ĐẠI HỌC QUỐC GIA HÀ NỘI
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ

NGÔ THỊ BẾN

CÁC KẾ HOẠCH QUẢN LÝ HÀNG ĐỢI ĐỘNG BLUE CHO TRUYỀN THÔNG ĐA PHƯƠNG TIỆN

LUẬN VĂN THẠC SỸ NGÀNH CÔNG NGHỆ THÔNG TIN

HÀ NỘI – 2016

ĐẠI HỌC QUỐC GIA HÀ NỘI
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ

NGÔ THỊ BẾN

CÁC KẾ HOẠCH QUẢN LÝ HÀNG ĐỢI ĐỘNG BLUE CHO TRUYỀN THÔNG ĐA PHƯƠNG TIỆN

Ngành: Công nghệ thông tin
Chuyên ngành: Truyền dữ liệu và Mạng máy tính
Mã số: Chuyên ngành đào tạo thí điểm

LUẬN VĂN THẠC SỸ NGÀNH CÔNG NGHỆ THÔNG TIN

NGƯỜI HƯỚNG DẪN KHOA HỌC: PGS.TS. Nguyễn Đình Việt

HÀ NỘI – 2016

LỜI CAM ĐOAN

Tôi xin cam đoan Luận văn "**Các kế hoạch quản lý hàng đợi động BLUE cho truyền thông đa phương tiện"** là sản phẩm của riêng cá nhân tôi, không sao chép lại của người khác, trừ những kiến thức tham khảo từ những nguồn tài liệu đã được chỉ rõ. Các tài liệu tham khảo đều có xuất xứ rõ ràng và được trích dẫn đúng quy cách, các kết quả, số liệu nêu trong luận văn là trung thực, một phần đã được công bố trên các tập trí khoa học chuyên ngành, phần còn lại chưa từng được công bố trên bất kì công trình luận văn nào khác.

Hà Nội, tháng 11 năm 2016

Học Viên

Ngô Thị Bến

LỜI CẢM ƠN

Tôi xin bầy tỏ lòng kính trọng và biết ơn sâu sắc tới thầy giáo hướng dẫn - PGS.TS. Nguyễn Đình Việt, người đã định hướng nghiên cứu, trực tiếp hướng dẫn, chỉ dẫn cho tôi phương pháp luận thực hiện luận văn. Thầy đã mang những kiến thức, kinh nghiệm, lòng nhiệt huyết tận tình hướng dẫn cho tôi trong suốt thời gian thực hiện luận văn.

Tôi xin bày tỏ lòng biết ơn sâu sắc tới các thầy cô giáo đã giảng dạy truyền thụ kiến thức cho tôi trong quá trình học tập tại trường Đại học Công nghệ - Đại học Quốc Gia Hà Nội.

Tôi xin châm thành cảm ơn các đồng nghiệp, bạn bè, gia đình đã động viên và tạo mọi điều kiện thuận lợi giúp tôi hoàn thành luận văn.

Hà Nội, tháng 11 năm 2016

Học Viên

Ngô Thị Bến

MỤC LỤC

DANH MỤC CÁC KÝ HIỆU, CHỮ VIẾT TẮT

Từ viết tắt	Nghĩa tiếng Anh	Nghĩa tiếng Việt
AIMD	Additive Increase Mulitplicative Decrease	Tăng theo cấp số cộng, giảm theo cấp số nhân
AQM	Active Queue Management	Quản lý hàng đợi tích cực
BA	Behavior Aggregation	Kết hợp hành vi
CLS	Controlled Load Service	Dịch vụ có tải được điều khiển
CQ	Custom Queue	Hàng đợi tự điều chỉnh
DiffServ	Differentiated Services	Các dịch vụ được phân loại
DNS	Domain Name System	Hệ thống tên miền
DSCP	Differentiated Service Code Point	Điểm mã dịch vụ phân loại
ECN	Explicit Congestion Notification	Thông báo nghẽn cụ thể
EWMA	Exponential Weighted Moving Average	Trung bình dịch chuyển có trọng số tăng theo hàm mũ
FIFO	First In First Out	Hàng đợi theo nguyên tắc vào trước ra trước
FTP	File Transfer Protocol	Giao thức truyền tệp tin
GS	Guaranteed Service	Dịch vụ được đảm bảo
IETF	Internet Engineering Task Force	Tổ chức đưa ra các khuyến nghị, định hướng phát triển mạng Internet
IntServ	Integrated Services	Các dịch vụ tích hợp
IP	Internet Protocol	Giao thức Internet
IPTV	Internet Protocol TeleVision	Truyền hình sử dụng giao thức IP
JPEG	Joint Photographic Expert Group	Tổ chức nghiên cứu về các chuẩn nén ảnh
MIMD	Multiplicative Increase Multiplicative Decrease	Tăng theo cấp số nhân, giảm theo cấp số nhân
MPEG	Moving Picture Expert Group	Nhóm các chuyên gia về hình ảnh chuyển động
PHB	Per Hop Behavior	Hành vi theo từng chặng
PQ	Priority Queue	Hàng đợi ưu tiên
QoS	Quality of Service	Chất lượng dịch vụ
RSVP	Resource Reservation Protocol	Giao thức dành trước tài nguyên
RTS	Real-Time System	Hệ thống thời gian thực
SLA	Service Level Agreement	Thỏa thuận mức dịch vụ
TC	Traffic Class	Phân lớp lưu lượng
TCA	Traffic Condition Ageement	Thỏa thuận điều kiện lưu lượng
TCP	Transmission Control Protocol	Giao thức điều khiển truyền dẫn
ToS	Type of Service	Loại dịch vụ
UDP	User Datagram Protocol	Giao thức truyền bản tin của người dùng
VoIP	Voice over IP	Thoại sử dụng giao thức IP

DANH MỤC CÁC BẢNG

DANH MỤC HÌNH VẼ

MỞ ĐẦU

Internet là một hệ thống kết nối mạng toàn cầu đảm bảo liên thông giữa các hệ thống máy tính và thiết bị trên diện rộng. Internet ngày càng phát triển không chỉ về số lượng kết nối mà còn sự đa dạng của các lớp ứng dụng, các dữ liệu được truyền đi không chỉ đơn thuần là dạng văn bản đơn giản, mà là dữ liệu đa phương tiện bao gồm cả âm thanh, hình ảnh tĩnh, động, … Các ứng dụng đa phương tiện phổ biến có thể kể đến như điện thoại qua mạng (Internet telephony), hội thảo trực tuyến (video conferencing), xem video theo yêu cầu (video on demand),.. đang ngày càng được sử dụng rộng rãi. Có thể nói mạng Internet đã làm một cuộc cách mạng thay đổi nhiều khía cạnh trong cuộc sống của chúng ta. Hơn nữa mạng Internet còn rẻ hơn nhiều so với các loại hình dịch vụ khác, do đó nó được sử dụng rộng khắp ở mọi quốc gia trên thế giới. Cùng với sự bùng nổ về nhu cầu sử dụng Internet và sự gia tăng của lưu lượng thông tin kéo theo vấn đề xảy ra tắc nghẽn trên Internet là không thể tránh khỏi. Vì vậy, để đảm bảo thông suốt đường truyền, kiểm soát tắc nghẽn tại nút mạng đóng một vai trò rất quan trọng cho Internet hoạt động hiệu quả và tin cậy với người sử dụng. Mô hình cung cấp dịch vụ kiểu cố gắng tối đa (best-effort) của Internet truyền thống đã không đủ để đáp ứng với những yêu cầu về chất lượng dịch vụ (QoS) khi có sự bùng nổ các luồng dữ liệu tham gia mạng và làm cho các nút mạng trung tâm nhanh chóng bị tắc nghẽn. Khi mạng IP ra đời đã thoả mãn được các yêu cầu cả về kỹ thuật lẫn chất lượng dịch vụ. Tuy nhiên để nâng cao chất lượng dịch vụ, đáp ứng được các yêu cầu của người sử dụng là một vấn đề thực sự khó khăn cho các nhà quản lý mạng, đặc biệt là trong hoàn cảnh hiện nay khi các luồng thông tin ngày càng đa dạng về chủng loại, đặc tính, mà yêu cầu chất lượng sử dụng thông tin thì ngày càng khắt khe. Việc yêu cầu chất lượng dịch vụ của người sử dụng cũng tạo ra sự cạnh tranh khắc nghiệt giữa các nhà cung cấp dịch vụ, yêu cầu các nhà cung cấp dịch vụ phải tìm ra các giải pháp mới để nâng cao chất lượng dịch vụ và tăng doanh thu cho mình.

Có nhiều hướng nghiên cứu để cải thiện chất lượng truyền dữ liệu đa phương tiện qua mạng, nhưng đối với một số kỹ thuật truyền thống chúng có hạn chế là gây ra gánh nặng truyền tải đối với âm thanh/hình ảnh video, do đó làm tiêu tốn thêm tài nguyên băng thông mạng. Khi có quá nhiều gói tin được đưa vào mạng, sẽ làm cho hiệu năng của mạng giảm đi vì các nút mạng không còn đủ khả năng lưu trữ, xử lý, truyền đi, chúng bắt đầu bị mất các gói tin dẫn đến sự tắc nghẽn trong mạng máy tính. Các nhà xây dựng mạng đã khéo léo đưa ra các mô hình mạng mới như mô hình mạng dịch vụ phân loại DiffServ và mạng dịch vụ tích hợp IntServ đồng thời kết hợp các mô hình mạng với nhau để lợi dụng ưu điểm của từng mạng và hạn chế nhược điểm của chúng. Bên cạnh đó các nhà thiết kế còn đi sâu vào tìm hiểu và thiết kế các phương pháp quản lý, giám sát các tiến trình truyền tin ngay bên trong bản thân của các thành phần nhỏ của mạng như router, chuyển mạch,… Điển hình các router được thiết kế theo cấu trúc CQS đã phần nào đơn giản hoá việc truyền tin và nâng cao chất lượng dịch vụ. Một trong những phương pháp đưa ra ở các router để cải thiện chất lượng

dịch vụ trong mạng IP thông dụng nhất là phương pháp quản lý hàng đợi tích cực AQM [9, 10, 17]. Đặc trưng của các hàng đợi AQM là điều chỉnh xác suất đánh dấu hoặc loại bỏ gói tin tại các bộ đệm router để ngăn ngừa hiện tượng tắc nghẽn xảy ra. Mục tiêu chính của luận văn là tập trung nghiên cứu và đánh giá các kế hoạch quản lý hàng đợi động cho truyền thông đa phương tiện, nhằm đảm bảo chất lượng dịch vụ QoS. Tập trung nghiên cứu chiến lược BLUE, đánh giá và so sánh chiến lược quản lý hàng đợi BLUE với các chiến lược hàng đợi RED, A-RED. Ngoài việc đưa ra các chiến lược quản lý hàng đợi thích hợp thì một giải thuật cho phép quản lý kiểm soát tắc nghẽn dựa trên sự kiện mất gói dữ liệu và mức độ sử dụng đường truyền thay vì chiếm dụng hàng đợi sẽ đem lại hiệu quả cao nếu được áp dụng vào từng trường hợp cụ thể, xử lý một cách tối ưu việc vận chuyển thông tin trong mạng.

1. Tình hình nghiên cứu trong nước, nước ngoài về AQM

Internet là mạng kết nối mở lớn nhất trên thế giới, mạng của các mạng. Sự phát triển nhanh chóng của Internet dẫn đến Internet phải đối mặt với sự bùng nổ về số lượng máy tính kết nối và sự đa dạng của các lớp ứng dụng triển khai trên nó. Ngày nay khi cơ sở hạ tầng của mạng Internet được nâng cao, đặc biệt là về băng thông, khả năng lưu trữ và xử lý của các máy chủ (servers) đã làm cho nhu cầu của các ứng dụng đa phương tiện qua mạng tăng lên nhanh chóng, các dịch vụ trên Internet không ngừng phát triển, xuất hiện trong mọi lĩnh vực như thương mại, chính trị, quân sự, nghiên cứu, giáo dục, văn hoá, xã hội...

Vì lưu lượng trên Internet có đặc tính bùng nổ nên hàng đợi (bộ đệm) tại các nút mạng (router) phải có kích thước đủ lớn, để đảm bảo cho các nút thực hiện chức năng store-and-forward một cách hiệu quả. Tuy nhiên, nếu thi hành chính sách phục vụ tại hàng đợi kiểu FIFO (Tail-Drop Queue) thì hàng đợi sẽ thường xuyên ở trạng thái đầy, làm tăng đáng kể thời gian trễ trung bình của các gói tin trong mạng. Do vậy, điều quan trọng là phải có các kỹ thuật để đảm bảo cho mạng đạt được thông lượng cao và thời gian trễ trung bình nhỏ. Quản lý hàng đợi tích cực AQM (Active Queue Management) là một trong các giải pháp quan trọng và hiệu quả cho điều khiển tránh tắc nghẽn trên Internet [17,21].

Thông thường có hai phương án để kiểm soát tránh tắc nghẽn là tăng hiệu suất các thiết bị phần cứng và dùng kỹ thuật phần mềm. Việc tăng hiệu suất các thiết bị là cần thiết, nhưng lại khá tốn kém, khó đồng bộ và hiệu quả chưa cao. Ngược lại, dùng kỹ thuật phần mềm để kiểm soát tắc nghẽn đã đem lại hiệu quả rất lớn. Trong kỹ thuật này có hai phương pháp được quan tâm và phát triển, đó là: cải tiến các giao thức điều khiển truyền thông và nâng cao các kỹ thuật quản lý hàng đợi tích cực AQM tại các nút mạng. Việc tăng hiệu năng của giao thức TCP thông qua các biến thể đã triển khai trên Internet và đã đem lại hiệu quả rất lớn. Tuy nhiên, do sự đa chuẩn của các loại mạng, sự phong phú các thiết bị kết nối và sự phức tạp các ứng dụng truyền thông nên điều quan trọng là cần có những cơ chế quản lý hàng đợi tích cực tại các nút mạng để hỗ trợ điều tiết lưu thông trên mạng, nhằm tránh và giải quyết tắc nghẽn.

Quản lý hàng đợi là một nhóm tổ hợp các phương pháp quản lý bộ đệm, đây là một trong những cơ chế cung cấp chất lượng dịch vụ (QoS). Quản lý hàng đợi quyết định việc phân phối bộ đệm và loại bỏ các gói đến theo một chính sách được quyết định trước.

Trong những năm gần đây, vấn đề nghiên cứu về chiến lược quản lý hàng đợi tích cực AQM trong mạng Internet đã phát triển mạnh mẽ và sôi động. Ở trong nước và nhiều nước trên thế giới cũng đã có nhiều công trình nghiên cứu tập trung vào nghiên cứu cải tiến các giao thức điều khiển từ đầu cuối đến đầu cuối (end-to-end) nhằm nâng cao hiệu năng của giao thức TCP, như: TCP NewReno, Vegas và các phương pháp quản lý hàng đợi tích cực, như: RED [12,13,14,26], ARED [28], ARIO [25], BLUE [13,24]… tại các nút mạng trung tâm. Thông qua các cơ chế đó, mỗi nút mạng đã kiểm soát được số lượng lớn các gói dữ liệu đến đồng thời trong hàng đợi của bộ định tuyến. Kết quả của những công trình nghiên cứu đã tập trung nghiên cứu một số giải pháp giải quyết vấn đề tránh tắc nghẽn duy trì tính ổn định của chất lượng mạng và hướng đến việc bảo đảm QoS trong môi trường mạng có mật độ gói tin dày đặc. Việc bảo đảm chất lượng dịch vụ liên quan mật thiết đến việc phân chia tài nguyên mạng (băng thông, bộ đệm). Tại mỗi nút mạng, việc phân chia băng thông, bộ đệm được thực hiện bằng bộ định trình lưu lượng theo một cơ chế định trình nhất định. Chất lượng dịch vụ toàn trình của mỗi ứng dụng phụ thuộc vào chất lượng dịch vụt tại mỗi nút mạng, và phụ thuộc vào gói tin của bộ định trình, thời gian gói tin bị trễ trong bộ đệm, khả năng mất gói tin do tràn bộ đệm. Có nhiều các kết quả khả thi từ việc nghiên cứu tăng cường khả năng bảo đảm chất lượng dịch vụ trong mạng IP nhằm ngăn ngừa hiện tương tắc nghẽn xảy ra. Tuy nhiên qua khảo sát các công trình nghiên cứu trong và ngoài nước cho thấy các giải thuật AQM vẫn còn hạn chế khi ứng dụng đòi hỏi đáp ứng thời gian thực như truyền phát video trên mạng. Do đó việc đảm bảo QoS và đáp ứng yêu cầu nêu trên và kết hợp các cơ chế nhằm đem lại hiệu quả cao nhất trong môi trường mạng phức tạp như hiện nay.

2. Mục tiêu, kết quả cần đạt được của luận văn

Mục tiêu chính của Luận văn là tập trung nghiên cứu và đánh giá hiệu suất của thuật toán quản lý hàng đợi BLUE - một chiến lược điển hình của thuật toán quản lý hàng đợi tích cực dựa vào tải nạp. Sau đó so sánh chiến lược này với các chiến lược quản lý hàng đợi khác như RED, A-RED, ARIO từ đó có những đánh giá, đưa ra các kết quả so sánh hiệu năng giữa các mô hình dựa trên các kết quả mô phỏng trên NS-2. Ngoài ra, vì mục đích cuối cùng là phải hướng tới người sử dụng, nên chúng tôi cũng đã dành một chương để trình bày tổng quan về truyền thông đa phương tiện trên mạng, đây là các dịch vụ ở mức ứng dụng, hiệu quả của nó phụ thuộc chặt chẽ vào các dịch vụ mức dưới.

Kết quả cần đạt được của luận văn: Nghiên cứu thuật toán RED, ARED, ARIO BLUE, tập trung nghiên cứu chiến lược quản lý hàng đợi BLUE. So sánh chiến lược này với các chiến lược quản lý hàng đợi khác từ đó có những đánh giá, đưa ra các kết

quả so sánh hiệu năng giữa các mô hình.

3. Đối tượng và phạm vi nghiên cứu

Đề tài tập trung nghiên cứu lý thuyết về truyền thông đa phương tiện và các yêu cầu bảo đảm QoS đồng thời nghiên cứu một số chiến lược quản lý hàng đợi động, hiệu quả tại gateway, đi sâu nghiên cứu về BLUE – Một chiến lược quản lý hàng đợi dựa vào tải nạp, có thể được cài đặt để hỗ trợ Internet hoạt động hiệu quả hơn..

Đề tài sử dụng bộ công cụ mô phỏng mạng NS2 để nghiên cứu sâu về BLUE và đánh giá, so sánh hiệu suất của nó với các chiến lược quản lý hàng đợi RED, ARED.

4. Phương pháp nghiên cứu

Để đạt được các mục tiêu trên, phương pháp nghiên cứu trong luận văn được kết hợp chặt chẽ giữa nghiên cứu lý thuyết với cài đặt mô phỏng kiểm chứng. Về lý thuyết, luận văn nghiên cứu, khảo sát các công trình liên quan để tìm những tồn tại, lựa chọn những vấn đề sẽ giải quyết. Hệ thống những vấn đề cần giải quyết, đề xuất mô hình lý thuyết, sử dụng những công cụ hỗ trợ để phân tích. Luận văn thực hiện mô phỏng bằng phần mềm mô phỏng mạng NS2 (Network Simulator) được các nhà nghiên cứu khoa học tin dùng.

5. Bố cục của luận văn

Luận văn gồm phần mở đầu, 4 chương nội dung, kết luận. Cụ thể nội dung của các chương trong luận văn được trình bày như sau:

Chương 1: Trình bày về truyền thông đa phương tiện và các yêu cầu chất lượng dịch vụ QoS và các phương pháp ảm bảo chất lượng dịch vụ trong truyền thông đa phượng tiện trên mạng.

Chương 2: Trình bày tổng quan về các chiến lược quản lý hàng đợi động AQM, tìm hiểu hai thuật toán tiêu biểu của AQM: RED, A-RED

Chương 3. Tập trung nghiên cứu sâu về chiến lược quản lý hàng đợi dựa vào tải nạp BLUE và đề xuất cải tiến giải thuật quản lý hàng đợi BLUE

Chương 4: Dựa trên bộ mô phỏng mạng NS để kiểm chứng các đánh giá hiệu suất đồng thời so sánh hiệu suất của chiến lược BLUE với các chiến lược quản lý hàng đợi khác: RED, A-RED

Phần kết luận nêu những kết quả chính của luận văn và hướng phát triển tiếp theo.

Chương 1. TỔNG QUAN VỀ TRUYỀN THÔNG ĐA PHƯƠNG TIỆN VÀ CÁC YÊU CẦU CHẤT LƯỢNG DỊCH VỤ

1.1. Các khái niệm cơ bản

1.1.1. Hệ thống truyền thông đa phương tiện

❖ *Các khái niệm*

- Media:

Media là phương tiện truyền đạt thông tin, đề cập đến các loại thông tin hay loại biểu diễn thông tin như dữ liệu văn bản, ảnh, âm thanh và video.

Multimedia là kỹ thuật mô phỏng và sử dụng đồng thời nhiều dạng phương tiện chuyển hoá thông tin và các tác phẩm từ kỹ thuật đó.

Phân loại media trong hệ thống đa phương tiện: Có nhiều cách phân loại, nhưng cách chung nhất là phân loại trên cơ sở khuôn mẫu (format) vật lý hay các quan hệ media với thời gian. Ta phân loại truyền thông dựa trên việc có hay không có chiều thời gian. Ngầm định này hướng tới hai lớp media: Media độc lập với thời gian và Media phụ thuộc thời gian (hoặc tính liên tục về thời gian).

+ *Media độc lập với thời gian (media tĩnh)*: Không bao hàm yếu tố thời gian, các nội dung và ý nghĩa của chúng không liên quan gì đến việc định thời luồng dữ liệu. Truyền thông tĩnh bao gồm các dữ liệu như văn bản, đồ hoạ, ảnh.

+ *Media phụ thuộc thời gian (media động)*: Bao hàm yếu tố thời gian, thông tin có quan hệ chặt chẽ với thời gian, phải được trình diễn trước người sử dụng vào những thời điểm xác định. Media phụ thuộc thời gian bao gồm animation (phim hoạt hoạ), audio (âm thanh), video, game online (trò chơi trực tuyến). Loại Media này phụ thuộc chặt chẽ vào tốc độ trình diễn. Ví dụ để truyền cảm giác chuyển động nhịp nhàng video phải thực hiện 25 hình mỗi giây (hoặc 30 hình trên giây phụ thuộc vào hệ thống video đang sử dụng). Tương tự, khi chúng ta phát lại một lời nhắn hoặc đoạn nhạc đã được ghi âm, chúng chỉ được cảm nhận tự nhiên khi đạt được tốc độ nhất định. Việc phát lại ở chế độ nhanh hơn hoặc chậm hơn sẽ làm giảm chất lượng và ý nghĩa của âm thanh. Vì những truyền thông này phải được phát liên tục ở tốc độ cố định mà chúng thường được gọi là truyên thông liên tục. Chúng cũng còn được gọi truyền thông đẳng thời vì mối quan hệ cố định giữa mỗi đơn vị truyền thông và thời gian.

Một hệ thống đa phương tiện cũng được coi là một hệ thống thời gian thực. Trong truyền thông đa phương tiện, có một khối lượng lớn dữu liệu cần truyền và trao đổi tương tác với nhau đặc biệt là yêu cầu về tương tác thời gian thực, các thông tin cần được truyền thông liên tục (âm thanh, video, ảnh) phải đảm bảo thời gian truyền thông nhất định.

- Dữ liệu Multimedia:

Dữ liệu multimedia được chia thành hai lớp là các dữ liệu liên tục và các dữ liệu không liên tục. Các dữ liệu liên tục bao gồm các dữ liệu âm thanh, video thay đổi theo thời gian. Các dữ liệu không liên tục là các dữ liệu không phục thuộc vào thời gian, các loại dữ liệu đặc trưng cho dạng này là các dữ liệu văn bản (có hoặc không có định dạng), hình ảnh tĩnh và các đối tượng đồ họa. Dữ liệu multimedia là dữ liệu ở các dạng thông tin

khác nhau. Các kiểu dữ liệu Multimedia là các dữ liệu ở các dạng thông tin như:

+ Văn bản (có hoặc không có định dạng)

+ Âm thanh (Sound)

+ Hình ảnh (là các hình ảnh được mã hóa sử dụng các dạng thức chuẩn như là JPEG hoặc MPEG.)

+ Video (ảnh động kết hợp âm thanh động)

+ Đồ hoạ (là các bản vẽ, minh họa được mã hóa)

+ Hoạt hình (hình ảnh sử dụng theo nguyên tắc chiếu phim)

Các đặc trưng chính của dữ liệu Multimedia bao gồm:

+ Có dung lượng lớn: Các dữ liệu video và âm thanh thường đòi hỏi các thiết bị lưu trữ lớn.

+ Thiếu cấu trúc: Các dữ liệu multimedia có khuynh hướng phi cấu trúc vì vậy các tác nghiệp quản trị dữ liệu chuẩn như chỉ số hoá, tìm kiếm nội dung, truy vấn dữ liệu thường là không áp dụng được.

+ Tính tạm thời: Một vài kiểu dữ liệu Multimedia như là Video, âm thanh và hoạt hình đều phụ thuộc vào yếu tố thời gian liên quan mật thiết đến việc lưu trữ, thao tác và mô tả chúng.

+ Các ứng dụng hỗ trợ: Các dữ liệu phi chuẩn có thể đòi hỏi các quy trình xử lý phức tạp như việc sử dụng các thuật toán nén dữ liệu đối với các ứng dụng dữ liệu multimedia.

- Hệ thống truyền thông đa phương tiện:

Hệ thống truyền thông đa phương tiện (Multimedia Communication System) là hệ thống cung cấp tích hợp các chức năng lưu trữ, truyền dẫn và trình diễn các kiểu phương tiện mang tin rời rạc (văn bản, hình ảnh, đồ hoạ…) và liên tục (audio, video) trong một môi trường thông tin số.

Yêu cầu của truyền thông đa phương tiện:

+ Băng thông đủ lớn

+ Có khả năng phân chia lưu lượng cho từng loại dữ liệu, từng loại dịch vụ.

+ Có chính sách QoS với từng loại dữ liệu

+ Khả năng thích ứng với nhiều thiết bị người dùng

+ Khả năng quản lý tốt, dễ dàng mở rộng, nâng cấp

1.1.2. Hệ thống thời gian thực

Hệ thống thời gian thực - RTS (Real-Time System) là hệ thống mà trong đó sự đúng đắn của việc thực hiện các thao tác không chỉ phụ thuộc vào việc thu được kết quả đúng mà còn phải đưa ra kết quả đúng thời điểm. RTS khác biệt với các hệ thống khác ở tính quan trọng của thời điểm cho ra kết quả, điều đó có nghĩa là tính đúng đắn của hệ thống thời gian thực không chỉ phục thuộc vào kết quả logic của thao tác mà còn phụ thuộc vào thời điểm tạo ra các kết quả.

Hệ thống thời gian thực được thiết kế nhằm cho phép trả lời lại các yếu tố kích thích phát sinh từ các thiết bị phần cứng trong một ràng buộc thời gian xác định. Các tác vụ của hệ thống phải có giới hạn về thời gian bắt buộc phải nằm trong khoảng thời hạn

kết thúc (deadlime) đó là khoảng thời gian mà một thao tác cần để hoàn thành. Có thể hiểu thêm RTS bằng cách hiểu thế nào là một tiến trình, một công việc thời gian thực. Nhìn chung, trong những RTS chỉ có một số công việc được gọi là công việc thời gian thực, các công việc này có một mức độ khẩn cấp riêng phải hoàn tất, ví dụ một tiến trình đang cố gắng điều khiển hoặt giám sát một sự kiện đang xảy ra trong thế giới thực. Bởi vì mỗi sự kiện xuất hiện trong thế giới thực nên tiến trình giám sát sự kiện này phải xử lý theo kịp với những thây đổi của sự kiện này. Sự thay đổi của sự kiện trong thế giới thực xảy ra rất nhanh, mỗi tiến trình giám sát sự kiện này phải thực hiện việc xử lý trong một khoản thời gian ràng buộc gọi là deadline, khoảng thời gian ràng buộc này được xác định bởi thời gian bắt đầu và thời gian hoàn tất công việc. Trong thực tế, các yếu tố kích thích xảy ra trong thời gian rất ngắn vào khoảng vài mili giây, thời gian mà hệ thống trả lời lại yếu tố kích thích đó tốt nhất vào khoảng dưới một giây, thường vào khoảng vài chục mili giây, khoảng thời gian này bao gồm thời gian tiếp nhận kích thích, xử lý thông tin và trả lời lại kích thích. Một yếu tố khác cần quan tâm trong RTS là những công việc thời gian thực này có tuần hoàn hay không? Công việc tuần hoàn thì ràng buộc thời gian ấn định theo từng chu kỳ xác định. Công việc không tuần hoàn xảy ra với ràng buộc thời gian vào lúc bắt đầu và lúc kết thúc công việc, ràng buộc này chỉ được xác định vào lúc bắt đầu công việc. Các biến cố kích hoạt công việc không tuần hoàn thường dựa trên kỹ thuật xử lý ngắt của hệ thống phần cứng.

Trong hệ thống thời gian thực chúng có các đặc điểm sau:

+ Các sự kiện bên trong và bên ngoài có thể xảy ra một cách định kỳ hoặc tự phát.
+ Sự đúng đắn của hệ thống còn phụ thuộc cả vào việc đáp ứng các ràng buộc thời gian.

1.1.3. Chất lượng dịch vụ QoS

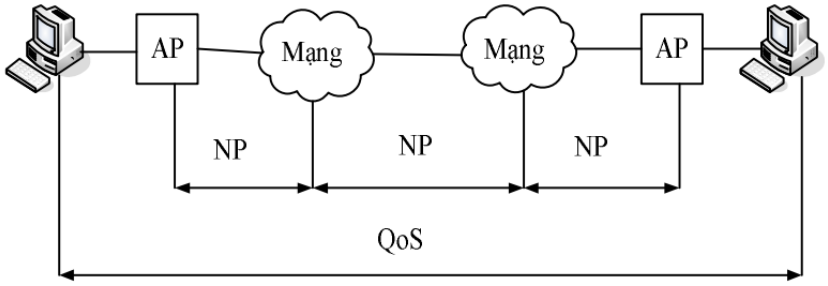
1.1.3.1. Khái niệm QoS

Như chúng ta đã biết con người cảm nhận chất lượng dịch vụ QoS (Quality of Service) bằng giác quan. Ví dụ, với người sử dụng dịch vụ thoại, cảm nhận về chất lượng dịch vụ cung cấp tốt khi thoại được rõ ràng, điều này có nghĩa là phải đảm bảo tốt về giá trị tham số trễ, biến thiên độ trễ và giá trị tham số mất gói tin với một tỉ lệ tổn thất nào đó có thể chấp nhận được. Nhưng đối với khách hàng là người sử dụng trong truyền số liệu ở ngân hàng thì điều quan trọng là độ tin cậy, có thể chấp nhận trễ lớn, biến thiên độ trễ lớn, nhưng thông số mất gói tin, độ bảo mật kém thì không thể chấp nhận được. Nhìn chung theo quan điểm của khách hàng thì họ mong muốn được cung cấp các dịch vụ mạng đảm bảo chất lượng. Theo khuyến nghị E800 ITU-T, chất lượng dịch vụ là "*Một tập các khía cạnh của hiệu năng dịch vụ nhằm xác định cấp độ thỏa mãn của người sử dụng đối với dịch vụ*". Như vậy QoS được xác định bằng các chỉ tiêu định tính và định lượng. Chỉ tiêu định tính thể hiện sự cảm nhận của khách hàng còn chỉ tiêu định lượng được thực hiện bằng các số đo cụ thể.

Trên quan điểm của nhà cung cấp dịch vụ mạng thì khái niệm chất lượng mạng là một chuỗi các tham số mạng có thể được xác định, được đo đạc và điều chỉnh để có thể đạt được mức độ hài lòng của khách hàng về dịch vụ. Nhà cung cấp dịch vụ có trách

nhiệm phải tổ hợp các tham số chất lượng mạng khác nhau thành tập hợp các tiêu chuẩn để có thể vừa đảm bảo lợi ích kinh tế của mình vừa thoả mãn tốt nhất yêu cầu của người sử dụng. Khi sử dụng dịch vụ, khách hàng chỉ biết đến nhà cung cấp dịch vụ chứ không quan tâm tới các thành phần của mạng. Công việc đảm bảo QoS cho các dịch vụ mà họ cung cấp cho người sử dụng là thực hiện các biện pháp để duy trì các mức QoS theo nhu cầu, với cơ sở hạ tầng mạng hiện có, thỏa mãn các tiêu chuẩn như độ tin cậy, tính bảo mật và băng thông với thời gian trễ chấp nhận được... Còn với các dịch vụ đa phương tiện chất lượng cao như nghe nhạc, xem phim trực tuyến, VoIP,… được truyền trên mạng thì quá trình phát và nhận theo thời gian thực đòi hỏi phải triển khai một mạng có hỗ trợ việc đảm bảo chất lượng dịch vụ.

Dưới đây biểu diễn một mô hình QoS tổng quát:



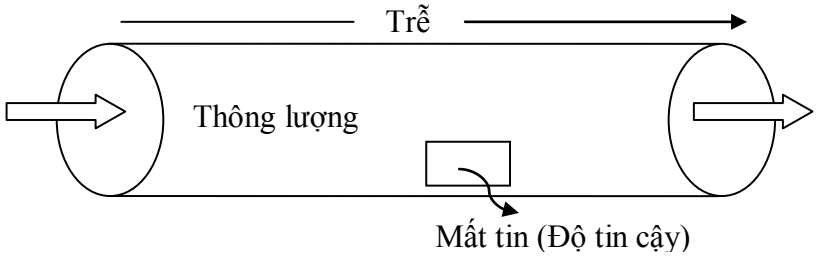
Hình 1.1. Mô hình QoS tổng quát

Trong hình vẽ, NP (Net Performance) là năng lực và hiệu quả của một mạng cụ thể. Nó bao gồm khả năng ứng xử, tính hiệu quả của mạng và chất lượng phục vụ mà mạng cung cấp. AP (Access Point) là điểm truy nhập mạng.

Việc đáp ứng chất lượng dịch vụ QoS trên mạng Internet và các mạng TCP/IP khác đã được nhóm IETF phát triển thêm dịch vụ như dành trước tài nguyên, sử dụng giao thức RSVP (Resource Reservation Protocol), RSVP cho phép yêu cầu dành riêng băng thông giữa các mạng nối kết bằng các bộ định tuyến thông qua việc yêu cầu các bộ định tuyến dành riêng một độ rộng dải thông của nó để dành cho những luồng lưu thông đặc biệt. Như vậy định nghĩa chất lượng dịch vụ theo quan điểm mạng chính là các cơ chế, công cụ đảm bảo cho các mức dịch vụ khác nhau thỏa mãn các tiêu chuẩn như độ tin cậy, tính bảo mật cao, băng thông đủ lớn với thời gian trễ cần thiết cho một ứng dụng đặc biệt nào đó.

1.1.3.2. Các tham số chính của QoS

Chất lượng dịch vụ bao gồm các tham số kỹ chính thuật như: độ trễ, thông lượng, tỷ số mất tin, jitter có thể được minh họa bằng Hình 1.1 dưới đây.



Hình 1.2. Các tham số QoS chính

❖ *Độ trễ (Delay)*

Độ trễ là thời gian cực đại để truyền một gói tin từ trạm nguồn đến trạm đích, bao gồm thời gian phát một gói tin lên đường truyền, thời gian xử lý tại các hàng đợi tại các router mà gói tin sẽ đi qua và thời gian truyền trên các đường truyền từ trạm nguồn đến trạm đích; nó phụ thuộc vào thời gian xử lý của nút mạng và băng thông đường truyền (thời gian gói tin chờ trong bộ nhớ đệm). Mỗi thành phần trong tuyến kết nối từ đầu cuối đến đầu cuối như: thiết bị phát, thiết bị truyền dẫn, thiết bị chuyển mạch và định tuyến đều có thể gây ra trễ.

Đối với các ứng dụng truyền thông đa phương tiện đòi hỏi độ trễ các gói tin nằm trong khoảng cho phép, có nghĩa là phải giới hạn bởi một ngưỡng cụ thể. Có nhiều dịch vụ, đặc biệt là dịch vụ tương tác thời gian thực như điện thoại Internet bị ảnh hưởng rất nhiều bởi trễ, truyền thông tương tác gặp khó khăn khi nếu độ trễ vượt quá ngưỡng cho phép. Độ trễ nhỏ hơn 150 ms là không gây ra vấn đề gì bởi vì với độ trễ trong khoảng này thì giác quan con người không cảm nhận được độ trễ này, nếu độ trễ nằm trong khoảng 150 đến 400 ms thì vẫn có thể được chấp nhận nhưng chất lượng kém hơn. Còn nếu độ trễ vượt quá 400 ms thì cực tệ không chấp nhận được khi đó cuộc đàm thoại sẽ bị ngắt quãng và người sử dụng sẽ đánh giá chất lượng thoại ở mức thấp. Do vậy, các gói tin đến trễ hơn một ngưỡng cho phép, các gói tin coi là bị mất.

❖ *Thông lượng (Throughput)*

Thông lượng quyết định khả năng truyền tin giữa hai điểm kết nối, thông lượng là số lượng bit được truyền thành công trong một đơn vị thời gian (đơn vị là bit/s). Đối với từng loại mạng khác nhau cho phép tốc độ luồng thông qua và kích thước gói tin khác nhau.

❖ *Tỉ số mất tin (Packet loss)*

Tỉ số mất tin là tỉ số giữa số gói tin bị mất (do bị lỗi hoặc bị huỷ bỏ do hàng đợi của nút mạng bị đầy) và tổng số gói tin được truyền. Mất gói tin thường xảy ra khi xuất hiện tắc nghẽn trên đường truyền của các gói, làm cho hàng đợi của nút mạng bị đầy. Việc mất gói này gây ra mất mát thông tin phía thu, tạo ra trễ khi phải truyền lại các gói bị mất hay truyền thông tin thông báo, điều này làm giảm các giá trị của các ứng dụng đa phương tiện và thậm chí gây tắc nghẽn trong mạng. Với truyền thông đa phương tiện, tỉ lệ mất gói từ 10-20% có thể chấp nhận được, phụ thuộc vào tín hiệu được mã hoá và được che giấu ở phía nhận như thế nào. Tuy nhiên, trong trường hợp tắc nghẽn nghiêm trọng, sự mất mát gói tin vượt quá 20%, tín hiệu ở phía đầu nhận là khó chấp nhận. Thường thì tỉ lệ mất gói ảnh hưởng nhiều tới dịch vụ điện thoại IP, VoIP hơn là các dịch vụ dữ liệu. Trong quá trình truyền thoại việc mất gói tin sẽ dẫn đến hiện tượng tiếng nói bị đứt đoạn, bị ngắt quãng, hoặc trong truyền dữ liệu hiện tượng mất gói tin làm bức ảnh bị mờ đi một phần. Tỉ lệ mất gói tin cao làm tăng độ trễ và jitter.

Ngoài ra còn có khái niệm kích thước mất tin: đó là số gói tin bị mất liên tiếp cực đại. Bên cạnh tỷ số mất tin ta có thể dùng khái niệm độ tin cậy: tỷ số mất tin tỷ lệ nghịch với độ tin cậy

❖ *Độ biến thiên độ trễ (Jitter)*

Biến thiên độ trễ là sự khác nhau về độ trễ của các gói tin khác nhau trong cùng một dòng lưu lượng. Trong mạng IP, các gói tin của cùng một ứng dụng có thể được truyền theo các đường đi khác nhau. Biến thiên độ trễ chủ yếu do sự sai khác về thời gian xếp hàng của các gói liên tiếp trong một luồng gây ra và là một trong những vấn đề nghiêm trọng của QoS. Khi biến biên độ trễ nằm trong khoảng dung sai định nghĩa trước thì không ảnh hưởng tới chất lượng dịch vụ, ngược lại nếu độ biến thiên độ trễ quá lớn sẽ làm cho kết nối mạng bị đứt quãng. Trong một số ứng dụng như ứng dụng thời gian thực không thể chấp nhận dung sai, biến thiên độ trễ lớn có thể được xử lý bằng bộ đệm, song nó lại làm tăng trễ.

1.1.3.3. Các mức QoS

Nói đến các mức dịch vụ là nói đến khả năng thực sự của QoS đầu cuối. Điều này có nghĩa là khả năng của mạng để cung cấp các dịch vụ cần thiết bởi lưu lượng mạng đặc biệt từ đầu cuối tới đầu cuối hoặc từ biên tới biên. Các dịch vụ này khác nhau theo các tham số của QoS như: băng thông, trễ, jitter, …

Có ba mức dịch vụ:

- Dịch vụ cố gắng tối đa (Best-Effort Service): Với dịch vụ này mạng sẽ khai thác hết khả năng trong giới hạn cho phép, nhưng không đảm bảo độ trễ và mất mát dữ liệu. Vì vậy, khi có nhiều luồng lưu lượng truyền đi trong mạng và vượt quá khả năng của mạng, dịch vụ không bị từ chối nhưng chất lượng dịch vụ giảm: thời gian trễ tăng, tốc độ giảm và mất dữ liệu. Với Best-Effort, dữ liệu đi vào mạng đều tuân theo quy tắc FIFO. Không có sự phân loại giữa các luồng.
- Dịch vụ phân loại (Differrentiated Service): còn gọi là QoS mềm. Một vài dòng lưu lượng của dịch vụ được ưu tiên hơn những dòng lưu lượng còn lại (ví dụ như cam kết các dịch vụ khác nhau như thoại, video sẽ có băng thông ổn định, tốc độ xử lý nhanh hơn, tỉ lệ mất gói ít hơn, …). Đây là sự ưu tiên thống kê, được cung cấp bởi việc phân loại lưu lượng và các công cụ như: hàng đợi ưu tiên (PQ) - , CQ, hàng đợi cân bằng có trọng số (WFQ),…
- Dịch vụ đảm bảo (Guaranteed service): còn được gọi là QoS cứng. đây là sự đặt trước tài nguyên cho các dịch vụ đặc biệt. Được cung cấp thông qua QoS với các công cụ: RSVP và CBQ. Dịch vụ được đảm bảo tuyệt đối về tài nguyên mạng dành cho nó, với điều khoản cụ thể như băng thông, trễ, mất gói…

1.1.3.4. Đảm bảo chất lượng dịch vụ (QoS) trong truyền thông đa phương tiện

Do tính đa dạng của dịch vụ, ứng dụng trên mạng, các yêu cầu về đảm bảo QoS cho các ứng dụng cũng hết sức đa dạng. Do đặc tính thời gian thực, các ứng dụng đa phương tiện yêu cầu băng thông lớn, không gian lưu trữ rộng, độ trễ, biến thiên trễ nhỏ, đồng bộ về thời gian, không gian. Hơn nữa các phương tiện khác nhau cũng có những yêu cầu khác nhau, các yêu cầu này được thoả mãn để truyền thông và thể hiện trong toàn hệ thống. Để đảm bảo khung thống nhất cho việc mô tả và đảm bảo các yêu cầu đa dạng người ta đánh giá QoS dựa trên tập các tham số cơ bản. Đối với truyền thông đa phương

tiện về chất lượng dịch vụ có thể được phân thành các loại: chất lượng qua cảm nhận (nghe, nhìn) của người sử dụng, chất lượng dịch vụ của ứng dụng hay chất lượng dịch vụ truyền dữ liệu qua mạng. Các đặc điểm để các yêu cầu QoS truyền thông đa phương tiện không chỉ được đánh giá bởi mức độ điều khiển quản trị vật lý và QoS mạng truyền dữ liệu mà còn bởi chất lượng thông tin cảm nhận của người dùng. Các ứng dụng khác nhau có yêu cầu chất lượng dịch vụ khác nhau.

Yêu cầu chất lượng của một số dịch vụ điển hình như sau:

- Dịch vụ yêu cầu cao về băng thông, chấp nhận trễ: Email, truyền dữ liệu.
- Dịch vụ có thể chấp nhận giảm một phần băng thông, nhưng yêu cầu cao về độ trễ: Thoại (VoIP), điện thoại video, hội nghị truyền hình.
- Dịch vụ chấp nhận độ trễ lớn: IPTV, Video theo yêu cầu
- Những dịch vụ đang phát triển có yêu cầu tương tác thời gian thực như: thương mại điện tử, e-banking, giao dịch chứng khoán, trò chơi trực tuyến,…

Các hệ thống đa phương tiện thường là phân tán, hoạt động trên mạng máy tính, các yêu cầu tài nguyên cho sự hoạt động của hệ thống thường là động, Ví dụ trong ứng dụng Video conference thì yêu cầu về tài nguyên phụ thuộc số người tham gia. Do đó, cần có các giải pháp để đảm bảo chất lượng các dịch vụ (đảm bảo QoS) của ứng dụng thoả mãn yêu cầu của người dùng. Mục đích chính của QoS là cung cấp băng thông riêng, điều khiển độ trễ và jitter, giảm tỷ lệ mất mát gói tin cho các luồng lưu lượng của các ứng dụng thời gian thực và tương tác. Một điều quan trọng nữa là nó cung cấp quyền ưu tiên cho một hoặc một vài luồng trong khi vẫn đảm bảo các luồng khác (có quyền ưu tiên thấp hơn) không mất quyền được phục vụ. Việc đảm bảo chất lượng dựa trên cơ sở là quản lý tài nguyên vì QoS phụ thuộc vào tài nguyên khả dụng của hệ thống, việc quản lý tài nguyên ở đây là:

+ Tính toán hoặc ước lượng được hiệu suất sử dụng tài nguyên
+ Dành tài nguyên cho dịch vụ
+ Lập lịch truy cập tài nguyên

1.2. Các ứng dụng đa phương tiện trên mạng Internet

Các ứng dụng đa phương tiện trên mạng chúng có yêu cầu QoS khác rất nhiều so với yêu cầu của các ứng dụng hướng dữ liệu truyền thống như: Web Text/Image, Email, FTP, DNS,… Với các ứng dụng đa dạng trên Internet, đòi hỏi phải đảm bảo các thông số QoS ở các mức độ khác nhau. Ở các ứng dụng truyền thống chúng có thể chấp nhận độ trễ và độ thăng giáng lớn nhưng không chấp nhận sự mất mát dữ liệu. Còn với các ứng dụng truyền thông đa phương tiện, chẳng hạn như dữ liệu audio, video chất lượng ứng dụng thay đổi rất nhạy với độ trễ, biến thiên độ trễ và phụ thuộc vào một số tham số mạng khác như băng thông, tỉ suất lỗi, ... Kiến trúc mạng truyền thống vốn được thiết kế chính cho truyền dữ liệu không phù hợp lắm với các ứng dụng đa phương tiện.

Hai đặc tính quan trọng của các ứng dụng đa phương tiện đó là: chấp nhận mất mát dữ liệu ở một mức độ nhất định; yêu cầu độ trễ nhỏ và chỉ thay đổi trong một phạm vi nhất định. Các yêu cầu về QoS cho các ứng dụng đa phương tiện cũng khác rất nhiều

so với yêu cầu của các ứng dụng hướng dữ liệu truyền thống.

1.2.1. Truyền video và audio đã được lưu trữ

Trong lớp ứng dụng này, người dùng tại các máy trạm (client) yêu cầu truy cập đến các file audio, video đã được nén và được lưu trữ trên các máy phục vụ (server). Các file âm thanh được lưu trước có thể gồm thu thanh bài giảng, bài hát, hoặc các đoạn băng được ghi âm từ trước,… Các file video có thể là những bộ phim, phim tài liệu, các đoạn video của những sự kiện thể thao, giải trí.. Tại một thời điểm nào đó, client yêu cầu một file audio/video từ server: Sau thời gian trễ vài giây, client sẽ chạy file audio/video trong khi vẫn tiếp tục nhận phần còn lại của file từ server. Đặc tính vừa chạy file, trong khi tiếp tục nhận những phần sau của file gọi là streaming. Nhiều ứng dụng còn cung cấp tính năng tương tác với người dùng (user interactivity): Pause, Resume, Jump, Skip. Khoảng thời gian từ lúc người dùng đưa ra yêu cầu (play, skip, forward) tới khi bắt đầu nghe/nhìn thấy trên máy client nên nằm trong khoảng từ 1 – 10 giây để có thể chấp nhận được. Yêu cầu của lớp ứng dụng này đối với độ trễ và jitter không chặt chẽ bằng ở trong ứng dụng thời gian thực như: điện thoại Internet, video conference thời gian thực. Các chương trình dùng để chơi các file audio/video được lưu trữ như: realPlayer, Windows Media Player, netshow [2] …

1.2.2. Phát sóng trực tiếp của audio và video

Các ứng dụng loại này tương tự như phát thanh và truyền hình quảng bá (broadcast) truyền thống, chỉ có điều nó được thực hiện trên Internet. Ứng dụng truyền dòng âm thanh và hình ảnh trực tiếp cho phép một người dùng bất kì có thể nhận được các chương trình truyền trực tiếp ở mọi nơi trên thế giới. Bởi vì các file audio, video truyền trực tiếp không được lưu giữ trước, người dùng không thể tương tác với một số tính năng như pause, forward, rewind,…được. Tuy nhiên, nếu dữ liệu được lưu giữ cục bộ tại máy của người dùng, một số ứng dụng có thể pause, rewind…Với các ứng dụng truyền hình, phát thanh trực tiếp thường được phát broadcast tới nhiều người dùng qua kĩ thuật multicast hoặc qua nhiều luồng unicast riêng. Chẳng hạn như với ứng dụng điện thoại Internet (Internet phone), Mobile TV hay hội thảo truyền hình (video conferencing), cho phép một người có thể giao tiếp bằng âm thanh và hình ảnh với một hay nhiều người khác theo kiểu thời gian thực. Đây là tương tác có cảm nhận, các thành viên tham gia có thể trao đổi với nhau thông qua tiếng nói và hình ảnh trong thời gian thực.

Hạn chế về thời gian của truyền hình, phát thanh trực tiếp là khắt khe hơn so với việc truyền audio, video được lưu trữ; với các ứng dụng loại này thì độ trễ tới 10 giây là có thể chấp nhận được.

1.2.3. Ứng dụng audio, video tương tác thời gian thực

Lớp ứng dụng này cho phép mọi người dùng audio, video để tương tác thời gian thực với người khác. Audio tương tác thời gian thực tiêu biểu được nhắc đến ở đây là điện thoại Internet. Với việc tận dụng môi trường mạng để truyền tín hiệu thoại nên ưu điểm lớn nhất của dịch vụ thoại dựa trên nền Internet này là giá thành rất rẻ, người sử dụng có thể gọi đi khắp thế giới với giá chỉ bằng khoảng 1/10 hoặc thấp hơn nữa so với

giá cước điện thoại truyền thống. Một số ứng dụng tương tác audio thời gian thực điển hình như: Voice Chat trong Yahoo Messenger, MSN Messenger, Skype, Zalo, … Với tương tác video thời gian thực, một ứng dụng cho phép truyền tải hình ảnh và âm thanh giữa hai hoặc nhiều địa điểm khác nhau điển hình như hội nghị truyền hình (video conferenceing), ứng dụng này cho phép nhiều người tham dự tại các địa điểm có thể giao tiếp trực tiếp với nhau bằng âm thanh và hình ảnh. Hiện nay đã có nhiều ứng dụng cho video thời gian thực như Microsoft Netmeeting, Yahoo Messenger, Skype, Tango... Trong các ứng dụng audio/video tương tác thời gian thực thì yêu cầu độ trễ nhỏ hơn vài trăm miligiây. Ví dụ với âm thanh: độ trễ nên nhỏ hơn 400 ms, còn nếu độ trễ lớn hơn 400 ms là không thể chấp nhận được vì với khoảng trễ đó có thể dẫn đến cuộc hội thoại mà các bên không hiểu nhau nói gì.

1.3. Các mô hình đảm bảo QoS cho truyền thông đa phương tiện

Mạng Internet truyền thống được xây dựng theo nguyên tắc “cố gắng tối đa” (Best Effort), đối với dịch vụ Best Effort thì các gói thông tin được truyền đi theo nguyên tắc "đến trước được phục vụ trước" mà không quan tâm đến đặc tính lưu lượng của dịch vụ là gì. Điều này dẫn đến rất khó hỗ trợ các dịch vụ đòi hỏi độ trễ thấp như các dịch vụ thời gian thực hay video. Các gói thông tin được lưu trữ, truyền đi mà không có sự đảm bảo chất lượng dịch vụ QoS. Sự tích hợp các lưu lượng đa phương tiện trên mạng Internet làm nảy sinh các yêu cầu quan trọng về QoS, tất cả các ứng dụng nhạy cảm thời gian thực đòi hỏi một mạng có QoS cao hơn QoS của mạng IP truyền thống dựa trên cơ chế cố gắng tối đa. Để khắc phục nhược điểm của mạng Internet truyền thống, người ta đã đề xuất một số mô hình đảm bảo chất lượng dịch vụ, trong đó có các mô hình khá phổ biến là mô hình Dịch vụ tích hợp - IntServ và mô hình Dịch vụ phân loại - DiffServ. Mỗi mô hình sẽ có những đặc điểm riêng để phù hợp với những yêu cầu QoS cho mạng IP của các loại dịch vụ như đã trình bày trong mục 1.2 ở trên. Mô hình IntServ là mô hình nâng cao hiệu năng hoạt động của mạng IP bằng việc hỗ trợ truyền các lưu lượng thời gian thực và đảm bảo băng thông cho từng luồng lưu lượng này bằng cách dự trữ tài nguyên từ đầu cuối đến đầu cuối đảm bảo cho các luồng lưu lượng thời gian thực được đảm bảo QoS theo yêu cầu. Mô hình DiffServ không xử lý theo từng luồng lưu lượng riêng biệt, do đó nó không sử dụng trạng thái của từng luồng trong các bộ định tuyến mà nó nhóm từng luồng lưu lượng riêng biệt đó thành các nhóm hoặc các lớp lưu lượng cùng với các tham số khác nhau của QoS lại với nhau. Đây là mô hình được coi là bước phát triển tiếp theo nhằm khắc phục hạn chế của mô hình tích hợp dịch vụ.

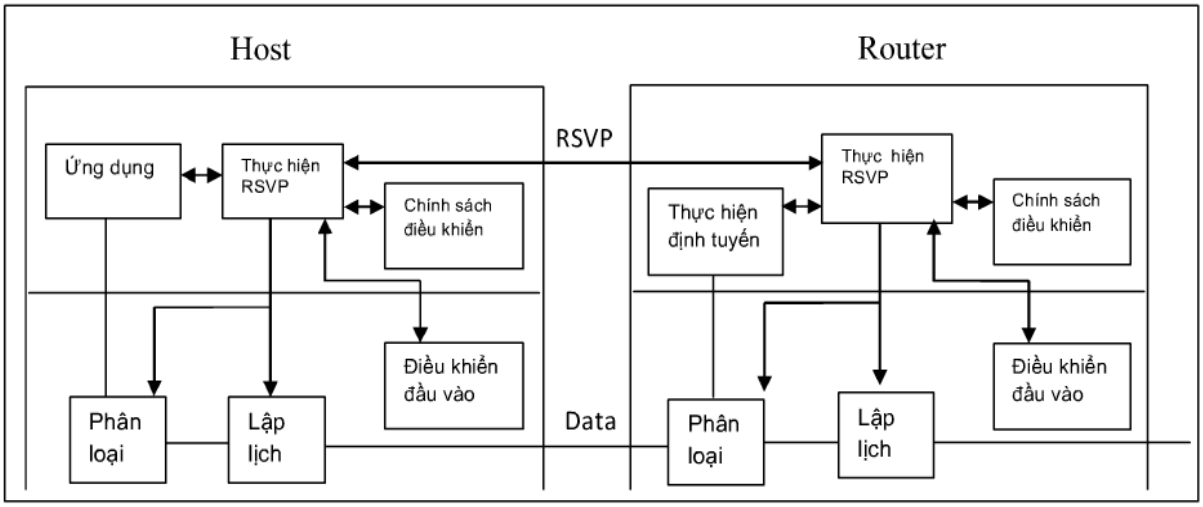
1.3.1. Mô hình dịch vụ tích hợp - IntServ

Cùng với nhu cầu ngày càng tăng trong việc cung cấp dịch vụ thời gian thực (thoại, video) và băng thông cao (đa phương tiện). Mô hình IntServ được được đưa ra bởi nhóm làm việc tại IETF với mục đích hỗ trợ chất lượng dịch vụ cho các ứng dụng từ đầu cuối tới đầu cuối. Mô hình này không những đáp ứng được các dịch vụ Best-Effort mà các dịch vụ thời gian thực cũng được thực thi qua mô hình này qua việc hỗ trợ chức năng dành trước băng thông trên Internet và các mạng tương tác. Các ứng dụng sẽ nhận được

băng thông đúng yêu cầu và truyền đi trong mạng với độ trễ cho phép.

➢ *Nguyên lý hoạt động của mô hình tích hợp dịch vụ*

IntServ sử dụng một giao thức đặc biệt RSVP để dành trước băng thông xác định trong mỗi bộ định tuyến dọc theo đường đi từ nguồn đến đích. Có nghĩa là mô hình dịch vụ tích hợp sẽ duy trì kết nối truyền thông giữa các trạm đầu cuối qua Router bằng cách sử dụng giao thức dành trước tài nguyên để tạo và duy trì trạng thái các luồng lưu lượng dọc theo đường đi của một luồng. Mỗi bộ định tuyến trên đường đi sẽ kiểm tra xem ở đó nó có đảm bảo tài nguyên được yêu cầu và duy trì tuyến khi được yêu cầu bởi yêu cầu dành trước tài nguyên. Khi điều kiện tối thiểu được đáp ứng, ứng dụng nguồn sẽ được thông báo xác nhận. Sau đó, ứng dụng có thể sử dụng đường truyền.



Hình 1.3. Nguyên lý hoạt động của mô hình dịch vụ tích hợp IntServ

Một ứng dụng muốn gửi gói tin đi theo luồng được dự trữ tài nguyên nhằm bảo đảm chất lượng của gói tin thì nó sẽ thực hiện việc truyền đi thông điệp dành trước tài nguyên RSVP tới các nút mạng. Giao thức RSVP cố gắng thiết lập một luồng dành trước cho yêu cầu QoS đó, nó có thể được chấp nhận nếu các ứng dụng phù hợp với chính sách lưu lượng và các Router có thể xử lý các yêu cầu QoS. Sau khi truyền đi thông điệp RSVP tới các nút mạng để dành trước tài nguyên. RSVP sẽ báo cho bộ lập phân loại và bộ lập lịch gói tin trong mỗi nút mạng xử lý và truyền các gói tin đó theo đúng luồng của nó.

Nếu các ứng dụng phân phát các gói tin đến bộ phân loại trong nút đầu tiên, nó sẽ ánh xạ luồng này vào lớp dịch vụ cụ thể để thực hiện yêu cầu QoS, luồng này được đóng gói với địa chỉ IP của bên gửi và được chuyển tới bọ lập lịch gói tin. Bộ lập lịch gói tin chuyển tiếp các gói tin đi đến các giao tiếp đầu ra phụ thuộc vào việc gói tin đó thuộc lớp lưu lượng nào đến các Router hoặc trạm bên phía nhận gói tin.

Giao thức RSVP là giao thức đơn giản, việc dành trước tài nguyên QoS chỉ thực thi theo một hướng, từ nút gửi đến nút nhận. Nếu ứng dụng muốn kết thúc việc dành trước tài nguyên cho luồng dữ liệu, nó gửi một thông điệp dành trước tài nguyên (bật các thông điệp bên trong giao thức RSVP nhằm xoá bỏ dự trữ và xoá bỏ tài nguyên) để giải phóng tài nguyên đã dự trữ để thực hiện QoS trên tất cả các Router nằm trong tuyến

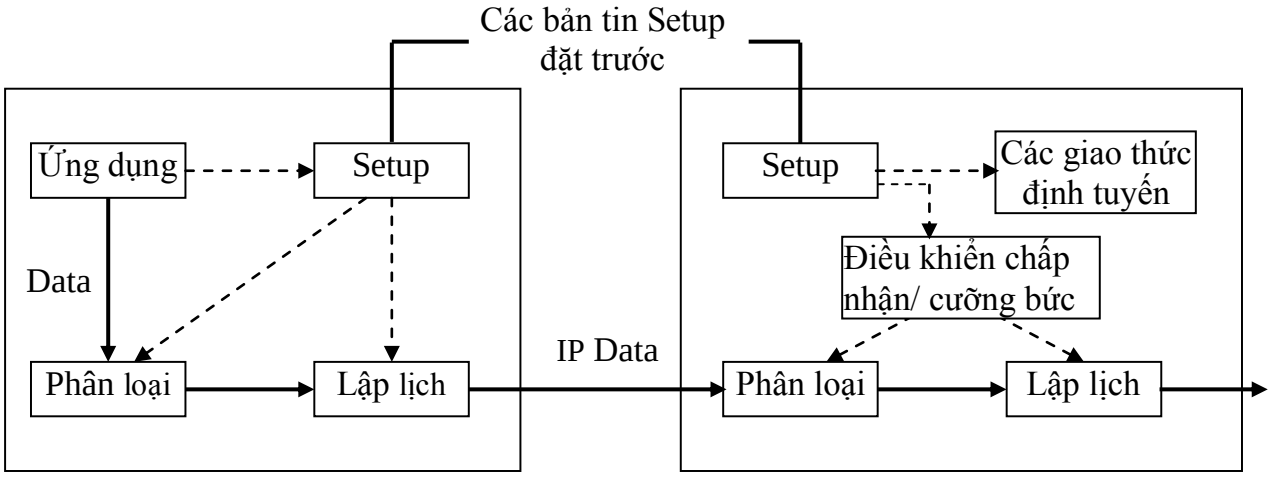
đường đi của gói tin.

Trong mô hình tích hợp dịch vụ, mỗi luồng IP được xác định bởi năm tham số sau:

- Địa chỉ IP đích
- Địa chỉ cổng đích
- Giao thức nhận dạng
- Địa chỉ IP nguồn
- Địa chỉ cổng nguồn

Để dự trữ tài nguyên cho mỗi luồng, ứng dụng đích phải cung cấp các đặc tính luồng. Một số đặc tính của luồng như: thông lượng trung bình, thông lượng bùng nổ (max), thời gian kéo dài sự bùng nổ thông lượng. Các yêu cầu chất lượng dịch vụ khác cho luồng, như độ trễ trung bình...

Tổng quan của mô hình IntServ là cung cấp mô hình dịch vụ cho Internet, liên quan tới mô hình truyền thống dựa trên dịch vụ cố gắng tối đa và lớp Internet IP. Giải pháp này yêu cầu router QoS phải lưu thông tin của tài nguyên còn lại (dung lượng của liên kết, không gian bộ đệm, khả năng tính toán của bộ chuyển tiếp…) sau cấp phát cho một luồng. Để thực hiện được điều này router phải xác định và lưu trữ thông tin của luồng, và đòi hỏi có sự thay đổi trong mô hình Internet (trạng thái mạng chỉ được lưu trữ ở đầu cuối).



Hình 1.4. Mô hình dịch vụ tích hợp IntServ

Mục đích của mô hình này là áp dụng việc đặt trước tài nguyên cho các luồng từ nguồn cho tới đích, các luồng này sẽ được bảm bảo QoS trong cả quá trình hoạt động. Trạng thái của các router được cấu hình động trong suốt quá trình thiết lập tuyến đường. Cơ chế này đòi hỏi phải có cơ chế điều khiển việc chấp nhận luồng lưu lượng vào mạng và và các thiết bị mạng phải có khả năng dành trước tài nguyên của nó để cung cấp các mức chất lượng dịch vụ tùy theo nhu cầu của người sử dụng. Điều này yêu cầu các bộ định tuyến phải có khả năng điều khiển các luồng lưu lượng. Trong kiến trúc của các dịch vụ tích hợp xác định hai lớp dịch vụ chính đó là:

- Dịch vụ được đảm bảo - GS (Guaranteed Service): cung cấp băng tần dành riêng, giới hạn độ trễ tối đa và không bị thất thoát gói tin trong hàng đợi. Nhược điểm của lớp

dịch vụ này là hiệu quả sử dụng tài nguyên mạng thấp vì nó đòi hỏi mỗi luồng lưu lượng có hàng đợi riêng. Chính vì vậy mà GS được áp dụng cho các dịch vụ với độ trễ của dịch vụ được xác định trước chẳng hạn như hội nghị truyền hình chất lượng cao, thanh toán tài chính thời gian thực,…

- Dịch vụ có tải được điều khiển - CLS (Controlled Load Service): các ứng dụng của dịch vụ này có thể chấp nhận khả năng mất dữ liệu và thay đổi độ trễ ở một mức độ nhất định. Dịch vụ này phù hợp cho các ứng dụng không nhạy cảm lắm với độ trễ hay mất gói như truyền hình multicast audio/video chất lượng trung bình.

Hai lớp dịch vụ GS và CLS phải được cài đặt các đường định tuyến và dự trữ các tài nguyên. Có 4 thành phần cơ bản trên router/switch để quản lý tài nguyên trong mạng IntServ bao gồm: Bộ kiểm soát thu nhận (Admission Control), bộ phân loại (Classifier), bộ lập lịch (Scheduler). Ba thành phần này cung cấp việc điều khiển lưu lượng (Traffic control) và giao thức dành trước tài nguyên (Resource Reservation Protocol).

- Kiểm soát thu nhận xem xét việc chấp nhận luồng đi vào mạng thực thi các thuật toán tại các router hoặc máy của người sử dụng để xác định xem một luồng mới có đáp ứng được các yêu cầu RSVP hay không. Thành phần điều khiển chấp nhận luồng thực hiện chấp nhận/quyết định cục bộ, tại thời điểm máy của người sử dụng yêu cầu dịch vụ dọc theo tuyến đường. Thành phần này không chỉ thực hiện việc quyết định có hay không mà nó còn thông báo cho ứng dụng yêu cầu về QoS thấp hơn có thể được đáp ứng

- Phân loại là việc xác định luồng gói tin IP trong các máy của người sử dụng và các Router. Sau đó các gói sẽ được phân ra các lớp khác nhau, phân loại và đưa các gói vào hàng đợi riêng của một luồng cho trước (hoặc của một tập) để sử dụng bởi thành phần lập lịch. Tất cả các gói tin có cùng lớp thì sẽ nhận được sự xử lý như nhau trong lập lịch gói tin.

- Lập lịch gói quản lý việc chuyển tiếp các gói khác nhau sử dụng hàng đợi và bộ định thời. Thành phần này phải bảo đảm các gói tin được phân bố và chuyển tới đầu ra theo luật. Tiến hành lập lịch trình để đáp ứng các yêu cầu QoS.

- Giao thức dành trước tài nguyên: các ứng dụng yêu cầu QoS thông qua bộ dự trữ tài nguyên sẽ thiết lập đường đi và dự trữ tài nguyên cho việc truyền dữ liệu trên mạng.

Giao thức dành trước tài nguyên(RSVP)

Giao thức dành tài nguyên RVSP (Resource Reservation Protocol) được sử dụng bởi IntServ được đặc tả trong RFC2205, các dịch vụ GS và CLS được mô tả trong RFC2210. RSVP có thể gửi yêu cầu đặt trước tài nguyên và đáp ứng tương ứng của thành phần chấp nhận luồng từ máy tính tới router, từ router tới router và từ router tới máy đích (hoặc nhiều một máy). Trong giao thức dành trước tài nguyên RSVP, các nguồn tài nguyên được dành trước theo các hướng độc lập. Máy chủ nguồn và máy chủ đích trao đổi các bản tin RSVP để thiết lập các trạng thái chuyển tiếp và phân loại gói tại mỗi nút. RSVP yêu cầu các máy nhận lưu lượng về yêu cầu chất lượng dịch vụ QoS cho luồng dữ liệu. Các ứng dụng tại máy nhận phải giải quyết các thuộc tính QoS sẽ được truyền tới RSVP. Sau khi phân tích các yêu cầu này, RSVP được sử dụng để gửi các bản tin tới tất

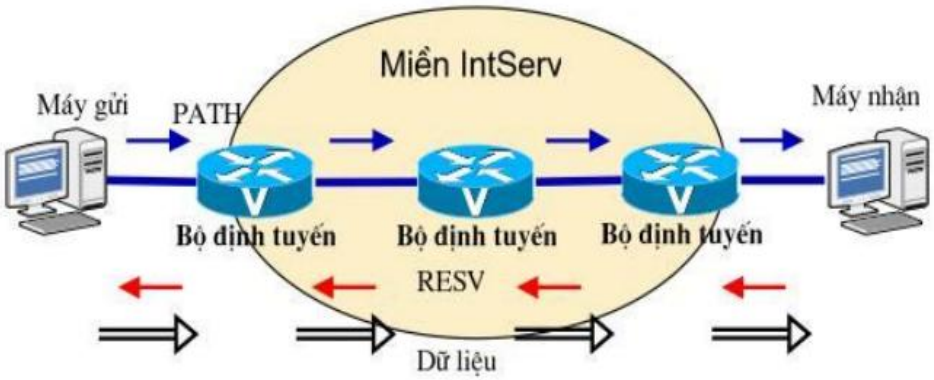
cả các nút nằm trên tuyến đường của gói tin.

RSVP không phải là một giao thức định tuyến do đó nó không cần xác định liên kết nào sẽ được dùng để dành trước mà nó dựa vào các giao thức định tuyến bên dưới để xác định tuyến đường cho một luồng. Một khi tuyến đường được xác định, RSVP bắt đầu thực hiện việc dành trước tài nguyên. Trong suốt quá trình thiết lập để dành tài nguyên, RSVP phải được thông qua mô đun điều khiển về chính sách và mô đun quản lý về việc chấp nhận tuyến đường. Mô đun điều khiển về chính sách xác định xem người dùng có đủ thẩm quyền để dành được nguồn tài nguyên hay không. Thiết bị chấp nhận tuyến đường xác định xem nút đó có đủ tài nguyên để cung cấp cho yêu cầu QoS hay không. Nếu cả hai bước kiểm tra đều tốt, các tham số được thiết lập trong bộ phân loại gói và trong bộ lập lịch để đạt được QoS mong muốn. Tiến trình này được thực hiện tại mọi router và máy tính dọc theo tuyến đường.

- *Nguyên lý hoạt động của RSVP:*

Một phiên làm việc của RSVP thường được xác định bởi 3 tham số sau:

+ Địa chỉ đích
+ Nhận dạng giao thức
+ Địa chỉ cổng đích



Hình 1.5. Nguyên lý hoạt động của giao thức dành trước tài nguyên RSVP

Máy gửi gửi thông điệp PATH tới máy nhận cho một luồng hay còn gọi là một phiên truyền thông. Thông điệp PATH chứa các đặc trưng của luồng sẽ được gửi đi, thông điệp PATH đi qua các Router trên đường dẫn tới đích (máy nhận), các Router đăng ký nhận dạng luồng và đặc tính luồng vào cơ sở dữ liệu. Thông điệp phản hồi - RESV (Resevation) được phát ngược trở loại từ máy nhận tới máy gửi, các Router xác nhận và chỉnh sửa thông tin yêu cầu đã được gửi trong thông điệp PATH và RESV. Khi máy nhận nhận được thông điệp PATH, nó gửi trở lại thông điệp RESV. Thông điệp RESV mang thông tin tài nguyên dự trữ của đường dẫn mà gói tin IP sẽ chuyển qua.

IntServ có ưu điểm là đảm bảo QoS tốt, tuy nhiên sử dụng tài nguyên mạng không hiệu quả vì băng thông đặt trước cho một kênh sẽ không sử dụng được cho kênh khác. Do đó IntServ khó áp dụng trong mạng lớn, không thích hợp với môi trường mạng di động không dây. Vì vậy việc áp dụng mô hình IntServ trong thực tế còn nhiều hạn chế.

1.3.2. Mô hình dịch vụ phân loại - DiffServ

Mô hình dịch vụ phân loại được phát triển nhằm mục đích cung cấp các lớp dịch vụ khác nhau cho các lưu lượng trên Internet và nhằm đạt được tính linh động trong quá trình truyền thông. Khác với mô hình IntServ là dựa trên từng luồng dữ liệu, mô hình DiffServ không xử lý từng luồng tin riêng biệt mà sử dụng các cơ chế phân loại, chỉnh dạng và lập lịch để cung cấp các dịch vụ với mức độ đảm bảo QoS khác nhau cho các lớp lưu lượng khác nhau. Và do đó đạt được hiệu quả cho các mạng lớn. Các lớp dịch vụ được gán các mức ưu tiên khác nhau. Việc đảm bảo QoS cho các luồng tin do đó chỉ mang tính tương đối, nghĩa là luồng tin nào ở lớp có mức ưu tiên cao hơn sẽ được phục vụ tốt hơn các luồng tin ở lớp có mức ưu tiên thấp hơn.

Trong DiffServ, băng thông và các tài nguyên mạng khác nhau được chia sẻ giữa các lớp lưu lượng. Mặt khác DiffServ hướng tới xử lý từng vùng dịch vụ phân biệt thay vì xử lý từ đầu cuối tới đầu cuối như trong mô hình dịch vụ tích hợp. DiffServ sử dụng 6 bit trong tiêu đề gói tin làm điểm mã dịch vụ phân loại – DSCP (Differentiated Service Code Point) để phân loại các gói tin của các dịch vụ khác nhau nhằm áp dụng các chính sách ưu tiên khác nhau cho các gói tin của các luồng tin thuộc các lớp khác nhau. Vì vậy DiffServ không cung cấp mức QoS cụ thể. DiffServ có ưu điểm là thích hợp cho các nút mạng có số lượng luồng tin lớn, song vẫn còn nhiều hạn chế về các mặt như: chỉ đảm bảo QoS mang tính tương đối, hiệu quả sử dụng tài nguyên chưa cao, khó thích ứng với các mạng vô tuyến và mạng hỗn hợp…

Diffserv trái ngược với Intserv là dựa trên từng luồng dữ liệu, nó phân loại các gói thành một số lượng không lớn các tập (gọi là các lớp) và do đó đạt được hiệu quả cho các mạng lớn. Các chức năng đơn giản được thực hiện tại router lõi, trong khi các chức năng phức tạp được triển khai tại các router biên. Tính linh động rất là cần thiết vì dịch vụ mới có thể xuất hiện và một số dịch vụ trở lên lỗi thời. Do đó Diffserv không cần thiết phải xác định dịch vụ như là Inserv, thay vào đó, nó cung cấp các thành phần chức năng mà trên đó dịch vụ có thể được xây dựng. Một gói đi và mạng mà không đề cập gì đến dịch vụ và mạng sẽ xác định luồng và cung cấp dịch vụ thích hợp. Việc thông tin giữa người dùng và dịch vụ sẽ nằm trong thỏa thuận mức dịch vụ - SLA (Service Level Agreement) và giàn xếp giữa một luồng xác định trước với Bản Thỏa Thuận về Lưu Lượng. Việc xác định SLA sẽ được cung cấp bao nhiêu tài nguyên sẽ được cấu hình tay.

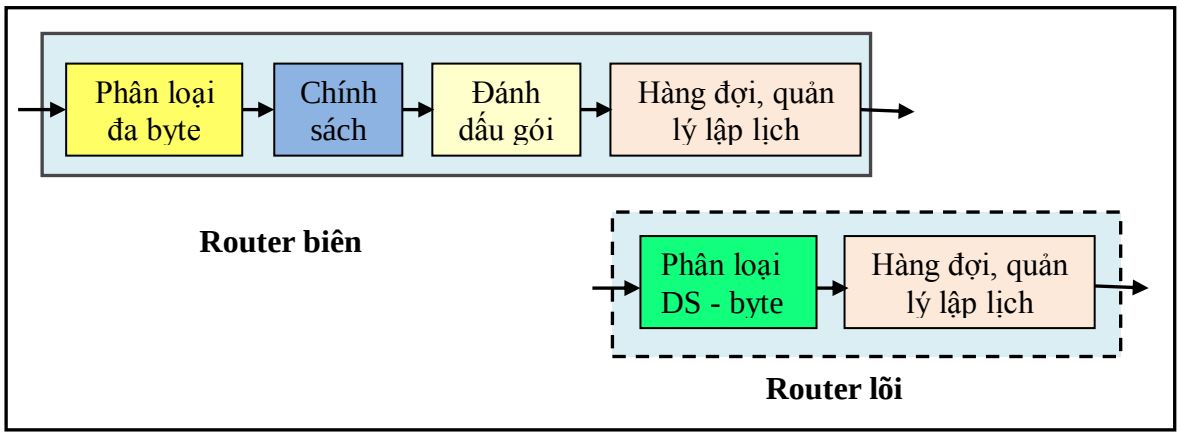
DiffServ định nghĩa một số tham số mà người sử dụng hiểu rõ cho ứng dụng của họ trong SLA như thỏa thuận điều kiện lưu lượng TCA (Traffic Condition Ageement), hồ sơ lưu lượng, các tham số hiệu năng (thông lượng, độ trễ, mất gói), cách thức xử lý các gói tin không phù hợp với thỏa thuận, luật đánh dấu và chia cắt lưu lượng.

Kiến trúc Diffserv bao gồm hai tập các thành phần chức năng:

- Tại biên của mạng, việc phân loại và điều khiển lưu lượng được thực hiện và các gói được phân vào các lớp.
- Tại lõi, một cơ chế phân loại đơn giản được thực hiện. Cơ chế hàng đợi dựa trên lớp

được áp dụng.

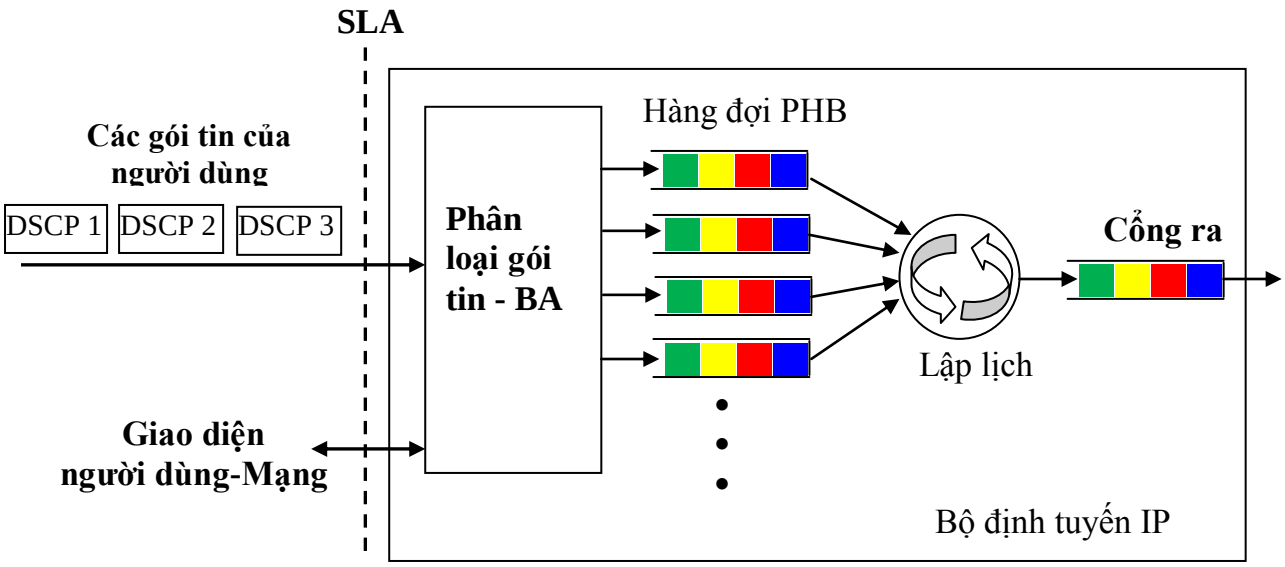
Sơ đồ khối kiến trúc DiffServ được mô tả cụ thể như sau:



Hình 1.6. Xử lý gói trong mô hình DiffServ

➢ *Nguyên lý hoạt động của mô hình dịch vụ phân loại*

Hình vẽ dưới đây mô tả các bước cơ bản trong việc cung cấp các dịch vụ Diffserv.



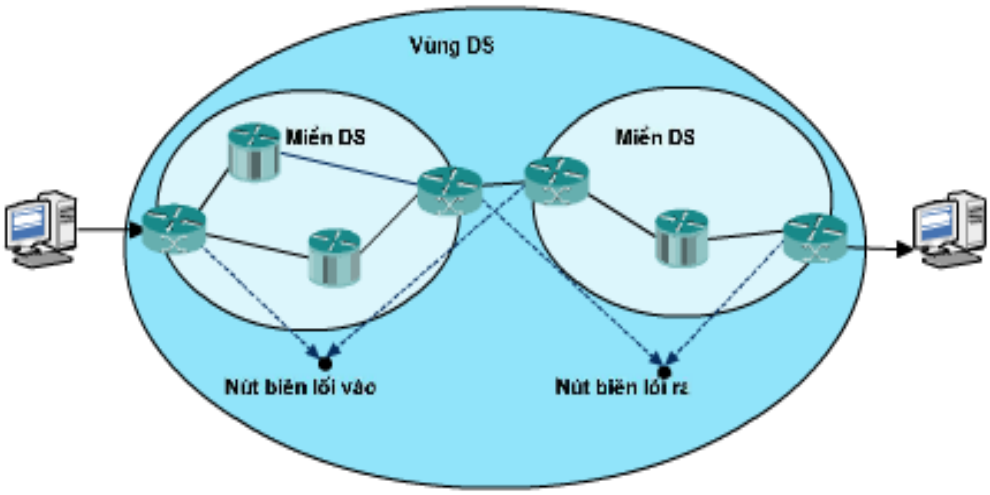
Hình 1.7. Mô hình các bước dịch vụ phân loại Diffserv

Các gói tin của người sử dụng đã được đánh dấu DSCP (hoặc chưa được đánh dấu) đi đến Router, Router kiểm tra trường DSCP của các gói tin và phân loại các gói tin theo phương pháp phân loại kết hợp ứng xử – BA. Khi một gói tin đã được đánh dấu DiffServ, nó được chuyển tiếp tới chặng tiếp theo thông qua một hành vi được gọi là Hành vi theo chặng - PHB (per-hop behavior), liên quan đến lớp của gói tin. PHB ảnh hưởng đến việc vùng đệm của một router và băng thông của liên kết được chia sẻ giữa các lớp lưu lượng cạnh tranh nhau. Một PHB được thực hiện cùng với quản lí hàng đợi và cơ chế lập lịch. Các router kiểm tra các trường DSCP, phân loại nó theo các quá trình đánh dấu và sau đó chuyển gói tới các hàng đợi tương ứng. Một kết nối đầu ra đa hàng đợi với các mức độ ưu tiên khác nhau. Kĩ thuật lập lịch được sử dụng để chuyển các gói ra khỏi hàng đợi và chuyển tới chặng kế tiếp. Kiến trúc DiffServ chỉ định nghĩa các mã DSCP ghi trong trường ToS và các PHB. Còn dịch vụ cụ thể như thế nào là do các nhà

cung cấp dịch vụ quy định.

1.3.2.1. Miền dịch vụ phân loại và điểm mã dịch vụ phân loại

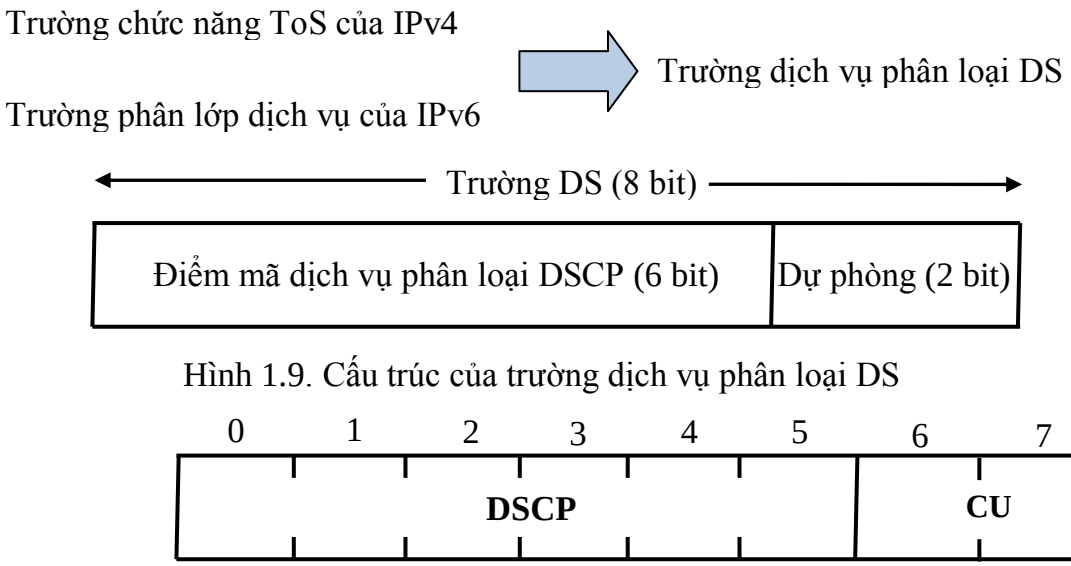
Một miền dịch vụ phân loại– DS (Diffierentiated Service) gồm các nút DS (còn gọi là các bộ định tuyến hỗ trợ cơ chế dịch vụ phân loại) hoạt động với một chính sách cung cấp dịch vụ chung và thiết lập các nhóm hành vi theo chặng – PHB (Per-hop Behavior) được thực hiện trên mỗi nút. Các nút biên DS trong miền DS phân loại và điều khiển lưu lượng đầu vào để đảm bảo các gói tin đi qua miền được đánh dấu thích hợp để lựa chọn một PHB từ một nhóm các PHB được hỗ trợ trong phạm vi miền. Các nút trong miền DS lựa chọn ứng xử chuyển tiếp cho các gói tin dựa trên điểm mã dịch vụ DSCP của chúng, sắp xếp vào một trong các PHB theo yêu cầu. Việc quản trị một miền phải đảm bảo tin cậy để bảo đảm rằng các nguồn tài nguyên tương ứng được cung cấp và được dự trữ để hỗ trợ các SLA yêu cầu.



Hình 1.8. Miền dịch vụ phân biệt DS

Các vùng DS có khả năng hỗ trợ các miền DS dọc theo đường định tuyến nối các miền trong vùng. Các miền DS trong vùng DS có thể hỗ trợ nội bộ trong các nhóm PHB khác nhau và các điểm mã khác nhau để sắp xếp PHB. Tuy nhiên, để cho phép các dịch vụ nối ngang qua miền, các miền DS ngang hàng phải thiết lập mỗi miền một SLA ngang hàng chứa thỏa thuận lưu lượng TCA phù hợp. Một vài miền DS trong một vùng DS có thể kế thừa một chính sách cung cấp dịch vụ chung và có thể hỗ trợ tập chung các nhóm PHB và các cách sắp xếp điểm mã dịch vụ phân loại DSCP, vì vậy có thể loại bỏ qui định lưu lượng giữa các miền DS đó.

DiffServ sử dụng trường kiểu dịch vụ ToS trong tiêu đề IPv4 và trường phân lớp lưu lượng TC (Traffic Class) trong tiêu đề IPv6 để đánh dấu gói. Đối với các bộ định tuyến hoạt động trong miền DS các trường chức năng này được thay bằng trường chức năng dịch vụ phân biệt DS. Trong 8 bit của trường DS, 6 bit được sử dụng cho việc đánh dấu các gói DiffServ và 2 bit cuối hiện tại chưa được sử dụng, để dự phòng. 6 bit được sử dụng cho đánh dấu các gói DiffServ được gọi là điểm mã dịch vụ phân loại DSCP. Hình 1.9 và hình 1.10 dưới đây chỉ ra cấu trúc của trường DS.



Hình 1.9. Cấu trúc của trường dịch vụ phân loại DS

DSCP: Differentiated Services CodePoint

CU: currently unused

Hình 1.10. Cấu trúc của byte ToS

Với 6 bit trong trường DSCP có thể tạo được tổ hợp 64 giá trị DSCP khác nhau. Các điểm mã dịch vụ DSCP được chia làm 3 nhóm như trong bảng 1.1.

Nhóm	Điểm mã DSCP	Ứng dụng
1	xxxxx0	Hoạt động chuẩn
2	xxxx11	Thử nghiệm/ nội bộ
3	xxxx01	Thử nghiệm/ nội bộ/ tiêu chuẩn tương lai

Bảng 1.1. Các nhóm điểm mã dịch vụ phân loại DSCP

Nhóm 1 gồm các điểm mã DSCP sử dụng cho toàn cầu, nhóm 2 và 3 sử dụng cho mục đích thử nghiệm và nội bộ miền DS riêng. Bit cuối cùng (bit thứ 6) của nhóm 1 được ấn định là bit 0, vì vậy mà số phân lớp dịch vụ của nhóm 1 có thể lên tới 32. Nhóm 2 có 2 bít cuối được ấn định là "11", nhóm 3 luôn luôn kết thúc với "01". 4 bit còn lại của cả hai nhóm này được phép nhận các giá trị khác nhau, như vậy số phân lớp dịch vụ tối đa của nhóm 2 và nhóm 3 là 16. Nhóm 2 không yêu cầu các hoạt động tiêu chuẩn và được sử dụng cho thử nghiệm. Nhóm 3 được dành cho việc thử nghiệm và sử dụng trong mạng nội bộ; tuy nhiên điểm khác biệt so với nhóm 2 là nhóm 3 có thể sử dụng cho các hoạt động tiêu chuẩn nếu cần thiết.

1.3.2.2. Hành vi theo chặng PHB (Per-hop Behavior)

Kiến trúc DiffServ định nghĩa hành vi theo chặng PHB cho việc xử lý chuyển tiếp gói tin tại mỗi node mạng áp dụng kết hợp hành vi BA. PHB liên quan đến các đặc tính về chất lượng dịch vụ như độ trễ, độ biến thiên độ trễ hay mất gói của gói tin khi đi qua node dịch vụ DiffServ.

Các node dịch vụ DiffServ sẽ ánh xạ các gói tin đến các chặng PHB tương ứng với các giá trị DSCP của nó. Bảng 1.2 biểu diễn việc ánh xạ giữa PHB và DSCP. DiffServ

không hoàn toàn có chức năng ánh xạ PHB đến DSCP mà nó chỉ thực hiện công việc này khi được yêu cầu. Các nhóm PHB là thành phần của các đặc tính DiffServ đó là: PHB chuyển tiếp nhanh (Expedited Forwarding), PHB chuyển tiếp đảm bảo (Assured Forwarding), PHB chọn lớp (Class Selector) và PHB mặc định (Default). Một node chuyển mạch có thể được hỗ trợ nhiều nhóm PHB tương tự nhau. Các node thực thi các PHB này sẽ sử dụng cơ chế đệm và lập lịch gói tin.

Bảng 1.2. Ánh xạ giữa PHB và DSCP

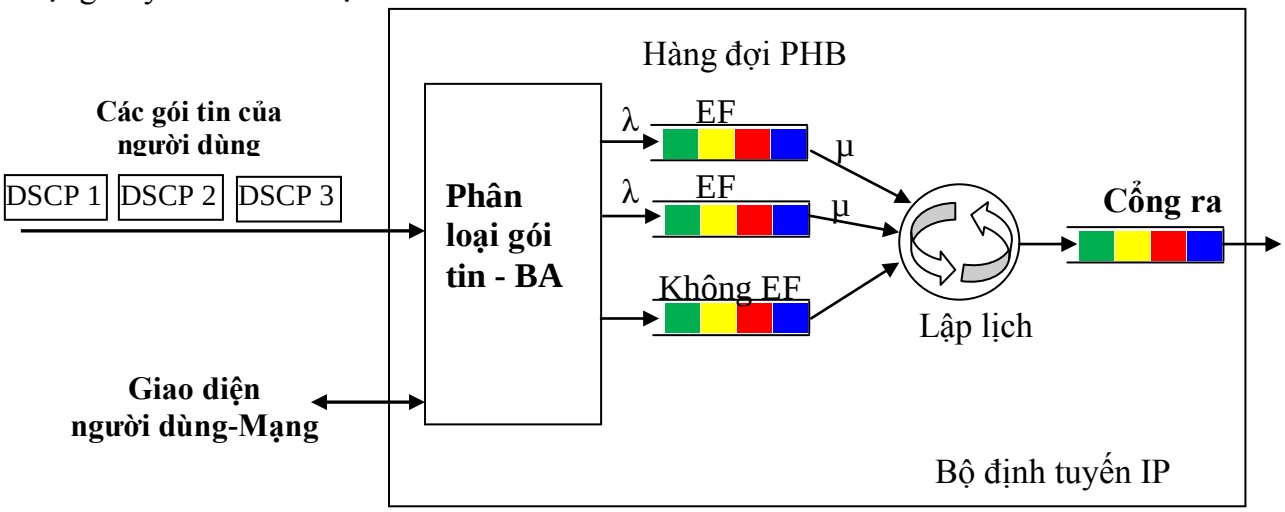
PHB	DSCP (thập phân)	DSCP (nhị phân)
EF	46	101110
AF43	38	100110
AF42	36	100100
AF43	34	100010
AF33	30	011110
AF32	28	011100
AF31	26	011010
AF23	22	010110
AF22	20	010100
AF21	18	010010
AF13	14	001110
AF12	12	001100
AF11	10	001010
CS7	56	111000
CS6	48	110000
CS5	40	101000
CS4	32	100000
CS3	24	011000
CS2	16	010000
CS1	8	001000
Mặc định	0	000000

- *PHB chuyển tiếp nhanh - EF (Expedited Forwarding)*

Hành vi theo chặng EF là hành vi có độ trễ, độ biến thiên độ trễ và tỉ lệ mất gói thấp mà một node DiffServ có thể thực thi. Vì vậy EF PHB được sử dụng cho những luồng có độ ưu tiên rất cao. Nó có thể được thực hiện bằng việc sử dụng các thuật toán như CBQ (Class Based Queue) hoặc sử dụng hàng đợi ưu tiên đơn lẻ. Một bộ định tuyến EF phải đảm bảo lưu lượng EF được đưa đến những bộ nhớ đệm nhỏ vì rung pha và trễ gây nên bởi thời gian mà gói sử dụng trong bộ nhớ đệm và hàng đợi.

Khi xảy ra hiện tượng quá tải, nút biên miền DS không cho phép lưu lượng dạng này đi vào trong miền vì nó là nguyên nhân gây tắc nghẽn tại các bộ định tuyến trong

miền DS. Vấn đề này được điều chỉnh bởi thỏa thuận mức dịch vụ SLA và xác định lưu lượng truyền có điều kiện.



Hình 1.11. Xử lý chuyển tiếp nhanh EF

Chuyển tiếp nhanh EF khả thi nếu băng thông đầu ra và kích thước bộ nhớ đệm đủ để các luồng lưu lượng ra với tốc độ phục vụ μ. Tốc độ phục vụ μ luôn lớn hơn tốc độ đầu vào λ tại các bộ đệm EF.

- *PHB chuyển tiếp đảm bảo - AF (Assred Forwarding)*

DiffServ được thực hiện trong NS2 theo nhóm chuyển tiếp đảm bảo. Một gói tin phụ thuộc vào một luồng có thể nhận 3 mức ưu tiên cùng với luồng đó, được dùng để cung cấp xác suất mất gói thấp hơn cho các gói tin đồng bộ trong kết nối TCP. Do không giống các gói tin khác, việc các gói tin đồng bộ bị mất có thể gây gián đoạn liên lạc (thời gian time-out) rất dài. Ngoài việc phân biệt mỗi luồng, thì tất cả các luồng được phân thành các lớp (tối đa là 4 lớp) và các lớp khác nhau sẽ nhận được mức độ xử lý khác nhau.

Chuyển tiếp đảm bảo với đặc điểm phân phối dữ liệu đảm bảo với khả năng mất gói thấp là điều kiện tốt nhất khi sử dụng các giao thức không thực hiện xử lý sửa lỗi hoặc không có giải pháp truyền lại gói.

AF là một nhóm PHB với 4 lớp đảm bảo chuyển tiếp gói khác nhau với 3 mức độ ưu tiên loại bỏ gói tin, mỗi lớp được gán một băng thông và khoảng nhớ đệm xác định. Nếu một gói phải bị loại bỏ, bộ định tuyến có cách nhận biết gói nào bị loại bỏ đầu tiên. Ngoài ra, mỗi lớp chuyển tiếp được phân bổ một số lượng cực nhỏ băng thông và bộ nhớ đệm. Nếu bộ nhớ đệm đầy, thì quá trình loại bỏ gói sẽ bắt đầu theo trật tự loại bỏ theo mức ưu tiên. Các phân lớp AF được thể hiện trên hình 1.12 và trên bảng 1.3:

Xác suất rơi gói tin

AF31	AF32	AF33	AF34
AF21	AF22	AF23	AF24
AF11	AF12	AF13	AF14

Phân lớp

Hình 1.12. Các phân lớp PHB chuyển tiếp đảm bảo

Bảng 1.3. Chi tiết các phân lớp PHB chuyển tiếp đảm bảo - AF

Lớp PHB	Phân lớp	Độ ưu tiên hủy gói	DSCP
AF4	AF41	Thấp	100010
	AF42	Trung bình	100100
	AF43	Cao	100111
AF3	AF31	Thấp	011010
	AF32	Trung bình	011100
	AF33	Cao	100010
AF2	AF21	Thấp	010010
	AF22	Trung bình	010100
	AF23	Cao	010110
AF1	AF11	Thấp	001010
	AF12	Trung bình	001100
	AF13	Cao	001110

Các nhóm chuyển tiếp đảm bảo AF hoạt động phụ thuộc lẫn nhau và không chứa đặc tính như độ trễ hay độ trượt. Việc mỗi nhóm cung cấp các đảm bảo dịch vụ phục vụ phụ thuộc vào các tài nguyên của các node, số lượng các luồng đến tại các node và các ưu tiên mức loại bỏ gói. Các tài nguyên tại các node chính và băng thông và không gian bộ đệm.

- *PHB lựa chọn lớp - CS (Class Selectors)*

DiffServ định các lựa chọn lớp CS để đưa ra tính tương thích ngược với việc sử dụng mức ưu tiên IP trong trường ToS của IPv4 header. Các CS có giá trị cao hơn sẽ có xác suất chuyển tiếp lớn hơn.

PHB	DSCP (thập phân)	DSCP (nhị phân)	Tên bit ưu tiên IP	Ưu tiên IP (nhị phân)	Ưu tiên IP (thập phân)
CS7	56	111000	Network Control	111	7
CS6	48	110000	Internetwork Control	110	6
CS5	40	101000	Critic/ECP	101	5
CS4	32	100000	Flash Override	100	4
CS3	24	011000	Flash	011	3
CS2	16	010000	Immidiate	010	2
CS1	8	001000	Priority	001	1
CS0	0	000000	Routine	000	0

Bảng 1.4. Quan hệ giữa giá trị ưu tiên IP và bộ lựa chọn lớp CS

- *PHB mặc định (Default PHB)*

PHB mặc định là một dịch vụ Best-Effort mà một miền DiffServ cung cấp. Miền DiffServ sẽ chuyển tiếp càng nhiều gói tin càng tốt, càng sớm càng tốt. Không có các đặc tính về độ trễ, biến thiên độ trễ và tỷ lệ mất gói. Việc thực hiện hành vi PHB khác sẽ ngăn cản hoạt động của các ứng dụng của PHB mặc định.

Chương 2. CÁC CHIẾN LƯỢC QUẢN LÝ HÀNG ĐỢI ĐỘNG AQM

Quản lý hàng đợi tích cực là một trong các giải pháp cho điều khiển tránh tắc nghẽn đảm bảo truyền thông liên tục và hiệu quả trên mạng Internet. Có rất nhiều thuật toán được đưa ra trong kĩ thuật quản lý hàng đợi như các thuật toán lập lịch hay các thuật toán quản lý bộ đệm. Nội dung chính của chương này là trình bày về chiến lược quản lý hàng đợi động (AQM) và một số chiến lược quản lý hàng đợi tích cực dựa theo kích thước hàng đợi và dựa vào tải nạp.

2.1. Cách tiếp cận truyền thống và hiệu quả

Kỹ thuật truyền thống và là kỹ thuật đơn giản nhất để quản lý kích thước hàng đợi là dựa vào cơ chế FIFO. Theo cơ chế này, tất cả các gói tin đến được xếp vào hàng đợi; khi hàng đợi đầy thì các gói tin đến sau đều bị loại bỏ; để chọn các gói tin truyền đi thì gói tin nào đến trước được phục vụ trước. Trong bộ mô phỏng NS, kỹ thuật này được cài đặt với tên gọi "DropTail". Do tính đơn giản của nó, kỹ thuật này được sử dụng nhiều năm trên Internet, tuy nhiên nó có 2 nhược điểm cơ bản được trình bày dưới đây.

2.1.1. Hiện tượng Lock-Out và Global Synchronization

Trong một vài tình huống đặc biệt, cơ chế "DropTail" cho phép một hoặc một vài dòng lưu lượng độc quyền chiếm hàng đợi (buffer), làm cho các gói tin của các kết nối khác không thể được nhận vào, tức là chúng bị "Lock-out".

Hiện tượng "lock-out" thường dẫn đến việc bên gửi của các kết nối TCP đi qua hàng đợi đó bị timeout khi đó theo thuật toán tránh tắc nghẽn, chúng sẽ đồng thời giảm kích thước cửa sổ phát và thực hiện rút lui theo hàm mũ theo thuật toán tránh tắc nghẽn, làm cho lưu lượng trên mạng giảm mạnh. Đó là hiện tượng đồng bộ (giảm lưu lượng) toàn cầu - "global synchronization", gây lãng phí dải thông của mạng.

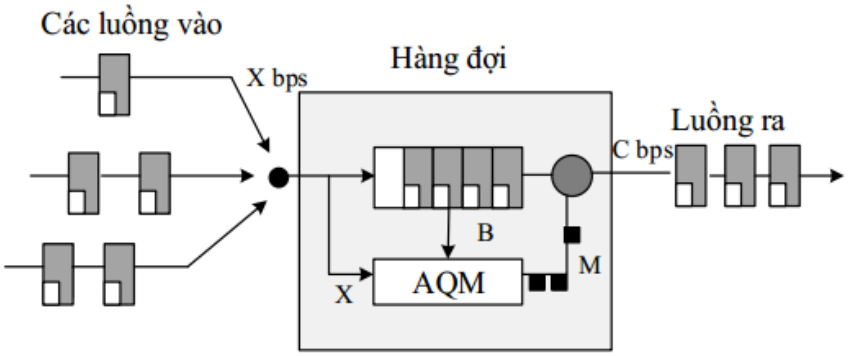
2.1.2. Hiện tượng Full Queues

Với "Tail-Drop", hàng đợi có thể thường xuyên ở trạng thái đầy hoặc gần đầy trong khoảng thời gian dài, vì việc loại bỏ gói tin đến chỉ thực hiện khi hàng đợi đã đầy. Ngoài ra, vì lưu lượng trên Internet thường xuyên có sự bùng nổ, nên hàng đợi tại các nút mạng phải có kích thước đủ lớn để có thể hấp thu được sự bùng nổ dữ liệu. Vì vậy, việc sử dụng "Tail-Drop" sẽ dẫn đến việc độ trễ và thăng giáng độ trễ của các gói tin đi qua mạng có giá trị cao. Ngoài "DropTail", hai phương pháp khác có thể được áp dụng khi hàng đợi đầy là loại bỏ ngẫu nhiên (Random Drop) hoặc loại bỏ ở đầu khi đầy (Drop Front). Theo phương pháp Random Drop, router loại bỏ ngẫu nhiên một gói tin trong hàng đợi khi hàng đợi đầy để dành chỗ cho gói tin đến. Còn với phương pháp Drop Font thì ngược lại với DropTail, router sẽ loạt bỏ gói tin ở đầu hàng đợi khi đầy và một gói tin đến. Cả hai cơ chế này có thể giải quyết được vấn đề "lock-out" nhưng không giải quyết được vấn đề thứ hai là "full-queues" như đã nói ở trên.

2.2. Chiến lược quản lý hàng đợi động AQM

Các chiến lược truyền thống chỉ thực hiện loại bỏ gói tin khi hàng đợi đầy. Khi tốc độ các gói tin đến cao hơn tốc độ gói tin đi của router, kích thước hàng đợi sẽ tăng dần lên, cuối cùng có thể vượt quá không gian cho phép của bộ đệm. Nếu một gói tin đến và

lúc đó hàng đợi đầy, gói tin sẽ bị loại bỏ. Phương pháp được sử dụng phổ biến là loại bỏ đuôi - DT (Drop Tail) lưu lượng. Đây là kiểu hàng đợi thụ động, các gói tin tự động bị loại bỏ khi hàng đợi đầy. Ưu điểm chính của phương pháp này là xử lý đơn giản, tuy nhiên phương pháp này có thể gây ra các ảnh hưởng xấu tới việc đồng bộ trong giao thức TCP. Vì vậy, cơ chế DT tương tác kém với các cơ chế điều khiển tắc nghẽn của TCP và dẫn đến hiệu suất thấp. Ý tưởng chính của kỹ thuật tránh tắc nghẽn là dự đoán trước khả năng tắc nghẽn và đưa ra một số hoạt động điều khiển để chống lại hoặc giảm thiểu khả năng tắc nghẽn. AQM là phương pháp chủ động thông báo với bên gửi khi mới bắt đầu có tắc nghẽn, trước khi xảy ra tràn bộ đệm. Bằng cách sử dụng cơ chế AQM, bên gửi được thông báo sớm về tắc nghẽn và có thể phản ứng phù hợp. Hình 2.1 trình bày mô hình quản lý hàng đợi tích cực trong mạng TCP/IP [27].



Hình 2.1. Mô hình quản lý hàng đợi tích cực

Khi các gói tin từ các nguồn khác nhau đến nút mạng với thông lượng *X (bps*), nút mạng sắp xếp các gói vào các hàng đợi (*B*) và lập lịch để đưa các gói tin đến vùng đệm (*M*) của các cổng ra thích hợp với thông lượng (*C bps*). Sau đó, gói tin được chuyển đến các nút mạng tiếp theo để đến được máy nhận.

Hàng đợi đầy chính là dấu hiệu của tắc nghẽn. Giải pháp cho vấn đề hàng đợi đầy này là cho phép gateway loại bỏ các gói tin trước khi hàng đợi đầy, vì vậy các thực thể đầu cuối có thể phản ứng lại tắc nghẽn khi hiện tượng này mới chớm có dấu hiệu xuất hiện. Cách tiếp cận này được gọi là quản trị hàng đợi động – AQM. Theo chiến lược này, gateway sẽ quyết định cách thức loại bỏ gói tin trong hàng đợi của nó. Điều cần phải nhấn mạnh là, AQM chỉ có tác dụng đối với các giao thức vận chuyển có áp dụng các cơ chế điều khiển lưu lượng và tắc nghẽn kiểu end-to-end, như TCP, không có tác dụng đối với các giao thức vận chuyển không áp dụng cơ chế điều khiển lưu lượng, như UDP chẳng hạn.

Nhìn chung, các chiến lược AQM đem lại các lợi ích sau:

2.2.1. Giảm số gói tin bị loại bỏ tại router

Sự bùng nổ các gói tin là không thể tránh được trong mạng chuyển mạch gói, đặc biệt là trong các ứng dụng đa phương tiện tương tác thời gian thực. Bằng việc giữ kích thước trung bình của hàng đợi nhỏ, AQM sẽ cung cấp dung lượng lớn hơn để hấp thu các bùng nổ xảy ra một cách ngẫu nhiên mà không cần loại bỏ hàng loạt gói tin đến khi hàng đợi bị đầy.

Với chiến lược quản lý hàng đợi truyền thống, các gói tin bị loại tăng nhanh khi hàng đợi đầy. Điều này gây ra những bất lợi sau: 1/ Sự giảm lưu lượng toàn cầu dẫn tới hiệu suất của việc sử dụng đường truyền thấp và thông lượng toàn tuyến của mạng giảm; 2/ TCP sẽ gặp khó khăn hơn trong việc khôi phục hàng loạt gói tin bị mất hơn là khôi phục từng gói tin bị mất một cách đơn lẻ; 3/ Lãng phí dải thông của mạng. Bằng việc giữ kích thước trung bình của hàng đợi nhỏ, AQM sẽ cung cấp khoảng không gian vùng đệm lớn hơn để có thể hấp thu các bùng nổ lưu lượng đưa vào mạng xảy ra một cách ngẫu nhiên mà không phải loại bỏ hàng loạt gói tin đến khi hàng đợi bị đầy.

2.2.2. Giảm độ trễ

Bằng việc giữ kích thước hàng đợi trung bình nhỏ, AQM có thể giảm độ trễ trung bình của các gói tin một cách đáng kể (theo công thức Little). Điều này đặc biệt quan trọng với các ứng dụng tương tác như Web, Telnet, hoặc những ứng dụng audio-video tương tác thời gian thực,.. những ứng dụng mà hiệu quả được đánh giá là tốt khi độ trễ thấp.

2.2.3. Tránh hiện tượng Lock-Out

Như trên đã trình bày, Lock-out là hiện tượng gói tin đến không vào được hàng đợi vì không còn chỗ trống. AQM đảm bảo rằng hầu như luôn luôn có vị trí trống trong bộ đệm khi một gói tin đến, do đó tránh được hiện tượng Lock-Out. Cũng với lý do đó, AQM có thể làm cho router không chống lại các luồng có thông lượng thấp nhưng có độ đột biến cao.

Trong chương này chúng ta sẽ nghiên cứu và đánh giá một số thuật toán tiêu biểu thuộc họ AQM, đó là RED, A-RED; chương tiếp theo sẽ trình bày về BLUE, là một chiến lược quản lý hàng đợi dựa trên lưu lượng gói tin đến - còn gọi là tải nạp và đây là nội dung chính của Luận văn này.

2.3. Chiến lược RED

Trong các cơ chế quản lý hàng đợi dựa trên chiều dài hàng đợi, hiện tượng tắc nghẽn được thể hiện dựa trên độ dài tức thời hoặc trung bình của hàng đợi và mục đích của quá trình điều khiển làm ổn định độ dài hàng đợi tại các nút mạng. Quản lý tắc nghẽn cho phép các thành phần mạng điều khiển tắc nghẽn bằng cách xác định thứ tự các gói được truyền đi dựa vào các quyền ưu tiên hoặc là các dịch vụ gán cho các gói tin đó. Nó cần tạo ra hàng đợi, chỉ định các gói tin tới hàng đợi và thiết lập các gói tin trong hàng đợi. Vấn đề quản lý hàng đợi cần làm sao để phòng tránh được tắc nghẽn trong mạng, do kích thước của hàng đợi là giới hạn nên chúng có thể bị đầy và tràn. Một khi hàng đợi bị đầy thì bất kì một gói tin nào đến đều không thể đưa được vào trong hàng đợi và nó sẽ bị loại bỏ. Việc loại bỏ này là loại bỏ đằng đuôi, điều này có nghĩa là bất kì nói tin nào đến (thậm chí các gói có độ ưu tiên cao) khi hàng đợi đã đầy đều bị loại bỏ. Do đó có hai yếu tố cần phải được thực hiện: thứ nhất là luôn đảm bảo hàng đợi không bao giờ đầy để có đủ chỗ cho các gói tin có độ ưu tiên cao. Thứ hai là phải có một cơ chế loại bỏ các gói có độ ưu tiên thấp trước các gói có độ ưu tiên cao. Và thuật toán phát hiện sớm ngẫu nhiên RED là một trong những kĩ thuật để ngăn ngừa tắc nghẽn và nó đáp ứng được hai yếu tố

trên. Thuật toán RED tận dụng các tính năng tác động ngược của TCP và rất phù hợp trong mạng IP. Các tác động ngược cho phép cắt giảm lưu lượng cấp phát vào mạng khi tốc độ đường truyền chậm. Tận dụng tính năng này, thuật toán RED thực hiện loại bỏ các gói tin ngẫu nhiên thậm chí trước khi sự tắc nghẽn xảy ra.

Thuật toán RED lần đầu tiên được đề xuất vào năm 1993 bởi Sally Floyd và Van Jacobson cho chức năng quản lý hàng đợi tích cực (AQM), sau đó nó được chuẩn hoá lại theo yêu cầu của IETF [14]. RED có khả năng chống hiện tượng đồng bộ toàn cục của các luồng TCP, duy trì khả năng đạt thông lượng qua hàng đợi RED cao cũng như độ trễ thấp cùng với cách đối xử công bằng giữa các kết nối TCP đi qua hàng đợi. RED gateway thực hiện loại bỏ gói tin trong hàng đợi theo chiến lược AQM, ngoài ra nó còn đánh dấu vào trường ECN trong gói tin TCP, để báo cho bên gửi biết về hiện tượng tắc nghẽn có dấu hiệu sắp xảy ra, cần có phản ứng tích cực (việc đánh dấu ECN là một tuỳ chọn của RED). Khi có dấu hiệu của tắc nghẽn xảy ra trong mạng, các bộ đệm của router được điền đầy và router bắt đầu loại bỏ các gói. Có hai vấn đề nan giải trong mạng: thứ nhất các gói bị mất sẽ phải được truyền lại, việc này làm tăng tải trong mạng đồng thời phát sinh ra trễ các luồng lưu lượng. Một vấn đề thứ hai là xảy ra hiện tượng đồng bộ toàn cục trên tất các các luồng. Với một sự bùng nổ lưu lượng dạng bó (burst), các hàng đợi được điền đầy và các gói đến sau bị loại bỏ. Kết quả là kết nối nhiều kết nối TCP bị ảnh hưởng (mất gói tin) và chyển sang chế độ khởi đầu chậm. Việc có nhiều kết nối TCP cùng chuyển sang chế độ khởi đầu chậm tại một thời điểm và cùng thoát khỏi chế độ này do đó sẽ gây ra thêm các bó lưu lượng lớn.

Một giải pháp cho việc có nhiều lưu lượng dạng bó đến router là xây dựng các bộ đệm đủ lớn để có thể nhớ đệm được các bó lưu lượng này tránh việc phải loại bỏ các gói. Nhưng giải pháp này không khả thi vì bộ đệm có kích thước càng lớn thì độ trễ hàng đợi càng lớn, làm giảm chất lượng dịch vụ và nếu có quá nhiều bó lưu lượng lớn đến kế tiếp nhau thì kích thước bộ đệm không đủ lớn để giữ được tất cả các lưu lượng này, điều này dễ gây ra tắc nghẽn. Một giải pháp tối ưu để giải quyết tắc nghẽn là thông báo cho các nguồn sinh luồng gói tin TCP tại thời điểm bắt đầu xảy ra tắc nghẽn để giảm tốc độ đến, nếu cần thiết thì giảm tốc độ các luồng khác. Do đó trong trường hợp tắc nghẽn thì giảm tải lưu lượng TCP đưa vào mạng mà không gây ra hiện tượng đồng bộ toàn cục.

- *Mục đích thiết kế thuật toán RED*

+ *Tránh tắc nghẽn*: RED được thiết kế để tránh tắc nghẽn hơn là giải quyết nó. Do đó RED được sử dụng để phát hiện ra tắc nghẽn ngay khi nó mới bắt đầu hình thành để duy trì mạng trong miền độ trễ thấp và thông lượng lớn.

+ *Tránh đồng bộ toàn cục*: Khi có dấu hiệu tắc nghẽn xảy ra trong mạng, router sẽ phải quyết định xem kết nối nào để gửi thông báo phản hồi. Bằng việc phát hiện sớm tắc nghẽn và chỉ thông báo cho các kết nối khi cần thiết do đó tránh được hiện tượng đồng bộ toàn thể luồng TCP.

+ *Giữ ổn định kích thước hàng đợi trung bình*: RED có thể điều khiển được kích thước hàng đợi trung bình do đó điều khiển được trễ hàng đợi.

2.3.1. Nguyên tắc hoạt động

Để đạt được những mục đích thiết kế của thuật toán như đã trình bày ở phần trên thì RED gateways phải thực hiện các công việc sau:

- Phát hiện sớm tắc nghẽn, bằng cách thường xuyên giám sát kích thước hàng đợi. Tránh tắc nghẽn bằng cách loại bỏ các gói tin trong hàng đợi theo một số quy tắc nhất định, để giữ cho kích thước hàng đợi trung bình đủ nhỏ, làm cho mạng hoạt động ở vùng có độ trễ thấp và thông lượng cao, trong khi vẫn cho phép kích thước hàng đợi dao động trong một miền nhất định để hấp thụ các thăng giáng lưu lượng ngắn hạn.

- Báo hiệu sớm tắc nghẽn tới nguồn phát. Khi có dấu hiệu của tắc nghẽn, ngoài việc dựa trên biện pháp loại bỏ ngẫu nhiên các gói tin nêu trên, cần áp dụng biện pháp đánh dấu vào trường ECN của gói tin với một xác suất nhất định. Các gói tin này được lựa chọn ngẫu nhiên để cho phép truyền đi cùng với dấu hiệu tắc nghẽn được đánh dấu để thông báo cho thực thể gửi TCP biết nhằm giảm lưu lượng đưa vào mạng (thông tin ECN được bên đích gửi cho bên nguồn trong gói tin ACK). Việc đánh dấu được thực hiện ngẫu nhiên để tránh hiện tượng đồng bộ toàn cục và không chống lại các dòng lưu lượng có giá trị trung bình thấp nhưng độ thăng giáng cao.

2.3.2. Giải thuật RED

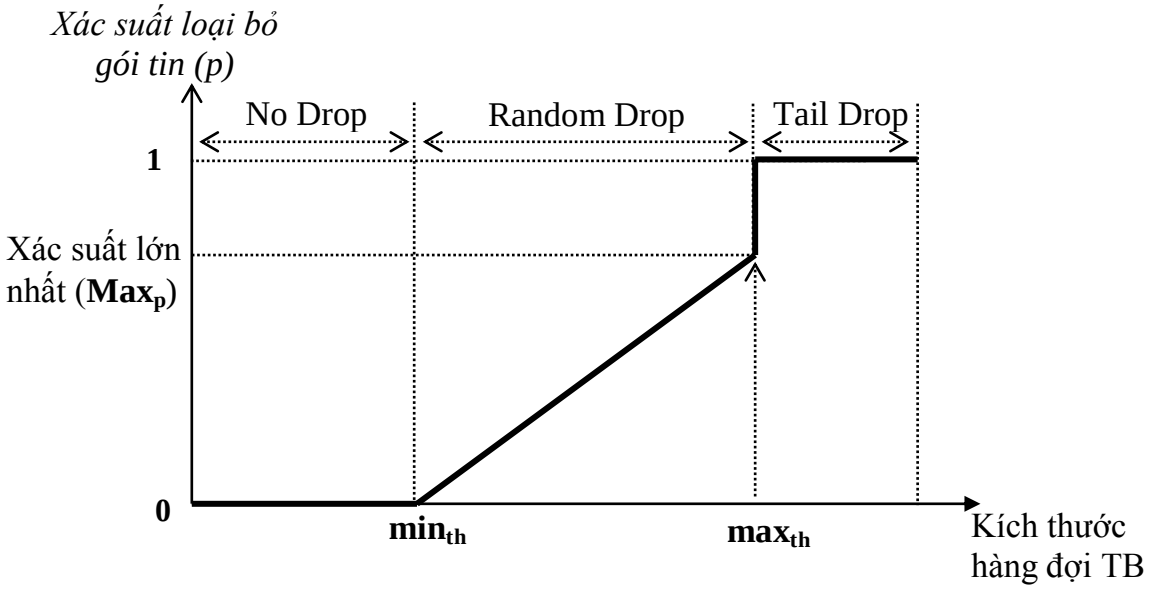
Thuật toán RED (Random Early Drop/Detection - Loại bỏ/Phát hiện sớm ngẫu nhiên) được sử dụng để điều khiển tránh tắc nghẽn dữ liệu tại các hệ định tuyến bằng cách kiểm tra độ dài trung bình hàng đợi với các gói dữ liệu đến và quyết định đánh dấu (để loại bỏ sau này nếu cần thiết) hoặc loại bỏ gói dữ liệu đến với xác suất tăng dần khi độ dài trung bình của hàng đợi vượt quá giá trị một ngưỡng xác định.

RED sẽ tính toán kích thước hàng đợi trung bình dựa trên bộ lọc thông thấp (Low-Pass Filter)), giá trị trung bình này còn được gọi là trung bình dịch chuyển có trọng số tăng theo hàm mũ - EWMA (Exponential Weighted Moving Average). Kích thước hàng đợi trung bình được so sánh với hai giá trị ngưỡng: ngưỡng dưới min_{th} và ngưỡng trên max_{th} để quyết định việc đánh dấu hoặc loại bỏ các gói tin trong hàng đợi. Hoạt động của RED được mô tả bởi ba quy tắc để xác định vị trí của mỗi gói tin gửi đến:

+ Khi kích thước hàng đợi trung bình nhỏ hơn ngưỡng dưới thì không có gói tin nào bị đánh dấu hay loại bỏ (hay gán xác suất loại bỏ gói bằng 0). Đây là trường hợp hoạt động bình thường.

+ Khi kích thước hàng đợi trung bình lớn hơn ngưỡng trên thì tất cả các gói đến đều bị loại bỏ. Khi các gói bị loại bỏ hoặc nếu tất cả các nguồn cùng hợp tác với nhau thì kích thước hàng đợi trung bình sẽ không vượt quá giá trị ngưỡng trên.

+ Khi kích thước hàng đợi trung bình biến thiên từ min_{th} đến max_{th} thì mỗi gói tin đến được đánh dấu hoặc loại bỏ một cách ngẫu nhiên tùy theo một hàm xác suất p.



Hình 2.2. Mối quan hệ giữa xác suất loại bỏ gói và kích thước hàng đợi trung bình.

Để cho RED hoạt động tốt thì phải chọn các giá trị min_{th}, max_{th} và hàm xác suất p như thế nào cho phù hợp. Giá trị min_{th} phải đủ lớn để đảm bảo rằng đường liên kết để dữ liệu đi được sử dụng với hiệu suất cao.Giá trị max_{th} phải lớn hơn min_{th}, ít nhất là phải gấp đôi. Nếu không thì RED cũng gây ra những ảnh hưởng như kiểu hàng đợi "cắt bớt phần đuôi" (DropTail hay FIFO).

Hình 2.2 có thể được xem như 3 pha của quá trình tránh tắc nghẽn:

- Pha thứ nhất: hoạt động bình thường
- Pha thứ hai: tránh tắc nghẽn
- Pha thứ ba: điều khiển tắc nghẽn

Nhìn chung giải thuật RED có 2 thuật toán tách biệt:

1) Tính kích thước hàng đợi trung bình: Giải thuật này quyết định mức độ bùng nổ cho phép trong hàng đợi tại gateway. Nó đưa ra bản mô tả các chu kì khi hàng đợi rỗng (chu kì rỗi) bằng cách đánh giá số lượng m của các gói nhỏ có thể được truyền trong suốt chu kì rỗi bởi router. Sau mỗi chu kì rỗi router lại tính toán kích thước hàng đợi trung bình như thể m gói đã đến được hàng đợi rỗng trong suốt chu kì đó.

2) Tính xác suất loại bỏ gói tin.

Giải thuật tính xác suất loại bỏ gói tin phải đảm bảo sao cho các gói tin được đánh dấu tại những khoảng không gian đều nhau, mục đích để tránh hiện tượng đồng bộ toàn cục các luồng TCP, trong khi vẫn giữ kích thước hàng đợi trung bình ở một giới hạn nhất định. Độ chiếm giữ hàng đợi lớn thì xác suất loại bỏ gói càng cao, độ chiếm giữ hàng đợi càng gần giá trị max_{th} thì xác suất loại bỏ gói dần tiến tới giá trị max_p.

Giải thuật tổng quát của RED gateway được mô tả như ở hình 2.3.

For each packet arrival
Caculate the average queue size *avg*
If $min_{th} \leq avg < max_{th}$
Caculate propability p_a
With propability p_a:
mark the ariving packet
else if $max_{th} \leq avg$
Mark or drop the ariving packet
Else
Accept the arriving packet.

Hình 2.3. Giải thuật tổng quát của RED

Khi kích thước hàng đợi trung bình nằm trong khoảng giá trị min_{th} và max_{th} thì mỗi gói đến đều được đánh dấu hoặc loại bỏ bằng một xác suất P_a, đây là một chức năng của kích thước hàng đợi trung bình. Tại mỗi thời điểm có một gói bị đánh dấu và xác suất gói bị đánh dấu từ một kết nối điển hình tỉ lệ với băng thông chia sẻ kết nối tại mỗi router.

Giải thuật chi tiết của RED tại gateway được mô tả như hình 2.4.

Khởi tạo:
$avg \leftarrow 0$
$count \leftarrow -1$
for mỗi gói tin đến
Tính kích thước hàng đợi trung bình avg:
if hàng đợi không rỗng
$avg \leftarrow (1-w_q)\ avg + w_q * q$
else
$m \leftarrow f(time - q\_time)$
$avg \leftarrow (1-w_q)^m\ avg$
if $min_{th} \leq avg < max_{th}$
$cout ++$
Tính toán xác suất p_a:
$p_b \leftarrow max_p(avg - min_{th}) / (max_{th} - min_{th})$
$p_a \leftarrow p_b / (1 - count * p_b)$
Với xác suất p_a:
đánh dấu gói tin đến
$count \leftarrow 0$
else if $avg \leq max_{th}$
đánh dấu gói tin đến
$count \leftarrow 0$
else $count \leftarrow -1$
Khi hàng đợi trở nên rỗng
q_time $\leftarrow time$

Hình 2.4. Giải thuật chi tiết của RED

Các biến thay đổi:

avg: kích thước hàng đợi trung bình

q_time: điểm bắt đầu hàng đợi rỗng

count: số lượng các gói đến ngay sau gói cuối cùng bị đánh dấu

Các tham số cố định:

w_q*:* trọng số hàng đợi

min_{th}*:* chiều dài ngưỡng nhỏ nhất của hàng đợi

max_{th}*:* chiều dài ngưỡng lớn nhất của hàng đợi

max_p*:* xác suất loại bỏ tối đa

Các tham số khác:

p_a: xác suất đánh dấu gói tin hiện tại

p_b*:* xác suất đánh dấu hoặc loại bỏ tạm thời

q: kích thước hàng đợi hiện tại

time: thời gian hiện tại

f(t): một hàm tuyến tính của thời gian t

Muốn RED có thể gán một xác suất huỷ bỏ cao hơn khi hàng đợi bị đầy mà không phải huỷ bỏ gói tin của mỗi khi có gói đến thì thay vì sử dụng kích thước thật của hàng đợi tại thời điểm đó, theo giải thuật, mỗi khi có gói đến, sẽ tính kích thước hàng đợi trung bình - *avg* bằng bộ lọc thông thấp (Low Pass Filter) với trọng số hàng đợi w_q, và sử dụng kích thước trung bình này để xác định xác suất. Việc sử dụng kích thước hàng đợi trung bình *avg* với mục đích là để tránh sự dao động quá nhanh của hàng đợi khi có những đợt gửi với thời gian ngắn. Giá trị của *avg* được cập nhật mỗi khi có gói tin gửi đến theo công thức sau:

$$avg = (1 - w_q) * avg + w_q * q$$

Với q là kích thước hàng đợi hiện thời; w_q là trọng số hàng đợi, và $0 \leq w_q \leq 1$. W_q còn được gọi là hệ số làm trơn, w_q càng nhỏ thì mức độ làm trơn càng cao, w_q càng lớn thì *avg* càng bám sát giá trị tức thời của q. Như vậy w_q quyết định độ lớn và độ kéo dài cho phép của sự bùng nổ lưu lượng.

Khi kích thước hàng đợi trung bình *avg* chạy từ min_{th} đến max_{th} thì xác suất p_b thay đổi tuyến tính từ 0 đến max_p:

$$p_b = max_p * (avg - min_{th}) / (max_{th} - min_{th}).$$

Khi có một gói tin đến nó được đánh dấu là gói tin loại bỏ, xác suất của gói tin loại bỏ thay đổi ngay lập tức. Xác suất loại bỏ gói tin tới là:

$$p_a = p_b / (1 - count * p_b)$$

Trong đó count là một bộ đếm dùng để tính toán số gói tin bị đánh dấu. Nó là số lượng các gói tin bị đánh dấu tính từ lần đánh dấu cuối cùng. Giá trị *count* càng lớn thì xác suất đánh dấu càng cao. RED gateway đánh dấu tất cả các gói tới khi kích thước trung bình hàng đợi vượt quá max_{th}. Giá trị max_p sẽ quyết định tần số loại bỏ gói là lớn hay nhỏ, nó quyết định *avg* sẽ nằm ở mức nào trong khoảng từ min_{th} đến max_{th}. Vì vậy tùy từng yêu cầu mà có thể thiết lập max_p cho phù hợp. Thí dụ, để *avg* nằm trong khoảng

giữa min_{th} và max_{th}, giá trị max_p phù hợp là 0.02. Tuy nhiên nếu tắc nghẽn thường xuyên xảy ra, cần chọn max_p lớn hơn, thí dụ bằng 0.1.

Một lựa chọn cho RED gateway là tính toán hàng đợi theo bit hơn là theo các gói. Với sự lựa chọn này thì kích thước trung bình hàng đợi sẽ phản ánh sẽ phản ánh chính xác độ trễ trung bình tại gateway. Khi lựa chọn này được sử dụng thì thuật toán phải được chỉnh sửa để chắc chắn rằng xác suất các gói bị đánh dấu tỉ lệ với kích thước gói theo bit:

$$p_b = \frac{PacketSize}{MaxPacketSize} * maxp * \frac{avg - \min th}{(\max th - \min th)}$$

Trong trường hợp này, một gói tin lớn dễ bị loại bỏ hơn là một gói tin nhỏ.

Nhiệm vụ đánh dấu các gói được thực hiện nhờ vào RED gateway tại mỗi router. RED gateway là thành phần của mạng thực hiện nhiệm vụ đánh dấu gói tin. RED là thuật toán được đưa ra và được điều khiển bởi các RED gateway trong router. Các gateway này thực hiện việc loại bỏ tắc nghẽn bằng việc sử dụng thuật toán RED để tính toán giá trị hàng đợi trung bình. Cổng này có thể thông báo tới các kết nối bị tắc nghẽn, hay việc loại bỏ các gói đến cổng hoặc thiết lập các bit trong phần tiêu đề của các gói. Khi kích thước hàng đợi trung bình vượt quá mức ngưỡng cho phép thì gateway sẽ loại bỏ hoặc đánh dấu các gói đến bằng một hàm xác suất của kích thước hàng đợi trung bình. Các RED gateway giữ cho kích thước hàng đợi trung bình thấp trong khi vẫn cho phép các gói đến dưới dạng bỏ đi vào hàng đợi. Trong suốt thời gian tắc nghẽn xác suất các gateway thông báo cho các kết nối giảm kích thước cửa sổ của nó phải cân xứng với băng thông của các kết nối được chia sẻ qua gateway. Các RED gateway được thiết kế để hỗ trợ các giao thức có cơ chế điều khiển tắc nghẽn lớp truyền tải như giao thức TCP.

2.3.3. Các tham số của RED

Các tham số có ảnh hưởng rất lớn đến đến hiệu quả điều khiển chống tắc nghẽn của thuật toán RED, vì vậy việc thiết lập giá trị cho các tham số có ý nghĩa rất quan trọng, có tính chất quyết định. Phần này chúng tôi sẽ trình bày song song hai cách thiết lập các tham số: định tính (bằng lý luận) và định lượng (bằng mô phỏng) để có thể chọn được một bộ tham số hợp lý nhất, mang lại hiệu quả cao nhất cho thuật toán. Phần mô phỏng đã được các tác giả [1] nghiên cứu rất kỹ bằng NS-2, tôi đã thực hiện lại các mô phỏng đó và thấy rằng các kết quả mà họ đưa ra là hoàn toàn chính xác. Vì vậy ở đây chúng tôi chỉ xin trích dẫn các kết quả đó kèm theo bình luận riêng mà không trình bày lại các mô phỏng.

a. Trọng số hàng đợi wq

RED gateway sử dụng bộ lọc thông thấp để tính toán kích thước hàng đợi trung bình. Theo đó, sự tăng ngắn hạn của kích thước hàng đợi hiện tại do một lưu lượng bùng nổ, hoặc một sự tắc nghẽn thoáng qua sẽ ít ảnh hưởng lớn đến kích thước hàng đợi trung bình. Bộ lọc thông thấp là trung bình dịch chuyển có trọng số tăng theo luỹ thừa - EWMA (Exponential Weighted Moving Average):

$$avg = (1-w_q)\, avg + w_q * q$$

Trọng số hàng đợi w_q đóng vai trò quyết định thời gian của bộ lọc thông thấp. Việc lựa chọn w_q quyết định hằng số thời gian cho quá trình tính toán kích thước hàng đợi trung bình. Nếu w_q quá nhỏ thì kích thước hàng đợi trung bình được đánh giá là phản ứng rất chậm với sự thay đổi kích thước hàng đợi thực. Còn nếu w_q quá cao thì kích thước hàng đợi trung bình được đánh giá là quá gần với kích thước hàng đợi tức thời. Chính vì vậy ta cần lập luận cho việc thiết lập các cận trên hoặc dưới cho tham số w_q.

- *Cận trên cho* w_q

Chúng ta thấy rằng, nếu w_q quá lớn, kích thước trung bình của hàng đợi *avg* luôn bám sát kích thước hàng đợi hiện tại và thăng giáng rất mạnh, điều đó sẽ dẫn đến sự thăng giáng mạnh của thời gian trễ (jitter). Ngoài ra, sự đột biến lưu lượng ngắn hạn cũng sẽ bị ngăn cản mạnh mẽ do hàng đợi dễ bị tràn và mất gói tin. Giả sử ban đầu hàng đợi rỗng (kích thước trung bình bằng 0), sau khi có các gói, số gói tin trong hàng đợi sẽ tăng từ 0 đến L (có L gói tin trong hàng đợi). Sau khi gói tin thứ L đến gateway, kích thước hàng đợi trung bình avg_L được xác định như sau:

$$avg_L = \sum_{i=1}^{L} i w_q (1-w_q)^{(L-i)} = w_q (1-w_q)^L \sum_{i=1}^{L} i \left(\frac{1}{1-w_q}\right)^i$$

Áp dụng kết quả tính tổng $\sum_{i=1}^{L} ix^i = \frac{x+(Lx-L-1)x^{L+1}}{(1-x)^2}$ ta xác định được biên trên đối với w_q như sau:

$$\text{avg}_L = L + 1 + \frac{(1-w_q)^{L+1}-1}{w_q}$$

- *Cận dưới cho* w_q

RED gateway được thiết kế sao cho thuật toán RED luôn giữ kết quả tính toán kích thước hàng đợi trung bình *avg* nằm dưới mức ngưỡng nào đó. Tuy nhiên sẽ không đạt được mục đích nếu như *avg* không phản ánh được một cách hợp lý kích thước hàng đợi trung bình hiện tại. Nếu w_q được thiết lập quá thấp, thì giá trị *avg* sẽ phản ứng rất chậm với sự thay đổi kích thước hàng đợi trong thực tế. Trong trường hợp này, RED gateway không phát hiện thấy sự bắt đầu của tắc nghẽn.

Khi đưa ra giá trị ngưỡng dưới min_{th}, tức là đã cho phép hấp thu bùng nổ đến L gói tin. Sau đó trọng số w_q phải được chọn thoả mãn bất phương trình $avg_L < min_{th}$:

$$avg_L = L + 1 + \frac{(1-w_q)^{L+1}-1}{w_q} < min_{th}$$

Theo các kết quả tính toán của các nghiên cứu về RED, người ta khuyến cáo $w_q \geq 0.001$. Giá trị tối ưu cho w_q là 0.002, ngoài ra, tuỳ theo điều kiện của mạng, ta có thể chọn w_q trong khoảng (0.002, 0.003) [1].

b. Các giá trị ngưỡng min_{th} ***và*** max_{th}

Giá trị tối ưu cho min_{th} và max_{th} phụ thuộc vào một số yếu tố, trong đó có kích thước trung bình mong muốn của hàng đợi, hay cũng chính là mức tắc nghẽn được phép

tại hàng đợi. Theo nguyên lý hoạt động của RED:

- Nếu lưu lượng trên mạng ít có các đột biến, min_{th} và max_{th} nên được thiết lập giá trị cao để tận dụng tối đa đường truyền.

- Nếu lưu lượng trên mạng thường xảy ra đột biến, min_{th} và max_{th} nên được thiết lập giá trị nhỏ để có thể hấp thu các đột biến lưu lượng. Tuy nhiên, các giá trị quá nhỏ sẽ dẫn đến lãng phí dải thông, vì số gói tin bị loại bỏ không cần thiết sẽ cao hơn.

- Khoảng ngưỡng (max_{th} - min_{th}) ảnh hưởng mạnh đến thăng giáng độ trễ, thông lượng và khả năng hấp thu các đột biến tạm thời tại hàng đợi. Để tránh được hiện tượng đồng bộ toàn cầu thì không nên để giá trị khoảng này quá thấp, *avg* sẽ nhanh chóng đạt tới ngưỡng trên. RED gateway sẽ làm việc hiệu quả nhất khi (max_{th} - min_{th}) bằng mức gia tăng điển hình của kích thước hàng đợi trung bình trong một khoảng thời gian khứ hồi RTT. Quy tắc nên tuân theo là thiết lập max_{th} ít nhất gấp đôi min_{th}. Sau khi kiểm nghiệm các mô phỏng trong [1], chúng tôi khuyến nghị rằng $min_{th} \geq 5$ và $max_{th} = 3\ min_{th}$.

c. Xác suất loại bỏ tối đa max_p

Giá trị max_p sẽ quyết định tần số loại bỏ gói là lớn hay nhỏ, nó quyết định *avg* sẽ nằm ở mức nào trong khoảng từ min_{th} đến max_{th}. Vì vậy tùy từng yêu cầu mà có thể thiết lập max_p cho phù hợp. Thí dụ, để *avg* nằm trong khoảng giữa min_{th} và max_{th}, giá trị max_p phù hợp là 0.02. Tuy nhiên nếu tắc nghẽn thường xuyên xảy ra, cần chọn max_p lớn hơn, thí dụ bằng 0.1. Về tham số này, chúng tôi cũng đồng ý với các tác giả trong [1] rằng nên chọn $max_p = 0.1$.

2.3.4. Một số đánh giá về RED

❖ **Ưu điểm:**

RED là một điển hình của các chiến lược quản lý hàng đợi động AQM, do vậy RED có đầy đủ các ưu điểm chung của chiến lược AQM, ngoài ra RED còn có một số ưu điểm khác biệt sau:

- *Tránh tắc nghẽn:* Nếu RED gateway thực sự loại bỏ gói tin đến khi kích thước hàng đợi trung bình đạt đến ngưỡng trên, thì RED gateway đảm bảo kích thước hàng đợi trung bình tính theo lý thuyết không vượt quá ngưỡng trên. Nếu trọng số hàng đợi wq được thiết lập một cách hợp lý thì RED gateway hoàn toàn có thể điều khiển được kích thước hàng đợi trung bình thực sự. Nếu RED gateway đánh dấu một bit trong header của gói tin đến khi kích thước hàng đợi trung bình vượt quá ngưỡng trên, thay vì loại bỏ nó, thì hiệu quả hoạt động của RED gateway còn phụ thuộc vào sự hợp tác của các nguồn để điều khiển kích thước hàng đợi trung bình.

- *Tránh đồng bộ toàn cục*: Tỷ lệ đánh dấu gói tin của RED gateway phụ thuộc vào mức độ tắc nghẽn. Ở giai đoạn tắc nghẽn thấp, RED gateway đánh dấu gói tin với một xác suất thấp, và khi tắc nghẽn tăng lên thì xác suất đánh dấu cũng tăng lên. Mặt khác, RED gateway chọn ngẫu nhiên các gói tin đến để đánh dấu; với phương pháp này xác suất đánh dấu một gói tin từ một kết nối cụ thể tỉ lệ với phần băng thông được chia sẻ của kết nối đó tại gateway. Như vậy, RED gateway tránh hiện tượng đồng bộ toàn cục bằng cách đánh dấu gói tin theo một tỷ lệ thấp nhất có thể và việc đánh dấu các gói tin một

cách ngẫu nhiên.

- *Đơn giản*: Thuật toán RED có thể được cài đặt với một chi phí vừa phải, không yêu cầu phải cài đặt đồng loạt cho tất cả các gateway trong mạng mà có thể triển khai dần.

- *Cực đại hoá công suất toàn cục*: Công suất được định nghĩa bằng tỷ lệ giữa thông lượng và độ trễ. Vì RED gateway điều khiển cho kích thước hàng đợi nhỏ, dẫn tới độ trễ nhỏ, mặt khác như các mô phỏng chúng tôi trình bày dưới đây, hệ số sử dụng đường truyền với RED và DropTail là xấp xỉ nhau, vì vậy công suất đường truyền cao hơn rất nhiều so với DropTail (điều này được minh chứng bằng các mô phỏng hỏng dưới đây).

- *Tính công bằng*: Một trong những mục tiêu quan trọng của một thuật toán quản lý hàng đợi là sự công bằng trong việc cấp phát đường truyền cho các kết nối chia sẻ. Về điểm này thì RED gateway có phần hạn chế. RED gateway không phân biệt các kết nối hay các lớp kết nối khác nhau. Đối với RED gateway, tỷ lệ các gói tin bị đánh dấu tỷ lệ với phần băng thông chia sẻ của kết nối đó tại gateway. Tuy nhiên nó không cố gắng đảm bảo tất cả các kết nối nhận được cùng một tỷ lệ dải thông, mặt khác nó không điều khiển được hiện tượng misbehaving users - hiện tượng một kết nối nào đó nhận được tỷ lệ băng thông lớn hơn rất nhiều so với các kết nối khác đi qua gateway.

❖ **Nhược điểm:**

- Một trong những vấn đề cơ bản của RED là nó dựa vào độ dài hàng đợi để đánh giá sự tắc nghẽn, trong khi sự tắc nghẽn chỉ xảy ra ở hàng đợi cố định và độ dài hàng đợi đem lại rất ít thông tin về tắc nghẽn.

- Việc cài đặt các tham số phù hợp cho RED khi thực thi ở những môi trường mạng khác nhau là rất khó. Để RED có thể hoạt động lý tưởng, cần phải có một số lượng đủ không gian hàng đợi và giá trị các tham số phù hợp.

- Do phép tính xác suất loại bỏ gói của RED được hình thành nên cơ sở mô hình tuyến tính, nên RED không đáp ứng bản chất phi tuyến của mạng. Vì vậy, cần có những thay đổi cho RED vì lưu lượng trên mạng đi theo từng đợt, gây ra những dao động quá nhanh của hàng đợi trong nút mạng.

- Cơ chế RED hoạt động phụ thuộc rất nhiều vào min_{th} và max_{th} trong khi tình trạng mạng luôn biến động bởi lưu lượng gói tin từ các tuyến khác nhau đến nút mạng. Ngoài ra, RED cũng không đảm bảo sự công bằng giữa các luồng, RED loại bỏ hay nhận gói nhưng không quan tâm đến băng thông của các luồng và cũng không hạn chế được luồng không thích nghi gây ảnh hưởng xấu đến luồng thích nghi.

2.4. Chiến lược A-RED

Như mục 2.3 ở trên đã trình bày về thuật toán RED, ta thấy chiến lược quản lý này cho phép mạng đạt được đồng thời thông lượng cao và độ trễ thấp do sử dụng quá trình loại bỏ ngẫu nhiên bằng việc cố gắng duy trì kích thước hàng đợi ở giá trị trung bình. Tuy nhiên RED lại có mặt hạn chế: kém hiệu quả khi có sự thay đổi nhanh chóng mật độ gói tin vào hàng đợi và các tham số điều chỉnh kém thích nghi làm tăng số gói tin bị loại bỏ. Về cơ bản RED yêu cầu phải loại bỏ đủ các gói để đạt được mục đích giữ cho kích thước hàng đợi không vượt quá giới hạn trên max_{th}. Khi kết nối có lưu lượng thấp, hay giá trị

max_p được chọn cao thì kích thước hàng đợi trung bình sẽ tiến đến gần giá trị min_{th}; còn khi đường truyền xảy ra tắc nghẽn nặng hơn, hoặc giá trị max_p được chọn thấp thì kích thước hàng đợi trung bình gần bằng hoặc lớn hơn max_{th}. Kết quả là, độ trễ hàng đợi trung bình rất nhạy cảm với tải (lưu lượng đưa vào mạng) và giá trị tối ưu của các tham số, do đó không thể dự đoán trước được. Độ trễ là một thành phần chính của chất lượng dịch vụ cung cấp cho người dùng, mạng phải có một sự ước lượng tốt để ước lượng chính xác độ trễ trung bình trong các router khi xảy ra tắc nghẽn. Để đạt được độ trễ trung bình có thể đoán trước được, cần điều chỉnh liên tục các tham số RED để đáp ứng sự thay đổi điều kiện của lưu lượng mạng hiện tại. Thêm vào đó độ thông qua trong RED (thông lượng) cũng nhạy cảm với tải và các tham số RED. RED thường không hiệu quả khi kích thước hàng đợi trung bình vượt quá giá trị max_{th}, khi lớn hơn giá trị này thì dẫn tới việc giảm thông lượng còn tăng tỉ lệ loại bỏ gói tin. Giải pháp đưa ra cho vấn đề trên là tìm ra một thuật toán kế thừa được các ưu điểm của thuật toán RED đồng thời hạn chế được các nhược điểm của RED. Người ta đã cải tiến RED bằng thuật toán ARED (Adaptive – RED). Về cơ bản A-RED vẫn dựa trên thuật toán RED nhưng chỉ thay đổi tham số max_p phù hợp trong trường hợp số luồng đến đồng thời lớn để giữ cho kích thước trung bình hàng đợi luôn nằm trong khoảng min_{th} và max_{th}. Ngoài ra thuật toán A-RED tự động thiết lập các tham số khác của RED như min_{th}, max_{th}, w_q, nó có thể tối thiểu hoá khả năng kích thước hàng đợi trung bình vượt quá giá trị max_{th} để phù hợp với điều kiện mạng nhằm cải thiện hiệu suất của mạng.

Bằng mô phỏng, chúng tôi thấy rằng A-RED đạt được độ dài hàng đợi trung bình đích với một miền rộng các kịch bản, mà không mất một ích lợi nào của RED. Điều này không chỉ giúp người quản trị mạng dự đoán trước được độ trễ hàng đợi trung bình, mà còn hạn chế được khả năng kích thước hàng đợi trung bình vượt quá max_{th}. Bởi vậy A-RED sẽ cho kết quả tốt hơn so với RED, làm tăng thông lượng, hạn chế được khả năng mất gói và giảm sự thăng giáng độ trễ hàng đợi.

2.4.1. Hoạt động của thuật toán A-RED

A-RED điều chỉnh thích ứng giá trị max_p để giữ cho kích thước hàng đợi trung bình nằm trong khoảng giá trị min_{th} và max_{th}. Để đạt được mục tiêu này có 4 cách:

- Max_p được hiệu chỉnh không chỉ giữ cho kích thước hàng đợi trung bình nằm giữa hai giá trị min_{th} và max_{th} mà còn giữ cho kích thước hàng đợi trung bình nằm trong một giải cho phép trong khoảng min_{th} và max_{th}.
- Max_p được điều chỉnh thích nghi chậm, sau những khoảng thời gian vượt quá thời gian khứ hồi (round-trip time), và được thực hiện với chi phí thấp.
- Giá trị max_p được duy trì trong miền [0.01 ; 0.5] (tương ứng với [1%, 50%]).
- Thay cho việc tăng theo cấp số nhân và giảm giá trị max_p thuật toán thực hiện chế độ tăng theo cấp số cộng giảm theo cấp số nhân (AIMD). Điều đó có nghĩa là khi tăng thì cộng thêm một lượng đủ nhỏ (α), khi giảm thì nhân với một giá trị nhỏ hơn 1 (β). Các giá trị α, β được chọn sao cho kích thước hàng đợi trung bình quay trở lại miền mục tiêu (khoảng (q_{low}, q_{high})) trong thời gian không quá 25s.

Thuật toán A-RED được trình bày trong hình 2.5

Every interval seconds:

If (avg > target and $max_p \leq 0.5$*)*

Tăng giá trị max_p

$max_p \leftarrow max_p + \alpha$;

Else if (avg <target and max_p ¿ *0.01)*

giảm max_p ;

$max_p \leftarrow max_p * \beta$;

Các biến :

avg: kích thước hàng đợi trung bình

Các tham số cố định :

interval: khoảng thời gian khoảng 0,5 s

target: giá trị mong đợi cho *avg* nằm trong khoảng

$[min_{th} + 0.4*(max_{th} - min_{th})$; $min_{th} + 0.6*(max_{th} - min_{th})]$

α: hệ số tăng; min (0.01; $max_p/4$)

β: hệ số giảm; 0.9

Hình 2.5. Thuật toán A-RED

A-RED có sự kế thừa thuật toán RED gốc, mặt khác A-RED được bổ sung thêm thuật toán hiệu chỉnh max_p. Ưu điểm của A-RED là ở chỗ hiệu chỉnh chậm và không thường xuyên giá trị max_p. Việc hiệu chỉnh max_p chỉ được thực hiện khi cần thiết sau những khoảng thời gian dài. Hầu như chi phí để thực hiện những thay đổi này chỉ là ở những thời điểm ngay sau khi có đột biến của tắc nghẽn (lưu lượng tăng hoặc giảm đột ngột). Để đảm bảo cho A-RED vẫn hoạt động tốt sau những thời điểm đột biến này, max_p luôn được giữ trong khoảng [0.01, 0.5]. Điều này đảm bảo trong suốt thời gian chuyển dịch trạng thái của mạng (qua những thời điểm mạng có đột biến), hiệu suất tổng thể của RED vẫn có thể chấp nhận được (ngay cả khi kích thước hàng đợi trung bình không nằm trong miền mục tiêu) và độ trễ trung bình cũng như thông lượng chịu ảnh hưởng với một mức độ không đáng kể.

2.4.2. Các tham số của A-RED

a) Phạm vi của max_p

Cận trên của giá trị max_p được thiết lập 0.5 và có thể được chỉnh sửa theo hai căn cứ. Đầu tiên cố gắng tối ưu RED để tốc độ loại bỏ gói tin không vượt quá 50% bởi nếu tốc độ này vượt quá 50% là không thể chấp nhận được.

Ngoài ra khi tỉ lệ loại bỏ gói tin thay đổi từ 1 đến max_p khi kích thước hàng đợi trung bình chạy từ min_{th} đến max_{th}, và tỉ lệ loại bỏ gói thay đổi từ $max_p \rightarrow 1$ khi kích thước hàng đợi trung bình thay đổi từ giá trị $max_{th} \rightarrow 2max_{th}$. Do đó với giá trị max_p được thiết lập tới giá trị 0.5 thì xác suất loại bỏ các gói thay đổi từ $0 \rightarrow 1$ khi kích thước hàng đợi thay đổi từ $min_{th} \rightarrow 2max_{th}$. Điều này giúp tăng cường hiệu năng truyền lớn ngay cả

khi tốc độ loại bỏ gói vượt quá 50%. Cận trên của max_p được thiết lập là 0.5 có nghĩa là khi tỉ lệ gói tin vượt quá 25%, kích thước hàng đợi trung bình có thể vượt quá phạm vi cho phép lên tới bốn lần[1].

Cận dưới của max_p được thiết lập 0.01 với mong muốn hạn chế miền của max_p. Bằng mô phỏng chúng tôi thấy rằng đối với những kịch bản với tỉ lệ loại bỏ gói tin nhỏ, RED thực hiện rất tốt với max_p được thiết lập là 0.1.

b) Tham số α, β

Khi xét đến giá trị α, β yêu cầu đặt ra là ngay cả khi hoạt động dưới điều kiện bình thường thì nếu giá trị max_p có thay đổi một lần cũng gần như không ảnh hưởng tới sự thay đổi của kích thước hàng đợi trung bình.

Khi giá trị max_p được điều chỉnh thích ứng với lưu lượng đến hàng đợi, thì xác suất loại bỏ gói ở trạng thái ổn định p cũng được duy trì và kích thước hàng đợi trung bình cũng sẽ dịch chuyển để phù hợp với giá trị max_p mới. Do đó $p < max_p$ khi max_p tăng lên bởi hệ số α, và giá trị hàng đợi trung bình avg có thể giảm từ giá trị $min_{th} + \frac{p}{\max_p}(max_{th} - min_{th})$ xuống $min_{th} + \frac{p}{\max_p + \alpha}(max_{th} - min_{th})$

Nó là sự giảm của giá trị: $\frac{\alpha}{\max_p + \alpha} + \frac{p}{\max_p}(max_{th} - min_{th})$

Giá trị max_p nhỏ hơn $0.2(max_{th} - min_{th})$, do đó kích thước hàng đợi trung bình không phụ thuộc vào giá trị max_p và để tránh hiện tượng kích thước hàng đợi giảm đột ngột từ giá trị biên trên xuống giá trị biên dưới. Vì $\frac{p}{\max_p} < 1$, nên ta chọn α sao cho:

$$\frac{\alpha}{\max_p + \alpha} < 0.2 \text{ hay } \alpha < 0.25\ max_p$$

Tương tự có thể kiểm tra việc giảm max_p theo cấp số nhân để không gây ra hiện tượng kích thước hàng đợi trung bình tăng từ giá trị biên giới tới giá trị biên trên. Phân tích tương tự như α cho thấy $\frac{p(1-\beta)}{\max_p \beta}(max_{th} - min_{th}) < 0.2(max_{th} - min_{th})$

Kích thước hàng đợi trung bình không nên thay đổi từ dưới phạm vi cho phép tới trên phạm vi cho phép trong một khoảng thời gian duy nhất.

Chọn β: $\frac{1-\beta}{\beta} < 0.2$; $\beta > 0.83$. Giá trị mặc định cho β là 0.9 (xem hình 2.5).

c) Thiết lập các tham số max_{th} và w_q

---

[1] Cho $\max_{th} = k\ min_{th}$, kích thước hàng đợi cho phép là $\frac{k+1}{2}\ min_{th}$, và với tỉ lệ loại bỏ gói tin tới gần 100% và $\max_p$ thiết lập 50%, kích thước hàng đợi trung bình tới gần $2\max_{th} = 2k\ min_{th}$.

A-RED loại bỏ sự phụ thuộc của RED vào tham số max_p và một số tham số khác thì ta có thể tự động thiết lập tham số max_{th}, w_q. Giá trị max_{th} được khuyến nghị nên gấp ba lần min_{th}. Trong trường hợp này thì kích thước hàng đợi trung bình tập trung xung quanh giá trị $2min_{th}$ do đó nó chỉ chịu ảnh hưởng của tham số min_{th} của RED.

Mặt khác, người ta đã chứng minh được rằng, w_q phụ thuộc vào tốc độ đường truyền, đường truyền tốc độ cao yêu cầu giá trị w_q nhỏ hơn, vì vậy trong mốt số bài báo các tác giả đã thiết lập w_q là một hàm của băng thông đường truyền:

$$w_q = 1- \exp(-1/C)$$

trong đó C là băng thông đường truyền, tính theo packet/s.

Điều này có ý nghĩa đặc biệt giúp người quản trị mạng có thể chọn chế độ tự động thiết lập các tham số A-RED. Theo đó, ta chỉ cần thiết lập độ trễ đích cho A-RED gateway, và đặt các giá trị khởi tạo cho maxp, tất cả các công việc còn lại đều do A-RED gateway đảm nhiệm.

2.4.3. Một số đánh giá về A-RED

❖ **Ưu điểm**

- Tính ổn định của A-RED không phụ thuộc vào tải mạng.
- A-RED có thể dự đoán trước độ trễ hàng đợi trung bình.
- A-RED tự động việc thiết lập các thông số của nó để đáp ứng thay đổi tải nạp.

❖ **Nhược điểm**

A-RED không làm rõ được rằng đó là chính sách tốt nhất và tối ưu của các thay đổi tham số.

2.4.4. So sánh thuật toán RED và A-RED

Thuật toán A-RED là thuật toán nâng cao của thuật toán RED do đó A-RED khắc phục được mặt hạn chế của RED:

- RED quản lý hàng đợi dựa trên kích thước trung bình của hàng đợi nên kích thước trung bình hàng đợi thay đổi theo các mức tắc nghẽn và quá trình thiết lập các tham số. Điều này được thể hiện bằng việc khi tắc nghẽn xảy ra nhẹ hay max_p cao thì kích thước hàng đợi gần tới giá trị min_{th}. Khi tắc nghẽn trong mạng nặng hay kích thước hàng đợi trung bình bằng hoặc lớn hơn max_{th}. Kết quả trễ hàng đợi trong thuật toán RED phụ thuộc vào tải lưu lượng và các tham số, do đó mà trễ hàng đợi không thể đoán trước.
- Một nhược điểm nữa của RED là khả năng thông qua trong thuật toán này cũng phụ thuộc nhiều vào tải lưu lượng và các tham số.

Do thuật toán A-RED quản lý kích thước trung bình của hàng đợi dựa trên việc tương thích giá trị max_p sao cho kích thước trung bình hàng đợi thay đổi trong khoảng min_{th} và max_{th} nên khắc phục được sự phụ thuộc của trễ hàng đợi và thông lượng của hàng đợi vào các tham số và tải lưu lượng.

2.5. Thuật toán A-RIO

2.5.1. Giới thiệu

Chúng ta biết rằng RED (hay A-RED) gateway đối xử với các gói tin đến một

cách bình đẳng, không có sự phân biệt theo các mức ưu tiên của chúng. Trên thực tế, người dùng hoàn toàn có quyền yêu cầu các mức chất lượng dịch vụ khác nhau tuỳ theo mức giá cả thoả thuận với nhà cung cấp và những nhà cung cấp dịch vụ phải có trách nhiệm đáp ứng được điều đó. Tuy nhiên nếu chỉ đơn thuần áp dụng RED (hoặc A-RED) sẽ dẫn tới không công bằng đối với các luồng lưu lượng: luồng phải trả nhiều tiền hơn cũng chỉ được cung cấp cùng một dịch vụ như các luồng trả ít hơn. Từ nhu cầu đó, David D. Clark và Wenjia Fang đã đề xuất thuật toán RIO [30] cải tiến RED bằng cách phân loại các gói tin đến theo hai mức ưu tiên. Việc này được thực hiện bằng cách gắn thẻ "In" hoặc "Out" cho mỗi gói tin đến dựa trên hồ sơ dịch vụ (service profile) đã được thoả thuận giữa khách hàng và nhà cung cấp dịch vụ. Theo đó, gói tin được gắn thẻ "In" (in-profile) nếu nó là gói tin nằm trong hồ sơ dịch vụ đã được thoả thuận; ngược lại gói tin được gắn thẻ "Out" (out-of-profile) khi nó nằm ngoài hồ sơ dịch vụ, cũng có thể coi đó như những gói tin không hợp lệ được đưa vào mạng. Khi tắc nghẽn xảy ra thì mạng sẽ "ưu tiên" loại bỏ gói tin "Out" nhanh hơn. Và không giống như đối với các chính sách phục vụ WFQ, tại các router trong mạng không có sự phân tách lưu lượng thành các luồng hay các hàng đợi khác nhau; mà các gói tin của tất cả người dùng được gộp chung vào trong một hàng đợi duy nhất, điều này theo [30], là phù hợp đối với mạng ngày nay.

2.5.2. Quản lý hàng đợi động trong kiến trúc DiffServ

Mục tiêu của AQM trong các mạng DiffServ có sự khác biệt về bản chất so với trong các mạng Best-effort. Trong khi mục tiêu của AQM trong các mạng Best-effort là để tránh tắc nghẽn thì trong các mạng DiffServ là loại bỏ có ưu tiên.

RIO viết tắt của RED with In/Out là một kỹ thuật AQM cơ bản phù hợp cho việc thiết lập xử lý từng chặng theo chuẩn AF. Xin được nhắc lại một chút về RIO: RIO là sự mở rộng của RED bằng cách sử dụng hai tập tham số để phân biệt các gói tin *In* (in-profile) và *Out* (out-of-profile). Để quyết định loại bỏ các gói tin *Out*, RIO sử dụng kích thước trung bình của hàng đợi tổng, cấu thành từ cả các gói *In* và *Out*. Đối với các gói *In*, nó sử dụng kích thước trung bình của hàng đợi ảo, được tạo bởi chỉ các gói *In*. RIO đã được mở rộng để xử lý với $n > 2$ mức ưu tiên theo một nguyên lý tương tự. Khi đó xác suất loại bỏ các gói tin có mức ưu tiên j ($1 \leq j < n$) phụ thuộc vào kích thước trung bình của hàng đợi ảo mức j (là hàng đợi tạo bởi chỉ các gói tin có mức ưu tiên từ 1 đến j). Đối với các gói tin có mức ưu tiên n (ưu tiên thấp nhất), thì xác suất loại bỏ là một hàm của kích thước hàng đợi "vật lý" (hàng đợi tổng cộng-total queue). Phương pháp gốc này có tên là RIO-C (RIO-Coupled) dùng để phân biệt với các phương pháp khác được đề xuất sau đó. Chẳng hạn, Weighted RED (WRED) sử dụng kích thước hàng đợi trung bình tổng cộng cho mọi mức ưu tiên, trong khi RIO-D (RIO-Decoupled) tính xác suất loại bỏ cho các gói tin mức j như một hàm theo số các gói tin trung bình có cùng mức ưu tiên.

RIO-C phân biệt các gói tin theo các mức ưu tiên bằng ba cách. Cách thứ nhất là dùng các ngưỡng khác nhau cho các mức ưu tiên khác nhau, sao cho việc loại bỏ bắt đầu sớm đối với các gói tin có mức ưu tiên cao hơn. Cách thứ hai là dùng xác suất loại bỏ tăng lên một cách tuyến tính theo các mức ưu tiên. Cách thứ ba dựa trên tính toán kết hợp

xác suất loại bỏ; trên thực tế, việc tính xác suất loại bỏ đối với gói tin có mức ưu tiên *j* sử dụng số gói tin trung bình của tất cả các gói tin có mức ưu tiên nhỏ hơn *j* mang lại một cách phân biệt tốt. Hai cách đầu phụ thuộc đơn thuần vào việc chọn các tham số và chúng không loại trừ lẫn nhau.

Như đã đề cập trước, không có một quy tắc chính xác nào cho việc thiết lập các tham số RED (hai ngưỡng min_{th}, max_{th}, xác suất loại bỏ tối đa max_p và trọng số hàng đợi w_q); thêm vào đó, các kết quả nghiên cứu cũng chỉ ra những khó khăn để tìm được một cấu hình RED thật sự hiệu quả. Vấn đề càng trở nên nghiêm trọng hơn đối với RIO: chẳng hạn, xét RIO với n mức ưu tiên, về nguyên tắc thì cần phải thiết lập $3n + 1$ tham số ($2n$ ngưỡng và n xác suất loại bỏ, cộng thêm một trọng số w_q – giả sử w_q được dùng cho mọi hàng đợi ảo). Rõ ràng là vấn đề thiết lập các tham số trở nên phức tạp hơn và trở thành một chủ đề cần được nghiên cứu. Các nghiên cứu [6,31] và [32] chỉ ra sự khó khăn trong việc hiệu chỉnh RIO để đạt được một hiệu năng có thể dự đoán trước.

2.5.3. Thuật toán quản lý hàng đợi A-RIO

A-RIO là một mở rộng trực tiếp cả hai thuật toán A-RED và RIO-C. A-RIO theo cách tiếp cận của A-RED, thực hiện một hiệu chỉnh tự động on-line đối với xxx nhằm đạt được một hiệu năng có thể dự đoán trước. Có nhiều cách tiếp cận đã được đưa ra nhằm hiệu chỉnh RED, ở đây A-RED được chọn vì tính đơn giản và hiệu quả của nó (cả về tư tưởng và cài đặt).

Cũng như A-RED ở chế độ tự động, A-RIO chỉ cần một tham số đầu vào là độ trễ đích, nó sẽ tự động ánh xạ sang tập các tham số router. Đặc trưng này rất có ý nghĩa đối với nhà cung cấp dịch vụ phân loại: cấu hình router theo độ trễ - một độ đo QoS liên quan trực tiếp đến đặc tả dịch vụ và đặc tả yêu cầu của người dùng- chắc chắn dễ hiểu hơn nhiều so với các tham số trừu tượng như ngưỡng hàng đợi, xác suất loại bỏ, hoặc trọng số trung bình...Về hiệu năng, A-RIO cố gắng đạt được thông lượng cao trong khi giữ cho độ trễ trong một khoảng có thể dự đoán được ngay cả khi tải nặng. Thuật toán A-RIO dựa trên hai nguyên lý chính. Nguyên lý thứ nhất là sử dụng một thể hiện đầy đủ của A-RED cho mỗi mức ưu tiên trong lớp AF (hàng đợi vật lý). Nguyên lý thứ hai là sử dụng các ngưỡng chồng nhau hoàn toàn cho tất cả các mức ưu tiên. Mã giả của A-RIO được trình bày trong hình 2.6, trong khi các khái niệm cho một hàng đợi với ba mức ưu tiên được thể hiện trong hình 2.7. Dưới đây là những điểm chính của A-RED được giữ không đổi cho A-RIO:

- Tham số xác suất loại bỏ tối đa $max_p^{(i)}$ dao động trong khoảng 0.01 và 0.5.

- Ngưỡng dưới min_{th} được tính theo một hàm của độ trễ đích d_t và dung lượng đường truyền C (packets/s) với cận dưới là 5 gói tin. Do đó, $min_{th} = \max(5, d_t.C/2)$. Ngưỡng trên max_{th} được tính cố định bằng $3.min_{th}$.

- w_q cũng được tính theo thông lượng đường truyền: $w_q = 1 – \exp(-1/C)$.

- Khoảng *gentle* của RED được sử dụng từ đầu đến cuối. Khoảng này, như trên hình 2.8, là $max_{th} \le avg^{(i)} \le 2.max_{th}$.

- Hàm hiệu chỉnh $max_p^{(i)}$ sử dụng luật AIMD (*Additive Increase Multiplicative Decrease* - tăng theo cấp số cộng giảm theo cấp số nhân). Luật này nhằm tránh sự thay đổi đột ngột của $max_p^{(i)}$ dẫn tới sự dao động mạnh của kích thước hàng đợi.

- Nếu tải thay đổi một cách đột ngột, kích thước hàng đợi trung bình có thể nằm ngoài miền mục tiêu. Các tham số α, β được chọn cố định sao cho kích thước hàng đợi trung bình quay lại miền mục tiêu trong thời gian không quá 25 giây.

Mục tiêu khi thiết kế A-RIO là đảm bảo được độ trễ luôn nằm trong một miền mong muốn cho trước, miền này được gọi là độ trễ đích (target delay). Mà nên áp dụng cho tất cả các lưu lượng có tải không đáng kể. Mục tiêu của thuật toán là giữ cho kích thước hàng đợi trung bình trong khoảng (q_{low}, q_{high}), ở đây $q_{low} = min_{th} + 0.4(max_{th} - min_{th})$ và $q_{high} = min_{th} + 0.6(max_{th} - min_{th})$.

Để giải thích rõ hơn điều này, chúng ta hãy xem xét trong kịch bản RIO-C với các ngưỡng không chồng nhau. Các mức ưu tiên là 1 (*In*) và 2 (*Out*). Nếu tỉ lệ các gói *In* là thấp so với dung lượng đường truyền, khi tải nặng thì các ngưỡng và xác xuất loại bỏ ứng với các gói *Out* sẽ được kích hoạt, giữ kích thước hàng đợi nằm trong khoảng min_2 và max_2. Tuy nhiên, nếu phần lớn lưu lượng là các gói *In*, kích thước trung bình sẽ nằm trong khoảng min_1 và max_1. Các ngưỡng so le này làm cho kích thước hàng đợi trung bình luôn thay đổi khi thay đổi lưu lượng trộn lẫn. Điều này dẫn tới không đảm bảo được độ trễ đích trong mọi kịch bản. Bởi vậy, A-RIO sử dụng ngưỡng chung cho tất cả các mức ưu tiên. Theo đó, cơ chế hiệu chỉnh A-RED kéo cho kích thước hàng đợi trung bình về một khoảng giới hạn, dẫn tới đảm bảo được độ trễ đích, bất kể lưu lượng được trộn lẫn như thế nào. Thuật toán A-RIO được thể hiện như ở hình 2.6.

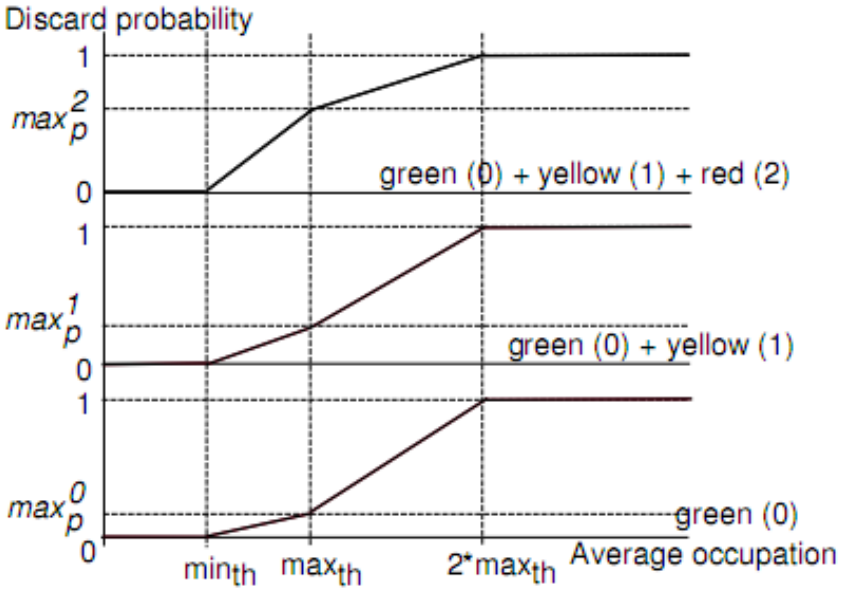
```
for mỗi gói tin đến với độ ưu tiên i,
    for mỗi mức ưu tiên  j = i, i + 1, .., n
        cập nhật avg(j): avg(j) ← avg(j) * (1 - wq) + q(j) *wq
         với mỗi đơn vị thời gian interval cập nhật maxp(j):
                if avg(j) > qhigh and maxp(j) < 0.5
                    tính hệ số tăng: α ← min(0.01, maxp(j) /4)
                    tăng maxp(j) : maxp(j) ← maxp(j) + α
                    if j < n then: maxp(j) ← min(maxp(j), maxp(j+1))
                 else  if avg(j) < qlow và maxp(j) > 0.01
                    giảm maxp(j) : maxp(j) ← maxp(j) * β
                    if j > 0 then: maxp(j) ← max(maxp(j), maxp(j-1))
     if minth < avg(i) ≤ maxth
        tính p(i) như trong A-RED
        loại gói tin này với xác suất p(i)
     else if maxth < avg(i) ≤ 2*maxth
        tính p(i)gentle như trong A-RED
        loại gói tin này với xác suất p(i)gentle
     else if avg(i) > 2 * maxth
        loại gói tin này
```

Hình 2.6. Thuật toán A-RIO

Các biến và các tham số cố định:

$avg^{(i)}$: kích thước trung bình ứng với mức ưu tiên *i* (là tổng số gói tin của các mức ưu tiên từ 1 đến *i*)

$max_p^{(i)}$: xác suất loại bỏ tối đa ứng với mức ưu tiên *i* (khi $avg^{(i)} = maxp^{(i)}$)
$p^{(i)}$: xác suất loại bỏ ứng với mức ưu tiên *i*
$p^{(i)}_{gentle}$: xác suất loại bỏ ứng với mức ưu tiên *i* trong vùng *gentle*
interval: 0.5 s; β (hệ số giảm): 0.9



Hình 2.7. A-RIO với ba mức ưu tiên

Tuy nhiên, việc sử dụng các ngưỡng chồng nhau có ảnh hưởng đến việc phân biệt mức độ ưu tiên các gói tin. Như đã đề cập trong phần 2.5.2, RIO-C phân biệt các mức ưu tiên theo ba cách cơ bản: dùng các ngưỡng khác nhau, dùng hàm loại bỏ khác nhau, và dùng hàng đợi ảo kết hợp. Trong A-RIO, với việc dùng các ngưỡng chồng nhau, chúng ta loại trừ cách thứ nhất. Mặt khác, A-RIO dựa vào hiệu chỉnh max_p và việc sử dụng các $max_p^{(i)}$ khác nhau cho các mức ưu tiên khác nhau đưa ra một phương pháp phân loại khác. Lưu ý rằng các thuật toán thích ứng hiệu chỉnh sao cho: $max_p^{(i)} \leq max_p^{(i+1)}, \forall i \in \{1,..., n-1\}$; cùng với các hàng đợi ảo kết hợp, hạn chế này nên cung cấp một sự đảm bảo cho việc phân loại.

Về vấn đề cài đặt, độ phức tạp của A-RIO tương đương với độ phức tạp của A-RED và RIO-C kết hợp. Hơn nữa, A-RIO không lưu hoặc tính toán thông tin từng luồng, vì vậy khả năng mở rộng (về lý thuyết) là không thành vấn đề.

CHƯƠNG 3. CHIẾN LƯỢC BLUE VÀ ĐỀ XUẤT CẢI TIẾN GIẢI THUẬT QUẢN LÝ HÀNG ĐỢI BLUE

Ngày nay, khi môi trường mạng có nhiều luồng đến đồng thời khá phổ biến, làm cho mật độ các gói tin vào hàng đợi có sự thay đổi nhanh chóng. Như đã trình bày ở trên về chiến lược RED và A-RED, ta thấy các thuật toán được thiết kế với mục tiêu giảm thiểu mất gói tin và độ trễ của hàng đợi, tránh hiện tượng đồng bộ hóa toàn cầu về nguồn, duy trì mức độ sử dụng đường truyền cao. Nhưng các chiến lược này lại không hiệu quả trong việc ngăn ngừa tỉ lệ mất gói tin cao. Một thuật toán quản lý hàng đợi tích cực khác được đề xuất đó là thuật toán BLUE. Bằng cách sử dụng cả mô phỏng và thử nghiệm, ta thấy rằng BLUE khắc phục được nhiều thiếu sót của RED, nó cải thiện hiệu suất của RED trong tất cả các khía cạnh, ngay cả khi được sử dụng với kích thước không gian bộ đệm tối thiểu. Điều này làm giảm độ trễ end-to-end qua mạng. BLUE là cơ chế quản lý hàng đợi tích cực dựa trên tải nạp để dự đoán khả năng sử dụng đường truyền, xác định tắc nghẽn và đưa ra cách xử lý hiệu quả hơn phương pháp dựa vào kích thước hàng đợi trung bình. Nó là một giải thuật cho phép quản lý kiểm soát tắc nghẽn dựa trên sự kiện mất gói dữ liệu và mức độ sử dụng đường truyền thay vì chiếm dụng hàng đợi. Mục đích của các cơ chế này là điều tiết gói tin vào nút mạng để ổn định lưu lượng gói tin đến, nhằm duy trì độ ổn định cho mạng.

3.1. Giải thuật BLUE

Năm 2002, Wu-chang Feng và cộng sự đề xuất cơ chế BLUE [13,24]. Ý tưởng chính đằng sau thuật toán quản lý hàng đợi BLUE là dựa trực tiếp trên sự mất gói tin và việc sử dụng các liên kết hơn là trên các độ dài trung bình hàng đợi tức thời. Điều này tương phản sắc nét với tất cả các kỹ thuật quản lý hàng đợi đã được sử dụng trong điều khiển tắc nghẽn trước đó. Thuật toán quản lý hàng đợi Blue sử dụng độ mất gói và độ khả dụng liên kết để quản lý tắc nghẽn bằng cách phát hiện và điều chỉnh tốc độ của các gói bị loại bỏ hoặc bị đánh dấu.

BLUE sử dụng một biến tham số xác suất p_m để đánh dấu các gói tin khi chúng vào hàng đợi. Xác suất này tăng/giảm một cách tuyến tính tùy thuộc vào tỉ lệ rơi (loại bỏ) gói tin hay mức độ sử dụng đường truyền. Nếu như hàng đợi liên tục hủy bỏ các gói tin vì nguyên nhân tải nạp lớn làm tràn bộ nhớ đệm thì BLUE sẽ tăng xác suất đánh dấu p_m, do đó tăng tốc độ gửi lại thông báo tắc nghẽn hoặc loại bỏ các gói tin. Ngược lại, nếu như hàng đợi trở nên trống rỗng hoặc đường truyền rỗi, BLUE lại giảm xác suất đánh dấu (hay loại bỏ) gói tin của nó. Điều này cho phép BLUE tự điều chỉnh tốc độ cần thiết để gửi thông báo tắc nghẽn trở lại nơi gửi hoặc cho rơi gói tin. Thuật toán đánh dấu (loại bỏ) các gói tin với xác suất p_m được trình bày như hình 3.1.

Dựa trên sự kiện mất gói tin (hay $Q_{len} > L$:

if ((now – last_update) > freeze_time) then

$p_m = p_m + \delta_1$

Last_update = now;

Dựa trên sự kiện đường truyền rỗi hay kích thước hàng đợi rỗng:

if ((now – last_update) > freeze_time) then

$p_m = p_m - \delta_2$

Last_update = now;

Hình 3.1. Giải thuật BLUE

Trong đó:

p_m : xác suất đánh dấu hoặc loại bỏ gói tin

freeze_time : là một tham số xác định khoảng thời gian tối thiểu giữa hai lần cập nhật liên tiếp của p_m

δ_1 : xác định lượng tăng lên của p_m khi hàng đợi tràn

δ_2 : xác định lượng giảm của p_m khi liên kết là nhàn rỗi

now : thời gian hiện hành

last_update : thời gian cuối cùng p_m thay đổi

Qlen: là độ dài hàng đợi hiện tại

L: xác định ngưỡng cho phép gói tin đến tại hàng đợi.

Lượng tăng của P_m thể hiện bởi δ_1 và lượng giảm của P_m thể hiện bởi δ_2. Đại lượng này luôn được cập nhật khi có sự thay đổi của P_m, khi kích thước hàng đợi vượt quá giá trị ngưỡng hiện tại, tại tốc độ 1/*freeze_time*. Tham số *freeze_time* thể hiện khoảng thời gian giữa các lần cập nhật thành công p_m, nó cho phép thay đổi xác suất đánh dấu trước khi giá trị được cập nhật lại. Giá trị này nên được ngẫu nhiên hoá để tránh đồng bộ trên toàn thể các luồng.

Từ thuật toán trên ta thấy xác suất đánh dấu gói tin được cập nhật khi kích thước hàng đợi vượt quá giá trị chính xác nào đó. Việc chỉnh sửa này cho phép giải phóng không gian hàng đợi khi các gói chiếm dụng quá lâu trong hàng đợi, đồng thời cho phép hàng đợi điều khiển trễ hàng đợi khi kích thước hàng đợi được sử dụng quá lớn.

Hoạt động của thuật toán trên có thể được mô tả theo 4 bước sau:

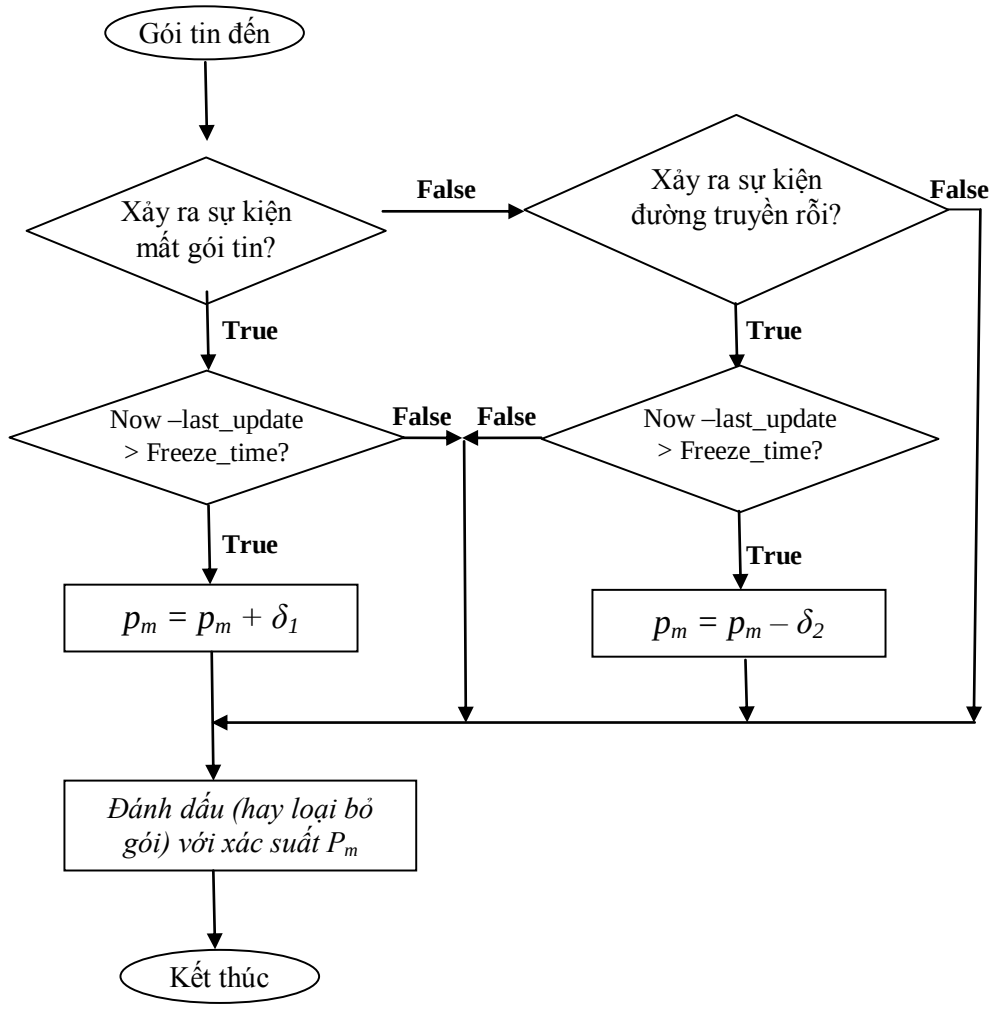
Bước 1: Kiểm tra nếu xảy ra sự kiện mất gói tin thì qua bước 2, nếu không, kiểm tra xem nếu sự kiện đường truyền rỗi thì qua bước 3, ngược lại qua bước 4.

Bước 2: Kiểm tra, nếu khoảng thời gian từ lần cập nhật cuối cùng đến thời điểm hiện tại mà lớn hơn ngưỡng cho phép thì tăng xác suất đánh dấu (hoặc loại bỏ) gói p_m lên. Qua bước 4.

Bước 3: Kiểm tra, nếu khoảng thời gian từ lần cập nhật cuối cùng đến thời điểm hiện tại mà lớn hơn ngưỡng cho phép thì xác suất đánh dấu (hoặc hủy bỏ) gói p_m xuống. Qua bước 4.

Bước 4: Đánh dấu (hoặc loại bỏ) gói tin đến với xác suất pm.

Giải thuật quản lý hàng đợi BLUE cũng có thể trình bày dạng lưu đồ như trong hình *3.2*.



Hình 3.2 Lưu đồ giải thuật BLUE

BLUE có cơ chế quản lý hàng đợi theo tải nạp hiệu quả hơn phương pháp dựa vào chiều dài hàng đợi trung bình. Điều này tương phản một cách rõ ràng với tất cả các thuật toán điều khiển hàng đợi tích cực đã biết, bởi các thuật toán này sử dụng không gian của hàng đợi như là một tiêu chuẩn trong việc điều khiển tránh tắc nghẽn. Nhưng ở chiến lược BLUE thì chúng ta thấy rằng cơ chế này đánh dấu gói tin đến mà không quan tâm đến luồng truyền gói tin đó. Điều này dẫn đến việc đánh rơi gói tin một cách ngẫu nhiên trong tất cả các luồng hoạt động, không đảm bảo sự công bằng giữa các luồng. Đồng thời, việc thiết lập giá trị cho các tham số *freeze_time*, δ_1 và δ_2 hợp lý cho từng môi trường mạng gặp rất nhiều khó khăn. Chính bởi điều này BLUE còn tồn tại một số vấn đề cần phải được cải tiến sau:

- Tham số *freeze_time* cần phải được thiết lập một cách tự động dựa trên thời gian khứ hồi hiệu quả nhằm cho phép bất kỳ sự thay đổi nào trong việc gán xác suất sẽ được phản ánh lại đến nơi gửi trước khi sự thay đổi tiếp theo xảy ra. Đối với đường truyền dài với độ trễ lớn như các đường truyền vệ tinh, *freeze_time* cần phải được tăng lên để phù hợp với thời gian khứ hồi lớn hơn.

- Tham số δ_1 và δ_2 phải được thiết lập phù hợp với tình trạng mạng, cho phép đường truyền có khả năng thích nghi hiệu quả với những thay đổi vĩ mô trong lưu lượng truyền đi qua đường kết nối. Đối với các đường truyền mà tại đó trung bình trong vài phút xảy ra sự thay đổi lưu lượng truyền thì δ_1 và δ_2 phải được thiết lập kết hợp với *freeze_time* để cho phép p_m thay đổi giá trị trung bình trong vài phút. Ngược lại, trong những môi trường mạng thay đổi lưu lượng gói tin đến nút mạng theo giây thì phải cập nhật các tham số *freeze_time*, δ_1 và δ_2 để p_m thích nghi từng giây.

3.2. Đánh giá về thuật toán BLUE:

❖ **Ưu điểm:**

- BLUE làm mất ít gói dữ liệu hơn
- BLUE sử dụng không gian bộ đệm nhỏ
- BLUE duy trì chiều dài hàng đợi ổn định hơn
- Loại bỏ những tác nhân chống lại các nguồn bùng phát.

❖ **Nhược điểm:**

- Không phát hiện tắc nghẽn sớm (các gói tin bị loại bỏ được cập nhật trong hàng đợi trên các luồng hoặc các sự kiện liên kết nhàn rỗi)
- Phản ứng chậm và phụ thuộc vào lịch sử
- Khi tất cả các gói tin được đánh dấu, nhưng các nguồn vẫn đang bị quá tải tại các liên kết nút cổ chai.

3.3. So sánh thuật toán RED và thuật toán Blue

Một vấn đề quan trọng trong quản lý hàng đợi bằng thuật toán Blue là điều khiển tắc nghẽn có thể được thực hiện bởi kích thước hàng đợi nhỏ nhất. Trong khi đó thuật toán RED thì lại yêu cầu kích thước hàng đợi lớn hơn cho cùng một mục đích. Do có kích thước bộ đệm nhỏ hơn có trễ đầu cuối qua mạng nhỏ hơn so với thuật toán RED, do đó nó cải thiện được nhược điểm của thuật toán điều khiển tắc nghẽn. Thêm vào đó các yêu cầu kích thước bộ đệm nhỏ hơn cho phép có nhiều bộ nhớ hơn để phân phối cho các gói có độ ưu tiên cao và giải phóng bộ nhớ trong router để cho các chức năng khác như lưu trữ các bảng định tuyến lớn. Blue cho phép các router thế hệ sau hoạt động tốt thậm chí cả trong trường hợp tài nguyên bộ nhớ bị giới hạn. Tuy nhiên thuật toán Blue ít nhạy cảm với các lựa chọn tham số hơn là đối với giải thuật RED.

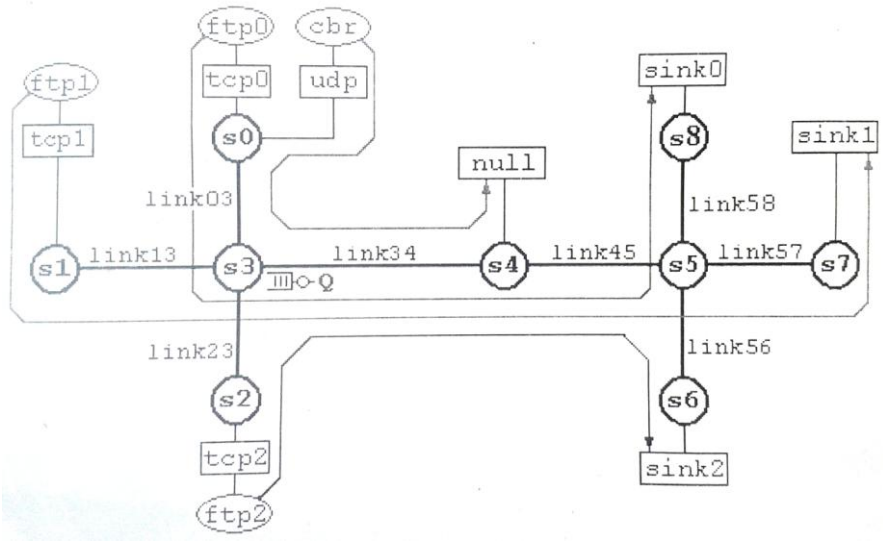
CHƯƠNG 4. ĐÁNH GIÁ HIỆU SUẤT CÁC CHIẾN LƯỢC QUẢN LÝ HÀNG ĐỢI RED, ARED VÀ BLUE BẰNG BỘ MÔ PHỎNG

Dựa trên các kết quả nghiên cứu của cơ chế quản lý hàng đợi tích cực dựa theo kích thước hàng đợi, tôi tiến hành cài đặt các mô hình trên phần mềm mô phỏng NS-2, nhằm kiểm nghiệm các đánh giá về mặt lý thuyết cũng như mô phỏng của RED, A-RED.

4.1 Đánh giá hiệu suất của chiến lược quản lý hàng đợi Red

Trước tiên chúng tôi tiến hành thực hiện lại các mô phỏng trong bài báo [1] (không trình bày ở đây), mục đích là kiểm nghiệm lại các đánh giá của các tác giả, và để làm cơ sở cho việc chọn các tham số RED phục vụ cho các mô phỏng sau này của chúng tôi. Sau khi thực hiện các mô phỏng này, chúng tôi nhận thấy các kết quả mà các tác giả đưa ra là hoàn toàn chính xác và có thể tin cậy. Sau đây là phần trình bày mô phỏng của chúng tôi nhằm so sánh, đánh giá hiệu suất các chiến lược RED và DropTail.

4.1.1 Cấu hình mạng mô phỏng



Hình 4.1. Topo mạng mô phỏng

Cho một mạng mô phỏng có cấu hình, các thực thể gửi/nhận và các nguồn sinh lưu lượng ... như hình vẽ trên. Mạng mô phỏng gồm 9 nút được đánh số từ s0 đến s8. Đường truyền giữa các node đều là full-duplex, không có lỗi:

Link03, link13, link23: 10 Mbps, 1ms

Link34, link45: 1.5 Mbps, 10 ms

Link 56, link57, link58: 10 Mbps, 1ms

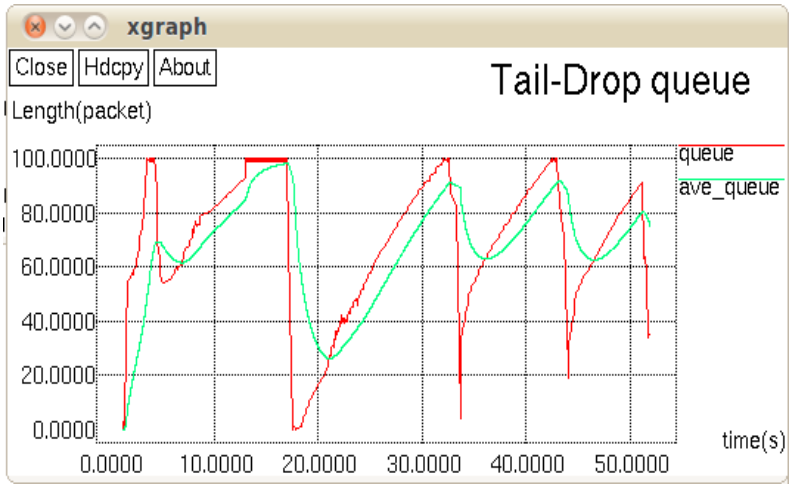
Các thực thể gửi gắn với ba luồng tcp là các nguồn FTP (ftp0, ftp1, fpt2), còn nguồn phát cho udp là nguồn CBR (nguồn sinh lưu lượng với tốc độ không đổi, ở đây ta chọn tốc độ phát cho cbr là 1.8Mbps). Các luồng tcp gửi các gói tin với kích thước 1000 bytes, kích thước cửa sổ tối đa là 64. Các thực thể gửi đưa lưu lượng vào mạng trong các khoảng thời gian như sau: ftp0:1.1s-51.1s; ftp1:1.5s-51.5s; ftp2:1.9s-51.9s; cbr: 13.0s-17s. Hàng đợi Q được đặt giữa nút s3 và nút s4 có kích thước bằng 100 gói tin. Tổng thời gian mô phỏng là 52s. Chúng ta sẽ thay đổi chính sách quản lý tại hàng đợi Q lần lượt là DropTail và RED và so sánh các kết quả.

Với mỗi mô phỏng chúng tôi đưa ra 3 đồ thị: kích thước hàng đợi trung bình, thông lượng sử dụng của mỗi kết nối và kích thước cửa sổ để đánh giá các đại lượng liên quan như độ trễ trung bình, thông lượng của từng kết nối, và nghiên cứu hiện tượng đồng bộ toàn cục. Sau đây là chi tiết về kết quả thu được từ các mô phỏng.

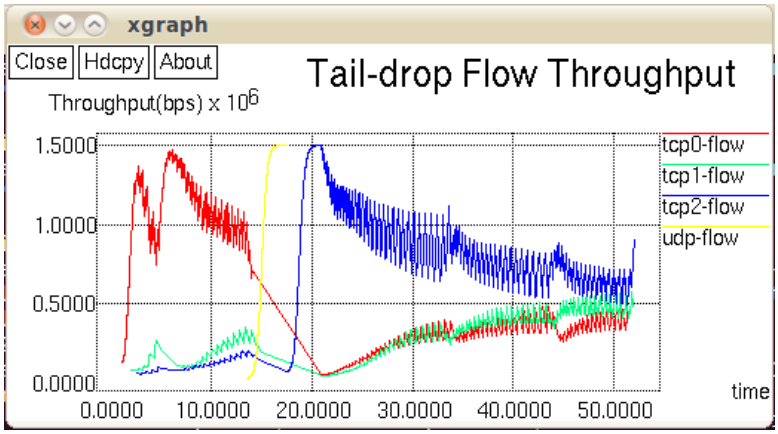
4.1.2 Mô phỏng với chính sách quản lý hàng đợi DropTail:

Sử dụng các tham số cấu hình mạng mô phỏng như mục trên, với việc thiết lập tất cả các hàng đợi đều là DropTail, chúng tôi nhận được các kết quả như ở hình 4.2.

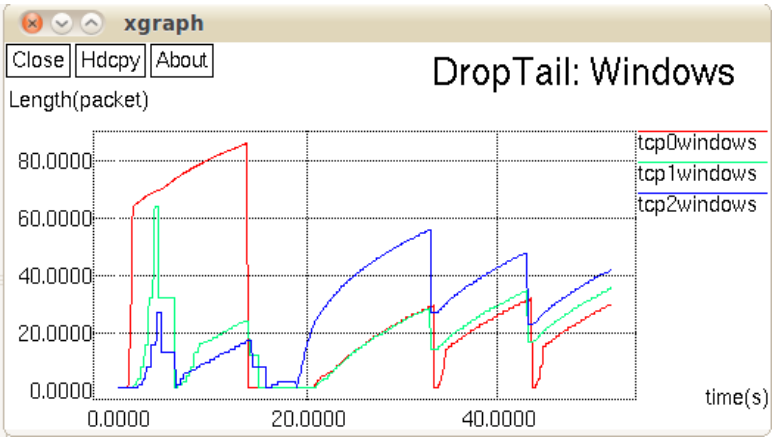
a) Kích thước hàng đợi trung bình của DropTail



b) Thông lượng các luồng của DropTail



c) Kích thước cửa sổ của các luồng của DropTail và RED



Hình 4.2. Các kết quả mô phỏng 1 với hàng đợi DropTail

Nhận xét:

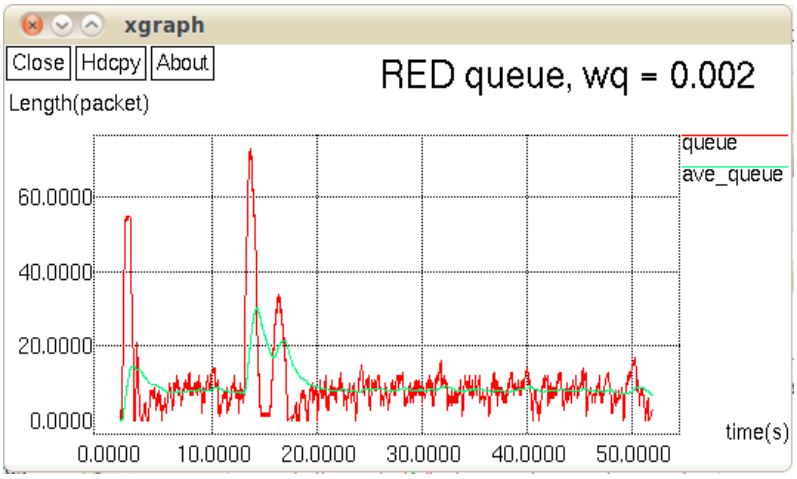
- Trong các khoảng có đột biến lưu lượng (13.0s-17.0s), nguồn cbr có tốc độ cao hơn dung lượng đường truyền ra tại nút 3, gây ra hiện tượng lock-out tại hàng đợi Q, dẫn tới hiện tượng global synchronization, thông lượng của các kết nối tcp đồng loạt giảm kích thước cửa sổ phát (hình 4.2c), dẫn tới thông lượng của các kết nối tcp giảm xuống rất nhỏ (hình 4.2b), cực tiểu khoảng 0.12Mbps. Do cơ chế rút lui theo hàm mũ (exponential backoff) của TCP, trạng thái này còn kéo dài sau khi nguồn cbr ngừng hoạt động. Mặt khác trong thời gian này kích thước hàng đợi hầu như đầy (hình 4.2a), dẫn tới độ trễ hàng đợi cao.

- Ngay cả khi nguồn cbr không hoạt động (từ 20s đến 51.9s), thông lượng của các kết nối tcp0, tcp1 và tcp2 cũng thăng giáng trong một miền rất rộng. Và cũng trong khoảng thời gian này hiện tượng đồng bộ toàn cục vẫn xuất hiện (ở các thời điểm 22.5s, 42.5s), các kết nối cùng tăng kích thước cửa sổ cho đến khi đạt đến ngưỡng thì đồng thời giảm xuống; kích thước hàng đợi vì thế cũng dao động trong một miền rất rộng (khoảng 20% - 30%) so với giá trị trung bình. Ngoài ra chiều dài hàng đợi trung bình (ave_queue) thường xuyên ở mức cao (cỡ 75 ± 15 packet).

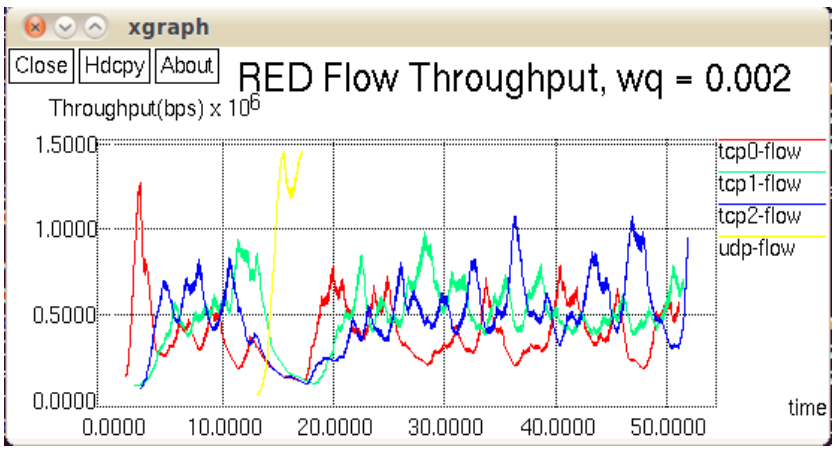
4.1.3 Mô phỏng với chính sách RED:

Các tham số được thiết lập cho RED như sau: minth = 5, maxth = 15, maxp = 0.1 và wq = 0.002. Kết quả mô phỏng được thể hiện ở các đồ thị trên hình 4.3.

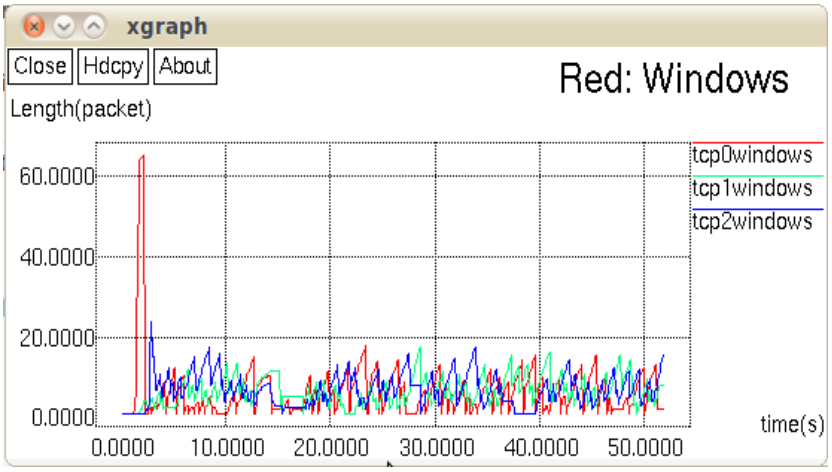
a) Kích thước hàng đợi trung bình của DropTail và RED



b) Thông lượng các luồng của RED



c) Kích thước cửa sổ của các luồng của RED



Hình 4.3. Các kết quả mô phỏng 1 với hàng đợi RED

Nhận xét:

- Chúng ta có thể thấy rằng, trong các giai đoạn đưa lưu lượng đột biến cbr vào, kích thước cửa sổ các kết nối tcp giảm xuống rất nhanh (hình 4.3c), kéo theo thông lượng của chúng giảm theo (hình 4.3b), nhưng sau khi ra khỏi giai đoạn đó thì các kết nối này nhanh chóng tăng kích thước cửa sổ lên, thông lượng vì thế nhanh chóng được hồi phục; mặt khác kích thước hàng đợi tăng lên nhưng nhanh chóng được kéo xuống;

- Trong giai đoạn không có đột biến (từ 20s trở đi) thì RED luôn duy trì được kích thước hàng đợi trung bình (ở khoảng 10 gói tin), kích thước hàng đợi hiện tại dao động ở mức nhỏ (10 ± 2).

Ngoài ra chúng tôi cũng đã so sánh độ trễ trung bình và độ lệch chuẩn của độ trễ trong toàn thời gian mô phỏng với hàng đợi DropTail và RED. Kết quả mô phỏng với hàng đợi DropTail được thể hiện trên bảng 4.1.

Kết nối	*Độ trễ trung bình (Mean delay)*	*Độ lệch chuẩn của độ trễ (Standard deviation of delay)*
TCP (s0-s8)	0.620883004129387	68.041217909686
TCP (s1-s7)	0.723091946788987	49.713802901559
TCP (s2-s6)	0.411800912368419	23.8705707028506
UDP (s0-s4)	0.410905662222221	7.18025976022592

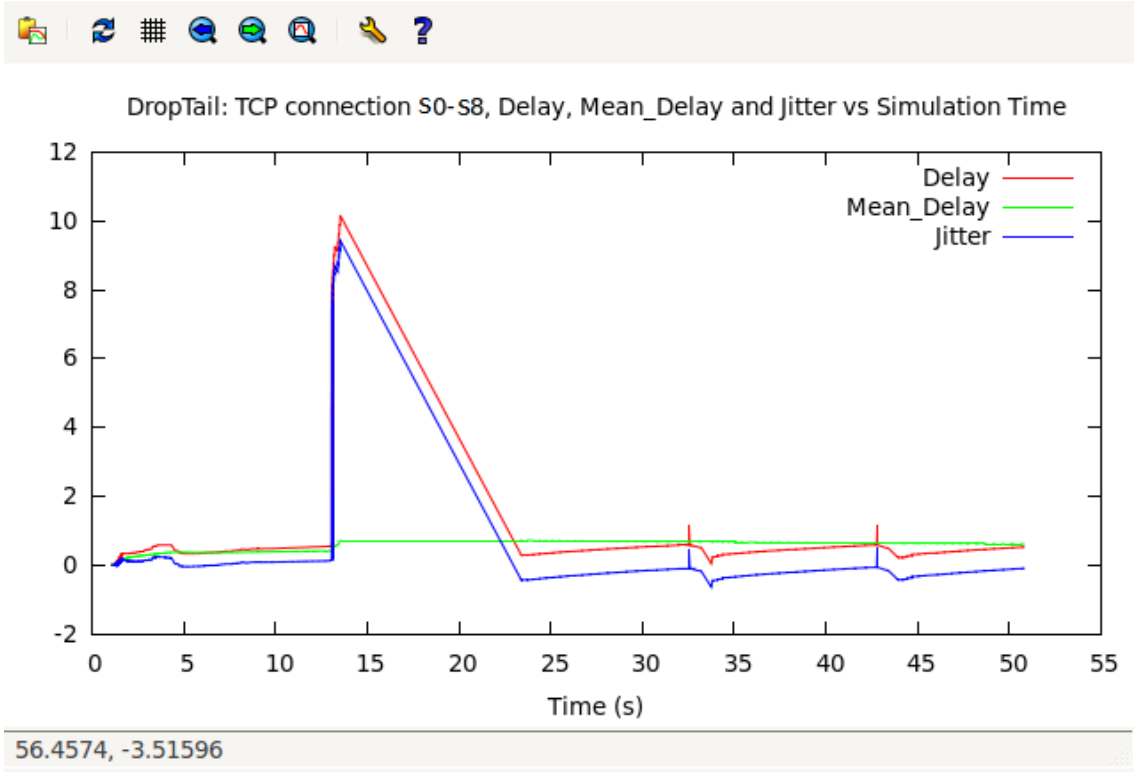
Bảng 4.1. So sánh độ trễ trung bình và độ lệch chuẩn của độ trễ với hàng đợi DropTail

Kết quả mô phỏng với hàng đợi RED được thể hiện trên bảng 4.2 dưới đây.

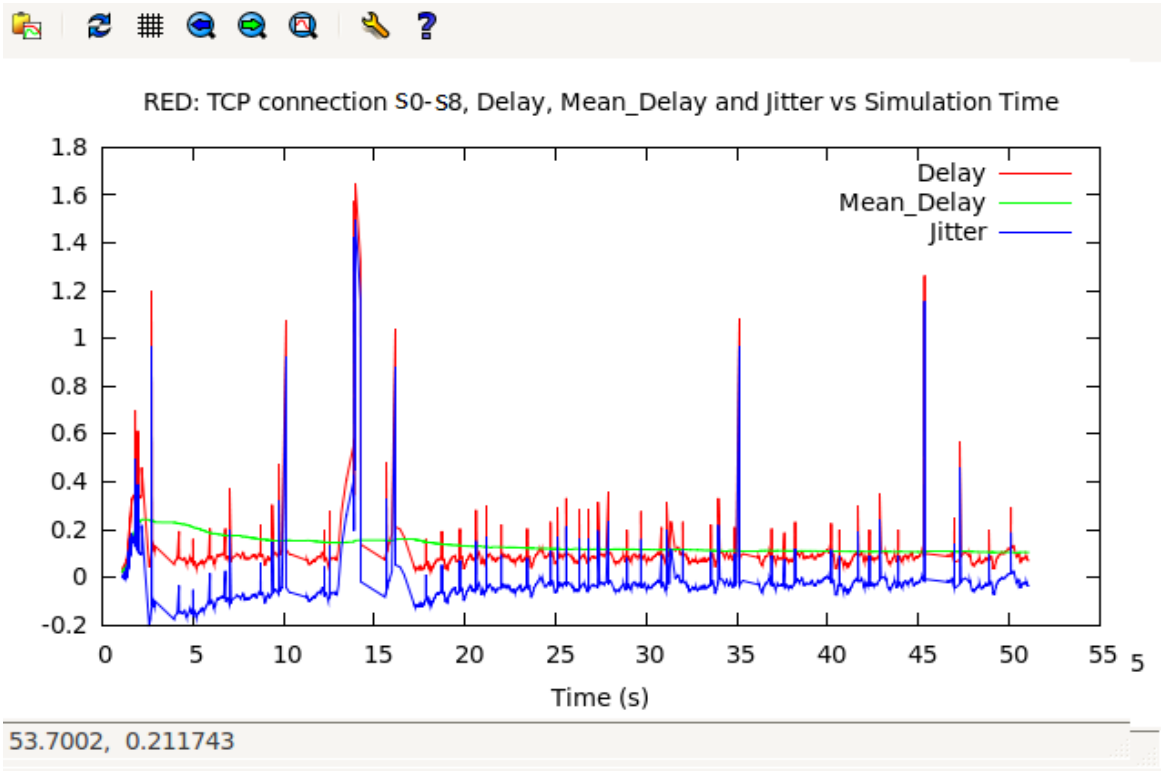
Kết nối	*Độ trễ trung bình (Mean delay)*	*Độ lệch chuẩn của độ trễ (Standard deviation of delay)*
TCP (s0-s8)	0.105534121971595	4.83272185887846
TCP (s1-s7)	0.0922332979999994	5.99194460746312
TCP (s2-s6)	0.0860523832394832	3.66091765635048
UDP (s0-s4)	0.11072745	3.69114334991382

Bảng 4.2. So sánh độ trễ trung bình và độ lệch chuẩn của độ trễ với hàng đợi RED

Hình 4.4 và 4.5 dưới đây là đồ thị hiển thị kết quả biểu diễn sự thay đổi của delay, mean_delay, jitter của kết nối TCP giữa s0-s8 theo thời gian mô phỏng của hàng đợi DropTail và hàng đợi RED.



Hình 4.4. Sự thay đổi của Delay, mean_delay, jitter của kết nối TCP giữa s0-s8 với hàng đợi DropTail

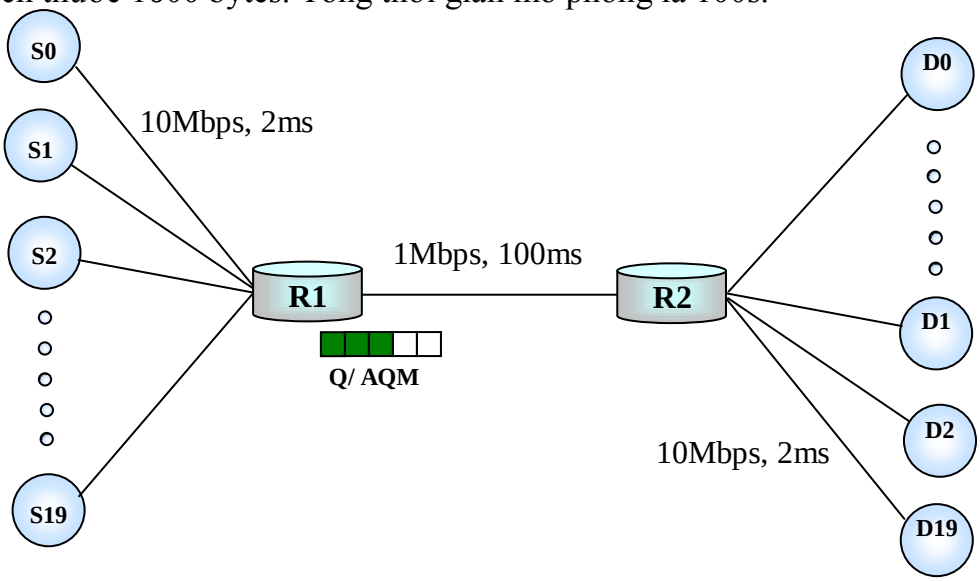


Hình 4.5. Sự thay đổi của Delay, mean_delay, jitter của kết nối TCP giữa s0-s8 với hàng đợi RED

Ở phần trên chúng ta đã thực hiện mô phỏng khi hệ thống trong trạng bình thường. Bây giờ chúng ta sẽ xem xét hệ thống mạng khi có tắc nghẽn xảy ra để đánh giá khả năng hấp thụ các lưu lượng đột biến của RED và DropTail.

Chúng tôi đã xây dựng mạng mô phỏng có số lượng các thực thể gửi và nhận lớn như trên hình 4.6. Bằng cách thay đổi hàng đợi DropTail, RED, A-RED, BLUE chúng ta sẽ so sánh hiệu năng của các chiến lược.

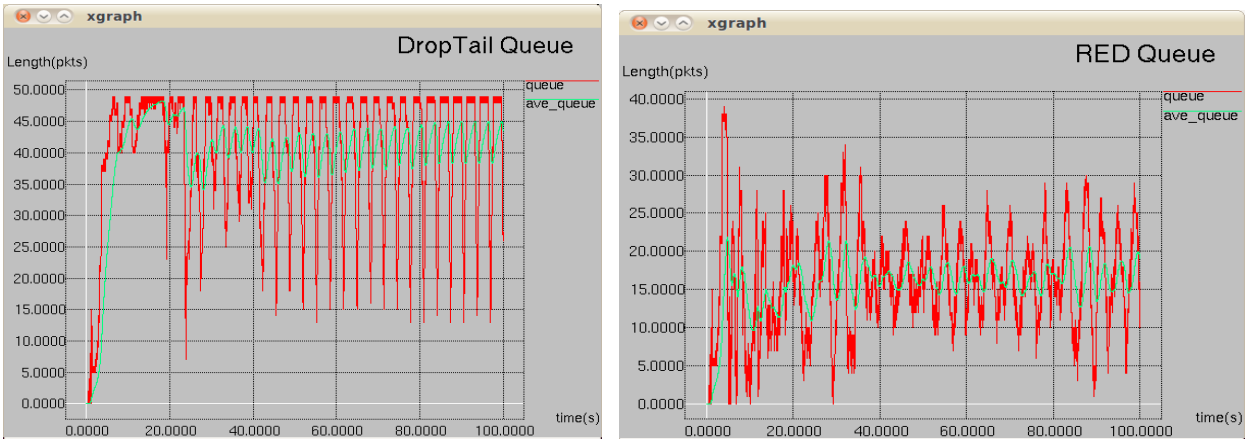
Mạng mô phỏng có tổng số nút là 40 bao gồm 20 nút nguồn (nút gửi dữ liệu) từ S0-S19 và 20 nút nhận D0-D19. Các thực thể gửi đều là TCP, kích thước cửa sổ gửi tối đa là 32 gói tin và đều xuất phát từ các nguồn sinh lưu lượng FTP. Các đường truyền đều là duplex-link, không lỗi; đường truyền từ các nút nguồn đến R1 có băng thông 10Mbps, độ trễ 2ms; đường truyền từ R2 đến các nút nhận đều là 10Mbps, 2ms. Các kết nối cùng chia sẻ đường truyền chung R1- R2 có băng thông, độ trễ lần lượt là 1Mbps, 100ms. Hàng đợi Q được đặt giữa R1-R2 có kích thước tối đa 50 gói tin, các luồng TCP gửi các gói tin có kích thước 1000 bytes. Tổng thời gian mô phỏng là 100s.



Hình 4.6. Cấu hình mạng mô phỏng RED/ A-RED/ BLUE

4.1.4. Khả năng hấp thụ các lưu lượng đột biến của RED

Sử dụng cấu hình mạng như hình 4.6: Các luồng ftp được đưa vào mạng, mỗi luồng cách nhau 2s. Chúng ta sẽ xem kết quả ở hình 4.7.



Hình 4.7. Kết quả mô phỏng 2 so sánh DropTail và RED

Nhìn vào các đồ thị trên hình 4.7 ta thấy rằng: khi ta đưa luồng lưu lượng đột biến vào mạng trong khoảng thời gian mô phỏng thì cả hai chiến lược DropTail và RED đều làm cho kích thước hàng đợi tăng. Nhưng với DropTail mỗi khi ta đưa luồng lưu lượng đột biến vào thì kích thước hàng đợi tăng đột ngột đến ngưỡng rồi giảm xuống rất nhanh: khoảng 20s đầu kích thước hàng đợi và kích thước hàng đợi trung bình dao động mạnh từ 0 đến 50 packet. Trong các khoảng thời gian còn lại của mô phỏng thì kích thước hàng đợi dao động ở ngưỡng từ 15 đến 50 packet, kích thước hàng đợi trung bình dao động trong khoảng 40±5 gói tin. Trong khi đó với RED thì kích thước hàng đợi dao động ổn định từ 5 đến 25 packet, kích thước hàng đợi trung bình dao động trong khoảng 16±3 gói tin.

Ngoài ra chúng tôi cũng đã thống kê một số giá trị trung bình trên toàn bộ thời gian mô phỏng và nhận được kết quả như trong bảng dưới 4.3 dưới đây.

Chiến lược	*Kích thước hàng đợi trung bình (gói tin)*	*Độ trễ hàng đợi trung bình (ms)*	*Hệ số sử dụng đường truyền (%)*
DropTail	40	213.33	95.51
RED	16	85.33	95.36

Bảng 4.3. Kết quả thống kê của mô phỏng 2 so sánh DropTail/RED

4.1.5. So sánh RED với Tail-Drop

Thông qua kết quả các mô phỏng về Tail-Drop và RED ở trên, chúng ta có thể đưa ra một số kết luận sau:

- DropTail không tránh được hiện tượng lock-out và global synchronization, không hỗ trợ sự chia sẻ dải thông công bằng giữa các kết nối; nhất là khi có lưu lượng bùng nổ thì hầu như toàn bộ đường truyền chỉ phục vụ cho lưu lượng bùng nổ đưa vào, không bảo vệ được các kết nối đang hoạt động.
- RED tránh được hiện tượng global synchronization, ngay cả khi có lưu lượng đột biến. Dựa trên mô phỏng ta thấy đột biến trong khoảng thời gian ngắn hạn được ngăn cản, đặc biệt là thông lượng được hồi phục rất nhanh sau khoảng thời gian tắc nghẽn; chia sẻ giải thông tương đối công bằng giữa các kết nối.
- RED duy trì kích thước hàng đợi nhỏ nên đạt được độ trễ thấp hơn rất nhiều so với RED, trong khi vẫn đảm bảo hệ số sử dụng đường truyền (bảng 4.3), vì vậy đạt được công suất đường truyền rất cao.

4.2. Đánh giá hiệu suất của chiến lược quản lý hàng đợi A-RED

Để kiểm chứng lại các đánh giá về A-RED bằng lý thuyết, chúng tôi đã tiến hành mô phỏng A-RED bằng NS-2. Chúng tôi vẫn sử dụng cấu hình mạng mô phỏng như ở hình 4.6. Ở đây tôi đã có điều chỉnh băng thông, độ trễ đường truyền giữa R1-R2 lần lượt là 2.5Mbps, 20ms. Mục đích sử dụng cấu hình mạng như trên để so sánh hiệu năng của A-RED với RED trong trường hợp mạng có đột biến lớn về lưu lượng. Dưới đây là phần trình bày chi tiết việc mô phỏng.

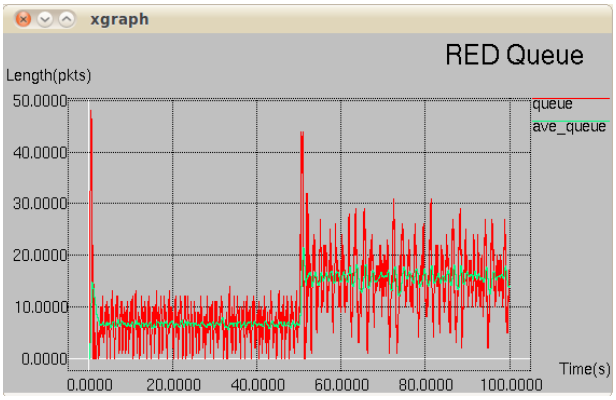
Các tham số RED được thiết lập là: min_{th} = 5, max_{th} = 15, max_p = 0.1 và w_q =

0.0025. Với A-RED, w_q được thiết lập tự động theo công thức (*), α = 0.02 và β = 0.9.

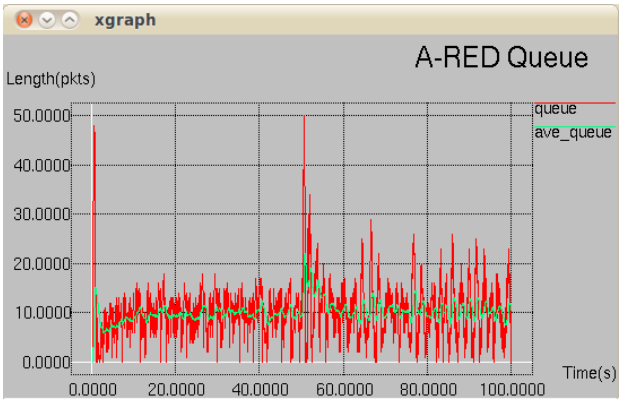
Với cấu hình mạng nêu trên, chúng tôi tiến hành 2 kịch bản mô phỏng ứng với hai cách gây đột biến: kịch bản mô phỏng 1 gây đột biến tăng lưu lượng, còn kịch bản mô phỏng 2 là đột biến giảm lưu lượng, chúng ta sẽ xem xét các kết quả cụ thể dưới đây.

4.2.1. Kịch bản mô phỏng 1: Tăng cường độ tắc nghẽn với các luồng lưu lượng

Kịch bản được thiết lập như sau: đầu tiên hai kết nối tcp0 và tcp1 được đưa vào mạng (ở 0.1s và 0.2s), đến nửa thời gian mô phỏng (giây thứ 50), 18 luồng mới (tcp2-tcp19) được đưa vào mạng, mỗi luồng cách nhau 0.1 giây. Mục đích của việc đưa các luồng tcp2 –tcp19 vào mạng để lưu lượng mạng được làm tăng đột ngột khi đó băng thông của các luồng truyền sẽ lớn hơn băng thông tại hàng đợi sẽ gây hiện tượng tắc nghẽn. Chúng ta sẽ theo dõi kết quả của các chiến lược trên các hình dưới đây.



Hình 4.8. RED với sự tăng cường độ tắc nghẽn



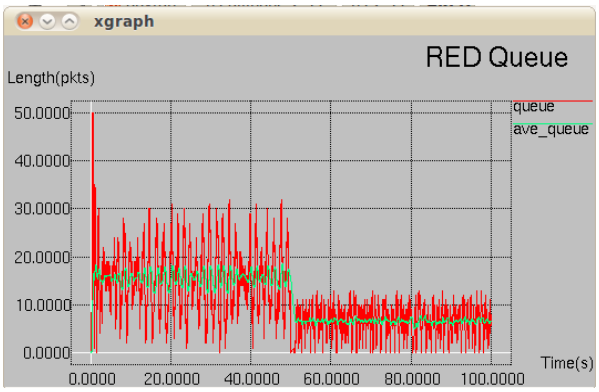
Hình 4.9. ARED với sự tăng cường độ tắc nghẽn

❖ **Nhận xét:**

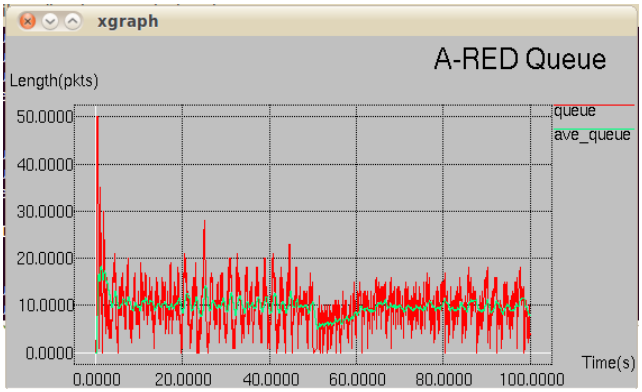
Trên hình 4.8 và 4.9 là đồ thị của kích thước hàng đợi hiện thời (màu đỏ) và kích thước hàng đợi trung bình (màu xanh) ứng với thuật toán RED và A-RED. Nhìn vào đồ thị chúng ta thấy rằng khi tắc nghẽn được tăng cường ở nửa thời gian mô phỏng (ở giây thứ 50), cả RED và A-RED đều làm kích thước hàng đợi tăng lên tối đa; dẫn tới kích thước hàng đợi trung bình tăng lên; với RED là từ 7 lên khoảng 16 gói tin, với A-RED là từ 9 đến 18 gói tin; tuy nhiên sau khoảng 10s (từ giây 60 trở đi), A-RED đã kéo kích thước hàng đợi trở về khoảng mục tiêu và dao động ở mức 10±3 gói tin, trong khi RED vẫn giữ kích thước trung bình ở mức cao (16±3 gói tin).

4.2.2. Kịch bản mô phỏng 2: Giảm cường độ tắc nghẽn với các luồng lưu lượng

Kịch bản được thiết lập như sau: đầu tiên tất cả các kết nối từ tcp0 đến tcp19 được đưa vào mạng (bắt đầu từ 0.1s, mỗi luồng cách nhau 0.1 giây), đến nửa thời gian mô phỏng (giây thứ 50), 18 luồng (từ tcp2-tcp19) ngừng hoạt động. Như vậy lưu lượng mạng được làm giảm đột ngột, phản ứng của từng chiến lược được thể hiện trong các đồ thị hình 4.10 và hình 4.11.



Hình 4.10 RED với sự giảm cường độ tắc nghẽn



Hình 4.11. ARED với sự giảm cường độ tắc nghẽn

❖ **Nhận xét:**

Nhìn vào đồ đồ thị trong trường hợp giảm cường độ tắc nghẽn ta thấy: với RED, tại thời điểm xảy ra đột biến, kích thước hàng đợi trung bình giảm xuống nhanh chóng và ổn định ở một mức mới thấp hơn (từ 16 ± 3 đến 7 ± 1 gói tin); còn với A-RED kích thước hàng đợi trung bình cũng giảm xuống, tuy nhiên mức giảm không đột ngột như RED (từ 10 ± 3 xuống 7 ± 1) và nó nhanh chóng được kéo lên và ổn định ở mức mục tiêu 10 ± 2 gói tin. Với các kết quả đã đưa ra ta thấy rằng hiệu năng của A-RED tốt hơn RED.

4.2.3. So sánh thuật toán RED và ARED

Thông qua các mô phỏng đã trình bày ở phần trên, chúng ta thấy được rằng thuật toán ARED có nhiều ưu điểm hơn so với thuật toán RED. ARED là phiên bản tiếp theo của RED do đó ARED khắc phục được những mặt hạn chế của thuật toán RED:

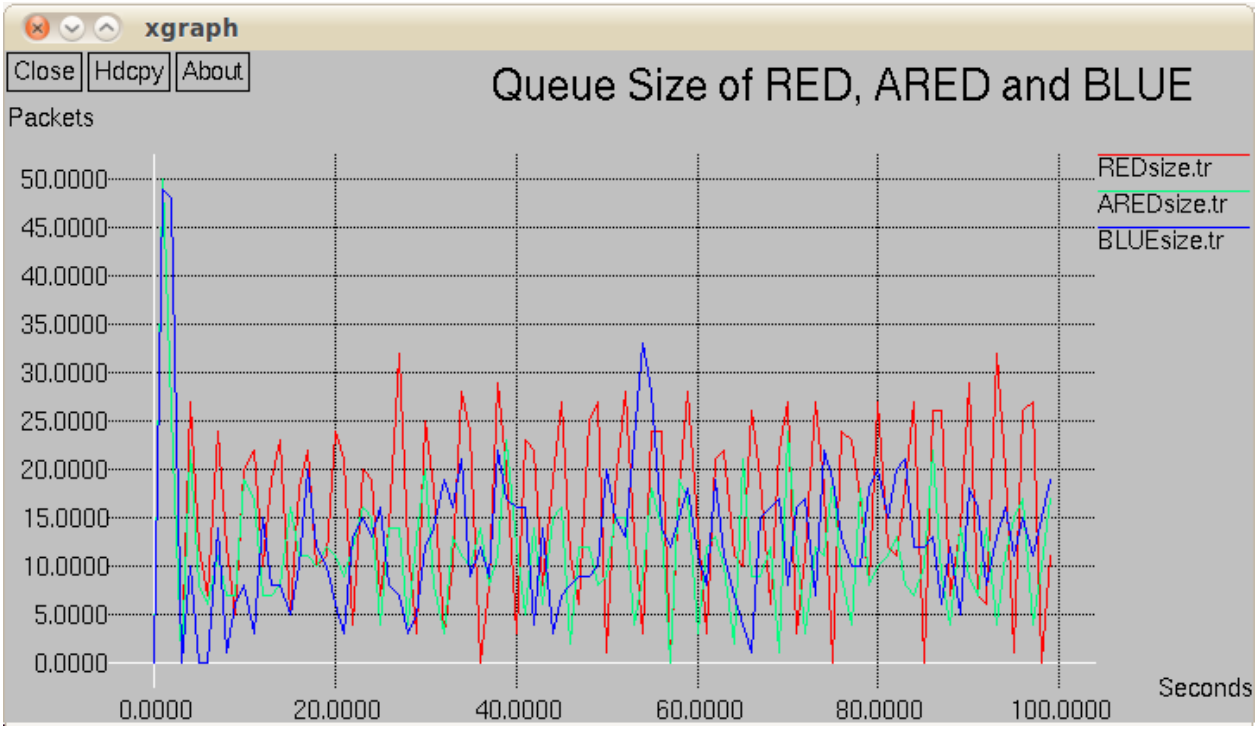
- RED quản lý hàng đợi dựa trên kích thước trung bình của hàng đợi nên kích thước trung bình hàng đợi thay đổi theo các mức tắc nghẽn và quá trình thiết lập các tham số. Điều này được thể hiện bằng việc khi tắc nghẽn xảy ra nhẹ hay max_p cao thì kích thước hàng đợi gần tới giá trị min_{th}. Khi tắc nghẽn trong mạng nặng hay kích thước hàng đợi trung bình bằng hoặc lớn hơn max_{th}. Kết quả trễ hàng đợi trong thuật toán RED phụ thuộc vào tải lưu lượng và các tham số, do đó mà trễ hàng đợi không thể đoán trước.

RED còn có nhược điểm là khả năng thông qua trong thuật toán này cũng phụ thuộc nhiều vào tải lưu lượng và các tham số.

4.3. Đánh giá hiệu suất của chiến lược quản lý hàng đợi BLUE

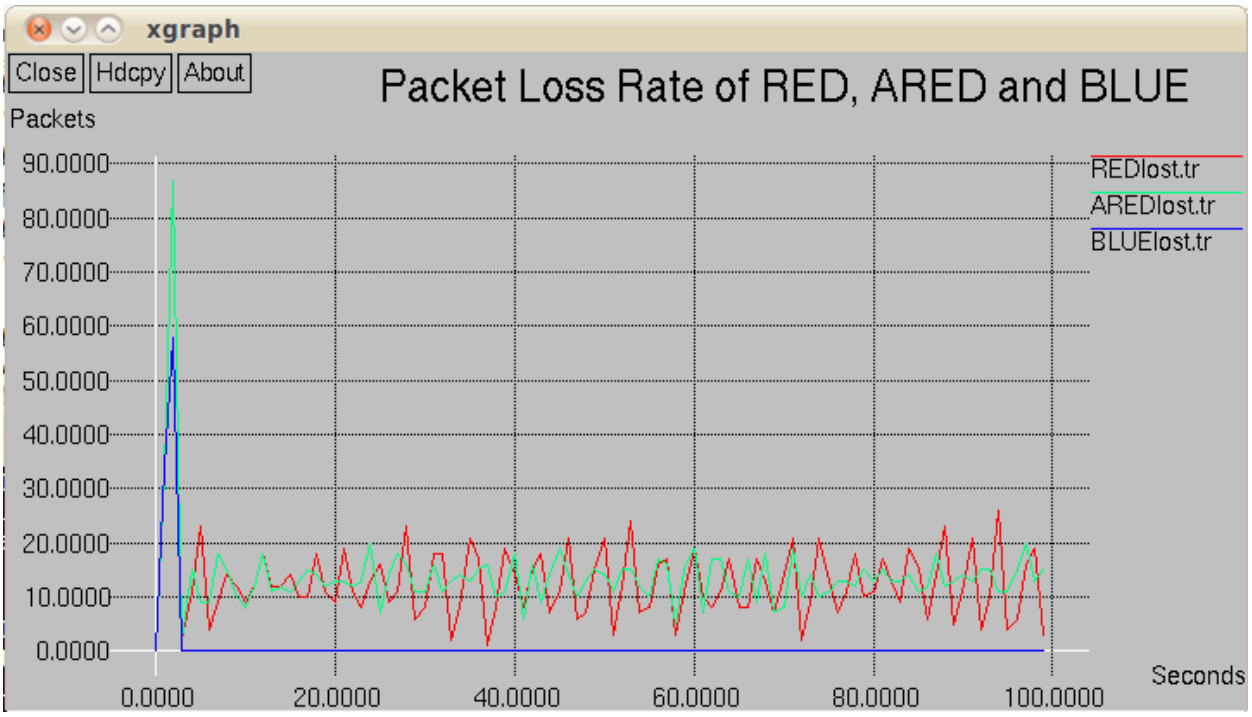
Trong phần này chúng tôi vẫn sử dụng cấu hình mạng như hình 4.6 và với việc thiết lập kịch bản mô phỏng và các tham số cho RED và ARED như mục 4.2 chúng ta sẽ xem xét các kết quả của từng giải thuật RED, A-RED và BLUE dựa trên các tham số: Kích thước hàng đợi trung bình, tỉ lệ mất gói tin và thông lượng sử dụng. Kết quả mô phỏng mạng được thể hiện ở trên các hình 4.12, 4.13 và 4.14.

Hình 4.12 cho ta thấy kích thước hàng đợi của RED dao động mạnh (từ 5packet – 25 packet), kích thước hàng đợi của ARED cũng dao động nhưng mức độ ít hơn so với RED, còn kích thước hàng đợi của BLUE thì mức độ dao động nhỏ hơn. Điều này chứng tỏ rằng độ trễ trung bình và độ lệch chuẩn của độ trễ của BLUE sẽ ít hơn so với RED và ARED.



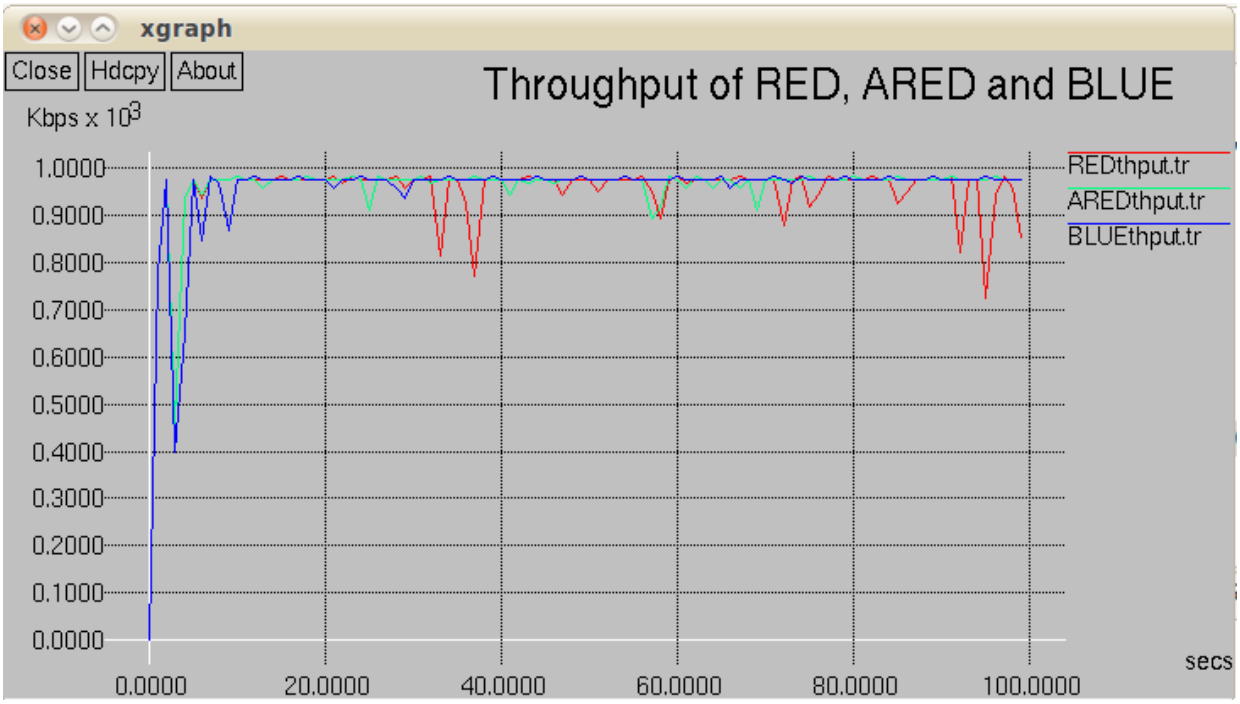
Hình 4.12. Kích thước hàng đợi của RED, A-RED và BLUE

Hình 4.13 cho thấy RED có tỉ lệ mất gói tin nhiều hơn so với ARED và BLUE, điều đó có nghĩa là BLUE có hiệu suất tốt hơn so với RED, ARED. Hàng đợi RED có nhiều gói tin bị mất ngay từ thời điểm bắt đầu của mô phỏng, hàng đợi ARED cũng có gói tin bị mất nhưng ít hơn so với RED, còn BLUE thì gần như không có gói tin bị loại bỏ



Hình 4.13. Tỉ lệ gói tin bị mất của RED, A-RED và BLUE

Hình 4.14 ta thấy trong cả 3 chiến lược băng thông luôn được sử dụng ở mức tối đa.



Hình 4.14. Thông lượng của RED, A-RED và BLUE

Như vậy dựa trên các kết quả đánh giá độ đo hiệu năng như: thông lượng, tỉ lệ mất gói tin, kích thước hàng đợi thì ta thấy rằng chiến lược BLUE có ưu điểm hơn hẳn so với chiến lược RED và A-RED ở tất cả các mặt.

KẾT LUẬN

Các thuật toán quản trị hàng đợi tích cực AQM có nhiều ưu điểm nổi bật hơn so với các chiến lược quản lý hàng đợi và lập lịch. Mục tiêu của quản lý hàng đợi tích cực là duy trì một xác suất chủ động loại bỏ gói hợp lý nhằm hạn chế được tình trạng tắc nghẽn trong khi vẫn đảm bảo được chất lượng của các luồng lưu lượng và tính công bằng trong quan hệ giữa các luồng lưu lượng khi trạng thái động học của mạng thay đổi.

Trên cơ sở nghiên cứu các ưu khuyết điểm của các giải thuật quản lý hàng đợi tích cực RED, ARED, BLUE và các giải pháp cải tiến nhằm nâng cao hiệu năng và chất lượng dịch vụ cho truyền thông đa phương tiện. Luận văn đã đạt những kết quả như sau:

- Nghiên cứu cơ sở lý thuyết về truyền thông đa phương tiện và các yêu cầu về chất lượng dịch vụ.
- Nghiên cứu các chiến lược quản lý hàng đợi tích cực: Đối với RED: Chúng tôi đã chỉ ra được những tính năng ưu việt của RED so với DropTail, đồng thời chỉ ra những hạn chế của RED trong những điều kiện mạng cụ thể. Đó là lý do cho sự phát triển của các thuật toán sau nó. Đối với A-RED: Bằng mô phỏng chúng tôi nhận thấy A-RED có những ưu thế nổi bật hơn so với RED. Đó là khả năng tự động thiết lập các tham số, tự động hiệu chỉnh xác xuất loại bỏ để duy trì kích thước hàng đợi trung bình mong muốn, trong khi vẫn đảm bảo được thông lượng cao cho mạng. Đối với BLUE: BLUE là một chiến lược quản lý hàng đợi dựa trên tải nạp, qua đó dự đoán khả năng sử dụng đường truyền liên kết, xác định tắc nghẽn và đưa ra cách xử lý. Mục đích của chiến lược là điều tiết gói tin vào nút mạng để ổn định lưu lượng gói tin đến, nhằm duy trì sự ổn định cho mạng.

Hướng phát triển tiếp theo của luận văn là mở rộng và phát triển các giải thuật cải tiến quản lý hàng đợi tích cực AQM nhằm hạn chế tối đa tắc nghẽn để mạng luôn duy trì được sự ổn định cao nhất về chất lượng. Thông qua đó có thể áp dụng các giải thuật cải tiến trên các mô hình mạng phức hợp, và mạng có tổn hao như các mạng không dây, di động và ứng dụng cài đặt trên môi trường mạng thực tế.

TÀI LIỆU THAM KHẢO

Tài liệu tiếng Việt:

[1]. Vũ Duy Lợi, Nguyễn Đình Việt, Ngô Thị Duyên, Lê Thị Hợi (2004), "*Đánh giá hiệu suất chiến lược quản lý hàng đợi RED bằng bộ mô phỏng NS*", Kỷ yếu Hội thảo Khoa học Quốc gia lần thứ hai về Nghiên cứu, Phát triển và Ứng dụng Công nghệ Thông tin và Truyền thông (ICT.rda'04), (Hà nội, 24-25/9/2004). NXB Khoa học và Kỹ thuật, Hà Nội, 5/2005, trang 394-403.

[2]. PGS.TS. Nguyễn Đình Việt, *Bài giảng Mạng và Truyền số liệu nâng cao*, 2008.

[3]. PGS.TS. Nguyễn Đình Việt, *Bài giảng đánh giá hiệu năng mạng máy tính*, 2008.

[4]. Lê Đình Danh (2007), *Thuật toán quản lý hàng đợi A-RIO*, Luận văn cao học, Khoa Công nghệ thông tin, Đại học Quốc gia Hà nội

[5]. Vũ Xuân Bảo (2011), *Đánh giá hiệu quả đảm bảo QoS cho truyền thông đa phương tiện của chiến lược quản lý hàng đợi động WRED,* Luận văn cao học, Khoa Công nghệ thông tin, Đại học Quốc gia Hà nội

[6]. Cao Diệp Thắng (2014), *Đánh giá hiệu năng và chất lượng dịch vụ mạng máy tính,* Luận án tiến sĩ, Đại học Bách Khoa Hà Nội

Tài liệu Tiếng Anh

[7]. NS Simulator for beginners - Eitan Altman & Tania Jimenez

[8]. Network advanced modeling in NS-2 - Giovanni Perbellini

[9]. Richelle Adams (2013), "Active Queue Management: A Survey", *IEEE communications surveys & tutorials*, Vol. 15, No. 3

[10]. C. V. Hollot, V. Misra, D. Towsley, and W. Gong (2002), "Analysis and design of controllers for AQM routers supporting TCP flows", *IEEE Trans. on Automat. Control*, No. 47

[11]. Diep Thang Cao, Thuc Hai Nguyen, Linh Giang Nguyen (2013) *Improving the video transmission quality over ip network.* Proceedings of the fifth International Conference on Ubiquitous and Future Network, ICUFN 2013, Da Nang, Vietnam, July. 2013

[12]. Lin Dong, Morris Robert (1997) *Dynamics of Random Early Detection*. Proceedings of ACM SIGCOMM, Vol.27. 1997

[13]. Delgermaa KHISHGEE, *Comparing Red and Blue algorithms in NS2*, Dokuz EylÜl University Graduate School of Natural and Applied Sciences, 2013

[14]. Floyd S., Jacobson V. (1993), "Random early detection gateways for congestion avoidance", *IEEE/ACM Trans. On Networking*, Vol. 1, No. 4

[15]. V. Firoiu and M. Borden (2000) *A study of active queue management for congestion control*. Proceeding of IEEE INFORCOM 2000, vol. 3, Tel-Aviv, Israel, Mar. 2000.

[16]. Thiruchelvi G, Raja J (2008) *A Survey On Active Queue Management Mechanisms. International Journal of Computer Science and Network Security* (IJCSNS), Vol.8 No.12, 2008.

[17]. Michael Welzl (2005), *Network Congestion Control Managing Internet Traffic*, John Wiley & Sons Ltd.

[18]. M. Natarajan and V. Santhi (2011) *Active Queue Management Algorithm for TCP Networks Congestion Control*. European Journal of Scientific Research

ISSN 1450-216X Vol.54 No.2 2011

[19]. G.F.Ali Ahammed, Reshma Banu (2010), "Analyzing the Performance of Active Queue Management Algorithms", *International journal of Computer Networks & Communications (IJCNC),* Vol.2 No.2

[20]. B. Zheng, M. Atiquzzaman (2006) DSRED: *A New Queue Management Scheme for the Next Generation Networks*. IEICE Trans. on Communications, Vol. E89-B, No. 3,2006

[21]. Bartek Peter Wydrowski (2003), *Techniques in Internet Congestion Control*, Electrical and Electronic Engineering Department The University of Melbourne.

[22]. Lin Dong, Morris Robert (1997) *Dynamics of Random Early Detection*. Proceedings of ACM SIGCOMM, Vol.27. 1997

[23]. Arash Dana1 and Ahmad Malekloo (2010), "Performance Comparison between Active and Passive Queue Management", *JCSI International Journal of Computer Science Issues,* Vol. 7, Issue 3, No. 5

[24]. W. Feng, K. Shin, D. Kandlur, and D. Saha (2002), "The BLUE Active Queue Management Algorithms", *IEEE/ACM Transactions on Networking*, Vol. 10, No. 4

[25]. Julio Orozco, David Ros (2003), "An Adaptive RIO (A-RIO) Queue Management Algorithm", *Reseach Report PI-1526,* IRISA.

[26]. W. Feng, K. Shin, D. Kandlur, and D. Saha (1999), *A Self-Configuring RED Gateway*. In Proc. IEEE INFOCOM

[27]. Bartek Peter Wydrowski (2003), *Techniques in Internet Congestion Control*, Electrical and Electronic Engineering Department The University of Melbourne.

[28]. S. Floyd, R. Gummadi, and S. Shenker. *"Adaptive RED: an algorithm for increasing the robustness of RED's Active Queue Management",* 2001.

[29]. Clark, D., Fang, W.: *Explicit Allocation of Best-Effort Packet Delivery Service. IEEE/ACM Transactions on Networking 6* (1998)

[30]. David D.Clark, Wenjia Fang (1998), "*Explixit Allocation of Best Effort Packet Delivery Service*", Labratory for Computer Sciences Computer Science Department, Massachusetts Institute of Technology Princeton University.

[31]. Park, W.H., Bahk, S., Kim, H.: *A Modied RIO Algorithm that Alleviates the Bandwidth Skew Problem in Internet Differentiated Service. In: Proceedings of IEEE ICC 2000.*

[32]. Malouch, N., Liu, Z.: *Performance Analysis of TCP with RIO Routers. Research Report RR-4469, INRIA (2002)*