

TỐI ƯU TRIỂN KHAI MẠNG CẢM BIẾN KHÔNG DÂY THỎA MÃN RÀNG BUỘC ĐA BAO PHỦ VÀ ĐA KẾT NỐI

Nguyễn Xuân Thắng*, Trịnh Thị Thu Huyền*, Nguyễn Thị Hạnh*

*Khoa Công nghệ thông tin, Trường Đại học Hà Nội

*Khoa Công nghệ số liên ngành, Trường Đại học Phenikaa

Tóm tắt: Bài báo đề xuất giải thuật Heuristic mới để giải quyết đồng thời hai bài toán quan trọng trong triển khai mạng cảm biến không dây: tối thiểu hóa số lượng nút mạng cần triển khai đồng thời thỏa mãn các ràng buộc về K- bao phủ và K- kết nối. Để đánh giá hiệu quả của giải thuật đề xuất, nhóm tác giả so sánh kết quả thực nghiệm với hai giải thuật đã được công bố trong các nghiên cứu gần đây. Các thực nghiệm được tiến trên 42 bộ dữ liệu thuộc hai tập dữ liệu và ba kịch bản thử nghiệm. Kết quả cho thấy giải thuật đề xuất có hiệu quả tốt hơn so giải thuật GA và KGCA đã được công bố tại [5, 6] trong đa số bộ dữ liệu, kể cả về số lượng nút cảm biến và số lượng nút chuyển tiếp cần thiết lập.

Từ khóa: Mạng cảm biến không dây, tối ưu triển khai mạng, K- bao phủ, K- kết nối

I. GIỚI THIỆU

Mạng cảm biến không dây (Wireless Sensor Network - WSN) có kiến trúc gồm hai thành phần chính là một tập các nút cảm biến và một trạm cơ sở. Sau khi được triển khai tại địa điểm cụ thể, các nút mạng liên tục thu thập thông tin trong phạm vi bán kính cảm biến, xử lý và trao đổi thông tin với các nút mạng khác hoặc gửi về trạm cơ sở thông qua kết nối không dây. Một trong số những ứng dụng tiêu biểu của WSN là triển khai mạng để theo dõi, giám sát đối tượng. Trong đó, một tập các đối tượng đã biết trước vị trí cần được giám sát (bao phủ) bởi các nút cảm biến. Thông tin về đối tượng được chuyển tiếp qua các nút mạng để gửi về trạm cơ sở [1].

Đối với ứng dụng giám sát đối tượng, bên cạnh vấn đề sử dụng năng lượng hiệu quả, bài toán triển khai WSN, hay tìm vị trí để đặt các nút cảm biến cũng là vấn đề cần quan tâm. Việc đặt những nút cảm biến này một cách ngẫu nhiên có thể không thực hiện được ý đồ hay nhiệm vụ giám sát của mạng và gây lãng phí vì phải dùng một số lượng lớn các nút cảm biến. Trên thực tế, do các nút mạng chỉ hoạt động tốt trong một vùng bán kính nhất định, việc triển khai các nút cảm biến để đảm bảo các yêu cầu về bao phủ và kết nối trong toàn mạng đã trở thành yêu cầu cấp thiết [2].

A. Yêu cầu về bao phủ đối tượng

Yêu cầu về bao phủ đối tượng trong WSN được đặt ra là phải xác định số lượng nút cảm biến và vị trí đặt chúng để đảm bảo mọi đối tượng đều được giám sát. Ở mức độ đơn giản nhất, yêu cầu mỗi đối tượng cần được bao phủ bởi ít nhất một nút cảm biến (gọi là yêu cầu 1- bao phủ). Tuy nhiên để cải thiện chất lượng dịch vụ của mạng yêu cầu bao phủ có thể mở rộng trong đó mỗi đối tượng cần được bao phủ bởi ít nhất K nút cảm biến (gọi là yêu cầu K- bao phủ, $K > 1$). Trong một số ứng dụng cụ thể, các đối tượng cần giám sát có mức độ ưu tiên khác nhau, khi đó yêu cầu bao phủ được mở rộng thành Q- bao phủ: đối tượng i cần được giám sát bởi ít nhất Q_i nút cảm biến [3].

B. Yêu cầu về kết nối

Kết nối không dây giữa các nút mạng của WSN thường không ổn định, đặc biệt là khi mạng được triển khai trong những môi trường phức tạp. Yêu cầu về kết nối trong WSN được đặt ra là phải đảm bảo đường truyền dữ liệu giám sát đối tượng về trạm cơ sở. Tương tự như yêu cầu về bao phủ, yêu cầu về kết nối cũng được chia thành 1- kết nối (mỗi đối tượng cần giám sát có ít nhất một đường truyền dữ liệu về trạm cơ sở); K- kết nối (mỗi đối tượng cần giám sát có ít nhất K đường truyền dữ liệu phân biệt về trạm cơ sở); và Q- kết nối (đối tượng i cần có ít nhất Q_i kết nối phân biệt về trạm cơ sở). Trong đó, hai kết nối (hay đường truyền dữ liệu) được coi là phân biệt khi hai tập các nút trung gian của hai kết nối không có nút nào trùng nhau [13].

Bài toán tối thiểu hóa số lượng nút cảm biến khi triển khai WSN thỏa mãn các ràng buộc về bao phủ và kết nối nói trên là các bài toán NP-khó và đã được một số nhà nghiên cứu quan tâm nhưng các nghiên cứu hiện tại chưa đề cập đến việc đồng thời giải quyết vấn đề đa bao phủ và đa kết nối [4]. Để giải quyết các bài toán NP-khó đặt ra trong bài báo này, có hai nhóm phương pháp chính là giải đúng và giải gần đúng. Các phương pháp giải đúng như quay lui, quy hoạch động, nói lỏng ràng buộc Lagrange... đảm bảo tìm ra được lời giải chính xác và chứng minh được sự tối ưu của lời giải với các bộ dữ liệu có kích thước nhất định (thường là nhỏ), với thời gian chạy giới hạn hoặc là chứng minh không tồn tại lời giải. Các phương pháp giải gần đúng Heuristic gồm mô phỏng luyện kim, tìm kiếm tabu, giải thuật di truyền ... không đảm bảo tìm được lời giải chính xác mà thường chỉ tìm được lời giải gần đúng (xấp xỉ). Tuy nhiên, thời gian chạy của các thuật toán này thường chấp nhận được với mọi bộ dữ liệu [4].

Bài báo đề xuất giải thuật Heuristic để giải bài toán tối thiểu hóa số nút cảm biến khi triển khai WSN thỏa mãn đồng thời hai ràng buộc về K- bao phủ và K- kết nối (gọi

Tác giả liên hệ: Nguyễn Xuân Thắng

Email: nxthang@hau.edu.vn

Đến tòa soạn: 23/12/2024, chỉnh sửa: 19/03/2025, chấp nhận đăng: 26/03/2025.

tất là bài toán K- bao phủ và K- kết nối). Tác giả đề xuất phương pháp giải quyết bài toán gồm hai pha. Pha một thực hiện tìm lời giải cho bài toán K- bao phủ, đầu ra của pha một sẽ trở thành đầu vào của pha hai. Pha hai thực hiện tìm lời giải cho bài toán K- kết nối. Trong cả hai pha, nhóm tác giả áp dụng các giải thuật Heuristic để tìm các kết quả tối ưu hoặc gần tối ưu.

Giải thuật đề xuất có điểm cải tiến so với hai giải thuật đã được công bố gần đây là giải thuật GA [5] và giải thuật KGCA [6]. Cụ thể, kết quả của giải thuật GA phụ thuộc khá nhiều vào bước xây dựng tập các vị trí tiềm năng. Khi miền cần giám sát có kích thước rộng, việc chọn ngẫu nhiên sẽ dẫn đến tập vị trí tiềm năng không đủ tốt từ đó ảnh hưởng xấu đến kết quả của giải thuật GA. Khi giải quyết vấn đề K-bao phủ, giải thuật KGCA có heuristic tương tự như giải thuật đề xuất trong bài báo. Điểm khác biệt là KGCA chỉ tìm cách đặt các nút cảm biến trên vùng giao nhau của hai đĩa tròn bao quanh mục tiêu. Còn giải thuật đề xuất trong bài báo tìm cách đặt các nút cảm biến trên vùng giao nhau của nhiều đĩa tròn nhất. Thực tế có những trường hợp số lượng đĩa tròn giao nhau nhiều hơn hai, khi đó kết quả của KGCA sẽ không tốt bằng kết quả của giải thuật đề xuất trong bài báo. Đối với bài toán K-kết nối, KGCA thực hiện đặt các nút chuyển tiếp trên đường đi ngắn nhất từ nút cảm biến đến trạm cơ sở (xác định bằng thuật toán Dijkstra), tuy nhiên cách tiếp cận này không phải lúc nào cũng mang lại kết quả tối ưu về số lượng nút chuyển tiếp. Thuật toán đề xuất trong bài báo thực hiện đặt các nút chuyển tiếp trên các cạnh của cây khung nhỏ nhất của đồ thị có các đỉnh là trạm cơ sở và nút đại diện cho cụm K- cảm biến gần nhau, do đó giúp giảm số lượng nút chuyển tiếp cần triển khai.

Để đánh giá hiệu quả của giải thuật đề xuất trong bài báo, nhóm tác giả tiến hành thực nghiệm và so sánh kết quả thực nghiệm với giải thuật GA [5] và giải thuật KGCA [6]. Tác giả xây dựng hai tập dữ liệu thực nghiệm, với ba kịch bản thực nghiệm khác nhau được áp dụng trên mỗi tập dữ liệu. Trong mỗi kịch bản thực nghiệm, nhóm tác giả thay đổi các tham số để tạo ra bảy bộ dữ liệu tương ứng. Các kết quả thực nghiệm cho thấy giải thuật đề xuất có hiệu quả tốt hơn so với hai giải thuật còn lại trong đa số bộ dữ liệu thử nghiệm, kể cả về số lượng nút cảm biến và số lượng nút chuyển tiếp cần thiết lập để thỏa mãn các ràng buộc K- bao phủ và K- kết nối.

Phần còn lại của bài báo được tổ chức như sau. Phần II tóm tắt các nghiên cứu liên quan. Phần III trình bày về phát biểu bài toán K- bao phủ và K- kết nối và giải thuật đề xuất để giải quyết bài toán. Phần IV mô tả kết quả thực nghiệm và so sánh giải thuật đề xuất trong bài báo với hai giải thuật đã được công bố trong các nghiên cứu gần đây. Kết luận và hướng phát triển tiếp theo được trình bày trong Phần V.

II. CÁC NGHIÊN CỨU LIÊN QUAN

Vấn đề đa bao phủ đối tượng và đa kết nối trong WSN là một chủ đề được nhiều nhà nghiên cứu quan tâm giải quyết. Với các mục tiêu tối ưu khác nhau, một số nhóm tác giả tập trung giải quyết bài toán K- bao phủ, trong khi một số công trình nghiên cứu khác tìm cách giải quyết bài toán K- kết nối. Tác giả của [7, 8] sử dụng thuật toán di truyền để giải quyết bài toán K- bao phủ với mục tiêu kéo dài thời gian tồn tại của mạng. Nghiên cứu [9] đề xuất một kỹ thuật heuristic để giải quyết bài toán Q- bao phủ với mục tiêu cực đại bao phủ và kéo dài thời gian sống của mạng thỏa

mãn ràng buộc mỗi đối tượng i_{th} phải được bao phủ bởi ít nhất Q_{imin} cảm biến. Tác giả của [10] đề xuất sử dụng thuật toán tối ưu bầy đàn (Particle Swarm Optimization - PSO) và PSO kết hợp logic mờ để giải quyết bài toán triển khai cảm biến trong WSN thỏa mãn các ràng buộc 1- bao phủ, K- bao phủ và Q- bao phủ.

Sheikhi và các cộng sự [11] đề xuất phương pháp tên là KCN để giải quyết bài toán K- kết nối. KCN bao gồm ba thuật toán: Thuật toán 1 tạo đồ thị kết nối tối thiểu K đỉnh, Thuật toán 2 xác định vị trí các nút chuyển tiếp để triển khai trên các cạnh của đồ thị và Thuật toán 3 giảm thiểu số lượng nút chuyển tiếp được triển khai. Kết quả mô phỏng cho thấy KCN có độ phức tạp thời gian thấp ($O(n^2)$) và có số lượng nút chuyển tiếp cần thiết ít hơn những phương pháp khác được so sánh trong công bố. Nghiên cứu [12] đề xuất thuật toán LINAR duy trì K-kết nối trong các mạng cảm biến di động (là mạng WSN có nút cảm biến di chuyển tự do và chúng có thể kết nối và giao tiếp với nhau mà không cần một cấu trúc liên kết cố định). Khi một nút cảm biến bị lỗi, thuật toán sẽ di chuyển nút lân cận tới để thay thế nhằm đảm bảo thỏa mãn ràng buộc K- kết nối.

Để giải quyết đồng thời vấn đề đa bao phủ, đa kết nối, đã có một số nghiên cứu bước đầu được triển khai. Gupta và cộng sự đề xuất phương pháp giải quyết bài toán K- bao phủ và K- kết nối dựa trên giải thuật di truyền [5], với các bước thực hiện chính như sau: i) Xác định một tập các vị trí tiềm năng để đặt các nút cảm biến và nút chuyển tiếp; ii) Sử dụng giải thuật di truyền để tìm ra tập vị trí đặt nút cảm biến và nút chuyển tiếp tối ưu. Hàm thích nghi được thiết kế giúp tối ưu hóa ba mục tiêu: số lượng nút mạng, số lượng bao phủ và số lượng kết nối. Trong nghiên cứu [6], nhóm tác giả đề xuất giải thuật Heuristic tên là KGCA để giải quyết bài toán K- bao phủ và K- kết nối. Ý tưởng của giải thuật là tìm ra các cặp mục tiêu có vị trí gần nhau để có thể đặt K nút cảm biến đồng thời bao phủ cặp mục tiêu này. Sau khi tìm được vị trí các nút cảm biến thỏa mãn điều kiện K- bao phủ, thuật toán KGCA thực hiện đặt các nút chuyển tiếp dọc theo đường đi ngắn nhất từ mỗi nút cảm biến đến trạm cơ sở. Đường đi ngắn nhất này được xác định bằng thuật toán Dijkstra có chỉnh sửa để phù hợp với yêu cầu của bài toán.

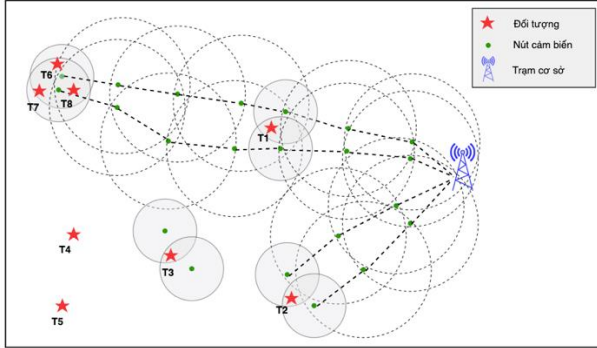
III. PHÁT BIỂU BÀI TOÁN VÀ GIẢI THUẬT

A. Phát biểu bài toán

Để biểu diễn khả năng bao phủ của một nút cảm biến, nhóm tác giả sử dụng mô hình đĩa nhị phân trong đó vùng bao phủ của nút cảm biến là một hình tròn có tâm s là vị trí của nút cảm biến, bán kính R_s là bán kính cảm nhận của cảm biến. Với đối tượng z trong không gian hai chiều, hàm cảm biến $f(z)$ có giá trị 1 nếu khoảng cách $d(s,z) \leq R_s$, nếu ngược lại $f(z)$ có giá trị 0. Khi $f(z)=1$ đối tượng z được giám sát (hoặc bao phủ) bởi nút cảm biến s .

Mô hình đĩa nhị phân cũng được nhóm tác giả sử dụng để biểu diễn khả năng kết nối của hai nút cảm biến. Trong mô hình này, mỗi nút cảm biến sẽ có bán kính giao tiếp không dây R_c . Bán kính này có thể có giá trị cố định hoặc thay đổi phụ thuộc vào sự thay đổi công suất thu, phát tín hiệu của nút cảm biến. Hai nút cảm biến i và j được coi là có kết nối với nhau nếu khoảng cách d_{ij} giữa hai nút không vượt quá giá trị $\min\{R_{ci}, R_{cj}\}$. Hàm kết nối $f(i,j)$ có giá trị 1 nếu khoảng cách $d(i,j) \leq \min\{R_{ci}, R_{cj}\}$, nếu ngược lại $f(i,j)$ có giá trị 0.

Bài toán K- bao phủ và K- kết nối trong WSN được phát biểu như sau: "Trên một miền phẳng 2D cho trước vị trí cố định đặt các đối tượng và trạm cơ sở, bài toán đặt ra là tối thiểu hóa số lượng các nút cảm biến và các nút chuyển tiếp đặt trên miền phẳng sao cho thỏa mãn điều kiện mỗi đối tượng có ít nhất K bao phủ và mỗi đối tượng có ít nhất K kết nối phân biệt về trạm cơ sở. Hai kết nối được coi là phân biệt khi hai tập các nút trung gian của hai kết nối không có nút nào trùng nhau."



Hình 1. Ví dụ về bài toán K- bao phủ và K- kết nối

Hình 1 mô tả một ví dụ của bài toán K- bao phủ và K- kết nối trong WSN, các đối tượng T_1, T_2, T_6, T_7 và T_8 thỏa mãn đồng thời hai tiêu chí K- bao phủ và K- kết nối, với giá trị K bằng 2. Ngược lại, đối tượng T_3 thỏa mãn điều kiện K- bao phủ nhưng không đáp ứng yêu cầu về K- kết nối. Ngoài ra, các đối tượng T_4 và T_5 không thỏa mãn cả hai tiêu chí K- bao phủ và K- kết nối.

Bảng I trình bày các ký hiệu được sử dụng để mô tả bài toán:

Bảng I. Bảng các ký hiệu dùng trong bài toán

Ký hiệu	Định nghĩa
$A(W \times H)$	Miền không gian 2D (dài x rộng) cần giám sát
T, n	Tập vị trí của n đối tượng cần giám sát
B, K	Vị trí trạm cơ sở, số lượng bao phủ và kết nối
S, m	Tập vị trí của m nút cảm biến tìm được
R_s, R_c	Bán kính cảm nhận và Bán kính giao tiếp của các nút cảm biến. Giả thiết các nút cảm biến có cùng giá trị R_s và R_c .
R, p	Tập vị trí của p nút chuyển tiếp tìm được
$dist(n_i, n_j)$	Khoảng cách Euclid giữa hai vị trí n_i và n_j , trong đó các vị trí có thể của nút cảm biến, nút chuyển tiếp hoặc đối tượng
$b_{i,j}$	Đại lượng thể hiện đối tượng t_i có được bao phủ bởi nút cảm biến s_j hay không.
c_{ij}	Đại lượng thể hiện nút n_i có kết nối được với nút n_j hay không.
sc_i	Đại lượng thể hiện có ít nhất K đường kết nối phân biệt từ đối tượng t_i về trạm cơ sở B. Nếu $sc_i = 1$ thì tổng số kết nối phân biệt từ t_i đến trạm cơ sở là $\geq K$. Nếu điều kiện này không thỏa mãn thì $sc_i = 0$.

Sử dụng các ký hiệu của Bảng 1, ta có mô tả của bài toán như sau:

Với tập

$$T = t_i(x_{ti}, y_{ti}) \mid 0 \leq x_{ti} \leq W, 0 \leq y_{ti} \leq H, i = 1 \dots n$$

biểu thị tập vị trí cho trước của các đối tượng, và vị trí $B(x_b, y_b)$ của trạm cơ sở trên miền $A=W \times H$.

Gọi tập

$$S = s_i(x_{si}, y_{si}) \mid 0 \leq x_{si} \leq W, 0 \leq y_{si} \leq H, i = 1 \dots m$$

là tập vị trí các nút cảm biến cần tìm, tập

$$R = r_i(x_{ri}, y_{ri}) \mid 0 \leq x_{ri} \leq W, 0 \leq y_{ri} \leq H, i = 1 \dots p$$

là tập vị trí các nút chuyển tiếp cần tìm trên miền A.

Mục tiêu của bài toán:

$$\begin{cases} \text{tối thiểu hóa } m = |S| & (1) \\ \text{tối thiểu hóa } p = |R| & (2) \end{cases}$$

Các ràng buộc của bài toán:

$$\sum_{j=1}^m b_{i,j} \geq K, \forall i = 1 \dots n. \quad (3)$$

$$\sum_{i=1}^n sc_i \geq n \quad (4)$$

Bài K- bao phủ và K- kết nối nói trên là các bài toán NP-khó, vì vậy nhóm tác giả đề xuất phương pháp giải quyết bài toán gồm hai pha. Pha một thực hiện tìm lời giải cho bài toán K- bao phủ, đầu ra của pha một sẽ trở thành đầu vào của pha hai. Pha hai thực hiện tìm lời giải cho bài toán K- kết nối. Trong cả hai pha, nhóm tác giả áp dụng các giải thuật Heuristic để tìm các kết quả tối ưu hoặc gần tối ưu.

B. Giải thuật cho bài toán K- bao phủ

1) Ý tưởng

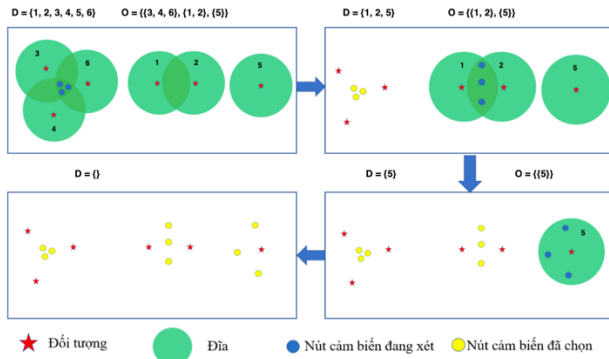
Xem xét các đĩa tròn có tâm là tọa độ của một đối tượng $t_i \in T$, bán kính là R_s . Gọi $D = \{d_1, d_2, \dots, d_n\}$ là tập gồm các đĩa tròn này, nhóm tác giả nhận thấy nếu đặt một nút cảm biến tại vùng giao nhau của 2 đĩa tròn d_i và d_j thì nút cảm biến này sẽ bao phủ đồng thời hai đối tượng t_i và t_j . Mở rộng nhận xét này nhóm tác giả đề xuất thuật toán đặt các nút cảm biến theo giải thuật tham lam gồm các bước như sau:

1. Xác định danh sách $O = \{o_1, o_2, \dots, o_m\}$, mỗi phần tử o_i là một tập con của D chứa các đĩa tròn giao với nhau. Trong trường hợp tập D có đĩa tròn d_j không giao với bất cứ đĩa tròn nào khác, thì tập con o_j tương ứng chỉ chứa một phần tử là d_j .

2. Tìm phần tử $o_{\max} \in O$ là tập con có chứa nhiều đĩa tròn giao nhau nhất. Đặt K nút cảm biến vào các vị trí ngẫu nhiên nằm trong vùng giao nhau của các đĩa tròn trong $o_{\max}$. Nếu $o_{\max}$ chỉ có một phần tử, tức là trong tập D không có đĩa tròn nào giao với nhau, thì đặt K nút cảm biến vào các vị trí ngẫu nhiên nằm trong đĩa tròn là phần tử của $o_{\max}$.

3. Các đĩa tròn thuộc $o_{\max}$ là của các đối tượng t đã thỏa mãn ràng buộc K- bao phủ. Loại bỏ các đĩa này khỏi tập D và lặp lại bước 1 cho tới khi tập D rỗng.

Hình 2 mô tả một ví dụ đặt các nút cảm biến trên vùng A có chứa $n = 6$ đối tượng thỏa mãn ràng buộc 3- bao phủ ($K = 3$ - mỗi đối tượng được bao phủ bởi ít nhất 3 nút cảm biến):



Hình 2. Ví dụ mô tả hoạt động của thuật toán giải quyết bài toán K- bao phủ

2) Thuật toán

Mã giả của thuật toán được trình bày trong Thuật toán 1.

1. Các hàm sử dụng trong thuật toán bao gồm:

- *intersectionList(D)*: trả về danh sách O gồm các phần tử là các đĩa tròn giao nhau.
- *findMax(O)*: trả về phần tử o_{max} của O có chứa nhiều đĩa tròn giao nhau nhất.
- *setSensor(o_{max})*: đặt K nút cảm biến ở các vị trí ngẫu nhiên trong vùng giao nhau của các đĩa thuộc o_{max} .
- *updateST(S_k, o_{max})*: cập nhật danh sách ST, là danh sách thể hiện đối tượng t_i được bao phủ bởi K nút cảm biến nào trong tập S.
- *removeDisk(D, o_{max})*: loại bỏ các đĩa thuộc o_{max} ra khỏi tập D.

Thuật toán 1: Giải quyết bài toán K- bao phủ dựa trên giải thuật tham lam

Input: T: Tập vị trí các đối tượng
Rs: Bán kính cảm biến

Output: S: Tập vị trí các nút cảm biến cần đặt
ST: Danh sách thể hiện đối tượng t_i được bao phủ bởi các nút cảm biến nào

Khởi tạo tập các đĩa tròn: $D \leftarrow \{d_i(t_i, R_s) | t_i \in T\}$

Khởi tạo $S \leftarrow \emptyset, ST \leftarrow \emptyset$

while $D \neq \emptyset$ **do**

$O = \text{intersectionList}(D)$

$o_{max} = \text{findMax}(O)$

$S_k = \text{setSensor}(o_{max})$

$S \leftarrow S + S_k$

$ST \leftarrow \text{updateST}(S_k, o_{max})$

$\text{removeDisk}(D, o_{max})$

end while

return S, ST

Các hàm *findMax()*, *setSensor()*, *updateST()*, *removeDisk()* tương đối đơn giản nên nhóm tác giả không trình bày chi tiết. Ý tưởng để triển khai hàm *intersectionList()* như sau:

1. Thiết lập O là cấu trúc dữ liệu kiểu tập hợp (set), là danh sách chỉ chứa các phần tử không trùng nhau. Ví dụ phần tử $o_p = \{1, 2, 3\}$ và phần tử $o_q = \{2, 3, 1\}$ được coi là hai

phần tử trùng nhau và chỉ một trong hai phần tử này được thêm vào O trong quá trình tìm kiếm.

2. Xác định các đĩa d_i mà không giao với bất cứ đĩa nào trong D, thêm phần tử $\{d_i\}$ vào tập O. Loại bỏ d_i khỏi D. Hai đĩa tròn được coi là giao nhau nếu hai đường bao cắt nhau tại hai điểm. Với cách hiểu như vậy, hai đĩa tròn tiếp xúc với nhau tại một điểm không được coi là giao nhau.

3. Với mỗi cặp đĩa tròn $(d_i, d_j) \in D$, xác định hai giao điểm P_1 và P_2 của cặp đĩa. Khởi tạo hai danh sách $O_1 = \{i, j\}$ và $O_2 = \{i, j\}$.

4. Với mỗi đĩa tròn $d_k \in D \mid k \neq i, j$, nếu P_1 nằm trong d_k nghĩa là d_i, d_j và d_k giao với nhau. Thêm k vào danh sách O_1 . Tương tự nếu P_2 nằm trong d_k thì thêm k vào danh sách O_2 .

5. Thêm O_1 và O_2 vào danh sách O. Sau khi duyệt hết các cặp đĩa tròn giao nhau (d_i, d_j) của tập D, danh sách O sẽ là danh sách cần tìm.

Mã giả của hàm *intersectionList()* được trình bày trong Thuật toán 2.

Thuật toán 2: *intersectionList(D)* - tìm danh sách O có mỗi phần tử là một tập các đĩa tròn giao nhau

Input: D: Tập các đĩa tròn

Output: Danh sách $O = \{o_1, o_2, \dots, o_m\}$, mỗi phần tử o_i là một tập con của D chứa các đĩa tròn giao với nhau.

Khởi tạo $O \leftarrow \text{set}(\emptyset)$

for $i \leftarrow 0$ **to** $|D|$ **do**

$\text{hasIntersection} = \text{False}$

for $j \leftarrow 0$ **to** $|D|$ **do**

if $i \neq j$ **and** $|\text{intersectionPoints}(D[i], D[j])| = 2$ **then**

$\text{hasIntersection} = \text{True}$

$P_1, P_2 = \text{intersectionPoints}(D[i], D[j])$

$O_1 = [i, j]; O_2 = [i, j]$

for $k \leftarrow 0$ **to** $|D|$ **do**

if $i \neq k$ **and** $j \neq k$ **and** P_1 in $D[k]$ **then**

$O_1 \leftarrow [O_1, k]$

end if

if $i \neq k$ **and** $j \neq k$ **and** P_2 in $D[k]$ **then**

$O_2 \leftarrow [O_2, k]$

end if

$O \leftarrow O.\text{add}(O_1, O_2)$

end for

end if

end for

if $\text{hasIntersection} = \text{False}$ **then**

$O \leftarrow O.\text{add}([i])$

end if

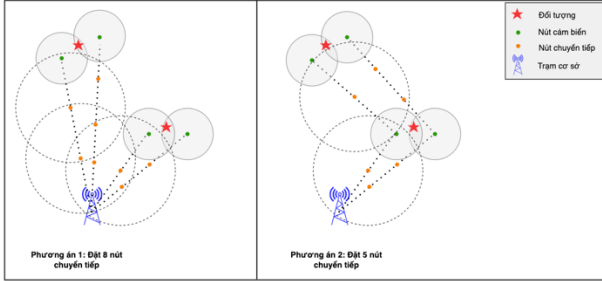
end for

return O

C. Giải thuật cho bài toán K- kết nối

1) Ý tưởng

Kết quả của Pha 1 là một phương án bố trí các nút cảm biến trên miền phẳng A thỏa mãn ràng buộc K- bao phủ. Để thỏa mãn ràng buộc K- kết nối, cách đơn giản nhất là đặt các nút chuyển tiếp trên các đoạn thẳng nối giữa các nút cảm biến bao phủ đối tượng với trạm cơ sở. Bằng cách này, mỗi đối tượng sẽ có K đường đi phân biệt về trạm cơ sở. Tuy nhiên, với cách thực hiện này không đảm bảo số lượng nút cảm biến cần dùng là tối thiểu. Xét một ví dụ được mô tả trong Hình 3:



Hình 3. Hai phương án đặt nút chuyển tiếp thỏa mãn ràng buộc K- kết nối ($K=2$)

Trong phương án 1 các nút chuyển tiếp được đặt trên các đoạn thẳng nối giữa nút cảm biến tới trạm cơ sở, phương án này cần 8 nút chuyển tiếp. Ở phương án 2, các nút cảm biến chuyển tiếp được đặt theo cách cho phép tận dụng các đường truyền đã thiết lập trước, phương án này cần 6 nút chuyển tiếp.

Xem xét phương án đặt nút cảm biến là kết quả của pha 1, nhóm tác giả nhận thấy các nút cảm biến có xu hướng tập trung thành các cụm K- cảm biến, do đó bài toán tối ưu hóa số lượng nút cảm biến chuyển tiếp trên toàn miền A có thể đơn giản hóa thành bài toán tối thiểu hóa số lượng nút chuyển tiếp giữa các cụm K- cảm biến.

Do các nút mạng có cùng bán kính giao tiếp nên số lượng nút chuyển tiếp cần đặt giữa hai nút cảm biến sẽ tỉ lệ thuận với khoảng cách Euclid giữa hai nút đó. Do đó để tối thiểu hóa số nút cảm biến chuyển tiếp cần xác định các cặp đoạn thẳng nối hai cụm K- cảm biến với nhau và với trạm cơ sở sao cho tổng chiều dài của các đoạn thẳng là nhỏ nhất, sau đó đặt các nút chuyển tiếp lên các cặp đoạn thẳng này.

Xuất phát từ hai nhận xét trên, nhóm tác giả đề xuất thuật toán đặt các nút chuyển tiếp trên miền A thỏa mãn ràng buộc K- kết nối gồm các bước như sau:

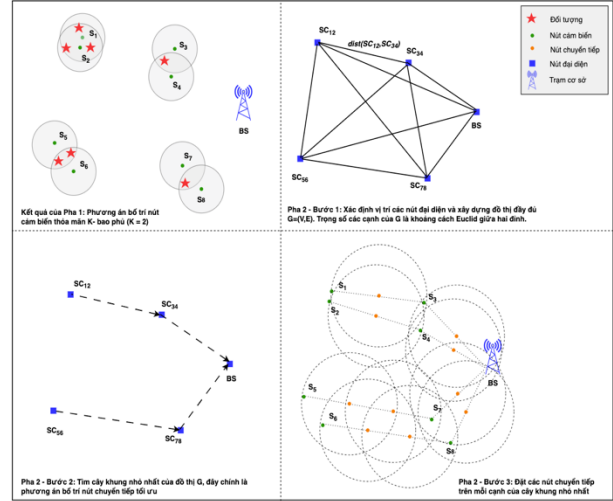
1. Xuất phát từ kết quả thu được tại pha 1, xác định vị trí các nút đại diện. Một nút đại diện có vị trí nằm ở tâm của nhóm K nút cảm biến giám sát cùng một mục tiêu. Xây dựng đồ thị đầy đủ $G=(V,E)$ có tập đỉnh là các nút đại diện và trạm cơ sở, mỗi cạnh có trọng số $w(u,v)$ là khoảng cách Euclid giữa hai đỉnh.

2. Tìm cây khung nhỏ nhất của đồ thị, cây khung nhỏ nhất gồm các cạnh kết nối hai nút đại diện với nhau hoặc kết nối một nút đại diện với trạm cơ sở sao cho các nút trong cây là liên thông với nhau và tổng độ dài các cạnh trong cây là nhỏ nhất [14]. Cây khung nhỏ nhất tìm được chính là phương án đặt các nút chuyển tiếp để kết nối các nút đại diện với nhau hoặc với trạm cơ sở sao cho tổng số nút chuyển tiếp là nhỏ nhất.

3. Trên mỗi cạnh của cây khung, đặt các nút chuyển tiếp sao cho thỏa mãn ràng buộc K- kết nối giữa hai cụm

K- cảm biến, hoặc giữa một cụm K- cảm biến và trạm cơ sở.

Hình 4 mô tả một ví dụ về các bước thực hiện thuật toán đặt các nút chuyển tiếp trên miền A sao cho thỏa mãn ràng buộc K- kết nối ($K=2$) và số nút chuyển tiếp cần dùng là ít nhất.



Hình 4. Ví dụ mô tả hoạt động của thuật toán giải quyết bài toán K- kết nối

2) Thuật toán

Mã giả của thuật toán được trình bày trong Thuật toán 3:

Thuật toán 3: Giải quyết bài toán K- kết nối

Input: S: Tập vị trí các nút cảm biến
 ST: Danh sách thể hiện nút t_i được bao phủ bởi các nút cảm biến nào trong S
 BS: Vị trí trạm cơ sở
Output: Danh sách $R = \{r_1, r_2, \dots, r_m\}$, vị trí của các nút chuyển tiếp cần thiết lập trên miền A.

// Xây dựng đồ thị

$SC \leftarrow findSC(S, ST)$ // Tạo danh sách các nút đại diện SC

$G \leftarrow buildGraph(SC, BS)$ // Tạo đồ thị $G=(V,E)$

// Thuật toán Prim

$ST_{min} \leftarrow \{BS\}$

$STE_{min} \leftarrow \emptyset$

for $v \in V \setminus ST_{min}$ **do**

$D[v] = w[BS, v]$

$near[v] = BS$

end for

$D[BS] \leftarrow 0; near[BS] \leftarrow BS$

for $i \leftarrow 0$ **to** $|E|$ **do**

$u = findMin(D)$

$ST_{min} \leftarrow ST_{min} + u$

$STE_{min} \leftarrow STE_{min} + (u, near[u])$

for $v \in V \setminus ST_{min}$ **do**

if $D[v] > w[u, v]$ **then**

```

    D[v] ← w[u, v] // w[u,v] là trọng số cạnh (u,v)
    near[v] ← u
end if
end for
end for

// Đặt các nút chuyển tiếp
R ← setNodes(STmin, STmax)
return R

```

IV. THỰC NGHIỆM VÀ ĐÁNH GIÁ

A. Kích bản và dữ liệu thực nghiệm

Để thử nghiệm và đánh giá thuật toán đề xuất trong bài báo, nhóm tác giả thiết kế hai tập dữ liệu thực nghiệm. Tập dữ liệu thứ nhất được thực hiện ở miền phẳng A có kích thước 2000m x 2000m (là miền có kích thước rộng). Trong tập dữ liệu này vị trí các đối tượng ở xa nhau nên số lượng đĩa tròn giao nhau là tương đối ít. Tập dữ liệu thứ hai được thực hiện ở miền phẳng A có kích thước 1000m x 1000m (là miền nhỏ hơn). Trong tập dữ liệu này các vị trí các đối tượng ở gần nhau hơn, do đó số lượng đĩa tròn giao nhau nhiều hơn. Về lý thuyết, giải thuật đề xuất sẽ hoạt động tốt hơn nếu số lượng đĩa tròn giao nhau tăng lên. Vì vậy, giải thuật được kỳ vọng sẽ cho kết quả tốt hơn khi được thử nghiệm với tập dữ liệu thứ hai.

Đối với mỗi tập dữ liệu ở trên, tác giả xây dựng 3 kịch bản thử nghiệm:

Kịch bản 1: Mục đích của kịch bản này là đánh giá sự ảnh hưởng của tham số K lên kết quả thử nghiệm của giải thuật. Để đánh giá tiêu chí này, tác giả xây dựng 7 bộ dữ liệu trên miền rộng (2000m x 2000m) và 7 bộ dữ liệu trên miền hẹp (1000m x 1000m). Trong mỗi bộ dữ liệu, số lượng đối tượng cần bao phủ trên miền là 100 đối tượng và bán kính cảm nhận R_s là 40m. Giá trị K được thay đổi tăng dần trong tập {2, 3, 4, 5, 6, 7, 8} để tạo ra 7 bộ dữ liệu tương ứng.

Kịch bản 2: Mục đích của kịch bản này là đánh giá sự ảnh hưởng của bán kính cảm nhận R_s lên kết quả thử nghiệm của giải thuật. Để đánh giá tiêu chí này, nhóm tác giả xây dựng 7 bộ dữ liệu trên miền rộng (2000m x 2000m) và 7 bộ dữ liệu trên miền hẹp (1000m x 1000m). Trong mỗi bộ dữ liệu, số lượng đối tượng cần bao phủ trên miền là 100 đối tượng và giá trị K là 4. Bán kính cảm nhận R_s (m) được thay đổi tăng dần trong tập {40, 50, 60, 70, 80, 90, 100} để tạo ra 7 bộ dữ liệu tương ứng.

Kịch bản 3: Mục đích của kịch bản này là đánh giá sự ảnh hưởng của số lượng đối tượng lên kết quả thử nghiệm của giải thuật. Để đánh giá tiêu chí này, nhóm tác giả xây dựng 7 bộ dữ liệu trên miền rộng (2000m x 2000m) và 7 bộ dữ liệu trên miền hẹp (1000m x 1000m). Trong mỗi bộ dữ liệu, bán kính cảm nhận R_s là 40 và giá trị K là 4. Số lượng đối tượng cần bao phủ được thay đổi tăng dần trong

Bảng II. Kết quả thực nghiệm theo ba kịch bản trên hai tập dữ liệu

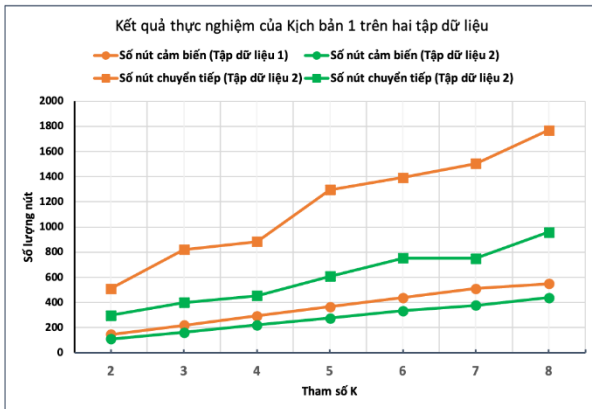
Kích bản					Kết quả					
Kích bản 1 (thay đổi K)					Số nút cảm biến		Số nút chuyển tiếp		Thời gian thực hiện	
Bộ dữ liệu		Số đối tượng	R_s	K	Tập dữ liệu 1	Tập dữ liệu 2	Tập dữ liệu 1	Tập dữ liệu 2	Tập dữ liệu 1	Tập dữ liệu 2
1N100	2N100	100	40	2	146	110	511	297	292	185
				3	219	162	821	400	460	257
				4	292	220	883	454	528	314
				5	365	275	1296	608	737	408
				6	438	334	1394	753	820	502
				7	511	377	1503	751	908	527
				8	548	439	1768	958	1058	647
Kích bản 2 (thay đổi R_s)					Số nút cảm biến		Số nút chuyển tiếp		Thời gian thực hiện	
Bộ dữ liệu		Số đối tượng	R_s	K	Tập dữ liệu 1	Tập dữ liệu 2	Tập dữ liệu 1	Tập dữ liệu 2	Tập dữ liệu 1	Tập dữ liệu 2
1N100	2N100	100	40	4	292	220	883	454	528	314
			50		275	192	733	368	458	262
			60		246	168	736	301	446	221
			70		224	147	554	252	356	189
			80		204	128	485	220	316	165
			90		196	106	414	175	283	134
			100		184	87	348	142	250	109
Kích bản 3 (thay đổi số lượng đối tượng)					Số nút cảm biến		Số nút chuyển tiếp		Thời gian thực hiện	
Bộ dữ liệu		Số đối tượng	R_s	K	Tập dữ liệu 1	Tập dữ liệu 2	Tập dữ liệu 1	Tập dữ liệu 2	Tập dữ liệu 1	Tập dữ liệu 2
1N100	2N100	100	40	4	292	220	883	451	528	314
1N150	2N150	150			456	280	1158	545	737	386
1N200	2N200	200			575	327	1589	610	981	440
1N250	2N250	250			657	396	1786	907	1109	600
1N300	2N300	300			738	426	1799	913	1162	621
1N350	2N350	350			824	464	2397	950	1453	658
1N400	2N400	400			886	472	2534	831	1545	615

tập {100, 150, 200, 250, 300, 350, 400} để tạo ra 7 bộ dữ liệu tương ứng.

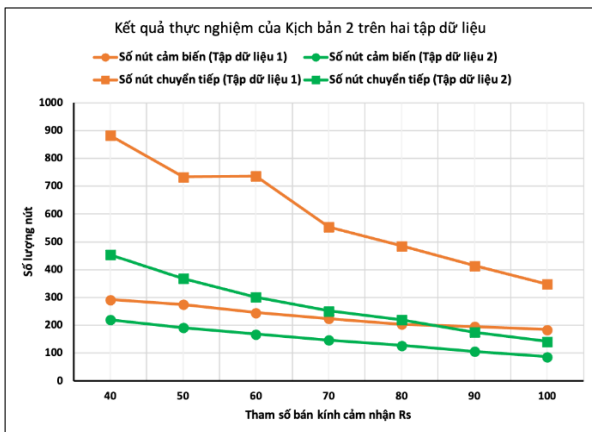
Với tất cả các bộ dữ liệu, tham số bán kính giao tiếp R_c được chọn $R_c=2R_s$ đây cũng là tỉ lệ thường dùng trong các công bố liên quan [5, 6, 8]. Trạm cơ sở B được đặt ở vị trí có tọa độ (5m, 5m) trên miền A. Mã nguồn của các thuật toán được viết bằng ngôn ngữ lập trình Python. Các thử nghiệm được tiến hành theo hình thức mô phỏng trên máy tính có cấu hình Intel Core i5, RAM 16GB, SSD 256GB.

B. Kết quả thực nghiệm

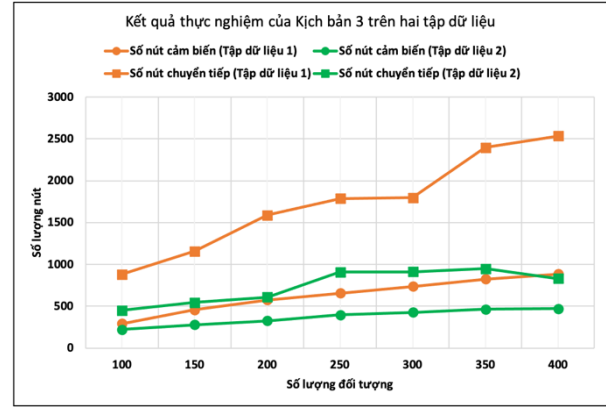
Bảng II trình bày các kết quả thực nghiệm của 3 kịch bản trên hai tập dữ liệu. Đối với cả ba kịch bản thử nghiệm, giải thuật cho kết quả tốt hơn khi hoạt động trên miền có diện tích nhỏ hơn (tập dữ liệu thứ hai). Vì khi miền A có kích thước nhỏ, số lượng các đĩa tròn giao nhau tăng lên, việc đặt nút cảm biến trong miền giao nhau của các đĩa tròn sẽ mang lại hiệu quả tốt hơn. Nhận xét này được thể hiện rõ hơn tại Hình 5, 6, 7 so sánh kết quả thực nghiệm trên hai tập dữ liệu. Khi xem xét ảnh hưởng của các tham số đến kết quả thực nghiệm của giải thuật nhóm tác giả nhận thấy sự phù hợp với các phân tích lý thuyết. Cụ thể, khi giá trị K, hoặc số lượng đối tượng cần bao phủ tăng lên thì số lượng nút cảm biến và nút chuyển tiếp cần thiết lập trên miền A cũng tăng lên tương ứng. Kết quả thực nghiệm giải thuật theo kịch bản số một và kịch bản số ba đã xác nhận điều này. Tương tự như vậy, khi bán kính cảm nhận R_s tăng lên thì số nút cảm biến cần đặt trên miền A giảm, dẫn tới số nút chuyển tiếp cần thiết lập cũng giảm theo, điều này được thể hiện trong kết quả thực nghiệm giải thuật theo kịch bản số hai.



Hình 5. Kết quả thực nghiệm theo kịch bản một



Hình 6. Kết quả thực nghiệm theo kịch bản hai



Hình 7. Kết quả thực nghiệm theo kịch bản ba

C. So sánh với giải thuật di truyền và KGCA

Trong phần này nhóm tác giả so sánh kết quả thực nghiệm giữa giải thuật đề xuất trong bài báo với hai giải thuật đã được công bố trong các nghiên cứu trước đây là giải thuật di truyền (GA) [5] và giải thuật KGCA [6]. Nghiên cứu các giải thuật GA và KGCA, so sánh với giải thuật đề xuất trong bài báo, nhóm tác giả nhận thấy:

- Kết quả của giải thuật GA phụ thuộc khá nhiều vào bước xây dựng tập các vị trí tiềm năng. Trong [5], tập vị trí tiềm năng được chọn bằng cách chia miền $A = W \times H$ thành các ô có kích thước $R_s \times R_s$ sau đó chọn ngẫu nhiên K vị trí trong ô là K thành viên của tập vị trí tiềm năng. Kích thước của miền A cũng ảnh hưởng đến kết quả của giải thuật GA, khi miền A rộng số lượng ô nhiều, việc chọn ngẫu nhiên sẽ dẫn đến tập vị trí tiềm năng không đủ tốt từ đó ảnh hưởng xấu đến kết quả của giải thuật.

- Khi giải quyết vấn đề K- bao phủ, giải thuật KGCA [6] có heuristic tương tự như giải thuật đề xuất trong bài báo. Điểm khác biệt là KGCA chỉ tìm cách đặt các nút cảm biến trên vùng giao nhau của hai đĩa tròn bao quanh mục tiêu. Còn giải thuật đề xuất trong bài báo tìm cách đặt các nút cảm biến trên vùng giao nhau của nhiều đĩa tròn nhất. Thực tế có những trường hợp số lượng đĩa tròn giao nhau nhiều hơn hai, khi đó kết quả của KGCA sẽ không tốt bằng kết quả của giải thuật đề xuất trong bài báo. Đối với bài toán K- kết nối, thực hiện đặt các nút chuyển tiếp trên đường đi ngắn nhất từ nút cảm biến đến trạm cơ sở không phải lúc nào cũng mang lại kết quả tối ưu về số lượng nút chuyển tiếp, như đã phân tích trong ví dụ tại Hình 3.

Tác giả thực nghiệm giải thuật GA, giải thuật KGCA và giải thuật đề xuất với 42 bộ dữ liệu của ba kịch bản trên hai tập dữ liệu miền rộng và miền hẹp như đã mô tả trong mục IV.A. Do ảnh hưởng của các tham số ngẫu nhiên trong thuật toán GA, với mỗi bộ dữ liệu, nhóm tác giả thực hiện chạy thuật toán GA 5 lần và lấy giá trị trung bình của các kết quả thực nghiệm. Kết quả thực nghiệm trên hai tập dữ liệu được trình bày trong Bảng III và Bảng IV.

Bảng III. Kết quả thực nghiệm của ba giải thuật trên tập dữ liệu một

Tập dữ liệu 1							
Kịch bản 1 (Số đối tượng = 100; $R_s = 40$)							
	Tham số K thay đổi						
	2	3	4	5	6	7	8
GA	555	648	864	1863	2145	2948	3810
KGCA	146	219	292	365	438	511	548

Số nút cảm biến	Giải thuật đề xuất	146	219	292	365	438	511	548
Số nút chuyển tiếp	GA	1138	2999	3580	5552	7639	9793	10521
	KGCA	567	754	1139	1320	1632	2075	2162
	Giải thuật đề xuất	511	721	883	1296	1394	1503	1768
Thời gian thực hiện	GA	1488	1588	2144	3739	4343	5686	6494
	KGCA	314	433	631	747	916	1137	1215
	Giải thuật đề xuất	292	460	528	737	820	908	1058
Kịch bản 2 (Số đối tượng = 100; K = 4)								
		Tham số Rs thay đổi						
		40	50	60	70	80	90	100
Số nút cảm biến	GA	864	612	501	352	293	265	256
	KGCA	292	277	255	238	226	216	200
	Giải thuật đề xuất	292	275	246	224	204	196	184
Số nút chuyển tiếp	GA	3580	2308	2211	1251	746	632	623
	KGCA	1139	1070	970	728	704	601	537
	Giải thuật đề xuất	883	733	736	554	485	414	348
Thời gian thực hiện	GA	2144	1690	1185	712	474	412	403
	KGCA	631	594	541	434	417	370	335
	Giải thuật đề xuất	528	458	446	356	316	283	250
Kịch bản 3 (Rs = 40; K = 4)								
		Số đối tượng thay đổi						
		100	150	200	250	300	350	400
Số nút cảm biến	GA	864	1604	2192	2340	2507	2980	2842
	KGCA	292	460	580	784	892	974	1048
	Giải thuật đề xuất	292	456	575	657	738	824	886
Số nút chuyển tiếp	GA	3580	5864	8645	10757	11263	14371	14210
	KGCA	1139	1730	1895	2572	3088	3206	4397
	Giải thuật đề xuất	883	1158	1589	1786	1799	2397	2534
Thời gian thực hiện	GA	2144	3308	4773	5707	6009	7536	7389
	KGCA	631	968	1106	1439	1706	1813	2328
	Giải thuật đề xuất	528	737	981	1109	1162	1453	1545

Bảng IV. Kết quả thực nghiệm của ba giải thuật trên tập dữ liệu hai

Tập dữ liệu 2								
Kịch bản 1 (Số đối tượng = 100; Rs = 40)								
		Tham số K thay đổi						
		2	3	4	5	6	7	8
Số nút cảm biến	GA	242	235	355	495	652	818	1013
	KGCA	111	172	226	282	351	413	475
	Giải thuật đề xuất	110	162	220	275	334	377	439
Số nút chuyển tiếp	GA	506	498	921	1608	2147	2559	4011
	KGCA	302	477	593	687	987	1054	1277
	Giải thuật đề xuất	297	400	454	608	753	751	958
Thời gian thực hiện	GA	516	540	581	940	1250	1514	2212
	KGCA	187	294	373	444	605	669	796
	Giải thuật đề xuất	185	257	314	408	502	527	647
Kịch bản 2 (Số đối tượng = 100; K = 4)								

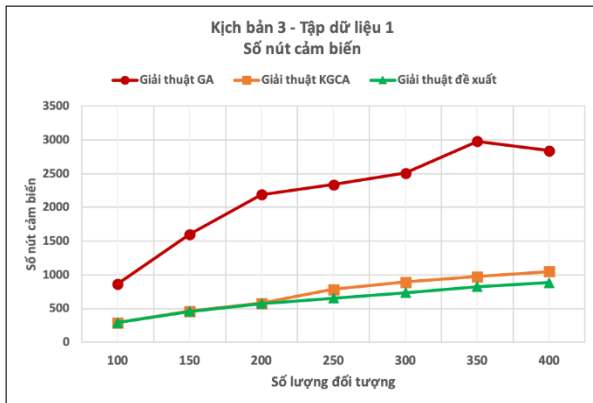
		Tham số Rs thay đổi						
		40	50	60	70	80	90	100
Số nút cảm biến	GA	355	293	244	212	185	158	138
	KGCA	226	223	210	193	178	166	157
	Giải thuật đề xuất	220	192	168	147	128	106	87
Số nút chuyển tiếp	GA	921	560	387	309	291	219	188
	KGCA	593	496	483	370	316	284	233
	Giải thuật đề xuất	454	368	301	252	220	175	142
Thời gian thực hiện	GA	581	400	301	251	227	182	158
	KGCA	373	312	319	264	233	213	187
	Giải thuật đề xuất	314	262	221	189	165	134	109
Kịch bản 3 (Rs = 40; K = 4)								
		Số đối tượng thay đổi						
		100	150	200	250	300	350	400
Số nút cảm biến	GA	355	649	761	1207	1522	1633	1759
	KGCA	226	312	471	631	849	966	1143
	Giải thuật đề xuất	220	280	327	396	426	464	472
Số nút chuyển tiếp	GA	921	1582	2737	4503	5577	5210	5957
	KGCA	593	895	1148	1592	1854	2195	2789
	Giải thuật đề xuất	454	545	610	907	913	950	831
Thời gian thực hiện	GA	581	2133	2194	3114	3200	3063	3559
	KGCA	373	741	1135	1371	1857	1801	2065
	Giải thuật đề xuất	314	386	440	600	621	658	615

Khi thực nghiệm theo kịch bản một trên hai tập dữ liệu, nhìn chung giải thuật KGCA và giải thuật đề xuất cho kết quả tương đương nhau. Nguyên nhân là ở kịch bản một, số lượng các đối tượng và bán kính cảm nhận R_s được thiết lập cố định: 100 đối tượng và 40m. Các đĩa tròn bán kính R_s tạo từ 100 đối tượng ít giao với nhau hoặc không có trường hợp giao nhau của từ ba đĩa tròn trở lên. Do vậy ở kịch bản này, giải thuật đề xuất không thể hiện ưu thế khi so sánh với giải thuật KGCA khi tìm phương án đặt các nút cảm biến trên miền A. Khi xét trên miền hẹp (tập dữ liệu thứ hai), giải thuật đề xuất có tốt hơn giải thuật KGCA một chút: số nút cảm biến giảm khoảng 8%, và số nút chuyển tiếp giảm khoảng 25% khi $K=8$.

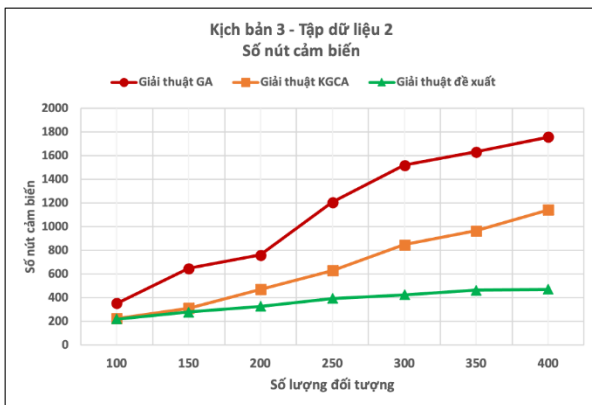
Hai giải thuật KGCA và giải thuật đề xuất trong bài báo đều cho kết quả tốt hơn đáng kể so với giải thuật GA ở cả hai tập dữ liệu, sự chênh lệch này thể hiện rõ hơn khi thực nghiệm ở miền rộng hoặc thực nghiệm với giá trị K lớn. Kết quả thực nghiệm này phù hợp với nhận xét của nhóm tác giả là giải thuật GA không hoạt động tốt khi tập vị trí tiềm năng có số lượng vị trí lớn.

Trong kịch bản hai, mặc dù số lượng đối tượng được thiết lập cố định là 100 đối tượng, nhưng bán kính cảm nhận R_s thay đổi tăng dần từ 40 đến 100. Thiết lập này làm tăng khả năng giao nhau của các đĩa tròn, đặc biệt là khi xét trong miền hẹp (tập dữ liệu thứ hai). Do đó kết quả thực nghiệm của giải thuật đề xuất trong bài báo tốt hơn rõ rệt so với giải thuật KGCA khi thử nghiệm trên tập dữ liệu thứ hai. Một điểm đáng lưu ý trong kịch bản thử nghiệm này là khi giá trị R_s lớn (80, 90, 100), kết quả thực nghiệm của giải thuật GA tiệm cận kết quả của KGCA. Có thể lý giải là khi tập vị trí tiềm năng đủ tốt (kích thước miền A nhỏ hoặc bán kính cảm nhận R_s lớn) thì heuristic của GA

tương đương với heuristic của KGCA và của thuật toán đề xuất trong bài báo.



Hình 8. So sánh số lượng nút cảm biến của ba thuật toán khi thực nghiệm theo kịch bản ba trên tập dữ liệu một



Hình 9. So sánh số lượng nút cảm biến của ba thuật toán khi thực nghiệm theo kịch bản ba trên tập dữ liệu hai

Khi thực nghiệm theo kịch bản thứ ba, tham số K và bán kính cảm nhận R_s được thiết lập cố định, số lượng đối tượng tăng dần từ 100 đến 400. Khi số lượng đối tượng cần bao phủ tăng lên thì khả năng xuất hiện các đĩa tròn giao nhau cũng tăng theo tương ứng, do đó giải thuật đề xuất sẽ cho kết quả tốt hơn. Dữ liệu thực nghiệm cũng thể hiện điều này như trong Hình 8 và Hình 9. Chênh lệch về hiệu quả hoạt động của các giải thuật thể hiện rõ hơn trong các bộ dữ liệu có kích thước miền hẹp (tập dữ liệu 2) và số lượng đối tượng cần bao phủ lớn ($n = \{250, 300, 350, 400\}$). Cụ thể, trên miền hẹp với $n = 250$ giải thuật đề xuất có số lượng nút cảm biến giảm khoảng 68,8% so với giải thuật GA và 37,2% so với giải thuật KGCA; với $n = 400$, tỉ lệ này lần lượt là 73,1% và 58,7%.

V. KẾT LUẬN

Vấn đề triển khai mạng cảm biến không dây đảm bảo các yêu cầu về bao phủ và kết nối là vô cùng quan trọng nhằm duy trì sự thông suốt của dữ liệu và đảm bảo chất lượng mạng. Nghiên cứu này đề xuất một giải thuật Heuristic mới để giải bài toán tối thiểu hóa số nút cảm biến khi triển khai mạng WSN đồng thời thỏa mãn các ràng buộc K- bao phủ và K- kết nối.

Đối với bài toán K- bao phủ, xuất phát từ nhận xét nếu một nút cảm biến được đặt vào vùng giao nhau của các đĩa tròn có tâm là tọa độ của đối tượng, có bán kính là bán kính

cảm biến R_s thì nút cảm biến đó có thể bao phủ nhiều đối tượng hơn, nhóm tác giả đề xuất giải thuật tham lam tìm các vùng giao nhau của nhiều đĩa tròn nhất để đặt nút cảm biến. Kết quả của giải thuật K- bao phủ được sử dụng làm đầu vào của giải thuật K- kết nối. Trong bài toán này, nhận thấy các nút cảm biến sau khi triển khai thỏa mãn ràng buộc K- bao phủ thường phân bố tập trung hình thành các cụm K- cảm biến, nhóm tác giả đã chuyển bài toán K- kết nối thành bài toán tìm cây khung nhỏ nhất của đồ thị đầy đủ $G = (V, E)$ với tập đỉnh là các nút đại diện (nằm ở tâm cụm) và nút trạm cơ sở, trọng số của mỗi cạnh là khoảng cách Euclid giữa hai đỉnh của đồ thị. Sau khi tìm được cây khung nhỏ nhất của đồ thị G , các nút chuyển tiếp sẽ được đặt theo các cạnh của cây khung sao cho thỏa mãn ràng buộc K- kết nối.

Tác giả xây dựng hai tập dữ liệu thực nghiệm, với ba kịch bản thực nghiệm khác nhau được áp dụng trên mỗi tập dữ liệu. Trong mỗi kịch bản thực nghiệm, nhóm tác giả thay đổi các tham số để tạo ra bảy bộ dữ liệu tương ứng. Giải thuật đề xuất và hai giải thuật khác là giải thuật di truyền và giải thuật KGCA được thực nghiệm trên 42 bộ dữ liệu nói trên. Các kết quả thực nghiệm cho thấy giải thuật đề xuất có hiệu quả tốt hơn so với hai giải thuật còn lại trong đa số bộ dữ liệu thử nghiệm, kể cả về số lượng nút cảm biến và số lượng nút chuyển tiếp cần thiết lập để thỏa mãn các ràng buộc K- bao phủ và K- kết nối. Chênh lệch về hiệu quả hoạt động của các giải thuật thể hiện rõ hơn trong các bộ dữ liệu có kích thước miền hẹp hoặc số lượng đối tượng cần bao phủ lớn ($n = \{300, 350, 400\}$). Khi bán kính cảm nhận R_s có giá trị lớn ($R_s = \{80, 90, 100\}$), hoặc tham số K có giá trị nhỏ ($K = \{2, 3\}$) thì kết quả của các giải thuật chênh lệch không đáng kể.

Nghiên cứu còn có một số hạn chế như: i) chưa tính đến sự ảnh hưởng của các vật cản trên miền phẳng chứa các đối tượng cần giám sát; ii) chưa tính toán tối ưu các tham số khác của mạng WSN như: năng lượng tiêu thụ của các cảm biến, khả năng chịu lỗi, độ trễ truyền dữ liệu, hay chất lượng dịch vụ của toàn mạng. Trong tương lai, nhóm tác giả dự định sẽ mở rộng nghiên cứu để khắc phục những tồn tại này.

LỜI CẢM ƠN

Nghiên cứu được tài trợ bởi đề tài nghiên cứu cấp cơ sở của Trường Đại học Hà Nội, tên đề tài “Tối thiểu hóa số lượng cảm biến khi triển khai mạng cảm biến không dây thỏa mãn ràng buộc đa bao phủ và đa kết nối”, mã số “DTGV-2025-01”.

TÀI LIỆU THAM KHẢO

- [1] M. A. Jamshed, K. Ali, Q. H. Abbasi, M. A. Imran and M. Ur-Rehman, "Challenges, Applications, and Future of Wireless Sensors in Internet of Things: A Review," in *IEEE Sensors Journal*, vol. 22, no. 6, pp. 5482-5494, 15 March 15, 2022, doi: 10.1109/JSEN.2022.3148128
- [2] Ijmaru GK, Ang KL-M, Seng JK. Wireless power transfer and energy harvesting in distributed sensor networks: Survey, opportunities, and challenges. *International Journal of Distributed Sensor Networks*. 2022;18(3). doi:10.1177/15501477211067740
- [3] Tamaris K, Preka I, Kandris D, Alexandridis A. Coverage and k-Coverage Optimization in Wireless Sensor Networks Using Computational Intelligence Methods: A Comparative Study. *Electronics*. 2020; 9(4):675. <https://doi.org/10.3390/electronics9040675>

- [4] Gong J. An application of meta-heuristic and nature-inspired algorithms for designing reliable networks based on the Internet of things: A systematic literature review. *Int J Commun Syst.* 2023; 36(5):e5416. doi:10.1002/dac.5416
- [5] Suneet Kumar Gupta, Pratyay Kuila, Prasanta K. Jana. Genetic algorithm approach for k-coverage and m-connected node placement in target based wireless sensor networks. *Computers & Electrical Engineering.* Vol 56, 2016, pp 544-556, ISSN 0045-7906, https://doi.org/10.1016/j.compeleceng.2015.11.009.
- [6] N. T. Hanh, H. T. T. Binh, N. Q. Huy, N. Van Thanh and B. T. Q. Mai, "Node placement optimization under K-Coverage and K-Connectivity constraints in wireless sensor networks," 2022 IEEE Symposium Series on Computational Intelligence (SSCI), Singapore, Singapore, 2022, pp. 01-08, doi: 10.1109/SSCI51031.2022.10022183.
- [7] M. Elhoseny, A. Tharwat, A. Farouk and A. E. Hassanien, "K-Coverage Model Based on Genetic Algorithm to Extend WSN Lifetime," in *IEEE Sensors Letters*, vol. 1, no. 4, pp. 1-4, Aug. 2017, Art no. 7500404, doi: 10.1109/LSSENS.2017.2724846.
- [8] N. Ebrahimian, G. Y. Sheramin, A. H. Navin and Z. Foruzandeh, "A Novel Approach for Efficient k-Coverage in Wireless Sensor Networks by Using Genetic Algorithm," 2010 International Conference on Computational Intelligence and Communication Networks, Bhopal, India, 2010, pp. 372-376, doi: 10.1109/CICN.2010.148.
- [9] S. Mnasri, A. Thaljaoui, N. Nasri and T. Val, "A genetic algorithm-based approach to optimize the coverage and the localization in the wireless audio-sensors networks," 2015 International Symposium on Networks, Computers and Communications (ISNCC), Yasmine Hammamet, Tunisia, 2015, pp. 1-6, doi: 10.1109/ISNCC.2015.7238591.
- [10] Manju, S. Singh, S. Kumar, A. Nayyar, F. Al-Turjman and L. Mostarda, "Proficient QoS-Based Target Coverage Problem in Wireless Sensor Networks," in *IEEE Access*, vol. 8, pp. 74315-74325, 2020, doi: 10.1109/ACCESS.2020.2986493.
- [11] Sheikhi, H., Hoseini, M. & Sabaei, M. k-Connected Relay Node Deployment in Heterogeneous Wireless Sensor Networks. *Wireless Pers Commun* 120, 3277–3292 (2021). https://doi.org/10.1007/s11277-021-08613-x
- [12] V. K. Akram, O. Dagdeviren and B. Tavli, "A Coverage-Aware Distributed k-Connectivity Maintenance Algorithm for Arbitrarily Large k in Mobile Sensor Networks," in *IEEE/ACM Transactions on Networking*, vol. 30, no. 1, pp. 62-75, Feb. 2022, doi: 10.1109/TNET.2021.3104356.
- [13] Phi Le Nguyen, Nguyen Thi Hanh, Nguyen Tien Khuong, Huynh Thi Thanh Binh, Yusheng Ji. Node placement for connected target coverage in wireless sensor networks with dynamic sinks. *Pervasive and Mobile Computing*, Vol 59, 2019, 101070, ISSN 1574-1192, https://doi.org/10.1016/j.pmcj.2019.101070.
- [14] P. Ayegba, J. Ayoola, E. Asani and A. Okeyinka, "A Comparative Study Of Minimal Spanning Tree Algorithms," 2020 International Conference in Mathematics, Computer Engineering and Computer Science (ICMCECS), Ayobo, Nigeria, 2020, pp. 1-4, doi: 10.1109/ICMCECS47690.2020.240900.

MINIMIZING NODE DEPLOYMENT OF WIRELESS SENSOR NETWORK TO SATISFY MULTI-COVERAGE AND MULTI-CONNECTIVITY CONSTRAINTS

Abstract: This paper proposes a new heuristic algorithm to simultaneously address two critical problems in the deployment of wireless sensor networks: minimizing the number of required network nodes while satisfying the constraints of K-coverage and K-connectivity. To evaluate the effectiveness of the proposed algorithm, the authors compare experimental results with those of two algorithms published in recent studies. Experiments were conducted on 42 datasets from two data collections across three test scenarios. The results demonstrate that the proposed algorithm outperforms the other two algorithms GA [5] and KGCA [6] in the majority of datasets, both in terms of the number of sensor nodes and relay nodes required.

Keywords: Wireless sensor network, K-coverage, K-connectivity, Node placement optimization



Nguyễn Xuân Thắng, Nhận học vị Tiến sỹ năm 2016 tại Đại học tổng hợp Kassel, Cộng hòa liên bang Đức. Từ năm 2006 tới nay, tác giả công tác tại Khoa Công nghệ thông tin - Trường Đại học Hà Nội. Lĩnh vực nghiên cứu: Công nghệ mạng không dây, tối ưu hóa và học máy, phát triển phần mềm hướng mô hình.

Email: nxthang@hanu.edu.vn



Nguyễn Thị Hạnh, Nhận học vị Tiến sỹ năm 2020. Hiện công tác tại Khoa Công nghệ số liên ngành, Đại học Phenikaa Lĩnh vực nghiên cứu: tối ưu triển khai mạng cảm biến không dây, các thuật toán trong tính toán tiến hoá, trí tuệ nhân tạo.

Email: hanh.nguyenthi@phenikaa-uni.edu.vn



Trịnh Thị Thu Huyền, Tốt nghiệp cử nhân Công nghệ thông tin năm 2023. Hiện công tác tại Khoa Công nghệ thông tin - Trường Đại học Hà Nội. Lĩnh vực nghiên cứu: Công nghệ mạng không dây, điện toán đám mây.

Email: huyenfit@hanu.edu.vn