

GIẢI PHÁP ĐIỀU KHIỂN CHỐNG TẮC NGHẼN TRONG MẠNG IoT

Hoàng Đăng Hải¹, Lê Thị Thùy Dương², Phạm Thiếu Nga²

¹ Học viện Công nghệ Bưu chính Viễn thông

² Trường Đại học Xây dựng Hà Nội

Tóm tắt: Tắc nghẽn mạng là một vấn đề tồn tại cơ bản trong mọi loại mạng. Với sự phát triển gia tăng của mạng Internet vạn vật (IoT – Internet of Things), số lượng thiết bị kết nối ngày càng nhiều, nguy cơ xảy ra tắc nghẽn mạng ngày càng nghiêm trọng. Mỗi trường mạng IoT có những đặc điểm khác biệt so với mạng Internet truyền thống. Do vậy, các cơ chế điều khiển chống tắc nghẽn (CC – Congestion Control) của mạng Internet truyền thống không thể áp dụng nguyên vẹn cho mạng IoT, đòi hỏi có những thay đổi phù hợp để bảo đảm thông lượng và chất lượng truyền tin. Bài báo phân tích các điểm khác biệt trong điều khiển chống tắc nghẽn giữa mạng IoT và mạng Internet truyền thống, khảo sát và phân tích một số công trình nghiên cứu liên quan. Trên cơ sở khảo sát các cơ chế điều khiển chống tắc nghẽn hiện có và phân tích các đặc thù của mạng IoT, bài báo tổng hợp một số hướng giải pháp điều khiển chống tắc nghẽn cho mạng IoT.

Từ khóa: Mạng IoT, Tắc nghẽn mạng, Điều khiển chống tắc nghẽn, Định trình, Quản lý bộ đệm tích cực

I. MỞ ĐẦU

Khái niệm mạng Internet vạn vật (IoT - Internet of Things) có từ khoảng năm 1999, được dùng để mô tả mạng của đa dạng các loại thiết bị có gắn cảm biến, kết nối vào Internet. IoT được xem như một công nghệ mạng mới kết nối vạn vật với mạng Internet, phục vụ nhu cầu tương tác đa dạng giữa thế giới vật lý (gồm các cảm biến, các bộ điều khiển) với thế giới số. Với định hướng kết nối vạn vật cho những vật thể thông minh có khả năng tương tác với nhau, IoT tạo thêm khả năng truyền tin mới giữa người với vật thể và giữa vật thể với vật thể, thay vì chỉ có một cơ chế truyền tin truyền thống là giữa người với người [3, 47]. Điều đó dẫn đến khả năng phải tiếp nhận và xử lý một lượng thông tin rất lớn từ một số lượng lớn các vật thể.

Một đặc trưng của mạng IoT là gồm rất nhiều bộ cảm biến (Sensor) và các bộ thực thi (Actuator) [45, 35]. Các bộ cảm biến làm nhiệm vụ thu thập dữ liệu từ môi trường. Bộ thực thi là một thiết bị thực hiện các tác vụ giám sát, điều khiển làm biến đổi môi trường, cụ thể là biến đổi điện năng thành một số dạng năng lượng nhất định như cơ năng, nhiệt năng, v.v. Hầu hết các ứng dụng IoT đều cần ít nhất một hoặc nhiều bộ

cảm biến và bộ thực thi. Trong các ứng dụng đó, mạng không dây đóng một vai trò quan trọng. Vì lẽ đó, mạng cảm biến không dây (Wireless Sensor Networks) được coi là một thành phần nền tảng của mạng IoT [18, 35]. Mặt khác, các công nghệ mạng không dây được sử dụng cho kết nối ở tầng vật lý gồm nhiều chủng loại như RFID, NFC, ZigBee, LoRa, WiFi, 4G/LTE, v.v. [3, 35]. Điều này gây khó khăn cho các cơ chế điều khiển và việc chuyển đổi giữa các giao thức trở nên phức tạp.

Các ứng dụng IoT hết sức đa dạng và phong phú, điển hình như y tế từ xa, vận tải thông minh, ngôi nhà thông minh, đô thị thông minh, nông nghiệp thông minh, tự động hóa trong nhà máy, theo dõi dây chuyền sản xuất, giám sát môi trường, ứng dụng trong công nghiệp, v.v. Sự đa dạng về công nghệ lớp vật lý và các dịch vụ, ứng dụng với các đặc tính lưu lượng và yêu cầu dịch vụ khác nhau của các ứng dụng IoT đặt ra những đòi hỏi khác biệt về các kiến trúc mạng truyền tin và các giao thức mạng nhằm đáp ứng yêu cầu của từng loại ứng dụng đó.

Các ứng dụng IoT có các đặc trưng dữ liệu đa dạng và yêu cầu về chất lượng dịch vụ (Quality of Service) rất khác nhau [14, 25]. Số lượng thiết bị IoT và lượng dữ liệu truyền qua mạng IoT có thể rất lớn. Do vậy, mạng IoT cần có cơ chế CC (Congestion Control) phù hợp với các đặc điểm đa dạng nêu trên.

Điều khiển chống tắc nghẽn trong mạng IoT có những khác biệt so với trong mạng Internet truyền thống. Một phần là do phương thức truyền tải dữ liệu của mạng IoT có thể theo nhiều cách: theo sự kiện, liên tục, theo truy xuất hoặc hỗn hợp. Trong ứng dụng theo sự kiện, lưu lượng mạng bình thường ở mức thấp và có thể đột ngột cao khi xảy ra sự kiện. Trong ứng dụng theo kiểu liên tục, các nút mạng cảm biến định kỳ gửi các gói tin đến đích sau những khoảng thời gian xác định. Trong ứng dụng theo kiểu truy xuất, nút mạng cảm biến sẽ gửi một lượng dữ liệu theo yêu cầu của nút đích. Ứng dụng hỗn hợp gồm cả ba thể loại ứng dụng nêu trên [5, 25]. Ứng dụng IoT có thể yêu cầu thời gian thực, độ tin cậy, nội dung đa phương tiện như âm thanh, hình ảnh, văn bản, video. Do vậy, ảnh hưởng của tắc nghẽn trong mạng IoT cần được xem xét cụ thể.

Các cơ chế CC cho mạng Internet truyền thống không còn phù hợp, không thể áp dụng ngay cho mạng IoT mà cần có sự điều chỉnh, sửa đổi phù hợp. Cơ chế CC của giao thức TCP (TCP-CC) được thiết kế chủ yếu cho mạng có dây truyền thống với giả thiết sự cố

Tác giả liên hệ: Hoàng Đăng Hải

Email: haihd@ptit.edu.vn

Đến tòa soạn: 03/2019, chỉnh sửa: 04/2019 chấp nhận đăng: 05/2019

tắc nghẽn đủ kéo dài để có phản hồi từ phía đầu cuối nhận. Cơ chế điều khiển tương đối chậm sau khi có thể một lượng lớn dữ liệu đã được gửi đi. Nếu lượng dữ liệu nhỏ, cơ chế này không hiệu quả [9]. Nhiều nghiên cứu đã chỉ ra TCP-CC không dùng được cho mạng IoT do phản ứng chậm của cơ chế điều khiển và khả năng phát hiện tắc nghẽn, ví dụ [2, 5, 9, 11, 19, 26, 46, 47]. Một số nghiên cứu đã chỉ ra những hạn chế cụ thể của TCP-CC về tính phức tạp, dự đoán tắc nghẽn chưa chính xác, suy giảm đáng kể thông lượng mạng, v.v. khi dùng cho mạng IoT, điển hình là các nghiên cứu trong [10, 25, 13, 3, 14, 17, 20, 18, 35, 37, 4, 12, 31, 28]. Trong vài năm qua, một số công trình nghiên cứu đã đề xuất cơ chế CC riêng cho mạng IoT, điển hình là các công trình [8, 9, 30, 34, 20, 21, 18, 29, 32, 35, 44, 2, 4, 5, 6, 7, 19, 24, 28, 31, 36, 39, 40, 43, 47].

Ngoài ra, môi trường mạng IoT khác biệt cũng tạo thêm nhiều khó khăn cho cơ chế CC. Điển hình là: kiến trúc mạng đa dạng và khác biệt (gồm nhiều phân đoạn Fog kết nối Cloud), các lớp giao thức hỗn hợp để liên kết mạng, các đầu cuối IoT đa dạng và có nhiều hạn chế về tài nguyên (bộ nhớ đệm, năng lực xử lý, kênh truyền).

Vì những lý do nêu trên, rất cần nghiên cứu phân tích các điểm khác biệt trong CC giữa mạng IoT và mạng Internet truyền thống, để từ đó có được những giải pháp điều khiển chống tắc nghẽn hiệu quả và phù hợp. Đó là trọng tâm chính của bài báo này. Ngoài ra, các đóng góp khác của bài báo là: khảo sát và phân tích một số công trình nghiên cứu liên quan, tổng hợp một số hướng giải pháp CC cho mạng IoT. Bộ cục phần còn lại của bài báo gồm: Phần 2 phân tích về vấn đề điều khiển chống tắc nghẽn trong mạng IoT so với mạng Internet truyền thống, Phần 3 trình bày một số nghiên cứu liên quan, Phần 4 trình bày một số hướng giải pháp và cuối cùng là phần kết luận.

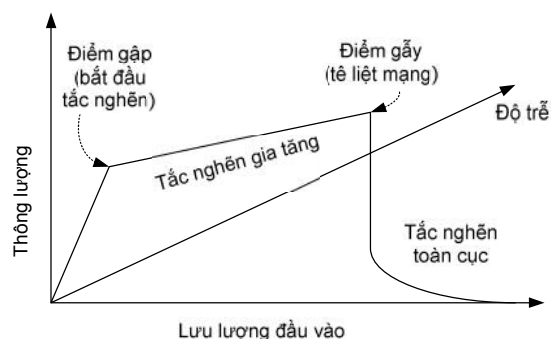
II. ĐIỀU KHIỂN CHỐNG TẮC NGHE TRONG MẠNG IOT SO VỚI MẠNG INTERNET TRUYỀN THỐNG

A. Điều khiển chống tắc nghẽn trong mạng Internet truyền thống

Tắc nghẽn là một hiện tượng phổ biến trên mạng, thường xảy ra khi các nút mạng không thể xử lý kịp các gói tin đến, bộ đệm lưu giữ gói tin ở nút mạng bị

đến được lưu vào bộ đệm nút mạng, chờ xử lý để đưa ra khỏi nút mạng theo một tuyến đường đã chọn để đến đích. Nếu lượng gói tin đến càng lớn, thời gian nghẽn mạng càng kéo dài, số gói tin bị loại bỏ do không còn chỗ lưu càng nhiều dẫn đến nguy cơ mạng tê liệt hoàn toàn.

Hình 1 biểu thị mối quan hệ giữa lưu lượng đầu vào, thông lượng và độ trễ. Khi lưu lượng đầu vào tăng, thông lượng tăng. Bên trái điểm gập (mạng không tắc nghẽn), bộ đệm có kích thước vừa đủ cho các gói tin đến, không loại bỏ gói tin nào. Độ trễ có thể tăng nhỏ khi có nhiều gói tin chờ xử lý. Trong đoạn giữa điểm gập và điểm gãy, số gói tin được lưu trong bộ đệm chờ được xử lý ngày càng tăng. Nếu bộ đệm không đủ lớn, một số gói tin có thể bị loại bỏ do chờ quá lâu hoặc do tràn bộ nhớ. Số lượng gói tin loại bỏ và số gói tin được xử lý, chuyển tiếp phụ thuộc vào năng lực xử lý của nút và tốc độ kênh truyền. Khi lưu lượng tiếp tục gia tăng, bắt đầu từ điểm gãy, toàn bộ gói tin đến đều bị vứt bỏ và mạng tê liệt hoàn toàn.

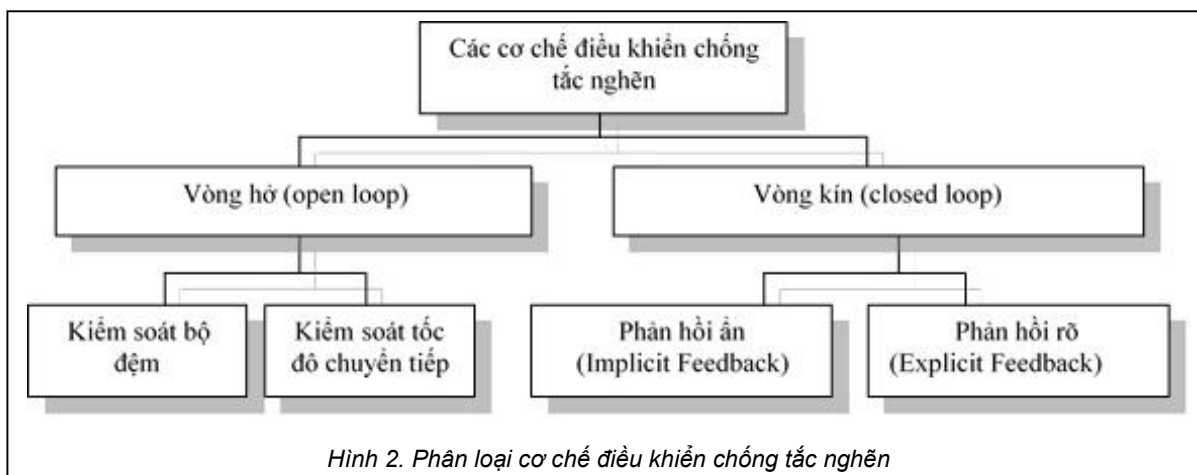


Hình 1. Hiện tượng tắc nghẽn mạng

Nguyên tắc chung của điều khiển chống tắc nghẽn là duy trì hoạt động của mạng ở bên trái điểm gãy, hoặc tối thiểu bên trái điểm gãy.

Các cơ chế điều khiển và chống tắc nghẽn có thể chia thành hai nhóm: cơ chế vòng hở và cơ chế vòng kín (xem hình 2) (tóm lược từ [16]).

- **Các cơ chế vòng hở:** Mỗi nút mạng tự kiểm soát lưu lượng đầu vào, đầu ra phù hợp với trạng thái nút, không có thông tin phản hồi từ phía mạng hoặc nút nhận. Cơ chế phổ biến có thể là quản lý bộ đệm



Hình 2. Phân loại cơ chế điều khiển chống tắc nghẽn

tràn. Mạng Internet được thiết kế theo cách lưu trữ và chuyển tiếp (Store and Forward), nghĩa là các gói tin

chống tràn, kiểm soát tốc độ chuyển tiếp gói tin, kiểm soát tiếp nhận gói tin đến. Cơ chế điều khiển luồng tin

cũng là một biện pháp nhằm hạn chế nút gửi phát đi quá nhiều gói tin so với khả năng xử lý của mạng.

- **Các cơ chế vòng kín:** Cơ chế này thường dùng cho kiểm soát tốc độ phát tin từ nút gửi với thông tin phản hồi từ nút nhận hoặc từ mạng. Phản hồi có thể là ẩn (implicit) hoặc rõ (explicit). Phản hồi ẩn thường do mạng cung cấp, căn cứ vào trạng thái mạng thực tế. Ví dụ thông qua bản tin ICMP (Internet Control Message Protocol) hoặc SNMP (Simple Network Management Protocol) báo về sự cố mạng xảy ra. Phản hồi rõ thường do nút nhận gói tin cung cấp về nút gửi tin. Bản tin gửi về thường chứa thông tin cụ thể về tỷ lệ mất gói, độ trễ. Bản tin ACK (Acknowledgement) của TCP là một ví dụ.

Đã có khá nhiều cơ chế chống tắc nghẽn cho mạng Internet truyền thống. Trong phần sau đây, bài báo trình bày tóm tắt các cơ chế điển hình nhất.

1) Cơ chế định trình (Scheduling)

Mục đích của các bộ định trình là kiểm soát tốc độ chuyển tiếp gói tin sao cho tránh gói tin phải đợi lâu (giảm độ trễ) và giảm tỷ lệ mất gói. Các gói tin đến nút mạng được sắp xếp vào bộ đệm không phải theo cách truyền thống là đến trước phục vụ trước (FIFO – First In First Out), mà theo cách có lựa chọn để chuyển tiếp đi phù hợp với tốc độ kênh truyền. Tổng hợp về các cơ chế định trình có thể xem trong [15].

Các công trình nghiên cứu trước đây (ví dụ xem [15]) đã chỉ ra rằng, nếu lưu lượng đầu vào mạng thỏa mãn điều kiện thùng rò (Leaky Bucket), thì sẽ có thể thiết kế cơ chế định trình phù hợp bảo đảm chất lượng dịch vụ (độ trễ, độ rung trễ, tỷ lệ mất gói), tránh được tắc nghẽn.

Thùng rò có hai tham số đặc trưng là tốc độ đến tối đa và kích thước bộ đệm tối đa, cho phép mạng chỉ chấp nhận một lượng gói tin gửi từ các nút đến mạng tối đa.

Do các gói tin đến từ đa dạng nguồn gửi với tốc độ phát rất khác nhau, các cơ chế định trình bình đẳng (Fair Queueing) đã được đề xuất, điển hình nhất là cơ chế Weighted Fair Queueing (WFQ) (xem ví dụ [15, 39]). Mặt khác, nhằm bảo đảm chất lượng dịch vụ (Quality of Service), các cơ chế WFQ thường được kết hợp với các cơ chế khác như: cơ chế tiếp nhận kết nối (Admission Control), cơ chế quản lý bộ đệm, cơ chế dành sẵn tài nguyên, cơ chế ưu tiên gói tin, v.v.

2) Cơ chế quản lý bộ đệm tích cực (Active Buffer Management)

Thay vì cơ chế loại bỏ khi tràn (DropTail) truyền thống, các cơ chế quản lý bộ đệm tích cực (Active Buffer Management) tìm cách phát hiện sớm nguy cơ tràn để loại bỏ các gói tin (tùy ý hoặc theo mức ưu tiên thấp hơn), nghĩa là phát hiện sớm nguy cơ tắc nghẽn mạng. Điển hình là các cơ chế RED (Random Early Detection), BLUE, FRED (Flow Random Early Detection), CHOCe (xem ví dụ [15, 1]).

Để phát hiện sớm nguy cơ tắc nghẽn, RED liên tục kiểm soát kích thước (hay độ dài) trung bình bộ đệm, so sánh nó với hai mức ngưỡng. Độ dài trung bình bộ đệm được đo bằng kỹ thuật EWMA (Exponential Weighted Moving Average). Nếu độ dài này nhỏ hơn

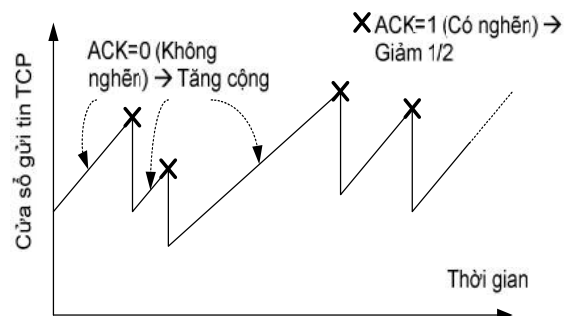
mức ngưỡng thấp, RED không bỏ gói tin. Nếu độ dài bộ đệm trong khoảng mức ngưỡng thấp và cao, RED sẽ loại bỏ gói tin ngẫu nhiên hoặc đánh dấu để bỏ khi cần và báo cho nút mạng kế tiếp biết bằng một bit cờ báo rõ (ECN - Explicit Congestion Notification). Nếu vượt qua mức ngưỡng cao, RED loại bỏ hoặc đánh dấu tất cả các gói tin đến.

Cơ chế RED tỏ ra rất phù hợp khi kết hợp với cơ chế CC của TCP. Tuy nhiên, việc xác định hai mức ngưỡng vô cùng khó khăn. Trong thời gian qua, đã có khá nhiều phiên bản RED như FRED, SRED, DRED, ARED và các đề xuất khác thay thế RED (ví dụ xem [15, 10, 1, 13, 37, 12]).

3) Các cơ chế TCP – CC

Cơ chế TCP-CC cơ bản nhất dựa trên thuật toán tăng cộng – giảm nhân (AIMD – Additive Increase, Multiplicative Decrease), dựa theo cửa sổ (Window-based). Cửa sổ W biểu thị cho số lượng gói TCP tối đa đang di chuyển trên mạng đối với một luồng tin TCP. Nút gửi TCP nhận biết tắc nghẽn thông qua bản tin ACK phản hồi từ nút nhận TCP. Dấu hiệu cơ bản để nhận biết tắc nghẽn là có lỗi mất gói tin, được bên nhận phát hiện thông qua kiểm tra số thứ tự của gói tin đến đích.

Nếu xảy ra tắc nghẽn, kích thước cửa sổ của TCP tại nút gửi giảm đi một nửa ($W := W \cdot 0.5$), ngược lại thì tăng lên một ($W := W + 1$).



Hình 3. Cơ chế TCP - CC

Cơ chế TCP-CC rất phổ biến trong Internet. So với phiên bản nguyên thủy, các phiên bản sau của TCP như TCP Reno, TCP New-Reno, TCP SACK, TCP SYN/ACK đã có nhiều cải tiến đáng kể hiệu quả chống tắc nghẽn. Các cải tiến quan trọng gồm: định cỡ cửa sổ, bổ sung pha khởi động chậm (Slow Start), cách tính thời gian quay vòng (RTT - Round Trip Time), tính Time-Out (RTO), cách xác định mất gói, tỷ lệ mất gói và ACK, v.v.

4) Các cơ chế tương tự TCP – CC

TCP không phù hợp cho các luồng tin đa phương tiện, do vậy các cơ chế tương tự TCP (TCP-like hay TCP-Friendly) đã ra đời. Các cơ chế này có thể dựa trên cửa sổ (Window-based) như TCP hoặc dựa theo tốc độ (Rate-based), song đa số là Rate-based vì có phản ứng nhanh hơn [15]. Theo kiểu cửa sổ, điển hình là các cơ chế EWA (Explicit Windows Adaptation), ETCP (Enhanced TCP), XCP (Explicit Control Protocol), QS-TCP (Quick Start TCP) [16]. Theo kiểu tốc độ, điển hình là các cơ chế RAP (Rate Adaptation

Protocol), RCAP (Rate Control Adaptive Protocol), TFRC (TCP Friendly Rate Control) [15].

Phương thức cơ bản của các cơ chế kiểm soát tốc độ là: tăng dần tốc độ nếu không thấy tắc nghẽn, đặt tốc độ ở mức cần thiết khi có tắc nghẽn. Phát hiện tắc nghẽn vẫn chủ yếu dựa vào tỷ lệ mất gói tính được ở phía nhận và gửi bản tin phản hồi về bên phát gói tin. Tốc độ phát gói tin được tính theo công thức dựa vào tỷ lệ mất gói, RTT, kích thước gói tin. Một số cải tiến bổ sung thêm giá trị RTO, hệ số TCP phù hợp.

5) Các cơ chế khác

Ngoài các cơ chế nêu trên, có một số cơ chế khác được đề xuất nhằm tăng hiệu quả chống tắc nghẽn, điển hình như: các cơ chế phát hiện sớm, các cơ chế thông báo tắc nghẽn, các cơ chế kiểm soát và tránh tắc nghẽn. Các cơ chế phát hiện tắc nghẽn có thể phân biệt nguyên nhân tắc nghẽn do lỗi bộ đệm, nhiễu, lỗi kênh truyền (lỗi kênh vô tuyến ở mạng Internet không dây), do độ trễ, v.v. Kiểm soát và tránh tắc nghẽn có thể thông qua cân bằng tải (Load Balancing), tái định tuyến (Re-routing, chọn tuyến ít tải hơn), điều chỉnh tốc độ phát gói ở lớp ứng dụng của nguồn phát, sử dụng play-out buffer, v.v.

B. Sự khác biệt của mạng IoT trong điều khiển chống tắc nghẽn

Xét về bản chất, mạng IoT thực tế là sự mở rộng của Internet. Tuy nhiên, các cơ chế CC trong mạng Internet truyền thống (gọi tắt là Internet CC hay I-CC) không còn phù hợp cho mạng IoT vì những sự khác biệt cơ bản sau.

- I-CC được thiết kế chủ yếu trên nền tảng mạng hữu tuyến, trong khi đó mạng IoT chủ yếu là môi trường vô tuyến.

Mạng hữu tuyến có tỷ lệ mất gói do lỗi kênh thấp hơn nhiều so với mạng vô tuyến. Do vậy, cơ chế I-CC chủ yếu dựa vào lỗi mất gói sẽ phát hiện sai khi có nhiễu, lỗi kênh vô tuyến hoặc suy giảm tín hiệu vô tuyến. Vấn đề phát sinh là lỗi kênh vô tuyến thường trong các khoảng thời gian rất ngắn và rất khó phân biệt mất gói tin do tràn bộ đệm hay do lỗi kênh vô tuyến. Việc giảm cửa sổ một nửa mỗi khi phát hiện có lỗi (mất gói hoặc lỗi kênh) khiến hiệu suất TCP trong mạng vô tuyến trở nên rất thấp, không thể chấp nhận được. Khi môi trường vô tuyến, ví dụ WiFi trở thành phổ biến, đã có khá nhiều đề xuất cải tiến cho TCP-CC. Tuy nhiên, nhiều nghiên cứu đã chỉ ra nhiều bất cập của I-CC cho mạng IoT và đề xuất các cơ chế mới (xem [10, 9, 25, 8, 13, 3, 14, 17, 20, 46, 18, 35, 37, 4, 11, 12, 31, 28]).

- Các thiết bị đầu cuối IoT khá đa dạng và thường hạn chế về tài nguyên (bộ đệm, năng lực xử lý, băng thông kênh truyền, nguồn pin)

Một cơ chế điều khiển dựa theo cửa sổ hoặc tốc độ tại nguồn phát theo kiểu TCP hoặc tương tự TCP sẽ không còn phù hợp do không đủ tài nguyên. Các cơ chế định trình hoặc quản lý bộ đệm tích cực truyền thống cũng khó lòng áp dụng do nút mạng có thể không đủ năng lực xử lý [19].

Do TCP không còn phù hợp, một số giao thức đã được phát triển cho IoT ví dụ như CoAP, CoCoA,

MQTT, AMQP, PCCP, CODA ở tầng truyền tải, giao thức chuyển đổi như 6LoWPAN [5, 9, 8, 3, 34, 24, 28]. Điểm cơ bản của các giao thức này là đơn giản, gọn nhẹ cho thiết bị IoT, song vẫn đảm bảo tính năng cần thiết và tiết kiệm năng lượng.

- Kết nối mạng trong IoT chủ yếu là từng chặng vô tuyến (Hop-by-Hop)

Cơ chế dựa trên phản hồi đầu cuối của I-CC không còn phù hợp do phản hồi từ bên nhận có thể bị trễ lớn, phản hồi về tỷ lệ mất gói bị sai lệch dẫn đến việc điều chỉnh tốc độ nguồn phát không đúng. Hầu hết các cơ chế I-CC đều có hiệu quả thấp đối với kênh truyền vô tuyến có tỷ lệ bit lỗi cao và khi trễ ở lớp MAC xấp xỉ RTO (Retransmission Time-Out) [2].

Điển hình trong kiến trúc mạng IoT là sự đa dạng về các tầng giao thức ở mỗi chặng do có sự đa dạng về công nghệ truyền dẫn ở lớp dưới. Yêu cầu về điều chỉnh, sửa đổi các tầng giao thức là điều cần thiết.

- Cơ chế AIMD truyền thống dùng chủ yếu cho I-CC không còn phù hợp

Như đã phân tích ví dụ trong [38, 2, 39], các luồng tin từ các đầu cuối IoT có các yêu cầu về băng thông rất khác nhau. Nếu áp dụng cơ chế AIMD truyền thống chung cho các luồng tin sẽ dẫn đến việc giảm thông lượng của tất cả các ứng dụng IoT. Mặt khác, AIMD có thể gây ra biến thiên lưu lượng không mong muốn.

I-CC và AIMD chủ yếu hiệu quả khi chuyển một lượng gói tin lớn với thời gian trễ đủ lớn. Nếu lượng dữ liệu cần truyền nhỏ, các cơ chế này không hiệu quả [2]. Mặt khác, RTT trong mạng IoT cũng cần điều chỉnh phù hợp để cơ chế AIMD hoạt động hiệu quả. Tuy nhiên, việc điều chỉnh các tham số để cơ chế AIMD phù hợp cho mạng IoT là điều rất khó. I-CC và AIMD phản ứng với tắc nghẽn rất chậm [39].

III. CÁC NGHIÊN CỨU LIÊN QUAN

Như đã nêu ở phần I và II, mạng IoT có những khác biệt so với mạng Internet truyền thống, do vậy các cơ chế CC của mạng Internet truyền thống không thể áp dụng cho mạng IoT nếu không có những thay đổi cho phù hợp. Các đề xuất cải tiến cơ chế I-CC và đề xuất cơ chế CC mới cho mạng IoT có thể được phân loại thành các nhóm sau đây.

A. Cơ chế định trình

Ý tưởng đánh dấu mức độ ưu tiên cho các gói tin được trình bày trong [34]. Khi có tắc nghẽn, cơ chế CC mới sẽ loại bỏ gói tin tùy theo mức độ ưu tiên và chuyển tiếp các gói tin theo các cách khác nhau. Tuy nhiên, cách phát hiện tắc nghẽn trong bài báo [34] vẫn dựa theo cơ chế RED truyền thống.

Một giao thức tương tự TCP có kết hợp điều chỉnh tốc độ và cơ chế WFQ cho CC trong mạng MANET (Mobile Ad hoc Network) được đề xuất trong [39]. Giao thức TFRC (TCP Friendly Rate Control) được sử dụng cho điều chỉnh tốc độ, song được hiệu chỉnh về thời gian phản ứng, cách tính RTT, cách tính tỷ lệ mất gói trung bình theo trọng số. Cơ chế WFQ được áp dụng nguyên vẹn.

Trong [41], một cơ chế định trình có ưu tiên được đề xuất cho các kết nối IoT thời gian thực và các kết nối Internet truyền thống. Các gói tin đến từ các nguồn phát khác nhau được đánh dấu mức ưu tiên và chuyển tiếp dựa theo tốc độ, kích thước gói tin và kiểu gói tin. Tuy nhiên, tác giả chỉ đề xuất hai bộ đệm, một cho các gói tin ưu tiên (yêu cầu thời gian thực) và một cho gói tin không ưu tiên. Cơ chế định trình dự tính thời gian chuyển tiếp các gói tin căn cứ vào hệ số ưu tiên của mỗi bộ đệm, số lượng gói tin trung bình trong mỗi bộ đệm và khả năng gia tăng lưu lượng ở mỗi thời điểm chuyển tiếp một gói tin.

B. Cơ chế quản lý bộ đệm tích cực

Bài báo [10] nghiên cứu về tắc nghẽn trong mạng Ad hoc và chỉ ra hạn chế của TCP. Tác giả chỉ ra nguyên nhân chính của tắc nghẽn là do tỷ lệ lỗi bit cao trong kênh vô tuyến gây ra bởi các hiệu ứng vô tuyến như giao thoa, terminal ẩn, kết nối phụ thuộc vị trí, fading. Đặc biệt, tác giả cho rằng một cơ chế bộ đệm tích cực thích ứng (Adaptive AQM) có thể áp dụng cho CC. Tuy nhiên, cơ chế AQM sử dụng vẫn chủ yếu dựa vào cơ chế RED truyền thống, nên còn có tồn tại của AQM, mặc dù tác giả có sử dụng logic mờ với RED.

Các tác giả trong [1] đã khảo sát và phân tích khả năng sử dụng các cơ chế AQM như RED, FRED, BLUE, SFB, CHOCe cho mạng IoT. Bài báo đã chỉ ra những tồn tại khi áp dụng các cơ chế này, đòi hỏi phải có sự thay đổi.

Trong [13], một cải tiến IRED được đề xuất dựa trên việc dùng chiều dài bộ đệm tức thời để tính toán tỷ lệ bỏ gói tin trong mạng IoT. Về cơ bản, IRED vẫn sử dụng thuật toán tương tự RED, song có cách tính tỷ lệ mất gói khác. IRED tính chiều dài bộ đệm tức thời mỗi khi có gói tin đến, do vậy việc tính toán cần nhanh, độ phức tạp vẫn cao. Điều này cũng khó áp dụng cho nút mạng IoT có tài nguyên hạn chế.

Việc tính toán bộ đệm thích hợp được đề cập đến trong [7, 19], theo đó bộ đệm phù hợp có thể giúp giảm độ trễ, tỷ lệ mất gói và tăng hiệu suất sử dụng kênh truyền. Cơ chế quản lý bộ đệm truyền thống cần được thay đổi để dùng được trong mạng IoT [19].

C. Cơ chế TCP-CC và tương tự TCP-CC

Các tác giả trong [11, 19] phân tích các vấn đề nảy sinh khi áp dụng TCP cho mạng IoT, cụ thể về các vấn đề như cơ chế CC, kiểm soát mất gói liên quan đến nguyên nhân mất gói, cơ chế tương tác với thủ tục ARQ (Automatic Repeat Request) lớp Link, nén dữ liệu trong tiêu đề gói để giảm tiêu phí, cơ chế duy trì kết nối TCP trong điều kiện kênh lỗi, độ trễ với các kết nối ngắn, độ tin cậy, đa tuyến, cách tính RTO, độ phức tạp tính toán. Bài báo đã chỉ ra sự cần thiết phải thay đổi nhiều tham số cho TCP như kích thước cửa sổ, giảm phân mảnh gói tin IP qua 6LoWPAN, kích thước phân mảnh (segment) lớn nhất, giảm độ phức tạp, thay đổi RTO, v.v. Một số giao thức mới thay thế cho TCP đã được trình bày trong [17].

Các tác giả trong [17] nghiên cứu một số giao thức CC mới thay cho TCP-CC như ARC (Adaptive Rate Control), CODA (Congestion Detection and Avoidance Algorithm), CCF (Congestion Control and Fairness), FA (Fusion Algorithm), BGRA (Biased

Geographical Routing Algorithm), HTAP (Hierarchical Tree Alternative Path Algorithm).

Cơ chế CODA được cho là khá phù hợp cho mạng cảm biến không dây (WSN - Wireless Sensor Networks). CODA sử dụng trạng thái kênh để phát hiện tắc nghẽn và bản tin áp ngược (back-pressure) để ép nút gửi giảm tốc độ phát tin. Tuy nhiên, cơ chế này chủ yếu hiệu quả cho từng cặp nút và áp dụng ở tầng MAC. CODA có thể sử dụng vòng hồ từng chặng hoặc điều chỉnh cả cụm với vòng kín [34].

CoAP là giao thức phổ biến hiện nay được đề xuất cho truyền tin trong mạng IoT, song CoAP lại không có cơ chế CC [9]. CoAP là giao thức xếp chồng lên UDP, tận dụng ưu điểm của UDP trong so sánh với MQTT (xếp chồng lên TCP). CoAP có thể làm phát sinh thêm tắc nghẽn [28] do CoAP giả thiết độ tin cậy giữa hai đầu gửi và nhận tin cũng như sử dụng cơ chế CC truyền thống tương tự TCP [24].

Một số nghiên cứu tìm cách cải tiến cơ chế CC trong CoAP. Trong [28] đã chỉ ra một số cải tiến CoAP giúp cho CC hiệu quả hơn, ví dụ sử dụng Proxy ở rìa mạng IoT, sử dụng tham số Max Age ở Proxy, thay đổi giá trị RTO [8]. Tác giả trong [21] đề xuất điều chỉnh RTT để xây dựng cơ chế CC cho CoAP.

Trong [25], các tác giả đã nghiên cứu về khả năng áp dụng các cơ chế phát hiện và CC truyền thống cho mạng IoT. Bài báo đã chỉ ra mất gói có thể do lỗi vô tuyến nhiều hơn là do lý do khác. Các tác giả cho rằng, quan sát chiều dài bộ đệm có thể là cách đơn giản để phát hiện tắc nghẽn. Tuy nhiên, cần xem xét thêm độ trễ và thời gian chuyển tiếp gói tin tại nút mạng. Ngoài ra, trạng thái kênh truyền cũng là yếu tố quan trọng cần xem xét. Các tác giả cũng nhận thấy: việc áp dụng cùng kiểu CC cho tất cả các nút mạng trong mạng IoT là không phù hợp, việc điều chỉnh tốc độ nguồn phát không áp dụng được cho mọi ứng dụng.

Trong kiến trúc kết nối IoT với Cloud, gateway hay Proxy được đề xuất là trung gian giữa hai phân đoạn mạng. Giao thức CoCoA được đề xuất để kết nối phân đoạn mạng IoT với Cloud. Tuy nhiên, so với CoAP, thay đổi chủ yếu trong CoCoA chỉ là tham số RTO [8, 21, 2, 12]. RTO được tính toán linh hoạt hơn dựa theo hai tham số là *Strong RTT* và *Weak RTT*. Việc tính toán RTT vẫn phụ thuộc vào bản tin phản hồi ACK. Bài báo [2] đề xuất sử dụng biến số RTT thay vì việc dùng giá trị RTT cố định trong phiên bản cũ của CoAP.

Nghiên cứu trong [38] chỉ ra cơ chế AIMD không còn phù hợp cho mạng IoT do tạo ra sự biến thiên thông lượng lớn trong các luồng tin. Bài báo đề xuất cơ chế giảm kích thước cửa sổ có chọn lọc PAISMD (Selective Multiplicative Decrease). PAISMD có thể dùng cho tầng giao vận hoặc tầng ứng dụng. PAISMD tính toán tốc độ phát ổn định để đặt lại khi có tắc nghẽn.

Một giao thức TCP mới được đề xuất cho IoT trong [36] có tên là *Compound TCP*. Cơ chế này xác định nguyên nhân mất gói do tràn bộ đệm hoặc mất gói tin do xảy ra xung đột ở lớp MAC. Bài báo đề xuất giữ lại một phần bộ đệm để cân bằng thông lượng cho các luồng tin Compound TCP khi xuất hiện lỗi kênh vô tuyến. Compound TCP tìm cách phân biệt nguyên

nhân mất gói do lỗi kênh hay do tràn bộ đệm. Cơ chế CC của Compound TCP dựa vào hai cửa sổ: cửa sổ theo tỷ lệ mất gói và cửa sổ theo trễ. Cơ chế này chủ yếu được dùng cho mạng IoT sử dụng WiFi dùng chung điểm truy nhập (Access Point). Tuy nhiên, việc xác định phân bộ đệm để dành cũng khá phức tạp.

D. Điều khiển chống tắc nghẽn thông qua cân bằng tải và định tuyến

Các tác giả trong [5] khảo sát cơ chế CC ở tầng MAC và 6LoWPAN. Bài báo chỉ ra hai nguyên tắc cơ bản để giảm thiểu tắc nghẽn là: điều khiển lưu lượng và điều khiển tài nguyên. Tác giả nêu hạn chế của phương pháp định tuyến với RPL (Routing Protocol for Low-Power and Lossy Networks) cũng như một số giao thức truyền thông khác trong mạng WSN. Bài báo chỉ ra ưu điểm của cơ chế áp ngược (Back Pressure) trong kiểm soát tắc nghẽn. Cơ chế áp ngược cho mạng với 6LoWPAN được trình bày cụ thể hơn trong [9].

Trong [30], một cơ chế CC theo từng chặng (Hop-by-Hop) đã được đề xuất cho mạng WSN. Tác giả cho rằng tắc nghẽn có nguy cơ xảy ra ở gần phía các nút nhận hơn. Bài báo đề xuất một chỉ số tắc nghẽn (*Congestion Index*) được thay đổi qua từng chặng. Dựa vào chỉ số này, xu hướng tắc nghẽn được dự báo. Tốc độ phát được điều chỉnh tăng nếu chỉ số tắc nghẽn dương và ngược lại sẽ giảm nếu chỉ số âm. Tương tự cách này, một cơ chế CC phân tán được đề xuất trong [29], trong đó có xem xét đến tốc độ bình đẳng *min-max* và các thay đổi của môi trường.

Bài báo [42] khảo sát các cơ chế CC theo kiểu tập trung và kiểu phân tán cho mạng WSN. Trong cơ chế tập trung, nút nhận định kỳ kiểm tra xác suất tắc nghẽn và gửi bản tin thông báo tới nút mạng gửi tin tương ứng về nguy cơ tắc nghẽn. Tuyến đường khác đến đích

có thể được chọn trên cơ sở tính toán các tuyến khả thi để chọn khi xảy ra tắc nghẽn. Trong cơ chế phân tán, việc kiểm tra tắc nghẽn được thực hiện trên toàn bộ mạng WSN. Biểu hình theo cơ chế này là giao thức DAIPaS thực hiện quảng bá bản tin Hello tới toàn mạng WSN. Mỗi nút mạng được gán một nhãn tương ứng với trạng thái của nút. Trạng thái này được tính dựa vào kích thước bộ đệm hiện tại, kênh truyền và năng lượng còn lại. DAIPaS căn cứ vào các nhãn để xác định các tuyến truyền tin và xác định tuyến mới khi có tắc nghẽn xảy ra trên một tuyến.

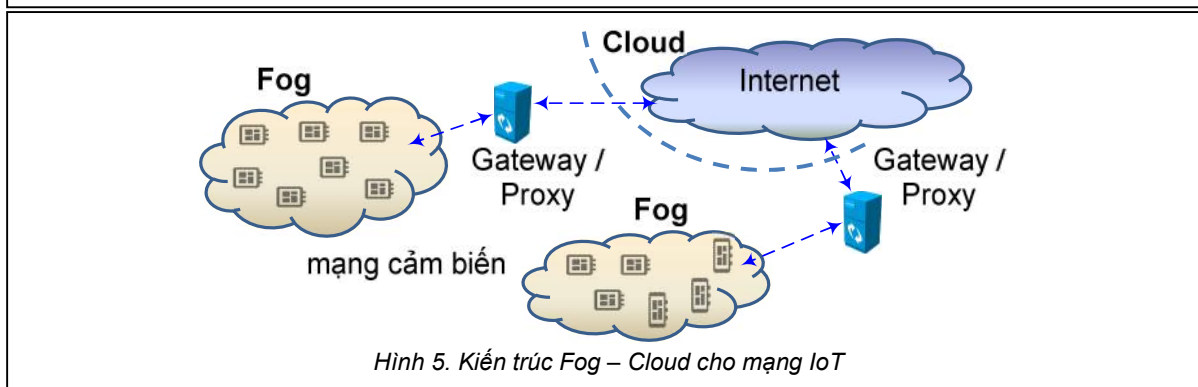
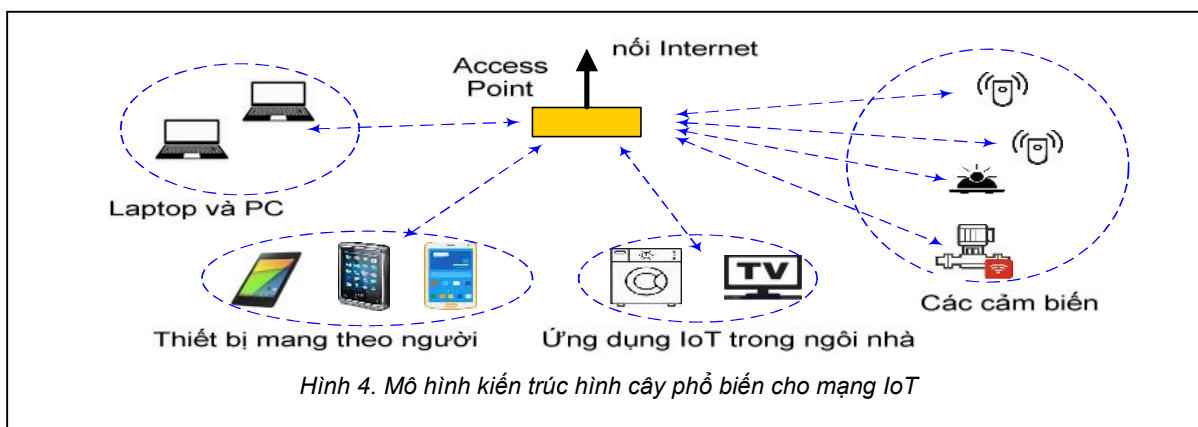
IV. GIẢI PHÁP ĐIỀU KHIỂN CHỐNG TẮC NGHẼN TRONG MẠNG IOT

Trong phần sau đây, bài báo sẽ trình bày một mô hình kiến trúc mạng IoT phổ biến và tổng hợp một số giải pháp điều khiển chống tắc nghẽn cho mạng IoT dựa trên mô hình này.

A. Kiến trúc mạng IoT phổ biến

Theo [35], cho đến nay vẫn chưa có một chuẩn mực thống nhất cho kiến trúc mạng IoT, mặc dù khá nhiều ứng dụng IoT đã được triển khai. Các nhà nghiên cứu đã đề xuất nhiều kiểu kiến trúc, song điển hình nhất là kiến trúc hình cây có phân lớp và kiến trúc dựa theo Fog-Cloud.

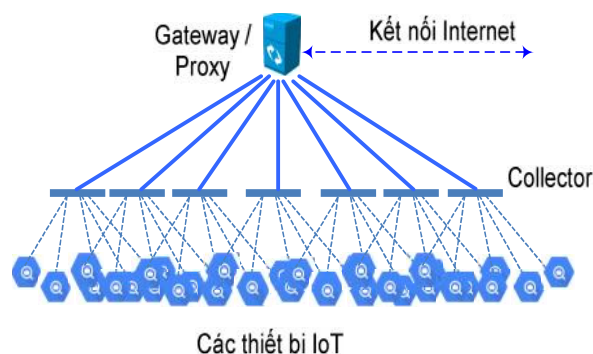
Kiến trúc hình cây có phân lớp phổ biến nhất được biểu thị trên hình 4 với một điểm truy nhập (Access Point) có thể dùng WiFi để kết nối ra Internet. Mỗi thiết bị IoT có thể sử dụng các giao thức thiết kế riêng cho mạng IoT như CoAP, MQTT để trao đổi thông tin với các máy tính trên Internet. Có ba phân lớp phổ biến trong ngăn xếp giao thức gồm: lớp cảm biến (các bộ cảm biến thu thập thông tin), lớp mạng để liên kết và lớp ứng dụng để chuyển tải các dịch vụ tới người dùng.



Kiến trúc Fog-Cloud được biểu thị trên hình 5. Fog là một khái niệm mới mô tả cụm mạng xử lý dữ liệu và sự kiện gần sát với các thiết bị IoT [27]. Một Fog có thể là một mạng cảm biến (WSN) theo kiến trúc hình cây, cụm, hoặc lưới. Các Fog kết nối với Cloud và Internet thông qua một Gateway hoặc Proxy.

Hình 6 là kiến trúc mạng IoT tổng quát theo hướng thông tin là trung tâm (Information-centric) hay dữ liệu là trung tâm (Data-centric). Đây là kiến trúc được cho là phù hợp nhất cho mạng IoT [33, 35, 40]. Các bộ thu thập (Collector) hay tổng hợp (Aggregator) có nhiệm vụ thu thập, lưu trữ, tổng hợp và chuyển tiếp dữ liệu theo xu hướng dữ liệu thường tập trung ở rìa mạng Internet [35, 22, 24, 23, 28, 33, 47].

Các cụm thiết bị IoT tạo thành các Fog, kết nối với Collector /Aggregator, tiếp đó liên kết với Gateway / Proxy để trao đổi thông tin với Internet.



Hình 6. Kiến trúc mạng IoT tổng quát theo hướng dữ liệu là trọng tâm (Data – Centric)

B. Giải pháp CC với bộ định trình có hiệu chỉnh

Các bộ định trình Fair Queueing (FQ) và điển hình là Weighted Fair Queueing (WFQ) được coi là cơ chế bảo đảm chất lượng dịch vụ phù hợp cho các luồng tin đầu vào với các đặc tính lưu lượng và yêu cầu chất lượng dịch vụ khác nhau (xem ví dụ [15]).

Như đã phân tích ở phần II và phần III, cơ chế WFQ trong mạng Internet truyền thông có thể được thay đổi mới để áp dụng cho mạng IoT. Theo sơ đồ trên hình 7, cơ chế định trình kiểu mới có thể gồm ba cấp: thiết bị IoT, Collector, Gateway/Proxy. Hình 7

mô tả kiến trúc tổng quát các bộ định trình WFQ được đề xuất cho ba cấp nêu trên.

Tại thiết bị IoT, có thể có nhiều luồng tin đến từ các bộ cảm biến khác nhau. Bộ định trình tại thiết bị IoT thực hiện sắp xếp các gói tin vào bộ đệm tại nút và chuyển tiếp chúng đến một Collector được xác định trước (qua định tuyến đối với nút Ad hoc). Bộ định trình tại Collector thu thập các luồng tin (tổng hợp) đến từ một cụm thiết bị IoT (một Fog), sắp xếp theo ưu tiên và chuyển tiếp đến Gateway. Bộ định trình Gateway tiếp nhận các luồng tin tổng hợp từ các Collector rồi chuyển tiếp đến Internet.

Nếu coi mỗi thiết bị IoT chỉ cung cấp một luồng tin có chung đặc tính, sơ đồ nêu trên hình 7 có thể thu về hai cấp định trình: tại Collector và tại Gateway.

Mặt khác, nếu chỉ phân loại lưu lượng dữ liệu do thiết bị IoT cung cấp thành có ưu tiên (critical data) và không ưu tiên (non-critical data), thì ta chỉ cần hai bộ đệm tách biệt tại Collector và hai tại Gateway. Mô hình WFQ có thay đổi áp dụng tại hai cấp với hai bộ đệm sẽ đơn giản hơn rất nhiều, phù hợp với năng lực xử lý và tài nguyên hạn chế. Ngoài ra, việc gán nhãn cho mỗi luồng tin thay vì gán nhãn cho mỗi gói tin giúp cho giảm thiểu được độ phức tạp.

Sự biến thiên của chiều dài mỗi bộ đệm biểu thị qua công thức (1) theo thời gian t như sau:

$$\frac{\partial q_i(t)}{\partial t} = \lambda_i(t) - F_{out}(t) \quad (1)$$

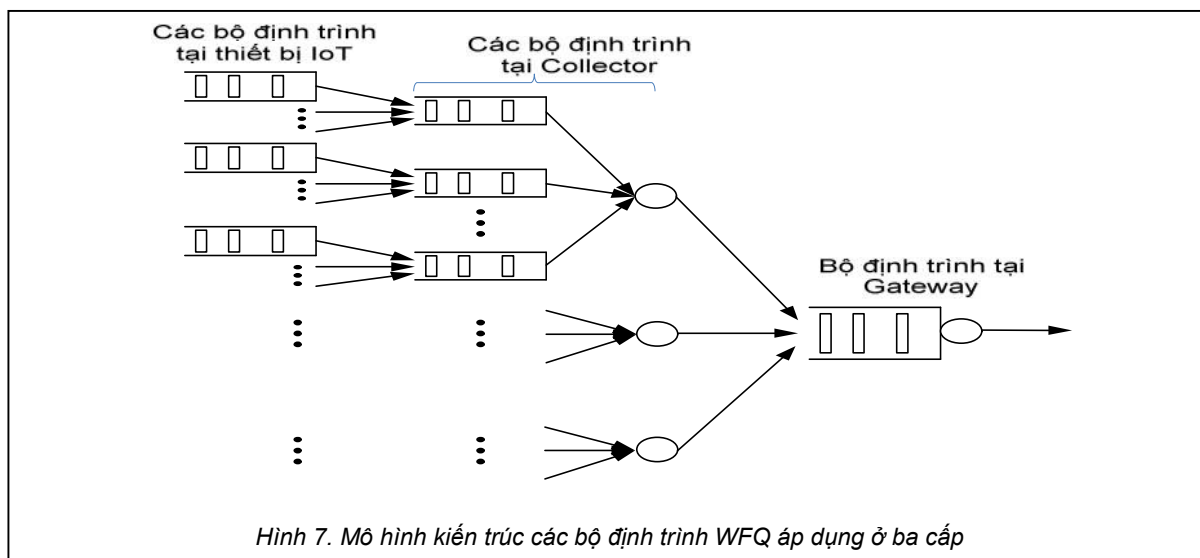
Trong đó, q_i là chiều dài bộ đệm i , λ_i là tốc độ luồng tin đến, F_{out} là tốc độ chuyển tiếp gói tin đi.

Thực hiện tích phân cho khoảng thời gian $[t_1, t_2]$, ta có tổng gói tin A lưu tại bộ đệm trong khoảng thời gian $[t_1, t_2]$ là:

$$A[t_1, t_2] = \delta + \lambda_i(t_2 - t_1) \quad (2)$$

Trong đó δ là số gói tin còn lại trong bộ đệm sau thời điểm t_2 , khi bộ định trình đã chuyển tiếp một lượng gói tin trong khoảng $[t_1, t_2]$. Công thức (2) giúp ích cho việc tính toán kích thước bộ đệm cần thiết.

Việc gán nhãn cho mỗi luồng tin có thể được thực hiện tương tự theo [15] như sau:



Hình 7. Mô hình kiến trúc các bộ định trình WFQ áp dụng ở ba cấp

$$\begin{aligned} S_i &= \max(T_i^k, F_i) \\ F_i &= S_i + L / \lambda_i \end{aligned} \quad (3)$$

Trong đó, S là nhãn bắt đầu, F là nhãn kết thúc của luồng tin i , T là thời điểm gói tin k đưa vào bộ đệm, L là kích thước gói tin k .

Khi bộ định trình chọn một gói tin trong số hai bộ đệm nêu trên, gói tin có nhãn F nhỏ hơn sẽ được chọn để chuyển tiếp. Điều này bảo đảm luồng tin có ưu tiên luôn được chuyển tiếp kịp thời.

Giả sử lượng gói tin gửi từ một thiết bị IoT không vượt quá B_m , ta có thể tính được kích thước bộ đệm của bộ định trình để không xảy ra mất gói tin như sau:

$$B_i \geq B_m + L_i (1 + \lambda_i / C_j) \quad (4)$$

Trong đó B_i là kích thước bộ đệm i của bộ định trình, C_j là tốc độ kênh chuyển tiếp gói tin của bộ định trình j .

Theo công thức (4), có thể bảo đảm không xảy ra tắc nghẽn tại Collector và Gateway do tràn bộ đệm. Do khuôn khổ có hạn, vấn đề phát hiện và xử lý lỗi kênh sẽ được xem xét ở một bài báo khác.

Độ trễ của mỗi gói tin k có thể được tính theo công thức như sau dựa theo phương pháp trong [15]:

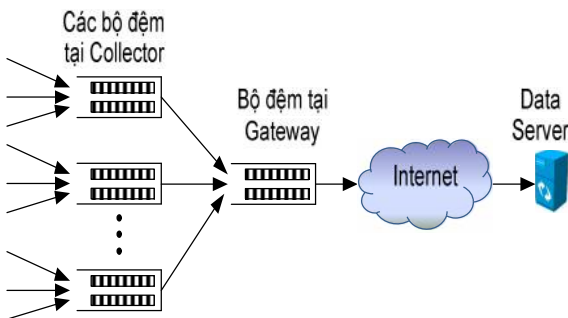
$$D \leq B_m / \lambda_i + L_i / C_j \quad (5)$$

Cách tính các tham số trên dùng được cho cả bộ định trình tại Collector và tại Gateway. Tổng hợp chung, ta có thể tính ra độ trễ khi chuyển tiếp gói tin từ một thiết bị IoT tới Gateway.

C. Giải pháp quản lý bộ đệm có hiệu chỉnh

Như đã phân tích ở phần II và phần III, các cơ chế quản lý bộ đệm tích cực trong mạng Internet truyền thống như RED, FRED, BLUE (xem ví dụ [15, 1]) có thể được thay đổi mới để áp dụng cho mạng IoT.

Tương tự hình 7, ta có thể chia thành hai bộ đệm ở mỗi cấp: một cho các luồng tin có ưu tiên, một không ưu tiên. Hình 8 biểu thị cơ chế quản lý bộ đệm ở hai cấp: Collector và Gateway.



Hình 8. Quản lý bộ đệm ở hai cấp

Cơ chế quản lý mỗi bộ đệm có sự khác nhau. Bộ đệm cho luồng tin có ưu tiên chỉ đánh dấu các gói tin và chỉ loại bỏ chúng tại Gateway trong một số tình huống tắc nghẽn nghiêm trọng.

Các vấn đề có thể phát sinh là cách phát hiện sớm nguy cơ tràn bộ đệm để loại bỏ các gói tin có mức ưu tiên thấp hơn. Có nhiều cách để phát hiện như dựa

vào: chiều dài bộ đệm, tốc độ gói tin đến, tốc độ gói tin đi, độ trễ của gói tin, dự đoán xác suất mất gói, thông báo tắc nghẽn từ mạng, v.v.

Cơ chế dựa vào chiều dài bộ đệm thực hiện tính toán chiều dài trung bình của mỗi bộ đệm, rồi so sánh với các mức ngưỡng để phát hiện nguy cơ tắc nghẽn. Chiều dài trung bình của bộ đệm Q_{ave} có thể được tính theo công thức:

$$Q_{ave} = (1 - \omega) Q_{ave} + \omega q_t \quad (6)$$

Trong đó q_t là chiều dài tức thời của bộ đệm, ω là một hệ số điều chỉnh có thể xác định qua thực nghiệm.

Như chỉ ra trong công thức (2), lượng gói tin tồn đọng trong bộ đệm liên quan mật thiết với tốc độ gói tin đến và tốc độ chuyển tiếp gói tin đi. Do vậy, tốc độ là một tham số có thể sử dụng trong tính toán chiều dài tức thời của bộ đệm một cách chính xác hơn.

Cơ chế dựa vào dự đoán xác suất mất gói thực hiện tính toán tỷ lệ mất gói tin thực tế và tỷ lệ sử dụng băng thông kênh truyền.

Xác suất mất gói cũng có thể ước tính thông qua tốc độ đến của luồng tin và tốc độ chuyển tiếp của bộ định trình cho các luồng tin. Ví dụ, xác suất mất gói tại bộ đệm Collector có thể tính như sau:

$$P = \max(0, 1 - \lambda_i / C_j) \quad (7)$$

Cơ chế dựa vào thông báo tắc nghẽn từ mạng có thể tận dụng thông tin phản hồi từ đầu cuối hoặc Gateway, hoặc qua dự đoán tỷ lệ chiếm dụng bộ đệm của Collector và Gateway cho toàn bộ các luồng tin.

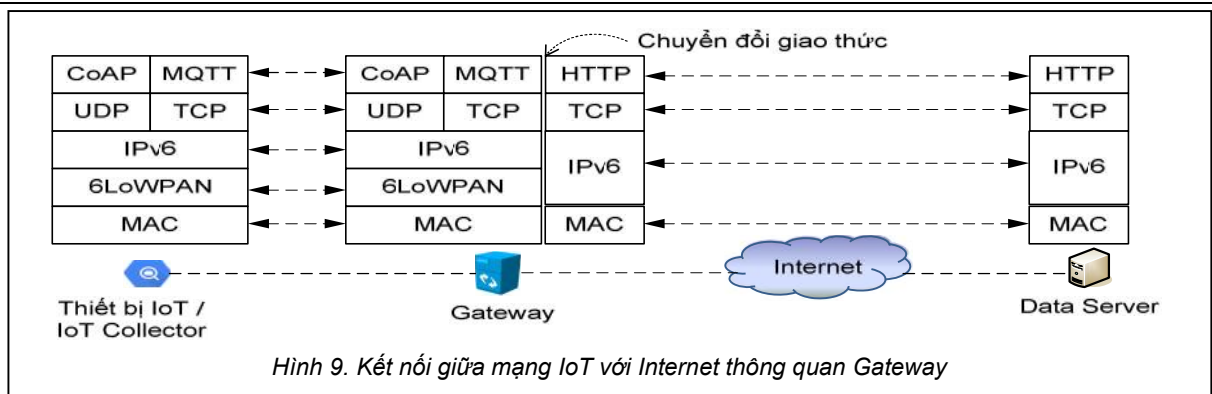
Tóm lại, các cơ chế dựa vào chiều dài bộ đệm, dự đoán xác suất mất gói, thông báo tắc nghẽn từ mạng, v.v. đều có thể hoạt động độc lập. Tuy nhiên, có thể thấy các cơ chế nêu trên đều sử dụng các tham số có sự liên quan đến nhau. Vì vậy, một giải pháp kết hợp các cơ chế nêu trên có thể sẽ hiệu quả hơn cho mạng IoT theo sơ đồ trên hình 6.

D. Giải pháp CC tương tự TCP có hiệu chỉnh

Giải pháp này theo nguyên lý cơ chế vòng kín, nghĩa là có thông tin phản hồi từ mạng hoặc nút nhận. Cơ chế có thể dựa trên cửa sổ (Window-based) như TCP hoặc dựa theo tốc độ (Rate-based). Trong phần tiếp theo, bài báo tập trung vào cơ chế dựa tốc độ với thông tin phản hồi tập trung ở Gateway.

Mạng IoT chủ yếu có kết nối mạng kiểu từng chặng (Hop-by-Hop) và cần có chuyển đổi giao thức ở Gateway. Hình 9 là sơ đồ chuyển đổi giao thức trong mạng IoT đang được sử dụng phổ biến hiện nay.

Như đã nêu ở phần II, các giao thức CoAP (Constrained Application Protocol), MQTT (Message Queuing Telemetry Transport) là hai giao thức được đề xuất cho mạng IoT. CoAP nằm phía trên giao thức UDP, còn MQTT nằm phía trên TCP. CoAP được thiết kế để chuyển đổi sang HTTP dễ dàng, tích hợp đơn giản với các hệ thống web. MQTT đòi hỏi kết nối TCP do cần có độ tin cậy và thứ tự bản tin chuyển đi. Tuy nhiên, như đã chỉ ra ở phần II, các giao thức này vẫn còn hạn chế về điều khiển chống tắc nghẽn và không phù hợp cho mạng IoT. Sau đây là đề xuất về một số thay đổi.



Theo sơ đồ mô tả trên hình 9, kết nối từ một thiết bị IoT đến Data server sẽ gồm hai đoạn: kết nối từ thiết bị IoT (qua Collector) tới Gateway và kết nối từ Gateway tới Data server. Ta có thể đưa ra giải pháp cho một lớp giao thức tương tự TCP với cơ chế CC cho kết nối phân đoạn nêu trên với những nguyên tắc thay đổi chủ yếu như sau.

- Tốc độ phát tối đa R_m được tính toán cho thiết bị IoT.
- Một chùm gói tin đầu tiên của kết nối được gửi tới Gateway để thăm dò tình trạng mạng với tốc độ $R_{init} < R_m$. Khi nhận được chùm gói tin này, Gateway đo thời gian trễ gói tin, dự đoán băng thông kênh truyền, kiểm tra xác suất mất gói tin, kiểm tra bộ đệm tại Gateway và thông tin tắc nghẽn (nếu có) từ mạng / đầu nhận. Tiếp đó Gateway sẽ gửi phản hồi về cho thiết bị IoT.
- Bản tin phản hồi chứa chỉ số tắc nghẽn (thông tin về tỷ lệ mất gói, độ trễ gói tin, tình trạng kết nối tới Internet, lỗi kênh (nếu có), băng thông dự đoán. Trên cơ sở đó, giao thức bên thiết bị IoT tính toán tốc độ phát phù hợp.
- Thiết bị IoT/Collector tiếp tục chuyển tiếp các gói tin qua Gateway tới Data server với tốc độ phát đã tính toán nếu chỉ số tắc nghẽn = 0.
- Nếu có bản tin phản hồi về chỉ số tắc nghẽn khác 0, tốc độ phát được tính lại và đặt ở mức phù hợp với chỉ số tắc nghẽn.
- Thời gian RTO được tính phù hợp với độ trễ gói tin và đủ để thiết bị IoT nhận biết có lỗi kênh truyền xảy ra. Tương tự tại Gateway, RTT và RTO được tính toán, dự đoán cho toàn tuyến.

Nguyên lý CC nêu trên thực chất gần giống với cơ chế của TFRC, song có một số điểm khác biệt là:

- Kết nối đầu cuối tới đầu cuối được chia làm hai đoạn ở Gateway, giúp tách biệt được các nguyên nhân tắc nghẽn sát thực tế hơn.
- Chùm gói tin thăm dò được sử dụng để tính tốc độ phát gói tin phù hợp ban đầu. R_m được tính toán để hạn chế tốc độ phát tối đa.
- Bản tin phản hồi được thực hiện bởi Gateway chứa chỉ số tắc nghẽn phục vụ cho việc điều khiển chống tắc nghẽn, có xem xét đến khả năng mất gói tin do lỗi kênh truyền vô tuyến.
- Điều chỉnh tăng giảm nhanh hơn AIMD do có phản hồi sớm trực tiếp từ Gateway.

Nhược điểm của giải pháp này là phải tách kết nối làm hai đoạn, song điều này hoàn toàn phù hợp với tính chất kết nối Hop-by-Hop của mạng IoT và phù hợp với kiến trúc Data – Centric có sử dụng Gateway đã nêu ở phần IV.A.

E. Nhận xét

Trong các phần trên, bài báo đã tổng hợp một số hướng giải pháp điều khiển chống tắc nghẽn cho mạng IoT với các đề xuất hiệu chỉnh cho ba nhóm cơ chế điều khiển. Các giải pháp trên căn cứ vào các đặc điểm riêng của mạng IoT và kết quả phân tích các ưu nhược điểm của các công trình nghiên cứu liên quan. Có thể nhận thấy các cơ chế tái định tuyến (chọn đường khác ít tắc hơn) và cân bằng tải khó phù hợp do phải mất nhiều thời gian tìm đường mới. Tái định tuyến chủ yếu chỉ cho đoạn kết nối từ Gateway tới đầu cuối. Do vậy, việc hiệu chỉnh các cơ chế định trình, quản lý bộ đệm và cơ chế giao thức CC phân đoạn như đã đề xuất là những hướng giải pháp CC khả thi cho mạng IoT.

Trong khuôn khổ có hạn, bài báo mới chỉ trình bày những nguyên tắc cơ bản nhất của ba nhóm giải pháp. Tuy nhiên, theo các giải pháp trong bài, ta có thể thiết kế các cơ chế điều khiển chống tắc nghẽn cụ thể cho mạng IoT.

V. KẾT LUẬN

Điều khiển chống tắc nghẽn là một yêu cầu cần thiết đối với mạng IoT do sự đa dạng về ứng dụng và thiết bị, sự đa dạng và những hạn chế của thiết bị IoT cũng như môi trường mạng IoT. Tuy nhiên, hiện vẫn chưa có cơ chế điều khiển chống tắc nghẽn phù hợp cho mạng IoT. Một phần do đây là lĩnh vực mới, một phần do những khó khăn trong môi trường mạng IoT. Các cơ chế CC cho mạng Internet truyền thống không còn phù hợp, không thể áp dụng cho mạng IoT. Chính vì vậy, nghiên cứu cơ chế CC phù hợp cho mạng IoT là một nhu cầu thực tế.

Bài báo đã trình bày và phân tích các điểm khác biệt trong điều khiển chống tắc nghẽn giữa mạng IoT và mạng Internet truyền thống. Qua khảo sát các công trình nghiên cứu liên quan, bài báo đã phân tích ưu nhược điểm của các giải pháp CC đã được đề xuất, chỉ ra những điểm còn tồn tại và những khó khăn thách thức khi áp dụng các cơ chế CC sẵn có trong môi trường mạng IoT.

Trong phần IV, bài báo đã đưa ra một kiến trúc tổng thể cho mô hình mạng và tổng hợp một số hướng giải pháp điều khiển chống tắc nghẽn cho mạng IoT với các đề xuất thay đổi về cơ chế điều khiển trong ba

nhóm giải pháp. Cụ thể là: giải pháp CC với cơ chế định trình có thay đổi theo phân cấp, đặc tính luồng tin; giải pháp quản lý bộ đệm tích cực có phân cấp với các cách phát hiện sớm tắc nghẽn; giải pháp CC tương tự TCP với hai phân đoạn mạng và một số cải tiến. Từ những giải pháp tổng thể đã nêu có thể xây dựng các cơ chế cụ thể. Đó là những hướng nghiên cứu phát triển tiếp trong thời gian tới.

LỜI CẢM ƠN

Bài báo này được thực hiện trong khuôn khổ đề tài ASEAN IVO “A Hybrid Security Framework for IoT Networks” của Viện NICT (Nhật Bản) và đề tài cấp Nhà nước mã số KC.01.08/16-20 của Bộ KH&CN. Các tác giả xin trân trọng cảm ơn các cơ quan đã tài trợ cho nhóm nghiên cứu.

TÀI LIỆU THAM KHẢO

- [1] G.F. Ali Ahammed, R. Banu, Analysing the Performance of Active Queue Management Algorithms, Inter. Journal of Computer Networks & Communications (IJCNC), Vol.2, No.2, March 2010, pp.1-19.
- [2] E. Ancillotti, S. Bolettrieri, R. Bruno, RTT-based Congestion Control for the Internet of Things. Proc. of 16th IFIP WG 6.2, Internl Conference, WWIC 2018, Boston, USA, June 18-20, 2018, pp.3-15.
- [3] A. Al-Fuqaha, M. Guizani, M. Mohammadi, et.al, Internet of Things: A Survey on Enabling Technologies, Protocols, and Applications. IEEE Communication Survey & Tutorials, Vol. 17, No.4, Fourth Quarter 2015, pp.2347-2376.
- [4] M. Ahmad, M. Hussain, B. Abbas, et.al. End-to-End Loss Based TCP Congestion Control Mechanism as a Secured Communication Technology for Smart Healthcare Enterprises. IEEE Access Journal, Vol.6. <arch 2018, pp.11641-11656.
- [5] HA. Al-kashoash, H. Kharrufa, Y. Al-Nidawi, A.H. Kemp. Congestion control in wireless sensor and 6LoWPAN networks: toward the Internet of Things. Wireless Networks, <https://doi.org/10.1007/s11276-018-1743-y>, 2018.
- [6] M. Alaslani, B. Shihada, Intelligent Edge: an Instantaneous Detection of IoT Traffic Load. Proc of IEEE Internl Conference on Communications (ICC), 2-24 May 2018, Kansas City, USA. 2018.
- [7] P. Bh M. Arora, S. Upadhyaya, N. Kashyap. Flexible congestion control using fuzzy logic for Wireless Sensor Networks. Inter. Journal of Computer Sciences and Engineering. Vol.6(5), May 2018, pp.492-499.
- [8] A. Betzler, C. Gomez, I. Demirkol, M. Kovatsch. Congestion Control for CoAP Cloud Services. Proc of IEEE Conference on Emerging Technology and Factory Automation (ETFA), 16-19 Sept. 2014.
- [9] A.P. Castellani. Design, implementation and experimentation of a protocol stack for the Internet of Things. PhD thesis, University of Padova. July 2012.
- [10] E. Natschek, A.B. Jantan, S. Khatun, S. Subramaniam. Fuzzy Active Queue Management for Congestion Control in Wireless Ad-Hoc. Inter. Arab Journal of Information Technology, Vol.4, No.1, Jan. 2007, pp.50-59.
- [11] C. Gomez, A. Arcia-Moret, J. Crowcroft. TCP in the Internet of Things: from ostracism to prominence. IEEE Journal of Internet Computing, Vol.22, Issue 1, Jan/Feb 2018, pp.29-41.
- [12] HM. Hasan, AI, Ahmed. A Comparative Analysis for Congestion Mechanism in CoAP and COCOA. Engineering and Technology Journal, Vol36, Part A, No.8, 2018, pp.867-877.
- [13] J. Huang, Q. Duan. Modeling and analysis on congestion control in the Internet of Things. Proc of IEEE Internl Conference on Communications (ICC), 10-14 June 2014, pp.434-437.
- [14] R. Hassan, AM. Jubair, K. Azmi, A. Bakar. Adaptive Congestion Control Mechanism in CoAP Application Protocol For Internet of Things (IoT). Proc of Internl Conference on Signal Processing and Communications (ICSC), 26-28 Dec. 2016.
- [15] Dang Hai Hoang. Quality of Service Control in the Mobile Wireless Environments. PeterLang Publisher, Frankfurt/M-Berlin-Bern-BruxellesNewYork-Oxford-Wien, US-ISBN 0-8204-6402-3, 2003.
- [16] CH. Phuong, HD.Hai. Điều khiển chống tắc nghẽn trong các mạng NGN-toàn IP. Tạp chí Bưu chính Viễn thông & CNTT. Chuyên san Các công trình nghiên cứu- Triển khai VT và CNTT. Vol.2, No.3. 2007, pp.30-42.
- [17] K. Khard, B. Sharma, TC. Aseri. Reliable and Congestion Control Protocols for Wireless Sensor Networks. Internl Journal of Engineering and Technology Innovation, Vol6, No.1, 2016, pp.68-78.
- [18] G. Kokkonis, KE. Psannis, M.Roumeliotis, et.al. Transferring Wireless High Update Rate Supermedia Streams Over IoT. Springer: New Advances in the Internet of Things. pp.93-103.
- [19] J.A. Khan, M. Shahzad, AR. Butt. Sizing Bu ers of IoT Edge Routers. Proc of 1st Internl Workshop on Edge Systems, Analytics and Networking, EdgeSys'18, 10-15 June 2018, Munic, Germany, pp.55-60.
- [20] L. Li, Y. He, X. Li. Congestion Control Technology of Internet of Things. Proc of Internl Conference on Electronic Information Technology and Intellectualization (ICEITI 2016).
- [21] JJ. Lee, KT. Kim, HY. Youn. Enhancement of congestion control of Constrained Application Protocol/ Congestion Control/Advanced for Internet of Things environment. Internl Journal of Distributed Sensor Networks. Journal IJDSN, Vol.12 (11) 2016.
- [22] S. Li,. Future IoT Network Architecture and Applications in Mobile Sensing. PhD thesis, the State University of New Jersey. Oct. 2018.
- [23] S. Li, NZ. Zhang, L. Kong, et.al. Joint Admission Control and Resource Allocation in Edge Computing for Internet of Things. Journal IEEE Network, Jan/Feb 2018, pp.72-79.
- [24] J. Misic, M. Z.Ali, V.B. Misic. Architecture for IoT domain with CoAP observe feature. IEEE Internet of Things Journal, Vol5. Iss. 2, Apr.2018, pp.1196-1205.
- [25] AK Mohamed, D.Djenouri, B.O. Jalel, N. Badache. Congestion Control Protocols in Wireless Sensor Networks: A Survey. IEEE Communications Surveys & Tutorials. Vol16, Iss. 3, 3rd Quarter 2014, pp.1369-1390.
- [26] A. Mishra. Performance Analysis of TCP Tahoe, Reno and New Reno for Scalable IoT Network Clusters in QualNet® Network Simulator. Internl Journal of

- Computer Sciences and Engineering, Vol.6, Issue 8, Aug. 2018, pp.347-355.
- [27] TS. Mohammed, OF. Khan, AS. Ibrahim, R. Mamlook. Fog Computing-Based Model for Mitigation of Traffic Congestion. Proc of 8th Internl Conference on Intelligent Systems, Modeling and Simulation. May 2018.
- [28] J. Misic, VB. Misic. Lightweight data streaming from IoT devices. Proc of IEEE Internl Conference on Communications (ICC 2018), 20-24 May 2018.
- [29] A. Mozo, JL. Lopez-Presa, AF. Anta. A distributed and quiescent max-min fair algorithm for network congestion control. ACM Journal Expert Systems with Applications, Vol.91, Issue C, Jan 2018, pp.492-512. Internl Journal of Computer Sciences and Engineering, Vol.6, Issue 8, Aug. 2018, pp.347-355.
- [30] M. Khedkar, RA. Vatti. Congestion Control in High Density Wireless Personal Area Networks. Internl Journal of Advanced Research (2016), Vol4, Issue 7, 2016, pp.1781-1788.
- [31] N. Mishra, LP. Verma, PK. Srivastava, A. Gupta. An Analysis of IoT Congestion Control Policies. Proc of Internl Conference on Computational Intelligence and Data Science (ICCIDIS 2018). Vol132, 2018, pp.444-450.
- [32] F. Ouakasse, S. Rakkak. An Adaptive Solution for Congestion Control in CoAP-based Group Communications. Internl Journal of Advanced Computer Science and Applications, Vol.8, No.6, 2017, pp.234-239.
- [33] SMA. Oteafy, HS. Hassanein. IoT in the Fog: A Roadmap for Data-Centric IoT Development. IEEE Communications Magazine, Mar. 2018, pp.157-163.
- [34] JH. Park, JH. Kim, SK. Lee. A Study on the Enhanced Congestion Control Mechanism for Multimedia Traffic in Sensor Networks. Internl Journal of Multimedia and Ubiquitous Engineering, Vol.10, No.8, 2015, pp.391-400.
- [35] P. Sethi, SR. Sarangi. Internet of Things: Architectures, Protocols, and Applications. Journal of Electrical and Computer Engineering, Vol.2017, 25 pages.
- [36] S. R. Pokhrel, C. Williamson. Modeling Compound TCP over WiFi for IoT. IEEE/ACM Transactions on Networkings, Vol.26, Issue 2, Apr. 2018, pp.864-878.
- [37] AA. Rezaee, F. Pasandideh. A Fuzzy Congestion Control Protocol Based on Active Queue Management in Wireless Sensor Networks with Medical Applications. ACM Journal Wireless Personal Communications, Vol39, Issue 1, Jan 2018, pp.815-842.
- [38] R.K. Lam, KC. Chen. Congestion Control for M2M Traffic with Heterogeneous Throughput Demands. Proc of IEEE Wireless Communications and Networking Conference (WCNC) 2013, pp.1452-1457.
- [39] Y.N. Reddy, PVS. Srinivas. A Combined TCP-friendly Rate control with WFQ Approach for Congestion Control for MANET. Internl Journal Computer Network and Information Security, Vol.6, 2018, pp.52-59.
- [40] T. Shreedhar, SK. Kaul, RD. Yates. ACP: Age Control Protocol for Minimizing Age of Information over the Internet. MobiCom'18, 29 Oct. 2018, pp.699-701.
- [41] R. Sharma, N. Kumar, T. Srinivas. Markov Chain based Priority Queueing Model for Packet Scheduling and Bandwidth Allocation. Proc of Internl Conference on Ubiquitous Communications and Network Computing (UBICNET 2017), 24 Dec. 2017, pp.91-103.
- [42] SA. Shah, B. Nazir, I.A. Khan. Congestion control algorithms in wireless sensor networks: Trends and opportunities. J King Saud Univ. Computer Information Science, Vol. 29(3), July 2017, pp. 236-245.
- [43] D. Shen, W. Yan, Y. Peng, et.al. Congestion Control and Traffic Scheduling for Collaborative Crowdsourcing in SDN Enabled Mobile Wireless Networks. Journal of Wireless Communications and Mobile Computing, Vol. 2018, 11 pages.
- [44] N. Thrimoorthy, T. Anuradha. Congestion Control in Wireless Sensor Network based on Predicted Sensor Position on Movement for Body Area Network Applications. Internl Journal of Computer Applications, Vol. 161, No.5, Mar. 2017, pp.19-23.
- [45] S. Thombre, RU. Islam, K. Andersson, MS. Hossain. IP based Wireless Sensor Networks: Performance Analysis using Simulations and Experiments. Journal of Wireless Mobile Networks, Ubiquitous Computing and Dependable Applications, Vol.7, No.3, Sept 2016, pp.53-76.
- [46] W. Shang, Y. Yu, R. Droms. Challenges in IoT Networking via TCP/IP Architecture. NDN Technical Report NDN-0038-2016.
- [47] AM. Yang, XL. Yang, JC. Chang, et.al. Research on a Fusion Scheme of Cellular Network and Wireless Sensor for Cyber Physical Social Systems. IEEE Access, Vol. 6, 16 March 2018, pp. 18786-18794.

SOLUTIONS FOR CONGESTION CONTROL IN IoT NETWORKS

Abstract: Network congestion is a basic problem that exists in every network. By the increased growth of the Internet of Things Networks (IoT Networks), the number of connected devices are more increased and the risk of network congestion becomes more serious. The IoT network environment has many features that are different from the conventional Internet. Thus, the network congestion control mechanisms of the conventional Internet could not be directly applied for IoT networks, calling for the need of suitable modifications in order to guarantee throughput and communication quality. This paper analyses the differences in network congestion control between IoT networks and the conventional Internet. We survey and analyse some related work. Based on analysing the principles of network congestion control and the special features of IoT networks, the paper synthesises three solutions approaches for congestion control in IoT networks with some modification proposals.

Keywords: IoT networks, Network congestion, Network congestion control, Scheduling, Active Buffer Management.



Hoàng Đăng Hải, TS. (1999), TSKH. (2002) tại CHLB Đức, PGS (2009). Hiện đang công tác tại Học viện Công nghệ Bưu chính Viễn thông. Lĩnh vực nghiên cứu: Mạng và hệ thống thông tin, các giao thức truyền thông, chất lượng dịch vụ, mạng IoT, an toàn thông tin.



Phạm Thiếu Nga, TS. (2000) tại CHLB Đức.

Hiện đang là giảng viên chính tại Khoa Công nghệ thông tin, Đại học Xây dựng, Hà Nội. Lĩnh vực nghiên cứu: logic mờ, điều khiển mờ, mạng và hệ thống thông tin, mạng WSN, mạng IoT, hệ trợ giúp quyết định, hệ chuyên gia.



Lê Thị Thuý Dương, ThS. tại Đại học Bách Khoa Hà Nội.

Hiện đang là giảng viên tại Khoa Công nghệ thông tin, Trường Đại học Xây dựng, Hà Nội. Lĩnh vực nghiên cứu: Mạng viễn thông, mạng cảm biến không dây, mạng IoT, điều khiển chống tắc nghẽn, hiệu năng mạng.