

Điều khiển mờ hỗ trợ giao thức CoAP nhằm chống tắc nghẽn mạng Internet vạn vật

Lê Thị Thùy Dương¹, Hoàng Đăng Hải^{2*}, Phạm Thiều Nga¹

¹Khoa Công nghệ thông tin, Trường Đại học Xây dựng Hà Nội, số 5 Đường Giải Phóng, Hà Nội;

²Học viện Công nghệ Bưu chính Viễn thông, km10, Đường Nguyễn Trãi, Hà Đông, Hà Nội.

*Email: haihd@ptit.edu.vn

Nhận bài: 07/3/2023; Hoàn thiện: 27/4/2023; Chấp nhận đăng: 11/5/2023; Xuất bản: 25/6/2023.

DOI: <https://doi.org/10.54939/1859-1043.j.mst.88.2023.22-33>

TÓM TẮT

Giao thức CoAP (Constrained Application Protocol) và một số cải tiến của nó vẫn còn hạn chế về khả năng phát hiện sớm tắc nghẽn và điều chỉnh tốc độ phát phù hợp với trạng thái biến động của mạng Internet vạn vật. Bài báo này đề xuất giải pháp thực thi một cơ chế điều khiển mờ với việc lựa chọn tham số đầu vào, đầu ra thích hợp cho chống tắc nghẽn mạng. Các tham số được đánh giá bằng công cụ mô phỏng. Các kết quả thí nghiệm mô phỏng chỉ ra việc lựa chọn tham số phù hợp lý thuyết, cho phép hệ điều khiển mờ đạt các chỉ số hiệu năng cao hơn so với cơ chế CoAP chuẩn.

Từ khoá: Giao thức CoAP; Điều khiển tắc nghẽn; Điều khiển mờ; Mạng IoT.

1. MỞ ĐẦU

CoAP (Constrained Application Protocol) là giao thức truyền tin phổ biến nhất trong mạng Internet vạn vật (IoT-Internet of Things) và đã được tổ chức IETF (Internet Engineering Task Force) chuẩn hóa với RFC 7252 [1]. CoAP là một giao thức ở tầng ứng dụng, hoạt động ở trên tầng giao thức UDP (User Datagram Protocol). Do UDP truyền tin không tin cậy, CoAP bổ sung cơ chế truyền dữ liệu có tin cậy tương tự như tất cả các phiên bản TCP (Transmission Control Protocol) với việc sử dụng phản hồi ACK (acknowledgment). Phản hồi này giúp CoAP nhận biết có mất gói xảy ra để phát lại gói tin. CoAP sử dụng một cơ chế điều khiển tắc nghẽn đơn giản dựa vào giá trị RTO (Retransmission Timeout) để phát lại. Mất gói được coi là dấu hiệu của tắc nghẽn. Khi đó, CoAP kích hoạt cơ chế phát lại gói tin đã mất do tắc nghẽn bằng một thuật toán rút lui theo hàm mũ nhị phân (BEB-Binary Exponential Backoff). Theo đó, RTO tăng gấp đôi, nghĩa là tốc độ phát lại sẽ giảm $\frac{1}{2}$ cho mỗi lần gửi lại. Cơ chế này có tác dụng làm giảm số gói tin gửi thêm vào mạng, do đó làm giảm nguy cơ tắc nghẽn mạng. Tuy nhiên, điều này dẫn đến hiệu quả truyền tin thấp trong điều kiện mạng biến động nhiều như mạng IoT. Tại các thời điểm có mất gói, RTO có thể trở nên rất lớn khiến cho CoAP không thể truyền ngay các gói tin tiếp theo mặc dù thời điểm kế tiếp không còn tắc nghẽn. Do RTO lớn, thời gian chờ phát tiếp kéo dài, CoAP rơi vào trạng thái nghỉ kéo dài không cần thiết. Vì vậy, CoAP kém hiệu quả khi truyền tin theo chùm gói như đã chỉ ra trong [2]. Những hạn chế của CoAP đã được chỉ ra trong [3-5].

Hầu hết các nghiên cứu cải tiến CoAP đều tập trung vào sửa đổi cách tính RTO cho phù hợp với điều kiện mạng. Các tác giả [6] đã chỉ ra những điểm tồn tại của các nghiên cứu trước. Một trọng tâm được quan tâm cần hiệu chỉnh RTO theo biến thiên của RTT (Round Trip Time). Việc cập nhật lại RTO giúp hạn chế tần suất phát lại cho phù hợp với biến động mạng IoT. Các cơ chế cải tiến điển hình là CoCoA, CoCoA+, BDP-CoAP, pCoCoA, iCoCoA [6-8]. CoCoA thay thế RTO cố định của CoAP bằng một biến RTO và một hàm rút lui với hệ số động VBF (Variable Backoff Factor) thay cho BEB. CoCoA+ cải tiến CoCoA bằng cách sử dụng hệ số nhân theo xác suất PBF (Probability Backoff Factor) để tính lại RTO. pCoAP sử dụng hệ số động để tính lại RTO dựa vào độ sai lệch tối đa cho phép của RTO, qua đó giảm thiểu được ảnh hưởng gây ra bởi RTT biến thiên. BDP-CoAP đưa ra cách tính RTO dựa vào băng thông tắc nghẽn đo được. iCoCoA sử dụng học máy tăng cường để tính lại RTO [7]. Bài báo [8] sử dụng hệ logic mờ để

hiệu chỉnh RTO theo xu hướng tăng hay giảm của RTT. Có thể nhận thấy, CoAP và các cải tiến nêu trên chỉ tập trung vào điều chỉnh tốc độ phát lại dựa vào việc tính toán, cập nhật RTO chứ không điều chỉnh tốc độ phát. Bốn cơ chế điều khiển tốc độ phát CoAP được phân tích trong [6] gồm RTT-CoAP, CoAP-R, BDP-CoAP và CoAP-SC. RTT-CoAP dựa vào biến thiên độ trễ qua RTT để điều khiển tốc độ phát theo bốn vùng lưu lượng: tắc nghẽn thấp, bình thường, trung bình và biến thiên lớn. Tùy vào lưu lượng thực tế, tốc độ phát được điều chỉnh theo cách tăng cộng – giảm cộng. CoAP-R điều chỉnh tốc độ theo cách tăng nhân – giảm nhân dựa vào thông tin cây định tuyến và băng thông cấp phát cho từng nút mạng IoT. BDP-CoAP tính toán băng thông cổ chai và biến thiên RTT để tính tốc độ phát với hệ số điều chỉnh tốc độ được đặt cứng là 0,6 hoặc 0,2 tùy theo tần số phát lại nhỏ hơn hoặc lớn hơn 20%. CoAP-SC dựa vào điều khiển luồng tin và số thứ tự gói tin để điều chỉnh tốc độ phát. Những cải tiến trên đã chỉ ra khả năng chống tắc nghẽn tốt hơn cho CoAP. Tuy nhiên, hạn chế của các cơ chế này là: chỉ khi mất gói mới biết tắc nghẽn, cơ chế điều khiển cứng theo vùng lưu lượng, đòi hỏi có thông tin chi tiết về từng nút trên mạng.

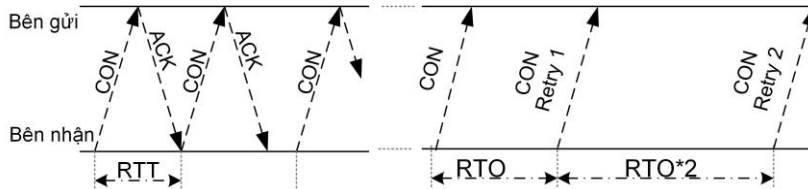
Có thể nhận thấy, CoAP và các cải tiến nó đều không có khả năng phát hiện sớm tắc nghẽn để có điều khiển tốc độ phát linh hoạt, kịp thời. Phát hiện sớm tắc nghẽn và điều khiển tốc độ phát chống tắc nghẽn phụ thuộc vào các tham số RTT, tải lưu lượng, hiện trạng mạng, tỷ lệ mất gói. Tuy nhiên, các tham số này đều biến thiên nhanh và thường không rõ ràng, không chắc chắn. Một hệ điều khiển mờ sẽ là một giải pháp hợp lý [9]. Khác với các nghiên cứu trước, hệ điều khiển mờ không tập trung vào cách tính RTO để điều chỉnh tốc độ phát lại mà điều khiển tốc độ phát tùy vào việc phát hiện sớm tắc nghẽn. Lựa chọn các tham số đầu vào và đầu ra cho hệ điều khiển mờ đóng vai trò quyết định. Trong bài báo này, chúng tôi đề xuất giải pháp thực thi một hệ điều khiển mờ chống tắc nghẽn cho CoAP với các khả năng lựa chọn các tham số đầu vào phù hợp nhằm sớm phát hiện tắc nghẽn và điều kiện tốc độ phát chống tắc nghẽn.

2. CƠ CHẾ ĐIỀU KHIỂN MỜ CHỐNG TẮC NGHẼN CHO COAP

2.1. Tổng quan về CoAP và các phiên bản cải tiến

Mặc dù có nhiều loại giao thức tầng ứng dụng khác cho mạng IoT, song CoAP là phổ biến nhất cho mạng IoT [3-6]. Chế độ truyền tin cậy của CoAP thực hiện với các gói tin CON (confirmable) và các gói tin phản hồi ACK (acknowledgment).

CoAP sử dụng cơ chế điều khiển tắc nghẽn theo kiểu dừng và đi tiếp (Stop-and-Go). Bên gửi dừng phát để chờ phản hồi ACK từ bên nhận và chỉ phát tiếp sau khi đã nhận được ACK cho gói tin đã phát trước đó. Cho mỗi lần phát, CoAP đặt một giá trị RTO. Nếu quá RTO mà bên gửi vẫn chưa nhận được ACK, gói tin đã gửi được coi là bị mất. Khi đó, bên gửi sẽ phải phát lại gói tin. Mỗi lần phát lại, RTO được tăng gấp đôi, nghĩa là giảm theo hàm mũ nhị phân (BEB). Hình 1 mô tả quá trình trao đổi các gói tin CON và ACK cũng như các lần phát lại khi mất gói tin.



Hình 1. Trao đổi gói tin và các lần phát lại của CoAP.

Các điểm tồn tại của CoAP đã được chỉ ra trong [3-5, 9]. Trong số các đề xuất cải tiến CoAP, mới chỉ có hai phiên bản được chuẩn hóa là CoCoA và CoCoA+ [6, 7]. Cơ chế CoCoA đưa ra cách tính RTO dựa vào đo lường liên tục RTT như sau:

$$RTTVAR_x = (1 - \beta)RTTVAR_x + \beta |RTT_x - RTT_{x,new}| \tag{1}$$

$$RTT_x = (1 - \alpha)RTT_x + \alpha RTT_{x,new} \tag{2}$$

$$RTO_x = RTT_x + 4 * RTTVAR_x \tag{3}$$

$$RTO_{overall} = 0,5 * RTO_x + 0,5 * RTO_{overall} \quad (4)$$

Trong đó, $RTO_{overall}$ là giá trị trung bình ước tính của RTT, x là ký hiệu cho độ điều chỉnh *mạnh* hay *yếu*, new biểu thị giá trị đo được ở chu kỳ hiện thời, $overall$ biểu thị giá trị ước tính tổng thể, α và β là các hệ số tương tự TCP. Giá trị *mạnh* khi bên gửi nhận được ACK ngay cho lần phát đầu tiên. Giá trị *yếu* khi ACK nhận được sau ít nhất một lần phát lại. Vấn đề phát sinh là: 1) RTO lớn sẽ dẫn đến thời gian chờ phát lâu; 2) RTO nhỏ sẽ khiến cho các lần phát lại diễn ra liên tiếp chỉ trong một khoảng thời gian ngắn, dẫn đến tăng thêm tắc nghẽn mạng. Do vậy, một hệ số backoff biến đổi VBF (Variable Backoff Factor) được bổ sung như sau:

$$VBF(RTO_{init}) = \begin{cases} 3 & RTO_{init} < 1s \\ 2 & 1 \leq RTO_{init} \leq 3s \\ 1,3 & RTO_{init} > 3s \end{cases} \quad (5)$$

$$RTO_{backoff} = RTO_{init} * VBF \quad (6)$$

Trong đó, RTO_{init} là giá trị ban đầu, $RTO_{backoff}$ là hiệu chỉnh trong chu kỳ rút lui (Backoff).

Cơ chế CoCoA+ cải tiến CoCoA bằng cách thay đổi cách tính $RTO_{overall}$ như sau:

$$RTO_{overall} = 0,5 * RTO_{strong} + 0,5 * RTO_{overall} \quad (7)$$

$$RTO_{overall} = 0,25 * RTO_{weak} + 0,5 * RTO_{overall} \quad (8)$$

Trong đó, RTO_{strong} và RTO_{weak} là giá trị ước lượng *mạnh* và *yếu* tương ứng cho RTO. Ngoài ra, CoCoA+ thay đổi hệ số VBF như sau:

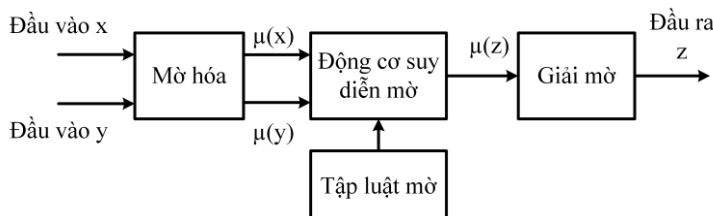
$$VBF = \begin{cases} 2,5 & RTO_{init} < 1s \\ 2 & 1 \leq RTO_{init} < 3s \\ 1,5 & RTO_{init} \geq 3s \end{cases} \quad (9)$$

Hệ số VBF được hiệu chỉnh theo RTO_{init} . Theo cách trên, CoAP và các phiên bản cải tiến chủ yếu dựa vào mật gói để hiệu chỉnh tốc độ phát lại bằng $1/RTO_{overall}$ với giá trị $RTO_{overall}$ được cập nhật mới sau mỗi lần phát lại. $RTO_{overall}$ càng lớn thì tốc độ phát lại càng giảm để giảm tắc nghẽn mạng. Các công thức tính toán RTT và RTO nêu trên đều chỉ dựa vào thực nghiệm. Rất khó để đưa ra một công thức tính toán chính xác. Các tham số đều ở dạng mờ, không rõ ràng, không chắc chắn. Một hệ điều khiển mờ sẽ phát huy được tác dụng trong trường hợp này.

2.2. Hệ điều khiển mờ và lựa chọn đầu vào, đầu ra

2.2.1. Các thành phần hệ điều khiển mờ

Một hệ điều khiển mờ (FCS- Fuzzy Control System) có *bốn* thành phần chính: mờ hóa, tập luật mờ, động cơ suy diễn mờ và phần giải mờ [10, 11]. Mờ hóa thực hiện biến đổi các giá trị rõ từ đầu vào rõ thành tập các giá trị mờ. Cơ sở luật mờ là tập các luật IF-THEN, xác định mối quan hệ mờ giữa đầu vào – đầu ra. Động cơ suy diễn mờ dựa vào các luật mờ để suy diễn đầu ra. Giải mờ tạo các giá trị đầu ra rõ. Hệ FCS có thể có nhiều đầu vào và nhiều đầu ra, song hệ phổ biến thường chỉ gồm vài đầu vào và một đầu ra để việc tính toán không phức tạp. Để giữ cho cơ chế điều khiển CoAP đơn giản, có thể chọn hệ FCS với hai đầu vào và một đầu ra như trên hình 2.



Hình 2. Các thành phần của hệ điều khiển mờ.

Các hàm $\mu(x)$, $\mu(y)$ và $\mu(z)$ biểu diễn hàm thuộc cho đầu vào x , đầu vào y và đầu ra z .

2.2.2. Lựa chọn đầu vào cho hệ điều khiển mờ

Như đã nêu ở trên, việc lựa chọn các tham số đầu vào và đầu ra đóng vai trò quyết định đối với hệ FCS. Các tham số được sử dụng để mô tả trạng thái mạng gồm: tốc độ phát gói, RTT, tải lưu lượng, băng thông, thông lượng. Từ các tham số này có thể dẫn xuất ra các đại lượng khác, ví dụ RTO được tính từ RTT. Các tham số có liên quan nhau, ví dụ tốc độ phát gói tăng dẫn đến tăng tải lưu lượng và thông lượng. Tuy nhiên, tải lưu lượng tăng không nhất thiết là do tốc độ phát, mà còn do gia tăng số lượng kết nối cùng chia sẻ (tranh) đường truyền. Yêu cầu đặt ra đối với hệ FCS là cần nhỏ gọn để phù hợp với mạng IoT. Vì vậy, chọn một đầu ra là tốc độ phát sẽ phù hợp để giảm độ phức tạp tính toán do hệ FCS cần điều khiển tốc độ phát. Các khả năng chọn đầu vào còn lại là: RTT, tải lưu lượng, băng thông, thông lượng. Hệ FCS cần lựa chọn được đầy đủ các tham số phản ánh được tình trạng tắc nghẽn mạng. Mặt khác, các đầu vào càng ít liên quan đến nhau càng tốt như đã chỉ ra trong [10, 11]. Có ít nhất 4 khả năng cho đầu vào là RTT, tải lưu lượng, băng thông và thông lượng. Lựa chọn một đầu vào là đơn giản nhất, song rõ ràng không thể hiện đủ trạng thái tắc nghẽn. Do vậy, cần chọn ít nhất 2 đầu vào từ các đại lượng trên.

RTT là tham số quan trọng nhất thể hiện tắc nghẽn mạng. Khi có nhiều gói tin phát đi, RTT sẽ tăng nhanh chóng và có thể rất lớn. Điều này phản ánh nguy cơ tắc nghẽn cao. RTT giảm đồng nghĩa với nguy cơ tắc nghẽn giảm. Do đó, có thể chọn RTT làm một đầu vào cho hệ FCS. Ngoài độ lớn của RTT, chiều biến thiên tăng hay giảm của RTT có ý nghĩa quan trọng cho phát hiện sớm tắc nghẽn. Việc đo lường RTT được thực hiện tại các thời điểm lấy mẫu rời rạc $k-1, k$. Chu kỳ lấy mẫu $T(k-1, k)$ được định nghĩa là khoảng thời gian giữa hai gói tin ACK liên tiếp. Gọi $RTT(k-1)$ và $RTT(k)$ là các giá trị RTT đo được tại các thời điểm $k-1$ và k , độ biến thiên tương đối của RTT tại k sẽ là $|RTT(k) - RTT(k-1)|$. Đặt RTT_{min} là giá trị nhỏ nhất của RTT tới thời điểm hiện tại, độ biến thiên tuyệt đối tại k của RTT sẽ là $RTT(k) - RTT_{min}(k)$. Có thể thấy, dùng độ biến thiên tuyệt đối sẽ tốt hơn vì không phải duy trì các giá trị đo RTT liên tiếp để xác định chiều hướng tăng hay giảm. Độ biến thiên tuyệt đối RTT càng lớn thì nguy cơ tắc nghẽn càng cao và ngược lại, độ biến thiên tuyệt đối RTT càng nhỏ thì nguy cơ tắc nghẽn càng thấp.

Vì lý do trên, có thể chọn độ biến thiên tuyệt đối chuẩn hóa của RTT làm đầu vào thứ nhất (đầu vào x) cho hệ FCS như đã chỉ ra trong [9] theo công thức sau:

$$x = \frac{RTT(k) - RTT_{min}(k)}{RTT_{max}(k) - RTT_{min}(k)} \quad (10)$$

Trong đó, $RTT_{max}(k)$ là giá trị lớn nhất của RTT tới thời điểm k hiện tại. Giá trị x đo được tại thời điểm k phản ánh độ lớn và chiều tăng hay giảm của RTT, và có $x \in [0, 1]$.

Vấn đề đặt ra là xác định các tham số của công thức (10). $RTT_{min}(k)$ có thể xác định chính xác sau khi gửi gói tin đầu tiên khi bắt đầu kết nối. Sau đó, giá trị nhỏ nhất RTT_{min} duy trì trong suốt thời gian kết nối. Do vậy, RTT_{min} không cần phải cập nhật sau mỗi chu kỳ k . Vì vậy:

$$x = \frac{RTT(k) - RTT_{min}}{RTT_{max}(k) - RTT_{min}} \quad (11)$$

Giá trị $RTT_{max}(k)$ có thể liên tục cập nhật theo các chu kỳ lấy mẫu bằng công thức:

$$RTT_{max}(k) = \max(RTT(k), RTT_{max}(k)) \quad (12)$$

Đối với $RTT(k)$, ta có thể: 1) tính trực tiếp RTT của gói tin hoặc 2) ước lượng $RTT(k)$ trung bình theo phương pháp EWMA (Exponential Weighted Moving Average) [12]. Theo cách thứ hai, bên gửi ước lượng $RTT(k)$ trung bình theo công thức:

$$RTT(k) = 0.75 \times RTT(k-1) + 0.25 \times RTT_m(k) \quad (13)$$

Trong đó, $RTT_m(k)$ là giá trị RTT thực tế đo được tại thời điểm k . Công thức (13) giúp làm mịn kết quả đo. Tuy nhiên, cách này lại làm giảm sự biến thiên tức thời của RTT. Do vậy, cách đo trực tiếp RTT phù hợp hơn cho hệ FCS vì vẫn giữ được biến thiên tức thời để điều khiển.

Như vậy, công thức (11) có thể chọn làm đầu vào x cho hệ FCS. Khác với [9], không cần cập

nhập RTT_{min} để tính x và giá trị RTT tức thời được sử dụng thay vì dùng công thức (13).

Đối với đầu vào y của hệ FCS, sẽ chọn từ các tham số còn lại. Bài báo đề xuất ba cách chọn: 1) tỷ lệ tải lưu lượng, 2) tỷ lệ gói tin đến đích, và 3) tỷ lệ thông lượng so với băng thông tắc nghẽn.

Tải lưu lượng được định nghĩa là tổng số gói tin đang gửi cho tới thời điểm hiện tại. Tỷ lệ tải lưu lượng được định nghĩa theo công thức sau:

$$y = \frac{L(k) - L_{min}(k)}{BDP(k) - L_{min}(k)} \quad (14)$$

Trong đó, $L(k)$ là tải lưu lượng thực tế đo tại thời điểm k , $L_{min}(k)$ là tải lưu lượng nhỏ nhất và $BDP(k)$ là tải lưu lượng lớn nhất được phép tại thời điểm k . $BDP(k)$ chính là tích số của băng thông tắc nghẽn và RTT (BDP - Bandwidth Delay Product), được tính theo [13] như sau:

$$BDP(k) = BW_{max}(k) \times RTT(k) \quad (15)$$

Trong đó, $BW_{max}(k)$ là băng thông tắc nghẽn tối đa ở thời điểm k , là băng thông hẹp nhất khi truyền một luồng gói tin từ bên gửi tới bên nhận [13], nghĩa là tốc độ phát lớn nhất được phép ở trạng thái mạng hiện tại. $BDP(k)$ thực chất là số gói tin tối đa được phép phát tại thời điểm k tương ứng với RTT và băng thông tắc nghẽn ở thời điểm hiện tại. Theo (14), $L(k)$ cần được giới hạn bởi số gói tin tối đa được phép. Giá trị y gần tới 1 khi $L(k)$ gần tới $BDP(k)$, phản ánh thực trạng tắc nghẽn cao. Giá trị y gần tới 0 khi $L(k)$ gần tới $L_{min}(k)$, phản ánh không tắc nghẽn.

Nếu chọn đầu vào y là tỷ lệ gói tin đến đích, tỷ lệ này được định nghĩa theo công thức sau:

$$y = \frac{Q(k) - A(k)}{Q(k)} \quad (16)$$

Trong đó, $Q(k)$ và $A(k)$ là số gói tin đã phát đi và nhận được ACK tới thời điểm k , tương ứng. Có thể thấy, $Q(k)$ thực chất là tải lưu lượng ở thời điểm k . Hiệu số $Q(k) - A(k)$ phản ánh số gói tin đã phát đi song vẫn chưa được xử lý ở thời điểm k . Giá trị (16) gần tới 1 khi $A(k)$ gần tới 0 (khi hầu hết gói tin không tới đích) và gần tới 0 khi $A(k)$ gần tới $Q(k)$ (khi gói tin được xử lý ngay). Tỷ số gần 1 phản ánh tắc nghẽn cao, còn tỷ số gần 0 là không có tắc nghẽn. Tuy nhiên, cách chọn này có hạn chế là $A(k)$ giảm rất nhanh tới 0 khi xuất hiện tắc nghẽn và $A(k)$ thường xấp xỉ $Q(k)$ khi không có tắc nghẽn. Tỷ lệ theo công thức (16) có sự tăng giảm đột biến giữa các giá trị 0 và 1.

Nếu chọn đầu vào y là tỷ lệ thông lượng so với băng thông tắc nghẽn, ta có:

$$y = \frac{\min(BW_{max}(k), \theta(k))}{BW_{max}(k)} \quad (17)$$

Trong đó, $\theta(k)$ là thông lượng tại thời điểm k . Hàm \min nhằm duy trì thông lượng không được vượt quá băng thông tắc nghẽn tối đa và $y \in [0, 1]$. Băng thông tắc nghẽn thực chất được coi như tốc độ phát lớn nhất được phép ở trạng thái mạng hiện tại. Công thức (17) khác với [9] ở chỗ sử dụng thông lượng thay vì dùng tốc độ chuyển gói tin và có cập nhật $BW_{max}(k)$ theo chu kỳ.

Như đã trình bày ở trên, cần lựa chọn đầy đủ các tham số quan trọng để phản ánh được tình trạng tắc nghẽn mạng và có thể chọn được từ 2 đến 3 đầu vào từ các tham số đã nêu với tiêu chí ít có sự liên quan giữa chúng. Đầu vào thứ nhất (đầu vào x) cho hệ FCS có thể chọn theo công thức (11). Công thức này khác với công thức (10) đã sử dụng trong [9] ở chỗ không cần cập nhật RTT_{min} sau mỗi chu kỳ lấy mẫu, do đó sẽ đơn giản hơn trong tính toán. Đối với các đầu vào khác, có thể thấy công thức (14) và (16) có mối liên quan đến các công thức (11) và (17). Trong công thức (14), ta thấy BDP được tính dựa vào RTT mà RTT đã được sử dụng trong công thức (11) cho đầu vào thứ nhất. Công thức (16) có sự tăng giảm đột biến, không phản ánh được ảnh hưởng của băng thông tắc nghẽn, đồng thời có liên quan đến thông lượng. Do vậy, sử dụng công thức (17) cho đầu vào thứ hai và chỉ sử dụng 2 đầu vào cho hệ FCS là lựa chọn phù hợp.

2.2.3. Lựa chọn đầu ra cho hệ điều khiển mờ

Do hệ FCS chỉ dùng một đầu ra như phân tích ở mục 2.2.1 và tốc độ phát là đại lượng cần điều khiển, có thể đưa ra hai cách lựa chọn đầu ra z của hệ FCS như sau:

- Chọn z là hệ số để điều chỉnh tốc độ bằng $R(k)/z$ với $z \neq 0$. Theo cách này, giá trị z càng lớn (nghĩa là càng tắc nghẽn) thì $R(k)$ càng cần điều chỉnh nhỏ đi và ngược lại. Tuy nhiên, khó xác định cận trên và dưới của z cho hệ FCS.

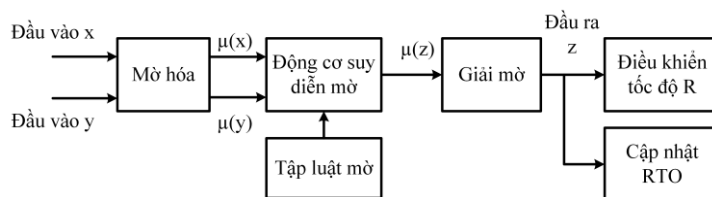
- Chọn z là giá trị biểu thị mức độ tắc nghẽn mạng để từ đó điều khiển tốc độ phát. Cách này đơn giản hơn và có thể điều chỉnh tốc độ phát theo công thức sau:

$$R(k) = R(k - 1) \pm |z| \frac{1}{RTT(k)} \tag{18}$$

Trong đó, $R(k)$ là tốc độ phát ở thời điểm cập nhật k , $R(k-1)$ là tốc độ phát ở thời điểm điều chỉnh $k-1$ trước đó, dấu cộng hoặc trừ là tùy vào dấu của biến đầu ra z . Khi đó có thể chọn giá trị z trong dải từ -1 đến $+1$. Giá trị z càng gần 1 thì mạng càng ít tắc nghẽn, cần điều chỉnh tăng tốc độ phát theo z để tăng hiệu suất truyền tin. Giá trị z càng gần -1 thì mạng càng tắc nghẽn. Cần điều chỉnh giảm tốc độ phát theo z để chống tắc nghẽn. Chu kỳ điều chỉnh tốc độ là một RTT.

2.3. Thiết kế hệ điều khiển mờ chống tắc nghẽn cho CoAP

Hệ điều khiển mờ FCS gồm hai đầu vào và một đầu ra như trên hình 3. Đầu ra z phục vụ cho điều khiển tốc độ R theo công thức (18).



Hình 3. Sơ đồ hệ điều khiển mờ FCS.

Mỗi đầu vào/đầu ra có thể có nhiều biến ngôn ngữ. Số lượng biến ngôn ngữ cho mỗi đầu vào/đầu ra thường là lẻ từ 3 đến 7 như khuyến nghị trong [14] tùy theo độ phức tạp của điều khiển. Để đơn giản cho CoAP, hệ FCS sử dụng 3 tập mờ cho các biến ngôn ngữ cho mỗi đầu vào, cụ thể là các tập mờ: *Small (XS)*, *Medium (XM)*, *Large (XL)* cho đầu vào x , các tập mờ: *Small (YS)*, *Medium (YM)*, *Large (YL)* cho đầu vào y . Cho đầu ra có 5 tập mờ là: *Very low (ZVL)*, *Low (ZL)*, *Medium (ZM)*, *High (ZH)*, *Very high (ZVH)*.

Dạng hình tam giác được sử dụng phổ biến để biểu diễn các hàm thuộc cho đầu vào và đầu ra [10, 11]. Từ cơ sở lý thuyết về hàm thuộc dạng tam giác [10, 11], bài báo chọn tập giá trị biểu diễn cho đầu vào x là [0; 0,1; 0,4; 0,8; 1], cho đầu vào y là [0; 0,25; 0,5; 0,75; 1] và cho đầu vào z là [-1; -0,6; -0,3; 0; 0,3; 0,6; 1]. Các giá trị được chọn căn cứ vào mức độ tác động của các tham số vào tắc nghẽn, có thể tùy chỉnh khi triển khai CoAP trong thực tế.

Từ các giá trị rõ của x và y ở đầu vào, khối mờ hóa xác định ra các độ thuộc tương ứng. Động cơ suy diễn mờ căn cứ vào các tập mờ đầu vào và luật mờ để xác định đầu ra. Do sử dụng 3 biến ngôn ngữ cho mỗi đầu vào, ta có tổng số $3 \times 3 = 9$ luật mờ. Phương pháp max-min được sử dụng cho động cơ suy diễn mờ và phương pháp xấp xỉ trọng tâm (CoG- Center of Gravity) [11, 14] được sử dụng cho giải mờ để chuyển đổi dữ liệu mờ đầu ra thành dữ liệu rõ z theo công thức:

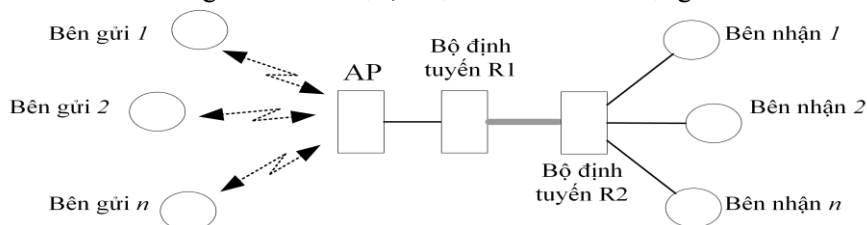
$$z = \frac{\sum_{i=1}^N z_i \times \mu(z_i)}{\sum_{i=1}^N \mu(z_i)} \tag{19}$$

Trong đó, z là đầu ra rõ, N là số tập mờ đầu ra, z_i là trọng tâm của mỗi tập mờ đầu ra, $\mu(z_i)$ là độ thuộc tương ứng. Giá trị rõ ở đầu ra nằm trong khoảng giữa $[-1, 1]$ biểu thị mức độ tắc nghẽn. Từ giá trị z , hệ FCS điều khiển tốc độ phát theo công thức (18).

3. MÔ PHỎNG, ĐÁNH GIÁ KẾT QUẢ

3.1. Công cụ, phương pháp mô phỏng

Bộ công cụ mô phỏng mạng NS3 [15] được dùng để thực hiện mô phỏng. Phần mềm FCS và CoAP được triển khai ở bên gửi và bên nhận, được cài ở các nút mạng như trên sơ đồ hình 4.



Hình 4. Sơ đồ mạng mô phỏng.

Sơ đồ hình 4 là cấu trúc điển hình của mạng IoT với n bên gửi và n bên nhận tin. AP (Access Point) là điểm truy nhập tập trung. Các bộ định tuyến R1 và R2 mô phỏng kết nối Internet và băng thông tắc nghẽn. Cấu hình chuẩn được tạo với: IP chuẩn, WiFi IEEE 802.11 tiêu chuẩn 2 Mbps theo NS3. Dữ liệu CoAP chuẩn [1] gồm: 106 Bytes cho gói tin CON, 49 Bytes cho gói tin ACK với tiêu đề chuẩn. Mỗi mô phỏng chạy thử 30 lần, lấy trung bình và tính toán khoảng tin cậy với phân bố chuẩn. Để đánh giá, bài báo chọn 4 kịch bản mô phỏng (4 nhóm thí nghiệm) sau.

- Kịch bản 1 với $n=10$, liên kết giữa AP và R1 là 5 Mbps với độ trễ 100 ms, liên kết giữa R1 và R2 là 10 Mbps với độ trễ 10 ms. Mục đích của nhóm thí nghiệm này là tạo ra một cấu hình mạng có mức độ tắc nghẽn thấp để kiểm chứng lý thuyết cho các tham số hệ FCS.

- Kịch bản 2 với $n=10$, liên kết giữa AP và R1 là 1 Mbps với độ trễ 240 ms, liên kết giữa R1 và R2 là 10 Mbps với độ trễ 100 ms. So với kịch bản 1, kịch bản này sử dụng băng thông liên kết thấp hơn và độ trễ lớn hơn nhằm tạo ra một cấu hình mạng có mức độ tắc nghẽn cao để kiểm chứng các tham số đầu vào hệ FCS trong tình huống này.

- Kịch bản 3 với $n=10$, liên kết giữa AP và R1 là 1 Mbps với độ trễ 240 ms, liên kết giữa R1 và R2 là 10 Mbps với độ trễ 10 ms. Cấu hình này tạo ra tình trạng mạng có tắc nghẽn thay đổi động để kiểm chứng kết quả biến thiên đầu vào/đầu ra của hệ FCS.

- Kịch bản 4 với $n=10$ gồm: 1 FCS, 1 CoAP để so sánh, 8 CoAP còn lại gửi liên tiếp từ thời điểm 120s. Liên kết AP và R1 là 250 Kbps với độ trễ 50 ms, liên kết R1 và R2 là 10 Kbps với độ trễ 300 ms. Cấu hình này tạo ra tình trạng tắc nghẽn nghiêm trọng (lưu lượng của 8 CoAP khác tăng vọt từ 120 s) nhằm so sánh FCS và CoAP.

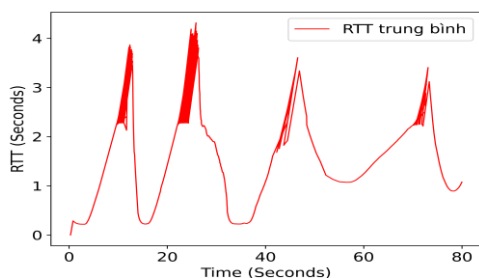
Số nút mạng n có thể thay đổi tùy ý, n tăng thì lưu lượng tăng, tắc nghẽn tăng và ngược lại. Tuy nhiên, do các kịch bản nêu trên đã tạo ra 4 tình huống tắc nghẽn mạng điển hình nên không cần thiết phải thay đổi n để tạo các tình huống tắc nghẽn tương tự.

3.2. Kết quả mô phỏng và bình luận

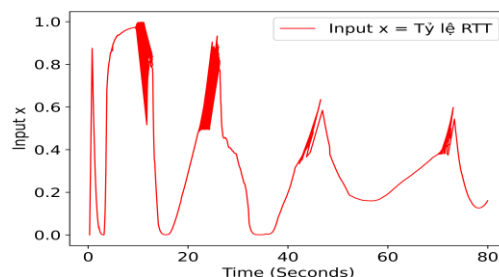
3.2.1. Nhóm thí nghiệm mô phỏng 1

Các hình từ 5 đến 12 mô tả kết quả mô phỏng cho các thí nghiệm với kịch bản 1. Hình 5 biểu thị biến thiên của RTT trung bình. Hình 6 biểu thị biến thiên của đầu vào x được chọn là tỷ lệ RTT theo công thức (11) tương ứng với biến thiên RTT trong hình 5.

Trên hình 5, RTT biến thiên theo mức độ tắc nghẽn song giá trị lớn nhất không vượt quá 4,2 s. Khi bắt đầu, RTT ở giá trị nhỏ. Khi số gói tin nhiều hơn, tắc nghẽn gia tăng dẫn đến RTT tăng. CoAP không thể phát tiếp ngay, do đó số gói tin phát giảm dẫn đến RTT giảm. Quá trình lặp lại trong khoảng tiếp theo. Ở hình 6, trong 5 s đầu, CoAP chưa thể xác định ngay được RTT_{max} nên x có xung nhọn gần 1 do mẫu số của (11) nhỏ. Sau khi xác định được RTT_{max} , tỷ số x tăng dần khi RTT tăng và giảm khi RTT giảm theo đúng công thức (11).

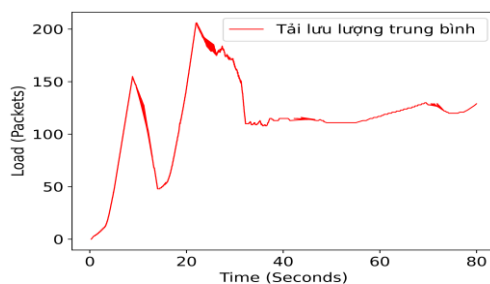


Hình 5. Biến thiên của RTT.

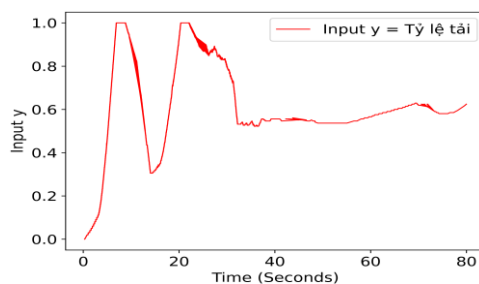


Hình 6. Chọn tỷ lệ RTT làm đầu vào x.

Hình 7 biểu thị tải lưu lượng, còn hình 8 biểu thị tỷ lệ theo công thức (14). Tải tăng dần trong khoảng từ 0 s đến 10 s. Từ 10 s đến 15 s, tắc nghẽn gia tăng nên CoAP không thể phát tiếp dẫn đến tải giảm. Tải đạt giá trị tối đa L_{max} tại 15 s. Tiếp đó, mạng bớt tắc nghẽn hơn nên tải có thể tăng tiếp. Đến thời điểm 20 s, tải lại đạt L_{max} do quá nhiều gói tin phát đi. Trong khoảng từ 35 s đến 80 s, tải không thể tăng tiếp do số gói tin phát đi đã cân bằng với khả năng truyền tin của kết nối $R1$ và $R2$. Hình 8 biểu thị giá trị y tương ứng tải với giá trị lớn nhất bằng 1 ở 10 s và 20 s. Trong các khoảng còn lại, y có giá trị nhỏ hơn 1. Như vậy, (14) có thể làm đầu vào y cho hệ FCS.

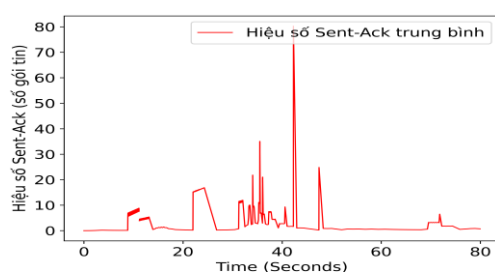


Hình 7. Biến thiên của tải lưu lượng.

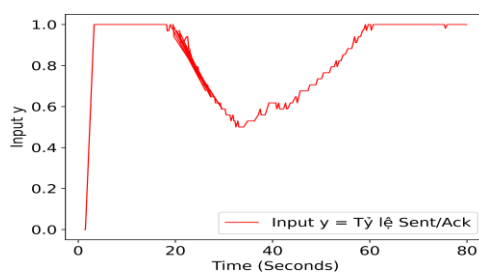


Hình 8. Chọn tỷ lệ tải làm đầu vào y.

Hình 9 biểu thị hiệu số Sent-Ack trung bình và hình 10 cho tỷ lệ Sent/Ack. Có thể thấy tham số y biến đổi rất ít, chủ yếu khi hiệu số Sent-Ack lớn, nghĩa là khi CoAP đã phát đi khá nhiều gói tin song chỉ có số lượng ít gói tin có phản hồi ACK trong tình huống tắc nghẽn mạng. Kết quả này cho thấy chọn tỷ lệ Sent/Ack ít phù hợp hơn như đã nêu ở phần tính toán.



Hình 9. Biến thiên của hiệu số Sent-Ack.



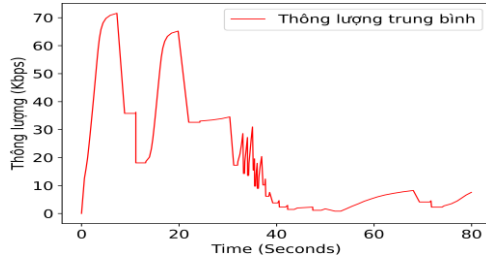
Hình 10. Chọn tỷ lệ Sent/Ack làm đầu vào y.

Hình 11 biểu thị thông lượng trung bình và hình 12 là đầu vào y. Tỷ số y biến đổi rất linh hoạt trong trường hợp này, phản ánh đúng trạng thái tắc nghẽn khi có quá nhiều gói tin phát đi. Khi mạng trở nên tắc nghẽn, thông lượng sẽ giảm (hình 11), tham số y biến thiên nhanh hơn biểu thị trạng thái động của liên kết mạng. Kết quả này phù hợp với tính toán ở phần lý thuyết.

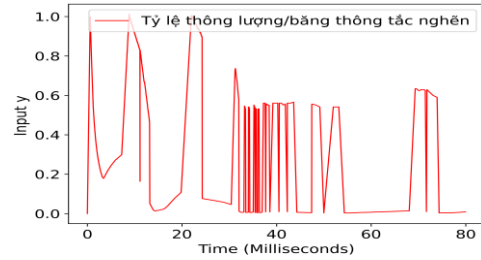
3.2.2. Nhóm thí nghiệm mô phỏng 2

Các hình từ 13 đến 20 mô tả kết quả mô phỏng cho các thí nghiệm với kịch bản 2. Hình 13 biểu thị biến thiên RTT khi có tắc nghẽn cao. RTT tăng nhanh tới giá trị 11 s ở thời điểm 45 s và duy trì ở giá trị 9 s do tắc nghẽn. Do RTT tăng nhanh, x nhanh chóng đạt bão hòa (bằng 1.0) ở

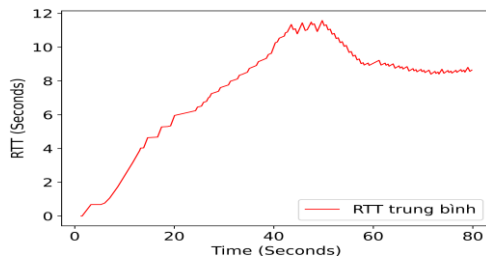
thời điểm 10 s và duy trì giá trị 1 cho tới thời điểm 50 s (hình 14). Theo công thức (11), khi RTT nhỏ hơn so với RTT_{max} , tham số x sẽ trở nên nhỏ hơn 1 từ sau 50 s như biểu thị trên hình 14.



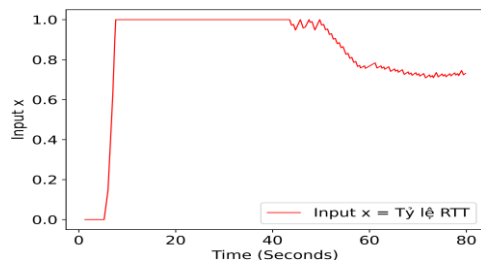
Hình 11. Biến thiên của thông lượng.



Hình 12. Chọn tỷ lệ thông lượng làm đầu vào.

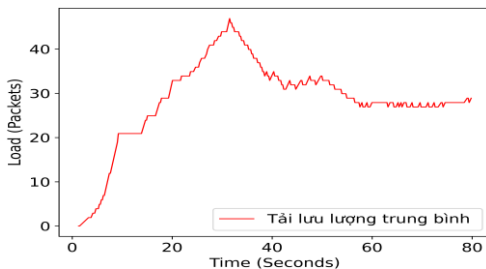


Hình 13. Biến thiên RTT khi tắc nghẽn cao.

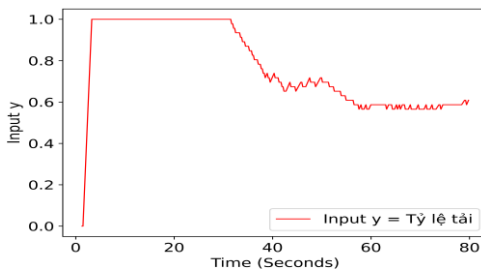


Hình 14. Đầu vào x khi tắc nghẽn cao.

Hình 15 và 16 biểu thị kết quả lựa chọn tỷ lệ tải lưu lượng theo công thức (14) trong trường hợp nguy cơ tắc nghẽn cao. Tương tự như ở nhóm thí nghiệm 1, các kết quả chỉ ra tỷ lệ tải lưu lượng có thể chọn làm đầu vào thứ 2 cho hệ FCS, song hệ số y sẽ ít thay đổi khi mạng bão hòa.

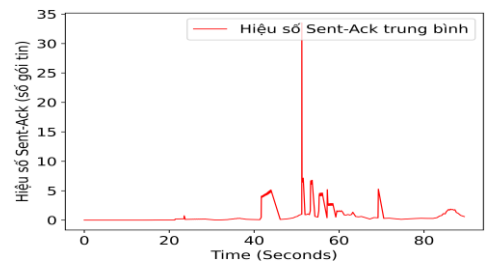


Hình 15. Tải lưu lượng khi tắc nghẽn cao.

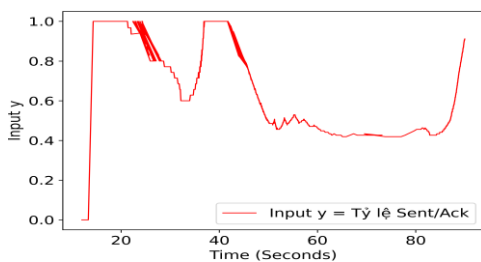


Hình 16. Tỷ lệ tải khi tắc nghẽn cao.

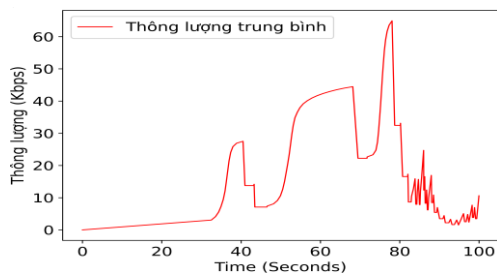
Hình 17 và 18 biểu thị kết quả lựa chọn theo công thức (16) trong trường hợp tắc nghẽn cao. So với nhóm thí nghiệm 1, tỷ lệ gói tin truyền thành công ít hơn do tắc nghẽn cao hơn. Tham số y vì vậy biến đổi nhiều hơn. Hình 19 và 20 biểu thị kết quả lựa chọn theo công thức (17). Kết quả mô phỏng phù hợp với tính toán ở phần lý thuyết. Trong trường hợp tắc nghẽn cao, thông lượng tăng chậm (hình 19) và biến thiên tùy vào điều kiện tắc nghẽn giảm bớt (dao động trong khoảng thời gian từ 35 s đến 90 s trong hình 19). Tham số y biến thiên tương ứng (hình 20).



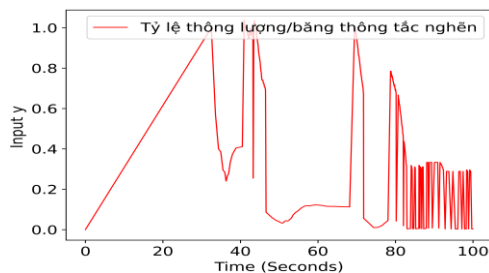
Hình 17. Hiệu số Sent/Ack khi tắc nghẽn cao.



Hình 18. Tỷ lệ Sent/Ack khi tắc nghẽn cao.

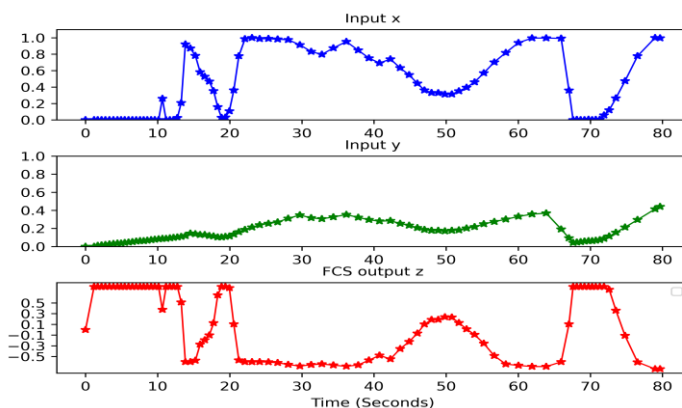


Hình 19. Thông lượng khi tắc nghẽn cao.



Hình 20. Tỷ lệ thông lượng khi tắc nghẽn cao.

3.2.3. Nhóm thí nghiệm mô phỏng 3

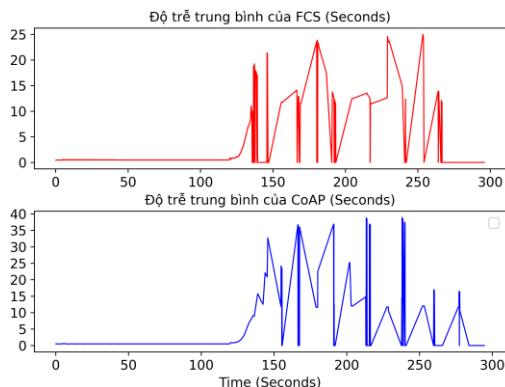


Hình 21. Biến thiên đầu ra z tương ứng với các đầu vào x và y .

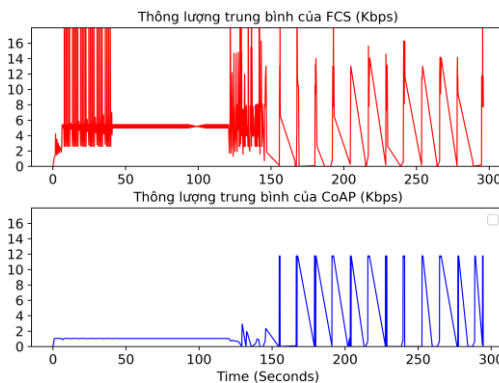
Hình 21 biểu thị đầu vào x , đầu vào y và đầu ra z của hệ FCS. Đầu vào x có giá trị gần 1 khi có tắc nghẽn và gần 0 khi không tắc nghẽn. Tương tự đối với đầu vào y . Giá trị đầu ra z âm biểu thị tắc nghẽn, càng gần tới $-0,5$ mạng càng tắc nghẽn. Đầu ra z dương khi không có tắc nghẽn.

3.2.4. Nhóm thí nghiệm mô phỏng 4

Hình 22 biểu thị độ trễ của FCS và CoAP trong kịch bản mô phỏng 4. Trong thời gian từ $0 s - 130 s$, cả FCS và CoAP đều không có tắc nghẽn. Khi 8 nút IoT khác phát thêm các gói tin, độ trễ của FCS và CoAP tăng dần. Từ thời điểm $130 s$, tắc nghẽn nghiêm trọng đã xảy ra dẫn đến mất gói tin cho cả FCS và CoAP. Độ trễ biến thiên lớn, dao động từ $0 s - 24 s$ ở FCS và từ $0 s - 38 s$ ở CoAP do timeout của các gói tin. Có thể thấy FCS mất ít gói tin hơn CoAP và độ trễ trung bình của FCS nhỏ hơn so với CoAP. Hình 23 so sánh thông lượng trung bình của FCS và CoAP trong điều kiện tắc nghẽn nghiêm trọng. Các chỉ số hiệu năng đo được biểu thị trong bảng 1.



Hình 22. Độ trễ của FCS và CoAP.



Hình 23. Thông lượng của FCS và CoAP.

Trong bảng 1, số gói tin phát đi của FCS là 827, của CoAP là 385. Số gói tin phát đi thành công của FCS là 745 (90,08%) cao hơn so với 260 (67,53%) của CoAP. Tỷ lệ mất gói tin của CoAP là 15,33% cao hơn so với 0,77% của FCS. Số gói tin phát lại của FCS là 198 (23,94%) nhỏ hơn so với 252 (65,45%) của CoAP. Số gói tin phát lại trùng lặp của FCS là 127 (15,36%) nhiều hơn 52 (13,50%) của CoAP. Độ trễ trung bình của FCS là 2191,57 ms nhỏ hơn nhiều so với 8142,17 ms của CoAP. Thông lượng trung bình của FCS là 5,562 Kbps cao hơn so với 4,998 Kbps trong CoAP. Kết quả cho thấy FCS đạt hiệu năng cao hơn CoAP.

Bảng 1. So sánh các chỉ số hiệu năng giữa FCS và CoAP.

	FCS	CoAP
Tổng số gói tin đã phát	827	385
Tổng số gói tin nhận được (tỷ lệ%)	745 (90,08%)	260 (67,53%)
Số gói tin bị mất (tỷ lệ%)	64 (0,77%)	59 (15,33%)
Số gói tin phát lại (tỷ lệ%)	198 (23,94%)	252 (65,45%)
Số gói tin bị phát lại trùng lặp (tỷ lệ%)	127 (15,36%)	52 (13,50%)
Độ trễ trung bình	2191,57 ms	8142,17 ms
Thông lượng trung bình	5,562 Kbps	4,998 Kbps

4. KẾT LUẬN

Bài báo đã trình bày nhu cầu điều khiển chống tắc nghẽn trong mạng IoT, phân tích các hạn chế trong cơ chế điều khiển của CoAP và các cải tiến nó. Trên cơ sở đó, bài báo đưa ra một giải pháp thực thi hệ điều khiển mờ FCS cho CoAP với đề xuất lựa chọn tham số đầu vào và đầu ra phù hợp nhằm chống tắc nghẽn mạng. Các kết quả thí nghiệm đã chứng tỏ giải pháp đưa ra phù hợp lý thuyết. Hệ FCS có hiệu quả tốt hơn CoAP và có thể áp dụng vào thực tế truyền tin cho các ứng dụng IoT thông minh hiện nay.

Các nghiên cứu tiếp theo có thể phân tích, đánh giá các điểm giao của các giá trị hàm thuộc đầu vào và tập luật mờ để đưa ra khả năng lựa chọn tham số tối ưu hơn cho mô hình FCS.

TÀI LIỆU THAM KHẢO

- [1]. RFC 7252, "The Constrained Application Protocol (CoAP)," available: <https://rfc-editor.org/info/rfc7252>.
- [2]. C. Bormann, Z. Shelby, "Block-Wise Transfers in the Constrained Application Protocol (CoAP)," Available: <https://rfc-editor.org/info/rfc7959>.
- [3]. H. Haile, K. Grinnemo, S. Ferlin, et.al., "End-to-end congestion control approaches for high throughput and low delay in 4G/5G cellular networks," in Computer Networks, Vol. 186-107692, pp. 1-22, (2021).
- [4]. H. Jiang, Q. Li, G. Shen, et.al., "When Machine Learning Meets Congestion Control: A Survey and Comparison," in Computer Networks, vol. 192-108033, pp. 1-23, (2021).
- [5]. F. Righetti, et al. "Investigating the CoAP Congestion Control Strategies for 6TiSCH-Based IoT Networks," in IEEE Access 11, pp. 11054-11065, (2023).
- [6]. M.A. Tariq, M. Khan, M.T.R. Khan, D. Kim, "Enhancements and Challenges in CoAP – A Survey," in Sensors, vol. 20, pp. 1-29, (2020).
- [7]. P.K. Donta, S.N. Srirama, et al., "iCoCoA: intelligent congestion control algorithm for CoAP using deep reinforcement learning," in Journal Ambient Intell Human Computer, Vol. 14, pp. 2951–2966, (2023).
- [8]. P. Aimtongkham, P. Horkaew, C. So-In, "An Enhanced CoAP Scheme Using Fuzzy Logic with Adaptive Timeout for IoT Congestion Control," in IEEE Access, Vol. 9, pp.58967-58981, (2021).
- [9]. T. N. Pham, D. H. Hoang, T. T. Duong Le, "Fuzzy Congestion Control and Avoidance for CoAP in IoT Networks," in IEEE Access, Vol. 10, pp. 105589-105611, (2022).
- [10]. L.A. Zadeh, "Outline of a new approach to the analysis of complex systems and decision processes," in IEEE Transactions on Systems, Man, and Cybernetics 3(1), pp. 28–44, (1973).

- [11]. T.J. Ross, “*Fuzzy Logic with Engineering Applications*,” Wiley Publisher, 3rd Edition, ISBN-10:047074376X, (2010).
- [12]. RFC 6298, “*Computing TCP's Retransmission Timer*,” available: <https://rfc-editor.org/info/rfc6298>
- [13]. N. Cardwell, Y. Cheng, C. S. Gunn, S. H. Yeganeh, and V. Jacobson, “*BBR: Congestion-Based Congestion Control*,” ACM Queue, Vol. 14, No. 5, pp. 50:20–53, (2016).
- [14]. H.J. Zimmerman, “*Fuzzy set theory - and its applications*,” Kluwer Academic Publishers, Springer Science, Fourth Edition (2001).
- [15]. *NS-3 Network Simulator, NS3.36*, available: <https://www.nsnam.org/>.

ABSTRACT

Fuzzy control to support CoAP for congestion avoidance in the Internet of Things networks

The CoAP (Constrained Application Protocol) protocol and its improvements are still limited in its ability to detect congestion early and adjust transmission rates to match the dynamic state of the Internet of Things networks. This paper proposes a solution to implement a fuzzy control mechanism for network congestion avoidance with the selection of appropriate input and output parameters. The parameters are evaluated by the simulation tool. The simulation results show that the parameter selection is suitable for the theory, allowing the fuzzy control system to achieve higher performance indices compared to the standard CoAP.

Keywords: CoAP protocol; Congestion Control; Fuzzy Control; IoT networks.