

Một thuật toán định tuyến cân bằng năng lượng trong mạng cảm biến không dây dựa trên SDN

Lê Đức Huy¹, Lê Hữu Bình², Đỗ Thành Công¹, Nguyễn Đỗ Hoàng Giang³

¹ Khoa Công nghệ Thông tin, Trường Đại học Kinh doanh và Công nghệ Hà Nội

² Khoa Công nghệ Thông tin, Trường Đại học Khoa học, Đại học Huế

³ Trường Trung học Phổ thông chuyên Khoa học Tự nhiên, Đại học Quốc gia Hà Nội

Tác giả liên hệ: Lê Hữu Bình, lhbinh@hueuni.edu.vn

Ngày nhận bài: 18/07/2023, ngày sửa chữa: 26/09/2023, ngày duyệt đăng: 25/11/2023

Định danh DOI: 10.32913/mic-ict-research-vn.v2023.n2.1240

Tóm tắt: Trong mạng cảm biến không dây (WSN), việc sử dụng năng lượng sao cho hiệu quả để kéo dài thời gian hoạt động của các nút cảm biến là điều cần thiết. Trong bài báo này, chúng tôi đề xuất một thuật toán định tuyến có xét đến mức tiêu thụ năng lượng tại các nút cảm biến. Mục tiêu của thuật toán được đề xuất là cân bằng mức tiêu thụ năng lượng, giảm thiểu số nút phải tiêu thụ nhiều năng lượng nhằm tăng thời gian hoạt động của chúng. Phương pháp của chúng tôi là xây dựng một hàm trọng số cho các kết nối có chứa tham số về năng lượng còn lại tại các nút. Sau đó, sử dụng kỹ thuật định tuyến tập trung dựa trên kiến trúc mạng điều khiển bằng phần mềm (SDN) để tìm lộ trình có trọng số tốt nhất sử dụng cho việc truyền dữ liệu. Kết quả mô phỏng sử dụng OMNeT++ cho thấy rằng, thuật toán được đề xuất cho phép tăng thời gian hoạt động của các nút, tăng thông lượng so với các thuật toán định tuyến phổ biến hiện hành.

Từ khóa: Mạng cảm biến không dây, định tuyến cân bằng năng lượng, SDN

Title: An Energy-Balanced Routing Algorithm in SDN-based Wireless Sensor Networks

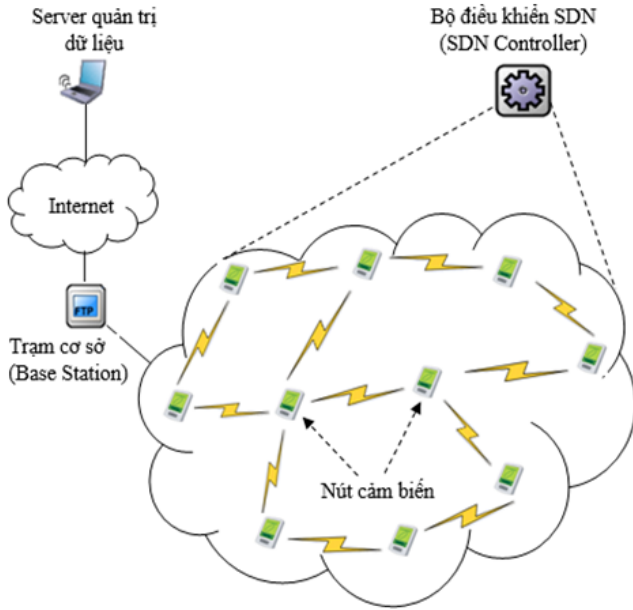
Abstract: In wireless sensor networks (WSN), it is necessary to use energy efficiently to extend the operating time of sensor nodes. In this study, we propose a routing algorithm that considers the energy consumption between sensor nodes. The goal of the proposed algorithm is to balance energy consumption and minimize the number of nodes that must consume a large amount of energy to increase their uptime. Our method constructs a weight function for wireless links that contains parameters for the energy remaining at the nodes. Then, a centralized routing mechanism based on a software-defined network (SDN) architecture is used to find the best weight route for data transfer. Simulation results on OMNeT++ show that the proposed algorithm increases the uptime of nodes and network throughput compared to the current popular routing algorithms.

Keywords: Wireless Sensor Networks, energy-balanced routing, software-defined network

I. GIỚI THIỆU

Thời gian gần đây, mạng cảm biến không dây (Wireless Sensor Networks - WSN) được sử dụng rộng rãi trong nhiều lĩnh vực, đặc biệt là trong giai đoạn bùng nổ các ứng dụng trên nền tảng IoT như hiện nay. Mạng WSN có thể hoạt động theo ba mô hình điều khiển khác nhau, đó là điều khiển theo mô hình tập trung, điều khiển theo mô hình phân cấp và điều khiển theo mô hình phân tán [1]. Với mô hình điều khiển tập trung, có một nút chứa thông tin toàn cục trong hệ thống mạng. Các chức năng điều khiển việc truyền dữ liệu từ các nút mạng về trạm cơ sở (Base Station

- BS) được thực hiện tại nút này. Với mô hình điều khiển phân cấp, toàn bộ các nút được chia thành từng cụm, mỗi cụm có một nút đóng vai trò như là “cụm trưởng”. Việc truyền dữ liệu từ các nút về BS được thực hiện thông qua nút cụm trưởng. Với mô hình điều khiển phân tán, các nút đóng vai trò ngang hàng nhau. Các giao thức điều khiển được thực hiện độc lập tại mỗi nút mạng. Trong giai đoạn gần đây, kiến trúc điều khiển tập trung dựa trên giải pháp mạng kiểm soát bằng phần mềm (Software Defined Networking - SDN) đã được áp dụng cho mạng WSN [2–10]. Nguyên lý cơ bản của mô hình này như cho thấy ở Hình 1. Các nút cảm biến được kết nối bộ điều khiển SDN một cách trực



Hình 1. Một ví dụ của mạng cảm biến không dây sử dụng SDN

tiếp hoặc gián tiếp (thông qua các nút khác) sử dụng giao thức luồng mở (open flow). Các giao thức điều khiển hoạt động của hệ thống mạng như định tuyến, báo hiệu, điều khiển tô-pô, v.v được thực hiện tập trung tại bộ điều khiển SDN. Mô hình WSN sử dụng SDN đã được triển khai để cải tiến hầu hết các giao thức điều khiển trong mạng WSN, điển hình như các giao thức định tuyến [4, 5, 10–12], điều khiển tô-pô [6, 7, 13], điều khiển phân cụm [8, 9]. Trong đó, định tuyến sử dụng SDN là chủ đề nhận được sự quan tâm của nhiều nhóm nghiên cứu trong nước và trên thế giới thời gian gần đây.

Trong [4], S. Wang cùng các cộng sự đã đề xuất một giao thức truyền thông tích hợp, được đặt tên là MINI-FLOW. Giao thức này tập trung vào ba cơ chế định tuyến dữ liệu, định tuyến đường lên (từ các nút cảm biến đến bộ điều khiển SDN), định tuyến đường xuống (từ bộ điều khiển SDN đến các nút cảm biến) và cung cấp thông tin khi khởi tạo mạng. Cơ chế định tuyến được thiết kế dựa trên một hàm trọng số heuristic bao gồm ba giá trị, tổng số bước truyền đến nút đích, cường độ tín hiệu nhận được và năng lượng còn lại. Bằng phương pháp thực nghiệm, các tác giả đã cho thấy giao thức MINI-FLOW mang lại hiệu quả cao khi xem xét vấn đề kết hợp giữa định tuyến và cân bằng tải. Trong [5], nhóm tác giả đã đề nghị một kiến trúc gọi là phần mềm dựa trên đa luồng cho WSN. Cơ chế này được phát triển dựa trên SDN và phân bổ các kênh riêng biệt cho lớp dữ liệu và lớp điều khiển. Giải pháp đề xuất được sử dụng để duy trì cấu trúc liên kết và quản lý việc sử dụng nguồn năng lượng. Các kết quả thực nghiệm cho thấy rõ cơ chế được đề xuất mang lại hiệu quả cao về

mặt thông lượng, độ trễ và mức tiêu thụ năng lượng trong mạng. Trong [10], một cơ chế định tuyến có xét đến năng lượng dựa trên SDN được đề xuất để cải thiện hiệu quả sử dụng năng lượng của WSN với hệ thống IIoT để hỗ trợ công nghiệp 4.0. Bộ điều khiển SDN ước tính mức năng lượng tại các nút quan trọng trong WSN và quyết định lộ trình tốt nhất dựa trên tình trạng sử dụng năng lượng của chúng thay vì tiêu chí đường đi ngắn nhất được sử dụng rộng rãi. Sử dụng phương pháp mô phỏng, nhóm tác giả đã cho thấy rõ phương pháp được đề nghị cho phép giảm thiểu tình trạng các nút của WSN cạn kiệt nguồn năng lượng và làm gián đoạn việc truyền dữ liệu một cách đột ngột.

Thông qua các công trình đã công bố được khảo sát ở trên chúng tôi nhận thấy rằng, việc ứng dụng kiến trúc SDN để điều khiển định tuyến trong mạng WSN là một giải pháp hữu hiệu. Đây cũng là giải pháp phù hợp với xu hướng phát triển của công nghệ mạng truyền thông không dây thế hệ mới. Trong công trình nghiên cứu này, chúng tôi cũng sử dụng kiến trúc SDN vào việc điều khiển định tuyến trong mạng WSN. Các đóng góp mới của công trình này được liệt kê như sau:

- Đề xuất một hàm để tính toán trọng số cho tất cả các kết nối không dây trong mạng WSN. Hàm trọng số này có chứa các tham số về tỷ lệ giữa năng lượng khởi đầu và năng lượng còn lại tại các nút.
- Đề xuất một thuật toán định tuyến mới với mục tiêu cân bằng việc sử dụng năng lượng cho mạng WSN dựa trên kiến trúc SDN. Thuật toán có tên là SEBR (SDN-based Energy Balanced Routing).

Các phần tiếp theo của bài báo được tổ chức như sau. Phần II tập trung trình bày thuật toán định tuyến được đề xuất. Phần III phân tích các kịch bản mô phỏng và thảo luận về các kết quả. Cuối cùng kết luận các kết quả đã đạt được và nêu ra hướng phát triển tiếp theo của chủ đề nghiên cứu này, được trình bày trong phần IV.

II. THUẬT TOÁN ĐỊNH TUYẾN ĐƯỢC ĐỀ XUẤT

Nội dung chính của phần này trình bày thuật toán định tuyến cân bằng năng lượng tiêu thụ tại các nút, được đề xuất cho mạng WSN dựa trên kiến trúc SDN, được đặt tên SEBR (SDN-based Energy Balanced Routing). Thuật toán SEBR hoạt động theo nguyên lý định tuyến tập trung, sử dụng một hàm trọng số có xét đến tỷ số giữa năng lượng khởi đầu và năng lượng còn lại tại mỗi nút cảm biến. Hàm trọng số này được sử dụng làm tiêu chí để tìm tập lộ trình từ mỗi nút đến trạm cơ sở sử dụng cho việc truyền dữ liệu cảm biến về server quản trị dữ liệu. Để tìm tập lộ trình, thuật toán tìm "đường đi ngắn nhất" được thực thi tại bộ điều khiển SDN. "Đường đi ngắn nhất" được hiểu theo nghĩa tổng trọng số theo hàm được đề xuất là nhỏ nhất.

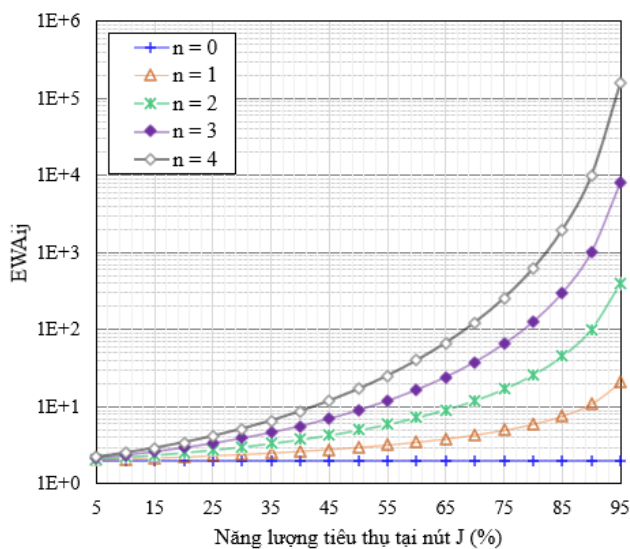
1. Hàm trọng số có xét đến mức tiêu thụ năng lượng tại mỗi nút

Mục tiêu của thuật toán SEBR là cân bằng mức năng lượng tiêu thụ tại các nút, giảm thiểu tình trạng một số nút tiêu thụ năng lượng lớn dẫn đến nhanh chóng bị tắt do hết năng lượng. Trong thuật toán SEBR, việc cân bằng mức tiêu thụ năng lượng được thực hiện thông qua một hàm trọng số có xét đến tỷ số giữa năng lượng khởi đầu và năng lượng còn lại tại mỗi nút, được đặt tên là EAW (Energy Awake Weight), được định nghĩa như sau:

$$EWA_{ij} = \begin{cases} w_{ij} + \left(\frac{E_j^{(s)}}{E_j^{(r)}}\right)^n & \text{nếu } E_j^{(r)} \geq 1\% \times E_j^{(s)} \\ \infty & \text{trường hợp ngược lại} \end{cases} \quad (1)$$

trong đó, w_{ij} là trọng số của đường truyền không dây từ nút I đến nút J , $E_j^{(s)}$ là năng lượng tại thời điểm nút J bắt đầu hoạt động và $E_j^{(r)}$ là năng lượng còn lại của nút J tại thời điểm xem xét, n là một số nguyên không âm.

Với hàm trọng số được thiết lập như ở (1), khi năng lượng còn lại của nút đích của mỗi kết nối càng nhỏ thì trọng số của các kết nối đến nút đó càng lớn. Trọng số tăng theo hàm bậc n của tỷ số giữa năng lượng khởi đầu và năng lượng còn lại. Sự phụ thuộc của trọng số EWA_{ij} vào năng lượng tiêu thụ tại nút J và bậc n theo phương trình (1) được thể hiện rõ ở Hình 2. Chúng ta dễ dàng quan sát rằng, khi bậc $n = 0$, trọng số EWA_{ij} là một hằng số vì nó không còn phụ thuộc vào năng lượng tiêu thụ tại nút J . Trường hợp này tương đương với hàm trọng số là hop-count, được sử dụng mặc định trong hầu hết cơ chế thức định tuyến của mạng WSN. Với các trường hợp còn lại, bậc n càng tăng



Hình 2. Ảnh hưởng của năng lượng tiêu thụ tại nút J và bậc n đến trọng số EWA_{ij} theo hàm (1)

thì trọng số EWA_{ij} càng tăng đột biến khi mức tiêu thụ năng lượng tại nút J lớn, nghĩa là năng lượng còn lại nhỏ. Khi sử dụng hàm trọng số này để tìm tập lộ trình tốt nhất (nghĩa là tập lộ trình có trọng số nhỏ nhất), các lộ trình này sẽ hạn chế đi qua các nút có năng lượng còn lại nhỏ, nhằm kéo dài thời gian hoạt động của các nút này. Trong công trình này, chúng tôi xác định bậc n sao cho việc sử dụng năng lượng tại các nút là cân bằng nhất, nhằm giảm thiểu tình trạng một số nút tiêu thụ năng lượng lớn do tải lưu lượng cao. Việc xác định n được thực hiện thông qua mô phỏng ở phần sau.

2. Thuật toán SEBR

Thuật toán 1 là mã giả lập của các bước thực thi thuật toán định tuyến được đề xuất, SEBR. Thuật toán này được thực thi tại bộ điều khiển SDN. Việc tính toán tập lộ trình đến BS cho tất cả các nút cảm biến được thực hiện định kỳ với một khoảng thời gian định trước, mà đó là IntervalTime ở bước 2. Cứ mỗi lần cập nhật lại bảng định tuyến, trọng số của tất cả các kết nối trong mạng được tính toán lại dựa trên tỷ số giữa năng lượng khởi đầu và năng lượng còn lại tại các nút cảm biến ở thời điểm xem xét. Trọng số này được cập nhật theo phương trình (1) (bước 5 trong thuật toán). Sau đó, bộ điều khiển SDN thực thi thuật toán “đường ngắn nhất”, cụ thể là thuật toán Dijkstra được sử dụng trong mô hình này để tìm tập lộ trình từ tất cả các nút cảm biến đến BS (bước 7). “Đường ngắn nhất” trong thuật toán SEBR được hiểu theo nghĩa tổng trọng số EWA được xác định theo phương trình (1) là nhỏ nhất. Thông tin các lộ trình tìm được sẽ được cập nhật đến các nút cảm biến sử dụng các gói điều khiển (bước 8).

Thuật toán 1: Tìm tập lộ trình từ các nút cảm biến đến BS trong mạng WSN sử dụng SDN

```

1 while (true) do
2     wait(IntervalTime);
3     Cập nhật năng lượng còn lại tại tất cả các nút cảm biến trong mạng thông qua các gói điều khiển;
4     for (mỗi kết nối từ nút I đến nút J) do
5         | Tính trọng số  $EWA_{ij}$  theo (1);
6     end
7     Thực thi thuật toán “đường ngắn nhất” để tìm lộ trình từ các nút cảm biến đến BS;
8     Cập nhật thông tin các lộ trình tìm được đến các nút cảm biến thông qua các gói điều khiển;
9 end

```

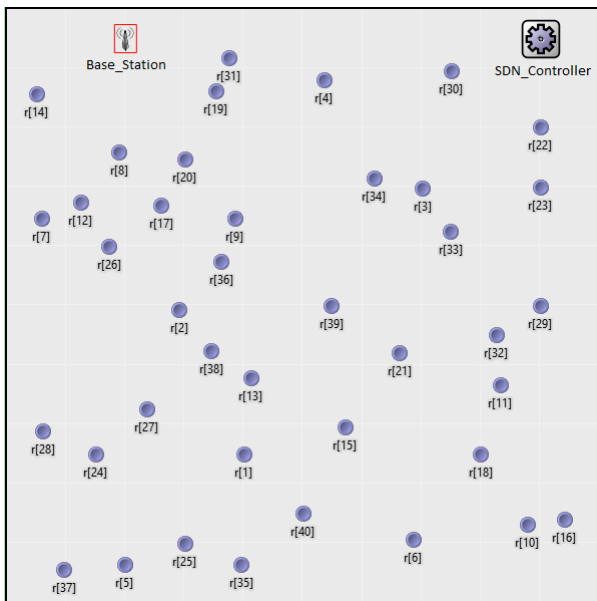
III. ĐÁNH GIÁ THỰC THI BẰNG MÔ PHỎNG

1. Kích bản mô phỏng

Để đánh giá hiệu quả thực thi của thuật toán định tuyến được đề xuất, SEBR, chúng tôi sử dụng phương pháp mô phỏng, so sánh với hai giao thức định tuyến điển hình thường được sử dụng trong mạng WSN, đó là DSDV (Destination-Sequenced Distance-Vector) và AODV (Ad Hoc On-Demand Distance Vector) [14] về mặt năng lượng tiêu thụ, thời gian hoạt động của các nút và thông lượng trung bình trong toàn mạng. Mô phỏng được thực thi trên OMNeT++ [15] và INET Framwork [16]. Các kịch bản mô phỏng được cài đặt như ở Bảng 1. Chúng tôi thiết lập một mạng WSN trong vùng diện tích $1000 \times 1000 [m^2]$. Tổng số nút cảm biến biến đổi trong khoảng từ 30 đến 50 tùy theo kịch bản mô phỏng. Cụ thể là sử dụng 5 kịch bản với tổng số nút 30, 35, 40, 45 và 50. Các nút được bố trí

Bảng I
CÁC THAM SỐ CỦA KỊCH BẢN MÔ PHỎNG

| Tham số | Giá trị |
|--------------------------------|--------------------------|
| Vùng diện tích mô phỏng | $1000 \times 1000 [m^2]$ |
| Tổng số nút cảm biến | Từ 30 đến 50 |
| Giao thức MAC | IEEE 802.11ac |
| Tốc độ dữ liệu | 54 [Mbps] |
| Công suất phát | 12 [dBm] |
| Độ nhạy thu | -76 [dBm] |
| Thuật toán định tuyến | DSDV, AODV and SEBR |
| Loại lưu lượng | UDP |
| Mô hình truyền sóng | Không gian tự do |
| Tần số sóng mang | 2,4 [GHz] |
| Năng lượng ban đầu của các nút | 2 [J] |
| Năng lượng còn lại khi nút tắt | $\leq 1\%$ |



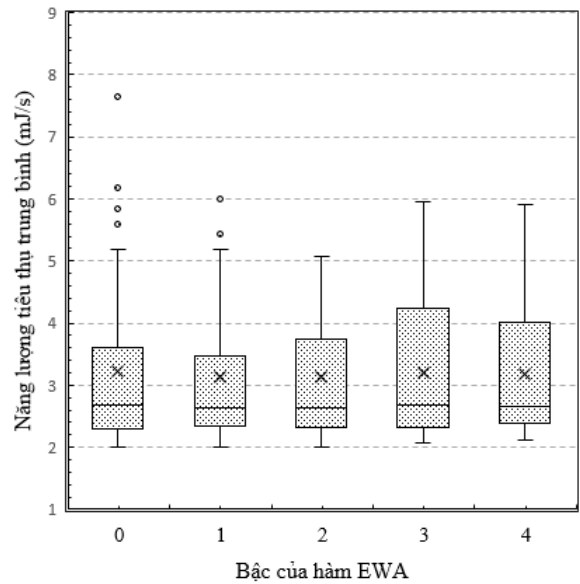
Hình 3. Một bản chụp của giao diện chương trình mô phỏng mạng WSN sử dụng SDN

ngẫu nhiên trong toàn bộ diện tích mạng. Hình 2 cho thấy một bản chụp giao diện trong quá trình thực thi mô phỏng với kịch bản 40 nút.

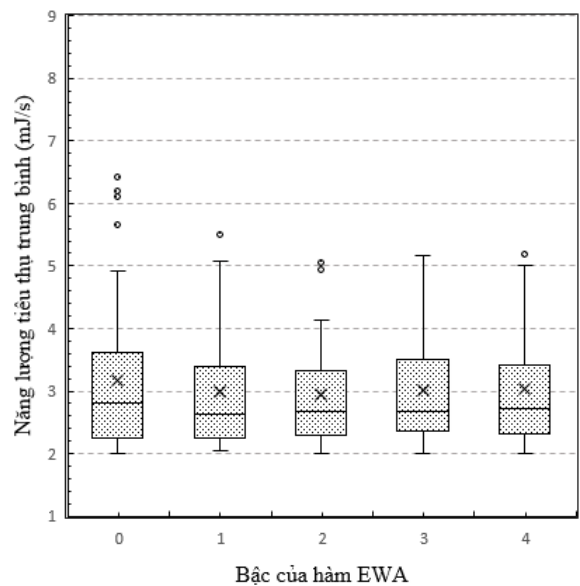
2. Kết quả mô phỏng

a) Xác định bậc n của hàm trọng số EWA_{ij}

Nội dung chính của phần này thực hiện một số kịch bản mô phỏng để xác định bậc n của hàm trọng số EWA_{ij} sao



Hình 4. So sánh mức tiêu thụ năng lượng trung bình trên tất cả các nút khi bậc của hàm trọng số (1) khác nhau khi tổng số nút cảm biến là 35

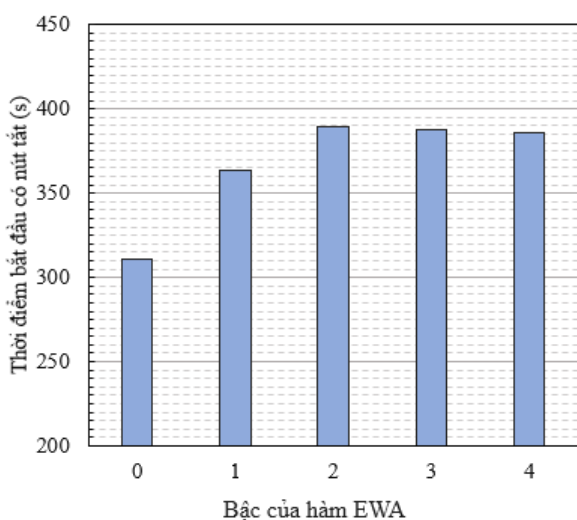


Hình 5. So sánh mức tiêu thụ năng lượng trung bình trên tất cả các nút khi bậc của hàm trọng số (1) khác nhau khi tổng số nút cảm biến là 45

cho hiệu quả sử dụng năng lượng tại các nút cảm biến là tốt nhất. Kết quả ở các Hình 4 và 5 cho thấy rõ sự khác nhau về năng lượng tiêu thụ trung bình tại tất cả các nút cảm biến khi bậc n khác nhau. Xét kịch bản có 35 nút cảm biến (kết quả ở Hình 4), khi $n = 0$ (tương đương với trường hợp định tuyến theo trọng số hop-count), mức tiêu thụ năng lượng tại các nút khác nhau rất lớn, phân bố trong khoảng từ 2,01 mJ/s đến 7,75 mJ/s. Đặc biệt là có 4 điểm ngoại lệ, đó là 4 nút có mức tiêu thụ năng lượng cao, các nút này sẽ sớm bị tắt do hết năng lượng. Khi $n = 1$, mức tiêu thụ năng lượng tại các nút cân bằng hơn, phân bố trong khoảng từ 2,01 mJ/s đến 5,99 mJ/s. Tuy nhiên, vẫn còn 2 nút ngoại lệ với mức tiêu thụ năng lượng cao. Xét các trường hợp còn lại của n ta thấy rằng, trường hợp $n = 2$ cho kết quả tốt nhất, mức tiêu thụ năng lượng của tất cả các nút cân bằng nhất với vùng giá trị từ 2,00 mJ/s đến 5,02 mJ/s và không có ngoại lệ nào. Kết quả đạt được hoàn toàn tương tự đối với kịch bản mà tổng số nút cảm biến là 45 (Hình 5), trường hợp $n = 2$ mang lại mức tiêu thụ năng lượng tại tất cả các nút cân bằng nhất.

Một độ đo hiệu năng khác có tác động lớn đến hiệu năng của hệ thống mạng WSN là thời gian hoạt động của các nút cảm biến. Chúng tôi cũng đã phân tích độ đo này để các định bậc n của hàm (1) sao cho thời gian hoạt động của các nút cảm biến là dài nhất. Kết quả thu được như ở Hình 6 cho thấy rằng, trường hợp bậc $n = 2$ mang lại kết quả tốt nhất.

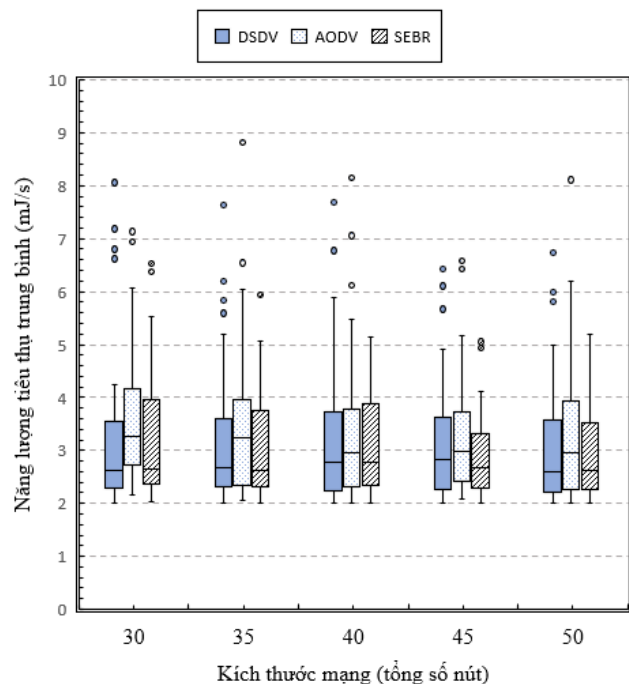
Trên cơ sở các kết quả khảo sát ảnh hưởng của n đến việc sử dụng năng lượng ở trên, chúng tôi chọn bậc n của hàm trọng số (1) là 2, giá trị này được sử dụng cho các kịch bản mô phỏng ở các phần còn lại dưới đây.



Hình 6. So sánh thời điểm bắt đầu có nút tắt do hết năng lượng khi bậc của hàm trọng số (1) khác nhau trong trường hợp tổng số nút cảm biến là 45

b) Phân tích mức tiêu thụ năng lượng tại mỗi nút

Trong phần này, chúng tôi so sánh mức tiêu thụ năng lượng tại tất cả các nút cảm biến của thuật toán được đề xuất so với các thuật toán DSDV và AODV. Kết quả thu được như cho thấy ở Hình 7. Khi sử dụng các thuật toán DSDV và AODV, có một số nút tiêu thụ năng lượng rất lớn, đó là các điểm ngoại lệ của các đồ thị hộp. Ví dụ, với kịch bản 30 nút và việc định tuyến được thực hiện bởi giao thức DSDV, có 4 ngoại lệ với giá trị trung bình là 6,61, 6,82, 7,23 và 8,11 mJ/s. Với kịch bản 35 nút cũng có 4 nút ngoại lệ tiêu thụ năng lượng lớn với mức trung bình là 5,51, 5,72, 6,15 và 7,82 mJ/s. Kết quả cũng hoàn toàn tương tự khi giao thức AODV được sử dụng để tìm tập lộ trình, tất cả các kịch bản mô phỏng với giao thức định tuyến này đều tồn tại các nút có mức tiêu thụ năng lượng cao. Cụ thể, với kịch bản 40 nút, có 3 nút tiêu thụ năng lượng lớn. Mức tiêu thụ trung bình của ba nút này là 8,16, 7,05 và 6,11 mJ/s. Các nút này sẽ bị tắt sớm do hết năng lượng, điều này ảnh hưởng đến việc truyền dữ liệu trong mạng. Nguyên nhân dẫn đến tình trạng này là do các nút trên (nút có mức tiêu thụ năng lượng là các điểm ngoại lệ trong đồ thị hộp) chịu tải lưu lượng lớn, bao gồm cả tải lưu lượng do chính nút đó phát ra và tải lưu lượng từ các nút khác truyền đến để chuyển tiếp về BS. Vì vậy, các nút này phải thường xuyên hoạt động với cường độ cao, dẫn đến tiêu thụ nhiều năng lượng. Với thuật toán SEBR, có nhiều điểm ngoại lệ đã được giải quyết. Điều này đồng nghĩa với

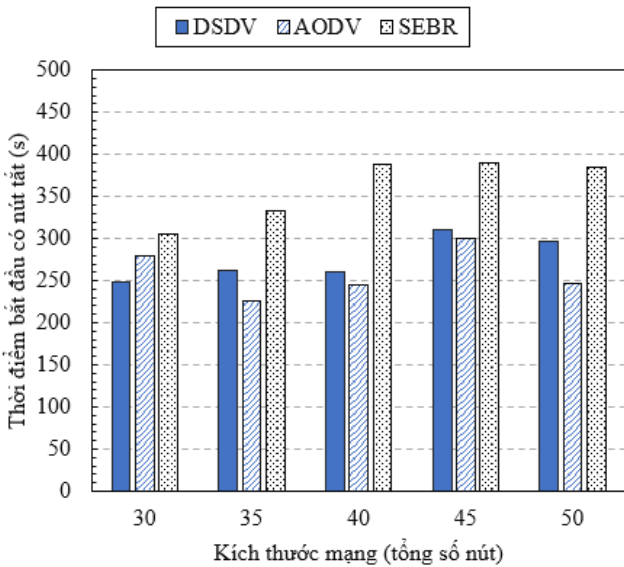


Hình 7. So sánh mức tiêu thụ năng lượng trên tất cả các nút khi sử dụng các thuật toán định tuyến DSDV, AODV và SEBR

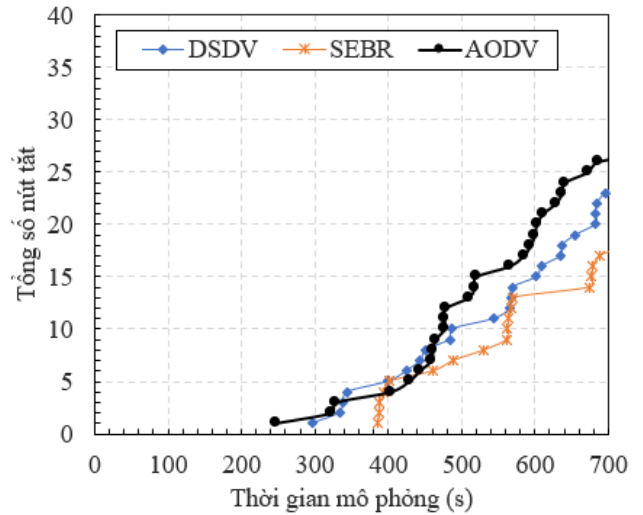
việc năng lượng tiêu thụ tại các nút đã cân bằng hơn. Cụ thể, với các kịch bản mô phỏng 40 và 50 nút thì không còn điểm ngoại lệ nào. Kịch bản 35 nút chỉ còn một nút ngoại lệ, kịch bản 45 nút còn 2 nút ngoại lệ. Tuy nhiên, mức tiêu thụ năng lượng tại các nút ngoại lệ này cũng đã giảm đi rõ ràng. Kết quả này đạt được là do thuật toán SEBR đã xem xét đến năng lượng còn lại tại các nút mỗi khi tính toán, cập nhật lại bảng định tuyến, dẫn đến tập lộ trình truyền dữ liệu trong mạng hạn chế đi qua các nút có năng lượng còn lại thấp hơn.

c) Phân tích thời gian hoạt động của các nút

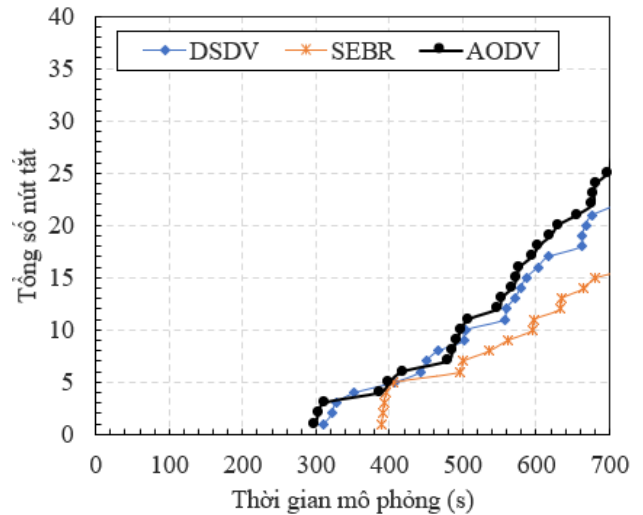
Vì mức sử dụng năng lượng tại các nút của thuật toán SEBR cân bằng hơn so với DSDV và AODV, nên thời gian hoạt động của các nút cũng kéo dài hơn. Kết quả này cho thấy rõ ở Hình 8, chúng tôi so sánh thời điểm bắt đầu có nút tắt do hết năng lượng của các thuật toán DSDV, AODV và SEBR. Ta thấy rằng, trong cả 5 kịch bản mà tổng số nút là 30, 35, 40, 45 và 50, thuật toán SEBR mang lại thời điểm có nút bắt đầu tắt lâu hơn so với các thuật toán DSDV và AODV. Cụ thể, với kịch bản mô phỏng 30 nút, tại thời điểm 248,1 giây bắt đầu có nút tắt nếu sử dụng thuật toán DSDV. Nếu sử dụng AODV, thời điểm bắt đầu có nút tắt là 280,15 giây. Trong khi đó, tại thời điểm 306,1 giây mới bắt đầu có nút tắt nếu sử dụng thuật toán SEBR. Như vậy, thuật toán SEBR đã kéo dài thời gian hoạt động ổn định của hệ thống mạng thêm 26 giây và 58 giây nếu so với các thuật toán AODV và DSDV theo thứ tự tương ứng. Trường hợp thời gian được kéo dài nhiều nhất là kịch bản 40 nút, thời gian kéo dài trong kịch bản này lên đến 128.2 giây khi so với thuật toán DSDV.



Hình 8. So sánh thời điểm bắt đầu có nút tắt do hết năng lượng của các thuật toán DSDV, AODV và SEBR



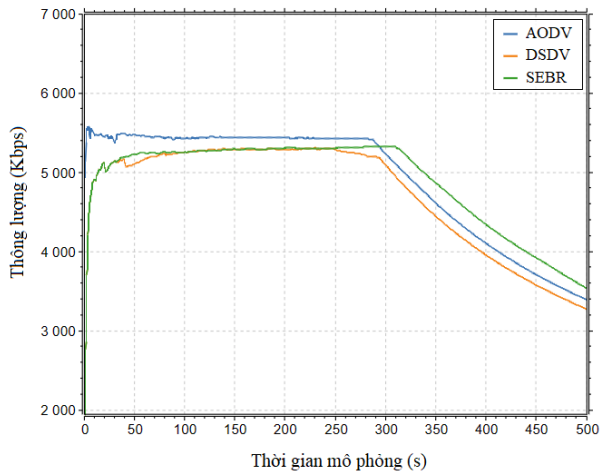
Hình 9. So sánh tổng số nút tắt theo thời gian mô phỏng của thuật toán DSDV, AODV và SEBR khi tổng số nút là 50



Hình 10. So sánh tổng số nút tắt theo thời gian mô phỏng của thuật toán DSDV, AODV và SEBR trong trường hợp tổng số nút là 45

Tiếp theo, chúng tôi khảo sát chi tiết về tổng số nút bị tắt do hết năng lượng theo thời gian mô phỏng. Các kết quả thu được như ở Hình 9 và Hình 10, tương ứng với các kịch bản mà tổng số nút là 45 và 50. Các kết quả thu được đã cho thấy rằng, với cả ba thuật toán SEBR, DSDV và AODV, tổng số nút bị tắt tăng dần theo thời gian mô phỏng. Điều này là hiển nhiên vì các nút hoạt động càng lâu thì năng lượng càng cạn kiệt. Tuy nhiên, nếu so sánh giữa các thuật toán SEBR, DSDV và AODV, trường hợp của thuật toán SEBR luôn có tổng số nút bị tắt nhỏ hơn hoặc bằng hai thuật toán còn lại.

Từ các kết quả khảo sát về việc tiêu thụ năng lượng tại các nút ở trên, chúng tôi có nhận xét rằng, thuật toán SEBR



Hình 11. So sánh thông lượng khi sử dụng thuật toán định tuyến AODV, DSDV và SEBR

hoạt động tốt hơn các thuật toán DSDV và AODV nếu xét về mặt hiệu quả sử dụng năng lượng. Cụ thể là cân bằng mức năng lượng được sử dụng tại các nút, kéo dài thời gian hoạt động ổn định của hệ thống mạng, giảm số nút bị tắt do hết năng lượng.

d) Phân tích thông lượng

Trong phần cuối cùng này, chúng tôi khảo sát thông lượng trung bình nhận được tại BS qua thời gian mô phỏng. Với độ đo này, thuật toán SEBR cũng thực thi hiệu quả hơn so với các thuật toán DSDV và AODV. Nhận định này thể hiện rõ bằng kết quả mô phỏng ở Hình 11. Trong đó, chúng tôi đo thông lượng trung bình nhận được tại BS theo thời gian mô phỏng. Các đồ thị trên Hình 11 cho thấy rằng, khi thời gian mô phỏng nhỏ hơn 250 giây, thông lượng trung bình của cả hai thuật toán DSDV và SEBR gần như là giống nhau, thuật toán AODV mang lại thông lượng cao nhất. Nguyên nhân là vì trong khoảng thời gian này, tình trạng một số nút bị tắt do hết năng lượng chưa xảy ra. Khi thời gian mô phỏng lớn hơn 250 giây, thông lượng của các thuật toán DSDV và AODV bắt đầu giảm. Nguyên nhân là do một số nút đã bị tắt, ảnh hưởng đến việc truyền dữ liệu trong toàn mạng. Đến thời điểm 321 giây, thông lượng của cả ba thuật toán DSDV, AODV và SEBR đều bắt đầu giảm, cũng với lý do một số nút đã bị tắt do hết năng lượng. Tuy nhiên, so sánh giữa ba thuật toán, thuật toán SEBR mang lại thông lượng trung bình cao nhất.

IV. KẾT LUẬN

Việc sử dụng năng lượng sao cho hiệu quả để kéo dài thời gian hoạt động của các nút cảm biến trong mạng WSN là điều hết sức cần thiết, đặc biệt là đối với các trường hợp mà WSN được triển khai cho các ứng dụng với tải lưu lượng lớn, yêu cầu độ tin cậy cao. Bài báo đã đề xuất một

thuật toán định tuyến có xét đến mức tiêu thụ năng lượng giữa các nút cảm biến, được đặt tên là SEBR. Mục tiêu của thuật toán SEBR là cân bằng mức tiêu thụ năng lượng, giảm thiểu số nút phải tiêu thụ nhiều năng lượng nhằm tăng thời gian hoạt động của chúng. Phương pháp của chúng tôi là xây dựng một hàm trọng số cho các kết nối không dây có chứa các tham số về năng lượng còn lại tại các nút. Sau đó, sử dụng cơ chế định tuyến tập trung dựa trên kiến trúc mạng điều khiển bằng phần mềm (SDN) để tìm lộ trình có trọng số tốt nhất để truyền dữ liệu. Kết quả mô phỏng trên OMNeT++ cho thấy rằng, thuật toán được đề xuất cho phép tăng thời gian hoạt động của các nút, tăng thông lượng mạng so với các thuật toán DSDV và AODV.

Trong xu thế phát triển của công nghệ mạng không dây hiện nay, WSN tích hợp với mạng 5G (5G-based WSN) và điện toán đám mây (Cloud-based WSN) đang làm một xu hướng phát triển của công nghệ mạng WSN. Việc phát triển thuật toán cho các mô hình WSN này là điều cần thiết. Đây cũng là hướng phát triển tiếp theo của công trình nghiên cứu này.

TÀI LIỆU THAM KHẢO

- [1] H. Mostafaei and M. Menth, "Software-defined wireless sensor networks: A survey," *Journal of Network and Computer Applications*, vol. 119, pp. 42–56, 06 2018.
- [2] K. M. Modieginiane, B. B. Letswamotse, R. Malekian, and A. M. Abu-Mahfouz, "Software defined wireless sensor networks application opportunities for efficient network management: A survey," *Computers Electrical Engineering*, vol. 66, pp. 274–287, 2018.
- [3] B. B. Letswamotse, R. Malekian, C.-Y. Chen, and K. M. Modieginiane, "Software defined wireless sensor networks (sdwsn): A review on efficient resources, applications and technologies," *Journal of Internet Technology*, vol. 19, pp. 1303–1313, 2018. [Online]. Available: <https://api.semanticscholar.org/CorpusID:69822630>
- [4] S. Wang, A. Hawbani, X. Wang, O. Busaileh, and L. Ping, "Heuristic routing for software defined wireless sensor network," in *2018 World Symposium on Digital Intelligence for Systems and Machines (DISA)*, 2018, pp. 39–44.
- [5] S. Manisekaran and R. Venkatesan, "An analysis of software-defined routing approach for wireless sensor networks," *Computers Electrical Engineering*, vol. 56, 07 2016.
- [6] R. Huang, Y. Dong, G. Bao, Y. Liu, M. Wei, J. Lu, and Y. Huo, "A new topology control algorithm in software defined wireless rechargeable sensor networks," *IEEE Access*, vol. 9, pp. 101 003–101 012, 2021.
- [7] Z. Geng, W. Xia, W. Cao, T. Wu, F. Yan, L. Shen, and J. Pang, "An energy-efficient hierarchical topology control algorithm in software-defined wireless sensor network," in *2021 13th International Conference on Wireless Communications and Signal Processing (WCSP)*, 2021, pp. 1–6.
- [8] Q. Liu, L. Cheng, R. Alves, T. Ozcelebi, F. Kuipers, G. Xu, J. Lukkien, and S. Chen, "Cluster-based flow control in hybrid software-defined wireless sensor networks," *Computer Networks*, vol. 187, p. 107788, 2021.
- [9] B. Cao, S. Deng, H. Qin, and Y. Tan, "A novel method of mobility-based clustering protocol in software defined sensor network," *EURASIP Journal on Wireless Communications and Networking*, vol. 2021, 04 2021.

- [10] M. Alenazi and S. Monti, "Software-defined network-based energy-aware routing method for wireless sensor networks in industry 4.0," *Applied Sciences*, vol. 12, p. 10073, 10 2022.
- [11] V. Duong Thi Thuy and L. Binh, "Irsml: An intelligent routing algorithm based on machine learning in software defined wireless networking," *ETRI Journal*, vol. 44, 08 2022.
- [12] M. U. Younus, M. K. Khan, and A. R. Bhatti, "Improving the software-defined wireless sensor networks routing performance using reinforcement learning," *IEEE Internet of Things Journal*, vol. 9, no. 5, pp. 3495–3508, 2022.
- [13] S. Roy, R. Dutta, N. Ghosh, and P. Ghosh, "Adaptive motif-based topology control in mobile software defined wireless sensor networks," in *2021 IEEE 18th Annual Consumer Communications & Networking Conference (CCNC)*. IEEE Press, 2021, p. 1–6. [Online]. Available: <https://doi.org/10.1109/CCNC49032.2021.9369601>
- [14] Perkins, et. al., *Ad hoc On-Demand Distance Vector (AODV) Routing*. Network Working Group, 2003. [Online]. Available: <https://www.rfc-editor.org/rfc/rfc3561.html>
- [15] András Varga and OpenSim Ltd., *OMNeT++ Simulation Manual - Version 6.x*, 2022. [Online]. Available: <https://http://www.omnetpp.org/documentation/>
- [16] A. Virdis and M. Kirsche, *Recent advances in network simulation - The OMNeT++ environment and its ecosystem*. Springer Nature Switzerland AG, 2019. [Online]. Available: <https://http://www.omnetpp.org/documentation/>



Đỗ Thành Công tốt nghiệp Cử nhân ngành CNTT tại Trường Đại học Công nghệ thông tin - Đại học Quốc gia Thành phố Hồ Chí Minh năm 2010, tốt nghiệp Thạc sĩ ngành Khoa học máy tính tại Học viện Kỹ thuật Quân sự năm 2013. Hiện là giảng viên khoa CNTT, trường Đại học Kinh doanh và Công nghệ Hà Nội.

Lĩnh vực nghiên cứu: ứng dụng AI trong xử lý và nhận diện hình ảnh, công nghệ mạng cảm biến không dây.

Email: congdt@hubt.edu.vn



Nguyễn Đỗ Hoàng Giang hiện là học sinh chuyên toán niên khóa 2021 – 2024 tại Trường Trung học Phổ thông chuyên Khoa học Tự nhiên, Trường Đại học Khoa học Tự nhiên, Đại học Quốc gia Hà Nội.

Lĩnh vực nghiên cứu: trí tuệ nhân tạo, máy tính nhúng và mạng máy tính.

Email: giangnguyen.061106@gmail.com

SƠ LƯỢC VỀ TÁC GIẢ



Lê Đức Huy tốt nghiệp Cử nhân ngành CNTT tại Trường Đại học Kinh doanh và Công nghệ Hà Nội, năm 2012. Tốt nghiệp Thạc sĩ ngành Khoa học máy tính tại Trường Đại học Công nghệ thông tin và Truyền thông Thái Nguyên, năm 2015. Hiện là nghiên cứu sinh tại Học viện Khoa học và Công nghệ (GUST), Viện Hàm lâm

Khoa học và Công nghệ Việt Nam. Hiện công tác tại Khoa CNTT, Trường Đại học Kinh doanh và Công nghệ Hà Nội.

Lĩnh vực nghiên cứu: các giao thức điều khiển trong mạng máy tính và Viễn thông, an ninh mạng.

Email: huyld@hubt.edu.vn



Lê Hữu Bình tốt nghiệp Kỹ sư ngành Điện tử - Viễn thông tại Trường Đại học Bách Khoa, Đại học Đà Nẵng năm 2001, tốt nghiệp Thạc sĩ ngành Khoa học máy tính tại Trường Đại học Khoa học, Đại học Huế, năm 2007, bảo vệ thành công luận án Tiến sĩ năm 2020 tại Học viện Khoa học và Công nghệ (GUST), Viện Hàm lâm Khoa

học và Công nghệ Việt Nam, ngành Máy tính, chuyên ngành Hệ thống thông tin. Hiện nay, ông đang công tác tại Khoa CNTT, Trường Đại học Khoa học, Đại học Huế.

Lĩnh vực nghiên cứu: công nghệ mạng không dây thế hệ mới, SDN (Software Defined Networking), mô phỏng và thiết kế mạng.

Email: lhbinh@hueuni.edu.vn