các đoạn đường. Nút ban đầu cho đường đi của mỗi con kiến được chọn một cách ngẫu nhiên và thông tin mùi (pheromone) được tính toán và đặt trên mỗi đoạn đường. Mỗi con kiến sẽ chọn đường đi theo các nguyên tắc:

+ Dựa vào thông tin mùi để lại có trên các đoạn đường để tính xác suất chọn đoạn đường tiếp theo và làm đường đi cho con kiến.

+ Đường đi có lượng mùi (pheromone) nhiều hơn thì sẽ được gán một xác suất lớn hơn. Ngược lại các đoạn đường đi có lượng thông tin mùi thấp hơn sẽ có xác suất được chọn thấp hơn. Con kiến sẽ tìm đường đi cho tới khi hoàn thành một đường đi của nó (thỏa mãn điều kiện dừng của con kiến).

+ Một đường đi đầy đủ và hoàn chỉnh được gọi là một lời giải đúng cho bài toán đặt ra. Các đường đi tìm được sẽ được phân tích, so sánh và đánh giá để tìm phương án tối ưu nhất có thể. Đó là lời giải tối ưu của bài toán.

+ Sau khi tất cả kiến trong đàn hoàn thành việc tìm đường đi của nó, ta sẽ tiến hành cập nhật thông tin mùi cho các cung. Số lượng của mùi sẽ được tính toán và điều chỉnh để tìm được phương án tối ưu tốt hơn:

- Các lời giải có đường đi ngắn hơn sẽ có khối lượng pheromone lớn hơn để đặt trên các cung đã được đi qua. Ngược lại, các lời giải cho đường đi dài hơn sẽ có khối lượng pheromone bé hơn.

- Con kiến sẽ chọn đường đi có lượng mùi lớn hơn và xác suất chọn đường đi đó cũng cao hơn.

Quá trình này được lặp đi lặp lại cho đến khi hầu hết các con kiến trong đàn kiến nhân tạo chọn cùng một đường đi, và đây chính là lời giải của bài toán.

THUẬT TOÁN ANT COLONY

Input: Số kiến, khởi tạo số thành phố, vị trí tọa độ các thành phố, điểm xuất phát, khởi tạo mùi.

Output: Chu trình đường đi ngắn nhất từ thành phố xuất phát qua các thành phố còn lại một lần và quay về lại thành phố ban đầu

Method:

Begin:

B1: Nhập dữ liệu đầu vào: số kiến, vị trí tọa độ các thành phố, khoảng cách giữa các thành phố, điểm xuất phát, khởi tạo mùi.

B2: Xây dựng phương án lựa chọn đường đi

B3: Cập nhật mùi

B4: Nếu chu trình đường đi là ngắn nhất thì qua B5 ngược lại thì quay lại B2

End

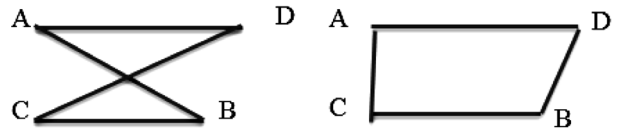
3. Giải pháp cải tiến thuật toán Ant Colony

Trong quá trình xem xét và tìm hiểu về giải thuật Ant Colony cho bài toán người bán hàng, chúng tôi đặt ra vấn đề liệu rằng có cách nào để cải tiến thuật toán Ant Colony để đường đi tìm được của các con kiến trong thuật toán ngắn hơn không?

Qua một quá trình xem xét và tìm hiểu chúng tôi nhận ra rằng trên chu trình đường đi mà thuật toán Ant Colony tìm được có những đoạn đường đi giao nhau, dựa vào đó chúng tôi có thể làm cho các đoạn đường đi này ngắn hơn bằng cách không lựa chọn các đoạn đường đi giao nhau trong chu trình đường đi tìm ra được tạo ra bởi thuật toán Ant Colony đó.

Chúng tôi đã dựa vào toán học để tìm cách

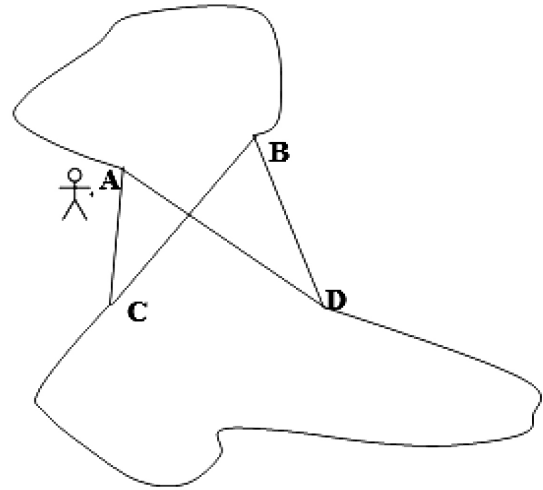
nhận ra hai đoạn đường đi giao nhau và tiến hành thực hiện loại bỏ những đoạn đường đi giao nhau đó để tạo thành một chu trình mới có độ dài ngắn hơn chu trình mà thuật toán Ant Colony đã tìm ra.



Hình 2. Chu trình có cắt nhau và chu trình cải tiến

Chẳng hạn, với một hành trình đi có giao nhau như Hình 2 là ABCD, chúng tôi tiến hành loại bỏ những đoạn đường đi giao nhau để tạo chu trình mới là ACBD có độ dài ngắn hơn.

Ý tưởng của thuật toán cải tiến được mô tả rõ hơn như sau: Giả sử người bán hàng cần di chuyển theo chu trình $A \rightarrow B \rightarrow C \rightarrow D \rightarrow A$. Như vậy theo Hình 2.1, ta thấy tổng chiều dài chu trình người đó cần di chuyển là $AB + BC + CD + DA$.

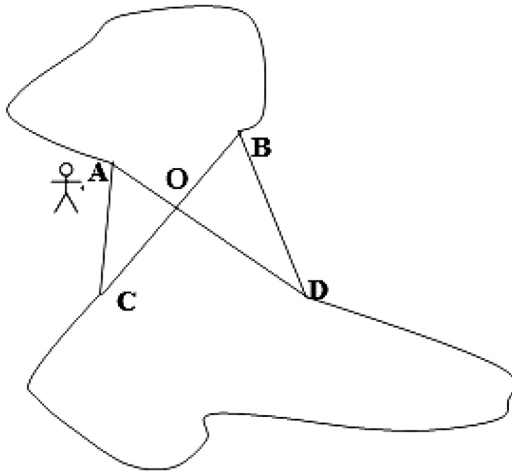


Hình 2.1

Trong chu trình di chuyển ta thấy có hai đoạn đường giao nhau là BC và DA.

Việc người bán hàng di chuyển qua hai đoạn đường này vì người bán hàng phải di chuyển theo chu trình $A \rightarrow B \rightarrow C \rightarrow D \rightarrow A$. Để làm cho chu trình di chuyển của người bán hàng ngắn hơn nhóm chúng tôi đề xuất người bán hàng di chuyển theo chu trình $A \rightarrow B \rightarrow D \rightarrow C \rightarrow A$. Việc này làm cho đường đi của chu trình di chuyển sẽ ngắn hơn vì theo toán học ta thấy đoạn đường $AC + BD < BC + DA$.

Điều này ta hoàn toàn có thể chứng minh được dựa vào Hình 2.2



$$DA=AO+OD$$

$$BC=BO+OC$$

$$\text{Nên } DA+BC=AO+OD+BO+OC$$

Trong tam giác AOC thì $AC < AO+OC$

Trong tam giác BOD thì $BD < BO+OD$

$$\text{Vì vậy ta có } AC+BD < AO+OC+BO+OD$$

$$\text{Hay } AC+BD < DA+BC$$

→ Chu trình mới ABDCA sẽ có chiều dài ngắn hơn chu trình có đường đi giao nhau ABCDA ban đầu.

3.1. Các bước xây dựng giải pháp

Bước 1: Xây dựng công thức xác suất lựa chọn đường đi cho con kiến k theo thuật toán Ant Colony: Chọn con kiến k trong số m con đang ở thành phố i, chọn ngẫu nhiên thành phố j với xác suất pij. Xác suất chọn ngẫu nhiên thành phố i được tính theo công thức sau:

$$P_{ij}^k(t) = \frac{[T_{ij}(t)]^\alpha \cdot [\eta_{ij}]^\beta}{\sum_{l \in N_i^k} [T_{il}(t)]^\alpha \cdot [\eta_{il}]^\beta} \text{ nếu } j \in N_i^k \quad [4]$$

Trong đó:

P_{ij}^k là xác suất con kiến k lựa chọn cạnh ij

T_{ij} là mùi của cạnh ij,

α là hệ số điều chỉnh ảnh hưởng của mùi T_{ij} ,

$\eta_{ij} = 1/d_{ij}$ là mùi khởi tạo ban đầu với d_{ij} là khoảng cách giữa hai thành phố i,j.

α, β chỉ ra mức độ ảnh hưởng của thông tin ban đầu và thông tin khai phá.

+ Duyệt qua các thành phố j thuộc S dựa trên xác suất pij.

Bước 2: Xây dựng thuật toán kiểm tra trong phương án có hai đường đi giao nhau hay không? Đây là phần cải tiến cơ bản của nhóm chúng tôi nhằm giúp tìm ra chu trình đường đi ngắn hơn so với thuật toán Ant Colony.

Input: Tọa độ 4 điểm A, B, C, D

Output: AB có giao với CD không?

Method:

B1: Kiểm tra C, D nằm khác phía so với đường thẳng AB

B2: Kiểm tra A, B nằm khác phía so với đường thẳng CD

B3: Nếu C, D nằm khác phía so với đường thẳng AB và A, B nằm khác phía so với đường thẳng CD thì AB và CD giao nhau thì thực hiện tìm đường đi không giao nhau thay thế → cập nhật lại chu trình.

B4: Kết thúc

Bước 3: Xây dựng công thức cập nhật mùi theo thuật toán Ant Colony

+ Bay hơi mùi các thành phố không nằm trong hành trình tốt nhất:

$$T_{ij}(t+1) = (1-\rho) \cdot T_{ij}(t) + T_{ij}(t+1) = (1-\rho) \cdot T_{ij}(t) + \sum_{k=1}^m \Delta T_{ij}^k(t) \quad [4]$$

Trong đó: ρ là tốc độ bay hơi ($0 < \rho < 1$),

$\Delta T_{ij}^k(t)$ là lượng mùi tăng thêm bởi con kiến k được tính theo công thức:

$$\Delta T_{ij}^k(t) = \begin{cases} 1/L^k(t), & \text{nếu con kiến k đi qua cạnh (i,j)} \\ 0, & \text{ngược lại} \end{cases}$$

[4]

+ Gia tăng mùi cho các thành phố nằm trong hành trình tốt nhất.

3.2. Nội dung thuật toán cải tiến Ant Colony

Input: Số kiến, khởi tạo số thành phố, vị trí tọa độ các thành phố, điểm xuất phát, khởi tạo mùi.

Output: Chu trình đường đi ngắn nhất từ thành phố xuất phát qua các thành phố còn lại một lần và quay về lại thành phố ban đầu

Method:

Begin:

B1: Nhập dữ liệu đầu vào: số kiến, vị trí tọa độ các thành phố, khoảng cách giữa các thành phố, điểm xuất phát, khởi tạo mùi.

B2: Xây dựng phương án lựa chọn đường đi dựa theo xác suất P_{ij}^k

B3: Nếu trong phương án lựa chọn có đoạn đường đi giao nhau thì loại bỏ đoạn đường đi giao nhau, cập nhật lại chu trình.

B4: Cập nhật mùi

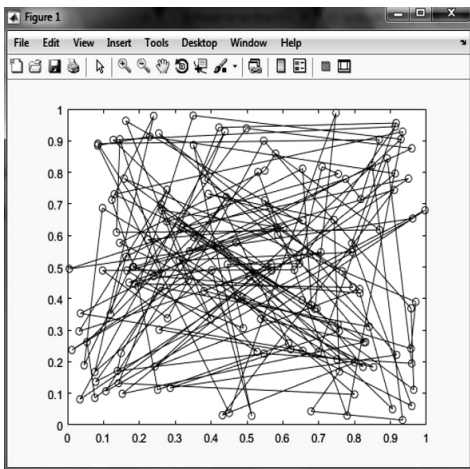
B5: Nếu chu trình đường đi là ngắn nhất thì qua B5 ngược lại thì quay lại B2

End

4. Kết quả thực nghiệm và kết luận

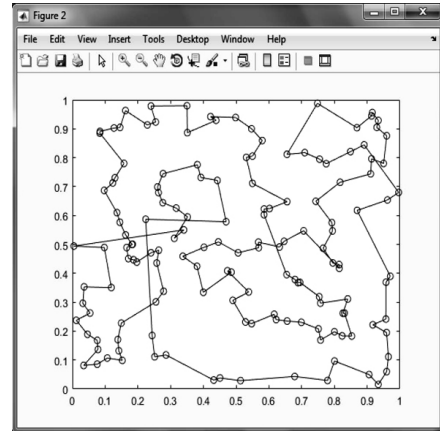
Nhóm chúng tôi đã xây dựng chương trình thực nghiệm và chạy chương trình thử nghiệm như sau:

Bước 1: Khởi tạo số thành phố ban đầu và tạo đường đi ngẫu nhiên giữa các thành phố: Các điểm đến (x,y) được tạo ngẫu nhiên trong không gian 2D, trong đó x, y thuộc $[0, 1]$



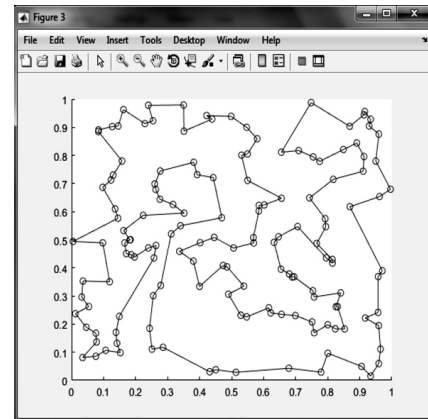
Hình 3. Khởi tạo số thành phố ban đầu và đường đi ngẫu nhiên giữa các thành phố

Bước 2: Chạy chương trình tìm đường đi ngắn nhất bằng giải thuật Ant Colony và ghi nhận chiều dài chu trình đường đi thu được



Hình 4. Chu trình đường đi do thuật toán Ant Colony tìm ra có chiều dài $L=10.6270$

Bước 3: Chạy chương trình tìm đường đi ngắn nhất bằng giải thuật cải tiến thuật toán Ant Colony và ghi nhận chiều dài chu trình đường đi thu được



Hình 5. Chu trình đường đi do thuật toán cải tiến Ant Colony có chiều dài $L= 9.6540$

Bước 4: So sánh kết quả chiều dài đường đi giữa thuật toán Ant Colony và thuật toán cải tiến Ant Colony do nhóm xây dựng.

Sau quá trình chạy thử nghiệm chương trình nhóm chúng tôi đã nhận thấy rằng: Việc cải tiến thuật toán Ant Colony bằng cách loại bỏ những đoạn đường đi giao nhau và thay thế bằng một chu trình đường đi không giao nhau thì chu trình đường đi thu được từ giải thuật cải tiến do nhóm xây dựng có độ dài ngắn hơn. Điều này đã được chúng tôi thực

nghiệm nhiều lần và cho kết quả khả quan. Bảng so sánh và đánh giá chiều dài thực hiện thuật toán cải tiến do nhóm cài đặt và thuật toán Ant Colony như sau:

Bảng 1. Bảng kết quả đánh giá giữa thuật toán Ant Colony và thuật toán cải tiến Ant Colony

| STT | Số thành phố | Chiều dài đường đi tìm được bởi thuật toán Ant Colony | Chiều dài đường đi tìm được bởi thuật toán cải tiến Ant Colony | Tỷ lệ % cải thiện |
|-----|--------------|---|--|-------------------|
| 1 | 10 | 36.054 | 35.531 | 1.45 |
| 2 | 100 | 90.599 | 87.023 | 3.95 |
| 3 | 150 | 106.601 | 101.869 | 4.44 |
| 4 | 200 | 122.123 | 115.679 | 5.28 |
| 5 | 250 | 134.874 | 127.367 | 5.58 |

Mỗi bộ dữ liệu trong bảng đều được tiến hành chạy 10 lần thuật toán Ant Colony và thuật toán cải tiến Ant Colony trên cùng một máy tính Intel(R) Core(TM) i5-4210U CPU 1.7 Hz, RAM 4.0GB. Kết quả về đoạn đường di chuyển trung bình của người bán hàng trong hai thuật toán được đưa ra trong Bảng 1.

Qua dữ liệu của Bảng 1 ta thấy việc sử dụng thuật toán cải tiến thuật toán Ant Colony đã giúp giảm đáng kể đường đi cho người bán hàng. Giả sử đơn vị tính của đường đi giữa các thành phố là km thì theo bảng số liệu ta có: Trường hợp người bán hàng di chuyển qua 10 thành phố thì số km giảm được là $36.054 - 35.531 = 523$ km. Nếu tăng số thành phố lên 100 thì đoạn đường giảm được là $90.599 - 87.023 = 3.576$ km. Trường hợp người bán hàng di chuyển qua 150 thành phố thì số km giảm được là $106.601 - 101.869 = 4.732$ km. Trường hợp số thành phố tăng lên 200 thì số km giảm được khi người bán hàng di chuyển là $122.123 - 115.679 = 6.444$ km. Số km càng được giảm nhiều hơn khi số thành phố là 250, cụ thể: $134.874 - 127.367 = 7.507$ km.

Từ việc phân tích bảng số liệu kết quả ở Bảng 1 ta thấy số thành phố càng tăng thì tỷ lệ cải thiện đường đi của thuật toán cải tiến Ant Colony do nhóm chúng tôi đề xuất càng tăng, đồng thời qua đó cho thấy thuật toán Ant Colony vẫn còn nhược điểm là đưa ra phương án đường đi có lộ trình còn dài cho người bán hàng. Đặc biệt khi số thành phố trong lộ trình càng tăng thì nhược điểm về chi phí đường đi của thuật toán Ant Colony càng thể hiện

rõ, trong khi đó nếu số thành phố trong lộ trình càng tăng cao thì thuật toán cải tiến Ant Colony càng thể hiện tính ưu việt vượt trội của nó và được thể hiện rõ trong cột tỷ lệ % cải thiện của Bảng 1.

5. Kết luận

Bài báo đã đề xuất được giải thuật cải tiến thuật toán Ant Colony để tìm được đường đi ngắn hơn cho bài toán người bán hàng, xây dựng chương trình hoàn chỉnh áp dụng cho giải thuật đó. Thuật toán cải tiến thuật toán Ant Colony được mô tả ở mục 3.2. Kết quả chạy chương trình do nhóm xây dựng trên nhiều bài toán thiết kế cho thấy tính đúng đắn của giải pháp đề xuất và tính hiệu quả về mặt chi phí đường đi so với thuật toán Ant Colony.

Tài liệu tham khảo

- [1] Marco Dorigo, Thomas Stützle, "Incremental Local Search in Ant Colony Optimization".
- [2] Phạm Hồng Luân, Dương Thành Nhân, "Nghiên cứu ứng dụng thuật toán aco (ant colony optimization) tối ưu thời gian và chi phí cho dự án xây dựng", Tạp chí Phát triển Khoa học và Công nghệ, tập 13, số q1–2010.
- [3] Đỗ Đức Đông và Hoàng Xuân Huân, "Multi-level Ant System: A new approach through the new pheromone update of Ant Colony Optimization", Kỷ yếu Hội nghị quốc tế Khoa học máy tính RIVF lần thứ 4, Tp.Hồ Chí Minh, Tháng 2/2006.
- [4] Anirudh Shekhawat, Pratik Poddar, Dinesh Boswa, "Ant colony Optimization Algorithms: Introduction and Beyond", Indian Institute of Technology Bombay, Artificial Intelligence Seminar 2009.
- [5] Marco Dorigo, Luca Maria Gambardella, "Ant Colony System: A Cooperative Learning Approach to the Traveling Salesman Problem", Accepted for publication in the IEEE Transactions on Evolutionary Computation, Vol.1, No.1, 1997.