

**ĐẠI HỌC HUẾ  
TRƯỜNG ĐẠI HỌC KHOA HỌC**

**LÊ VĂN HÒA**

**ĐIỀU KHIỂN CÔNG BẰNG LƯỜNG  
TRONG MẠNG CHUYỂN MẠCH CHÙM QUANG**

**LUẬN ÁN TIẾN SĨ KHOA HỌC MÁY TÍNH**

**HUẾ - NĂM 2019**

**ĐẠI HỌC HUẾ  
TRƯỜNG ĐẠI HỌC KHOA HỌC**

**LÊ VĂN HÒA**

**ĐIỀU KHIỂN CÔNG BẰNG LUỒNG  
TRONG MẠNG CHUYỂN MẠCH CHùm QUANG**

**CHUYÊN NGÀNH: KHOA HỌC MÁY TÍNH**

**MÃ SỐ: 9480101**

**LUẬN ÁN TIẾN SĨ KHOA HỌC MÁY TÍNH**

**Người hướng dẫn khoa học:**

**1. PGS. TS. VÕ VIẾT MINH NHẬT**

**2. TS. NGUYỄN HOÀNG SƠN**

**HUẾ - NĂM 2019**

## **LỜI CAM ĐOAN**

Tôi xin cam đoan đây là công trình nghiên cứu do tôi thực hiện dưới sự hướng dẫn của PGS. TS. Võ Viết Minh Nhật và TS. Nguyễn Hoàng Sơn. Những nội dung trong các công trình đã được công bố chung với các tác giả khác đã được sự chấp thuận của đồng tác giả khi đưa vào luận án. Các số liệu và kết quả nghiên cứu được trình bày trong luận án là trung thực, khách quan và chưa được công bố bởi tác giả nào trong bất kỳ công trình nào khác.

**Nghiên cứu sinh**

**Lê Văn Hòa**

## **LỜI CẢM ƠN**

Trước hết tôi xin bày tỏ lòng biết ơn chân thành và sâu sắc đến PGS. TS. Võ Viết Minh Nhật và TS. Nguyễn Hoàng Sơn là những người Thầy đã tận tình hướng dẫn chỉ bảo, động viên và giúp đỡ để tôi có thể hoàn thành được luận án này.

Tôi xin trân trọng cảm ơn sự giúp đỡ của Quý Thầy Cô trong Khoa Công nghệ Thông tin - Trường Đại học Khoa học Huế đã quan tâm, giúp đỡ, hướng dẫn trong suốt quá trình học tập.

Tôi xin trân trọng cảm ơn Quý Thầy Cô, Ban chủ nhiệm Khoa Du lịch - Đại học Huế đã tạo điều kiện thuận lợi trong công tác để tôi có đủ thời gian hoàn thành luận án này. Tôi xin cảm ơn Quý Thầy Cô, cán bộ quản lý Phòng Đào tạo Sau đại học – Trường Đại học Khoa học, Đại học Huế đã giúp đỡ tôi hoàn thành kế hoạch học tập.

Cuối cùng tôi xin chân thành cảm ơn các bạn đồng nghiệp, người thân trong gia đình luôn động viên, giúp đỡ tôi về mọi mặt trong suốt quá trình nghiên cứu, học tập.

**Nghiên cứu sinh**

**Lê Văn Hòa**

# MỤC LỤC

<b>MỤC LỤC</b> .....	<b>iv</b>
<b>DANH MỤC CÁC TỪ VIẾT TẮT</b> .....	<b>vi</b>
<b>CÁC KÝ HIỆU TOÁN HỌC ĐƯỢC SỬ DỤNG</b> .....	<b>x</b>
<b>DANH MỤC CÁC HÌNH VẼ</b> .....	<b>xiii</b>
<b>DANH MỤC CÁC BẢNG</b> .....	<b>xvi</b>
<b>MỞ ĐẦU</b> .....	<b>1</b>
<b>CHƯƠNG 1. TỔNG QUAN VỀ CÔNG BẰNG TRONG MẠNG CHUYỂN MẠCH CHÙM QUANG</b> .....	<b>7</b>
1.1 Các mô hình chuyển mạch trong truyền thông quang.....	8
1.2 Nguyên tắc hoạt động của mạng OBS.....	10
1.3 Các hoạt động bên trong mạng OBS .....	12
1.3.1 Tập hợp chùm .....	12
1.3.2 Báo hiệu chùm .....	14
1.3.3 Lập lịch chùm.....	16
1.3.4 Xử lý tranh chấp chùm.....	17
1.4 Vấn đề công bằng trong mạng OBS .....	18
1.4.1 Khái niệm và phân loại công bằng trong mạng OBS.....	18
1.4.2 Công bằng độ trễ .....	20
1.4.3 Công bằng thông lượng.....	21
1.4.4 Công bằng khoảng cách .....	22
1.4.5 Kết hợp công bằng thông lượng và công bằng khoảng cách .....	26
1.4.6 Đánh giá các giải pháp công bằng tại nút biên mạng OBS.....	27
1.5 Các mục tiêu nghiên cứu của luận án .....	29
1.6 Tiểu kết Chương 1 .....	30
<b>CHƯƠNG 2. TẬP HỢP CHÙM GIẢM ĐỘ TRỄ VÀ CÔNG BẰNG ĐỘ TRỄ</b>	<b>31</b>
2.1 Mô hình tập hợp chùm giảm độ trễ.....	32
2.1.1 Vấn đề độ trễ trong hoạt động tập hợp chùm .....	32
2.1.2 Các công trình nghiên cứu liên quan .....	32

2.1.3 Phương pháp tập hợp chùm giảm độ trễ iBADR.....	42
2.1.4 Phương pháp tập hợp chùm giảm độ trễ OBADR.....	48
2.1.5 Ảnh hưởng của trọng số $\alpha$ đến OBADR.....	52
2.1.6 Ảnh hưởng của OBADR đến hoạt động lập lịch chùm .....	55
2.2 Mô hình tập hợp chùm công bằng độ trễ .....	59
2.2.1 Các công trình nghiên cứu liên quan .....	59
2.2.2 Phương pháp tập hợp chùm công bằng độ trễ BADF.....	60
2.3 Tiểu kết Chương 2.....	72
<b>CHƯƠNG 3. CÔNG BẰNG THÔNG LƯỢNG DỰA TRÊN CẤP PHÁT BĂNG</b>	
<b>THÔNG VÀ ĐÁP CHÙM</b>	<b>73</b>
3.1 Mô hình cấp phát băng thông công bằng dựa trên thông lượng .....	74
3.1.1 Giới thiệu về cấp phát băng thông công bằng .....	74
3.1.2 Các công trình nghiên cứu liên quan .....	75
3.1.3 Phương pháp cấp phát băng thông công bằng dựa trên thông lượng TFBA77	
3.1.4 Phân tích ảnh hưởng của TFBA đến việc lập lịch tại liên kết ra .....	87
3.1.5 Nhận xét.....	91
3.2 Mô hình đáp chùm hiệu quả băng thông và công bằng thông lượng.....	91
3.2.1 Các công trình nghiên cứu liên quan .....	91
3.2.2 Phương pháp đáp chùm .....	93
3.2.3 Nhận xét.....	99
3.3 Tiểu kết Chương 3.....	99
<b>KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN CỦA LUẬN ÁN.....</b>	<b>100</b>
<b>DANH MỤC CÁC CÔNG TRÌNH LIÊN QUAN ĐẾN LUẬN ÁN.....</b>	<b>101</b>
<b>TÀI LIỆU THAM KHẢO.....</b>	<b>102</b>

## DANH MỤC CÁC TỪ VIẾT TẮT

<b>Từ viết tắt</b>	<b>Thuật ngữ tiếng Anh</b>	<b>Diễn giải ý nghĩa</b>
ACK	Acknowledgement,	Gói điều khiển thông báo việc truyền thông/lập lịch thành công
NACK	Negative Acknowledgement	Gói điều khiển thông báo việc truyền thông/lập lịch thất bại
AON	All-Optical Network	Mạng toàn quang
ATM	Asynchronous Transfer Mode	Kiểu truyền thông không đồng bộ
BADF	Burst Assembly for Delay Fairness	Tập hợp chùm công bằng độ trễ
BADR-EAT*	BADR with Extra Assembly Time	Tập hợp chùm giảm độ trễ với thời gian tập hợp chùm mở rộng
BASTP*	Burst Assembly based on Size and Time Prediction	Tập hợp chùm giảm độ trễ dựa trên dự đoán kích thước và thời gian tập hợp
BCP	Burst Control Packet	Gói điều khiển chùm
BLD	burst length-based differentiation	Phân biệt dựa vào kích thước chùm
DFI	Delay Fairness Index	Chỉ số công bằng độ trễ
DWDM	Density Wavelength Division Multiplexing	Ghép kênh phân chia bước sóng mật độ cao
FDL	Fiber Delay Line	Đường trễ quang
FDM	Frequency Division Multiplexing	Ghép kênh phân chia tần số
FPP	Fair Prioritized Preemption	Điều khiển dựa trên ưu tiên công bằng
GMPLS	Generalized Multiprotocol Label Switching	Chuyển mạch nhãn đa giao thức suy rộng

<b>Từ viết tắt</b>	<b>Thuật ngữ tiếng Anh</b>	<b>Diễn giải ý nghĩa</b>
HBP	Hop Based Preemption	Điều khiển dựa trên số chặng
Hop-FCR	Hop-by-hop routing using Forward Channel Reservation	Định tuyến từng chặng với đặt trước kênh theo hướng truyền đi
Hop-LC	Hop-by-hop routing using Link Connectivity	Định tuyến từng chặng dựa trên số kết nối của liên kết ra
Hop-N-FCR	Hop-by-hop routing using Neighborhood Forward Channel Reservation	Định tuyến từng chặng với đặt trước kênh theo hướng truyền về
iBADR	improved Burst Assembly for Delay Reduction	Tập hợp chùm giảm độ trễ cải tiến
IE-BADR*	Immediate Estimation-based BADR	Tập hợp chùm giảm độ trễ dựa trên ước tính nhanh
IP	Internet Protocol	Giao thức mạng Internet
JET	Just Enough Time	Giao thức báo hiệu với thời gian đặt trước tài nguyên vừa đủ
JIT	Just In Time	Giao thức báo hiệu với đặt trước tài nguyên ngay lập tức
JK-BADR*	Jacobson/Karels algorithm-based BADR	Tập hợp chùm giảm độ trễ dựa trên giải thuật Jacobson/Karels
LAUT	Latest Available Unscheduled Time	Thời điểm chưa được lập lịch sau cùng nhất
LSOS	Link State based Offset Selection	Chọn thời gian offset dựa trên trạng thái liên kết
MGRP	Monitoring Group Drop Probability	Xác suất đánh rơi theo nhóm
MMFP	Max-Min Fairness Preemption	Ưu tiên dựa trên công bằng max-



<b>Từ viết tắt</b>	<b>Thuật ngữ tiếng Anh</b>	<b>Diễn giải ý nghĩa</b>
		min
MTBA-TP*	Mixed-Threshold Burst Assembly based on Traffic Prediction	Tập hợp chùm giảm độ trễ dựa trên dự đoán lưu lượng
O/E/O	Optical/Electronic/Optical	Chuyển đổi quang - điện - quang
OBADR	Optimal Burst Assembly for Delay Reduction	Tập hợp chùm giảm độ trễ tối ưu
OBS	Optical Burst Switching	Chuyển mạch chùm quang
OCS	Optical Circuit Switching	Chuyển mạch kênh quang
OPS	Optical Packet Switching	Chuyển mạch gói quang
OTD	Offset Time based Differentiation	Phân biệt dựa trên thời gian bù đắp
OXC	Optical Cross Connect	Thiết bị chuyển mạch quang
POQA*	Prediction and Offset QoS Assembly	Tập hợp chùm hỗ trợ QoS dựa trên thời gian offset và dự đoán
QoS	Quality of Service	Chất lượng dịch vụ
QDBAP	QoS Differentiation Burst Assembly with Padding	Tập hợp chùm phân biệt chất lượng dịch vụ kết hợp với đệm chùm
RCBP	Resource Consumption Based Preemptive	Ưu tiên dựa trên tiêu thụ tài nguyên
RDFP	Rate and Distance Fairness Preemption	Ưu tiên công bằng tốc độ và khoảng cách
RTT	Round-Trip Time	Thời gian khứ hồi
RFP	Rate Fairness Preemption	Ưu tiên công bằng tốc độ
TFBA	Throughput-based Fair Bandwidth Allocation	Cấp phát băng thông công bằng dựa trên thông lượng
TFI	Throughput Fairness Index	Chỉ số công bằng thông lượng

<b>Từ viết tắt</b>	<b>Thuật ngữ tiếng Anh</b>	<b>Diễn giải ý nghĩa</b>
TW- EWMA	Time Windows based Exponentially Weighted Moving Average	Trung bình dịch chuyển có trọng số dựa trên cửa sổ thời gian
WDM	Wavelength Division Multiplexing	Ghép kênh phân chia bước sóng

*\* Các phương pháp được luận án đặt tên để dễ dàng cho việc tham chiếu.*

## CÁC KÝ HIỆU TOÁN HỌC ĐƯỢC SỬ DỤNG

Ký hiệu	Ý nghĩa
$AB_i$	Bảng thông cung cấp cho luồng $i$
$AT_i$	Thông lượng thực tế của luồng $i$
$B_{min}$	Ngưỡng kích thước chòm tối thiểu
$B(i)$	Kích thước hàng đợi $i$
$D(i)$	Độ trễ gói tin trong hàng đợi $i$
$E_i$	Tải hiệu quả của kết nối $i$
$F_i$	Tỉ lệ cấp phát băng thông công bằng cho hàng đợi $i$
$K$	Tổng số luồng (kết nối)
$L$	Độ dài chòm hoàn thành của lần tập hợp chòm hiện thời
$L^e$	Độ dài chòm ước tính của lần tập hợp chòm hiện thời
$L_w$	Độ dài chòm trong khoảng thời gian ước tính
$L_w(i)$	Độ dài chòm trong khoảng thời gian ước tính của hàng đợi $i$
$L_{min}$	Ngưỡng độ dài chòm tối thiểu
$L_{max}$	Ngưỡng độ dài chòm tối đa
$L(i)$	Độ dài chòm hoàn thành của hàng đợi $i$
$L^e(i)$	Độ dài chòm ước tính của hàng đợi $i$
$L_j$	Độ dài chòm hoàn thành ở lần tập hợp thứ $j$
$L_j^e$	Độ dài chòm ước tính ở lần tập hợp thứ $j$
$M$	Số lần tập hợp chòm sau cùng nhất
$P_i^U$	Xác suất mất chòm của phần luồng tốt của luồng $i$
$P_i^O$	Xác suất mất chòm của phần luồng xấu của luồng $i$
$P^U$	Tổng xác suất mất chòm của phần luồng tốt

<b>Ký hiệu</b>	<b>Ý nghĩa</b>
$P^O$	Tổng xác suất mất chùm của phần luồng xấu
$P$	Tổng xác suất mất chùm của liên kết ra
$P_i$	Tổng xác suất mất chùm của luồng $i$
$Q$	Tổng số hàng đợi
$R_E$	Lỗi ước tính trong lần tập hợp chùm hiện thời
$\bar{R}_E$	Lỗi ước tính trung bình trong các lần tập hợp chùm
$t_1$	Thời điểm gửi gói điều khiển
$t_1(i)$	Thời điểm gửi gói điều khiển của hàng đợi $i$
$t_2$	Thời điểm gửi chùm dữ liệu
$t_2(i)$	Thời điểm gửi chùm dữ liệu của hàng đợi $i$
$T_a$	Ngưỡng thời gian tập hợp chùm; $T_a$ cũng là độ trễ tập hợp chùm (thời gian mà các gói tin đợi trong hàng đợi trước khi được gộp vào một chùm)
$T_a(i)$	Ngưỡng thời gian tập hợp chùm của hàng đợi $i$
$T_o$	Thời gian <i>offset</i> (offset time)
$T_o(i)$	Thời gian <i>offset</i> của hàng đợi $i$
$T_e(i)$	Ngưỡng thời gian ước tính trên hàng đợi $i$
$T_j$	Ngưỡng thời gian tập hợp chùm thứ $j$ trong mô hình tập hợp chùm giảm độ trễ BASTP
$T_w$	Cửa sổ thời gian ước tính
$W$	Tổng số bước sóng của liên kết ra
$\lambda_i$	Tốc độ đến của luồng $i$
$\lambda^U$	Tốc độ đến của phần luồng tốt
$\lambda^O$	Tốc độ đến của phần luồng xấu
$\lambda_{cur}$	Tốc độ gói tin đến của lần tập hợp chùm hiện thời

<b>Ký hiệu</b>	<b>Ý nghĩa</b>
$\lambda_{prev}$	Tốc độ gói tin đến của lần tập hợp chòm trước đó.
$\lambda_{cur}(i)$	Tốc độ gói tin đến của lần tập hợp chòm hiện thời tại hàng đợi $i$
$\lambda_{avg}$	Tốc độ gói tin đến trung bình của những lần tập hợp chòm trước đó
$\lambda_{avg}(i)$	Tốc độ gói tin đến trung bình của những lần tập hợp chòm trước đó của hàng đợi $i$
$\mu$	Tốc độ phục vụ trung bình
$1/\mu$	Độ dài chòm trung bình
$\omega$	Tỉ lệ băng thông có thể sử dụng tối đa của liên kết ra
$\varepsilon$	Tham số điều khiển trong mô hình POQA
$\eta, \beta$ và $\varphi$	Tham số điều khiển trong mô hình JK-BADR
$\sigma_i$	Hệ số ưu tiên (trọng số) công bằng của hàng đợi $i$ trong công thức tính DFI và TFI

## DANH MỤC CÁC HÌNH VẼ

Hình 1.1 So sánh sự khác biệt giữa các loại chuyển mạch quang tại nút lõi OBS.....	9
Hình 1.2 Quá trình tập hợp chùm và tách chùm tại các nút biên OBS.....	11
Hình 1.3 Sự tách biệt giữa kênh điều khiển và kênh truyền dữ liệu.....	11
Hình 1.4 Kiến trúc chung của nút biên vào OBS .....	12
Hình 1.5 Đặc điểm luồng chùm được sinh ra sau tập hợp, trong đó ON là khoảng băng thông bị chiếm dụng và OFF là khoảng băng thông nhàn rỗi giữa 2 chùm liên tiếp .....	14
Hình 1.6 Nguyên tắc hoạt động của giao thức JET.....	16
Hình 1.7: Phân loại công bằng dựa trên vị trí thực hiện.....	19
Hình 1.8 So sánh độ trễ đệm chùm giảm được của mô hình tập hợp chùm giảm độ trễ.....	20
Hình 1.9 Một ví dụ của vấn đề công bằng khoảng cách trong đó chùm càng gần đến đích có xác suất mất mát càng cao .....	23
Hình 1.10 Kiến trúc nút biên vào OBS được nghiên cứu với các mô đun chức năng được bổ sung.....	30
Hình 2.1 Hai mô đun chức năng điều khiển công bằng được đề xuất: mô đun giảm độ trễ và mô đun công bằng độ trễ, trong kiến trúc nút biên vào OBS .....	31
Hình 2.2 So sánh các phương pháp tập hợp chùm giảm độ trễ .....	33
Hình 2.3 So sánh tỉ lệ lỗi ước tính trung bình của IE-BADR, JK-BADR, POQA, BADR-EAT, MTBA-TP và BASTP với tải chuẩn hóa đến 0.5 .....	39
Hình 2.4 Phân bố tỉ lệ lỗi ước tính của IE-BADR, JK-BADR, POQA, BADR-EAT, MTBA-TP và BASTP trong 100 lần tập hợp chùm liên tiếp .....	39
Hình 2.5 Tỉ lệ lỗi ước tính trung bình gần như không đổi với tải chuẩn hóa từ 0.1 đến 0.9....	39
Hình 2.6 So sánh số gói tin thừa trong 100 chùm sinh ra đầu tiên.....	40
Hình 2.7 Phương pháp dự đoán theo cửa sổ của TW-EWMA .....	42
Hình 2.8 Tỉ lệ lỗi ước tính trung bình của các phương pháp tập hợp chùm trước đây với phương pháp tập hợp chùm cải tiến (iBADR).....	46
Hình 2.9 Phân bố lỗi ước tính của 100 chùm sinh ra đầu tiên của BASTP và iBADR.....	47
Hình 2.10 Số gói tin thừa trong 100 chùm sinh ra đầu tiên.....	47

Hình 2.11 So sánh tỉ lệ lỗi ước tính trung bình giữa các phương pháp tập hợp giảm độ trễ.....	50
Hình 2.12 Phân bố lỗi ước tính trong 100 lần tập hợp chùm liên tiếp của phương pháp OBADR với BASTP .....	51
Hình 2.13 Số gói tin thừa trong 100 chùm sinh ra liên tiếp .....	51
Hình 2.14 Trường hợp tốc độ luồng các gói tin đến không có nhiều biến đổi.....	52
Hình 2.15 Trường hợp tốc độ luồng các gói tin đến có nhiều biến đổi với các trường hợp tăng/giảm đột biến .....	53
Hình 2.16 So sánh lỗi ước tính trung bình với trường hợp $\alpha$ động và $\alpha$ tĩnh ( $\alpha = 0.5$ ) khi thay đổi thời gian tập hợp chùm ( $T_a$ ) từ 2.5 ms đến 7.0 ms .....	54
Hình 2.17 Sự biến thiên giá trị $\alpha$ động trong 100 lần tập hợp chùm liên tiếp với $T_a=6$ ms và $T_a=3$ ms.....	55
Hình 2.18 Hai hoạt động chính tại nút biên: tập hợp chùm và lập lịch chùm ở cổng ra.....	56
Hình 2.19 So sánh tỉ lệ mất chùm giữa OBADR và tập hợp chùm truyền thống.....	58
Hình 2.20 So sánh tỉ lệ mất chùm của OBADR với tập hợp chùm truyền thống.....	58
Hình 2.21 Một ví dụ về 3 ngưỡng thời gian tập hợp chùm và 3 giá trị thời gian offset.....	59
Hình 2.22 Ví dụ về 3 chùm ưu tiên có $x_i$ phân bố trong không gian ( $D, T_a$ ).....	62
Hình 2.23 So sánh chỉ số DFI giữa BADF và POQA .....	66
Hình 2.24 So sánh giá trị $x_i = D(i)/T_a(i)$ giữa 3 lớp ưu tiên giữa giải thuật BADF và giải thuật POQA .....	66
Hình 2.25 So sánh giá trị $T_a(i)$ của 3 lớp ưu tiên với giải thuật BADF.....	67
Hình 2.26 So sánh độ trễ đệm chùm trung bình giữa BADF và POQA trong trường hợp không xem xét đến độ trễ tăng thêm do ước tính sai.....	68
Hình 2.27 So sánh độ trễ đệm chùm trung bình giữa BADF và POQA trong trường hợp có xem xét đến độ trễ tăng thêm do ước tính sai.....	68
Hình 2.28 So sánh độ trễ đệm chùm trung bình của 3 lớp ưu tiên giữa BADF và POQA trong trường hợp không xem xét đến độ trễ tăng thêm.....	69
Hình 2.29 So sánh độ trễ đệm chùm trung bình của 3 lớp ưu tiên giữa giải thuật BADF và giải thuật POQA trong trường hợp xem xét đến độ trễ tăng thêm .....	69
Hình 2.30 So sánh lỗi ước tính giữa giải thuật BADF và giải thuật POQA.....	70

Hình 2.31 So sánh tỉ lệ lãng phí băng thông giữa giải thuật BADF và giải thuật POQA .....	70
Hình 2.32 So sánh tỉ lệ gửi lại giữa giải thuật BADF và giải thuật POQA.....	70
Hình 2.33 So sánh lỗi ước tính trên mỗi lớp ưu tiên giữa giải thuật BADF và giải thuật POQA .....	71
Hình 3.1 Hai mô đun chức năng điều khiển công bằng: mô đun đắp chòm và mô đun công bằng thông lượng, được bổ sung trong kiến trúc nút biên vào OBS .....	74
Hình 3.2 Ví dụ về cấp phát băng thông công bằng của 2 luồng chia sẻ cùng một liên kết.....	74
Hình 3.3 Kiến trúc nút biên vào OBS hỗ trợ đa dạng dịch vụ.....	77
Hình 3.4 Sự hội tụ của $y_1$ , $y_2$ và $y_3$ qua quá trình xử lý tranh chấp chòm .....	82
Hình 3.5 Hình thái mạng mô phỏng .....	83
Hình 3.6 So sánh tỉ lệ mất byte giữa 3 kết nối của phương pháp TFBA trong 2 trường hợp (1) tổng tải không vượt quá khả năng liên kết và (2) tải luồng 3 tăng đột biến vượt quá khả năng liên kết .....	84
Hình 3.7 So sánh tỉ lệ mất byte trung bình trên cả 3 kết nối của TFBA, RFP và MMFP .....	84
Hình 3.8 So sánh tỉ lệ mất byte trung bình trên cả 3 kết nối giữa TFBA, RFP và MMFP với thời gian mô phỏng tăng lên 10s .....	85
Hình 3.9 So sánh tỉ lệ mất byte của Kết nối 1 giữa TFBA, RFP và MMFP .....	85
Hình 3.10 So sánh tỉ lệ mất byte của Kết nối 2 giữa TFBA, RFP và MMFP .....	86
Hình 3.11 So sánh tỉ lệ mất byte của Kết nối 3 giữa TFBA, RFP và MMFP .....	86
Hình 3.12 So sánh chỉ số TFI của phương pháp TFBA với RFP và MMFP.....	87
Hình 3.13 Ví dụ về 3 luồng đến được nhóm vào phần luồng tốt và phần luồng xấu .....	88
Hình 3.14 Sơ đồ chuyển trạng thái trong mô hình Markov đa chiều .....	89
Hình 3.15 So sánh tỉ lệ mất chòm giữa mô hình phân tích và mô phỏng với TFBA .....	91
Hình 3.16 Ví dụ về (a) QoS dựa vào thời gian offset và (b) QoS dựa vào kích thước chòm...	92
Hình 3.17 Một ví dụ về mô hình đắp chòm trên 3 lớp: (a) trước khi đắp chòm; (b) sau khi đắp chòm .....	93
Hình 3.18 So sánh số byte đắp giữa QDBAP và POQA .....	97
Hình 3.19 Độ dài chòm hoàn thành thuộc class <sub>0</sub> trong 50 lần tập hợp chòm liên tiếp.....	97
Hình 3.20 So sánh dựa trên chỉ số công bằng thông lượng giữa QDBAP và POQA.....	98
Hình 3.21 So sánh công bằng thông lượng (dựa trên tỉ lệ tải thực tế class <sub>i</sub> trên khả năng đáp ứng băng thông $TB_i(y_i)$ ) giữa POQA và QDBAP .....	99



## DANH MỤC CÁC BẢNG

Bảng 1.1 So sánh các giải pháp xử lý tranh chấp trong mạng OBS .....	18
Bảng 1.2 Các loại công bằng trong mạng OBS .....	19
Bảng 1.3 Các công bố về giải pháp công bằng luồng trong mạng OBS .....	27
Bảng 2.1 So sánh các phương pháp tập hợp chòm giảm độ trễ đã công bố .....	36
Bảng 2.2 Trung bình kích thước tối đa và tối thiểu của các chòm sinh ra .....	38
Bảng 2.3 Ảnh hưởng cặp giá trị ngưỡng ( $L_{min}, L_{max}$ ) đến lỗi ước tính (với tải chuẩn hóa 0,5) .....	41
Bảng 2.4 Lỗi ước tính $\bar{R}_E$ với tải chuẩn hóa thay đổi và các giá trị $\alpha$ từ 0.1 đến 0.9 .....	53
Bảng 3.1 Tỷ lệ thông lượng đạt được tối đa trên mỗi liên kết với tải chuẩn hóa đến khác nhau .....	78
Bảng 3.2 Các tham số sử dụng trong mô hình phân tích .....	87
Bảng 3.3 Độ dài trung bình của những chòm hoàn thành với tải chuẩn hóa đến 0.2 .....	96

# MỞ ĐẦU

## 1. Tính cấp thiết của đề tài

Sự phát triển không ngừng của Internet trong một vài thập niên trở lại đây, cùng với sự bùng nổ các loại hình dịch vụ truyền thông, đã làm gia tăng không ngừng nhu cầu về băng thông truyền thông. Điều này đã đặt ra một thách thức lớn trong việc tìm kiếm công nghệ truyền thông phù hợp nhằm nâng cao khả năng truyền thông của mạng thế hệ mới. Mạng quang, cùng với công nghệ ghép kênh bước sóng WDM, đã mang đến một giải pháp hiệu quả đáp ứng được những yêu cầu này [24], [36].

Truyền thông quang, từ khi ra đời vào đầu thập niên 90 cho đến nay, đã trải qua nhiều thế hệ phát triển: từ những mô hình định tuyến bước sóng ban đầu với những đường quang (*lightpath*) đầu cuối dành riêng cho đến các mô hình chuyển mạch gói quang [36] được đề xuất gần đây, với ý tưởng được lấy từ các mạng chuyển mạch gói điện tử. Tuy nhiên, với một số hạn chế về mặt công nghệ, như không thể sản xuất các bộ đệm quang (tương tự bộ nhớ RAM trong mạng điện) hay các bộ chuyển mạch gói quang ở tốc độ nano giây, chuyển mạch gói quang chưa thể trở thành hiện thực. Một giải pháp thỏa hiệp là mô hình chuyển mạch chùm quang (OBS).

Một đặc trưng tiêu biểu của truyền thông trong mạng chuyển mạch chùm quang là phần (gói) điều khiển BCP tách rời với phần (chùm) dữ liệu (*data burst*). Nói một cách khác, để thực hiện truyền một chùm quang, gói điều khiển được hình thành và được gửi đi trước một khoảng thời gian *offset* đủ để đặt trước tài nguyên và cấu hình chuyển mạch tại các nút trung gian dọc theo hành trình mà chùm quang sẽ đi qua từ nút nguồn đến nút đích. Thêm vào đó, mạng OBS dành riêng một số kênh (bước sóng) cho gói tin điều khiển, trong khi các kênh còn lại được dùng cho việc truyền dữ liệu. Như vậy, việc truyền gói điều khiển hoàn toàn tách rời với phần dữ liệu về mặt không gian (trên kênh truyền khác) và cũng như về mặt thời gian (gửi đi trước một khoảng thời gian *offset*) [65].

Với cách truyền tải dữ liệu như mô tả, rõ ràng mạng OBS không cần đến các vùng đệm quang để lưu tạm thời các chùm quang trong khi chờ đợi việc xử lý chuyển

mạch tại các nút lõi, cũng như không yêu cầu các chuyển mạch tốc độ nano giây. Tuy nhiên, cách truyền thông này cũng đặt ra một áp lực là làm thế nào để một gói điều khiển có thể kịp đặt trước tài nguyên và cấu hình chuyển mạch thành công tại các nút lõi, đảm bảo cho việc chuyển tiếp chùm quang đi sau nó. Đó chính là nhiệm vụ của các hoạt động như đặt trước tài nguyên, lập lịch, xử lý tắc nghẽn ... Ngoài ra một vấn đề khác cũng được nhiều nhà nghiên cứu mạng OBS quan tâm là làm sao đảm bảo được sự công bằng (*fairness*) giữa các luồng truyền thông khác nhau chia sẻ cùng liên kết bên trong mạng OBS.

Trong mạng máy tính, vấn đề công bằng được hiểu là việc phân phối các nguồn tài nguyên mạng cho các ứng dụng khác nhau sao cho đạt được công bằng về phân bổ tài nguyên mạng [38]. Nghiên cứu vấn đề công bằng trong mạng máy tính thường nhắm đến 2 mục tiêu: (1) cải tiến cấu trúc mạng bằng cách thêm các khối chức năng (*modules*) về phân bổ tài nguyên công bằng và (2) đưa ra một kịch bản mới nhằm đạt được mục tiêu công bằng. Do đó công bằng trong mạng máy tính thường được chia thành 2 loại, đó là công bằng vĩ mô (*macro fairness*) và công bằng vi mô (*micro fairness*) [38]. Công bằng vi mô là nhằm đạt đến sự công bằng trong việc phân phối tài nguyên mạng một cách mịn hơn cho các lớp ưu tiên khác nhau; trong khi công bằng vĩ mô nghiên cứu xử lý các vấn đề rộng hơn (trên toàn mạng) mà ở đó có thể có sự kết hợp của các vấn đề công bằng vi mô giữa các nút mạng khác nhau.

Trong mạng OBS, vấn đề công bằng được nghiên cứu theo 3 hướng chính: công bằng về độ trễ (*delay fairness*) [69], công bằng về thông lượng (*throughput fairness*) [53] và công bằng về khoảng cách (*distance fairness*) [10]. Việc đảm bảo công bằng giữa các luồng chia sẻ chung tài nguyên trong mạng OBS có một ý nghĩa rất quan trọng, một mặt nhằm vừa đảm bảo sự phân biệt chất lượng dịch vụ đã cam kết, mặt khác tối ưu hiệu năng truyền thông của mỗi luồng và toàn mạng (chẳng hạn, dựa trên tỉ lệ mất mát dữ liệu, tỉ lệ sử dụng băng thông, tỉ lệ độ trễ đầu cuối ...).

## **2. Động lực nghiên cứu**

Hiện đã có một số nghiên cứu về vấn đề công bằng trong mạng OBS được đề xuất mà có thể được phân thành 2 nhóm tiếp cận chính dựa trên vị trí thực hiện:

- Nhóm giải pháp công bằng tại nút biên và

- Nhóm giải pháp công bằng tại nút lõi.

Với nhóm giải pháp công bằng tại nút biên, có 2 hướng nghiên cứu chính gồm: (1) công bằng độ trễ và (2) công bằng thông lượng. Với công bằng độ trễ, đã có một vài đề xuất trong [69], [70] trong đó ý tưởng chung là gửi sớm các chùm có mức ưu tiên cao nhằm giảm độ trễ của chúng. Với công bằng thông lượng, ý tưởng của các đề xuất trong [67], [51], [53] là phân bổ băng thông công bằng cho các kết nối (*connections*) chia sẻ chung cùng một liên kết (*link*). Các phương pháp này chủ yếu sử dụng tiếp cận ánh xạ công bằng max-min được đề xuất đối với mạng IP truyền thống thành điều khiển công bằng trong mạng OBS.

Với nhóm giải pháp công bằng tại nút lõi, vấn đề công bằng được biết đến là công bằng khoảng cách [25], [42], [50], [62], trong đó giải pháp cho vấn đề công bằng là xử lý các trường hợp không công bằng về mất mát dữ liệu giữa các luồng có hành trình dài so với luồng có hành trình ngắn hơn.

Trong mạng OBS, nút biên đóng một vai trò quan trọng trong điều khiển công bằng luồng, bởi vì:

1. Nút biên điều khiển lưu lượng của các luồng (kết nối đầu cuối) một cách công bằng trước khi truyền vào bên trong mạng lõi; các nút lõi lúc này chủ yếu là xử lý công bằng các luồng đã được đưa vào;
2. Chỉ có nút biên mới có các bộ đệm, nên vấn đề điều khiển công bằng về độ trễ, thông lượng... được thực hiện dễ dàng hơn và
3. Nút lõi không có bộ đệm nên xử lý công bằng ở nút lõi gần như phụ thuộc vào các hoạt động điều khiển tại nút biên.

Dựa vào những đặc điểm đó luận án tập trung vào việc nghiên cứu điều khiển công bằng tại nút biên, với hai hoạt động chính là điều khiển công bằng độ trễ và điều khiển công bằng thông lượng.

Cho đến nay đã có một số nghiên cứu về hai loại công bằng này tại nút biên, nhưng vẫn còn một số vấn đề cần cải tiến và hoàn thiện hơn:

- **Đối với công bằng độ trễ:** các giải pháp trong [69], [70] sử dụng khái niệm gửi sớm gói điều khiển trước khi chùm được hoàn thành [47], [48], [63], [66] và mở rộng trên các hàng đợi khác nhau. Tuy nhiên, các giải pháp này

vẫn còn một số hạn chế như: lỗi ước tính (độ lệch giữa kích thước chùm hoàn thành và kích thước chùm ước tính) vẫn còn lớn; trong một số trường hợp sự công bằng về độ trễ bị vi phạm, nên cần có các giải pháp tốt hơn; chưa đề xuất đại lượng (chỉ số đo) để so sánh hiệu quả công bằng độ trễ giữa các giải pháp đã đề xuất.

- **Đối với công bằng thông lượng:** các tác giả trong [67], [51], [53] đã ánh xạ vấn đề công bằng max-min trong mạng IP thành tỉ lệ mất mát dữ liệu tương ứng trong mạng OBS. Tuy nhiên, việc sử dụng công thức ErlangB như là một độ đo lý thuyết cho tỉ lệ mất mát chỉ có thể áp dụng được cho loại luồng dữ liệu đến có phân bố Poisson. Thực tế lưu lượng Internet chủ yếu là các luồng non-Poisson [2] nên cần thiết phải có một cách tiếp cận khác về công bằng thông lượng sao cho có thể áp dụng cho nhiều loại luồng dữ liệu khác nhau. Một đại lượng (chỉ số đo) cho việc đánh giá hiệu quả đối với các giải pháp công bằng thông lượng cũng chưa được đề xuất.

Các vấn đề chưa được giải quyết trên chính là động lực để luận án tiến hành nghiên cứu, cải tiến và đề xuất mới một số phương pháp điều khiển công bằng luồng tại nút biên vào nhằm nâng cao hiệu quả truyền thông của mạng OBS.

### 3. Mục tiêu nghiên cứu

Mục tiêu nghiên cứu của luận án là nghiên cứu, cải tiến và đề xuất mới một số giải pháp điều khiển công bằng luồng tại nút biên vào nhằm nâng cao hiệu quả truyền thông của mạng OBS. Cụ thể:

- Nghiên cứu và đề xuất một số cải tiến về tập hợp chùm giảm độ trễ nhằm làm giảm độ trễ truyền thông qua mạng OBS;
- Nghiên cứu và đề xuất giải pháp tập hợp chùm công bằng độ trễ nhằm đồng thời phân biệt QoS theo độ trễ, làm giảm độ trễ và công bằng về độ trễ giữa các luồng ưu tiên khác nhau.
- Nghiên cứu và đề xuất giải pháp điều khiển công bằng thông lượng, mà có thể áp dụng cho các loại luồng đến Poisson và non-Poisson.
- Nghiên cứu và đề xuất giải pháp đáp chùm sau tập hợp chùm nhằm nâng cao

hiệu quả sử dụng băng thông và đảm bảo công bằng thông lượng.

#### 4. Đối tượng và phạm vi nghiên cứu

- *Đối tượng nghiên cứu:* Các mô hình, giải thuật tập hợp chùm và điều khiển công bằng luồng trong mạng chuyển mạch chùm quang.

- *Phạm vi nghiên cứu:* Nút biên mạng chuyển mạch chùm quang.

#### 5. Phương pháp nghiên cứu

- *Phương pháp nghiên cứu lý thuyết:* Tổng hợp các công bố liên quan đến các mô hình, giải thuật điều khiển công bằng độ trễ và công bằng thông lượng trong mạng OBS. Phân tích, đánh giá ưu và khuyết điểm của các đề xuất đã công bố để làm cơ sở cho việc cải tiến hoặc đề xuất mới.

- *Phương pháp mô phỏng, thực nghiệm:* Cài đặt các giải thuật cải tiến và đề xuất mới nhằm chứng minh tính đúng đắn của các giải thuật này. Hệ mô phỏng NS2 [71], gói mô phỏng Obs-0.9a tạo dữ liệu mô phỏng và các phương pháp điều khiển công bằng luồng được cài đặt bằng ngôn ngữ Java/Eclipse.

#### 6. Cấu trúc luận án

Luận án bao gồm phần mở đầu, ba chương nội dung, phần kết luận và danh mục các tài liệu tham khảo. Cụ thể:

Chương 1 “**Tổng quan về công bằng trong mạng chuyển mạch chùm quang**” giới thiệu về các mô hình chuyển mạch trong truyền thông quang, nguyên tắc hoạt động của mạng OBS và vấn đề công bằng luồng trong mô hình mạng này. Trên cơ sở phân tích và đánh giá các giải pháp xử lý không công bằng khác nhau trong mạng OBS, các vấn đề nghiên cứu của luận án được xác định trong chương này.

Chương 2 “**Tập hợp chùm giảm độ trễ và công bằng độ trễ**” trình bày các cải tiến và đề xuất mới của luận án về tập hợp chùm giảm độ trễ và công bằng độ trễ bao gồm: (1) giải pháp tập hợp chùm giảm độ trễ iBADR và OBADR khi xem xét trên từng hàng đợi tập hợp chùm và (2) đề xuất mô hình tập hợp chùm với điều khiển công bằng độ trễ BADF khi xem xét đồng thời trên các hàng đợi có mức ưu tiên khác nhau.

Chương 3 “**Công bằng thông lượng dựa trên cấp phát băng thông và đáp chùm**” trình bày đề xuất giải pháp điều khiển công bằng thông lượng TFBA áp dụng

được cho nhiều loại luồng đến khác nhau và đề xuất mô hình đắp chòm sau tập hợp QDBAP nhằm nâng cao hiệu quả sử dụng băng thông và tăng tính công bằng thông lượng.

“**Kết luận và hướng phát triển của luận án**” nêu những đóng góp của luận án và hướng phát triển.

# CHƯƠNG 1. TỔNG QUAN VỀ CÔNG BẰNG TRONG MẠNG CHUYỂN MẠCH CHÙM QUANG

Sự gia tăng nhanh chóng của các dịch vụ “đói” băng thông trong thời gian gần đây đã đặt ra một nhu cầu lớn về băng thông, vốn đã vượt quá khả năng đáp ứng của cơ sở hạ tầng truyền thông Internet hiện có. Tăng khả năng băng thông và giảm chi phí truyền thông đang là động lực cho việc phát triển Internet toàn quang thế hệ mới.

Đã có một số đề xuất nhằm tận dụng các lợi thế của truyền thông quang, trong đó các mô hình chuyển mạch quang được đặc biệt quan tâm. Đề xuất đầu tiên là mô hình chuyển mạch kênh quang (OCS) dựa trên việc định tuyến bước sóng theo một đường quang (*lightpath*) duy nhất được thiết lập sẵn trên mỗi liên kết từ nguồn đến đích. Một thay thế cho mô hình OCS là chuyển mạch gói quang (OPS), trong đó phần điều khiển của gói quang được chuyển đổi quang/điện/quang (O/E/O) và được xử lý trong môi trường điện tại mỗi nút trung gian, trong khi phần dữ liệu phải chờ một khoảng thời gian nhất định để được chuyển tiếp đến nút tiếp theo [9], [65].

Nhằm cung cấp một cơ sở hạ tầng Internet toàn quang khả thi và linh hoạt, mô hình chuyển mạch chùm quang (OBS) do đó đã được đề xuất [16], [25] với hai đặc điểm khác biệt quan trọng:

- Dữ liệu đến từ các mạng truy cập được tập hợp (gộp) thành các chùm (*burst*) tại nút biên vào của mạng OBS và có thể được ghép kênh theo các cấp độ khác nhau.
- Dữ liệu và gói điều khiển được truyền trên các kênh (bước sóng) tách biệt nhau và chỉ có kênh điều khiển phải chịu chi phí chuyển đổi O/E/O.

Chương này của luận án đầu tiên sẽ giới thiệu các mô hình chuyển mạch khác nhau trong truyền thông quang và lý do vì sao OBS là khả thi đối với Internet toàn quang thế hệ tiếp theo. Nguyên tắc hoạt động và các hoạt động bên trong của mạng OBS, như tập hợp, báo hiệu, lập lịch và xử lý tắc nghẽn chùm, sẽ được trình bày tiếp theo, trong đó các giải thuật tập hợp chùm và đặc điểm của luồng sau khi tập hợp sẽ được tập trung mô tả. Các vấn đề công bằng luồng, bao gồm công bằng độ trễ, công



bằng thông lượng và công bằng khoảng cách trong mạng OBS theo đó sẽ được phân tích để chỉ ra rằng có một nhu cầu cần có thêm các nghiên cứu mới để tìm ra các giải pháp nhằm đảm bảo và nâng cao hiệu quả công bằng truyền thông trong mạng OBS.

### **1.1 Các mô hình chuyển mạch trong truyền thông quang**

Những cải tiến gần đây của kỹ thuật ghép kênh đa bước sóng (WDM) đã cho phép một sợi quang có thể mang đến 256 bước sóng độc lập [1]. Kết quả thực nghiệm của NEC và Alcatel cho thấy rằng mỗi bước sóng có thể truyền dữ liệu với tốc độ 10Tps.

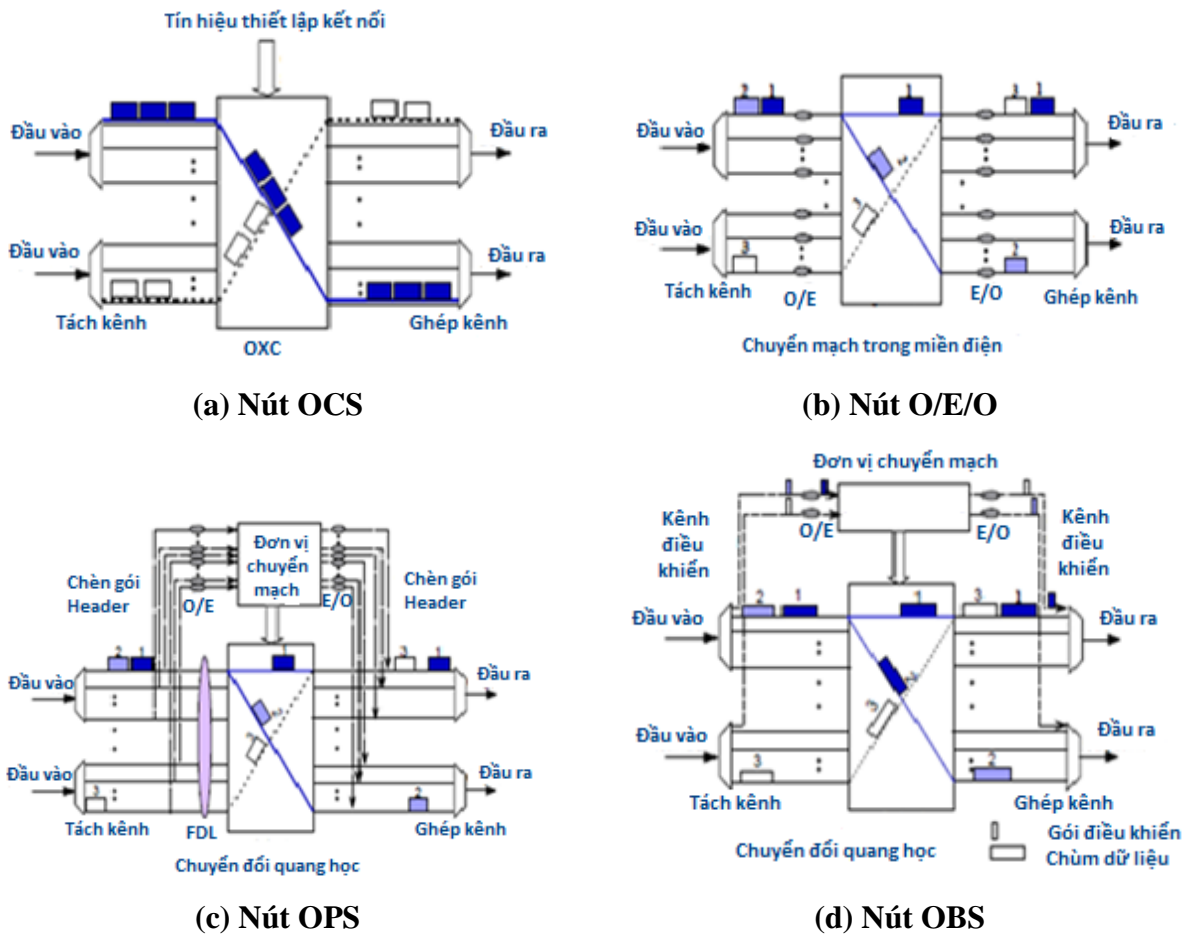
Tuy nhiên, vấn đề đặt ra là việc chuyển đổi dữ liệu giữa hai miền quang và điện đã tạo nên hiện tượng thắt nút cổ chai. Để đạt được tiềm năng về băng thông và các lợi ích của mạng ghép kênh đa bước sóng mang lại thì việc chuyển đổi này phải được tối thiểu hóa. Một vài công nghệ chuyển mạch quang đã được đề xuất nhằm tận dụng lợi thế về khả năng truyền tải tốc độ cao của sợi quang. Những tiếp cận sớm liên quan đến mạng chuyển mạch kênh quang (OCS), trong đó các đường quang (*lightpath*) điểm–điểm thường được thiết lập trong khoảng thời gian tương đối dài. Mạng OCS dùng cách truyền gói tin theo kiểu lưu trữ và chuyển tiếp, trong đó mỗi bộ chuyển tiếp (chuyển mạch) quang thực hiện những hoạt động chuyển đổi tín hiệu quang/điện/quang (O/E/O). Tuy nhiên mạng OCS sớm bộc lộ các hạn chế như không dễ dàng hỗ trợ những biến đổi lưu lượng của mạng hoặc những yêu cầu kết nối thay đổi thường xuyên.

Giao thức IP đã trở thành một giao thức có ảnh hưởng lớn đối với các dịch vụ mạng trên Internet và có mặt ở khắp mọi nơi, nên đã có nhiều nghiên cứu nhằm tích hợp mạng IP với mạng WDM. Mạng chuyển mạch gói (OPS) đã được đề xuất nhằm hỗ trợ việc tích hợp này. Với mạng chuyển mạch gói, mỗi gói tin bao gồm phần điều khiển (*header*) và phần dữ liệu (*data*) được truyền trong mạng quang. Khi gói tin đến, phần đầu gói tin sẽ được chuyển đổi sang tín hiệu điện và được xử lý, còn phần dữ liệu sẽ được đưa vào các bộ đệm quang, chẳng hạn các cuộn dây làm trễ tín hiệu (FDL), cho tới khi phần đầu gói tin được xử lý hoàn tất. Mạng OPS được dự báo là rất thành công trong việc giải quyết những thiếu sót, hạn chế và kém hiệu quả của mạng OCS. Tuy nhiên, những hạn chế về công nghệ để sản xuất các chuyển mạch tốc

độ cao (nano giây) và chi phí quá cao cho những bộ đệm quang hiện nay đã ngăn cản việc triển khai công nghệ này vào thực tế.

Mạng chuyển mạch chùm quang (OBS) đã thu hút các nhà nghiên cứu nhờ khả năng phân phối băng thông động và theo yêu cầu hơn mạng OCS. Nó đem lại nhiều cải tiến mặt kinh tế và cho phép tích hợp việc điều khiển và quản lý. So sánh với OPS, việc lắp đặt hệ thống mạng OBS sẽ thực tế hơn và nó kết hợp được những ưu điểm của cả hai mạng OPS và OCS.

Kiến trúc cơ bản của một nút chuyển mạch với các mô hình chuyển mạch khác nhau được thể hiện trong Hình 1.1.



**Hình 1.1** So sánh sự khác biệt giữa các loại chuyển mạch quang tại nút lõi OBS [65]

Với chuyển mạch kênh quang (OCS), mô hình chuyển mạch trong Hình 1.1a bao gồm một đường quang khi đã được thiết lập thì tất cả dữ liệu vào trên bước sóng  $\lambda$  sẽ đi ra trên cùng bước sóng. Trung bình, thời gian kết nối của OCS trong vòng 1 phút hoặc lâu hơn và thời gian ngắt kết nối là thường vài trăm mili giây (ms). Các kết

nói với thời gian ngắn hơn thường sẽ làm rời rạc việc truyền dữ liệu dẫn đến chi phí điều khiển tăng cao. Hơn nữa, băng thông được phân bổ trên bước sóng  $\lambda$  tại mọi thời điểm là không cần thiết. Trong thực tế, hầu hết các ứng dụng hiện nay chỉ cần kết nối đến bước sóng  $\lambda$  tại một vài thời điểm. Khi môi trường mạng có sự biến động về mật độ dữ liệu đến thì các chùm được truyền đi có thể chỉ trong vài giây hoặc ít hơn.

Để khắc phục các hạn chế của OCS, mô hình chuyển mạch O/E/O đã được giới thiệu như Hình 1.1b. Ở đây phương pháp ghép kênh được thực hiện trong môi trường điện tử và có sử dụng bộ đệm. Vì mỗi đơn vị dữ liệu đều cần phải chuyển đổi O/E/O, phương pháp này là không đủ khả năng mở rộng để hỗ trợ hàng trăm bước sóng với tốc độ 40Gbps hoặc lớn hơn mà có thể đạt được trong tương lai [36].

Mạng chuyển mạch gói quang OPS (Hình 1.1c) chỉ thực hiện chuyển đổi O/E/O đối với phần điều khiển. Một khác biệt quan trọng trong chuyển đổi O/E/O là ở cách tiếp cận, trong đó phần điều khiển có khả năng gửi chậm hơn so với phần dữ liệu của nó và chỉ phần điều khiển được xử lý nên giảm được yêu cầu về tốc độ đối với chuyển đổi O/E/O nhưng vẫn truyền dữ liệu ở tốc độ cao. Tuy nhiên, mạng OPS khó thực hiện được trong tương lai gần vì yêu cầu sử dụng một số lượng lớn các thiết bị chuyển mạch O/E/O cho mỗi bước sóng.

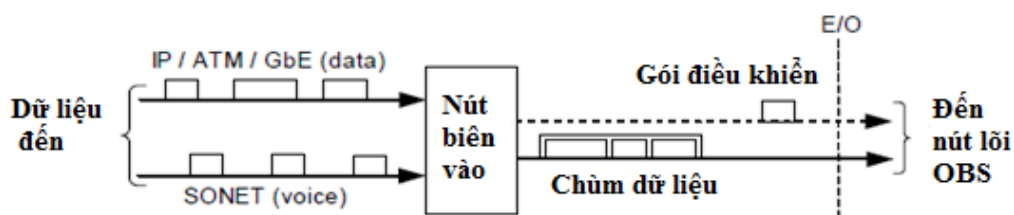
Trong mô hình chuyển mạch OBS, chỉ có một vài kênh điều khiển phải chuyển đổi O/E/O như trong Hình 1.1d. Do phần dữ liệu được truyền trong môi trường toàn quang, nên OBS tách biệt giữa kênh truyền chùm dữ liệu và phần gói điều khiển. Mạng OBS do đó giảm được chi phí chuyển đổi O/E/O mà vẫn đảm bảo xử lý được dữ liệu ở tốc độ cao, tận dụng được những tiến bộ của công nghệ trong tương lai và giải pháp hứa hẹn cho Internet toàn quang thế hệ tiếp theo.

Ngoài ra, để giảm thiểu việc mất mát dữ liệu, các đường trễ quang và chuyển đổi bước sóng [9] có thể được sử dụng tại các nút trung gian trong mạng OBS. Đây đang là một thách thức cho các nhà khoa học trong việc chế tạo các thiết bị phù hợp với tốc độ lên tới nano giây, nhưng có thể thực hiện được trong tương lai [18], [67].

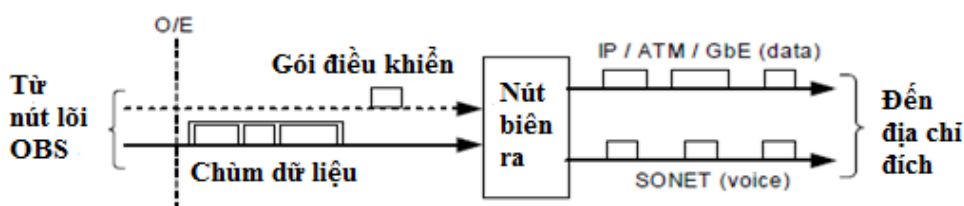
## **1.2 Nguyên tắc hoạt động của mạng OBS**

Trong mạng OBS, các loại dữ liệu đến khác nhau được tập hợp tại nút biên vào

và được truyền dưới dạng các chùm (*burst*) (Hình 1.2a). Tại nút biên ra, các chùm sẽ được tách thành các gói dữ liệu ban đầu để chuyển đến đích mong muốn (Hình 1.2b).



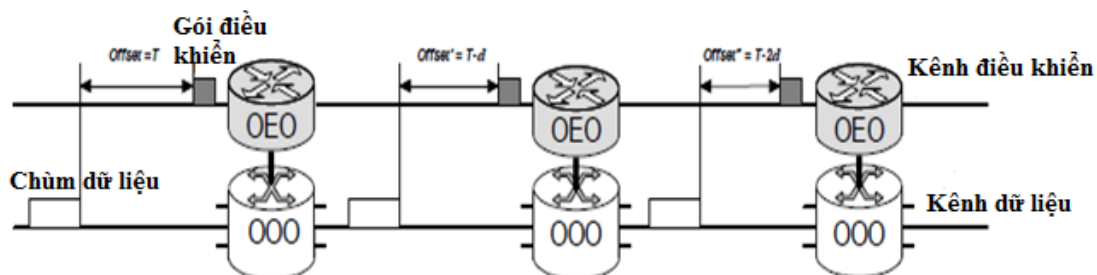
(a) Quá trình tập hợp chùm



(b) Quá trình tách chùm

Hình 1.2 Quá trình tập hợp chùm và tách chùm tại các nút biên OBS [65]

Hình 1.3 mô tả sự tách biệt giữa kênh truyền dữ liệu và kênh truyền gói điều khiển. Mỗi chùm sẽ có một gói điều khiển chứa thông tin về độ dài chùm nhằm đặt trước tài nguyên tại mỗi nút trung gian; chùm dữ liệu theo sau một khoảng thời gian *offset* mà khoảng thời gian này cần tính toán đủ để gói điều khiển kịp đặt trước tài nguyên tại mỗi nút trung gian. Trên mỗi kênh điều khiển, có thể có nhiều gói điều khiển; mỗi gói điều khiển được chuyển đổi O/E/O tại mỗi nút trung gian để cấu hình các chuyển mạch. Thời gian *offset* nhằm để bù đắp cho sự chậm trễ của quá trình cấu hình chuyển mạch tại mỗi nút. Nếu thời gian *offset* đủ lớn thì các chùm dữ liệu sẽ được chuyển tiếp mà không phải chịu bất kỳ một độ trễ nào tại các nút trung gian. Với cách truyền tải này, bộ đệm quang là không cần thiết tại mỗi nút trung gian.

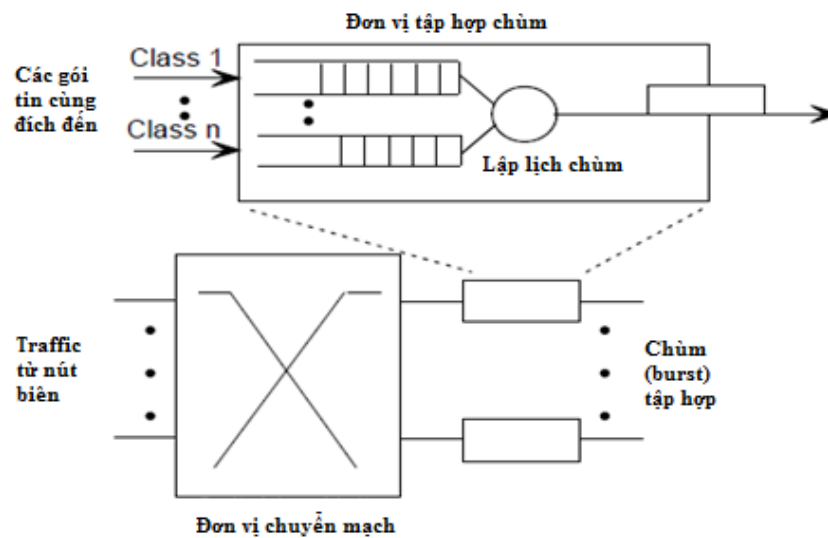


Hình 1.3 Sự tách biệt giữa kênh điều khiển và kênh truyền dữ liệu [65]

## 1.3 Các hoạt động bên trong mạng OBS

### 1.3.1 Tập hợp chùm

Tập hợp chùm là một phương pháp gộp các gói tin từ nhiều mạng truy cập khác nhau (chẳng hạn như các gói IP, ATM...) thành các chùm có kích thước lớn hơn tại nút biên vào của mạng OBS. Kiến trúc nút biên mạng OBS có thể được mô tả như trong Hình 1.4, bao gồm các đơn vị tập hợp chùm để gộp các gói tin thành các chùm. Thông thường người ta sẽ sử dụng một hàng đợi cho mỗi đích đến và mỗi lớp QoS. Một chùm sau khi tập hợp sẽ được lập lịch trên một trong các kênh ra khả dụng bởi gói điều khiển tương ứng của nó, đồng thời thời gian *offset* được điều chỉnh một cách phù hợp trước khi nó được chuyển vào mạng lõi [67].



Hình 1.4 Kiến trúc chung của nút biên vào OBS [67]

Liu và cộng sự trong [21] đã chỉ ra rằng, giải thuật tập hợp có ảnh hưởng đến đặc điểm của luồng chùm sinh ra. Các mục tiếp theo của luận án sẽ trình bày một số giải thuật tập hợp chùm thông dụng và các đặc điểm của chúng.

#### 1.3.1.1 Các giải thuật tập hợp chùm

Có 3 giải thuật tập hợp chùm thông dụng: tập hợp chùm dựa trên bộ đếm (ngưỡng) thời gian, tập hợp chùm dựa trên ngưỡng độ dài chùm và tập hợp chùm lai, là sự kết hợp của ngưỡng thời gian và ngưỡng độ dài chùm [2], [64].

Trong giải thuật tập hợp chùm dựa trên ngưỡng thời gian, một bộ đếm thời gian được kích hoạt khi có gói tin đầu tiên đến hàng đợi rỗng và sau từng khoảng thời gian

có định  $T$  (ngưỡng thời gian), các gói tin trong hàng đợi được gộp thành một chùm. Với giải thuật tập hợp chùm dựa trên ngưỡng độ dài, một ngưỡng độ dài (tối thiểu) được thiết lập và chùm được tạo thành khi độ dài hàng đợi vượt quá ngưỡng này.

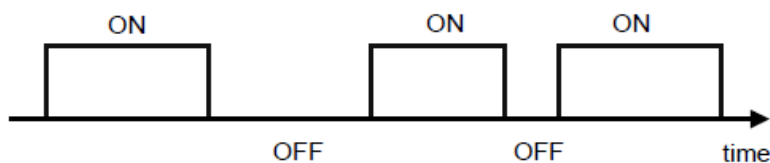
Trong giải thuật tập hợp chùm dựa trên thời gian, ngưỡng thời gian cần được thiết lập một cách thích hợp. Nếu giá trị ngưỡng quá lớn độ trễ của chùm được sinh ra tại nút biên có thể không đáp ứng được yêu cầu độ trễ tối đa của các gói tin được mang bên trong nó. Tuy nhiên, nếu thời gian quá bé sẽ có quá nhiều chùm được sinh ra và kết quả là có quá nhiều gói điều khiển cần phải xử lý. Trong khi giải thuật tập hợp dựa trên ngưỡng thời gian sinh ra các chùm có kích thước thay đổi, thì giải thuật tập hợp dựa trên ngưỡng độ dài lại sinh ra các chùm bằng nhau, nhưng không kiểm soát được độ trễ của chúng (vì thời gian hình thành các chùm là không giống nhau). Để khắc phục những nhược điểm này, giải thuật tập hợp chùm lai (vừa dựa trên bộ đếm thời gian và vừa dựa trên độ dài chùm) đã được đề xuất trong [64], trong đó một chùm được hình thành khi một trong hai ngưỡng này đạt đến.

Các giải thuật tập hợp chùm thích nghi [5] cũng đã được đề xuất nhằm tối ưu hiệu năng của mạng OBS, trong đó một trong hai giá trị ngưỡng thời gian/ngưỡng độ dài hoặc cả 2 được điều chỉnh tự động theo lưu lượng đến của các gói tin. Phương pháp này được chứng minh là cải thiện đáng kể hiệu năng của mạng, tuy nhiên nó làm gia tăng độ phức tạp của giải thuật tập hợp.

Sau khi một chùm được hoàn thành, nó sẽ được đệm tại nút biên trong một khoảng thời gian *offset* trước khi được truyền vào trong mạng lõi. Trong khoảng thời gian *offset* này, các gói tin đến từ các mạng truy cập có thể tiếp tục đến hàng đợi tập hợp, nhưng không được tập hợp vào chùm hiện tại vì thông tin độ dài của chùm này đã được mang trong gói điều khiển và đã được gửi đi. Các gói tin đến trong khoảng thời gian này do đó sẽ có độ trễ trung bình lớn hơn so với các gói tin khác vì nó phải chờ đến lần tập hợp sau. Để giảm độ trễ này, một số phương pháp tập hợp chùm giảm độ trễ đã được đề xuất nhằm cho phép gộp các gói tin đến trong khoảng thời gian *offset* vào chùm hiện tại và độ dài chùm do đó phải được ước tính để mang trong gói điều khiển được gửi sớm [47], [48].

### 1.3.1.2 Đặc điểm của luồng sau tập hợp chùm

Vấn đề ảnh hưởng của các giải thuật tập hợp chùm đến đặc điểm luồng chùm sinh ra bên trong mạng OBS cũng đã được nghiên cứu. Các nghiên cứu tập trung vào hai loại luồng chính: *short range* (luồng *Poisson*) và *long range* (luồng *Self-similar*). Trong hầu hết các nghiên cứu trước đây, các gói tin đến tại hàng đợi từ nhiều nguồn thường được giả định có phân bố *Poisson*. Đối với giải thuật tập hợp chùm dựa trên ngưỡng thời gian hay ngưỡng độ dài, các chùm sinh ra có phân bố *Gauss* [58], nhưng với giải thuật tập hợp theo ngưỡng lai chỉ có phần thời gian giữa các chùm (phần OFF trong Hình 1.5) là có phân bố *Gauss*. Các tác giả trong [22] đã chứng minh rằng luồng *short range* được sinh ra sau tập hợp chùm sẽ có tính *short range* được giảm bớt và hiệu năng được cải thiện hơn.



**Hình 1.5** Đặc điểm luồng chùm được sinh ra sau tập hợp, trong đó ON là khoảng băng thông bị chiếm dụng và OFF là khoảng băng thông nhàn rỗi giữa 2 chùm liên tiếp

Một đặc tính quan trọng trong lưu lượng Internet ngày nay là tính *long range* mà đã được chứng minh làm tăng mất mát dữ liệu và tăng độ trễ trong việc sử dụng tài nguyên. Mặc dù Ge và cộng sự trong [2] đã chỉ ra rằng giải thuật tập hợp chùm làm giảm tính *long range*, nhưng các gói tin khi qua mạng OBS (tại nút biên ra) là không thay đổi tính *long range* của nó.

### 1.3.2 Báo hiệu chùm

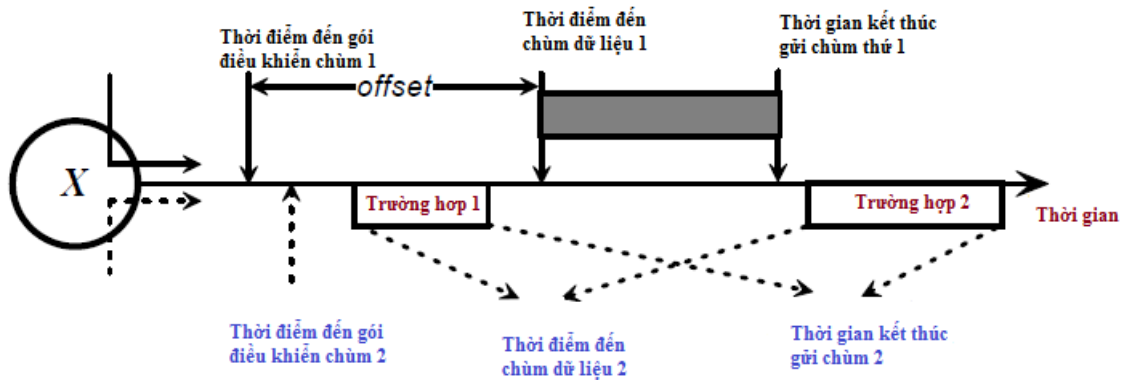
Mặc dù khái niệm chuyển mạch chùm đầu tiên được giới thiệu cho các hệ thống tập trung TDMA [13] và mạng ATM [17] vào năm 1990, nhưng mãi đến năm 1997 các giao thức hỗ trợ cho WDM mới bắt đầu phát triển [21]. Có hai cơ chế truyền tải đã được đề xuất bao gồm: “gọi và chờ” (*tell & wait*) và “gọi và gửi” (*tell & go*) [17]. Khi một chùm cần được truyền tải, hệ thống cố gắng đặt trước tài nguyên bằng cách gửi một thông báo *request*. Mỗi nút trung gian khi nhận được thông báo này sẽ tiến hành đặt trước tài nguyên cho chùm trên một liên kết ở công ra. Nếu băng thông yêu cầu được đáp ứng trên tất cả các chặng (*hop*), một ACK sẽ được gửi phản hồi từ nút

đích nhằm thông báo cho nút nguồn biết để tiến hành gửi chùm. Nếu không, một gói NACK sẽ được gửi ngược lại nhằm giải phóng tài nguyên đã được đặt trước đó và bắt đầu truyền lại thông báo *request* sau một khoảng thời gian. Đối với “gọi và gửi”, các chùm không đặt trước tài nguyên tại nút trung gian nào. Do đó, một chùm cần được làm trễ trước khi đơn vị chuyển mạch tìm được liên kết ra; nếu không, nút trung gian sẽ gửi ngược lại một NACK để bắt đầu quá trình truyền lại chùm. Widjaja và cộng sự trong [17] đã so sánh hiệu năng (như dựa trên thông lượng, độ trễ...) của hai cách tiếp cận nêu trên và kết quả cho thấy rằng tiếp cận “gọi và chờ” có độ trễ thấp hơn và thông lượng tốt hơn so với tiếp cận “gọi và gửi”.

Giao thức JIT được Yoo và cộng sự đề xuất trong [29] có thể được coi là một biến thể của “gọi và chờ”, trong đó mỗi yêu cầu truyền tải chùm được gửi theo một lịch trình tương ứng và sau đó thông tin được thông báo cho mỗi nút để thực hiện truyền tải chùm dữ liệu. Ở đây, thuật ngữ “*Just in time*” có nghĩa rằng một chùm khi đến tại một nút trung gian thì chuyển mạch đã được cấu hình ngay; giao thức này sau đó được mở rộng cho phù hợp với mạng OBS. Tuy nhiên, giao thức JIT không có khả năng xử lý tập trung và khả năng mở rộng tốt; JIT phải gửi một bản sao đến tất cả các nút đối với từng lịch trình. Thêm vào đó việc lập lịch, đồng bộ hóa các liên kết đã làm cho quá trình thực thi gặp nhiều khó khăn. Để khắc phục các hạn chế này Yoo và cộng sự trong [29] đã đề xuất một giao thức tốt hơn có tên là JET.

JET là giao thức báo hiệu được sử dụng cho hầu hết mạng OBS ngày nay, vì không cần bộ đệm quang và không cần chờ xử lý tại các nút trung gian. Để làm được điều này JET cho phép mỗi gói điều khiển mang thông tin độ dài chùm trong khoảng thời gian *offset* để đặt trước tài nguyên cho chùm. Như được chỉ ra trong Hình 1.6, bảng thông được dành riêng cho chùm đầu tiên từ thời điểm chùm đến, thay vì thời điểm gói điều khiển đến. Tại mỗi nút trung gian, thời gian *offset* được cập nhật (giảm) tương ứng với thời gian cấu hình/chuyển mạch như được mô tả trong Hình 1.3. Lưu ý rằng, độ trễ của các gói điều khiển có thể khác nhau vì nhiều lý do khác nhau. Ngoài ra, khi xem xét vấn đề định tuyến trong mạng OBS, thời gian *offset* có thể không đủ nếu hành trình thay thế quá dài. Trong trường hợp như vậy một khoảng thời gian *offset* bổ sung (*extra offset time*) thường được thêm vào [8].





**Hình 1.6 Nguyên tắc hoạt động của giao thức JET [65]**

Một tính năng quan trọng của giao thức JET là thông tin độ dài chùm luôn được mang theo trong gói điều khiển, cho phép đặt trước tài nguyên tại mỗi nút trung gian mà nó đi qua. Điều này giúp cho các nút trung gian đánh giá liệu có đủ tài nguyên để đặt trước cho chùm hay không. Một ví dụ được thể hiện trong Hình 1.6 với đặt trước tài nguyên cho hai trường hợp 1 và 2: Trường hợp 1 không thể thành công với giao thức JIT, nhưng có thể thành công với giao thức JET vì thời điểm kết thúc gửi chùm 2 trước thời điểm đến của chùm 1; Trường hợp 2 đều có thể thành công với giao thức JIT và JET vì thời điểm đến của chùm 2 sau thời điểm kết thúc của chùm 1.

### 1.3.3 Lập lịch chùm

Trong mạng chuyển mạch chùm quang, một vấn đề quan trọng khác là lập lịch chùm. Khi một gói điều khiển đến tại một nút, một thuật toán lập lịch được gọi để lập lịch cho chùm đến tương ứng trên liên kết ra. Dựa vào thông tin chứa trong gói điều khiển, bộ lập lịch biết được thời điểm đến của chùm và khoảng thời gian chùm cần được lập lịch. Mục đích chính của việc lập lịch này là tối thiểu các “khoảng hở” (*gap*) trong mỗi lịch biểu của kênh, trong đó một khoảng hở là khoảng thời gian (băng thông) nhàn rỗi giữa thời điểm kết thúc chùm đã được lập lịch và thời điểm đến của chùm đến trên cùng bước sóng ra. Lập lịch trong mạng OBS khác với lập lịch trong mạng IP truyền thống. Trong mạng IP, nút trung gian sẽ đệm các gói tin đến trong bộ đệm điện tử và sẽ lập lịch chúng lên kênh ra khi có băng thông rỗi. Trong mạng OBS, một chùm đến tại một nút phải được gửi (chuyển tiếp) ngay đến nút tiếp theo vì không có bộ đệm chùm; nếu không, chùm sẽ bị đánh rơi.

Lập lịch trong mạng OBS có thể lấp đầy hoặc không lấp đầy khoảng trống. Các

thuật toán khác nhau chủ yếu ở kiểu và số lượng thông tin trạng thái được duy trì tại một nút ở mỗi kênh. Trong lập lịch không lấp đầy khoảng trống, thuật toán lập lịch cần duy trì thời điểm chưa lập lịch khả dụng gần nhất  $LAUT_i$  (*Latest Available Unscheduled Time*) trên mỗi kênh dữ liệu  $Data_i$  ( $i = 0, 1, \dots, W$ ). Đối với các thuật toán lấp đầy khoảng trống, thời điểm bắt đầu và kết thúc mỗi chùm đã được lập lịch trên mọi kênh dữ liệu phải được duy trì. Khoảng trống (*voids*) là khoảng thời gian (băng thông nhàn rỗi) giữa hai chùm đã lập lịch trên một kênh dữ liệu.

#### **1.3.4 Xử lý tranh chấp chùm**

Với việc sử dụng giao thức JET, các nút biên có thể gửi chùm mà không cần phải báo nhận ACK. Tuy nhiên, điều này đòi hỏi các nút lõi mạng OBS phải giải quyết tốt vấn đề tranh chấp giữa các chùm. Tranh chấp chỉ xảy ra khi các chùm có sự xung đột tài nguyên (bước sóng) tại một cổng ra. Để giải quyết vấn đề này người ta thường sử dụng phương pháp “lệch hướng”, có thể là theo bước sóng, theo không gian và theo thời gian.

- Theo bước sóng: một chùm tranh chấp có thể được gửi trên một bước sóng khác thông qua việc chuyển đổi bước sóng.
- Theo không gian: một chùm tranh chấp có thể được gửi đến một cổng ra khác, sau đó định tuyến trên một tuyến đường khác để đến đích [7].
- Theo thời gian: bằng cách đi qua một đường trễ quang FDL, chùm tranh chấp có thể trì hoãn việc ra cổng ra sau một khoảng thời gian cố định [8], [29].

Nếu chùm tranh chấp không thể lệch hướng, do thiếu bước sóng cung cấp, cổng ra, hay đường trễ quang FDL, việc mất mát dữ liệu là không thể tránh khỏi. Trường hợp này được gọi là phương pháp đánh rơi chùm chủ động (với những chùm không ưu tiên). Ngoài ra, một chùm có thể chiếm quyền ưu tiên của một chùm khác dựa trên mức độ ưu tiên hoặc đặc điểm luồng đến. Vokkarane và cộng sự trong [57], [58] đã đề xuất giải pháp phân đoạn chùm và áp dụng các phương pháp xử lý cho từng đoạn, thay vì toàn bộ chùm mỗi khi xử lý tranh chấp xảy ra. Bảng 1.1 tóm lược so sánh một số phương pháp xử lý tranh chấp. Chú ý rằng, các giải pháp xử lý tranh chấp có thể được thực hiện cùng nhau.

**Bảng 1.1 So sánh các giải pháp xử lý tranh chấp trong mạng OBS**

<b>Phương pháp xử lý tranh chấp</b>	<b>Thuận lợi</b>	<b>Khó khăn</b>
Chuyển đổi bước sóng	Tỉ lệ mất chùm thấp	Công nghệ chưa trưởng thành và tốn kém
Sử dụng đường trễ (FDL)	Tương đối đơn giản và các công nghệ hỗ trợ đã trưởng thành	Tăng độ trễ, tạo ra nhiều khoảng trống và thiết bị khá cồng kềnh
Định tuyến lệch hướng	Không yêu cầu thêm phần cứng	Chi phí phát sinh có thể lớn, nguy cơ mất mát dữ liệu cao
Phân đoạn chùm	Xử lý vấn đề tranh chấp mịn hơn	Việc điều khiển các đoạn chùm khá phức tạp

Các giải pháp xử lý tranh chấp nêu trên được xem là các phương pháp thụ động. Các phương pháp chủ động cũng có thể được sử dụng như bằng cách thống kê tỉ lệ mất chùm trên các bước sóng khác nhau và sắp xếp chúng theo các mức độ ưu tiên khác nhau. Các chùm đến có mức độ ưu tiên cao nhất được đưa vào bước sóng có ưu tiên cao nhất và ngược lại. Tuy nhiên, phương pháp này chỉ có thể thực hiện tại nút biên nếu không có sự hỗ trợ chuyển đổi bước sóng.

## **1.4 Vấn đề công bằng trong mạng OBS**

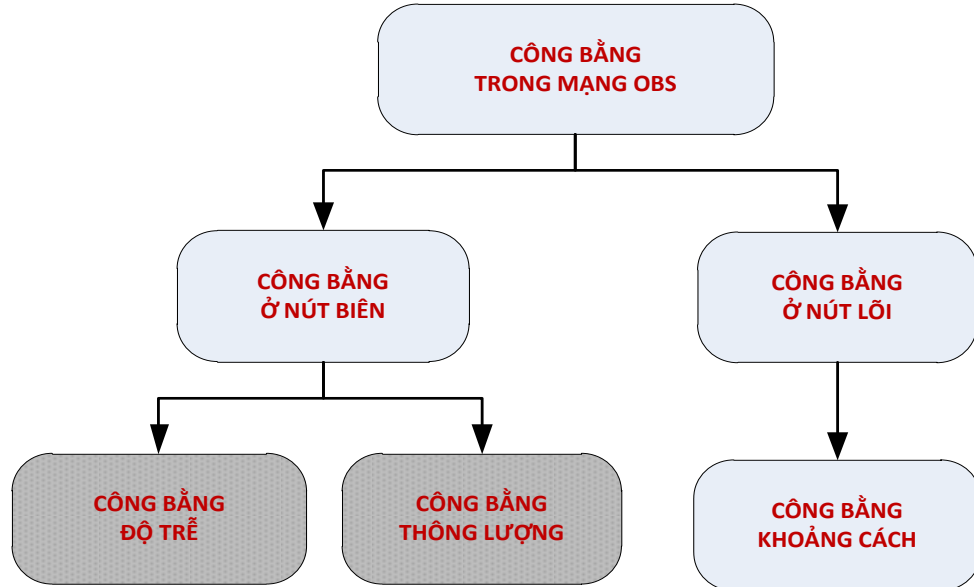
### **1.4.1 Khái niệm và phân loại công bằng trong mạng OBS**

Theo Denda và cộng sự trong [38], công bằng được biết đến là sự hài lòng của các cá nhân khi tham gia vào quá trình phân bổ tài nguyên. Trong mạng OBS, vấn đề công bằng được nghiên cứu tại nút biên và nút lõi (xem Hình 1.7).

Về công bằng tại nút biên, các nghiên cứu tập trung theo hai hướng: (1) công bằng độ trễ (*delay fairness*) [69] và công bằng thông lượng (*throughput fairness*) [53]. Trong công bằng độ trễ, các chùm được phân loại theo lớp QoS của chúng, chùm thuộc lớp QoS cao sẽ có thời gian đệm chùm<sup>1</sup> ngắn hơn so với chùm thuộc lớp QoS thấp; mục tiêu công bằng độ trễ là nhằm đảm bảo cho các chùm thuộc QoS cao được gửi đi sớm hơn nên tăng khả năng đặt trước thành công tài nguyên hơn. Vấn đề

<sup>1</sup> Thời gian đệm chùm (*buffering time*) bao gồm thời gian tập hợp (*assembly time*) và thời gian *offset*.

công bằng thông lượng hay còn được gọi là công bằng tốc độ (*rate fairness*) lại đề cập đến việc cấp phát băng thông công bằng cho mỗi luồng<sup>2</sup> theo tỉ lệ băng thông được cung cấp và băng thông khả dụng của liên kết ra chung.



**Hình 1.7 Phân loại công bằng dựa trên vị trí thực hiện**

Tại nút lõi, vấn đề công bằng được xem xét chủ yếu là công bằng khoảng cách (*distance fairness*). Công bằng khoảng cách đề cập đến việc thực hiện xử lý tranh chấp công bằng, chẳng hạn về mất mát dữ liệu, dựa vào độ dài hành trình (số chặng) từ nút nguồn đến đích. Cụ thể, đó là vấn đề không công bằng về mất mát dữ liệu giữa luồng có hành trình dài so với luồng có hành trình ngắn hơn [10]. Bảng 1.2 mô tả tóm tắt 3 loại công bằng phổ biến trong mạng OBS.

**Bảng 1.2 Các loại công bằng trong mạng OBS**

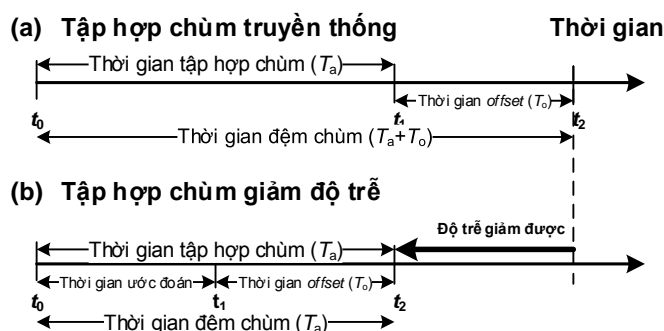
STT	Loại công bằng	Mô tả
1	Công bằng độ trễ ( <i>delay fairness</i> )	Đề cập đến việc thiết lập độ trễ đệm chùng công bằng cho các chùng thuộc các lớp QoS khác nhau.
2	Công bằng thông lượng ( <i>throughput fairness</i> )	Đề cập đến việc phân bổ băng thông công bằng giữa các luồng.
3	Công bằng khoảng cách	Đề cập đến việc xử lý tranh chấp công bằng, như về

<sup>2</sup> Trong luận án này, khái niệm luồng (*flow*) được hiểu dòng dữ liệu di chuyển “bên trong” một kết nối đầu-cuối (*connection*) của một dịch vụ cụ thể và đảm bảo một mức độ chất lượng dịch vụ nào đó.

STT	Loại công bằng	Mô tả
	( <i>distance fairness</i> )	mất mát dữ liệu, dựa vào độ dài hành trình (số nút trung gian) từ nút nguồn đến đích.

### 1.4.2 Công bằng độ trễ

Đã có một số đề xuất nhằm giảm độ trễ đệm chùm tại nút biên mạng OBS [15], [47], [48], [63], [66], [69], trong đó ý tưởng chung là gửi sớm gói điều khiển một khoảng thời gian *offset* trước khi chùm hoàn thành. Như vậy so với mô hình tập hợp chùm truyền thống (Hình 1.8a), các mô hình trong [15], [47], [48], [63], [66], [69] giảm được một khoảng trễ là bằng thời gian *offset* ( $T_o$ ). Tuy nhiên, do thông tin về độ dài chùm được hoàn thành phải được mang trong gói điều khiển, nên các tác giả trong [15], [47], [48], [63], [66], [69] đã đề xuất các phương pháp ước tính độ dài chùm hoàn thành khác nhau. Trong các nghiên cứu này, thời gian ước tính ( $T_a - T_o$ ) đóng một vai trò quan trọng, trong đó khoảng thời gian này càng dài thì độ chính xác ước tính càng cao, nhưng khối lượng tính toán càng nhiều; trong khi khoảng thời gian này càng ngắn thì độ chính xác ước tính càng thấp, nhưng khối lượng tính toán càng ít.



Hình 1.8 So sánh độ trễ đệm chùm giảm được của mô hình tập hợp chùm giảm độ trễ

Trong các mô hình tập hợp chùm giảm độ trễ nêu trên, chỉ có Sui và cộng sự trong [69] đề xuất mô hình tập hợp chùm giảm độ trễ kết hợp với hỗ trợ phân biệt dịch vụ. Cụ thể, các tác giả trong [69] sử dụng thời gian *offset* khác nhau cho các chùm ưu tiên khác nhau và điều chỉnh thời gian tập hợp đối với các chùm ưu tiên khác nhau sao cho chùm ưu tiên cao luôn có thời gian đệm chùm ngắn hơn chùm ưu tiên thấp. Mục tiêu điều chỉnh thời gian tập hợp của [69] không chỉ nhằm làm cho

chùm ưu tiên cao hơn sẽ có độ trễ đầu cuối ngắn hơn, mà còn nhằm bảo đảm độ trễ của các chùm không được vượt quá ràng buộc độ trễ đầu cuối của chúng. Cách tiếp cận này được Sui và cộng sự trong [69] diễn dịch là một giải pháp công bằng độ trễ.

### 1.4.3 Công bằng thông lượng

Công bằng thông lượng hay còn có tên gọi khác là công bằng tốc độ (*rate fairness*) đề cập đến việc phân bổ băng thông công bằng cho các kết nối theo tỷ lệ băng thông được cung cấp và băng thông khả dụng của liên kết chung [53]. Mỗi kết nối (*connection*) đầu cuối mang một luồng dữ liệu thuộc về một dịch vụ nào đó; Các kết nối cùng đầu cuối có thể chia sẻ cùng liên kết (*link/fiber*) hay cùng bước sóng trên một liên kết. Do đó, nếu không có cơ chế cô lập và bảo vệ dịch vụ, các luồng xấu<sup>3</sup> có thể gửi quá nhiều lưu lượng đến các nút lõi, mà kết quả là các luồng tốt<sup>4</sup> phải chịu xác suất mất dữ liệu cao chung. Một giải pháp đơn giản thường được sử dụng là điều khiển lưu lượng ngay từ ngõ vào, trong đó một bộ điều khiển sẽ làm nhiệm vụ hạn chế và loại bỏ lưu lượng vượt quá mức băng thông cấp phát của mỗi kết nối. Tuy nhiên, cách làm này thật sự không hiệu quả vì băng thông không sử dụng hết của một số kết nối sẽ bị lãng phí nếu không được sử dụng cho lưu lượng vượt quá này.

Dựa trên mô hình phân bổ băng thông công bằng max-min trong mạng IP trong [16], Liu và cộng sự trong [67] đã đề xuất một lược đồ cấp phát băng thông công bằng giữa các kết nối sao cho khi tắc nghẽn xảy ra các kết nối tốt sẽ được bảo vệ từ các kết nối xấu trong khi vẫn tận dụng được băng thông khả dụng. Tuy nhiên, băng thông không thể được sử dụng triệt để trong mạng OBS vì luôn tồn tại khoảng trống không thể tránh khỏi hình thành giữa các chùm. Do những khác biệt lớn này, các thuật toán hàng đợi công bằng được phát triển cho các mạng IP không thể áp dụng trực tiếp vào mạng OBS để đạt được cùng sự công bằng max-min. Do đó, Liu và cộng sự trong [67] đã chuyển đổi việc chia sẻ công bằng này thành xác suất mất mát tương ứng với mỗi kết nối, có tên gọi là xác suất mất mát về mặt lý thuyết. Giải thuật được đề xuất trong [67] cố gắng duy trì xác suất mất mát thực tế của mỗi kết nối dao

---

<sup>3</sup> Luồng xấu là luồng có lưu lượng vượt quá mức băng thông được cung cấp.

<sup>4</sup> Luồng tốt là luồng có lưu lượng không vượt mức băng thông được cung cấp.

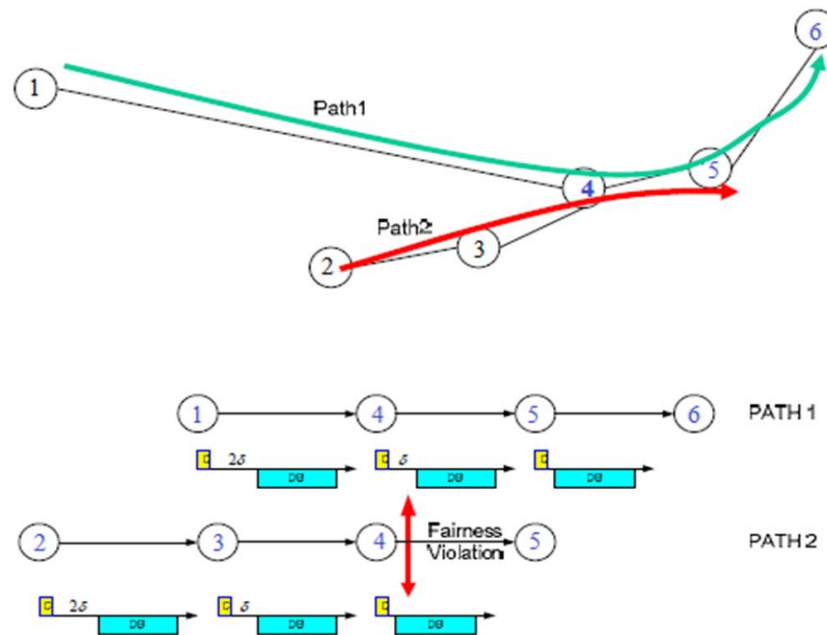
động quanh mức lý thuyết này để đảm bảo công bằng giữa các kết nối.

Tuy nhiên, cách duy trì tỉ lệ mất mát thực tế gần với mức lý thuyết trong [67] có thể dẫn đến hạn chế lưu lượng vào và có thể làm cho việc sử dụng băng thông mạng không hiệu quả. Orawiwattanakul và cộng sự trong [53] đã đề xuất phương pháp RFP nhằm phân bổ băng thông công bằng cho tất cả các luồng dựa trên các tiêu chí công bằng max-min trong [67] và đồng thời xử lý tranh chấp công bằng giữa các chùm. Cụ thể, trong trường hợp có tranh chấp, RFP cho phép các chùm thuộc luồng xấu chiếm kênh từ các chùm thuộc luồng tốt. Các chùm thuộc luồng xấu chỉ được ưu tiên khi tất cả các bước sóng đều bận (tức là tất cả các kết nối đang được sử dụng nhưng vẫn còn băng thông nhàn rỗi) để đảm bảo việc sử dụng băng thông mạng tối ưu. Hơn nữa, chỉ có các nút biên theo dõi tốc độ đến của các chùm và các nút lõi thực hiện hoạt động phân bổ băng thông dựa trên RFP chỉ khi tốc độ đến của lưu lượng vào thay đổi đáng kể. Vì vậy, RFP không gây ra tải nặng trong mạng lõi. Tuy nhiên, phương pháp RFP phải luôn duy trì hai gói điều khiển FBCP (*Forward BCP*) và BBCP (*Back BCP*) để trao đổi thông tin từ nút nguồn và đích, do đó làm tăng đáng kể độ phức tạp của giải thuật và băng thông cho việc trao đổi thông tin.

#### **1.4.4 Công bằng khoảng cách**

Công bằng khoảng cách được biết đến là vấn đề công bằng giữa luồng có hành trình dài so với luồng có hành trình ngắn hơn, trong đó luồng có hành trình dài do phải đi qua nhiều nút trung gian hơn nên có tỉ lệ (xác suất) mất mát dữ liệu cao hơn so với luồng có hành trình ngắn [10]. Hơn nữa, do đa số các mạng OBS hiện nay sử dụng giao thức JET [29] để báo hiệu, giải pháp xử lý tranh chấp tại một nút mạng OBS là dựa trên việc so sánh thời gian *offset* (trong đó chùm có thời gian *offset* càng lớn sẽ được ưu tiên hơn chùm có thời gian *offset* bé) nên một vấn đề không công bằng khoảng cách khác là chùm càng về gần đích thì xác suất bị đánh rơi càng cao do thời gian *offset* của nó càng bé do bị trừ dần [44]. Như được mô tả trong Hình 1.9, Path1 và Path2 có cùng độ dài hành trình (đi qua 4 nút trung gian) thì đáng lẽ phải có cùng mức độ ưu tiên khi có tranh chấp xảy ra, nhưng thực tế không phải vậy. Giả sử tại nút 4 xảy ra tranh chấp tài nguyên, chùm đi trên Path2 sẽ bị đánh rơi (vì thời gian *offset* còn lại của nó nhỏ hơn thời gian *offset* của chùm đi trên Path1), mặc dù nó gần

đến đích hơn chòm đi trên Path1.



**Hình 1.9 Một ví dụ của vấn đề công bằng khoảng cách trong đó chòm càng gần đến đích có xác suất mất mát càng cao [44]**

Đã có một số giải pháp được đề xuất cho vấn đề công bằng khoảng cách như sau:

#### 1.4.4.1 Điều khiển công bằng dựa trên thời gian offset

Liên quan đến điều khiển công bằng dựa trên thời gian *offset*, Tan và cộng sự trong [43] đã đề xuất một phương pháp lựa chọn thời gian *offset* dựa trên trạng thái liên kết LSOS nhằm quản lý thời gian *offset* và chọn thời gian *offset* dựa trên trạng thái liên kết cho các chòm có độ dài hành trình khác nhau sao cho chúng đang xử lý một cách công bằng tại các nút lõi. Cụ thể, trạng thái của một liên kết được ánh xạ thành khả năng lập lịch tương ứng với các nhóm thời gian *offset* khác nhau. Các khả năng lập lịch này được tính toán tại các nút lõi và được thu thập bởi các nút biên một cách định kỳ. Dựa trên thông tin này nút biên xác định được thời gian *offset* phù hợp cho các chòm có hành trình dài ngắn khác nhau. Tuy nhiên, cách tiếp cận này đòi hỏi các nút biên vào phải thường xuyên thu thập thông tin về khả năng lập lịch từ các nút lõi và do đó làm tiêu tốn đáng kể băng thông cho việc trao đổi thông tin.

Để cải tiến LSOS, Rashed và cộng sự trong [3] đã đề xuất ba giải thuật là A-LSOS, 3-LSOS và 1-LSOS, trong đó với A-LSOS, nút biên vào sẽ thu thập thông tin



về trạng thái liên kết từ tất cả các nút trung gian, trong khi với 3-LSOS, nút biên vào chỉ thu thập từ 3 nút trung gian kế tiếp nó và với 1-LSOS nút biên vào chỉ thu thập từ nút trung gian kế tiếp nó. Cách làm của 3-LSOS và 1-LSOS là nhằm giảm tải của các thông tin về khả năng lập lịch được gửi về nút biên vào nhưng đồng thời cũng làm giảm độ chính xác của thời gian *offset* xác định được.

Cũng liên quan đến điều khiển công bằng dựa trên thời gian *offset*, Hailong và cộng sự trong [25] đã đề xuất một phương pháp giám sát xác suất đánh rơi theo nhóm MGDP, trong đó các chùm có hành trình ngắn có xu hướng bị đánh rơi nhiều hơn để dành nhiều tài nguyên cho các chùm có hành trình dài. Cụ thể, MGDP phân loại các chùm vào các nhóm tương ứng với thời gian *offset* hiện thời của nó, trong đó mỗi nhóm sẽ được gán một ngưỡng (xác suất) đánh rơi nhằm để xác định có cần thiết đánh rơi một chùm đến hay không. MGDP giám sát hiệu năng tắc nghẽn của mỗi nhóm và tính toán lại ngưỡng đánh rơi này. Tuy nhiên, MGDP chỉ đạt được sự công bằng đối với từng nút đơn. Xét trên toàn mạng, MGDP không hiệu quả về mặt tắc nghẽn, công bằng và sử dụng băng thông.

#### **1.4.4.2 Điều khiển công bằng trong phân bổ tài nguyên**

Để tích hợp việc điều khiển phân bổ công bằng tài nguyên vào trong giao thức báo hiệu, Zhou và cộng sự trong [7] đã đề xuất một giao thức mới có tên gọi là BJIT (*Balanced JIT*) cải tiến từ giao thức JIT. Trong BJIT, một chùm được truyền từ nút hiện tại đến nút kế tiếp sẽ căn cứ vào số lượng bước sóng rồi cung cấp cho chùm đó. Theo đó, BJIT ưu tiên cho luồng có hành trình dài sử dụng nhiều tài nguyên bước sóng hơn so với chùm có hành trình ngắn. Kết quả là xác suất rơi của chùm có hành trình dài giảm hơn so với chùm có hành trình ngắn. Tuy nhiên, BJIT chỉ mới cải thiện đáng kể về vấn đề công bằng luồng mà chưa xem xét tối ưu thông lượng mạng. Do đó, Zhou và cộng sự trong [8] tiếp tục cải tiến BJIT bằng cách đưa ra thêm một lược đồ điều khiển công bằng dựa trên quyền ưu tiên (*preemption fairness scheme*) với 2 chức năng chính: tính toán kích thước chùm và tính toán số nút trung gian. Do đó, khi có tranh chấp tài nguyên xảy ra, luồng có hành trình dài vẫn được ưu tiên hơn, nhưng khi hai chùm có cùng hành trình thì yếu tố kích thước của chùm sẽ được tính đến, trong đó chùm có kích thước lớn hơn sẽ được ưu tiên hơn.

#### 1.4.4.3 Điều khiển công bằng trong định tuyến

Một cách tiếp cận về điều khiển công bằng trong định tuyến được công bố trong [61], [62], trong đó Gao và cộng sự đã đưa ra 3 phương pháp: Hop-FCR, Hop-N-FCR và Hop-LC. Khác với định tuyến dựa trên thông tin phản hồi như trong [43], Hop-FCR xem xét tình trạng hiện thời về bước sóng rồi trên các liên kết ra ứng viên và chọn liên kết ít bị tắc nghẽn nhất. Với Hop-LC, phương pháp này xem xét số đường đi có thể để đến đích của các nút lân cận và chọn nút có số đường đi đến đích cao nhất. Một kết hợp của Hop-FCR và Hop-LC được tích hợp trong Hop-N-FCR, trong đó thông tin cục bộ và thông tin lân cận được xem xét đồng thời khi thực hiện chọn đường đi. Việc tích hợp giải pháp công bằng luồng trong định tuyến rõ ràng sẽ nâng cao đáng kể tính công bằng giữa các luồng, nhưng cách làm này làm tăng đáng kể chi phí tính toán và độ trễ đầu cuối vì thời gian tính toán tăng lên tại mỗi nút.

#### 1.4.4.4 Điều khiển công bằng trong lập lịch

Kết hợp điều khiển công bằng trong lập lịch, Lee và cộng sự trong [46] đã đề xuất một giải thuật lập lịch kết hợp tại một nút mạng OBS đóng vai trò vừa nút biên vào và nút lõi. Có 3 loại chùm đến được phân biệt tại một nút mạng OBS: (1) chùm OHG (*one hop going*) là chùm có độ dài hành trình chỉ một chặng và xuất phát từ nút hiện thời, (2) chùm SHG (*several hop going*) là chùm có độ dài hành trình lớn hơn một chặng và xuất phát từ nút hiện thời, và (3) chùm TDB (*transit data burst*) là chùm quá cảnh ở nút hiện thời. Khi có tranh chấp giữa chùm SHG và TDB, TDB luôn được ưu tiên hơn vì nó đã đi qua nhiều nút trung gian, tiêu thụ nhiều tài nguyên nên cần được ưu tiên để tăng hiệu năng mạng. Tuy nhiên, cũng để tránh trường hợp không công bằng đối với chùm SHG, một số bước sóng sẽ được dành riêng cho loại chùm này, nhưng chỉ đối với chặng đầu tiên. Khi vượt qua thành công chặng đầu tiên, chùm SHG sẽ trở thành TDB ở nút tiếp theo. Đối với OHG, việc tìm kiếm khoảng trống để lập lịch cho nó giữa các TDB là được ưu tiên. Cách làm này giúp cho việc sử dụng băng thông trở nên hiệu quả hơn. Tuy nhiên, các tác giả trong [46] chưa chứng minh được số lượng hiệu quả số bước sóng dành riêng cho chùm SHG, tỉ lệ mất dữ liệu công bằng giữa các loại chùm và giải pháp cho chùm OHG khi khó tìm thấy khoảng trống phù hợp.

Cũng kết hợp điều khiển công bằng trong lập lịch, Orawiwattanakul và cộng sự trong [49] đã đề xuất giải thuật HBP, trong đó 2 yếu tố chính được xem xét gồm tổng số chặng mà chòm đã đi qua và tổng số chặng còn lại mà chòm phải đi tiếp để đến đích. Khi có sự tranh chấp tài nguyên xảy ra, chòm có hành trình dài hơn và đã đi qua nhiều chặng hơn sẽ được ưu tiên lập lịch. Trong [50], Orawiwattanakul và cộng sự đã công bố một cách tiếp cận tương tự HBP nhưng dựa trên việc tiêu thụ tài nguyên, có tên gọi là giải thuật RCBP. Trong cách tiếp cận này, mỗi liên kết được gán một chi phí tài nguyên, tỉ lệ thuận với tỉ lệ mất dữ liệu của liên kết. Chi phí tài nguyên của mạng có thể được xác định bởi là số bước sóng khả dụng, lưu lượng đến trung bình, lưu lượng thời gian thực.... Do đó, khi có tranh chấp xảy ra, chòm đã tiêu thụ nhiều tài nguyên mạng và sẽ tiêu thụ thêm ít tài nguyên hơn để đi đến đích sẽ được ưu tiên hơn.

Không chỉ dựa trên độ dài hành trình đã đi qua và độ dài hành trình phải đi tiếp như trong [50], Hsu và cộng sự trong [10] còn xem xét thêm cả các yếu tố như thời gian *offset* và độ dài chòm trung bình trong giải thuật lập lịch của mình, có tên gọi là FPP. Cụ thể, các tác giả này đã đề xuất một hàm ưu tiên (trong đó công bằng và giảm thiểu mất mát dữ liệu là được tính đến) để xác định quyền ưu tiên của các chòm khi có tranh chấp xảy ra. Kết quả mô phỏng cho thấy rằng, FPP là hiệu quả hơn MGDSP và BJIT về cả công bằng và mất mát dữ liệu.

#### ***1.4.5 Kết hợp công bằng thông lượng và công bằng khoảng cách***

Việc điều khiển công bằng có xem xét đồng thời cả về thông lượng và về khoảng cách để có thể mang lại một hiệu quả tốt hơn về hiệu năng toàn mạng (như giảm mất mát dữ liệu, tăng thông lượng truyền thông hay giảm độ trễ đầu cuối ...), nhưng vẫn đảm bảo tính công bằng giữa các luồng dữ liệu. Cụ thể, Orawiwattanakul và cộng sự trong [54] đã đưa ra một giải pháp cấp phát băng thông công bằng kết hợp với việc xem xét công bằng về khoảng cách, có tên gọi là RDFP. Chi tiết về sự kết hợp này như sau:

1. Băng thông đầu tiên được phân bổ theo công bằng max-min cho các luồng như trong [53] và lược đồ RDFP sẽ bảo vệ các kết nối tốt từ các kết nối xấu.
2. RDFP chỉ cung cấp công bằng về khoảng cách đối với các kết nối có lưu

lượng nhỏ hơn so với băng thông tối đa đã được cấp phát (kết nối tốt).

3. Nếu một kết nối gửi lưu lượng lớn hơn so với băng thông đã được cấp phát cho nó (kết nối xấu), lưu lượng vượt quá sẽ có một xác suất đánh rơi cao hơn và công bằng về khoảng cách sẽ không được cung cấp cho kết nối này.

#### **1.4.6 Đánh giá các giải pháp công bằng tại nút biên mạng OBS**

Một thống kê và phân loại các công bố về giải pháp công bằng trong mạng OBS trong hơn 10 năm qua được mô tả trong Bảng 1.3.

**Bảng 1.3 Phân loại các giải pháp đã được đề xuất về công bằng trong mạng OBS**

<b>TT</b>	<b>Năm</b>	<b>Tác giả</b>	<b>Hoạt động</b>	<b>Phương pháp đánh giá</b>	<b>Loại công bằng</b>
1	2004	B. Zhou [7]	phân bổ tài nguyên	mô phỏng	khoảng cách
2	2004	L. Hailong [25]	tính thời gian <i>offset</i>	mô phỏng	khoảng cách
3	2004	S.K. Tan [43]	tính thời gian <i>offset</i>	mô phỏng	khoảng cách
4	2005	Y. Liu [67]	cấp phát băng thông	mô phỏng	thông lượng
5	2005	Z. Sui [70]	tập hợp chùm	mô phỏng	độ trễ
6	2006	Z. Sui [69]	tập hợp chùm	mô phỏng	độ trễ
7	2007	B. Zhou [8]	phân bổ tài nguyên	mô phỏng	khoảng cách
8	2007	S.Y. Lee [46]	lập lịch	mô hình markov	khoảng cách
9	2007	Orawiwattanakul [49]	lập lịch	mô phỏng	khoảng cách
10	2008	Orawiwattanakul [50]	lập lịch	mô phỏng	khoảng cách
11	2008	Orawiwattanakul [51]	cấp phát băng thông	mô phỏng	thông lượng
12	2008	C.F. Hsu [10]	lập lịch	mô phỏng	khoảng cách
13	2008	X. Gao [61]	định tuyến	mô hình toán	khoảng cách

TT	Năm	Tác giả	Hoạt động	Phương pháp đánh giá	Loại công bằng
14	2009	X. Gao [62]	định tuyến	mô hình toán	khoảng cách
15	2009	Orawiwattanakul [53]	cấp phát băng thông	mô phỏng + mô hình markov	thông lượng
16	2010	Orawiwattanakul [54]	cấp phát băng thông	mô phỏng	thông lượng + khoảng cách
17	2013	S. Tariq [42]	định tuyến	mô phỏng	khoảng cách
18	2013	A.N.Z. Rashed [3]	tính thời gian <i>offset</i>	mô phỏng	khoảng cách

Tuy nhiên, nếu chỉ xét tại nút biên vào, chỉ có 2 loại công bằng chính được xem xét: (1) công bằng độ trễ và (2) công bằng thông lượng. Cụ thể, Sui và cộng sự trong [69], [70] cho rằng công bằng độ trễ đạt được là khi các chùm có mức ưu tiên cao có độ trễ đệm chùm ngắn hơn. Với công bằng thông lượng, Các tác giả trong [67], [51], [53] đều ánh xạ giải pháp công bằng max-min ở trong mạng IP sang thành vấn đề cấp phát băng thông công bằng ở trong mạng OBS. Các mô hình công bằng đều chưa đưa ra một độ đo cụ thể nào cho vấn đề công bằng và chủ yếu chỉ xét ảnh hưởng của vấn đề công bằng đến các tiêu chí đo hiệu năng khác như: tỉ lệ mất dữ liệu, lỗi ước tính... như được chỉ ra ở Bảng 1.4.

**Bảng 1.4 Đánh giá vấn đề công bằng ở nút biên mạng OBS**

Loại công bằng	Đánh giá công bằng	Ảnh hưởng công bằng đến các đơn vị hiệu năng
Công bằng độ trễ	Các chùm có mức ưu tiên cao sẽ có độ trễ đệm chùm ngắn hơn	Độ trễ, lỗi ước tính
Công bằng thông lượng	Ánh xạ giải pháp công bằng max-min trong mạng IP sang thành cấp phát băng thông công bằng trong mạng OBS	Tỉ lệ mất dữ liệu

Để đánh giá hiệu quả hơn về vấn đề công bằng, luận án đề xuất 2 chỉ số công bằng: *DFI* (được trình bày trong Mục 2.2.2.2) để đánh giá công bằng độ trễ và *TFI*

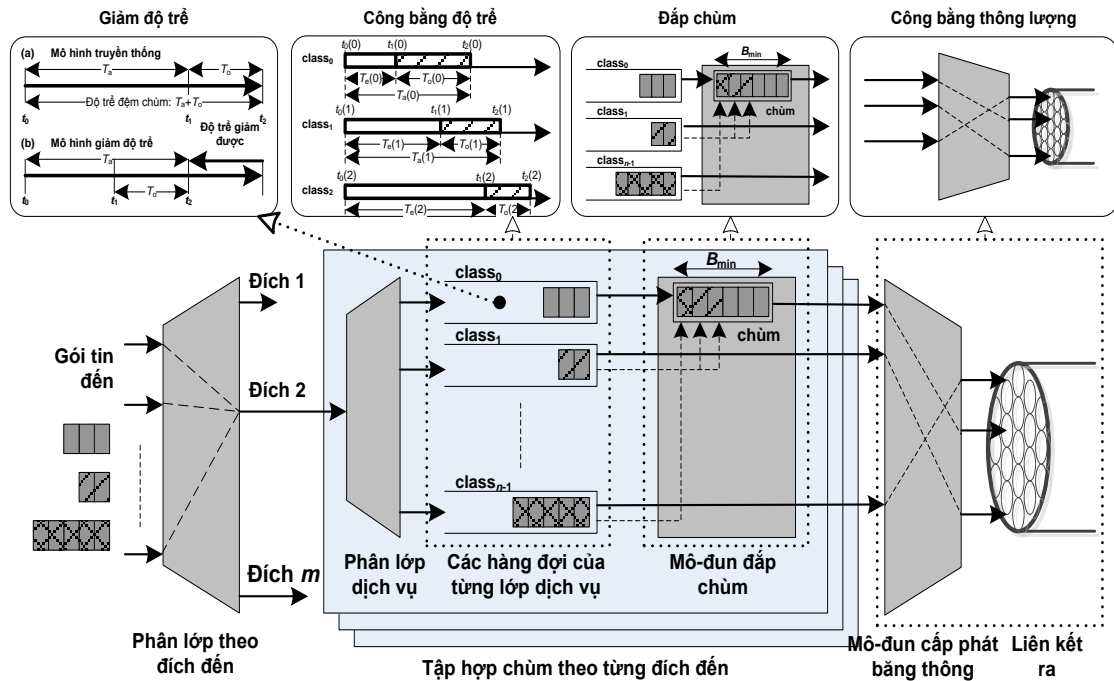
(được trình bày trong Mục 3.1.2.4) để đánh giá vấn đề công bằng thông lượng, cả 2 chỉ số này đều dựa trên chỉ số công bằng của Jain [39]. Cũng như các nghiên cứu khác, luận án cũng xem xét ảnh hưởng của vấn đề công bằng đến các chỉ số (tiêu chí) đo lường hiệu năng mạng khác như: độ trễ, lỗi ước tính, tỉ lệ mất dữ liệu, tỉ lệ sử dụng băng thông... khi so sánh các đề xuất trong luận án với các giải pháp đã công bố trước đây.

## **1.5 Các mục tiêu nghiên cứu của luận án**

Dựa trên phân tích các giải pháp được công bố về công bằng trong mạng OBS và mục tiêu nghiên cứu đã được xác định trong phần Mở đầu, luận án sẽ tập trung vào điều khiển công bằng luồng tại nút biên nhằm nâng cao hiệu quả công bằng độ trễ và công bằng thông lượng. Bốn mục tiêu chính được luận án xem xét gồm:

- *Mục tiêu 1:* cải tiến và đề xuất giải pháp tập hợp chùm giảm độ trễ đối với từng hàng đợi riêng biệt.
- *Mục tiêu 2:* cải tiến và đề xuất giải pháp tập hợp chùm công bằng độ trễ khi xem xét nhiều hàng đợi tương ứng với các lớp QoS khác nhau.
- *Mục tiêu 3:* cải tiến và đề xuất giải pháp điều khiển công bằng thông lượng áp dụng cho các loại luồng đến Poisson và non-Poisson.
- *Mục tiêu 4:* đề xuất giải pháp đắp chùm sau tập hợp chùm nhằm nâng cao hiệu năng truyền thông.

Các đề xuất cho mục tiêu 1, 2 sẽ được trình bày trong Chương 2 và các đề xuất cho mục tiêu 3, 4 sẽ được trình bày trong Chương 3 của luận án. Để triển khai được các phương pháp, nút biên mạng OBS cần bổ sung các mô đun chức năng gồm: Mô đun giảm độ trễ, Mô đun công bằng độ trễ, Mô đun đắp chùm và Mô đun công bằng thông lượng (Hình 1.10).



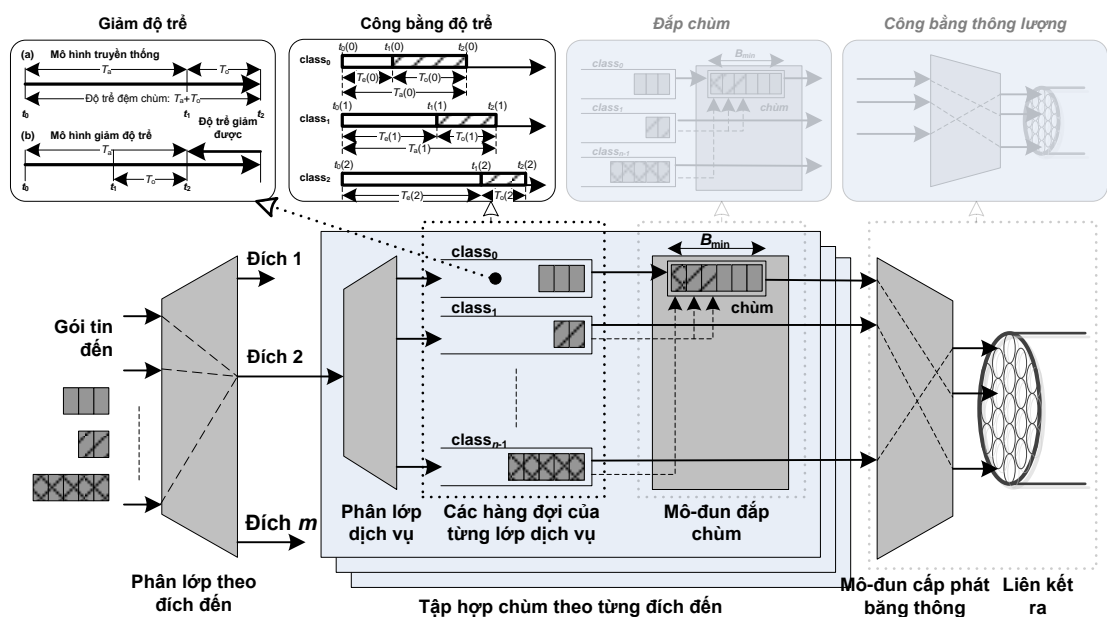
**Hình 1.10** Kiến trúc nút biên vào OBS với các mô-đun chức năng được bổ sung

## 1.6 Tiểu kết Chương 1

Chương đầu tiên của luận án đã giới thiệu tổng quan về mạng OBS và các hoạt động bên trong mạng, trong đó tập hợp chùm tại nút biên mạng được tập trung phân tích vì nó có ảnh hưởng quan trọng đến vấn đề công bằng luồng trong toàn mạng. Luận án cũng đã phân tích và đánh giá các phương pháp đã được công bố cho đến nay về điều khiển công bằng. Đó chính là cơ sở để luận án cuối cùng xác định được bốn mục tiêu cần nghiên cứu, cũng như đề xuất kiến trúc nút biên vào với các mô-đun chức năng được thêm vào nhằm đảm bảo triển khai các phương pháp công bằng luồng, nâng cao hiệu năng truyền thông của mạng OBS.

## CHƯƠNG 2. TẬP HỢP CHÙM GIẢM ĐỘ TRỄ VÀ CÔNG BẰNG ĐỘ TRỄ

Tập hợp chùm là một hoạt động tại nút biên mạng OBS thực hiện tập hợp (gộp) dữ liệu đến từ các mạng truy cập (chẳng hạn các gói IP, ATM...) thành các đơn vị mang dữ liệu lớn hơn, được gọi là chùm dữ liệu (chùm). Chùm là đơn vị truyền thông chính bên trong mạng OBS. Tại mỗi nút biên, các hàng đợi được phân bố tương ứng với các đích đến và các lớp QoS (Hình 2.1). Dữ liệu đến từ các mạng truy cập sẽ được phân loại theo đích đến và lớp QoS của chúng để đưa vào hàng đợi phù hợp. Khi một ngưỡng tập hợp chùm (theo thời gian hoặc độ dài) đạt đến, dữ liệu trong mỗi hàng đợi sẽ được tập hợp thành một chùm mà sau đó sẽ được gửi qua mạng OBS.



**Hình 2.1 Hai mô-đun chức năng điều khiển công bằng được đề xuất: mô-đun giảm độ trễ và mô-đun công bằng độ trễ, trong kiến trúc nút biên vào OBS**

Trong tập hợp chùm, nếu từng hàng đợi được xem xét riêng biệt thì mục tiêu của tập hợp chùm có thể chỉ nhằm làm giảm độ trễ đệm chùm tại mỗi hàng đợi; Nhưng nếu được xem xét trong môi trường có phân biệt QoS thì công bằng độ trễ (một trong những tiêu chí để đánh giá hiệu năng mạng) sẽ là mục tiêu cần hướng đến, trong đó việc giảm độ trễ, phân biệt QoS theo độ trễ và công bằng về độ trễ giữa các lớp QoS sẽ được xem xét đồng thời.



Chương 2 sẽ trình bày các đề xuất đầu tiên của luận án (Hình 2.1) bao gồm:

- Mô hình tập hợp chòm giảm độ trễ cải tiến (iBADR) và mô hình tập hợp chòm giảm độ trễ tối ưu OBADR, khi xét trên từng hàng đợi riêng biệt (không xét đến QoS) và
- Mô hình tập hợp chòm công bằng độ trễ (BADF) đối với các luồng QoS trong các hàng đợi có phân biệt QoS.

## 2.1 Mô hình tập hợp chòm giảm độ trễ

### 2.1.1 Vấn đề độ trễ trong hoạt động tập hợp chòm

Độ trễ đầu cuối của một chòm khi được truyền qua mạng OBS chủ yếu là do bốn thành phần gây nên: (1) độ trễ tập hợp chòm tại nút biên vào, (2) thời gian *offset* để đặt trước tài nguyên của gói điều khiển, (3) độ trễ chuyển tiếp chòm tại các nút lõi và (4) độ trễ truyền bá trong mạng lõi. Hai độ trễ cuối thường phụ thuộc vào đường truyền đã lựa chọn và băng thông khả dụng trên đường truyền này, nên không thể giảm được với một giao thức đã được cài đặt. Chỉ có 2 độ trễ tập hợp và thời gian *offset* là có thể giảm được. Kết hợp của hai độ trễ này có tên gọi chung là độ trễ đệm chòm.

Đã có một số giải pháp được đề xuất nhằm làm giảm độ trễ đầu cuối dựa trên hoạt động tập hợp chòm, trong đó ý tưởng chính là gửi sớm gói điều khiển trước khi chòm được hoàn thành. Cách làm này làm giảm đáng kể độ trễ đệm chòm, nhưng cần phải ước tính độ dài của chòm được hoàn thành bởi vì thông tin này cần được mang trong gói điều khiển. Tuy nhiên, cách tiếp cận này sẽ gây ra lỗi ước tính và có ảnh hưởng đáng kể đến độ trễ tập hợp chòm. Phần tiếp theo sẽ phân tích chi tiết và đánh giá các phương pháp tập hợp chòm giảm độ trễ đã công bố.

### 2.1.2 Các công trình nghiên cứu liên quan

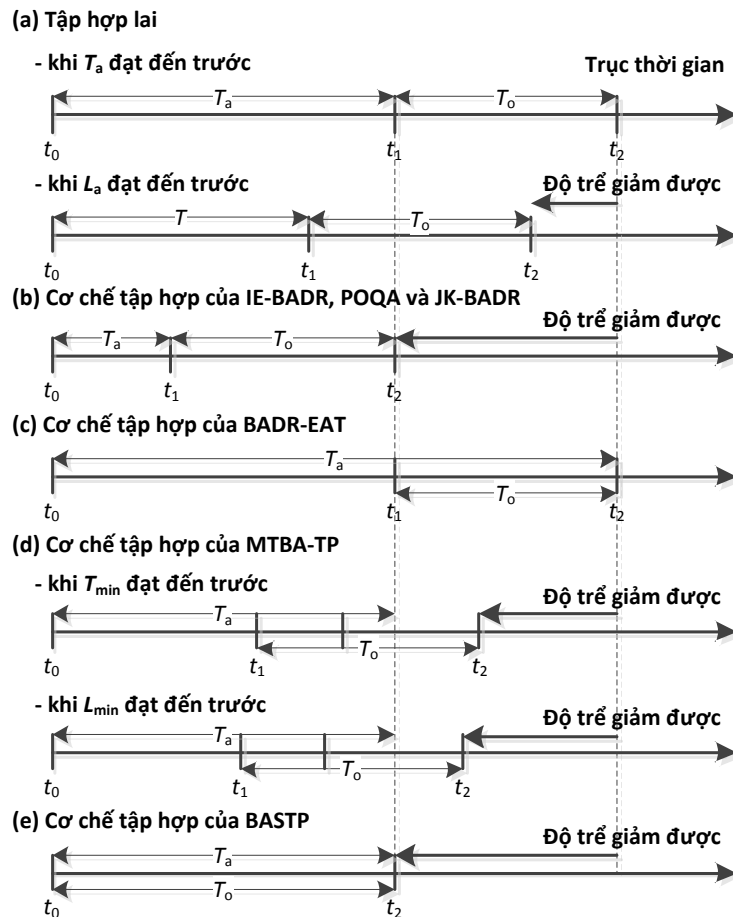
#### 2.1.2.1 Phân tích các phương pháp tập hợp chòm giảm độ trễ đã công bố

Phương pháp tập hợp chòm giảm độ trễ đầu tiên được đề xuất bởi Hashiguchi và cộng sự trong [47], được đặt tên là IE-BADR, trong đó gói điều khiển được gửi đi tại thời điểm  $t_1$  trước khi hoàn thành chòm (xem Hình 2.2b). Bằng cách này, chòm được gửi đi ngay tại thời điểm  $t_2$  mà không phải chờ đợi một khoảng thời gian *offset*

nào như trong phương pháp tập hợp chùm truyền thống. Điều đó có nghĩa là các gói tin được tập hợp trong chùm hiện thời giảm được một độ trễ  $T_o$ . Tuy nhiên, do thông tin về độ dài chùm cần được mang trong gói điều khiển để phục vụ cho việc đặt trước tài nguyên tại các nút trung gian, ước tính độ dài chùm tại thời điểm  $t_1$  là cần thiết. Các tác giả trong [47] đã đề xuất một cách ước tính độ dài chùm dựa trên tốc độ đến của gói tin trong khoảng thời gian ước tính ( $t_1 - t_0$  hay  $T_a - T_o$ ) bởi Công thức 2.1

$$L^e = \delta \times L_w \times \frac{T_a}{T_a - T_o} \quad (2.1)$$

trong đó  $L_w$  là độ dài hàng đợi trong khoảng thời gian ước tính và  $\delta$  là một tham số điều khiển,  $0 \leq \delta \leq 1$ .



**Hình 2.2 So sánh các phương pháp tập hợp chùm giảm độ trễ**

Các phương pháp tập hợp chùm được đề xuất bởi Sui và cộng sự trong [69], [70], được đặt tên là POQA, và bởi Mikoshi và cộng sự trong [48], được đặt tên là JK-BADR, là tương tự với IE-BADR, nhưng khác ở cách ước tính độ dài chùm. Cụ

thể, POQA ước tính độ dài chùm bằng cách sử dụng một bộ lọc tuyến tính AAR (*adaptive auto-regressive*) với các thống kê chiều dài các chùm đo được trong  $M - 1$  lần tập hợp trước đó và độ dài hàng đợi hiện thời trong khoảng thời gian ước tính ( $L_w$ ). Độ dài chùm ước tính do đó được xác định bởi Công thức 2.2.

$$L^e = \sum_{j=1}^{M-1} w(j) \times L_j + q \times L_w \times \frac{T_a}{T_a - T_o} \quad (2.2)$$

trong đó  $L_j$  là độ dài chùm đo được ở lần tập hợp thứ  $j$  ( $1 \leq j \leq M$ ) và  $w(j)$  là trọng số (tham số điều khiển) của nó. Lưu ý rằng  $q = w(M)$  và  $\sum_{i=1}^M w(i) = 1$

JK-BADR ước tính độ dài chùm dựa trên thuật toán Jacobson/Karels [35] với một số thay đổi. Đầu tiên lỗi ước tính  $\varepsilon(j)$  của lần tập hợp thứ  $j$  được tính là khoảng cách giữa độ dài chùm đo được  $L_j$  và độ dài chùm ước tính  $L_j^e$ . Lỗi ước tính này sau đó được sử dụng cho việc tính toán độ dài chùm ước tính của lần tập hợp chùm thứ  $j + 1$ . Một tham số  $\Delta(j + 1)$ , được gọi là độ lệch của lần tập hợp chùm thứ  $j + 1$  cũng được tính toán dựa trên độ lệch  $\Delta(j)$  và lỗi ước tính  $\varepsilon(j)$  của lần tập hợp chùm thứ  $j$ . Đại lượng này cũng tham gia vào việc xác định độ dài ước tính cuối cùng. Các phương trình được sử dụng để ước tính độ dài chùm của JK-BADR là:

$$\begin{aligned} \varepsilon(j) &= L_j - L_j^e \\ L_{j+1}^e &= L_j^e + \eta \times \varepsilon(j) \\ \Delta(j+1) &= \Delta(j) + \eta \times (|\varepsilon(j)| - DL(j)) \\ L_{j+1} &= \beta \times L_{j+1}^e + \varphi \times DL(j+1) \end{aligned} \quad (2.3)$$

trong đó  $\eta$ ,  $\beta$  và  $\varphi$  là các tham số điều khiển.

Trong hướng tiếp cận tập hợp chùm của mình, được đặt tên BADR-EAT, Fukushima và cộng sự trong [66] cho phép tập hợp các gói tin đến trong khoảng thời gian *offset*  $T_o$  vào chùm hiện thời (Hình 2.2c) và đề xuất cách ước tính độ dài chùm dựa trên tốc độ đến trung bình  $\lambda_{avg}$  của  $M$  gói tin đến sau cùng nhất:

$$L^e = L + \lambda_{avg} \times T_o \quad (2.4)$$

trong đó  $L$  là độ dài chùm đo được trong khoảng thời gian  $T_a$ .

Nếu thời gian tập hợp chùm được mở rộng là khoảng thời gian mà các gói tin

đến đều được tập hợp trong chùm hiện thời thì BADR-EAT là tương tự với IE-BADR, POQA và JK-BADR, nhưng với thời gian tập hợp chùm lớn hơn ( $T_a + T_o$ ).

Cách tiếp cận của Liu và cộng sự trong [15], được đặt tên là MTBA-TP, cũng tương tự với JK-BADR, nhưng sử dụng phương pháp tập hợp chùm kết hợp (lai). Cụ thể, gói điều khiển sẽ được gửi đi khi thời gian tập hợp chùm đạt đến một ngưỡng thời gian được định trước  $T_a$  hoặc độ dài chùm đạt đến một ngưỡng độ dài tối thiểu  $L_{min}$ . Các tác giả trong [15] cũng đề xuất một phương pháp ước tính độ dài chùm dựa trên sự tăng/giảm của tốc độ gói tin đến của lần tập hợp chùm hiện thời ( $\lambda_{cur}$ ) so với lần tập hợp chùm trước đó ( $\lambda_{pre}$ ) như Công thức 2.5

$$L^e = L + \left( \lambda_{pre} + (\lambda_{cur} - \lambda_{pre}) \times \frac{T_o}{T_w + T_o} \right) \times T_o \quad (2.5)$$

trong đó của số thời gian  $T_w + T_o$  là khoảng thời gian giữa 2 thời điểm hoàn thành chùm liên tiếp. Nếu bộ đếm thời gian đạt đến ngưỡng thời gian  $T_a$  trước ( $T_w = T_a$ ), MTBA-TP tương đương với mô hình tập hợp chùm dựa trên ngưỡng thời gian truyền thống (Hình 2.2a); nhưng nếu ngưỡng độ dài tối thiểu  $L_{min}$  đạt đến trước (Hình 2.2d), các gói tin được tập hợp trong chùm hiện thời sẽ giảm được một khoảng thời gian là  $t_1 + T_o - T_a$ .

Với đề xuất của Jiang và cộng sự trong [63], được đặt tên là BASTP, gói điều khiển được gửi ngay khi có gói tin đầu tiên đến tại hàng đợi rỗng. Thông tin được mang trong gói điều khiển bao gồm thời gian *offset*  $T_o$ , chính là thời gian tập hợp chùm  $T_a$ , và độ dài ước tính  $L^e$  đã được tính toán trong lần tập hợp chùm sau cùng nhất. BASTP là một phương pháp tập hợp chùm lai, tức là dựa trên cả bộ đếm thời gian và độ dài chùm tối đa. Hai ngưỡng này được điều chỉnh một cách linh động. Cụ thể, ngưỡng thời gian của lần tập hợp chùm hiện thời ( $T_a$ ) được tính toán là trung bình của  $M$  ngưỡng thời gian trước đó ( $T_j, 1 \leq j \leq M$ ) bằng Công thức 2.6.

$$T_a = \frac{\sum_{j=1}^M T_j}{M} \quad (2.6)$$

Tương tự, độ dài chùm ước tính  $L^e$  của lần tập hợp sau được tính toán dựa trên một cặp ngưỡng độ dài chùm tối thiểu và tối đa ( $L_{min}, L_{max}$ ), bước điều chỉnh *step*

( $1 \leq step \leq N$ ) và tổng bước điều chỉnh ( $N$ ):

$$L^e = L_{\min} + step \times \frac{(L_{\max} - L_{\min})}{N} \quad (2.7)$$

Lưu ý rằng, bước điều chỉnh được thực hiện từng bước ( $step = step \pm 1$ ) phụ thuộc vào sự tăng/giảm của lưu lượng đến.

Như vậy, lỗi ước tính trong BASTP có thể được triệt tiêu khi ngưỡng độ dài (tức độ dài ước tính  $L^e$ ) đạt đến trước nhưng chỉ trong trường hợp kích thước các gói tin đến là bằng nhau hoặc độ dài ước tính  $L^e$  là bội số của kích thước các gói tin đến. Trong trường hợp các gói tin đến có kích thước thay đổi hoặc ngưỡng thời gian  $T_a$  đạt đến trước, luôn tồn tại một lỗi ước tính nhất định  $R_E = L - L^e$ .

Một so sánh các phương pháp tập hợp chùm giảm độ trễ đã được công bố được trình bày trong Bảng 2.1.

**Bảng 2.1 So sánh các phương pháp tập hợp chùm giảm độ trễ đã công bố**

	<b>IE-BADR</b>	<b>POQA</b>	<b>JK-BADR</b>	<b>BADR-EAT</b>	<b>MTBA-TP</b>	<b>BASTP</b>
Phương pháp tập hợp chùm	Dựa trên ngưỡng thời gian	Dựa trên ngưỡng thời gian	Dựa trên ngưỡng thời gian	Dựa trên ngưỡng thời gian	Dựa trên ngưỡng lai	Dựa trên ngưỡng lai
Đặc điểm ngưỡng	Cố định	Cố định	Cố định	Cố định	Cố định	Thích nghi
Phương pháp ước tính	Dựa vào tốc độ trung bình các gói tin đến trong khoảng thời gian ước tính	Dựa vào độ dài của M chùm sau cùng nhất	Dựa vào lỗi ước tính của lần tập hợp kế trước	Dựa vào mật độ của M gói tin sau cùng	Dựa vào tốc độ đến của gói tin sau cùng nhất	Dựa vào M lần tập hợp chùm sau cùng nhất
Độ trễ giảm được	$T_o$	$T_o$	$T_o$	$T_o$	$t_1 + T_o - T_a$	$T_o$

### 2.1.2.2 So sánh và đánh giá dựa trên kết quả mô phỏng

Với mục tiêu mô phỏng là:

- So sánh tỉ lệ lỗi ước tính trung bình của các phương pháp đã công bố được tính bởi Công thức 2.8

$$\overline{R_E} = \frac{\sum_{j=1}^M (|L_j - L_j^e| / L_j)}{M} \quad (2.8)$$

trong đó  $M$  là số lần tập hợp chùm,  $L_j$  và  $L_j^e$  là kích thước hoàn thành và kích thước ước tính của chùm thứ  $j$ .

- So sánh số gói tin thừa được chuyển cho chùm tiếp theo trong 100 lần tập hợp chùm liên tiếp giữa các phương pháp tập hợp chùm đã được công bố.
- Phân tích cách chọn ngưỡng của BASTP, là phương pháp tốt nhất trong số các phương pháp tập hợp chùm đã công bố.

Luận án tiến hành cài đặt các phương pháp tập hợp chùm giảm độ trễ trên một máy tính với cấu hình 2.4 GHz Intel Core 2 CPU, 2G RAM. Các gói tin đến tại hàng đợi của nút biên mạng có phân bố Poisson với kích thước thay đổi ngẫu nhiên trong khoảng [500, 1000] bytes. Lưu lượng tải chuẩn hoá<sup>5</sup> đến tại hàng đợi thay đổi từ 0.1 đến 0.9. Dữ liệu được trích xuất từ NS2 [71] với gói hỗ trợ obs-0.9a. Các tham số tập hợp chùm bao gồm:  $T_a = 6$  ms,  $T_o = 2$  ms.

Các mô phỏng trong luận án đa số được thực hiện trong thời gian 1 giây. Thực tế, với lưu lượng khá lớn các gói tin đến (khoảng 50000 gói/giây) tại mỗi hàng đợi tập hợp chùm; nếu trung bình có 20 gói được tập hợp trong một chùm, số lượng chùm sinh ra trong mỗi giây mô phỏng là 2500. Hiệu quả đánh giá của mỗi giải thuật tập hợp chùm được tính là trung bình hiệu quả của 2500 lần tập hợp chùm liên tiếp, nên đảm bảo tính tin cậy và chính xác.

Để đảm bảo độ tin cậy cho việc so sánh, các tham số phải được cài đặt thống nhất đối với tất cả các phương pháp. Trong đề xuất của BASTP, một cặp ngưỡng độ dài chùm ( $L_{\min}$ ,  $L_{\max}$ ) được yêu cầu xác định trước nhằm phục vụ cho việc tính toán

---

<sup>5</sup> Trong luận án này, lưu lượng tải chuẩn hoá được tính bằng tỉ lệ giữa tốc độ đến với khả năng đáp ứng băng thông của liên kết ra.

độ dài ước tính. Tuy nhiên cặp ngưỡng này lại phụ thuộc vào ngưỡng thời gian tập hợp  $T_a$ , với một tốc độ luồng gói tin đến được cho. Do vậy, luận án thực hiện cài đặt các phương pháp IE-BADR, POQA, JK-BADR, BADR-EAT và MTBA-TP với cùng tải chuẩn hóa 0.5 và ngưỡng thời gian tập hợp  $T_a = 6ms$ . Các kết quả về giá trị trung bình của kích thước chùm tối đa và tối thiểu ( $L_{min}$ ,  $L_{max}$ ) của các phương pháp này, được chỉ ra trong Bảng 2.2, sẽ giúp xác định được cặp ngưỡng độ dài chùm phù hợp nhất ( $L_{min}$ ,  $L_{max}$ ) cho BASTP. Cụ thể, cặp ngưỡng độ dài chùm được xác định cho BASTP trong trường hợp này là (130000, 170000), được tính trung bình từ các cặp kích thước chùm tối đa và tối thiểu của các phương pháp khác.

**Bảng 2.2 Trung bình kích thước tối đa và tối thiểu của các chùm sinh ra**

Phương pháp	IE-BADR	POQA	JK-BADR	BADR-EAT	MTBA-TP
<b>Độ dài tối thiểu (byte)</b>	129000	133000	129000	180000	129000
<b>Độ dài tối đa (byte)</b>	172000	173000	176000	230000	169000

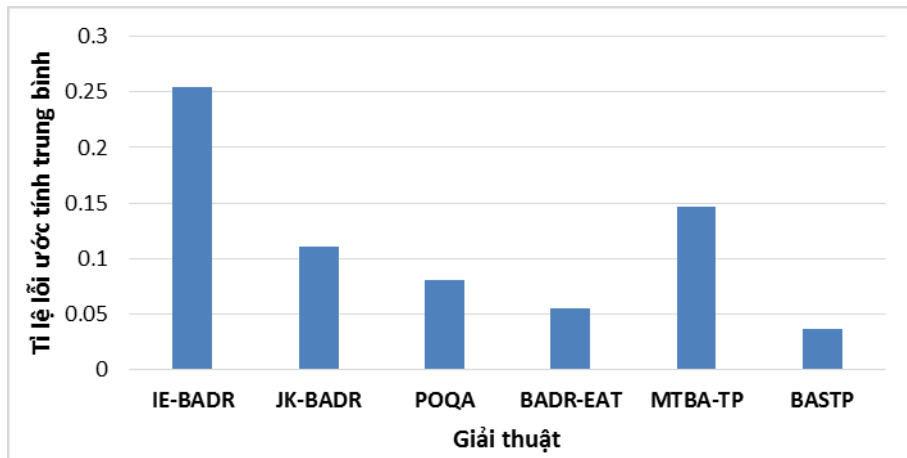
Lưu ý rằng phương pháp BADR-EAT có thời gian tập hợp chùm là  $T_a + T_o$ , thay vì chỉ là  $T_a$ , nên có kích thước chùm tối đa và tối thiểu lớn hơn so với các phương pháp khác; nhưng nếu xét về tốc độ đến (số bytes đến trong một đơn vị thời gian) nó là tương đương đối với tất cả các phương pháp.

#### **a. So sánh tỉ lệ lỗi ước tính trung bình**

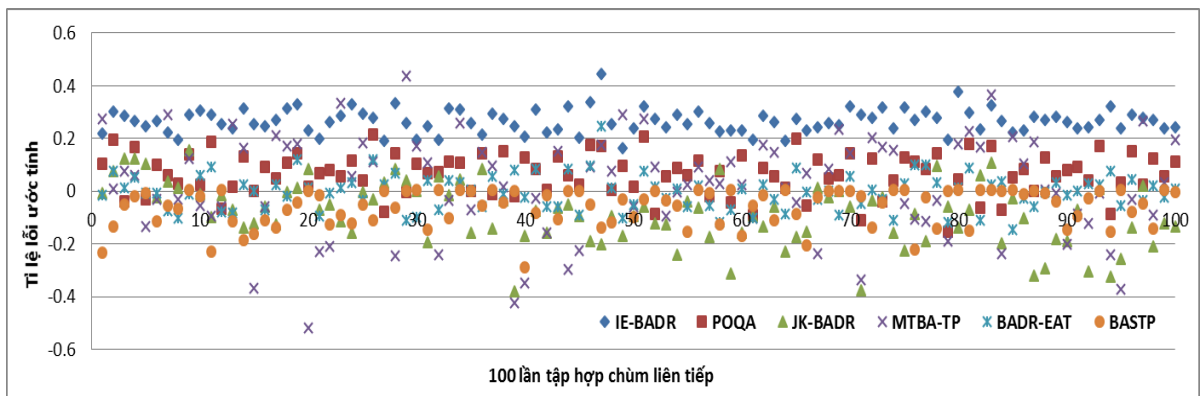
Hình 2.3 mô tả so sánh về tỉ lệ lỗi ước tính trung bình giữa các phương pháp đã công bố, trong đó có thể thấy rằng các phương pháp dựa trên thống kê như BASTP, BADR-EAT và POQA cho lỗi ước tính trung bình thấp hơn so với các phương pháp còn lại. Để tìm hiểu sâu hơn về kết quả này, phân bố lỗi ước tính của 100 lần tập hợp chùm liên tiếp tiếp tục được xem xét.

Như mô tả trong Hình 2.4, lỗi ước tính có phân bố xung quanh giá trị 0. Thực tế, lỗi ước tính xảy ra với một trong hai trường hợp sau: (1) kích thước chùm hoàn thành lớn hơn kích thước chùm ước tính, nên các gói tin thừa sẽ được chuyển sang cho lần tập hợp kế tiếp và (2) kích thước chùm hoàn thành nhỏ hơn kích thước chùm ước tính, lúc này có một sự lãng phí về băng thông do tài nguyên được đặt trước nhiều hơn yêu cầu cần sử dụng. Như mô tả trong Hình 2.4, phương pháp BASTP cho

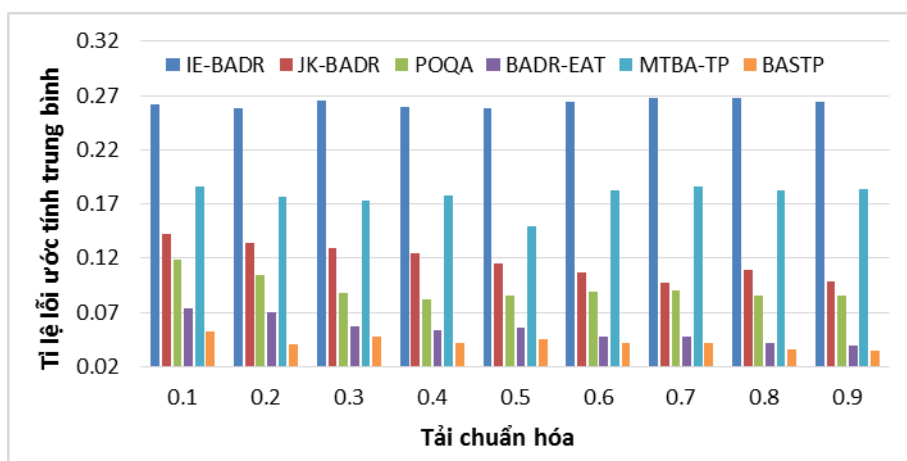
kết quả tập hợp tốt nhất với phân bố lỗi ước tính gần với giá trị 0 nhất.



**Hình 2.3** So sánh tỉ lệ lỗi ước tính trung bình của IE-BADR, JK-BADR, POQA, BADR-EAT, MTBA-TP và BASTP với tải chuẩn hóa đến 0.5



**Hình 2.4** Phân bố tỉ lệ lỗi ước tính của IE-BADR, JK-BADR, POQA, BADR-EAT, MTBA-TP và BASTP trong 100 lần tập hợp chòm liên tiếp



**Hình 2.5** Tỉ lệ lỗi ước tính trung bình gần như không đổi với tải chuẩn hóa từ 0.1 đến 0.9

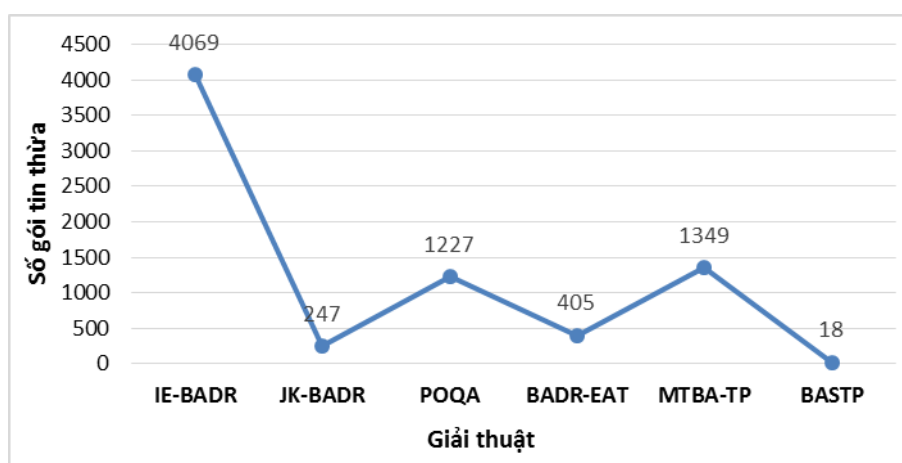
Một câu hỏi được đặt ra là liệu lưu lượng tải đến có tác động đến lỗi ước tính



hay không. Luận án tiến hành khảo sát trường hợp tải chuẩn hóa thay đổi từ 0.1 đến 0.9 và kết quả trong Hình 2.5 chỉ ra rằng lỗi ước tính gần như không phụ thuộc vào mật độ lưu lượng tải đến.

### b. So sánh số gói tin thừa được chuyển cho lần tập hợp chòm tiếp theo

Như đã phân tích trong Mục 2.1.2.2a, nguyên nhân của việc tạo ra các gói tin thừa là do chiều dài chòm ước tính thấp hơn so với kích thước thật của chòm được hoàn thành. Kết quả là các gói tin thừa sẽ chịu một độ trễ gia tăng bằng với thời gian tập hợp chòm  $T_a$  tiếp theo. Do đó việc làm giảm số gói tin thừa cũng đồng nghĩa với việc làm giảm đáng kể độ trễ trung bình của toàn mạng OBS.



Hình 2.6 So sánh số gói tin thừa trong 100 chòm sinh ra đầu tiên

Hình 2.6 so sánh số gói tin thừa của các phương pháp trong 100 lần tập hợp chòm liên tiếp, trong đó phương pháp BASTP có số gói tin thừa được chuyển cho chòm tiếp theo là thấp nhất.

### c. Phân tích cách chọn ngưỡng của phương pháp BASTP

Như được mô tả trong các Hình 2.3, 2.4, 2.5 và 2.6, phương pháp BASTP luôn cho kết quả mô phỏng tốt nhất. Tuy nhiên, kết quả này thường đi kèm với việc chọn cặp giá trị ngưỡng ( $L_{min}$ ,  $L_{max}$ ). Một thử nghiệm về mặt mô phỏng đối với các cặp ngưỡng khác nhau được chỉ ra trong Bảng 2.3, trong đó nếu ngưỡng được chọn quá thấp thì ngoài lỗi ước tính khá lớn, số lượng chòm sinh ra cũng nhiều và điều này sẽ làm tăng khả năng tắc nghẽn ở trong mạng lõi. Ngược lại nếu ngưỡng được chọn cao sẽ làm cho lỗi ước tính tăng cao hơn nhưng bù lại số gói tin thừa sẽ giảm, có nghĩa là sẽ làm giảm được độ trễ tăng thêm.

**Bảng 2.3 Ảnh hưởng cặp giá trị ngưỡng ( $L_{min}$ ,  $L_{max}$ ) đến lỗi ước tính (với tải chuẩn hóa 0.5)**

Cặp ngưỡng (bytes)	Lỗi ước tính trung bình	Số chùm sinh ra	Số gói thừa trong 100 chùm đầu tiên	Số chùm thừa trong 100 chùm đầu tiên	Tổng số gói trong 100 chùm đầu tiên
$L_{min}=10000$ , $L_{max}=50000$	0.05565637	2693	120	100	959
$L_{min}=50000$ , $L_{max}=80000$	0.050960653	2227	110	100	4775
$L_{min}=80000$ , $L_{max}=120000$	0.050175172	1720	99	99	8140
$L_{min}=120000$ , $L_{max}=160000$	0.036239469	401	70	70	13207
$L_{min}=160000$ , $L_{max}=200000$	0.158322079	172	4	4	15085

### 2.1.2.3 Nhận xét

Tóm lại, các phương pháp tập hợp chùm nêu trên đều cố gắng giảm độ trễ đệm chùm ( $T_a + T_o$ ) về thành độ trễ tập hợp ( $T_a$ ). Tuy nhiên, các phương pháp này vẫn bộc lộ các điểm chưa được xử lý triệt để sau:

(1) Lỗi ước tính là đáng kể, như được chỉ ra trong Hình 2.3. Lỗi ước tính sẽ gây ra lãng phí băng thông đặt trước nếu độ dài chùm ước tính dài hơn độ dài chùm được hoàn thành. Trong trường hợp kích thước chùm ước tính nhỏ hơn tổng gói tin đến thực tế tại một hàng đợi, các gói vượt quá sẽ được tập hợp trong chùm tiếp theo. Kết quả là các gói này phải chịu một độ trễ tăng thêm bằng độ trễ đệm chùm.

(2) Các phương pháp IE-BADR, POQA, BADR-EAT, MTBA-TP và BASTP chưa sử dụng hiệu quả lỗi ước tính cho các lần tập hợp chùm tiếp theo. Trong [30], JK-BADR sử dụng lỗi ước tính để điều chỉnh trực tiếp độ dài ước tính cho lần tập hợp chùm tiếp theo. Tuy nhiên cách làm này là không trơn, nên lỗi ước tính không hội tụ được về một giá trị tối thiểu (lý tưởng là giá trị zero).

(3) Để ước tính độ dài chùm hoàn thành, các phương pháp dựa trên thống kê được sử dụng như dựa trên các độ dài chùm đo được (trong POQA), dựa trên các ngưỡng thời gian trong  $M$  lần tập hợp chùm sau cùng nhất (trong BASTP) hay dựa

trên tốc độ trung bình của  $M$  gói tin đến sau cùng nhất (trong BADR-EAT). Các cách tiếp cận này giúp việc ước tính chính xác hơn, nhưng phải chịu chi phí tính toán lớn, đặc biệt khi tốc độ các gói tin đến tại nút biên OBS là rất cao. Việc giảm khối lượng tính toán bằng cách giảm cửa sổ thời gian ước tính một cách phù hợp do đó là rất cần thiết, nhưng vẫn phải đảm bảo được độ chính xác ước tính.

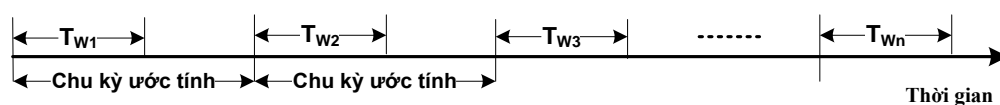
(4) Các phương pháp IE-BADR, JK-BADR, POQA và BADR-EAT sử dụng kỹ thuật tập hợp chùm dựa trên ngưỡng thời gian cố định ( $T_a$ ) và sau đó cố gắng ước tính độ dài chùm trong lần tập hợp chùm hiện thời ( $L^e$ ). MTBA-TP sử dụng một phương pháp lai nhưng vẫn với ngưỡng thời gian cố định ( $T_a$ ) và một ngưỡng độ dài định trước ( $L_{min}$ ). Cải tiến hơn, BASTP sử dụng một phương pháp lai, trong đó các ngưỡng thời gian ( $T_a$ ) và ngưỡng độ dài ( $L^e$ ) được ước tính trước một cách linh động. Tuy nhiên, do  $T_a$  được tính là trung bình các ngưỡng thời gian  $T_j$  của  $M$  lần tập hợp chùm trước đó, nên nó không phản ánh đúng xu hướng tăng/giảm gần đây nhất của lưu lượng đến; hơn nữa việc điều chỉnh từng bước  $L^e$  sẽ không đáp ứng kịp khi lưu lượng đến bùng phát và tăng đột ngột. Một vấn đề khác mà BASTP phải đối mặt là việc chọn cặp giá trị  $L_{min}$  và  $L_{max}$  sao cho phù hợp như đã được chỉ ra trong Bảng 3.3.

Các phân tích, so sánh và đánh giá này (đã được công bố trong Công trình [CT1]) chính là cơ sở để luận án đề xuất các cải tiến về tập hợp chùm giảm độ trễ được trình bày trong các mục tiếp theo.

### 2.1.3 Phương pháp tập hợp chùm giảm độ trễ iBADR

#### 2.1.3.1 Giới thiệu về phương pháp ước tính tốc độ đến TW-EWMA

Nhằm ước tính tốc độ của các gói tin đến tại một hàng đợi, Salad và cộng sự trong [23] đã đề xuất phương pháp TW-EWMA. Khác với các phương pháp ước tính khác thường đếm hết các gói tin đến trong các khoảng thời gian quan sát liên tục (chu kỳ ước tính), phương pháp TW-EWMA sử dụng một cửa sổ thời gian ước tính nhỏ hơn ( $T_w$ ) nhằm giảm chi phí tính toán trên hệ thống (Hình 2.7).



**Hình 2.7 Phương pháp dự đoán theo cửa sổ của TW-EWMA**

Phương pháp TW-EWMA ước tính tốc độ đến trung bình của các gói tin dựa

trên 2 yếu tố: (1) tốc độ đến trung bình tích lũy trước đó ( $\lambda_{avg}$ ) và tốc độ đến trong cửa sổ quan sát hiện thời ( $\lambda_{cur}$ ) theo Công thức 2.9

$$\lambda_e = (1 - \alpha) \times \lambda_{avg} + \alpha \times \lambda_{cur} \quad (2.9)$$

trong đó  $\alpha$  là một trọng số thể hiện mối tương quan giữa tốc độ đến trung bình trước đó với tốc độ đến hiện thời và  $\alpha$  có miền giá trị nằm trong đoạn  $[0, 1]$ ;  $\lambda_{cur}$  được xác định dựa trên độ dài chùng ( $L_w$ ) trong cửa sổ quan sát ( $T_w$ ) theo Công thức 2.10

$$\lambda_{cur} = \frac{L_w}{T_w} \quad (2.10)$$

### 2.1.3.2 Mô tả phương pháp tập hợp chùng giảm độ trễ iBADR

Phương pháp tập hợp chùng giảm độ trễ được luận án đề xuất iBADR (*improved Burst Assembly for Delay Reduction*) cũng dựa trên ý tưởng gửi sớm gói tin điều khiển tại thời điểm  $t_1 = T_a - T_o$  (ở đây  $T_a$  luôn lớn hơn  $T_o$ ) và chùng tương ứng được gửi đi tại thời điểm  $t_2 = T_a$ ; kết quả là các gói tin được tập hợp trong chùng hiện thời sẽ giảm được một độ trễ  $T_o$  (xem Hình 2.3b). Nhưng do thông tin độ dài chùng cần được mang trong gói điều khiển tại thời điểm nó được gửi đi, nên việc ước tính độ dài chùng là cần thiết. Có nhiều cách tiếp cận khác nhau có thể giúp ước tính chính xác độ dài chùng, trong đó cách ước tính dựa trên thống kê thường có ưu điểm hơn đối với các sự kiện rời rạc, như được dẫn chứng ở Hình 2.3. Nhưng do phải tính toán trên một lượng dữ liệu lớn nên các phương pháp ước tính dựa trên thống kê thường có độ phức tạp lớn và việc giảm cửa sổ thời gian là một giải pháp nhằm giảm nhẹ thời gian tính toán. Luận án sử dụng phương pháp TW-EWMA (xem Mục 2.1.3.1) để ước tính tốc độ các gói tin đến, từ đó ước tính được độ dài chùng sẽ hoàn thành. Cửa sổ ước tính trong trường hợp này là  $T_w = T_a - T_o$ . Tốc độ các gói tin đến được ước tính dựa trên TW-EWMA như sau:

$$\lambda_e = (1 - \alpha) \times \lambda_{avg} + \alpha \times \frac{L_w}{T_a - T_o} \quad (2.11)$$

trong đó  $\lambda_{avg}$  tốc độ trung bình của các gói tin đến trước đó,  $L_w$  là số gói tin đến trong cửa sổ ước tính hiện thời và  $\alpha$  là một trọng số thể hiện mức độ tác động của tốc độ trung bình và tốc độ hiện thời của các gói tin đến đối với việc ước tính.

Các tác giả trong [23] thiết lập  $\alpha$  bằng một giá trị cố định (0.3), mà điều này thực tế không phản ánh được bản chất thay đổi bất thường của lưu lượng đến; kết quả là lỗi ước tính là đáng kể. Luận án đề xuất thay đổi  $\alpha$  một cách linh động chuyển biến theo tỉ lệ của tốc độ đến hiện thời ( $\lambda_{cur}$ ) so và tốc độ trung bình ( $\lambda_{avg}$ ) của của các gói tin đến như Công thức 2.12.

$$\frac{\alpha}{1-\alpha} = \frac{\lambda_{cur}}{\lambda_{avg}} \Rightarrow \alpha = \frac{\lambda_{cur}}{\lambda_{avg} + \lambda_{cur}} \quad (2.12)$$

Để tăng độ chính xác của việc ước tính hơn nữa, luận án điều chỉnh linh động ngưỡng thời gian tập hợp chòm hiện thời dựa trên lỗi ước tính trung bình của các lần tập hợp chòm trước đó theo Công thức 2.13.

$$R = (1-\alpha) \times \bar{R}_E + \alpha \times \frac{(L-L^e)}{L} \quad (2.13)$$

Ngưỡng thời gian  $T_a$  cho lần tập hợp sau được xác định bằng tổng của ngưỡng thời gian  $T_a$  của lần tập hợp liền trước đó và một gia số điều chỉnh dựa trên lỗi ước tính trung bình ( $T_a \times R$ ):  $T_a := T_a + T_a \times R$ . Bằng cách này, ngưỡng thời gian tập hợp chòm hiện thời được đẩy về gần hơn với thời điểm độ dài hoàn thành chòm bằng với độ dài ước tính. Do đó, phương pháp tập hợp chòm giảm độ trễ cải tiến có tên gọi là iBADR.

### 2.1.3.3 Mô tả giải thuật iBADR

Phương pháp iBADR có giải thuật được mô tả chi tiết như sau:

#### Giải thuật 1: iBADR

##### Input:

$T_a$ ; // ngưỡng thời gian tập hợp  
 $T_o$ ; // giá trị thời gian offset  
 $S_q$ ; // danh sách các gói tin đến trong hàng đợi

##### Output: $\bar{R}_E$ ;

// lỗi ước tính trung bình

##### Begin

1  $\lambda_{avg} := 0$ ; // tốc độ đến trung bình các gói tin  
 2  $R_E := 0$ ; // khởi tạo lỗi ước tính  
 3  $t_1 := T_a - T_o$ ; // thời điểm gửi gói điều khiển  
 4  $t_2 := T_a$ ; // thời điểm gửi chòm dữ liệu  
 5  $M := 0$ ; // số chòm sinh ra

```

6   $b := 0;$  // khởi tạo bộ đệm chòm
7   $KT := \text{false};$  // kiểm tra thời điểm gửi gói điều khiển
8  while ( $S_q \neq \emptyset$ ) do
9     $p :=$  gói tin đến hàng đợi;  $S_q := S_q \setminus \{p\};$ 
10    $T_q := s_p;$  //  $s_p$  là thời điểm đến gói tin  $p$ 
11    $b := b + L_p;$  //  $L_p$  là kích thước gói tin  $p$ 
12   if ( $(T_q \geq t_1)$  and  $(KT = \text{false})$ ) then // giai đoạn 1: gửi gói điều khiển
13      $L := b;$  // độ dài chòm hiện thời
14      $\lambda_{\text{cur}} := L / (T_a - T_0);$  // tốc độ đến các gói tin trong cửa số ước tính
15      $\alpha := \lambda_{\text{cur}} / (\lambda_{\text{cur}} + \lambda_{\text{avg}});$  // điều chỉnh trọng số  $\alpha$ 
16      $\lambda_{\text{avg}} := (1 - \alpha) \times \lambda_{\text{avg}} + \alpha \times \lambda_{\text{cur}};$ 
17      $L^e := L + T_0 \times \lambda_{\text{avg}};$  // độ dài chòm ước tính được
18      $KT := \text{true};$  // kiểm tra gói điều khiển đã gửi
19   end if
21   if ( $T_q \geq t_2$ ) then // giai đoạn 2: gửi chòm dữ liệu
22      $L := b;$  // độ dài chòm hoàn thành
23      $R := (1 - \alpha) \times \bar{R}_E + \alpha \times (L - L^e) / L;$ 
24      $T_a := T_a + T_a \times R;$  // điều chỉnh  $T_a$  theo tỉ lệ lỗi ước tính  $R$ 
25      $b := 0;$ 
26      $t_1 := (T_a - T_0) + T_q;$ 
27      $t_2 := T_a + T_q;$ 
28      $KT := \text{false};$ 
29      $R_E := R_E + |L - L^e| / L;$  // cập nhật lỗi ước tính
30      $M := M + 1;$  // cập nhật số chòm sinh ra
31      $\bar{R}_E := R_E / M;$  // cập nhật lỗi ước tính trung bình
32   end if
33 end while
34 return  $\bar{R}_E;$ 
End

```

Độ phức tạp tính toán thời gian của giải thuật iBADR chủ yếu thực hiện ở vòng lặp **while** (từ dòng 8 đến dòng 33); do độ phức tạp của các lệnh trong vòng lặp **while** là  $O(1)$ , nên độ phức tạp tính toán của giải thuật là  $O(N)$ , ở đây  $N$  là số gói tin đến trong hàng đợi  $S_q$ . Độ phức tạp tính toán của giải thuật iBADR là tương đương với độ phức tạp tính toán của các giải thuật tập hợp chòm giảm độ trễ đã được công bố; do chúng đều tuân theo nguyên tắc tập hợp chòm là duyệt qua các gói tin đến

trong hàng đợi  $S_q$ .

#### 2.1.3.4 So sánh và đánh giá dựa trên kết quả mô phỏng

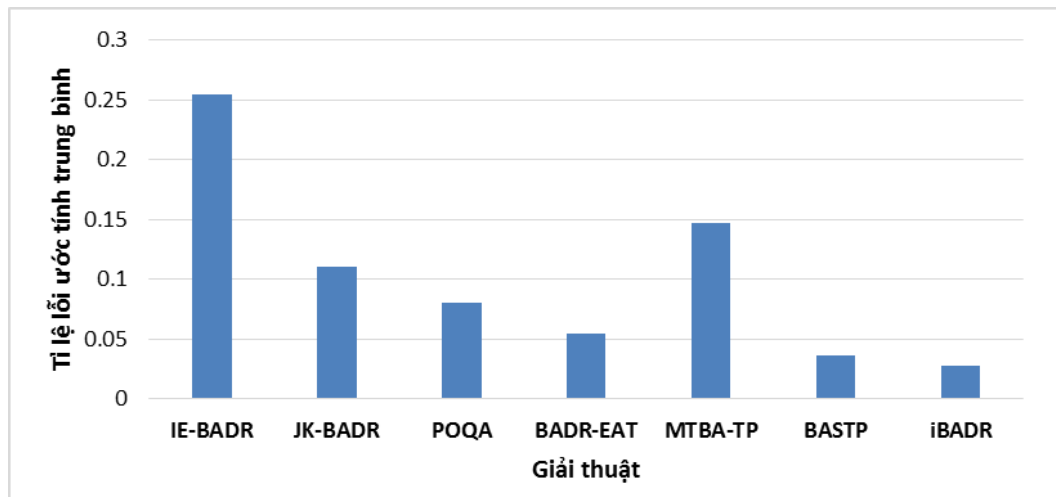
Với các mục tiêu mô phỏng bao gồm:

- So sánh tỉ lệ lỗi ước tính trung bình ( $\bar{R}_E$ ) của iBADR với các phương pháp đã công bố.
- So sánh số gói tin thừa phải chuyển cho chòm tiếp theo trong 100 chòm sinh ra đầu tiên.

Luận án sử dụng các tham số cài đặt trong phần này tương tự Mục 2.1.2.2.

##### a. So sánh tỉ lệ lỗi ước tính trung bình

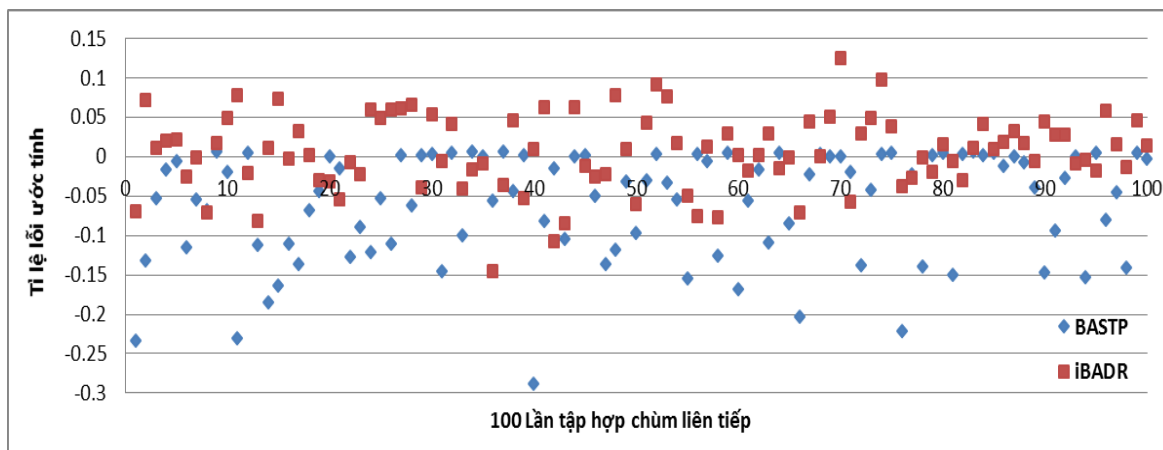
Kết quả ở Hình 2.8 cho thấy rằng phương pháp iBADR có tỉ lệ lỗi ước tính nhỏ nhất. Điều này có được là do, đầu tiên việc cải tiến thuật toán TW-EWMA với  $\alpha$  thay đổi linh hoạt theo tốc độ các gói tin đến (Công thức 2.12) đã làm cho việc ước tính trở nên chính xác hơn. Hơn nữa, việc điều chỉnh ngưỡng thời gian tập hợp chòm dựa trên lỗi ước tính (Công thức 2.13) đã giúp kéo độ dài chòm hoàn thành ( $L$ ) về gần hơn với độ dài chòm ước tính ( $L^e$ ).



**Hình 2.8 Tỉ lệ lỗi ước tính trung bình của các phương pháp tập hợp chòm trước đây với phương pháp tập hợp chòm cải tiến (iBADR)**

Để thấy rõ hơn điều này, luận án trích xuất lỗi ước tính của 100 lần tập hợp chòm liên tiếp của iBADR và BASTP, phương pháp tập hợp chòm tốt nhất trong số các phương pháp đã được công bố trước đây. Kết quả Hình 2.9 cho thấy rằng tỉ lệ lỗi ước tính của iBADR phân bố xung quanh 0 (trục hoành), trong khi rất nhiều lỗi ước

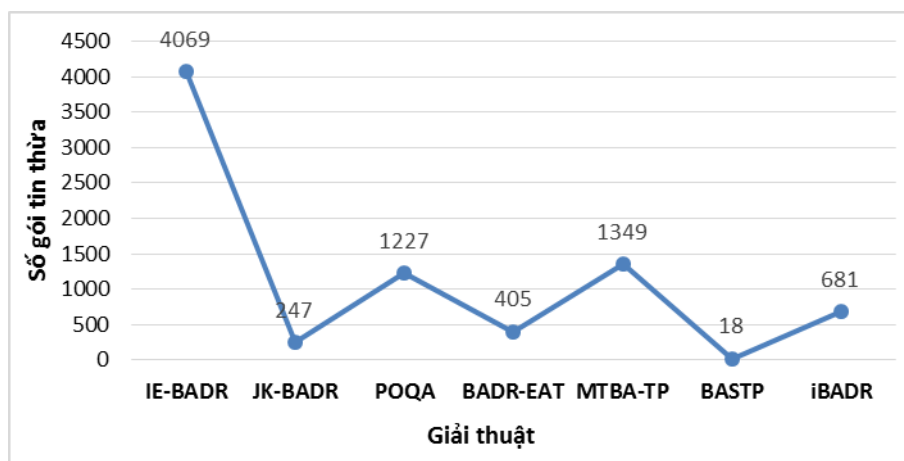
tính của BASTP phân bố xa hơn.



Hình 2.9 Phân bố lỗi ước tính của 100 chùm sinh ra đầu tiên của BASTP và iBADR

**b. So sánh số gói tin thừa trong 100 chùm sinh ra liên tiếp**

Tuy nhiên, kết quả Hình 2.9 cũng cho thấy rằng, lỗi ước tính của BASTP đa số chỉ phân bố ở miền âm (ở dưới trục hoành), trong khi của iBADR là xung quanh trục hoành. Điều này phản ánh một điều rằng, BASTP chỉ gây ra lãng phí băng thông do độ dài ước tính luôn lớn hơn độ dài chùm thực tế hoàn thành; trong khi iBADR có khá nhiều chùm hoàn thành có kích thước vượt quá độ dài chùm ước tính. Kết quả là các gói tin thừa phải được chuyển tập hợp vào chùm sau và chúng sẽ chịu một độ trễ tăng thêm bằng với thời gian tập hợp chùm của lần sau ( $T_a$ ). Hình 2.10 mô tả một so sánh về số gói tin thừa của iBADR và các phương pháp đã công bố, trong đó iBADR có số chùm thừa là đáng kể.



Hình 2.10 Số gói tin thừa trong 100 chùm sinh ra đầu tiên



### 2.1.3.5 Nhận xét

Dựa trên kết quả mô phỏng, phương pháp iBADR cho tỉ lệ lỗi ước tính giảm hơn so với BASTP. Tuy nhiên, nếu xét về số gói tin thừa phải chuyển cho chòm tiếp theo thì iBADR sinh ra tương đối nhiều như Hình 2.10. Nguyên nhân của vấn đề này là do iBADR không sử dụng một giá trị ngưỡng độ dài để làm cận trên, kết quả là kích thước các chòm sinh ra có biên độ dao động khá ngẫu nhiên. Để giải quyết vấn đề này luận án đã đề xuất một phương pháp tập hợp chòm mới tốt hơn mà sẽ được trình bày trong phần tiếp theo. Phương pháp tập hợp chòm giảm độ trễ iBADR được đề xuất trong mục này đã được công bố trong [CT2].

### 2.1.4 Phương pháp tập hợp chòm giảm độ trễ OBADR

#### 2.1.4.1 Mô tả phương pháp tập hợp chòm giảm độ trễ OBADR

Phương pháp OBADR (*Optimal Burst Assembly for Delay Reduction*) là một cải tiến tiếp theo của iBADR, trong đó ngoài áp dụng phương pháp ước tính độ dài chòm TW-EWMA với  $\alpha$  được điều chỉnh linh hoạt, quá trình tập hợp chòm là một kết hợp của 2 giai đoạn tập hợp: giai đoạn đầu dựa vào bộ đếm thời gian và giai đoạn thứ 2 dựa vào ngưỡng độ dài. Cụ thể của phương pháp OBADR được mô tả như sau:

- **Giai đoạn 1:** khi gói tin đầu tiên đến hàng đợi, bộ đếm thời gian (*timer*) được kích hoạt. Gói điều khiển chỉ được gửi vào mạng lõi khi *timer* đạt đến ngưỡng  $T_w$ , là kích thước của cửa sổ thời gian. Độ dài ước tính ( $L^e$ ) đồng thời cũng được tính toán dựa trên phương pháp TW-EWMA với  $\alpha$  được điều chỉnh linh hoạt.
- **Giai đoạn 2:** Tiến trình tập hợp chòm vẫn được tiếp tục, nhưng bây giờ dựa trên ngưỡng độ dài ước tính  $L^e$ . Chòm chỉ được hoàn thành khi số lượng gói tin đến trong hàng đợi đạt đến ngưỡng  $L^e$ .

Với OBADR, lỗi ước tính sẽ được giảm thiểu; đặc biệt lỗi ước tính sẽ bằng 0 khi các gói tin đến có cùng kích thước. Trong trường hợp các gói tin đến có kích thước thay đổi, điều kiện để hoàn thành một chòm là tổng độ dài các gói tin trong hàng đợi nằm trong đoạn  $[L^e - max_p, L^e]$ , trong đó  $max_p$  là kích thước lớn nhất có thể của các gói tin đến. Rõ ràng cách tiếp cận này có gây ra một ít lãng phí về mặt băng

thông, tức là băng thông được đặt trước dựa trên chiều dài ước tính có thể nhiều hơn so với độ dài thực tế của chùm được hoàn thành; nhưng nó đảm bảo rằng không có gói tin thừa nào bị chuyển sang chùm tiếp sau và do đó không có độ trễ tăng thêm, đây là yếu tố tối ưu của OBADR.

#### 2.1.4.2 Mô tả giải thuật OBADR

Giải thuật OBADR được mô tả chi tiết như sau:

##### Giải thuật 2: OBADR

###### Input:

$T_a$ ; // ngưỡng thời gian tập hợp  
 $T_o$ ; // giá trị thời gian offset  
 $S_q$ ; // danh sách các gói tin đến trong hàng đợi  
 $max_p$ ; // kích thước lớn nhất của gói tin đến trong hàng đợi  $S_q$

**Output:**  $\bar{R}_E$ ; // lỗi ước tính trung bình

###### Begin

```

1   $\lambda_{avg} := 0$ ; // tốc độ đến trung bình các gói tin
2   $R_E := 0$ ; // khởi tạo lỗi ước tính
3   $t_1 := T_a - T_o$ ; // thời điểm gửi gói điều khiển
4   $M := 0$ ; // số chùm sinh ra
5   $b := 0$ ; // khởi tạo bộ đệm chùm
6   $KT := false$ ; // kiểm tra thời điểm gửi gói điều khiển
7  while ( $S_q \neq \emptyset$ ) do
8     $p :=$  gói tin đến hàng đợi;  $S_q := S_q \setminus \{p\}$ ;
9     $T_q := s_p$ ; //  $s_p$  là thời điểm đến gói tin  $p$ 
10    $b := b + L_p$ ; //  $L_p$  là kích thước gói tin  $p$ 
11   if ( $(T_q \geq t_1)$  and  $(KT = false)$ ) then // Giai đoạn 1: gửi gói điều khiển
12      $L := b$ ; // độ dài chùm hiện thời
13      $\lambda_{cur} := L / (T_a - T_o)$ ; // tốc độ đến các gói tin trong cửa sổ ước tính
14      $\alpha := \lambda_{cur} / (\lambda_{cur} + \lambda_{avg})$ ; // điều chỉnh trọng số  $\alpha$ 
15      $\lambda_{avg} := (1 - \alpha) \times \lambda_{avg} + \alpha \times \lambda_{cur}$ ;
16      $L^e := L + T_o \times \lambda_{avg}$ ; // độ dài chùm ước tính được
17      $KT := true$ ; // kiểm tra gói điều khiển đã gửi
18   end if
19   if ( $L^e - max_p \leq |b| \leq L^e$ ) then // Giai đoạn 2: gửi chùm
20      $L := b$ ; // độ dài chùm hoàn thành
21      $R_E := R_E + |L - L^e| / L$ ; // cập nhật tổng lỗi ước tính
22      $b := 0$ ;

```

```

23       $t_1 := t_1 + T_q$ ;
24       $KT := \text{false}$ ;
25       $M := M + 1$ ;           // cập nhật số chòm sinh ra
26  end if
27  end while
28   $\bar{R}_E := R_E / M$ ;
29  return  $\bar{R}_E$ ;

```

**End**

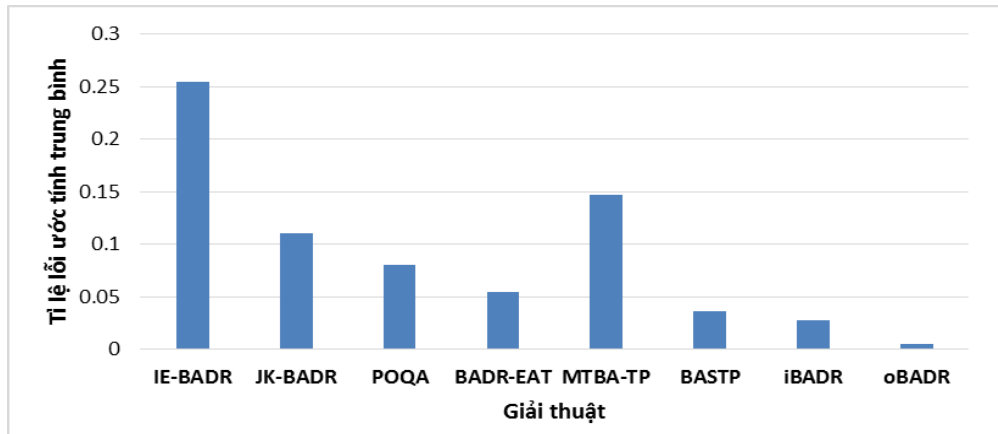
Độ phức tạp tính toán thời gian của giải thuật OBADR chủ yếu thực hiện ở vòng lặp **while** (từ dòng 7 đến dòng 27), do độ phức tạp của các lệnh trong vòng lặp **while** là  $O(1)$ , nên độ phức tạp tính toán của giải thuật là  $O(N)$ , ở đây  $N$  là số gói tin đến trong hàng đợi  $S_q$ . Độ phức tạp tính toán của giải thuật OBADR là tương đương với độ phức tạp tính toán của các giải thuật tập hợp chòm giảm độ trễ đã được công bố; do chúng tuân theo nguyên tắc của tập hợp chòm khi duyệt qua tất cả các gói tin đến trong hàng đợi  $S_q$ .

#### 2.1.4.3 So sánh và đánh giá dựa trên kết quả mô phỏng

Các tham số mô phỏng là tương tự như trong Mục 2.1.2.2. Mục đích mô phỏng nhằm đánh giá hiệu quả của OBADR so với iBADR và các phương pháp đã được công bố trước đó.

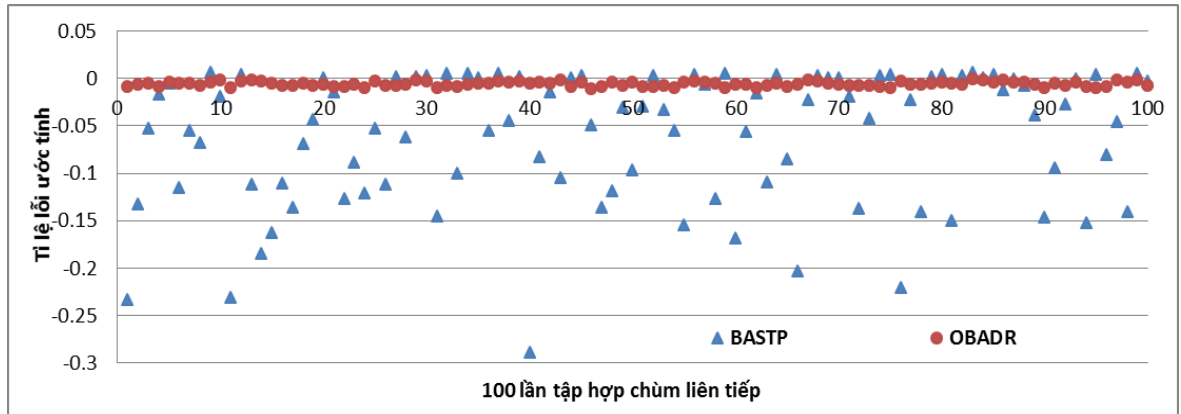
##### a. So sánh tỉ lệ lỗi ước tính trung bình

Hình 2.11 cho thấy OBADR có tỉ lệ lỗi ước tính ( $\bar{R}_E$ ) thấp hơn tất cả các phương pháp đã được đề xuất trước đó.



Hình 2.11 So sánh tỉ lệ lỗi ước tính trung bình giữa các phương pháp tập hợp giảm độ trễ

Cụ thể hơn với tải chuẩn hóa đến là 0.5, Hình 2.12 so sánh phân bố tỉ lệ lỗi ước tính của OBADR so với BASTP (phương pháp tập hợp chùm giảm độ trễ tốt nhất đã công bố), trong đó OBADR có phân bố tỉ lệ lỗi ước tính gần với 0, trong khi phân bố tỉ lệ lỗi ước tính của BASTP xa hơn nhiều.

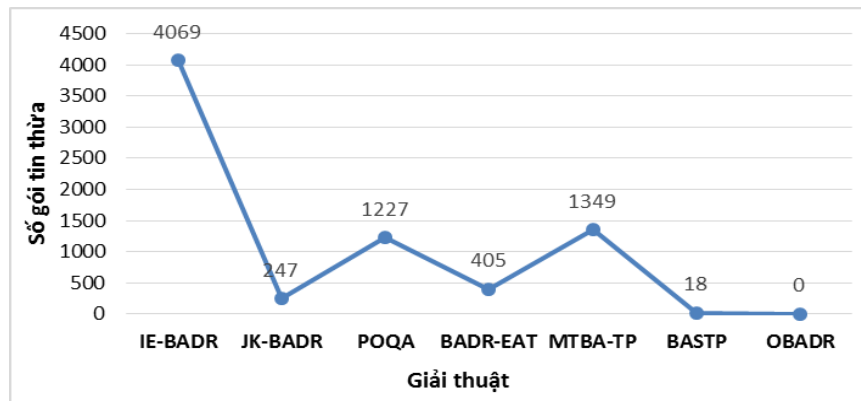


**Hình 2.12 Phân bố lỗi ước tính trong 100 lần tập hợp chùm liên tiếp của phương pháp OBADR với BASTP**

Điều này có được nhờ các cải tiến trong phương pháp tập hợp chùm của OBADR nên chiều dài chùm hoàn thành bao giờ cũng gần bằng hoặc bằng kích thước chùm ước tính (Do độ dài ước tính được sử dụng làm ngưỡng tập hợp chùm trong Giai đoạn 2 của OBADR). Tỉ lệ lỗi ước tính là không thể tránh khỏi do sự đa dạng kích thước các gói tin đến; nếu trong trường hợp kích thước của các gói tin đến bằng nhau hoặc là bội số của nhau thì lỗi ước tính của OBADR là bằng không.

**b. So sánh số gói tin thừa trong 100 chùm sinh ra liên tiếp**

Như thể hiện ở Hình 2.13, phương pháp OBADR không có số gói tin thừa, nhờ độ dài ước tính được sử dụng làm ngưỡng độ dài.



**Hình 2.13 Số gói tin thừa trong 100 chùm sinh ra liên tiếp**

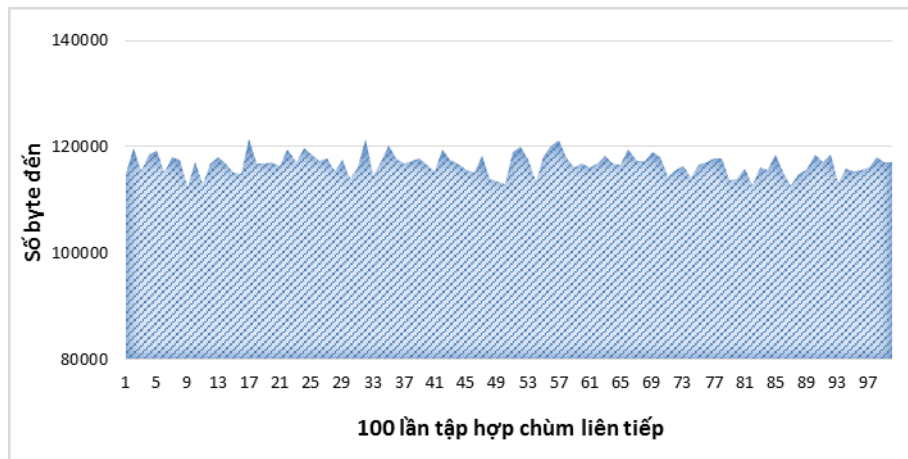
#### 2.1.4.4 Nhận xét

Thông qua mô phỏng đã khẳng định được rằng phương pháp OBADR với việc thay đổi  $\alpha$  theo Công thức 2.12 sẽ cho lỗi ước tính thấp hơn so với trường hợp  $\alpha$  cố định bằng 0.3. Tuy nhiên, trường hợp nào áp dụng  $\alpha$  động sẽ có hiệu quả và miền giá trị tốt nhất nào cho  $\alpha$ , phần tiếp theo của luận án sẽ làm rõ vấn đề này. Phương pháp tập hợp chùm giảm độ trễ OBADR được đề xuất trong mục này đã được công bố trong [CT3].

#### 2.1.5 Ảnh hưởng của trọng số $\alpha$ đến OBADR

Theo Công thức 2.12, tốc độ ước tính phụ thuộc vào tốc độ trung bình của các gói tin đến trước đó  $\lambda_{avg}$  và tốc độ các gói tin đến trong cửa sổ quan sát hiện thời  $\lambda_{cur}$ . Xét trong điều kiện luồng các gói tin đến ít có biến động (như Hình 2.14), tốc độ các gói tin đến trong cửa sổ quan sát  $\lambda_{cur}$  gần như không khác biệt so với tốc độ trung bình của các gói tin đến trước đó  $\lambda_{avg}$ , tức là  $\lambda_{cur} = \lambda_{avg}$ , khi đó tốc độ ước tính ( $\lambda_e$ ) trong Công thức 2.9 được viết lại là:

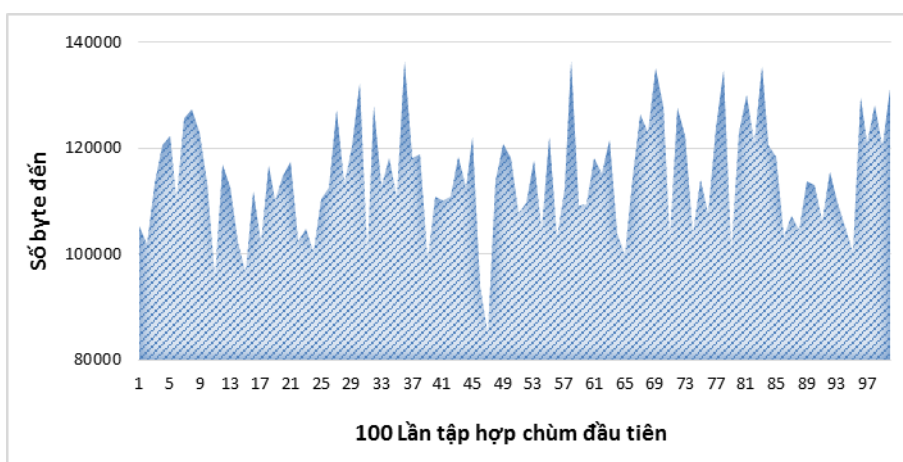
$$\lambda_e = (1 - \alpha) \times \lambda_{avg} + \alpha \times \lambda_{avg} \Rightarrow \lambda_e = \lambda_{avg} \quad (2.14)$$



**Hình 2.14 Trường hợp tốc độ luồng các gói tin đến không có nhiều biến đổi**

Rõ ràng  $\alpha$  đã biến mất trong công thức tính tốc độ ước tính, hay nói cách khác  $\alpha$  không có tác động đến việc ước tính độ dài chùm. Tuy nhiên, trong trường hợp luồng các gói tin đến có nhiều biến động (Hình 2.15) như đã được cài đặt trong các mô phỏng trước của luận án với các trường hợp tốc độ đến tăng/giảm đột biến, tức là  $\lambda_{cur} \neq \lambda_{avg}$ , nên  $\alpha$  sẽ có tác động đáng kể đến việc ước tính độ dài chùm. Một đánh giá

dựa trên mô phỏng trong Mục 2.1.5.1 và 2.1.5.2 sẽ làm rõ tác động của  $\alpha$  đến việc ước tính độ dài chùm. Các tham số mô phỏng là tương tự như Mục 2.1.2.2.



**Hình 2.15 Trường hợp tốc độ luồng các gói tin đến có nhiều biến đổi với các trường hợp tăng/giảm đột biến**

### 2.1.5.1 Khảo sát sự biến đổi của trọng số $\alpha$ khi tải đến thay đổi

Với tải chuẩn hóa thay đổi từ 0.1 đến 0.9 và các giá trị  $\alpha$  được khảo sát từ 0.1 đến 0.9, kết quả thu được trong Bảng 2.4 cho thấy rằng lỗi ước tính tối thiểu (các ô có nền màu sẫm) có phân bố tương ứng với  $\alpha$  trong khoảng (0.4, 0.6). Như vậy, việc thiết lập  $\alpha$  cố định rõ ràng không phù hợp đối với các tải đến khác nhau.

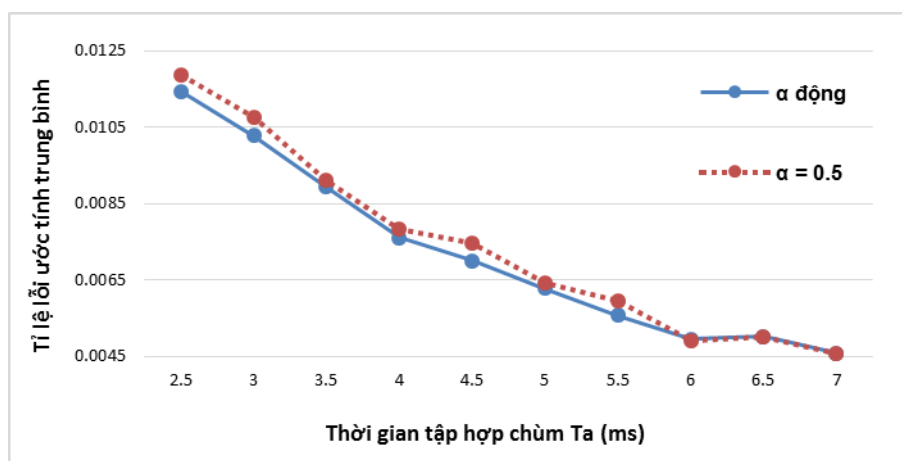
**Bảng 2.4 Lỗi ước tính  $\bar{R}_E$  với tải chuẩn hóa thay đổi và các giá trị  $\alpha$  từ 0.1 đến 0.9**

Tải \ $\alpha$	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9
0.1	0.02740	0.02656	0.02546	0.02651	<b>0.02457</b>	0.02612	0.02705	0.02667	0.02651
0.2	0.01361	0.01359	0.01341	<b>0.01304</b>	0.01348	0.01370	0.01445	0.01350	0.01343
0.3	0.00905	0.00861	0.00912	<b>0.00842</b>	0.00867	0.00897	0.00883	0.00896	0.00909
0.4	0.00686	0.00675	0.00695	0.00672	0.00705	<b>0.00651</b>	0.00665	0.00666	0.00673
0.5	0.00551	0.00519	0.00549	0.00526	<b>0.00489</b>	0.00545	0.00521	0.00518	0.00518
0.6	0.00447	0.00456	0.00445	<b>0.00426</b>	0.00443	0.00452	0.00437	0.00462	0.00439
0.7	0.00414	0.00377	0.00377	0.00373	<b>0.00370</b>	0.00382	0.00384	0.00399	0.00385
0.8	0.00327	0.00354	0.00332	0.00333	0.00342	<b>0.00326</b>	0.00332	0.00343	0.00337
0.9	0.00300	0.00300	0.00308	0.00299	<b>0.00289</b>	0.00298	0.00316	0.00299	0.00311

Một kết quả khác cũng được rút ra từ Bảng 2.4 là khi tải đến tăng thì lỗi ước tính tương ứng giảm với mọi giá trị  $\alpha$  được khảo sát. Nguyên nhân là do Giai đoạn 2 của phương pháp OBADR sử dụng giải thuật tập hợp chùm dựa trên độ dài chùm ước tính  $L^e$ , nên khi tải tăng, các gói tin đến nhiều giúp ngưỡng độ dài ( $L^e$ ) luôn đạt đến trước, đẩy kích thước thật của chùm hoàn thành  $L$  về gần với  $L^e$ . Điều này đã làm giảm đáng kể lỗi ước tính.

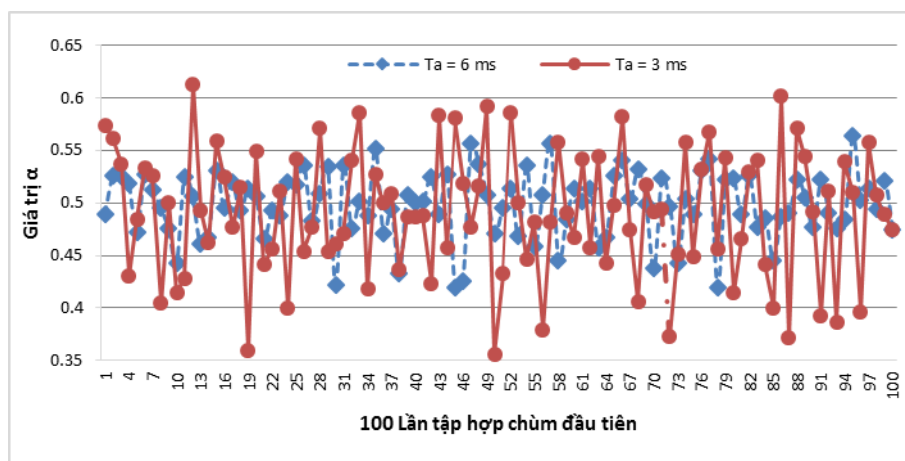
### 2.1.5.2 So sánh hiệu quả tập hợp chùm khi $\alpha$ cố định và $\alpha$ thay đổi

Như chỉ ra trong Bảng 2.4, giá trị  $\alpha$  cố định tối ưu có phân bố từ 0.4 đến 0.6. Với tải chuẩn hóa 0.5, một so sánh giữa  $\alpha$  cố định và  $\alpha$  động dựa trên lỗi ước tính trung bình được chỉ ra trong Hình 2.16.



**Hình 2.16** So sánh lỗi ước tính trung bình với trường hợp  $\alpha$  động và  $\alpha$  tĩnh ( $\alpha = 0.5$ ) khi thay đổi thời gian tập hợp chùm ( $T_a$ ) từ 2.5 ms đến 7.0 ms

Kết quả trong Hình 2.16 cho thấy khi thời gian tập hợp chùm bé (từ 2.5 ms đến 5.5 ms),  $\alpha$  động cho kết quả lỗi ước tính trung bình tốt hơn so với  $\alpha$  cố định ( $\alpha = 0.5$ ). Điều này là do khi thiết lập thời gian tập hợp chùm bé thì cửa sổ ước tính  $T_w = T_a - T_o$  sẽ nhỏ,  $\alpha$  động lúc này sẽ góp phần phản ánh sự thay đổi đột ngột luồng chùm đến tốt hơn, mà minh chứng là giá trị  $\alpha$  sẽ có sự thay đổi (dao động) lớn hơn như được chỉ ra ở Hình 2.17. Nếu thời gian tập hợp lớn, do việc tính trung bình trong khoảng thời gian lớn sẽ không phản ánh được các thay đổi đột ngột của luồng chùm đến mà kết quả là làm cho giá trị  $\alpha$  ít thích nghi hơn, nên lỗi ước tính không có sự thay đổi đáng kể so với giá trị  $\alpha$  cố định (từ 6 ms đến 7 ms).



**Hình 2.17 Sự biến thiên giá trị  $\alpha$  động trong 100 lần tập hợp chùm liên tiếp với  $T_a=6\text{ ms}$  và  $T_a=3\text{ ms}$**

Hình 2.16 cũng phản ánh một điều là lỗi ước tính có xu hướng giảm khi tăng thời gian tập hợp chùm. Thực tế độ chính xác của việc ước tính phụ thuộc vào kích thước của cửa sổ ước tính, trong đó cửa sổ càng lớn thì việc ước tính càng chính xác hơn. Tuy nhiên, việc thiết lập thời gian tập hợp lớn cũng sẽ làm gia tăng kích thước các chùm sinh ra, gây thêm khó khăn trong quá trình lập lịch chùm trong mạng lõi do có ít khoảng trống phù hợp với kích thước chùm.

### 2.1.5.3 Nhận xét

Dựa trên các kết quả mô phỏng việc điều chỉnh giá trị  $\alpha$  linh hoạt (Như trong Công thức 2.12) theo tốc độ của luồng dữ liệu đến đã làm tăng hiệu quả của việc ước tính độ dài chùm hoàn thành. Rõ ràng tính chất của luồng đến có tác động đáng kể đến độ chính xác của việc ước tính tính độ dài chùm hoàn thành, được thể hiện thông qua tỉ lệ lỗi ước tính như trong Bảng 2.4 và Hình 2.16. Kết quả này cũng khẳng định hiệu quả của việc điều chỉnh linh hoạt  $\alpha$  theo tốc độ của luồng dữ liệu đến. Kết quả trong phần này đã được công bố trong [CT4].

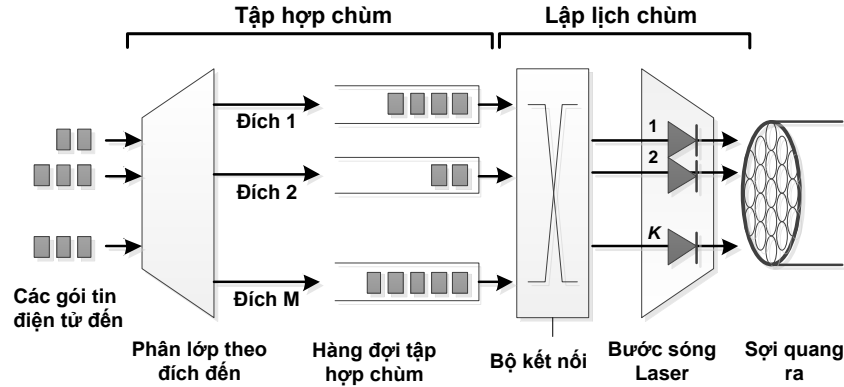
## 2.1.6 Ảnh hưởng của OBADR đến hoạt động lập lịch chùm

### 2.1.6.1 Phân tích ảnh hưởng của OBADR dựa trên phương pháp Engset

Như được mô tả trong Hình 2.18, nút biên vào OBS có 2 hoạt động chính: tập hợp chùm và lập lịch chùm. Mô hình tập hợp chùm tối ưu giảm độ trễ (OBADR) đã được trình bày trong Mục 2.1.4. Kết quả ra của hoạt động này là tập các chùm mà sẽ được phân phối lên các kênh của sợi quang ra. Đặt  $Q$  là số hàng đợi tương ứng với số



nguồn sinh chùm và  $W$  là số kênh bước sóng khả dụng trên sợi quang ra, kết quả thu được là một mô hình  $W$  máy chủ phục vụ  $Q$  khách hàng. Nếu  $W \geq Q$  thì sẽ không có mất dữ liệu khi đặt trước tài nguyên, vấn đề chỉ xảy ra khi  $Q > W$ .



**Hình 2.18 Hai hoạt động chính tại nút biên: tập hợp chùm và lập lịch chùm ở cổng ra**

Như đã phân tích trong [4], phương pháp Engset đã được sử dụng để xem xét khả năng đặt trước tài nguyên bước sóng sau quá trình tập hợp chùm. Với phương pháp tập hợp chùm truyền thống thì gói điều khiển sẽ được gửi tại thời điểm  $T_a$  (thời gian tập hợp chùm) và sau khoảng thời gian  $T_o$  (thời gian đặt trước tài nguyên) thì chùm dữ liệu sẽ được gửi. Gọi  $B$  là khoảng thời gian mà bộ đệm bận [4],  $B$  sẽ được tính bởi Công thức 2.15

$$B = \frac{1}{\mu} + \left( \frac{\lambda_{avg}}{\mu} \right) \times (T_a + T_o) \quad (2.15)$$

trong đó,  $\lambda_{avg}$  là tốc độ đến trung bình các gói tin và  $\mu$  là tốc độ phục vụ; như vậy  $1/\mu$  và  $1/\lambda_{avg}$  lần lượt là độ dài trung bình (theo đơn vị thời gian) của các gói tin và thời gian đến trung bình của các gói tin.

Phương pháp Engset sử dụng giá trị thời gian trung bình ON/OFF trong đó ON là thời gian kênh truyền bị chiếm dụng và OFF là thời gian kênh truyền nhàn rỗi. Theo Zalesky và cộng sự trong [4] giả sử thời gian truyền dữ liệu xấp xỉ  $B$  thì ta có giá trị thời gian trung bình ON và OFF tương ứng là  $T_o + B$  và  $1/\lambda_{avg} + T_a$ .

Xác suất  $\{P_k | k = 0, 1, \dots, W\}$  tương ứng với số lượng kênh bận được tính bởi Công thức 2.16

$$P_k = \frac{\binom{Q}{k} \times \rho_k}{\sum_{q=0}^W \binom{Q}{q} \times \rho^q} \quad (2.16)$$

trong đó

$$\rho \approx (T_o + B) / \left( \frac{1}{\lambda_{avg}} + T_a \right) \quad (2.17)$$

Với lưu lượng tải chùm được cung cấp  $O$  (Công thức 2.18)

$$O = \rho \sum_{k=0}^Q (Q - W) \times P_k \quad (2.18)$$

và lưu lượng tải chùm đến thật sự  $C$  (Công thức 2.19)

$$C = \sum_{k=0}^W k \times P_k \quad (2.19)$$

xác suất mất chùm  $P$  được tính bởi Công thức 2.20

$$P = (O - C)/O \quad (2.20)$$

Công thức 2.20 là xác suất tắc nghẽn chùm trong trường hợp tập hợp chùm truyền thống với độ trễ đệm chùm là tổng của thời gian tập hợp chùm và thời gian *offset*. Tuy nhiên với phương pháp tập hợp chùm giảm độ trễ OBADR, thời gian *offset*  $T_o$  được bao gồm trong thời gian tập hợp chùm, do đó Công thức 2.15 được thay đổi thành Công thức 2.21

$$B = \frac{1}{\mu} + \left( \frac{\lambda_{avg}}{\mu} \right) \times T_a \quad (2.21)$$

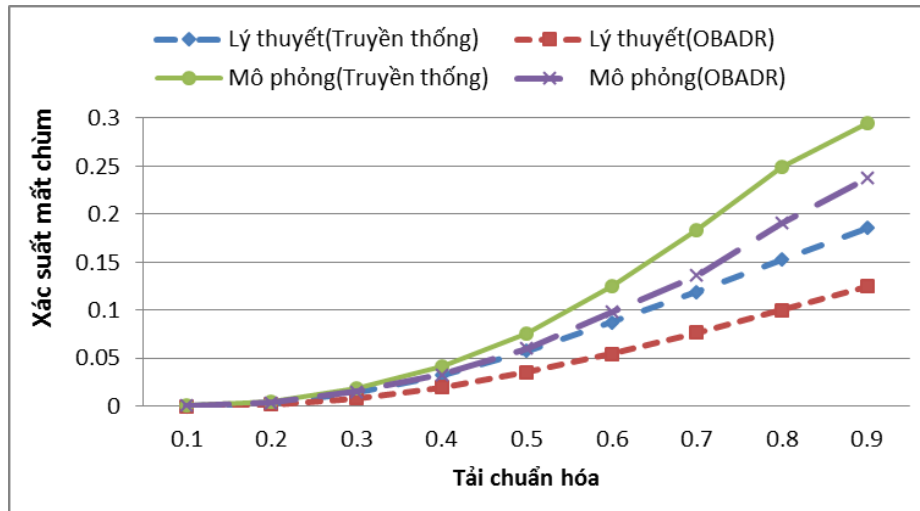
Các phương trình còn lại không có thay đổi.

### 2.1.6.2 So sánh hiệu quả lập lịch giữa mô hình phân tích và kết quả mô phỏng

Luận án xem xét một nút biên vào với 12 hàng đợi và 8 bước sóng trên liên kết ra. Chiều dài trung bình các gói tin đến hàng đợi là 1000 bytes, thời gian tập hợp chùm  $T_a = 0.6 \mu s$  và thời gian *offset*  $T_o = 0.2 \mu s$ ; tải chuẩn hóa được khảo sát từ 0.1 đến 0.9. Kết quả phân tích được vẽ trên phần mềm Mathematica [72] và kết quả mô phỏng được trích xuất từ NS2 [71].

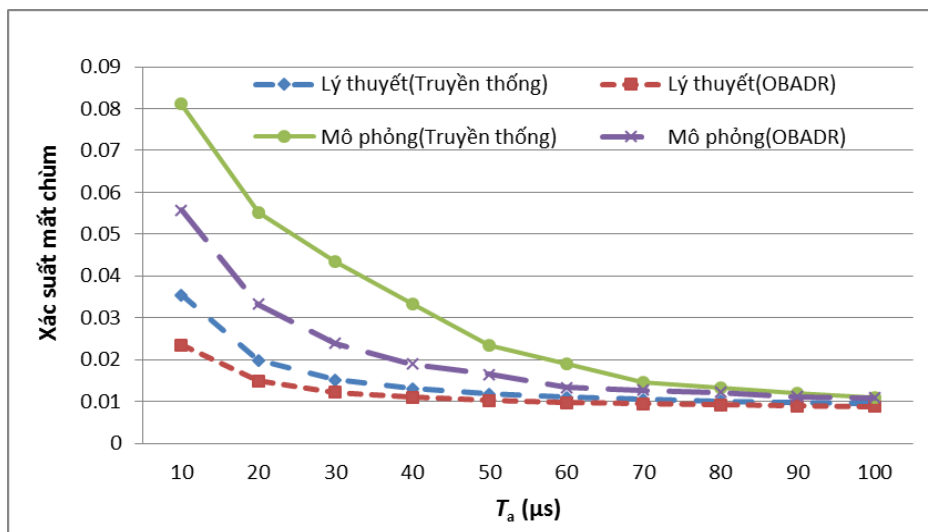
Như được mô tả trong Hình 2.19, OBADR cho kết quả tốt hơn so với phương pháp tập hợp chùm truyền thống về lý thuyết và mô phỏng. Khi tải đến thấp, tỉ lệ mất chùm về lý thuyết và mô phỏng đều không đáng kể; nhưng khi tải đến cao, tắc nghẽn

xảy ra nhiều hơn làm cho tỉ lệ mất chùm tăng về mặt lý thuyết và mô phỏng. Điều này làm cho khoảng cách giữa lý thuyết và mô phỏng tăng, nhưng khoảng cách theo tỉ lệ là không thay đổi đáng kể (khoảng 3%).



**Hình 2.19** So sánh tỉ lệ mất chùm giữa OBADR và tập hợp chùm truyền thống

Với thay đổi ngưỡng thời gian tập hợp chùm  $T_a$  từ 10 đến 100  $\mu s$ , Hình 2.20 cho thấy rằng OBADR vẫn tốt hơn so với phương pháp tập hợp chùm truyền thống. Khi ngưỡng thời gian tập hợp chùm thấp, việc tập hợp làm gia tăng số lượng các chùm sinh ra trong mạng dẫn đến tỉ lệ mất chùm tăng lên, điều này làm cho khoảng cách giữa lý thuyết và mô phỏng tăng hơn so với thời gian tập hợp chùm lớn.



**Hình 2.20** So sánh tỉ lệ mất chùm của OBADR với tập hợp chùm truyền thống

### 2.1.6.3 Nhận xét

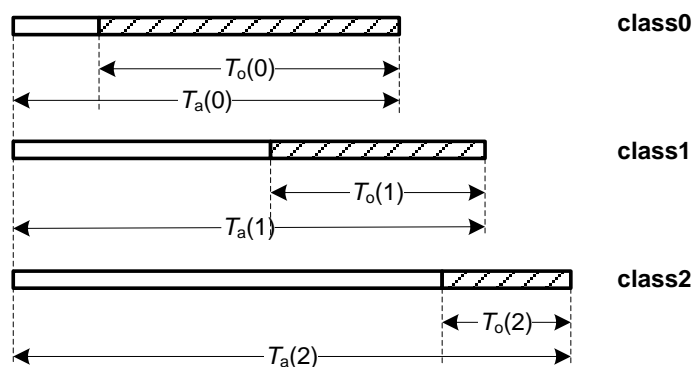
Phương pháp OBADR đã chứng tỏ được hiệu quả trong quá trình tập hợp chùm

góp phần giảm độ trễ đáng kể, trong đó việc không có gói tin bị mất do đặt tài nguyên không đủ đã làm cho các chòm không phải chịu bất kỳ một độ trễ tăng thêm nào. Với một số ứng dụng có yêu cầu cao về độ trễ thì yếu tố này góp phần nâng cao hiệu năng của mạng đáp ứng độ trễ của ứng dụng đồng thời góp phần điều khiển công bằng độ trễ khi mở rộng trên các lớp QoS khác nhau như được trình bày ở trong Mục 2.2. Phương pháp tập hợp chòm giảm độ trễ OBADR được đề xuất trong mục này đã được công bố trong [CT3].

## 2.2 Mô hình tập hợp chòm công bằng độ trễ

### 2.2.1 Các công trình nghiên cứu liên quan

Các mô hình tập hợp chòm giảm độ trễ đã công bố đều có ý tưởng chung là gửi sớm gói điều khiển một khoảng thời gian *offset* trước khi hoàn thành chòm. Trong các mô hình này, chỉ có POQA trong [69] là có kết hợp với hỗ trợ phân biệt dịch vụ. Cụ thể, các tác giả trong [69] đã sử dụng các thời gian *offset* khác nhau cho các chòm có lớp ưu tiên khác nhau và điều chỉnh thời gian tập hợp đối với các chòm sao cho chòm ưu tiên cao luôn có thời gian đệm chòm ngắn hơn chòm ưu tiên thấp. Như ví dụ được chỉ ra trong Hình 2.21, chòm thuộc lớp ưu tiên cao nhất class0 có thời gian tập hợp chòm  $T_a(0)$  bé nhất và giá trị thời gian *offset*  $T_o(0)$  lớn nhất, trong khi lớp ưu tiên thấp nhất class2 có thời gian tập hợp chòm  $T_a(2)$  dài nhất và giá trị thời gian *offset*  $T_o(2)$  bé nhất.



**Hình 2.21 Một ví dụ về 3 ngưỡng thời gian tập hợp chòm và 3 giá trị thời gian offset**

Mục tiêu của việc điều chỉnh thời gian tập hợp của POQA không chỉ nhằm làm cho chòm ưu tiên cao hơn sẽ có độ trễ đầu cuối ngắn hơn, mà còn nhằm bảo đảm độ trễ của các chòm không được vượt quá ràng buộc độ trễ đầu cuối của chúng. Cách

tiếp cận này được các tác giả trong [69] diễn dịch là một giải pháp cho công bằng độ trễ. Tuy nhiên, không có một xem xét nào về mối tương quan trong việc điều chỉnh giám độ trễ giữa các chùm ưu tiên khác nhau và cũng không có một cách định lượng hiệu quả công bằng nào của giải pháp được đề xuất. Phần tiếp theo của luận án sẽ phân tích và đề xuất một tiếp cận về công bằng độ trễ và cách đánh giá mức độ công bằng này giữa các luồng chùm ưu tiên khác nhau.

## **2.2.2 Phương pháp tập hợp chùm công bằng độ trễ BADF**

### **2.2.2.1 Giới thiệu về công bằng độ trễ trong mạng OBS**

Đã có một số cách diễn dịch khác nhau về khái niệm công bằng, nhưng ý tưởng chung của nó là việc làm hài lòng đối với các cá nhân, như phân phối các nguồn tài nguyên mạng giữa các ứng dụng khác nhau sao cho sự công bằng đạt được khi tài nguyên mạng được phân bổ một cách công bằng [38]. Trong mạng OBS, khái niệm về công bằng cũng đã được xem xét dưới hình thức công bằng thông lượng [53] và công bằng khoảng cách [10]. Cụ thể, công bằng thông lượng đề cập đến việc phân bổ băng thông công bằng cho các kết nối theo tỷ lệ băng thông cung cấp và băng thông khả dụng trên một liên kết ra chung; trong khi công bằng khoảng cách là vấn đề công bằng giữa luồng có hành trình dài so với luồng có hành trình ngắn hơn, trong đó luồng có hành trình dài do phải đi qua nhiều nút trung gian hơn nên có tỉ lệ mất mát dữ liệu cao hơn so với luồng có hành trình ngắn. Với công bằng độ trễ được đề xuất trong [69], các chùm có ưu tiên càng cao sẽ có thời gian đệm chùm càng ngắn. Tuy nhiên, cách diễn dịch này vẫn chưa thể hiện được bản chất của sự đáp ứng công bằng đối với các cá nhân trong khái niệm công bằng. Vì vậy, luận án bổ sung khái niệm công bằng độ trễ như sau: Công bằng độ trễ là sự hài lòng về độ trễ giữa các chùm ưu tiên khác nhau, sao cho tỉ lệ trung bình về độ trễ đầu - cuối với giới hạn độ trễ của chúng là xấp xỉ nhau. Ngoài ra, để đáp ứng yêu cầu về sự phân biệt ưu tiên dựa trên độ trễ trong mạng OBS, hai ràng buộc sau được bổ sung vào.

1. Chùm có mức độ ưu tiên càng cao thì có độ trễ đầu cuối càng thấp; và
2. Độ trễ đầu cuối của một chùm không lớn hơn giới hạn độ trễ tối đa của nó (Ví dụ: RTT của các gói IP được mang trong chùm).

Như vậy khái niệm “Công bằng độ trễ” của luận án bổ sung đã bao hàm khái

niệm công bằng độ trễ được đề xuất trong [69].

### 2.2.2.2 Chỉ số công bằng độ trễ

Như đề cập trong Mục 2.2.1, độ trễ đầu cuối của các gói tin trong mạng OBS phụ thuộc chính vào độ trễ đệm chòm tại nút biên vào. Do đó, sự công bằng về độ trễ đệm chòm đối với các chòm có mức độ ưu tiên khác nhau đóng vai trò chính đối với công bằng độ trễ đầu cuối của các gói tin.

Gọi  $D(i)$  là độ trễ trung bình dữ liệu phải chờ trong hàng đợi  $i$  trước khi được tập hợp thành một chòm và  $T_a(i)$  là thời gian tập hợp chòm trên hàng đợi  $i$ , đại lượng  $x_i = D(i)/T_a(i)$  sẽ phản ánh mức độ trễ dữ liệu trong hàng đợi  $i$ . Luận án đề xuất công thức tính chỉ số công bằng độ trễ  $DFI$  cho các chòm ưu tiên khác nhau dựa trên công thức của Jain trong [39] như sau:

$$DFI = \frac{\left(\sum_{i=1}^n \sigma_i x_i\right)^2}{n \sum_{i=1}^n (\sigma_i x_i)^2} \quad (2.22)$$

Mức độ công bằng sẽ tăng khi  $DFI$  tiến đến 1, và bằng 1 khi  $\sigma_1 x_1 = \sigma_2 x_2 = \dots = \sigma_n x_n$ , trong đó  $\sigma_i$  là trọng số của  $x_i$ ,  $0 < \sigma_i < 1$  và  $\sum_{i=1}^n \sigma_i = 1$ .

Với công thức của Jain trong [39], đại lượng  $x_i$  thông thường là độ trễ, thông lượng hay tỉ lệ mất dữ liệu. Tuy nhiên, nếu chỉ dựa trên độ trễ giảm trong mỗi hàng đợi thì sẽ không phản ánh được mối tương quan của các mức độ giảm độ trễ giữa các hàng đợi (công bằng độ trễ). Vì vậy, luận án đề xuất đại lượng  $x_i = D(i)/T_a(i)$  nhằm phản ánh tỉ lệ thay đổi độ trễ giữa các hàng đợi. Thêm vào đó, trong công thức  $DFI$ , Luận án bổ sung trọng số  $\sigma_i$  nhằm biểu diễn mức độ ưu tiên của các hàng đợi khác nhau; giá trị  $\sigma_i$  càng lớn thì mức độ ưu tiên càng cao, nhưng các ràng buộc sau cần được đảm bảo:  $0 < \sigma_i < 1$  và  $\sum_{i=1}^n \sigma_i = 1$ .

Để cho đơn giản trong quá trình xử lý và cài đặt, luận án giả thiết các giá trị  $\sigma_i$  bằng nhau. Khi đó, Công thức 2.22 có thể viết lại là:

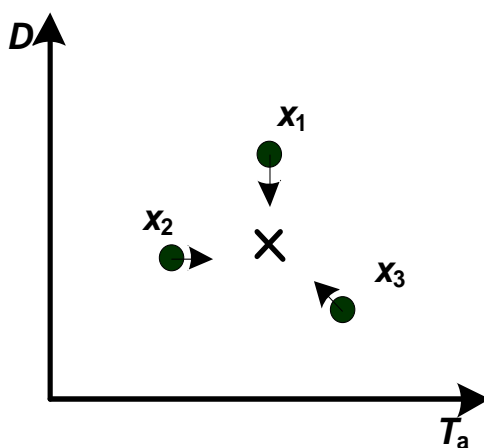
$$DFI = \frac{\left(\sum_{i=1}^n x_i\right)^2}{n \sum_{i=1}^n (x_i)^2} \quad (2.23)$$

Cũng tương tự với Công thức 2.22, mức độ công bằng sẽ tăng lên khi  $DFI$  tiến đến 1 và bằng 1 khi  $x_1 = x_2 = \dots = x_n$ .

Lưu ý rằng,  $D(i)$  là một biến và thay đổi phụ thuộc vào tốc độ dữ liệu đến tại hàng đợi  $i$ . Nếu muốn đẩy giá trị  $DFI$  về 1,  $T_a(i)$  cần được điều chỉnh sao cho các  $x_i$  gần nhau. Như ví dụ được chỉ ra trong Hình 2.22, với sự phân bố của các giá trị  $x_i$  trong không gian  $(D, T_a)$ , việc điều chỉnh  $DFI$  đến 1 tương đương với việc điều chỉnh tất cả các  $x_i$  tiến đến tâm của chúng. Theo đó, việc điều chỉnh  $DFI$  đến 1 như sau:

- Xác định tâm của tất cả giá trị  $x_i$ : 
$$\bar{x} = \frac{\sum_{i=1}^n x_i}{n}$$

- Di chuyển mỗi giá trị  $x_i$  đến tâm  $x_i = x_i + \theta \times (\bar{x} - x_i)$ , trong đó  $\theta$  là bước di chuyển và  $0 \leq \theta \leq 1$ . Trong mô phỏng được trình bày ở Mục 2.3.2.5, giá trị  $\theta$  được chọn là 0.1.



Hình 2.22 Ví dụ về 3 chòm ưu tiên có  $x_i$  phân bố trong không gian  $(D, T_a)$

### 2.2.2.3 Phương pháp tập hợp chòm 2 giai đoạn

Phương pháp tập hợp chòm giảm độ trễ được luận án đề xuất BADF (*Burst Assembly for Delay Fairness*) cũng dựa trên ý tưởng gửi sớm gói điều khiển (xem Mục 2.1.3 và 2.1.4), với những điểm mới đến từ mô hình tập hợp chòm 2 giai đoạn được đề xuất. Giai đoạn 1 là tập hợp chòm dựa trên ngưỡng thời gian ước tính và Giai đoạn 2 là tập hợp chòm dựa trên ngưỡng ước tính. Chi tiết của mô hình tập hợp chòm 2 giai đoạn như sau:

**Giai đoạn 1:** khi có gói tin đầu tiên đến tại hàng đợi  $i$ , bộ đếm thời gian (*timer*)

bắt đầu được kích hoạt. Gói điều khiển chỉ được gửi khi *timer* đạt đến ngưỡng thời gian ước tính  $T_e(i) = T_a(i) - T_o(i)$ . Độ dài chòm ước tính  $L^e(i)$  được tính toán dựa vào phương pháp TW-EWMA [23]:

$$L^e(i) = T_a(i) \times \left( (1 - \alpha(i)) \times \lambda_{avg}(i) + \alpha(i) \times \lambda_{cur}(i) \right) \quad (2.25)$$

trong đó  $\alpha(i)$  là một trọng số trên hàng đợi  $i$ ,  $\lambda_{avg}(i)$  là tốc độ gói tin đến trung bình trong những lần tập hợp chòm trước và  $\lambda_{cur}(i)$  là tốc độ gói tin đến trung bình trong khoảng thời gian ước tính  $T_e(i)$ . Lưu ý rằng  $\lambda_{cur}(i) = L_w(i)/T_e(i)$ , với  $L_w(i)$  là độ dài chòm đo được trong khoảng thời gian  $T_e(i)$  của hàng đợi  $i$ .

Trong giai đoạn này, giá trị  $\alpha(i)$  được điều chỉnh tăng/giảm tùy thuộc vào tốc độ gói tin đến tại hàng đợi  $i$ , được tính bởi  $\alpha(i) = \lambda_{cur}(i)/(\lambda_{avg}(i) + \lambda_{cur}(i))$ , thay vì phải giữ cố định như trong TW-EWMA.

**Giai đoạn 2:** giải thuật tập hợp chòm tiếp tục được thực hiện cho đến khi hoặc ngưỡng độ dài  $L^e(i)$  đạt đến hoặc ngưỡng thời gian  $T_a(i)$  đạt đến.

Nếu giá trị ngưỡng độ dài đạt đến trước, lỗi ước tính sẽ được tối thiểu hóa (như được trình bày trong Mục 2.2); đặc biệt lỗi ước tính sẽ bằng 0 khi tất cả các gói tin đến đều có kích thước như nhau hoặc độ dài chòm ước tính  $L^e(i)$  là bội số của kích thước các gói tin đến. Trong trường hợp ngưỡng độ dài chòm đạt đến trước và các gói tin đến có kích thước khác nhau, điều kiện để hoàn thành một chòm là  $L^e(i) - \max_p(i) \leq |B(i)| \leq L^e(i)$ , với  $|B(i)|$  là kích thước của chòm hoàn thành từ hàng đợi  $i$  và  $\max_p(i)$  là kích thước tối đa có thể của các gói tin đến tại hàng đợi  $i$ . Cách tiếp cận này sẽ tạo ra một lượng nhỏ băng thông lãng phí (khi băng thông đặt trước dựa trên độ dài ước tính lớn hơn độ dài chòm được hoàn thành), nhưng nó đảm bảo rằng sẽ không có gói tin dư thừa nào được chuyển sang chòm tiếp theo và do đó sẽ không có một độ trễ tăng thêm nào. Trong trường hợp ngưỡng thời gian  $T_a(i)$  đạt đến, lỗi ước tính sẽ có xu hướng tăng hơn, do chòm hoàn thành ngắn hơn so với chòm ước tính, tuy nhiên vẫn sẽ không có một độ trễ tăng thêm nào trong trường hợp này.

#### 2.2.2.4 Mô tả giải thuật BADF

Giải thuật tập hợp chòm công bằng độ trễ có tên BADF được mô tả như sau:



### Giải thuật 3: BADF

#### Input:

$S_q$ ; // danh sách các gói tin đến trong hàng đợi  $i$   
 $D_{max}(i)$ ; // độ trễ tối đa trên hàng đợi  $i$

#### Output: $D_{avg}(i)$

// độ trễ trung bình các gói tin trong hàng đợi  $i$

#### Begin

```
1   $\theta := 0.1$ ;  
2   $D_{avg}(i) := T_a / 2$ ; // khởi tạo độ trễ trung bình hàng đợi  $i$   
3   $B(i) := 0$ ; // khởi tạo bộ đệm chòm  
4   $KT := false$ ; // kiểm tra thời điểm gửi gói điều khiển  
5   $T_e(i) := T_a(i) - T_o(i)$ ;  
6   $t_1(i) := T_e(i)$ ; // xác định thời điểm gửi gói điều khiển  
7  while ( $S_q \neq \emptyset$ ) do  
8     $p :=$  gói tin đến hàng đợi;  $S_q := S_q \setminus \{p\}$ ;  
9     $T(i) := s_p$ ; //  $s_p$  là thời điểm đến gói tin  $p$   
10    $B(i) := B(i) + L_p$ ; //  $L_p$  là kích thước gói tin  $p$   
11   if ( $(T(i) \geq t_1(i))$  and  $(KT = false)$ ) then // GD 1: gửi gói điều khiển  
12      $L(i) := B(i)$ ; // độ dài chòm hiện thời  
13      $\lambda_{cur}(i) := i / T_e(i)$ ; // tốc độ đến các gói tin trong cửa số ước tính  
14      $\alpha(i) := \lambda_{cur}(i) / (\lambda_{cur}(i) + \lambda_{avg}(i))$ ; // điều chỉnh trọng số  $\alpha(i)$   
15      $\lambda_{avg}(i) := (1 - \alpha(i)) \times \lambda_{avg}(i) + \alpha(i) \times \lambda_{cur}(i)$ ;  
16      $L^e(i) := L(i) + T_o(i) \times \lambda_{avg}(i)$ ; // độ dài chòm ước tính được  
17      $KT := true$ ; // kiểm tra gói điều khiển đã gửi  
18   end if  
19   if ( $(|B(i)| \in [L_e(i) - \max_p(i), L_e(i)])$  or  $(t(i) \geq T_a(i))$ ) then //GD 2: gửi chòm  
20      $L(i) := B(i)$ ; // độ dài chòm hoàn thành  
21      $B(i) := 0$ ;  
22     Xác định độ trễ trung bình các gói tin trong chòm hiện thời  $D(i)$ ;  
23      $x_i := D(i) / T_a(i)$ ; // xác định giá trị  $x_i$   
24      $\bar{x} := \frac{\sum_{j=1}^n x_j}{n}$ ; // xác định tâm  
25     if ( $D(i) / (x_i + \theta \times (\bar{x} - x_i)) \leq T_o(i)$ ) then // xác định  $T_a(i)$   
26        $T_a(i) := T_a(i)$ ;  
27     else if ( $T_o(i) < D(i) / (x_i + \theta \times (\bar{x} - x_i)) < RTT(i)$ ) then  
28        $T_a(i) := D(i) / (x_i + \theta \times (\bar{x} - x_i))$ ;  
29     else  
30        $T_a(i) := D_{max}(i)$ ;
```

```

31             end if
32     end if
33      $T_e(i) := T_a(i) - T_o(i);$  // cập nhật cửa sổ ước tính
34      $t_1(i) := T_e(i) + S_p;$  // cập nhật thời điểm gửi gói điều khiển
35      $KT := \text{false};$ 
36      $D_{\text{avg}}(i) := (D_{\text{avg}}(i) + D(i))/2;$  // cập nhật độ trễ trung bình
37     end if
38 end while
39 return  $D_{\text{avg}}(i);$ 
End

```

Độ phức tạp tính toán thời gian của giải thuật BADF chủ yếu thực hiện ở vòng lặp **while** (từ dòng 7 đến dòng 38), do độ phức tạp của các lệnh trong vòng lặp **while** là  $O(1)$ , nên độ phức tạp tính toán của giải thuật là  $O(N)$ , ở đây  $N$  là số gói tin đến trong hàng đợi  $S_q$ , và tương đương độ phức tạp thời gian của giải thuật POQA.

### 2.2.2.5 So sánh và đánh giá dựa trên kết quả mô phỏng

Mô phỏng được cài đặt trong một PC với cấu hình 2.4 GHz Intel Core 2 CPU, 2G RAM; dữ liệu được trích xuất từ NS2 [71] với gói hỗ trợ mạng OBS obs-0.9a. Các gói tin đến thuộc 3 lớp ưu tiên ( $K = 3$ ), nên sẽ có 3 hàng đợi được sử dụng cho việc tập hợp chùm tại nút biên vào. Quá trình đến của các gói tin tại các hàng đợi có phân bố Poisson và kích thước của chúng phân bố ngẫu nhiên trong khoảng [500, 1000] bytes. Giả sử giới hạn độ trễ (tính bằng mili giây) của các gói tin đến nằm trong khoảng [0.4, 1.0], giả thiết rằng các gói tin có giới hạn độ trễ nằm trong khoảng [0.4, 0.6) được phân bổ vào hàng đợi thứ 1 (có mức ưu tiên cao nhất), các gói tin có độ trễ nằm trong khoảng [0.6, 0.8) được phân bổ vào hàng đợi thứ 2 (có mức ưu tiên ở trung bình) và các gói tin còn lại nằm trong khoảng [0.8, 1.0] được phân bổ vào hàng đợi thứ 3 (có mức ưu tiên thấp nhất). Giá trị thời gian *offset* được thiết lập lần lượt là 0.3, 0.2 và 0.1 (ms) cho các hàng đợi 1, 2 và 3 tương ứng.

Mục tiêu mô phỏng bao gồm:

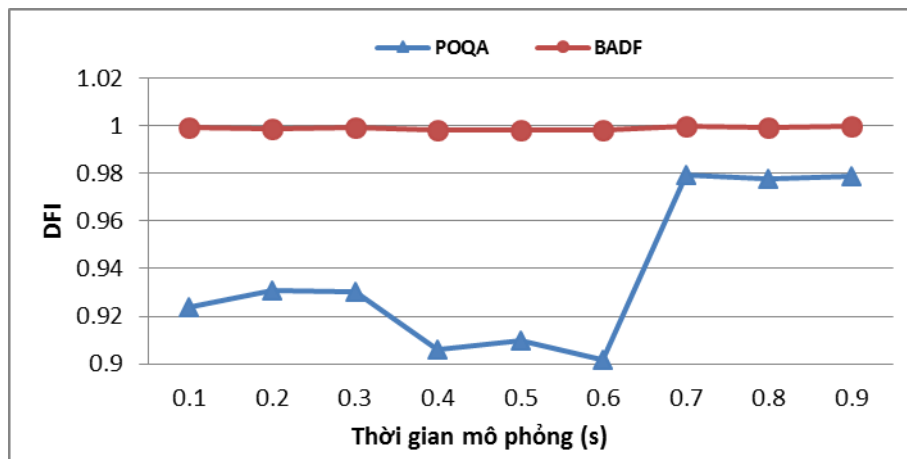
- So sánh chỉ số DFI giữa giải thuật BADF và giải thuật POQA;
- Phân tích hiệu quả của công bằng độ trễ đến thời gian tập hợp chùm  $T_a(i)$  và độ trễ đệm chùm;

- So sánh lỗi ước tính (Công thức 2.8) giữa BADF và POQA.

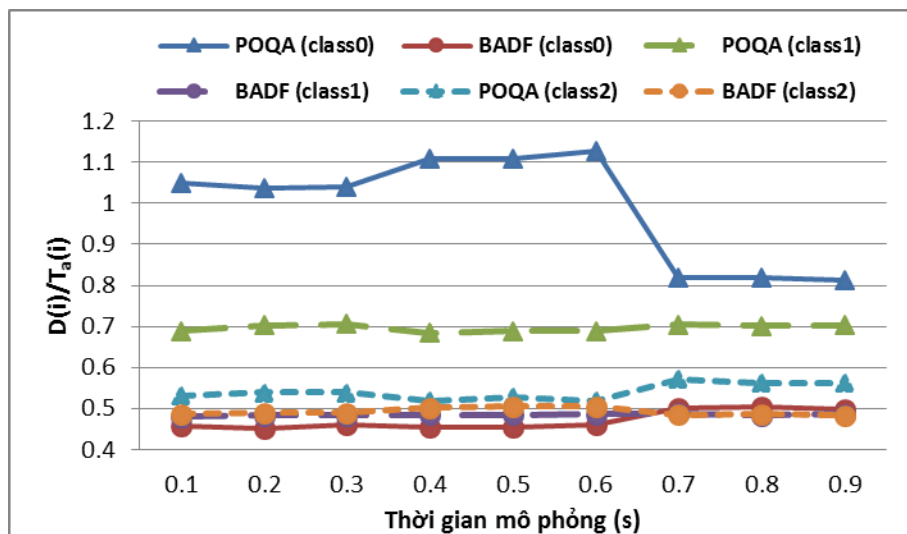
Luận án tiến hành 3 giai đoạn mô phỏng (trong 1s) với tải chuẩn hóa khác nhau: trong khoảng thời gian [0.1, 0.3] cả 3 hàng đợi có tải chuẩn hóa đến như nhau 0.2; tải chuẩn hóa 0.3, 0.2 và 0.1 lần lượt đến các hàng đợi 1, 2 và 3 trong khoảng thời gian [0.4, 0.6]; và tải chuẩn hóa 0.1, 0.2 và 0.3 lần lượt đến các hàng đợi 1, 2 và 3 trong khoảng thời gian [0.7, 0.9].

### a. So sánh chỉ số $DFI$ giữa giải thuật BADF và giải thuật POQA

Như thể hiện trong Hình 2.23, chỉ số  $DFI$  của giải thuật BADF gần tiến về 1 và tốt hơn so với giải thuật POQA. Điều này có nghĩa là BADF đạt đến gần mức tối ưu về công bằng độ trễ.



Hình 2.23 So sánh chỉ số  $DFI$  giữa BADF và POQA



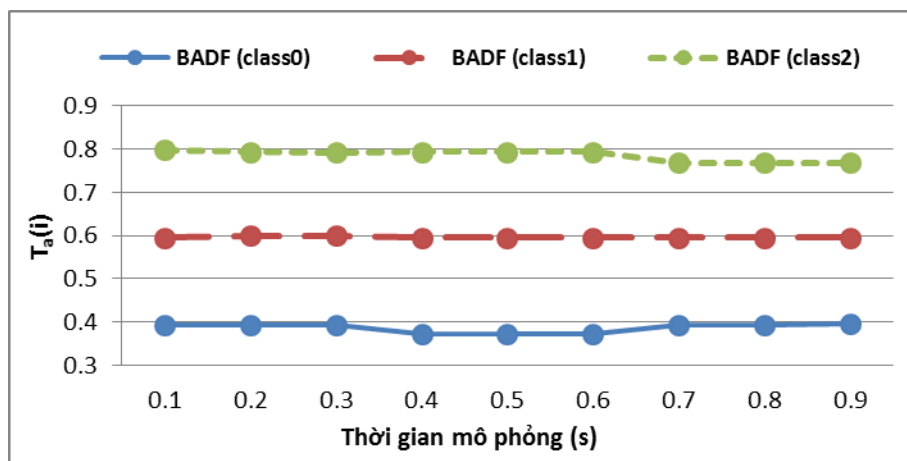
Hình 2.24 So sánh giá trị  $x_i = D(i)/T_a(i)$  của 3 lớp ưu tiên giữa BADF và POQA

Để làm rõ hơn về yếu tố ảnh hưởng đến sự công bằng độ trễ, luận án xem xét

đến tỉ lệ giữa độ trễ dữ liệu và ngưỡng thời gian tập hợp chùm của mỗi hàng đợi ( $x_i = D(i)/T_a(i)$ ) của giải thuật BADF và giải thuật POQA. Như được thể hiện trong Hình 2.24, khoảng cách giữa 3 lớp ưu tiên (class0, class1 và class2) trong giải thuật POQA là xa hơn nhiều so với giải thuật BADF ở cả 3 giai đoạn mô phỏng. Điều này phản ánh một thực tế là  $x_i$  của các lớp ưu tiên trong giải thuật BADF được đẩy đến tâm của chúng.

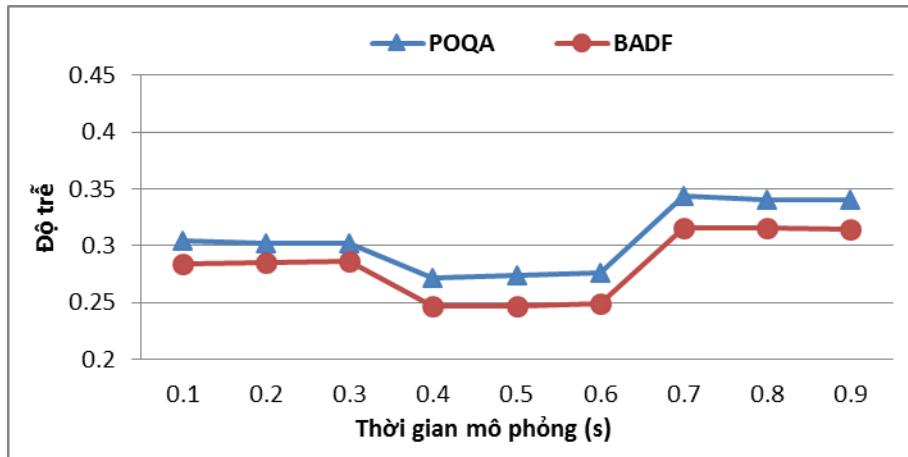
### b. Phân tích hiệu quả của công bằng độ trễ đến thời gian tập hợp chùm $T_a(i)$ và độ trễ đệm chùm

Như được thể hiện trong Hình 2.25, thời gian tập hợp chùm  $T_a(i)$  giảm khi tốc độ đến của các gói tin tăng, với class0 trong khoảng thời gian mô phỏng [0.4, 0.6] và với class2 trong thời gian mô phỏng [0.7, 0.9]. Điều này được giải thích là khi tốc độ đến của các gói tin tăng ngưỡng độ dài chùm luôn đạt đến trước và do đó giá trị  $T_a(i)$  được giảm đến thời gian tập hợp chùm thực tế, kết quả là khoảng cách giữa  $T_a(i)$  và độ trễ trung bình giữa các gói tin trên các hàng đợi được thu hẹp lại với nhau. Khi tốc độ đến của các gói tin giảm, với class2 trong khoảng thời gian mô phỏng [0.4, 0.6] và với class1 trong thời gian mô phỏng [0.7, 0.9] thì giá trị  $T_a(i)$  tăng, tuy nhiên do chúng không thể vượt quá giới hạn cận trên của nó (0.4 với class0 và 0.8 với class2) nên giá trị của nó không thay đổi.

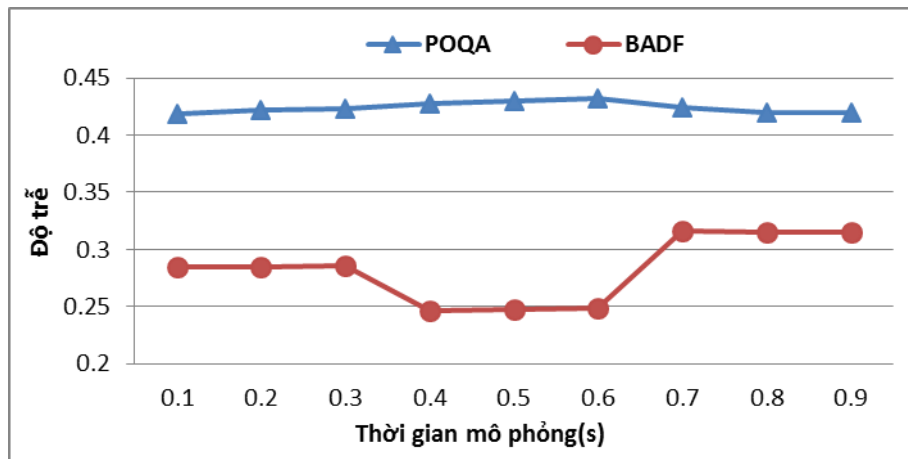


Hình 2.25 So sánh giá trị  $T_a(i)$  của 3 lớp ưu tiên với giải thuật BADF

Việc tối ưu hóa chỉ số  $DFI$  cũng có một tác động đáng kể đến độ trễ đệm chùm của các lớp ưu tiên. Cụ thể, độ trễ đệm chùm trung bình của 3 lớp ưu tiên trong giải thuật BADF là thấp hơn so với POQA (Hình 2.26 và 2.27).

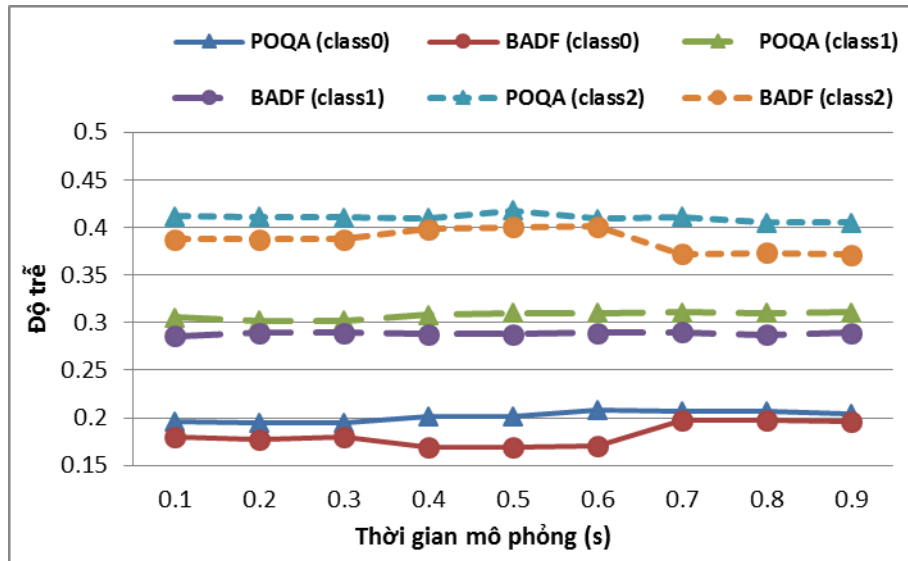


**Hình 2.26** So sánh độ trễ đệm chùm trung bình giữa BADF và POQA trong trường hợp không xem xét đến độ trễ tăng thêm do ước tính sai

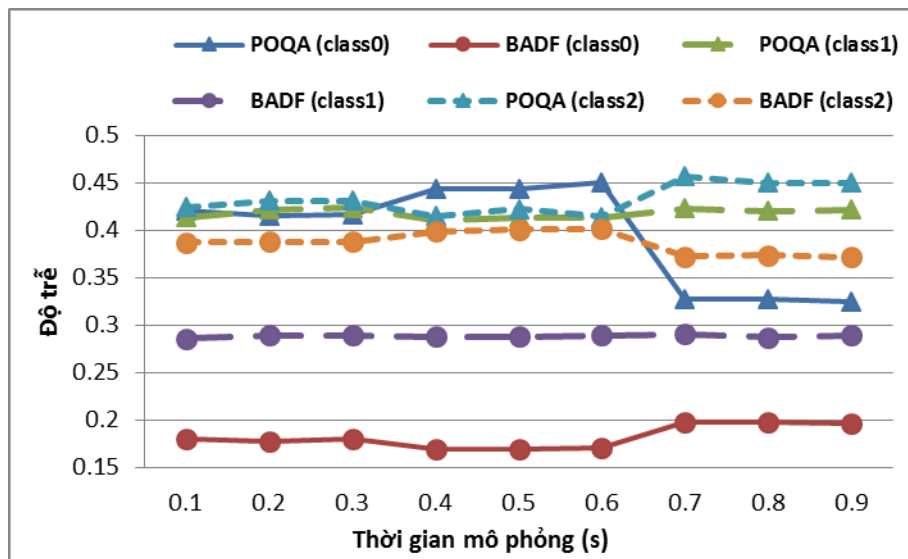


**Hình 2.27** So sánh độ trễ đệm chùm trung bình giữa BADF và POQA trong trường hợp có xem xét đến độ trễ tăng thêm do ước tính sai

Để xem xét rõ hơn trên từng lớp, Hình 2.28 chỉ ra độ trễ đệm chùm của mỗi lớp trong giải thuật BADF luôn thấp hơn so với giải thuật POQA. Tuy nhiên, có một điều cần lưu ý rằng luôn tồn tại một lỗi ước tính đáng kể (xem Mục 2.2.2.5c) trong giải thuật POQA khi tốc độ gói tin đến tăng lên và do đó độ dài chùm hoàn thành lớn hơn so với độ dài chùm ước tính. Các gói tin thừa trong trường hợp này phải chuyển sang chùm tiếp theo và kết quả là chúng phải chịu một độ trễ tăng thêm chính bằng thời gian tập hợp chùm trên hàng đợi đó  $T_a(i)$ . Vì vậy độ trễ đệm chùm trung bình thật sự trong giải thuật POQA là nhiều hơn. Hình 2.29 cũng cho thấy độ trễ đệm chùm trung bình thật sự ở 3 lớp ưu tiên trong giải thuật POQA so với giải thuật BADF.



**Hình 2.28** So sánh độ trễ đệm chùm trung bình của 3 lớp ưu tiên giữa BADF và POQA trong trường hợp không xem xét đến độ trễ tăng thêm

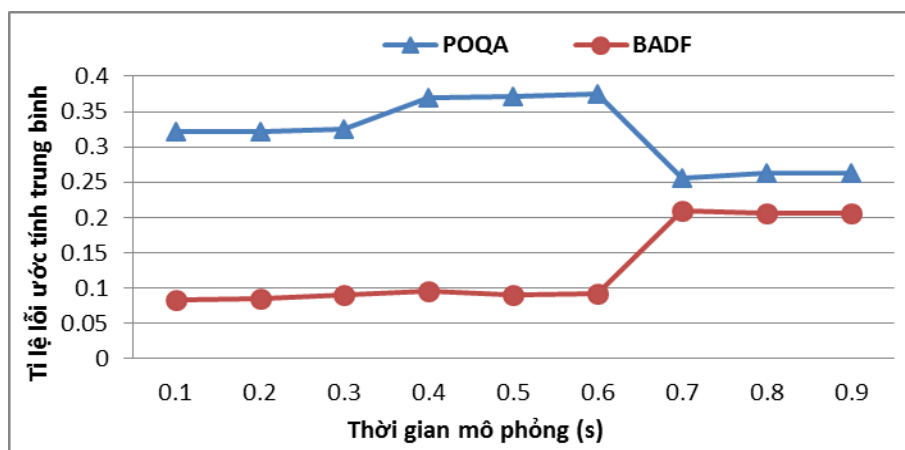


**Hình 2.29** So sánh độ trễ đệm chùm trung bình của 3 lớp ưu tiên giữa BADF và POQA trong trường hợp xem xét đến độ trễ tăng thêm

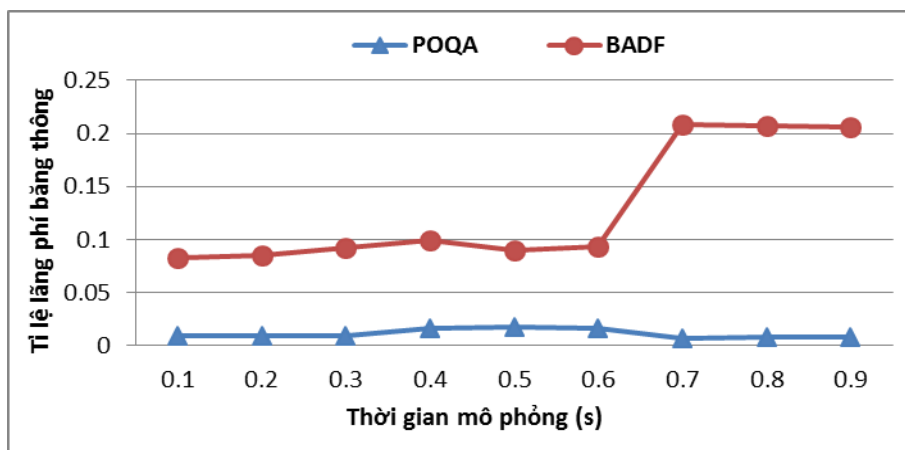
**c. So sánh lỗi ước tính giữa giải thuật BADF với giải thuật POQA**

Hình 2.30 chỉ ra một so sánh về tỉ lệ lỗi ước tính (được tính bởi Công thức 2.8) giữa giải thuật BADF với giải thuật POQA, trong đó lỗi ước tính của BADF là nhỏ hơn nhiều so với POQA. Nguyên nhân của vấn đề này đến từ mô hình tập hợp chùm 2 giai đoạn của giải thuật BADF. Theo đó, một chùm được hoàn thành khi độ dài chùm ước tính hoặc ngưỡng thời gian tập hợp chùm đạt đến. Trong trường hợp độ dài chùm ước tính đạt đến trước, sẽ không xảy ra lỗi ước tính đáng kể. Nhưng nếu

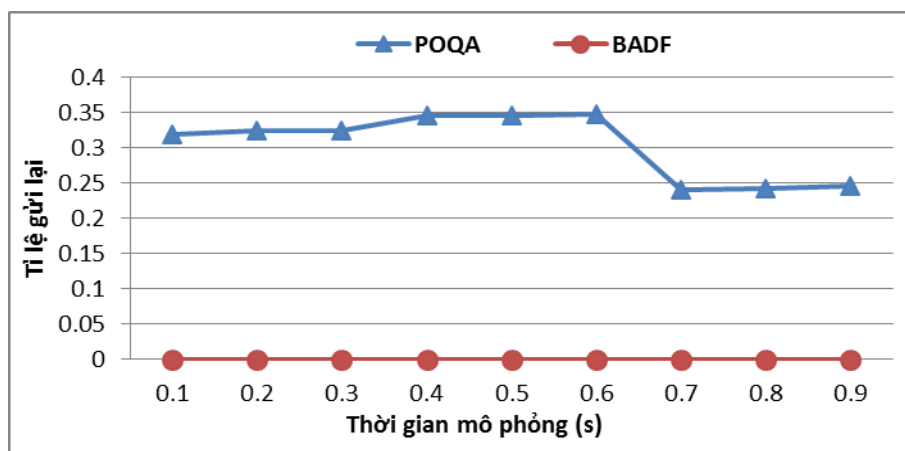
ngưỡng thời gian đạt đến trước, sẽ có một số lỗi ước tính được sinh ra do kích thước thật của chùm bẫy giờ bé hơn kích thước chùm ước tính. Lỗi ước tính này sẽ gây hiện tượng lãng phí băng thông như Hình 2.31, nhưng nó không tạo ra bất kỳ gói tin dư thừa nào như Hình 2.32, nên không có độ trễ tăng thêm nào trong giải thuật BADF.



Hình 2.30 So sánh lỗi ước tính giữa BADF và POQA

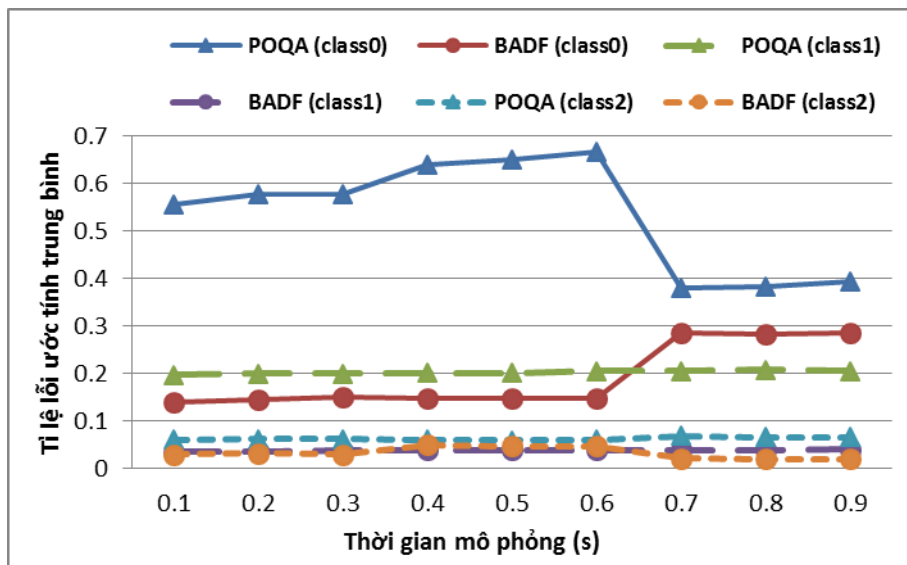


Hình 2.31 So sánh tỉ lệ lãng phí băng thông giữa BADF và POQA



Hình 2.32 So sánh tỉ lệ gửi lại giữa BADF và POQA

So sánh lỗi ước tính trong mỗi lớp ưu tiên giữa giải thuật BADF và giải thuật POQA được thể hiện trong Hình 2.33. Với giải thuật BADF, lỗi ước tính của *class0* tăng lên trong khoảng [0.7, 0.9] bởi vì việc làm giảm tốc độ đến của các gói tin trong lớp này đã làm cho ngưỡng thời gian luôn đạt đến đầu tiên. Với giải thuật POQA, vì là tập hợp chum theo ngưỡng thời gian nên thời gian ở *class0* luôn được đặt cố định, khi mật độ đến của các gói tin trong *class0* tăng (trong khoảng [0.4, 0.6]) hoặc giảm (trong khoảng [0.7, 0.9]) luôn có tăng hoặc giảm lỗi ước tính tương ứng.



Hình 2.33 So sánh lỗi ước tính trên mỗi lớp ưu tiên giữa BADF và POQA

### 2.2.2.6 Nhận xét

Rõ ràng, việc tập hợp chum có tác động đáng kể không chỉ đến độ trễ đầu cuối của các gói tin, mà còn ảnh hưởng đến sự công bằng độ trễ giữa các chum có mức độ ưu tiên khác nhau trong môi trường đa dạng dịch vụ. Giải thuật BADF là hiệu quả trong việc điều khiển công bằng độ trễ giữa các hàng đợi QoS khác nhau, khi so sánh dựa trên chỉ số *DFI*, lỗi ước tính, tỉ lệ gửi lại. Tuy nhiên, tồn tại của giải thuật BADF là tỉ lệ lãng phí băng thông còn lớn trung bình khoảng 12% (như được chỉ ra ở Hình 2.31), nhưng so với tỉ lệ mất phải gửi lại của giải thuật POQA trung bình khoảng 30%, thì giải thuật BADF vẫn tốt hơn. Giải thuật BADF và các kết quả ở trên đã được công bố trong [CT5].



## **2.3 Tiểu kết Chương 2**

Trong chương này, luận án đã đề xuất 2 mô hình tập hợp chòm giảm độ trễ có tên là iBADR [CT2], OBADR [CT3] và một mô hình tập hợp chòm đảm bảo công bằng độ trễ BADF [CT5]. Dựa vào kết quả mô phỏng, giải thuật iBADR và OBADR cho kết quả giảm độ trễ tốt hơn các đề xuất đã được công bố. Giải thuật BADF đã đạt được sự công bằng độ trễ tốt hơn so với POQA, đồng thời giảm độ trễ và giảm thiểu lỗi ước tính trên các hàng đợi.

## CHƯƠNG 3. CÔNG BẰNG THÔNG LƯỢNG DỰA TRÊN CẤP PHÁT BẰNG THÔNG VÀ ĐÁP CHÙM

Để vận chuyển dữ liệu thuộc các lớp dịch vụ (QoS) khác nhau qua mạng OBS, mỗi kết nối (*connection*) chỉ mang một luồng<sup>6</sup> dữ liệu thuộc về một lớp QoS. Các kết nối có cùng đích có thể chia sẻ cùng một liên kết (*link/fiber*) hay cùng một (nhóm) bước sóng của một liên kết ra. Do đó, nếu không có cơ chế cô lập và bảo vệ dịch vụ, các kết nối xấu<sup>7</sup> có thể gửi quá nhiều lưu lượng, mà kết quả là các kết nối tốt<sup>8</sup> phải chịu xác suất mất dữ liệu cao chung của toàn liên kết.

Để giải quyết vấn đề này các tác giả trong [67], [53] đã đề xuất phương pháp cấp phát băng thông công bằng (FBA) bằng cách lấy ý tưởng từ phương pháp công bằng max-min trong mạng IP [16] và ánh xạ thành tỉ lệ mất mát lý thuyết trên mỗi kết nối. Phương pháp FBA này cố gắng duy trì tỉ lệ mất mát thật trên mỗi kết nối về gần với tỉ lệ mất mát lý thuyết. Tuy nhiên do việc sử dụng công thức ErlangB để tính xác suất mất mát lý thuyết nên phương pháp FBA chỉ phù hợp cho các luồng Poisson.

Thực tế, đa số các luồng trong mạng Internet là non-Poisson (chẳng hạn, Self-similar) [2], nên cần có một cách tiếp cận khác mà có thể áp dụng cho nhiều loại luồng khác nhau. Hơn nữa các tác giả trong [67], [53] chưa đưa ra được một thang đo hợp lý cho việc đánh giá hiệu quả công bằng của phương pháp mà họ đề xuất. Chương này sẽ trình bày các đề xuất tiếp theo của luận án, tương ứng với hai mô đun chức năng được bổ sung trong kiến trúc nút biên vào OBS như được mô tả trong Hình 3.1, bao gồm:

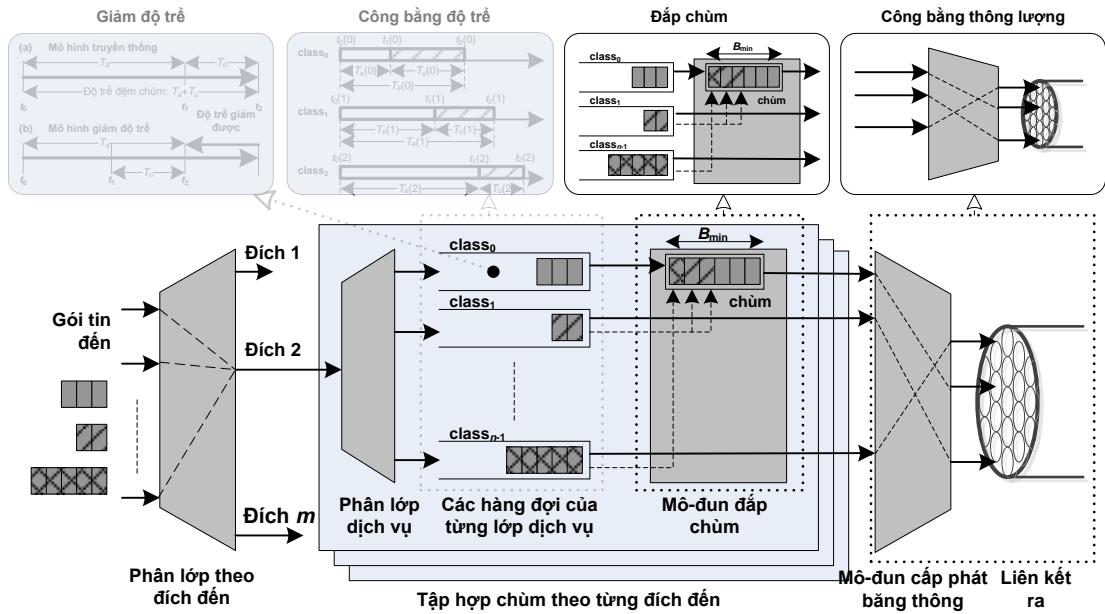
- Mô hình cấp phát băng thông công bằng dựa trên thông lượng (TFBA);
- Mô hình đáp chùm (QDBAP) nhằm tối ưu việc sử dụng băng thông và tăng tính công bằng thông lượng của các kết nối chia sẻ cùng liên kết.

---

<sup>6</sup> Trong luận án này khái niệm luồng và kết nối là có thể thay thế cho nhau.

<sup>7</sup> Kết nối xấu là kết nối có thông lượng vượt quá băng thông cấp phát cho nó.

<sup>8</sup> Kết nối tốt là kết nối có thông lượng không vượt quá băng thông cấp phát cho nó.

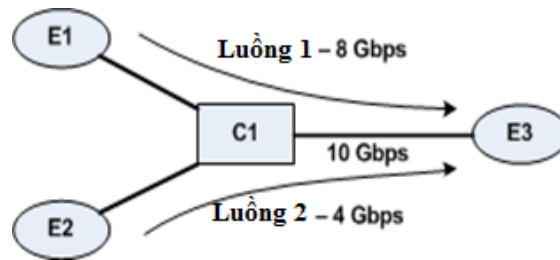


**Hình 3.1 Hai mô-đun chức năng điều khiển công bằng: mô-đun đáp chùm và mô-đun công bằng thông lượng, được bổ sung trong kiến trúc nút biên vào OBS**

### 3.1 Mô hình cấp phát băng thông công bằng dựa trên thông lượng

#### 3.1.1 Giới thiệu về cấp phát băng thông công bằng

Cấp phát băng thông công bằng, còn được gọi là công bằng tốc độ (*rate fairness*), đề cập đến việc cấp phát băng thông cho các kết nối theo tỷ lệ của băng thông cung cấp với băng thông khả dụng [53].



**Hình 3.2 Ví dụ về cấp phát băng thông công bằng của 2 luồng chia sẻ cùng một liên kết**

Như ví dụ trong Hình 3.2, giả sử có 2 luồng dữ liệu chia sẻ cùng một liên kết chung, gồm Luồng 1 từ E1 đến E3 và Luồng 2 từ E2 đến E3; Luồng 1 yêu cầu băng thông 8 Gbps, trong khi Luồng 2 muốn băng thông 4 Gbps. Theo phương pháp cấp phát băng thông truyền thống, Luồng 1 và Luồng 2 đều được cấp phát cùng một lượng băng thông 5 Gbps. Cách phân bổ này là không công bằng, vì Luồng 2 không bao giờ sử dụng hết băng thông trong khi Luồng 1 luôn thiếu băng thông. Để giải quyết vấn đề này, đã có một số phương pháp được đề xuất mà sẽ được phân tích ở

phần sau.

### 3.1.2 Các công trình nghiên cứu liên quan

Cho đến nay, các mô hình cấp phát băng thông công bằng trong mạng OBS đều dựa trên mô hình phân bổ băng thông công bằng max-min trong mạng IP [16]. Thực tế, vấn đề cấp phát băng thông công bằng max-min trong mạng IP chỉ có hiệu quả khi tổng yêu cầu của các cá nhân lớn hơn khả năng của liên kết. Chính sách công bằng max-min có thể tóm tắt như sau: (1) băng thông được phân bổ theo thứ tự tăng về nhu cầu; (2) băng thông được phân bổ không bao giờ được lớn hơn nhu cầu; và (3) tất cả các yêu cầu không thỏa mãn sẽ được phân bổ đều ở lượng băng thông còn thừa. Theo chính sách này, ban đầu băng thông được phân bổ cho nhu cầu nhỏ nhất; các nhu cầu còn lại không thỏa mãn sẽ được chia đều lượng băng thông còn thừa. Như ví dụ trong Hình 3.2, ban đầu Luồng 1 và Luồng 2 sẽ được cung cấp một lượng băng thông đúng với nhu cầu của Luồng 2 là 4 Gbps. Sau đó lượng băng thông thừa 2 Gbps được chia đều cho các luồng không thỏa mãn, ở đây chỉ có Luồng 1 nên 2 Gbps được chia cho nó. Kết quả là, Luồng 1 được cấp phát 6 Gbps và luồng 2 được cấp phát 4 Gbps.

Có một sự khác biệt quan trọng trong mạng OBS là không có bộ đệm như trong mạng IP. Cụ thể, vấn đề cấp phát băng thông công bằng trong mạng IP có thể đạt được thông qua các thuật toán phân bổ công bằng, trong đó nếu tổng băng thông yêu cầu vượt quá khả năng của liên kết thì các gói tin sẽ được lưu tạm tại các bộ đệm và lại được phục vụ sau đó khi hệ thống rỗi. Do đó, các thuật toán hàng đợi công bằng có thể khai thác hết khả năng của liên kết. Tuy nhiên, băng thông không thể được sử dụng triệt để trong mạng OBS vì luôn tồn tại một số khoảng trống không thể tránh khỏi được sinh ra giữa các chùm được lập lịch trên các kênh ra. Do những khác biệt lớn này, các thuật toán hàng đợi công bằng cho mạng IP không thể áp dụng trực tiếp vào mạng OBS để đạt được cùng công bằng max-min.

Để khắc phục điều đó, phương pháp MMFP trong [67] đã chuyển đổi việc cấp phát băng thông công bằng max-min ( $F_i$ ) thành tỉ lệ mất công bằng ( $P_i$ ) của mỗi kết nối. Tải hiệu quả ( $E_i$ ) cho mỗi kết nối được định nghĩa là:

$$E_i = \sum_{j=1}^N \min\{F_i, F_j\} \quad (3.1)$$

trong đó  $N$  là số lượng các kết nối chia sẻ chung một liên kết,  $i$  và  $j$  là chỉ số về kết nối  $1 \leq i, j \leq N$ .  $F_i$  là một đại lượng xác định tỉ lệ mất công bằng trong thực tế.

Trong trường hợp tốc độ đến của các kết nối nhỏ hơn so với mức công bằng (các kết nối chưa vượt quá tỉ lệ cung cấp), tỉ lệ mất được xác định bởi

$$P_i = Er(E_i) = \frac{(E_i)^W / W!}{\sum_{k=1}^W (E_i)^k / k!} \quad (3.2)$$

trong đó  $W$  là số bước sóng trên liên kết ra.

Tuy nhiên, nếu tốc độ đến của các kết nối lớn hơn so với tỉ lệ công bằng (các kết nối vượt quá tỉ lệ cung cấp), tỉ lệ mất được xác định bởi

$$P_i = \frac{A_i - F_i \times (1 - Er(E_i))}{A_i} \quad (3.3)$$

trong đó  $A_i$  là tốc độ đến thực tế của kết nối  $i$ .

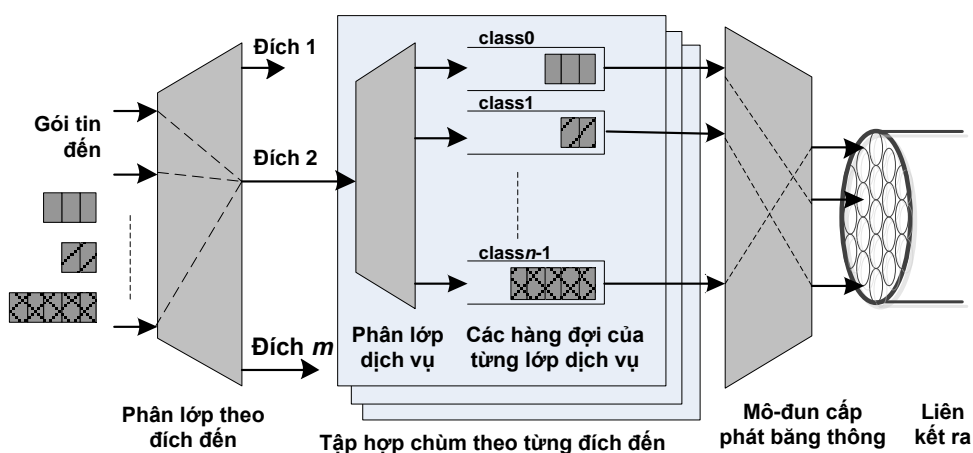
Liu và cộng sự trong [67] cố gắng duy trì tỉ lệ mất thực tế về gần với tỉ lệ mất lý thuyết ( $P_i$ ) này. Tuy nhiên, cách duy trì tỉ lệ mất thực tế về sát với mức lý thuyết của MMFP có thể dẫn đến hạn chế lưu lượng vào và sử dụng băng thông dư thừa không hiệu quả. Orawiwattanakul và cộng sự trong [53] đã đề xuất phương pháp RFP mà cũng dựa trên công bằng max-min nhằm phân bổ băng thông công bằng đối với tất cả các luồng và đồng thời xử lý tranh chấp công bằng giữa các chùm. Cụ thể trong trường hợp có tranh chấp, RFP cho phép các chùm thuộc luồng tốt chiếm kênh của luồng xấu. Các chùm thuộc luồng xấu chỉ được ưu tiên khi tất cả các bước sóng đều bận (tức là tất cả các kênh đều đang được khai thác và vẫn còn băng thông nhàn rỗi) để tận dụng băng thông dư thừa. Hơn nữa, chỉ có các nút biên theo dõi tốc độ các chùm đến và các nút lõi thực hiện phân bổ băng thông dựa trên RFP chỉ khi tốc độ chùm đến thay đổi đáng kể. Vì vậy, RFP không gây ra tải nặng trong mạng lõi. Tuy nhiên, 2 gói điều khiển FBCP (*Forward BCP*) và BBCP (*Back BCP*) luôn được duy trì trong RFP để trao đổi thông tin từ nguồn và đích đã làm tăng đáng kể độ phức tạp của hệ thống và chiếm dụng băng thông cho việc trao đổi thông tin.

Tóm lại, vẫn tồn tại các hạn chế đối với 2 mô hình cấp phát băng thông công bằng trên trong đó MMFP chưa khai thác hiệu quả băng thông dư thừa, trong khi RFP phải luôn duy trì 2 gói điều khiển FBCP và BBCP, mà điều này tiêu tốn một lượng băng thông đáng kể cho việc trao đổi. Ngoài ra, RFP cần duy trì mô hình dự báo tốc độ lưu lượng đến của các luồng, nhưng không phải khi nào cũng thu được dự báo chính xác. Cả 2 mô hình này đều sử dụng công thức ErlangB để tính xác suất mất mát mà nó chỉ phù hợp với các luồng Poisson. Các mô hình trên đều hướng đến vấn đề công bằng thông lượng nhưng chưa đưa ra được thang đo hợp lý để đánh giá tính công bằng. Mô hình cấp phát băng thông công bằng được đề xuất sau đây của luận án sẽ khắc phục được các hạn chế này.

### 3.1.3 Phương pháp cấp phát băng thông công bằng dựa trên thông lượng TFBA

#### 3.1.3.1 Kiến trúc nút biên vào hỗ trợ đa dạng dịch vụ

Xét một nút biên vào với kiến trúc như được chỉ ra ở Hình 3.3.



**Hình 3.3 Kiến trúc nút biên vào OBS hỗ trợ đa dạng dịch vụ**

Các gói tin đến đầu tiên sẽ được phân lớp theo cùng đích đến; với mỗi đích đến các gói tin tiếp đó sẽ được phân lớp dựa trên các lớp dịch vụ vào các hàng đợi khác nhau. Khi một ngưỡng tập hợp chùm (ngưỡng thời gian hoặc ngưỡng độ dài) đạt đến, các gói tin trong các hàng đợi này được tập hợp thành một chùm. Tập các chùm thuộc về cùng một lớp dịch vụ tạo thành một luồng dữ liệu tách rời tại liên kết ra của một nút biên vào OBS.

Các luồng tách rời có thể chia sẻ cùng một liên kết ra chung. Thông thường, thông lượng của mỗi luồng bị giới hạn bởi băng thông được cung cấp cho chúng.

Nhưng do một số luồng không sử dụng hết băng thông được cấp phát, băng thông dư thừa sẽ được cấp phát cho các luồng thiếu băng thông. Việc cấp phát băng thông sẽ được thực hiện lại nếu có biến động đáng kể về tốc độ của các luồng đến.

### 3.1.3.2 Tỷ lệ băng thông sử dụng tối đa của mỗi liên kết trong mạng OBS

Trong mô hình cấp phát băng thông công bằng MMFP [67], khả năng sử dụng băng thông của mỗi liên kết được cho là  $C = 1$ ; tuy nhiên điều này là không đúng vì luôn tồn tại các khoảng trống giữa các chùm được lập lịch trên các kênh ra trong mạng OBS. Do đó Orawiwartanakul và cộng sự trong [53] đã bổ sung một hệ số  $\omega = 0.7$  thể hiện mức băng thông có thể được sử dụng tối đa của mỗi liên kết. Giá trị hệ số này cũng được chứng minh bằng kết quả mô phỏng của luận án được chỉ ra trong Bảng 3.1. Với tải chuẩn hóa đến thay đổi từ 0.5 đến 1.0, thông lượng đạt được tối đa trên một liên kết trung bình là 0.72. Do vậy, hệ số  $\omega = 0.7$  tiếp tục được sử dụng cho các mô phỏng sau này của luận án.

**Bảng 3.1 Tỷ lệ thông lượng đạt được tối đa trên mỗi liên kết với tải chuẩn hóa khác nhau**

Tải chuẩn hóa	0.5	0.6	0.7	0.8	0.9	1.0
Thông lượng tối đa	0.48616	0.572186	0.67152	0.72123	0.7213	0.71945

### 3.1.3.3 Phương pháp cấp phát băng thông công bằng dựa trên thông lượng TFBA

Ý tưởng của phương pháp cấp phát băng thông công bằng được luận án đề xuất TFBA (*Throughput based Fair Bandwidth Allocation*) là điều chỉnh thông lượng thực tế về gần với băng thông cấp phát công bằng nhằm đảm bảo sự công bằng trong việc cấp phát băng thông giữa các luồng. Khi một chùm đến không thể lập lịch ở trên các kênh ra của liên kết ra, hệ thống giả định rằng tổng nhu cầu của luồng vượt quá băng thông của các kết nối ra, nên một yêu cầu tái phân bổ băng thông được kích hoạt. Quá trình phân bổ băng thông công bằng dựa trên thông lượng được thực hiện theo 4 bước:

*Bước 1: Xác định tỉ lệ công bằng  $F_i$  cho mỗi kết nối*

Nếu tốc độ đến của luồng  $i$  có thay đổi đáng kể, băng thông đầu tiên được chia đều cho các kết nối, tỉ lệ  $F_i$  cho mỗi kết nối được xác định là mức tối thiểu giữa thông lượng thật ( $A_i$ ) với băng thông phân bổ công bằng. Kết nối có thông lượng thực tế ít

hơn so với lượng băng thông được phân bổ sẽ không tham gia chia sẻ băng thông thừa trong lần lặp tiếp theo. Việc phân bổ được tiếp tục cho đến khi băng thông được phân bổ ( $m$ ) không có thay đổi so với lần lặp trước ( $m = m_{prev}$ ) hoặc tất cả các kết nối đều được thỏa mãn ( $m = 0$ ).

*Bước 2: Xác định băng thông phân bổ công bằng  $AB_i$  cho mỗi kết nối*

Đặt  $B_w$  là băng thông tối đa ở liên kết ra, băng thông phân bổ công bằng cho kết nối  $i$  được xác định theo Công thức 3.4

$$AB_i = F_i \times B_w \quad (3.4)$$

trong đó  $F_i$  là tỉ lệ công bằng như được xác định ở Bước 1.

*Bước 3: Đo giá trị thông lượng thực tế  $AT_i$  của mỗi kết nối*

Thông lượng thực tế được xác định theo Công thức 3.5

$$AT_i = p_w(i) / T_w(i) \quad (3.5)$$

trong đó  $p_w(i)$  là lượng dữ liệu đến trong cửa sổ thời gian  $T_w(i)$ .

*Bước 4: Xử lý tranh chấp chùng*

Vấn đề tranh chấp chùng được giải quyết dựa trên việc so sánh giữa  $AT_i$  và  $AB_i$  nhằm xác định xem chùng đến có thuộc luồng xấu (luồng quá tải) hay không. Theo đó nếu  $AT_i > AB_i$  chùng đến thuộc luồng xấu, nó sẽ bị loại bỏ để dành tài nguyên cho các chùng thuộc luồng tốt (luồng không quá tải). Ngược lại, nếu  $AT_i < AB_i$  chùng đến thuộc luồng tốt, việc xem xét tỷ lệ  $AT_i / AB_i$  sẽ được tính đến, nếu giá trị  $AT_i / AB_i$  thấp hơn giá trị  $AT_j / AB_j$  của chùng tranh chấp, chùng tranh chấp sẽ bị đánh rơi. Ngược lại, nếu giá trị  $AT_i / AB_i$  lớn hơn chùng tranh chấp  $AT_j / AB_j$  chùng đến sẽ bị loại bỏ.

Giải thuật cấp phát băng thông công bằng TFBA được mô tả như sau:

#### **Giải thuật 4: TFBA**

##### **Input:**

- $b_i$ ; // chùng đến  $i, i=1, 2, \dots, n.$
- $(\lambda'_i, \lambda_i)$ ; // tốc độ trước và hiện tại của luồng  $i$

##### **Output:** $L = \{lost_i\}$

// tập byte bị mất ở các kết nối

##### **Begin**

- 1  $\omega := 0.7; C := 1.0; B_w := 1GB; th := 0.1;$
- 2 **if** ( $b_i$  không thể lập lịch) **then**
- 3     **if** ( $|\lambda_i - \lambda'_i| > th$ ) **then** // nếu tốc độ đến có sự thay đổi



```

4       $S := \emptyset; m := N;$            //  $N$  là số lớp QoS
5       $U := \omega \times C;$            // khả năng thật sự của liên kết ra
6      repeat                         // cấp phát băng thông công bằng
7           $FS := U / m;$ 
8           $m_{prev} := m;$ 
9           $F_l := \min\{\lambda_l, FS\}, l \notin S;$  // tỉ lệ công bằng của kết nối  $l$ 
10          $S := \{j: \lambda_j \leq FS\};$  // tập các kết nối thuộc phần luồng tốt
11          $U := U - \sum_{j \in S} \lambda_j;$  // băng thông dư thừa
12          $m := m - |S|;$            // số lượng kết nối chia sẽ băng thông
                                           // dư thừa ở lần lặp sau
13     until ( $m = m_{prev}$  or  $m = 0$ );
14     end if
15          $AB_i := F_i \times Bw;$  // băng thông cấp phát cho kết nối  $i$ 
16          $AT_i := p_w(i) / T_w(i);$  // băng thông thực tế, ở đây  $p_w(i)$  là số
                                           // byte đến trong cửa sổ thời gian  $T_w(i)$ 
17     if ( $AT_i > AB_i$ ) then
18          $lost_i := lost_i + b_i;$ 
19     Else
20         if ( $AT_i / AB_i < AT_j / AB_j$ ) then // trường hợp  $y_i < y_j$ 
21              $lost_j := lost_j + b_j;$ 
22         Else
23              $lost_i := lost_i + b_i;$ 
24         end if
25     end if
End

```

Độ phức tạp tính toán thời gian của giải thuật TFBA chủ yếu thực hiện ở quá trình cấp phát băng thông **repeat** – **until** (dòng 6 đến 13). Trong vòng lặp **repeat** – **until** có 3 vòng lặp khác, vòng lặp thứ 1 là phân bổ tỉ lệ công bằng  $F_l$  ở dòng 9, vòng lặp thứ 2 là xác định tập các kết nối thuộc phần luồng tốt ở dòng 10 và vòng lặp thứ 3 là tính băng thông dư thừa ở dòng 11. Tuy nhiên, 3 vòng lặp này là độc lập với nhau và có độ phức tạp là  $O(N)$  với  $N$  là số kết nối. Trong vòng lặp **repeat** – **until** trường hợp xấu nhất là duyệt qua  $N$  lần, nên độ phức tạp tính toán của giải thuật là  $O(N^2)$ . Như vậy độ phức tạp tính toán của giải thuật TFBA là tương đương với độ phức tạp của 2 giải thuật MMFP và RFP.

### 3.1.3.4 Chỉ số công bằng thông lượng

Cấp phát băng thông công bằng trong tiếp cận của luận án cũng dựa trên ý tưởng của công bằng max-min. Tuy nhiên công bằng được dựa trên tỉ lệ của thông lượng thực tế với băng thông được cung cấp, thay vì các xác suất mất mát như trong [67], [53]. Cụ thể, đặt  $y_i = AT_i / AB_i$  là tỉ lệ của thông lượng thực tế ( $AT_i$ ) và băng thông cấp phát công bằng ( $AB_i$ ) của luồng  $i$ . Dựa trên công thức của Jain trong [39], luận án đề xuất chỉ số công bằng thông lượng  $TFI$  như sau:

$$TFI = \frac{\left(\sum_{i=1}^n \sigma_i y_i\right)^2}{n \sum_{i=1}^n (\sigma_i y_i)^2} \quad (3.6)$$

trong đó  $\sigma_i$  là trọng số thể hiện mức độ sử dụng băng thông so với mức được cung cấp giữa các luồng, ở đây  $0 < \sigma_i < 1$  và  $\sum_{i=1}^n \sigma_i = 1$ .

Mục tiêu của cấp phát băng thông công bằng là điều chỉnh thông lượng thực tế ( $AT_i$ ) của mỗi luồng về gần với băng thông cấp phát công bằng ( $AB_i$ ) của nó. Tức là đẩy  $TFI$  về 1. Các tác giả trong [39] cũng đã chứng minh rằng, chỉ số bằng 1 khi:

$$\sigma_1 y_1 = \sigma_2 y_2 = \dots = \sigma_n y_n \quad (3.7)$$

Với công thức của Jain trong [39] thì giá trị  $y_i$  thông thường là độ trễ, thông lượng hay tỉ lệ mất dữ liệu. Tuy nhiên, việc dựa trên giá trị thông lượng của các hàng đợi sẽ không phản ánh được mối tương quan tỉ lệ giữa thông lượng với lượng băng thông được cung cấp của các kết nối. Do vậy, luận án đề xuất đại lượng  $y_i = AT_i / AB_i$  nhằm điều chỉnh công bằng thông lượng. Luận án cũng bổ sung trọng số  $\sigma_i$  vào công thức  $TFI$  để này thể hiện mức độ ưu tiên đối với các kết nối;  $\sigma_i$  càng lớn thì mức độ ưu tiên càng cao, nhưng phải đảm bảo ràng buộc  $0 < \sigma_i < 1$  và  $\sum_{i=1}^n \sigma_i = 1$ .

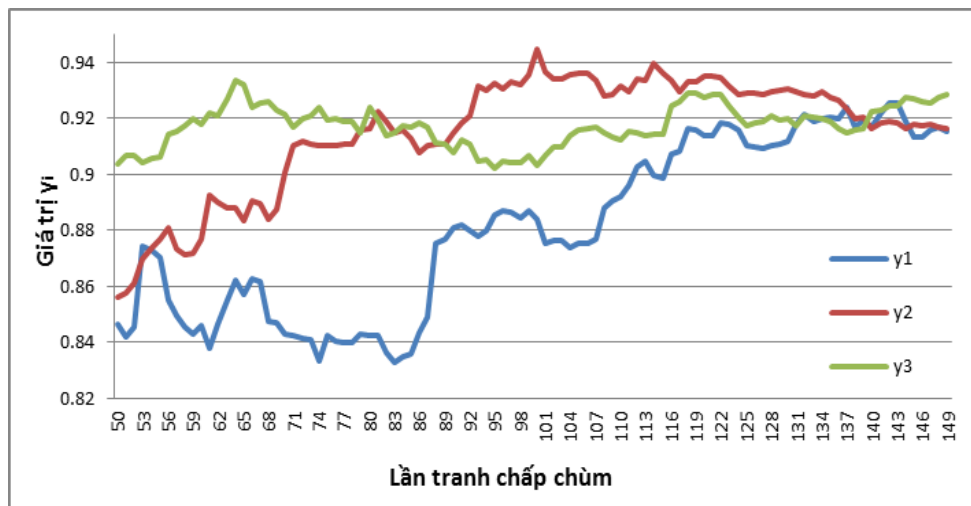
Để cho đơn giản trong quá trình xử lý và cài đặt, luận án thiết lập các giá trị  $\sigma_i$  bằng nhau. Do đó, Công thức 3.6 được viết lại là:

$$TFI = \frac{\left(\sum_{i=1}^n y_i\right)^2}{n \sum_{i=1}^n (y_i)^2} \quad (3.8)$$

Công bằng tăng khi  $TFI$  tiến đến 1 và bằng 1 khi:  $y_1 = y_2 = \dots = y_n$ .

### 3.1.3.5 Phân tích ảnh hưởng của TFBA đến công bằng thông lượng

Hiệu quả công bằng thông lượng của TFBA chủ yếu dựa trên việc phân bổ lại băng thông (dòng 15 của thuật toán TFBA) và quá trình xử lý tranh chấp chòm (dòng 7 đến dòng 25). Cụ thể, khi có hiện tượng tranh chấp chòm (dòng 2), tính công bằng của mỗi kết nối ( $F_i$ ) được tính toán lại (dòng 6 đến dòng 13), băng thông của mỗi kết nối ( $AB_i$ ) được phân bổ lại (dòng 15) và sau đó là lựa chọn chòm bị đánh rơi. Một điều có thể nhận thấy rằng quá trình lựa chọn các chòm bị đánh rơi sẽ góp phần đẩy thông lượng thực tế càng về gần với lượng băng thông được cung cấp. Nếu chòm đến thuộc luồng xấu (dòng 17) nó sẽ bị đánh rơi và thông lượng thực tế của luồng này giảm đi về gần với lượng băng thông được phân bổ ( $y_i \rightarrow 1$ ). Trong trường hợp chòm đến thuộc phần luồng tốt (dòng 19), việc so sánh tỉ lệ của thông lượng thực với lượng băng thông được cấp phát giữa các luồng ( $y_i$  và  $y_j$ ) được tính đến: nếu  $y_i < y_j$  chòm  $j$  bị đánh rơi, ngược lại  $y_i > y_j$  chòm  $i$  bị đánh rơi. Rõ ràng, quá trình đánh rơi chòm sẽ đẩy  $y_j$  về gần với  $y_i$  hoặc  $y_i$  gần hơn với  $y_j$ . Theo Công thức 3.6 và 3.7, điều này sẽ đẩy  $TFI$  tiến về 1 và do đó tăng hiệu quả công bằng thông lượng. Kết quả mô phỏng được chỉ ra ở Hình 3.4 cho thấy rằng các giá trị của  $y_1$ ,  $y_2$  và  $y_3$  hội tụ dần qua quá trình xử lý tranh chấp chòm.



Hình 3.4 Sự hội tụ của  $y_1$ ,  $y_2$  và  $y_3$  qua quá trình xử lý tranh chấp chòm

### 3.1.3.6 So sánh và đánh giá dựa trên kết quả mô phỏng

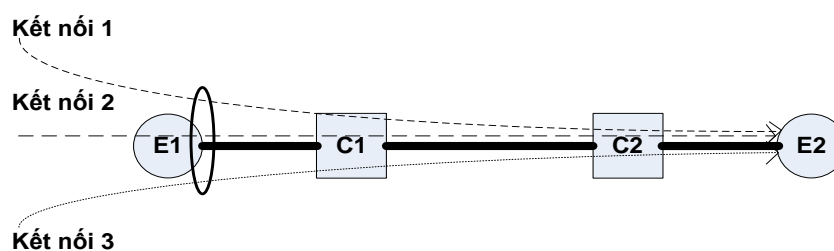
Không mất tính tổng quát, có thể xem xét các gói tin đến tại các hàng đợi của nút biên OBS có phân bố Poisson và có kích thước nằm trong khoảng [500, 1000]

bytes. Phương pháp tập hợp lại là được triển khai với ngưỡng độ dài là 20000 bytes và ngưỡng thời gian là  $10\mu s$ . Số bước sóng trên mỗi liên kết ra là  $W = 8$ , số kết nối  $K = 3$ . Băng thông của mỗi liên kết là 1Gbps, nhưng mức độ sử dụng băng thông thực tế của nó là  $\omega = 0.7$ . Mô phỏng được thực hiện trong khoảng thời gian 1.0s trong đó, (1) từ 0 đến 0.5 s tải chuẩn hóa đến của 3 kết nối 1, 2, và 3 lần lượt là 0.1, 0.2, và 0.3 (trường hợp tổng tải không vượt quá khả năng đáp ứng của băng thông); và (2) từ 0.5 đến 1.0 s tải chuẩn hóa luồng 3 tăng lên 0.6, trong khi tải chuẩn hóa của luồng 1 và 2 không thay đổi (trường hợp mà tổng tải vượt quá khả năng đáp ứng của băng thông).

Với mục tiêu mô phỏng là:

- So sánh tỉ lệ mất byte giữa các kết nối và tỉ lệ mất byte trung bình giữa TFBA, MMFP và RFP.
- So sánh tính công bằng dựa trên chỉ số *TFI* giữa TFBA, MMFP và RFP.

Với mục tiêu mô phỏng chỉ xem xét tỉ lệ mất byte của các kết nối khi chia sẻ chung một (hay một nhóm) liên kết ra nên mạng Dumbbell được chọn như Hình 3.5 là đủ để đánh giá giải thuật đề xuất. Môi trường mô phỏng là NS2 [71] với gói hỗ trợ mạng OBS obs-0.9a. Các mô phỏng được triển khai trên một máy tính PC với cấu hình 2.4 GHz Intel Core 2 CPU, 2G RAM.

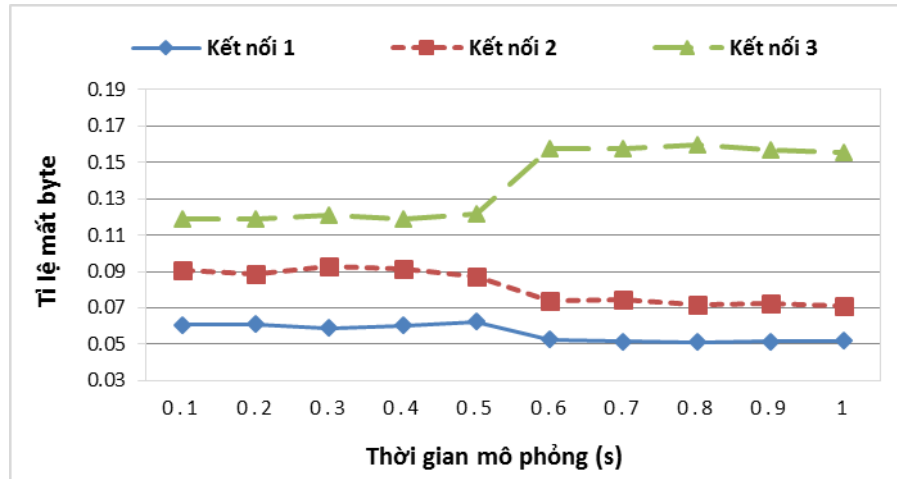


**Hình 3.5 Hình thái mạng mô phỏng**

#### **a. So sánh tỉ lệ mất byte giữa các kết nối**

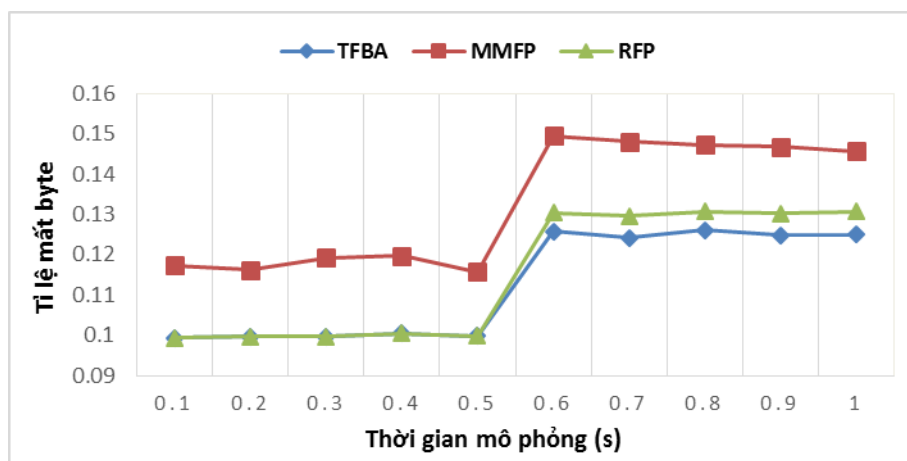
Hình 3.6 mô tả tỉ lệ mất byte giữa 3 kết nối của phương pháp cấp phát băng thông công bằng TFBA trong 2 trường hợp: (1) khi tổng tải 3 luồng đến chưa vượt quá băng thông liên kết (từ 0.0 đến 0.5), và (2) khi luồng 3 có tải tăng đột biến làm tổng tải các luồng đến vượt quá khả năng băng thông liên kết (từ 0.5 đến 1.0). Như thể hiện trong Hình 3.6, trước 0.5s tỉ lệ mất byte giữa các luồng giữ ở mức công bằng, tỉ lệ thuận với thông lượng đến thực tế. Tuy nhiên từ 0.5 s đến 1.0 s, do tải

luồng 3 tăng đột ngột làm tổng tải vượt quá khả năng băng thông của liên kết nên tỉ lệ mất byte của luồng này phải chịu tăng lên vì nó được xem là luồng xấu. Việc tăng đánh rơi các chùm thuộc luồng 3 đã sinh ra nhiều băng thông rỗi hơn, nên tăng khả năng lập lịch cho luồng 1 và 2; kết quả là tỉ lệ mất byte của luồng 1 và 2 có giảm nhẹ.



**Hình 3.6 So sánh tỉ lệ mất byte giữa 3 kết nối của phương pháp TFBA trong 2 trường hợp (1) tổng tải không vượt quá khả năng liên kết và (2) tải luồng 3 tăng đột biến vượt quá khả năng liên kết**

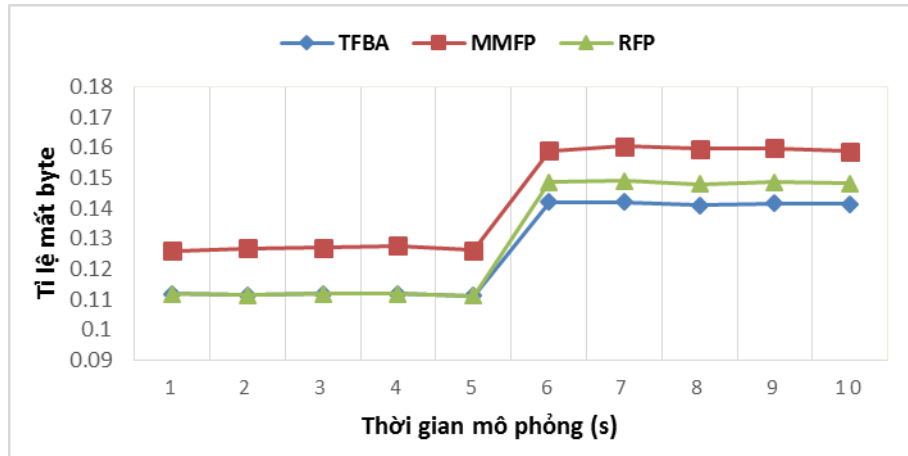
So sánh với RFP và MMFP, TFBA cho hiệu quả tốt nhất về tỉ lệ mất byte như được chỉ ra trong Hình 3.7. Cụ thể, với các luồng đến có thông lượng vượt quá khả năng liên kết (từ 0.5 đến 1.0s), TFBA không chỉ điều phối cấp phát băng thông công bằng giữa các luồng mà còn tối thiểu được dữ liệu mất mát.



**Hình 3.7 So sánh tỉ lệ mất byte trung bình trên cả 3 kết nối của TFBA, RFP và MMFP**

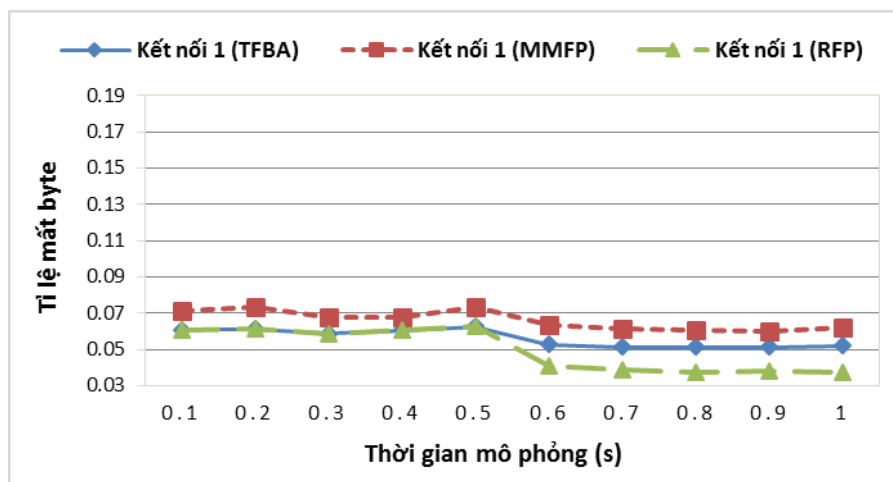
Luận án cũng tiến hành tăng thời gian mô phỏng lên 10s, trong đó từ 0 đến 5s có tải chuẩn hóa ở cả 3 luồng đến lần lượt là 0.1, 0.2 và 0.3 và từ 6 đến 10s có tải

chuẩn hóa luồng 3 tăng lên 0.6 trong khi luồng 1 và 2 không thay đổi. Hình 3.8 chỉ ra rằng kết quả mô phỏng trong 10s là tương đồng với kết quả mô phỏng trong Hình 3.7 (thời gian mô phỏng 1s). Thực tế, với các tham số mô phỏng được thiết lập trong Mục 3.1.3.6, lượng các gói tin đến tại các hàng đợi ở nút biên, cũng như số chùm sinh tại liên kết ra trong 1 giây mô phỏng là rất lớn. Điều này đã đảm bảo độ tin cậy về mặt thống kê đối với các kết quả mô phỏng được.

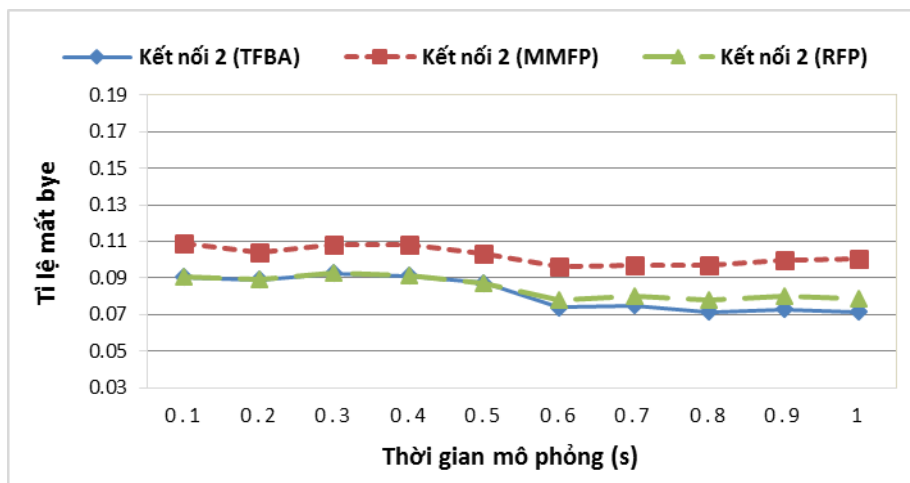


**Hình 3.8 So sánh tỉ lệ mất byte trung bình trên cả 3 kết nối giữa TFBA, RFP và MMFP với thời gian mô phỏng tăng lên 10s**

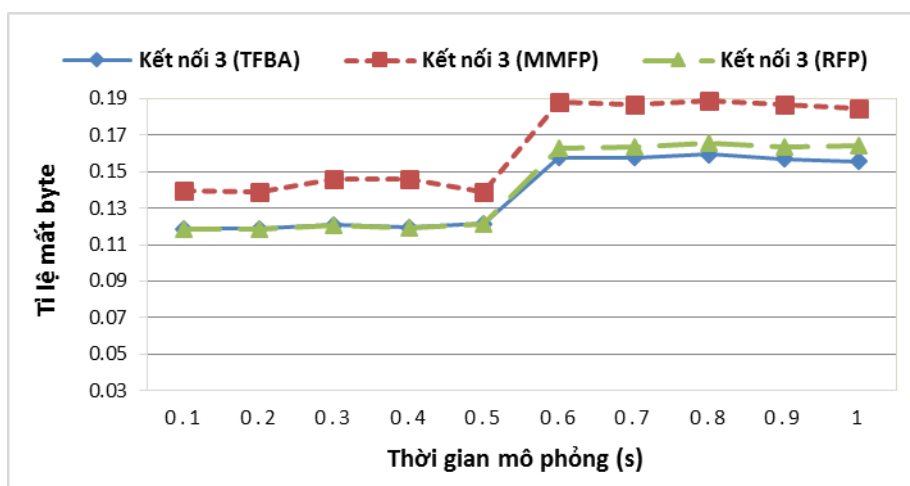
Khi so sánh đối với từng kết nối, Hình 3.10 và Hình 3.11 cho thấy rằng Kết nối 2 và Kết nối 3 của mô hình cấp phát băng thông TFBA có tỉ lệ mất byte thấp nhất khi so sánh với RFP và MMFP, nhưng trong Hình 3.9, Kết nối 1 của mô hình TFBA lại cao hơn so với mô hình RFP. Nguyên nhân là do mô hình RFP luôn ưu tiên cho các luồng có lưu lượng thấp, mà điều này thực tế là không công bằng.



**Hình 3.9 So sánh tỉ lệ mất byte của Kết nối 1 giữa TFBA, RFP và MMFP**



Hình 3.10 So sánh tỉ lệ mất byte của Kết nối 2 giữa TFBA, RFP và MMFP



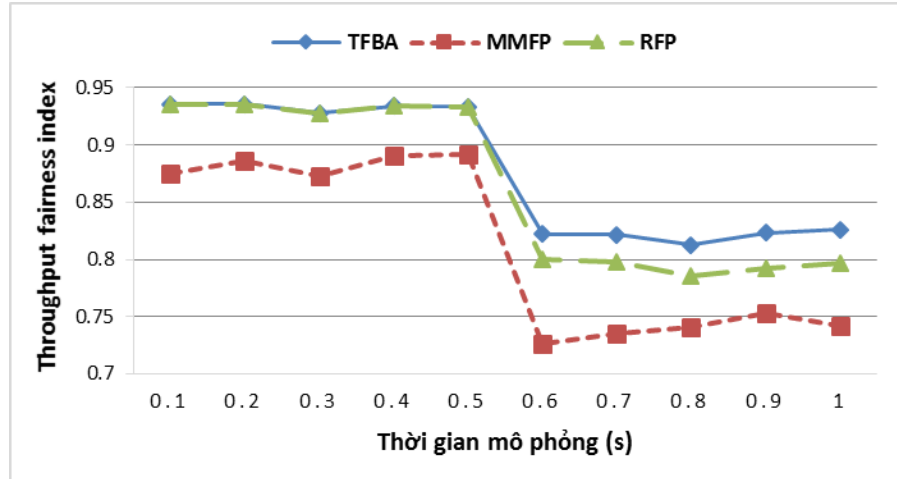
Hình 3.11 So sánh tỉ lệ mất byte của Kết nối 3 giữa TFBA, RFP và MMFP

Mô hình TFBA thể hiện tính công bằng hơn (được chỉ ra trong Hình 3.12 ở mục tiếp theo), và cũng tối thiểu được tỉ lệ mất trung bình đối với tất cả các luồng như trong Hình 3.7 và Hình 3.8.

### b. So sánh hiệu quả công bằng dựa trên chỉ số *TFI*

Để đánh giá hiệu quả của các mô hình cấp phát băng thông công bằng, luận án đề xuất phương pháp đo dựa trên chỉ số *TFI* nhằm so sánh đánh giá giữa TFBA với RFP và MMFP. Kết quả trong Hình 3.12 cho thấy rằng khi tổng tải các luồng đến chưa vượt quá khả năng băng thông của liên kết, sự công bằng (thông qua *TFI*) trong cấp phát băng thông giữa các luồng đạt mức cao nhất (xấp xỉ 1), nhưng khi tổng tải của các luồng vượt quá khả năng của liên kết, các mô hình cấp phát băng thông đều thực hiện cấp phát lại băng thông và vấn đề công bằng giữa các luồng bị giảm xuống.

Tuy nhiên trong cả 2 trường hợp (trước và sau 0.5s), mô hình TFBA luôn đạt được *TFI* tốt nhất, thể hiện sự công bằng trong cấp phát băng thông đối với cả 2 trường hợp hợp tải đến dưới và vượt khả năng băng thông của liên kết.



Hình 3.12 So sánh chỉ số *TFI* của phương pháp TFBA với RFP và MMFP

### 3.1.4 Phân tích ảnh hưởng của TFBA đến việc lập lịch tại liên kết ra

#### 3.1.4.1 Mô hình phân tích

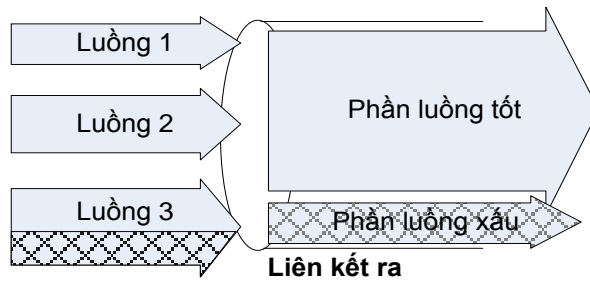
Để phân tích hiệu năng của giải thuật TFBA trên liên kết ra, một mô hình phân tích được xây dựng và xác suất mất chùm trung bình được xác định như một thước đo của hiệu năng này. Các tham số được sử dụng trong mô hình phân tích được liệt kê như Bảng 3.2.

Bảng 3.2 Các tham số sử dụng trong mô hình phân tích

Tham số	Giải thích
$K$	Tổng số luồng trong hệ thống
$\mu$	Tốc độ phục vụ
$W$	Số bước sóng trên liên kết ra
$\lambda_i$	Tốc độ đến của luồng $i$
$AB_i$	Băng thông cung cấp cho luồng $i$
$\lambda^U$	Tốc độ đến phần luồng tốt
$\lambda^O$	Tốc độ đến phần luồng xấu
$P_i^U$	Xác suất mất chùm của phần luồng tốt của luồng $i$
$P_i^O$	Xác suất mất chùm của phần luồng xấu của luồng $i$



Luận án xem xét  $K$  luồng chòm đến tại một liên kết ra chung và chúng có thể được gán vào một trong 2 nhóm: (1) nhóm phần luồng tốt hoặc (2) nhóm phần luồng xấu. Một ví dụ về 3 luồng đến như trong Mục 3.1.3.5, Luồng 1 và Luồng 2 được xem là các luồng tốt vì tốc độ đến thấp hơn băng thông được cung cấp. Riêng Luồng 3, do tốc độ đến vượt quá lượng băng thông được cung cấp, nên phần luồng vượt quá sẽ được đưa vào phần luồng xấu và phần luồng không vượt quá sẽ được đưa vào phần luồng tốt.



**Hình 3.13 Ví dụ về 3 luồng đến được nhóm vào phần luồng tốt và phần luồng xấu**

Với tốc độ đến là  $\lambda_k$  và băng thông được cung cấp là  $AB_i$ ,  $i = 1, 2, \dots, K$ , xác suất luồng  $i$  thuộc nhóm luồng tốt ( $P_i^U$ ) hoặc nhóm luồng xấu ( $P_i^O$ ) là được xác định bởi Công thức 3.9 và Công thức 3.10.

$$P_i^O = \begin{cases} 0 & \text{nếu } \lambda_i \leq AB_i \\ \frac{\lambda_i - AB_i}{AB_i} & \text{ngược lại} \end{cases} \quad (3.9)$$

$$P_i^U = 1 - P_i^O \quad (3.10)$$

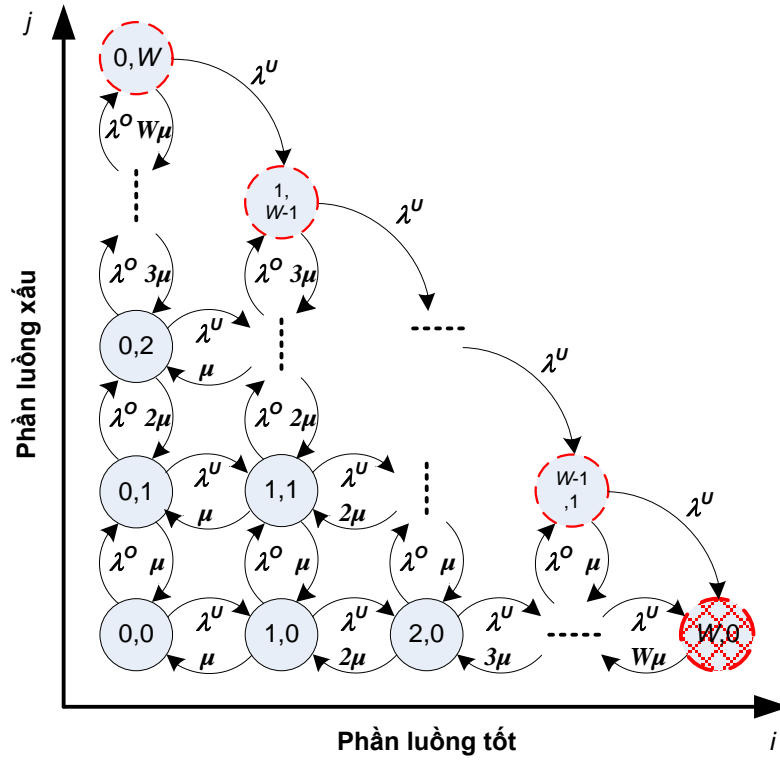
Do đó, tốc độ đến của phần luồng tốt ( $\lambda^U$ ) và phần luồng xấu ( $\lambda^O$ ) là:

$$\lambda^U = \sum_{i=1}^K (\lambda_i \times P_i^U) \quad (3.11)$$

$$\lambda^O = \sum_{i=1}^K (\lambda_i \times P_i^O) \quad (3.12)$$

trong đó  $1/\mu$  là kích thước trung bình của chòm (cho cả phần luồng tốt và phần luồng xấu).

Để ước tính xác suất mất chòm, luận án sử dụng mô hình Markov [11], [32]. Tại một liên kết ra với  $W$  bước sóng, giả thiết rằng phần luồng tốt và phần luồng xấu có phân phối Poisson với tốc độ đến trung bình là  $\lambda^O$  và  $\lambda^U$ , biểu đồ chuyển trạng thái trong mô hình đa chiều như được thể hiện trong Hình 3.13.



**Hình 3.14 Sơ đồ chuyển trạng thái trong mô hình Markov hai chiều**

Đặt  $\pi_{i,j}$  xác suất trạng thái mà tại đó  $i$  bước sóng đã cung cấp cho phản lương tốt và  $j$  bước sóng đã cung cấp cho phản lương xấu (trong Hình 3.14, được biểu thị bằng ký hiệu bởi cặp  $(i,j)$ ), trong đó  $0 \leq i \leq W$ ,  $0 \leq j \leq W$  và  $0 \leq (i+j) \leq W$ , hệ phương trình trạng thái cân bằng thu được bao gồm:

$$(\lambda^U + \lambda^O + (i+j)\mu)\pi_{i,j} = \lambda^U \pi_{i-1,j} + \lambda^O \pi_{i,j-1} + (i+1)\mu\pi_{i+1,j} + (j+1)\mu\pi_{i,j+1} \quad (3.13)$$

với  $0 \leq i < W$ ,  $0 \leq j < W$ ,  $0 \leq i+j < W$

$$(\lambda^U + W\mu)\pi_{0,W} = \lambda^O \pi_{0,W-1} \quad (3.14)$$

$$(\lambda^U + (j+i)\mu)\pi_{i,j} = \lambda^U \pi_{i-1,j+1} + \lambda^U \pi_{i-1,j} + \lambda^O \pi_{i,j-1} \quad (3.15)$$

với  $0 < i < W$ ,  $0 < j \leq W-i$ ,  $i+j = W$

$$W\mu\pi_{W,0} = \lambda^U \pi_{W-1,1} + \lambda^U \pi_{W-1,0} \quad (3.16)$$

Lưu ý rằng xác suất  $\pi_{i,j} = 0$ , với  $i, j < 0$ .

Đặt  $\gamma^O = \lambda^O/\mu$  và  $\gamma^U = \lambda^U/\mu$  là tải đến của phản lương tốt và phản lương xấu; từ các Công thức 3.13, 3.14, 3.15 và 3.16 theo [32] ta có thể xác định  $\pi_{i,j}$  như sau:

$$\pi_{i,j} = \sum_{i=1}^W \frac{(\gamma^U)^i}{i!} \frac{(\gamma^O)^{W-i}}{(W-i)!} \pi_{0,0} \quad (3.16)$$

trong đó  $\pi_{0,0}$  được tính là:

$$\pi_{0,0} = \left[ \sum_{i=1}^W \sum_{j=1}^{W-i} \frac{(\gamma^U)^i}{i!} \frac{(\gamma^O)^j}{j!} \right]^{-1} \quad (3.17)$$

Theo các quy tắc chuyển đổi trạng thái như trong Hình 3.14, xác suất mất chùm xảy ra ở các trạng thái  $(i, W-i)$ ,  $0 \leq i \leq W$ , do các nguyên nhân sau:

- Khi một chùm thuộc phân luồng tốt đến và không có bước sóng nào rời, một bước sóng đã được cấp cho chùm thuộc phân luồng xấu sẽ được giải phóng để dành tài nguyên cho chùm thuộc phân luồng tốt đến. Chùm thuộc phân luồng xấu sẽ bị đánh rơi và hệ thống chuyển sang trạng thái  $(i+1, W-i-1)$ ;
- Khi một chùm thuộc phân luồng xấu đến và không có bước sóng nào rời, nó sẽ bị đánh rơi;
- Ở trạng thái  $(W, 0)$ , một chùm thuộc phân luồng tốt hoặc luồng xấu đến và không có bước sóng nào rời thì chúng đều bị đánh rơi.

Cuối cùng, xác suất mất chùm của chùm thuộc phân luồng xấu và của chùm thuộc phân luồng tốt được xác định bởi Công thức 3.18 và Công thức 3.19.

$$P^O = \sum_{i=0}^W \pi_{i,W-i} \quad (3.18)$$

$$P^U = \pi_{W,0} \quad (3.19)$$

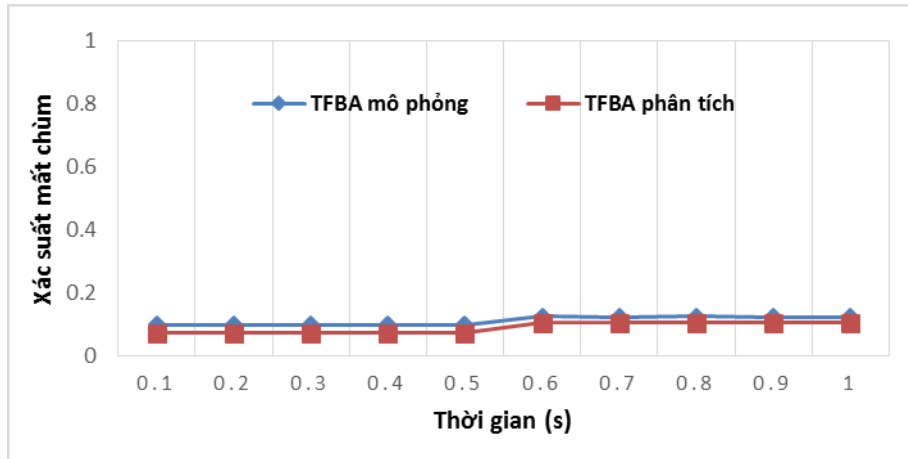
Xác suất mất chùm trên toàn liên kết ra là:

$$P = \frac{(P^U \lambda^U + P^O \lambda^O)}{(\lambda^U + \lambda^O)} \quad (3.20)$$

### 3.1.4.2 So sánh hiệu quả lập lịch giữa mô hình phân tích và kết quả mô phỏng

Luận án sử dụng Mathematica [72] để vẽ đường phân tích, với các tham số như trong Mục 3.2.3.5. Kết quả thể hiện trong Hình 3.15 cho thấy xác suất mất chùm của mô hình phân tích là tương đồng với kết quả mô phỏng (Hình 3.7). Điều này khẳng định tính đúng đắn của mô hình TFBA mà luận án đã đề xuất. Sai số giữa mô hình

phân tích và mô phỏng ở cả 2 giai đoạn là khoảng 2%.



Hình 3.15 So sánh tỉ lệ mất chùm giữa mô hình phân tích và mô phỏng với TFBA

### 3.1.5 Nhận xét

Phương pháp TFBA dựa trên chỉ số công bằng về thông lượng nên có thể áp dụng đối với nhiều loại luồng đến, thay vì chỉ với luồng Poisson như trong [67], [53]. Hơn nữa, đề xuất của luận án xác định phân luồng xấu nhanh chóng hơn bằng việc so sánh giữa  $AT_i$  và  $AB_i$  (nếu  $AT_i > AB_i$  thì chùm đến thuộc luồng xấu), thay vì duy trì 2 gói điều khiển *FBCP* và *BBCP* để trao đổi thông tin như trong [50] làm tiêu tốn nhiều băng thông. Phương pháp tiếp cận của luận án cũng tối ưu hóa được việc sử dụng băng thông với việc điều chỉnh băng thông thực tế về với băng thông cấp phát công bằng dựa trên khả năng băng thông thực tế của mỗi liên kết ( $\omega \times C$ ). Phương pháp TFBA và các kết quả ở trên đã được công bố trong [CT6].

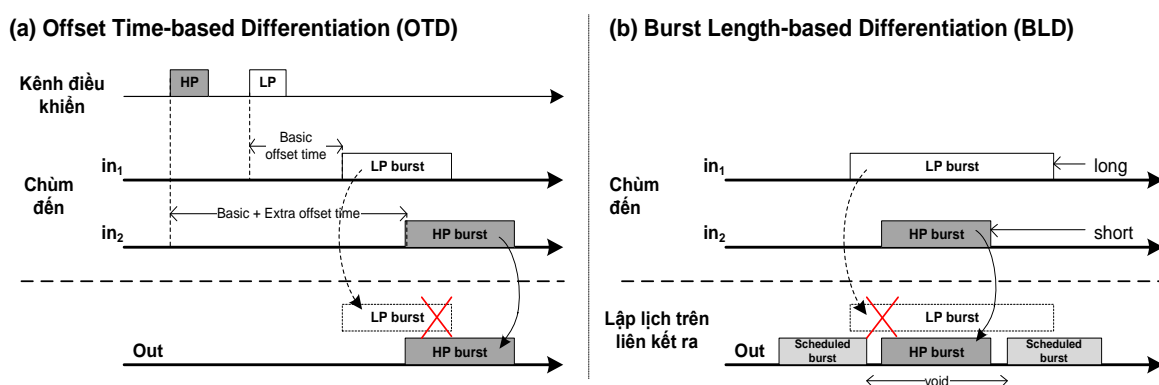
## 3.2 Mô hình đáp chùm hiệu quả bằng thông và công bằng thông lượng

### 3.2.1 Các công trình nghiên cứu liên quan

Vấn đề phân biệt QoS là cần thiết trong môi trường mạng hỗ trợ đa dạng dịch vụ, bao gồm cả mạng OBS. Cung cấp sự phân biệt QoS có thể được thực hiện ở nút biên, nút lõi hoặc tại cả nút biên và lõi [33]. Trong luận án này sự phân biệt chất lượng dịch vụ được thực hiện tại nút biên.

Có hai cách tiếp cận được đề xuất để phân biệt QoS ở tại nút biên: (1) phân biệt dựa trên thời gian *offset* (OTD) [20], [31]; (2) phân biệt dựa vào kích thước độ dài chùm (BLD) [34], [59]. Với OTD, một thời gian *offset* bổ sung (*extra offset*) sẽ được thêm cho các chùm ưu tiên, mà kết quả là các chùm ưu tiên này sẽ có nhiều cơ hội

hơn để đặt trước tài nguyên so với các chùm không ưu tiên như thể hiện ở trong Hình 3.16a. Với BLD, do các chùm có kích thước bé sẽ có nhiều cơ hội hơn để được lập lịch vào các khoảng trống, các gói tin có mức ưu tiên cao sẽ được tập hợp thành các chùm có kích thước bé, nhằm tăng khả năng lập lịch thành công cho các chùm này như thể hiện ở trong Hình 3.16b. Bất kể là theo phương pháp OTD hay BLD, các chùm chỉ được gửi đi sau một khoảng thời gian đệm chùm (bao gồm thời gian tập hợp chùm và thời gian *offset*), việc giảm độ trễ đệm chùm rõ ràng sẽ giúp giảm độ trễ đầu cuối trong mạng [14]. Luận án chủ yếu tập trung vào phương pháp OTD nhằm giảm độ trễ tương ứng cho các chùm trong môi trường hỗ trợ đa dạng dịch vụ.



**Hình 3.16** Ví dụ về (a) QoS dựa vào thời gian offset và (b) QoS dựa vào kích thước chùm

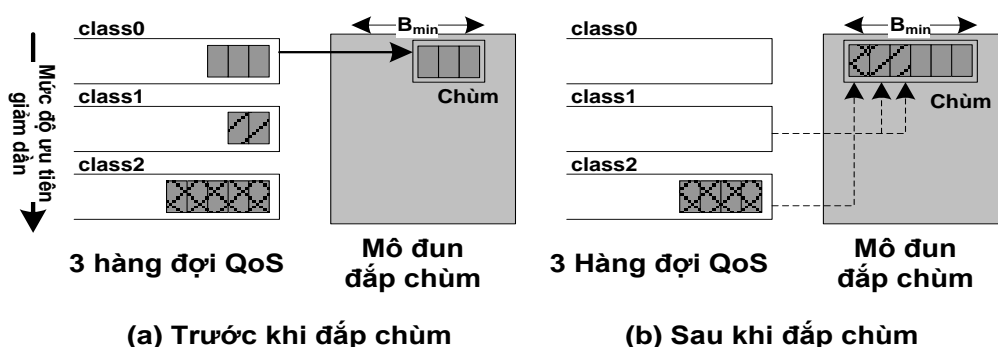
Đã có một số giải pháp được đề xuất nhằm giảm độ trễ đệm chùm như được mô tả trong Mục 2.1. Trong các giải pháp được đề xuất đó chỉ có POQA trong [69] (đã được phân tích trong Mục 2.2) đảm bảo công bằng độ trễ giữa các chùm có mức độ QoS khác nhau (khi không xem xét đến độ trễ tăng thêm do ước tính sai). Tuy nhiên, cũng giống như các giải pháp khác về QoS, POQA sẽ tạo ra các chùm có kích thước bé khi tốc độ đến của các gói tin thấp và thời gian tập hợp chùm ngắn. Theo nghiên cứu của Amin và cộng sự trong [1] và Sarwar và cộng sự trong [45], kích thước các chùm phải lớn hơn hoặc bằng một ngưỡng  $B_{min}$ , nhằm có thể được xử lý (chẳng hạn chuyển mạch) bởi các chuyển mạch vật lý hiện có. Do đó, các chùm sinh ra có kích thước nhỏ hơn  $B_{min}$  phải được đắp thêm bởi các byte đệm (*padded bytes*). Cách làm này rõ ràng là không hiệu quả về mặt sử dụng băng thông của các sợi dẫn quang. Một giải pháp cho vấn đề này là tăng thời gian tập hợp chùm sao cho kích thước các chùm sinh ra lớn hơn hoặc ít nhất là bằng ngưỡng  $B_{min}$ . Tuy nhiên, điều này sẽ làm tăng độ

trở truyền thông đầu cuối của một chùm qua mạng. Phần tiếp theo của luận án sẽ trình bày một đề xuất kết hợp đắp chùm với tập hợp chùm tại nút biên vào của mạng OBS có hỗ trợ đa dạng dịch vụ nhằm nâng cao hiệu quả sử dụng băng thông và tăng tính công bằng về thông lượng.

### 3.2.2 Phương pháp đắp chùm

#### 3.2.2.1 Giới thiệu về mô hình đắp chùm

Mô hình đắp chùm mà luận án đề xuất có tên gọi là QDBAP (*QoS Differentiation Burst Assembly with Padding*) được mô tả như trong Hình 3.17, trong đó việc đắp chùm được thực hiện bằng cách huy động các gói tin từ hàng đợi có QoS thấp sang đắp cho chùm có QoS cao hơn.



Hình 3.17 Một ví dụ về mô hình đắp chùm trên 3 lớp

Khi mật độ các gói tin của luồng  $i$  ( $i = 1, 2, \dots, K-1$ ) đến tại nút biên cao, độ dài của chùm lớn hơn ngưỡng  $B_{min}$  thì không cần phải đắp chùm. Tuy nhiên nếu mật độ các gói tin thuộc luồng  $i$  đến tại nút biên thấp, ngưỡng thời gian  $T_a(i)$  nhanh chóng đạt đến nhưng độ dài chùm  $B_i < B_{min}$ ; do đó có một nhu cầu cần đắp các gói tin từ hàng đợi  $j$  sang chùm  $i$ ,  $j > i$  (từ hàng đợi có QoS thấp sang chùm có QoS cao hơn) để tránh phải dùng byte đệm nhằm tăng hiệu quả sử dụng băng thông. Thêm vào đó, việc đắp chùm này còn góp phần tăng công bằng thông lượng giữa các lớp QoS khác nhau.

#### 3.2.2.2 Quy tắc đắp chùm

Quy tắc đắp chùm được luận án đề xuất như sau:

1. **Chỉ chọn các gói tin từ hàng đợi QoS thấp sang đắp cho chùm QoS cao hơn** (xem Hình 3.17): Lý do là vì hàng đợi càng có QoS cao thì thời gian

tập hợp chòm càng ngắn, như được chỉ ra trong Hình 2.16, nên khả năng đạt đến ngưỡng  $B_{min}$  sớm hơn so với các hàng đợi khác. Cách làm này không chỉ tránh phải dùng byte độn, mà còn làm giảm thời gian tập hợp chòm đối với các gói tin được đắp lên.

2. *Các gói tin được chọn từ hàng đợi QoS thấp theo hình thức đến trước phục vụ trước.* Nghĩa là, các gói tin đến hàng đợi đầu tiên được chuyển lên trước; điều này làm giảm độ trễ tập hợp chòm của chúng.
3. *Các gói tin được chọn từ các hàng đợi QoS thấp sẽ đắp vào đuôi của chòm QoS cao hơn* (xem Hình 3.17): Theo Sarwar và cộng sự trong [45] đã chứng minh, phần đuôi của chòm có xác suất chong lạp (với các chòm đã được lập lịch khác) cao hơn so với các phần còn lại (giữa hay đầu) của chòm.
4. *Chỉ chọn các gói tin từ hàng đợi QoS thấp mà gói điều khiển của nó chưa được gửi đi.* Trong mô hình tập hợp chòm giảm độ trễ như được chỉ ra trong Hình 2.16, gói điều khiển được gửi đi sớm (tại thời điểm  $t_1(i) = T_a(i) - T_o(i)$ ), nên nếu các gói tin được chọn từ các hàng đợi có gói điều khiển đã được gửi đi, độ dài chòm được tập hợp thực tế sau này (sau khoảng thời gian  $T_a(i)$ ) sẽ ngắn hơn nhiều so với thông tin dự đoán về độ dài chòm được mang trong gói điều khiển đã được gửi đi. Kết quả là việc đặt trước tài nguyên (băng thông) ở các nút tiếp theo sẽ quá thừa, gây lãng phí băng thông.

### 3.2.2.3 Mô tả giải thuật đắp chòm QDBAP

Giải thuật đắp chòm QDBAP được mô tả như sau:

#### Giải thuật 5: QDBAP

##### Input:

$q_i$ ; // danh sách các gói tin đến trong hàng đợi  $i$   
 $B_{min}$ ; // kích thước chòm tối thiểu  
 $T_a(i)$ ; // thời gian tập hợp chòm trên hàng đợi  $i$   
 $T_o(i)$ ; // thời gian offset trên hàng đợi  $i$

##### Output: BURST( $i$ );

// tập các chòm được tập hợp

##### Begin

```

1  BURST(i) := ∅;
2  B(i) := 0;           // B(i) là chùm đang được tập hợp
3  KT := false;       // kiểm tra thời điểm gửi gói điều khiển
4  while (qi ≠ ∅) do
5      p := gói tin đến hàng đợi; qi := qi \ {p};
6      t(i) := sp;     // sp thời điểm đến gói tin p
7      if (B(i) = 0) then // hàng đợi qi đang rỗng
8          Ta(i) := Ta(i) + sp;
9          t1(i) := Ta(i) - To(i); // t1(i) là thời điểm gửi gói điều khiển
10     end if
11     B(i) := B(i) + Lp; // Lp là kích thước của gói tin p
12     if ((T(i) ≥ t1(i)) and (KT = false)) then // gửi gói điều khiển
13         L(i) := B(i); // độ dài chùm hiện thời
14         Lε(i) := (1 - α(i)) × Lavg(i) + α(i) × L(i) ×  $\frac{T_a(i)}{T_a(i) - T_o(i)}$ ;
15         if (Lε(i) < Bmin) then
16             Lε(i) := Bmin; // độ dài ước tính bằng Bmin
17         end if
18     end if
19     if (t(i) ≥ Ta(i)) then // gửi chùm
20         L(i) := B(i); // độ dài chùm hoàn thành
21         if (L(i) < Bmin) then // đắp chùm
22             j := i+1;
23             while ((L(i) < Bmin) and (j ≤ n) and (t(i) < t1(j))) do
24                 if (L(j) ≥ Bmin - L(i)) then
25                     L(j) := L(j) - (Bmin - L(i));
26                     L(i) := Bmin;
27                     Cập nhật thời gian trên hàng đợi qj;
28                 else
29                     L(i) := L(i) + L(j);
30                     L(j) := 0;
31                     Tắt bộ đếm thời gian trên hàng đợi qj;
32                 end if
33             j := j + 1;
34         end while
35     end if
36     BURST(i) := BURST(i) ∪ B(i); // một chùm mới hoàn thành
37     B(i) := 0;

```



```

38          $L_{avg}(i) := L(i)$ ;
39     end if
40 end while
41 return  $BURST(i)$ 
End

```

Độ phức tạp tính toán thời gian của giải thuật QDBAP chủ yếu được thực hiện trong vòng lặp **while** (dòng 4 đến dòng 40), trong vòng lặp **while** này có một vòng lặp **while** khác (từ dòng 23 đến dòng 34). Vòng lặp **while** trong có độ phức tạp là  $O(N)$  với  $N$  là số hàng đợi QoS. Vòng lặp **While** ngoài có độ phức tạp là  $O(M)$  với  $M$  là số gói tin đến trong hàng đợi. Do đó, độ phức tạp tính toán của giải thuật QDBAP là  $O(M \times N)$ .

### 3.2.2.4 So sánh và đánh giá dựa trên kết quả mô phỏng

Với mục tiêu của mô phỏng là:

- So sánh băng thông lãng phí khi phải sử dụng byte độn và
- So sánh chỉ số công bằng thông lượng  $TFI$  (được tính bởi Công thức 3.8).

Luận án tiến hành xem xét nút biên vào OBS như được chỉ ra ở Hình 3.1. Các gói tin đến tại nút biên vào OBS được giả định là có phân bố Poisson, trong đó độ dài của chúng thay đổi trong đoạn [500, 1000] bytes. Giả sử rằng có 3 lớp QoS là class0, class1 và class2, được xem xét theo thứ tự ưu tiên giảm dần. Ba hàng đợi tập hợp chòm  $q_0$ ,  $q_1$  và  $q_2$ , với thời gian tập hợp chòm lần lượt là  $T_a(0) = 0.4$  ms,  $T_a(1) = 0.45$  ms,  $T_a(2) = 0.5$  ms, các giá trị thời gian *offset* được thiết lập là  $T_o(0) = 0.3$  ms,  $T_o(1) = 0.25$  ms,  $T_o(2) = 0.2$  ms.

**Bảng 3.3 Độ dài trung bình của những chòm hoàn thành với tải chuẩn hóa đến 0.2**

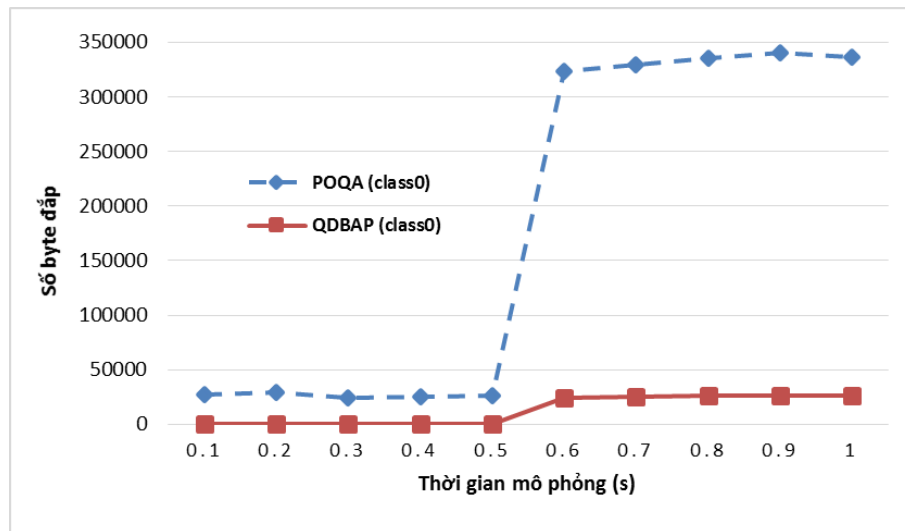
Thời gian (s)		0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1
Độ dài chòm (bytes)	$class_0$	32500	32400	32500	32400	32500	32600	32500	32600	32500	32500
	$class_1$	38600	38500	38800	38500	38600	38600	38500	38500	38600	38600
	$class_2$	42400	42600	42500	42500	42500	42600	42500	42600	42600	42500

Với tải đến của các gói tin là như nhau ở cả 3 hàng đợi  $q_0$ ,  $q_1$  và  $q_2$  là 0.2, Bảng 3.3 cho thấy rằng kích thước trung bình của những chòm hoàn thành trong class0 là bé nhất. Do đó, luận án lựa chọn giá trị  $B_{min}$  là 30000 bytes cho các mô phỏng sau này. Việc chọn lọc này cũng phù hợp với khuyến cáo của Kantarci và

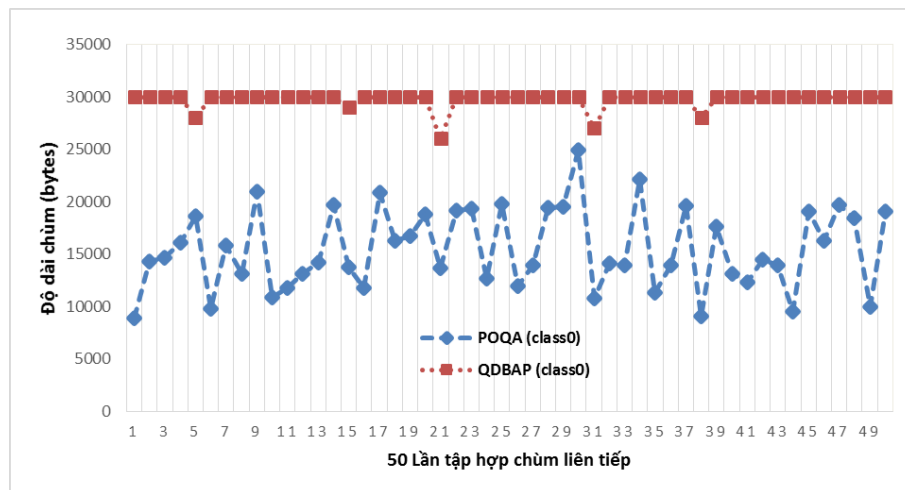
cộng sự trong [20] cho rằng kích thước của một chùm tối thiểu phải nằm trong đoạn [1.25, 30] Kbytes. Hai kịch bản được xem xét là: Kịch bản 1 từ 0 s đến 0.5 s tải chuẩn hóa như nhau (0.2) đến tại cả 3 hàng đợi, và Kịch bản 2 từ 0.6 s đến 1.0 s tải chuẩn hóa đến tại các hàng đợi  $q_0$ ,  $q_1$  và  $q_2$  lần lượt là 0.1, 0.25 và 0.25.

### a. So sánh tỉ lệ lãng phí băng thông

Trong Kịch bản 1, với mật độ đến là như nhau ở cả 3 lớp ưu tiên, số byte đệm được sử dụng cho POQA và QDBAP là không đáng kể vì không có nhiều chùm tạo ra nhỏ hơn  $B_{min}$ , trừ các chùm thuộc class0 do thời gian tập hợp chùm ngắn. Bằng cách chuyển các gói tin từ hàng đợi  $q_1$  và  $q_2$  sang chùm thuộc class0, nên không có số byte đệm ở giải thuật QDBAP (xem Hình 3.18).



Hình 3.18 So sánh số byte đệm giữa QDBAP và POQA

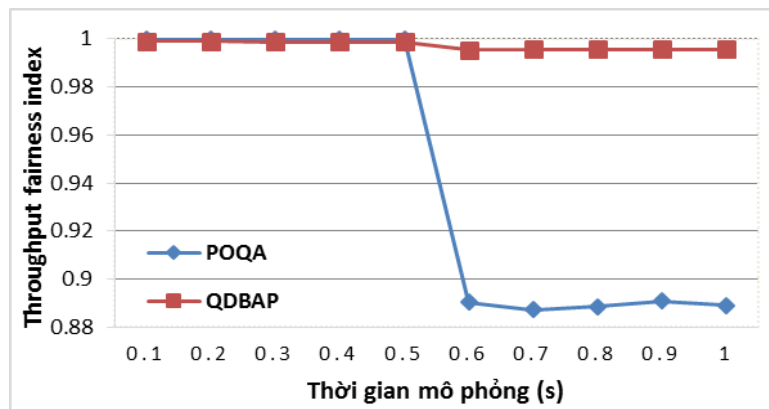


Hình 3.19 Độ dài chùm hoàn thành thuộc class0 trong 50 lần tập hợp chùm liên tiếp

Tuy nhiên, khi giảm mật độ các gói tin đến class0 trong Kịch bản 2, tất cả các chùm thuộc class0 đều nhỏ hơn  $B_{min}$ . Nhiều byte độn được sử dụng trong POQA, nhưng rất ít được sử dụng trong QDBAP vì phương pháp này di chuyển các gói tin từ hàng đợi  $q_1$  và  $q_2$  lên đầu cho chùm thuộc class0. Tuy nhiên, vì Quy tắc 4 (xem Mục 3.2.2.2) chỉ cho phép điều này thực hiện nếu gói điều khiển ở  $q_1$  và  $q_2$  chưa được gửi đi, vẫn còn một số chùm ở class0 nhỏ hơn  $B_{min}$  (xem Hình 3.19) và số byte độn vẫn còn tồn tại ở QDBAP.

### b. So sánh, đánh giá vấn đề công bằng thông lượng

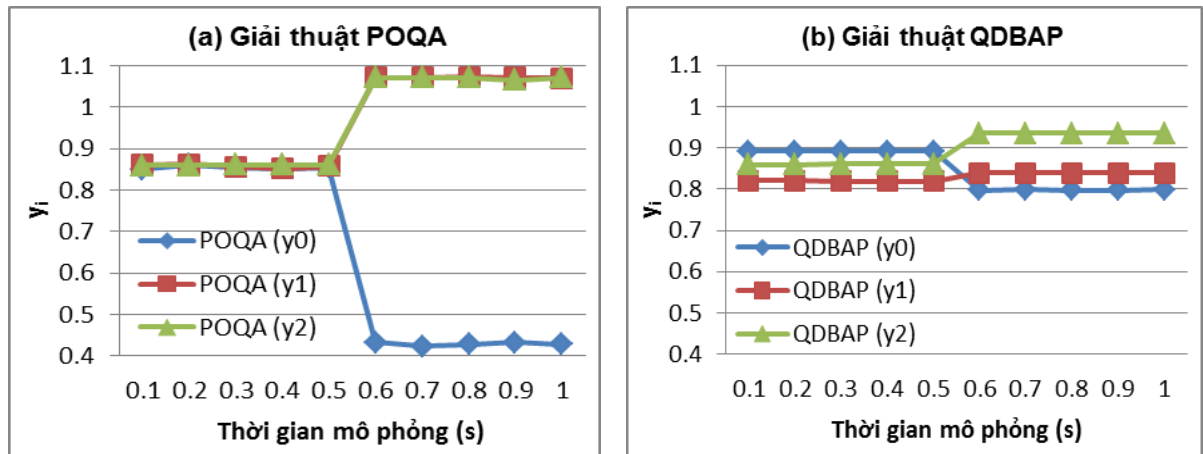
Như đã được mô tả trong Mục 3.1.3.2, khả năng sử dụng băng thông của mỗi liên kết là  $\omega = 0.7$ , giả sử rằng băng thông được cung cấp đều cho cả 3 class là  $AB_i = 0.2333$ . Trong Kịch bản 1, với tải chuẩn hóa đến ở mỗi lớp là 0.2 nên tải các chùm hoàn thành sau tập hợp của POQA cũng là 0.2. Kết quả là,  $y_0 = y_1 = y_2$  (xem Hình 3.21a) và chỉ số  $TFI = 1$  (xem Hình 3.20). Với QDBAP, có một lượng ít các chùm thuộc class0 bé hơn  $B_{min}$  nên cần huy động dữ liệu (gói tin) thuộc class1 (tuy nhiên con số này là không đáng kể), kết quả là  $y_0 \approx y_1 \approx y_2$  và  $TFI \approx 1$  (xem Hình 3.21b).



Hình 3.20 So sánh dựa trên chỉ số công bằng thông lượng giữa QDBAP và POQA

Trong Kịch bản 2, tải chuẩn hóa đến của class0, class1 và class2 có sự thay đổi là 0.1, 0.25 và 0.25, do đó giá trị  $y_0$ ,  $y_1$  và  $y_2$  ở POQA có sự thay đổi lớn (xem Hình 3.21a), đây là nguyên nhân làm cho chỉ số  $TFI$  của POQA giảm còn khoảng 0.89 (giá trị trung bình). Tuy nhiên, với QDBAP, bằng cách di chuyển các gói tin từ hàng đợi  $q_1$  và  $q_2$  lên chùm thuộc class0, làm cho tải chuẩn hoá đến thật sự của class1 và class2 là giảm xuống, trong khi tải chuẩn hoá đến thật sự của class0 lại tăng lên. Điều này, giúp cân bằng các giá trị  $y_0$ ,  $y_1$  và  $y_2$  hơn (xem Hình 3.21b), mà kết quả làm cho chỉ số

$TFI$  (trung bình 0.99) không giảm đáng kể so với Kịch bản 1.



**Hình 3.21** So sánh công bằng thông lượng (dựa trên tỉ lệ tải thực tế  $class_i$  trên khả năng đáp ứng bằng thông  $TB_i (y_i)$ ) giữa POQA và QDBAP

### 3.2.3 Nhận xét

Trong mục này, luận án đã đề xuất một phương pháp đáp chùng với các chùng có kích thước bé hơn  $B_{min}$  trong môi trường đa dạng dịch vụ. Từ kết quả mô phỏng cũng đã chỉ ra rằng, phương pháp đáp chùng thật sự mang lại hiệu quả tối đa khi số lượng các chùng bé hơn  $B_{min}$  lớn. Ngoài việc giảm đáng kể lượng băng thông lãng phí, phương pháp đáp chùng còn mang lại hiệu quả trong việc điều khiển công bằng thông lượng giữa các lớp dịch vụ khác nhau. Tuy nhiên, hạn chế của phương pháp đáp chùng là làm tăng độ phức tạp thời gian của giải thuật (khi có thêm vòng lặp **while** từ dòng 23 đến 34 của giải thuật QDBAP), nhưng độ phức tạp thời gian của giải thuật vẫn ở mức đa thức nên có thể triển khai được. Phương pháp QDBAP đã được công bố trong [CT7].

### 3.3 Tiểu kết Chương 3

Trong chương này, luận án đã giới thiệu một phương pháp cấp phát băng thông công bằng dựa trên thông lượng TFBA. Giải thuật TFBA có thể áp dụng cho nhiều loại luồng đến, đồng thời nâng cao đáng kể hiệu năng của mạng và điều phối công bằng thông lượng, kết quả này đã được phản ánh trong [CT6]. Ngoài ra, luận án đã đề xuất một phương pháp đáp chùng kết hợp với tập hợp chùng, nhằm tối ưu băng thông và điều phối công bằng thông lượng giữa các lớp dịch vụ khác nhau có tên là QDBAP. Kết quả trong phần này cũng đã được công bố trong [CT7].

# KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN CỦA LUẬN ÁN

## KẾT LUẬN:

Chuyển mạch chùm quang trên mạng WDM được xem là một giải pháp đầy triển vọng cho mạng Internet thế hệ tiếp theo, bởi vì OBS khắc phục được những hạn chế về công nghệ của chuyển mạch gói quang hiện tại và khai thác băng thông linh hoạt, tốt hơn chuyển mạch kênh quang. Một trong những vấn đề quan trọng của mạng OBS là làm thế nào để điều khiển công bằng giữa các luồng dịch vụ khác nhau. Với mục đích đó luận án đã tập trung nghiên cứu các mô hình, giải thuật điều khiển công bằng trong mạng OBS với các hướng tiếp cận khác nhau. Kết quả mà luận án đã đạt được bao gồm:

1. Tổng hợp phân tích, đánh giá và phân loại các phương pháp điều khiển công bằng trong mạng OBS. Qua đó chỉ ra được các điểm tồn tại của các công bố trước đây và đây chính là cơ sở để luận án đề xuất và cải tiến một số mô đun chức năng, giải thuật điều khiển công bằng tốt hơn.
2. Đề xuất 2 mô hình tập hợp chùm giảm độ trễ có tên là iBADR [CT2], OBADR [CT3] nhằm giảm độ trễ tập hợp chùm tốt hơn trên các hàng đợi.
3. Đề xuất mô hình tập hợp chùm đảm bảo công bằng độ trễ BADF [CT5].
4. Đề xuất mô hình cấp phát băng thông công bằng dựa trên thông lượng TFBA, áp dụng cho nhiều loại luồng đến khác nhau [CT6].
5. Phương pháp đáp chùm QDBAP [CT7] cũng đã được đề xuất nhằm tối ưu băng thông sử dụng và góp phần điều khiển công bằng thông lượng.

## HƯỚNG PHÁT TRIỂN LUẬN ÁN:

Từ những kết quả đạt được trong luận án một số vấn đề cần được quan tâm nghiên cứu trong thời gian tới:

1. Nghiên cứu vấn đề điều khiển công bằng khoảng cách, để thấy được vai trò của công bằng khoảng cách trong vấn đề truyền và nhận dữ liệu trong mạng.
2. Xây dựng mô hình điều khiển công bằng kết hợp bao gồm giữa độ trễ với thông lượng, hay giữa thông lượng với khoảng cách hoặc cả 3 loại công bằng trên.

## DANH MỤC CÁC CÔNG TRÌNH LIÊN QUAN ĐẾN LUẬN ÁN

- [CT1]. **Lê Văn Hòa**, Võ Viết Minh Nhật, Nguyễn Hoàng Sơn (2016), “Phân tích các giải thuật tập hợp chòm giảm độ trễ tại nút biên mạng OBS”, *Tạp chí Khoa học và công nghệ (Đại học Khoa học, Đại học Huế)*, tập 6, số 1, trang: 9-20.
- [CT2]. **Lê Văn Hòa**, Võ Viết Minh Nhật, Nguyễn Hoàng Sơn (2017), “Một hướng tiếp cận tập hợp chòm cải tiến nhằm giảm độ trễ tại các nút biên mạng OBS”, *Tạp chí Khoa học và công nghệ (Đại học Huế)*, tập 126, số 2A, trang: 19-30.
- [CT3]. Vo Viet Minh Nhat, **Le Van Hoa**, Nguyen Hoang Son (2017), “A model of optimal burst assembly for delay reduction at ingress OBS nodes”, *Turkish Journal of Electrical Engineering & Computer Sciences (SCIE)*, vol. 25, no. 5, pp. 3970-3982.
- [CT4]. **Lê Văn Hòa**, Võ Viết Minh Nhật, Nguyễn Hoàng Sơn (2018), “Ảnh hưởng của tính chất luồng dữ liệu đến hiệu quả tập hợp chòm giảm độ trễ tại nút biên mạng OBS”, *Hội nghị khoa học quốc gia lần thứ XI về nghiên cứu cơ bản và ứng dụng Công nghệ thông tin (FAIR)*, trang: 57-64.
- [CT5]. Vo Viet Minh Nhat, **Le Van Hoa**, Le Manh Thanh (2018), “On the delay fairness through the burst assembly for service difference”, *ETRI Journal (SCIE)*, vol. 40, no. 3, pp. 347-354.
- [CT6]. **Le Van Hoa**, Vo Viet Minh Nhat, Le Manh Thanh (2018) “Throughput-based Fair Bandwidth Allocation in OBS Networks”, *ETRI Journal (SCIE)*, vol. 40, no. 5, pp. 624-633.
- [CT7]. Vo Viet Minh Nhat, **Le Van Hoa**, Nguyen Hoang Son, Le Manh Thanh (2018), “A Model of QoS Differentiation Burst Assembly with Padding for Improving the Performance of OBS Networks”, *Turkish Journal of Electrical Engineering & Computer Sciences (SCIE)*, vol 26, no. 4, pp. 1783-1795.

## TÀI LIỆU THAM KHẢO

- [1] A.A. Amin et al. Development of an Optical-Burst Switching Node Testbed and Demonstration of Multibit Rate Optical Burst Forwarding. *Journal of Lightwave Technology*, 27(16):3466–3475, 2009.
- [2] A. Ge, F. Callegati, and L. S. Tamil. On Optical Burst Switching and Self-Similar Traffic. *IEEE Communications Letters*, 4:98–100, 2000.
- [3] A.N.Z. Rashed, A.E.N. Mohamed, and O.M.A. Dardeer. Offset Time Management for Fairness Improvement and Blocking Probability Reduction in OBS Networks. *International Journal of Advanced Research in Electronics and Communication Engineering (IJARECE)*, 2(11):846–857, 2013.
- [4] A. Zalesky et al. Performance Analysis of an OBS Edge Router. *IEEE Photonics Technology Letters*, 16:695–698, 2004.
- [5] B. Kantarsi and S. Oktuk. Adaptive Threshold based Burst Assembly in OBS Networks. *Electrical and Computer Engineering, CCECE '06. Canadian Conference on, 2006*.
- [6] B. Kantarci, S. Oktug, and T. Atmaca. Performance of OBS techniques under self-similar traffic based on various burst assembly techniques. *Elsevier Computer Communications*, 30:315–325, 2007.
- [7] B. Zhou and M. A. Bassiouni. Improving fairness in optical-burst-switching networks. *Journal of Optical Networking*, 3(4):214–228, 2004.
- [8] B. Zhou, M. A. Bassiouni and G. Li. Using constrained preemption to improve dropping fairness in optical burst switching networks. *Telecommunication System*, 34:181–194, 2007.
- [9] C. Gauger. Contention resolution in Optical Burst Switching networks. in *Advanced Infrastructures for Photonic Networks: WG 2 Intermediate Report*, pages 62–82, 2002.
- [10] C. F. Hsu and L. C. Yang. On the fairness improvement of channel scheduling in optical burst-switched networks. *Photon Network Communication*, 15:51–66,

2008.

- [11] D. Gross, J. F. Shortle, J. M. Thompson, and C. M. Harris. *Fundamentals of Queueing Theory*, New York: John Wiley&Sons, 2008.
- [12] D. J. Blumenthal, P. R. Prucnal, and J. R. Sauer. Photonic packet switches: architectures and experimental implementations. *Proceedings of the IEEE*, vol. 82, pages 1650–1667, 1994.
- [13] D. L. Mills, C. G. Boncelet, J. G. Elias, P. A. Schragger, and A. W. Jackson. Highball: a high speed, reservedaccess,wide-area network. *Tech. Rep. 90-9-3*, Electrical Engineering Department, University of Delaware, 1990.
- [14] G. Hu and M. Kohn. Evaluation of Packet Delay in OBS Edge Nodes. In: *IEEE International Conference on Transparent Optical Networks*, Nottingham, UK, pages 66–69, 20 November 2006.
- [15] H. Liu and S. Jiang. A mixed-length and time threshold burst assembly algorithm based on traffic prediction in OBS network. *Int. Journal of Sensing, Computing & Control.*, 2(2):87–93, 2012.
- [16] I. Stoica, S. Shenker, and H. Zhang. CoreStateless Fair Queueing. *IEEE/ACM Transactions on Networking*, 11(1):33–46, 2003.
- [17] I. Widjaja. Performance analysis of burst admission-control protocols. *IEEE Proceeding of Communications*, vol.142, pages 7–14, 1995.
- [18] J. Ramamirtham and J. Turner. Design of Wavelength Converting Switches for Optical Burst Switching. in *Proceedings of INFOCOMM*, vol.1, pages 362–370, 2002.
- [19] K. Aparna, S. Venkatachalam, and G. Babu. WDM Optical Network. *Wireless Communication*, 2(5):120–125, 2010.
- [20] K. Dolzer and C. Gauger. On burst assembly in optical burst switching networks - A performance evaluation of just-enough-time. *Teletraffic Science and Engineering*, 4: 149–160, 2001.
- [21] K. H. Liu. WDM optical networks, *IP Over WDM*, pp. 99-154, 2002.
- [22] K. Laevens. Traffic characteristics inside optical burst switched networks. in *Proceeding of Opticomm*, pages 137–148, 2002.



- [23] K. Salad and F. Haidari. On the performance of a simple packet rate estimator. *IEEE/ACS International Conference on Computer Systems and Applications*, pages 392–395, 2008.
- [24] K. Lu, G. Xiao, and I. Chlamtac. Analysis of blocking probability for distributed lightpath establishment in WDM optical networks. *IEEE/ACM Transactions on Networking*, 13(1):187–197, 2004.
- [25] L. Hailong, T.W. Liak, and T. Li-Jin. A Distributed Monitoring-based Fairness Algorithm in Optical Burst Switching Networks. *IEEE International Conference on Communications (ICC)*, vol. 3, pages 1564–1568, 2004.
- [26] L. Xu, H. G. Perros, and G. Rouskas. Techniques for optical packet switching and optical burst switching. *IEEE Communications Magazine*, 39(1):136–142, 2001.
- [27] M. Al-Shargabi, S. Shamsan, A. Ismail, S. M. Idris, and F. Saeed. Ensuring the Fairness among the Network Traffic Types over OBS Networks. *1<sup>st</sup> International Conference of Recent Trends in Information and Communication Technologies (IRICT)*, pages 219–227, 2014.
- [28] M. Hayashitani, K. Okazaki, and N. Yamanaka. A New Burst Assembly Technique Supporting Fair QoS about the Number of Hops in OCBS Multi-hop Networks. In: *The 5th International Conference on Optical Internet (COIN)*, pages 40–42. 2006.
- [29] M. Yoo and C. Qiao. Just-enough-time (JET): A high speed protocol for bursty traffic in optical networks. In: *Proceeding of IEEE/LEOS Conf. on Technologies For a Global Information Infrastructure*, pages 26–27, 1997.
- [30] M. Yoo, C. Qiao, and S. Dixit. QoS Performance of Optical Burst Switching in IP-Over-WDM Networks. *IEEE Journal on Selected Areas in Communications*, 18:2062–2071, 2000.
- [31] M. Yoo, C. Qiao, and S. Dixit. Optical burst switching for service differentiation in the next-generation optical internet. *IEEE Commun Mag*, 39: 98–104, 2010.
- [32] D. Pevac, R. Bojovic, and I. Petrovic. Modelling and Performance Evaluation

- of Optical Burst Switched Node with Deflection Routing and Dynamic Wavelength Allocation. *Elec. Energ.*, vol. 21, no. 2, pp. 183-194, 2008.
- [33] N. Akar et al. A survey of quality of service differentiation mechanisms for optical burst switching networks. *Opt Switch Netw*, 7:1–11, 2010.
- [34] N. Barakat and E. H. Sargent. On Optimal Ingress Treatment of Delay-Sensitive Traffic in Multi-Class OBS Systems. In: *Proc. 3rd International Workshop on Optical Burst Switching*, San Jose (CA), October 2004.
- [35] Peterson and L. Larry. Computer networks: a system approach. *Morgan Kaufmann*, pages 5–52, 1996.
- [36] P. K. Chandra, A. K. Turuk, and B. Sahoo. Survey on optical burst switching in WDM networks. In: *2009 International Conference on Industrial and Information Systems (ICIIS)*, pages 83–88, 2009.
- [37] P. Shanmugapriya and M. DevaPriya. Enhancing Fairness in OBS Networks. *International Journal of Computer Science and Information Technologies (IJCSIT)*, 5(1):38–42, 2014.
- [38] R. Denda, A. Banchs, and W. Effelsberg. The Fairness Challenge in Computer Networks. *QofIS 2000, LNCS 1922*, pages 208–220, 2000.
- [39] R. Jain, D. M. Chiu, and W. R. Hawe. A quantitative measure of fairness and discrimination for resource allocation in shared computer system. *DEC technical report TR301*, vol. cs.NI/9809, no. DEC-TR-301, pages 1–38, 1984.
- [40] S. Kim, Y. Kim, B. Y. Yeon, and M. Kang. An integrated congestion control mechanism for optimized performance using two-step rate controller in optical burst switching networks. *Computer Networks*, 51:606–620, 2007.
- [41] S. Kim, B. Mukherjee, and M. Kang. Integrated Congestion-Control Mechanism in Optical Burst Switching Networks. *Global Telecommunications Conference (GLOBECOM)*, pages 1973-1977, 2005.
- [42] S. Tariq and M. Bassiouni. Improving Fairness of OBS Routing Protocols in Multimode Fiber Networks. *International Conference on Computing, Networking and Communications (ICNC)*, pages 1146-1150, 2013.
- [43] S. K. Tan, G. Mohan, and K. C. Chua. Link scheduling state information based

- offset management for fairness improvement in WDM OBS networks. *Computer Networks*, 45:819–834, 2004.
- [44] S. Sarwar, L. Wallentin, G. Franzl, and H. R. van As. Composite burst assembly with high priority packet in the middle burst, in *Broadband Communications", Networks and Systems BROADNETS 2008.5th International Conference*, London, UK, 2008.
- [45] S. Sarwar, F. Zeeshan, L. Aslam L, M. M. Yousaf M M and W. U. Qounain. Novel Composite Burst Assembly for OBS-Networks. *Sindh Univ Res J (Sci Ser)*, 49(4): 773–778, 2017.
- [46] S.Y. Lee, I. Y. Hwang, Y. S. Lee, and H. S. Park. A study on Offset Time Based Burst Generation Scheme for Optical Burst Switching Networks. In: *International Conference on Systems and Networks Communications (ICSNC)*, Cap Esterel, France, pages 1-5, 2007.
- [47] T. Hashiguchi, X. Wang, H. Morikawa, and T. Aoyama. Burst assembly mechanism with delay reduction for OBS networks. In: *Conference on the Optical Internet*, pages 664–666, 2003.
- [48] T. Mikoshi and T. Takenaka. Improvement of burst transmission delay using offset time for burst assembly in optical burst switching. In: *7th Asia-Pacific Symposium on Information and Telecommunication Technologies*, pages 13–18, 2008.
- [49] T. Orawiwattanakul and Y. Ji. Improving Fairness for Multi-Hop Bursts in Optical Burst Switching Networks. *Asia-Pacific Advanced Network (APAN) Network Research Workshop*, pages 1-5, 2007.
- [50] T. Orawiwattanakul and Y. Ji. Resource Consumption Based Preemption for providing fairness in optical burst switching networks. In: *4th International Conference on Broadband Communications, Networks and Systems*, pages 12-16, 2007.
- [51] T. Orawiwattanakul and Y. Ji. Preemption scheme for providing rate fairness in optical burst switching networks. In: *International Conference on High Performance Switching and Routing (HPSR)*, pages 39–44, 2008.

- [52] T. Orawiwattanakul and Y. Ji. Preemption Scheme for Improving Source Level Fairness in Optical Burst Switching Networks. In: *4th International Conference on Innovations in Information Technology*, pages 710–714, 2008.
- [53] T. Orawiwattanakul and Y. Ji. Fair Bandwidth Allocation in Optical Burst Switching Networks. *Journal of Lightwave Technology*, 27(16):3370–3380, 2009.
- [54] T. Orawiwattanakul, Y. Ji , and N. Sonehara. Fair bandwidth allocation with distance fairness provisioning in optical burst switching networks. *IEEE Global Telecommunications Conference (GLOBECOM)*, pages 1–5, 2010.
- [55] T. Tachibana, B. O. Nassar, and S. Kasahara. Hop-Based Burst-Cluster Transmission: Fairness Improvement in High-Performance OBS Networks. *Journal of Optical Communication Network*, 3:542–552, 2011.
- [56] V. M. Vokkarane, K. Haridoss, and J. P. Jue. Threshold-based burst assembly policies for QoS support in optical burst-switched networks. In: *Proc. SPIE OptiComm, 2002*.
- [57] V. M. Vokkarane and J. P. Jue. Segmentation-based non-preemptive scheduling algorithms for optical burstswitched networks. *Journal of Lightwave Technology*, 23(10):3125–3137, 2005.
- [58] V. M. Vokkarane and J. P. Jue. Prioritized burst segmentation and composite burst-assembly techniques for QoS support in optical burst-switched networks. *IEEE Journal on Selected Areas in Communications*, 21(7):1198–1209, 2003.
- [59] V. M. Vokkarane, K. Haridoss, and J.P. Jue. Threshold-Based Burst Assembly Policies for QoS Support in Optical Burst-Switched Networks. *Proc SPIE/IEEE*, 4874:125–136, 2002.
- [60] W.S. Park, M. Shin, H. W. Lee, and S. Chong. A Joint Design of Congestion Control and Burst Contention Resolution for Optical Burst Switching Networks. *Journal of Lightwave Technology*, 27(17):3820–3830, 2009.
- [61] X. Gao and M.A. Bassiouni. Improving Fairness with Novel Adaptive Routing in Optical Burst-Switched Networks. *Journal of Lightwave Technology*, 27(20):4480–4492, 2009.

- [62] X. Gao and M.A. Bassiouni. Fairness-Improving Adaptive Routing in Optical Burst Switching Mesh Networks, *Communications, 2008. ICC '08. IEEE International Conference on*, Beijing, China, pages 5209–5213, 2008.
- [63] X. Jiang, N. Zhu, and L. Yuan. A novel burst assembly algorithm for OBS networks based on burst size and assembly time prediction. *Journal of Computational Information Systems*, 9(2):463–475, 2013.
- [64] X. Yu, Y. Chen, and C. Qiao. Study of traffic statistics of assembled burst traffic in optical burst switched networks. in *Proceeding of Opticomm*, pages 149–159, 2002.
- [65] Y. Chen, C. Qiao, and X. Yu. Optical Burst switching: a new area in optical networking research. *Network, IEEE*, 18(3):16–23, 2004.
- [66] Y. Fukushima, T. Yokohira, and Y. Noine. A burst assembly method to reduce end-to-end delay in optical burst switching networks. *WSEAS Transactions on Communications*, 8(8):894–903, 2009.
- [67] Y. Liu, K. C. Chua, and G. Mohan. Achieving max-min fairness in WDM optical burst switching networks. *Workshop on High Performance Switching and Routing (HPSR)*, pages 187–191, 2005.
- [68] Y. Xiong, M. Vandenhouste, and H. Cankaya. Control architecture in optical burst-switched WDM networks. *IEEE Journal on Selected Areas in Communications*, 18:1838–1851, 2000.
- [69] Z. Sui, Q. Zeng, and S. Xiao. Adaptive assembly on delay fairness in optical burst switched networks”, *J. Opt. Commun.*, 27(1):35–38, 2006.
- [70] Z. Sui, Q. Zeng, and S. Xiao. An offset differential assembly method at the edge of OBS network. *Proc. of SPIE Optical Transmission, Switching and Subsystems III.*, Vol. 6021, pages 1-6, 2005.
- [71] <https://www.isi.edu/nsnam/ns/>
- [72] <https://www.wolfram.com/mathematica/>