

PHÂN TÍCH CÁC GIẢI THUẬT TẬP HỢP CHÙM GIẢM ĐỘ TRỄ TẠI NÚT BIÊN MẠNG OBS

Lê Văn Hòa^{1*}, Võ Việt Minh Nhật¹, Nguyễn Hoàng Sơn²

¹Khoa Công nghệ thông tin, Trường Đại học Khoa học – Đại học Huế

²Khoa Toán, Trường Đại học Khoa học – Đại học Huế

*Email:levanhhoa@hueuni.edu.vn

TÓM TẮT

Tập hợp chùm đóng một vai trò quan trọng trong việc giảm độ trễ đầu cuối của các gói tin khi chúng được vận chuyển qua một mạng chuyển mạch chùm quang. Đã có một số đề xuất nhằm làm giảm độ trễ các gói tin tập hợp tại các nút biên vào OBS. Tư tưởng chung của các đề xuất là gửi sớm gói điều khiển trước khi hoàn thành chùm. Tuy nhiên do thông tin về độ dài chùm là cần được mang theo trong gói điều khiển, nên các đề xuất đã đưa ra các cách tiếp cận khác nhau nhằm dự đoán kích thước thật của chùm. Bài viết này sẽ phân tích và đánh giá các giải thuật tập hợp chùm giảm độ trễ dựa trên độ trễ giảm được và lỗi ước tính giữa độ dài thật so với độ dài ước tính.

Từ khóa: Mạng OBS, nút biên, tập hợp chùm, giảm độ trễ, lỗi ước tính.

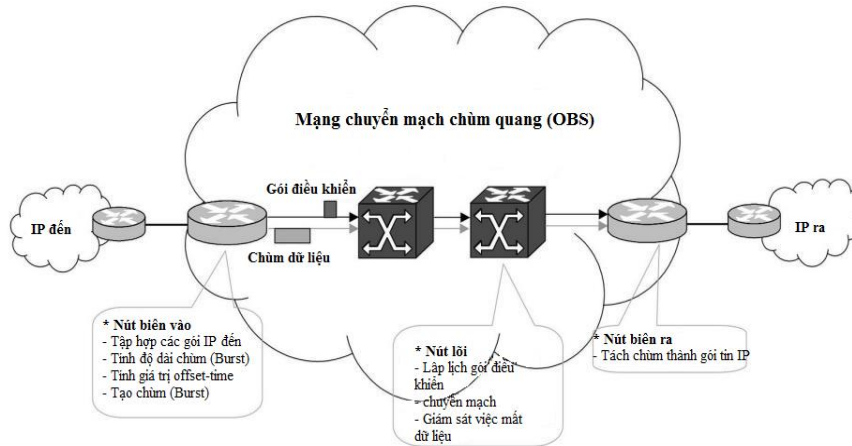
1. MỞ ĐẦU

Chuyển mạch chùm quang (*Optical Burst Switching, OBS*) [2] được xem là một công nghệ hứa hẹn cho việc thực thi Internet quang thế hệ tiếp theo, nhằm đáp ứng sự tăng trưởng nhanh chóng của lưu lượng Internet và việc triển khai gia tăng các dịch vụ mới (như VoIP, video theo yêu cầu, điện toán đám mây, các trung tâm dữ liệu ...). Việc thực thi chuyển mạch chùm quang là nhằm sử dụng hiệu quả hơn băng thông mạng sợi quang, tạo ra một cơ sở hạ tầng mạng linh hoạt và có thể cấu hình ở mức chùm và xử lý được các kiểu lưu lượng *bursty* được sinh ra bởi các dịch vụ nêu trên.

Kiến trúc tiêu biểu của một mạng chuyển mạch chùm quang (mạng OBS) bao gồm các nút lõi kết nối với các nút biên dưới dạng hình lưới (Hình 1). Các nút biên có nhiệm vụ tập hợp các gói điện tử đến từ các mạng truy cập (chẳng hạn các gói IP...) thành các gói lớn hơn, được gọi là chùm dữ liệu (*burst*), là các đơn vị truyền thông chính bên trong mạng OBS. Mỗi nút biên duy trì các hàng đợi tương ứng với các đích đến và cả các lớp QoS, nếu cần thiết. Khi một ngưỡng tập hợp chùm đạt đến, các gói tin trong một hàng đợi sẽ được tập hợp thành một chùm mà nó sẽ được gửi qua mạng sau đó. Một gói điều khiển chùm (*Burst Control Packet, BCP*)

Phân tích các giải thuật tập hợp chùm giảm độ trễ tại nút biên mạng OBS

được gửi đi trước trên một kênh điều khiển dành riêng để đặt trước các băng thông được yêu cầu và cấu hình các chuyển mạch tại các nút trung gian dọc theo hành trình từ nguồn đến đích. Chùm tương ứng theo sau một khoảng thời gian bù đắp (*offset-time*) trên một trong các kênh dữ liệu khả dụng; chùm sẽ được chuyển mạch toàn quang tại tất cả các nút trung gian dọc theo hành trình này.



Hình 1. Kiến trúc của mạng OBS.

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

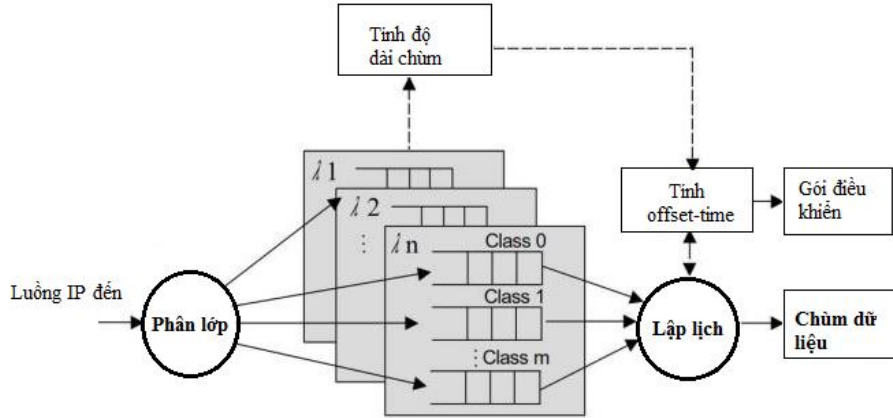
Đã có một số nỗ lực nhằm làm giảm độ trễ đầu cuối dựa trên hoạt động tập hợp chùm, trong đó ý tưởng chính là gửi gói điều khiển đi sớm trước khi chùm được hoàn thành. Cách làm này làm giảm đáng kể độ trễ đệm chùm, nhưng cần phải ước tính độ dài của chùm sẽ được hoàn thành bởi vì thông tin này phải được 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 của các giải thuật. Bài viết này sẽ phân tích và đánh giá các giải thuật đã được đề xuất dựa trên độ trễ đệm chùm và lỗi ước tính.

Các phần tiếp theo của bài báo này gồm: Phần 2 giới thiệu tổng quan về tập hợp chùm; Trình bày các đề xuất về vấn đề tập hợp chùm giảm độ trễ ở trong Phần 3; Phần 4 là so sánh lỗi ước tính của các đề xuất dựa trên mô phỏng; cuối cùng là kết luận ở Phần 5.

2. TỔNG QUAN VỀ TẬP HỢP CHỤM

Tập hợp chùm là một hoạt động được thực thi tại nút biên vào của mạng OBS. Như được chỉ ra trong Hình 2, các gói IP đến được phân loại dựa trên đích đến, hoặc theo QoS nếu

cần thiết, để đưa vào hàng đợi tương ứng. Dựa trên một giải thuật tập hợp chòm đã được cài đặt, sau khoảng thời gian tập hợp chòm gói điều khiển sẽ được gửi đi trước nhằm đặt trước tài nguyên trên hành trình, chòm dữ liệu sẽ theo sau một khoảng thời gian bù đắp.



Hình 2. Mô hình tập hợp chòm tại nút biên vào mạng OBS.

Có 2 mô hình tập hợp chòm truyền thống: tập hợp chòm dựa trên thời gian (*time-based*) [1] và tập hợp chòm dựa trên độ dài (*length-based*) [5]. Trong mô hình dựa trên thời gian, một bộ đếm thời gian (*timer*) được khởi động khi gói tin đầu tiên đến tại một hàng đợi rỗng và một chòm chỉ được hình thành khi bộ đếm thời gian đạt đến một ngưỡng thời gian định trước (ký hiệu là T_a). Trong mô hình dựa trên độ dài, một ngưỡng độ dài (ký hiệu là L_a) chỉ định số lượng gói tin tối đa được tập hợp trong một chòm, hoặc kích thước chòm tối đa theo *byte* nếu các gói tin đến có kích thước thay đổi. Khi độ dài chòm đạt đến ngưỡng này, một chòm sẽ được tạo thành và được gửi vào trong mạng lõi OBS.

Mô hình dựa trên bộ đếm thời gian giới hạn được độ trễ tập hợp trung bình nhưng có thể sinh ra các chòm có kích thước rất nhỏ khi lưu lượng đến thấp, trong khi mô hình dựa trên độ dài có thể đảm bảo độ dài chòm được hình thành nhưng có thể gây ra độ trễ tập hợp chòm rất lớn. Với lý do này, mô hình lai (mô hình dựa trên cả bộ đếm thời gian và độ dài chòm) đã được đề xuất [9,10], trong đó một chòm sẽ được hình thành khi một trong hai ngưỡng này đạt đến.

Tuy nhiên, cho dù dựa trên bộ đếm thời gian, độ dài chòm hay cả hai, các giải thuật tập hợp chòm truyền thống đều chịu một độ trễ đệm chòm, bao gồm độ trễ tập hợp chòm (T_a hoặc $T < T_a$ khi mà ngưỡng độ dài L_a đạt đến trước) và thời gian bù đắp (T_o) như được chỉ ra trong Hình 3a. Độ trễ này có thể là đáng kể đối với một số gói tin bị giới hạn độ trễ, do vậy việc giảm độ trễ đệm chòm là cần thiết. Đã có một số đề xuất về tập hợp chòm giảm độ trễ, mà sẽ được phân tích trong phần tiếp theo.

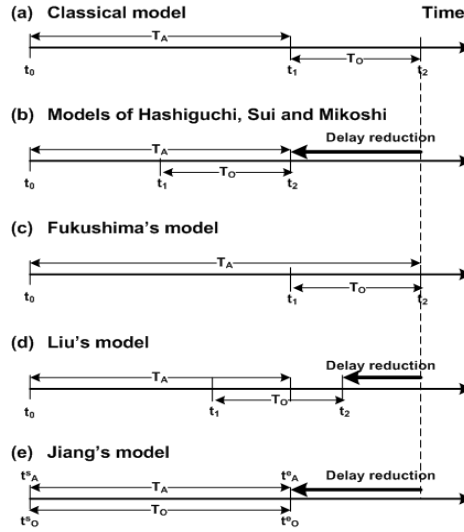
3. CÁC ĐỀ XUẤT TẬP HỢP CHÒM GIẢM ĐỘ TRỄ

Mô hình tập hợp chòm giảm độ trễ được đề xuất đầu tiên bởi Hashiguchi [6], trong đó gói điều khiển được gửi đi tại thời điểm t_1 trước khi hoàn thành chòm (xem Hình 3b). Bằng cách

này, chòm được gửi đi ngay tại thời điểm t_2 , mà không phải chờ đợi một thời gian bù đắp nào như trong mô hình tập hợp chòm truyền thống. Điều đó có nghĩa là các gói tin được tập hợp trong chòm hiện thời giảm được một độ trễ T_o . Tuy nhiên, do thông tin về độ dài chòm phải được mang trong gói điều khiển mà nó phục vụ cho việc đặt trước các tài nguyên được yêu cầu tại các nút trung gian, ước tính độ dài chòm tại thời điểm t_1 là cần thiết. Hashiguchi đã đề xuất một cách ước tính độ dài chòm dựa trên tốc độ đến của gói tin trong khoảng thời gian ước tính (tức là $t_1 - t_0$ hay $T_a - T_o$) bởi công thức sau:

$$L_e = \alpha \times L_w \times \frac{T_a}{T_a - T_o} \quad (1)$$

trong đó L_w là độ dài chòm được hình thành trong khoảng thời gian ước tính và α là một tham số điều khiển.



Hình 3. Mô hình tập hợp chòm giảm độ trễ.

Các mô hình tập hợp chòm của Sui [12] và Mikoshi [7] là tương tự với của mô hình của Hashiguchi, nhưng khác ở phương pháp ước tính độ dài chòm. Cụ thể, Sui ước tính độ dài chòm bằng cách sử dụng một bộ lọc tuyến tính AAR (*Adaptive Auto-Regressive*) với các thống kê chiều dài các chòm đo được trong $M - 1$ lần tập hợp trước đó và lượng gói tin đến trong khoảng thời gian ước tính (L_w). Độ dài chòm ước tính do đó là như sau:

$$L_e = \sum_{i=1}^{M-1} w(i)L(i) + \alpha \times L_w \times \frac{T_a}{T_a - T_o} \quad (2)$$

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

Mikoshi ước tính độ dài chòm dựa trên thuật toán Jacobson/Karels [4] với một số thay đổi. Đầu tiên lỗi ước tính $E(n)$ của lần tập hợp thứ n là khoảng cách giữa độ dài chòm đo được $L(n)$ và độ dài chòm ước tính $L_e(n)$. Lỗi ước tính này sau đó được sử dụng cho việc tính toán độ

dài chòm ước tính của lần tập hợp chòm thứ $n + 1$. Một tham số $D(n + 1)$, được gọi là độ lệch của lần tập hợp chòm thứ $n + 1$ cũng được tính toán dựa trên độ lệch $D(n)$ và lỗi ước tính $E(n)$ của lần tập hợp chòm thứ n . 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 chòm của Mikoshi là như sau:

$$\begin{aligned} E(n) &= L(n) - L_e(n) \\ L_e(n+1) &= L_e(n) + \alpha \times E(n) \\ D(n+1) &= D(n) + \alpha \times (|E(n)| - D(n)) \\ L_e(n+1) &= \mu \times L_e(n+1) + \varphi \times D(n+1) \end{aligned} \quad (3)$$

trong đó α , μ và φ là các tham số điều khiển.

Trong mô hình tập hợp chòm của mình, Fukushima [11] cho phép tập hợp các gói tin đến trong khoảng thời gian bù đắp T_o vào chòm hiện thời (Hình 3c) và đề xuất cách ước tính độ dài chòm dựa trên tốc độ đến trung bình λ_{avg} của M gói tin đến sau cùng nhất:

$$L_e = L + \lambda_{avg} \times T_o \quad (4)$$

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

Nếu chúng ta khái quát hoá thời gian tập hợp chòm như là khoảng thời gian mà các gói tin đến đều được tập hợp trong chòm hiện thời thì mô hình tập chòm của Fukushima là tương tự với các mô hình của Hashiguchi, Sui và Mikoshi, nhưng thời gian tập hợp chòm là $T_a + T_o$.

Cách tiếp cận của Liu [3] cũng tương tự với Fukushima, nhưng với một giải thuật tập hợp lại được sử dụng. Cụ thể, gói điều khiển sẽ được gửi đi khi thời gian tập hợp chòm đạt đến một ngưỡng thời gian được định trước T_a hoặc độ dài chòm đạt đến một ngưỡng độ dài tối thiểu L_{min} . Liu đề xuất một phương pháp ước tính độ dài chòm dựa trên sự tăng/giảm của tốc độ gói tin đến của lần tập hợp chòm hiện thời (λ_{cur}) so với lần tập hợp chòm trước đó (λ_{pre}) như sau:

$$L_e = L + \left(\lambda_{pre} + (\lambda_{cur} - \lambda_{pre}) \times \frac{T_o}{T_w + T_o} \right) \times T_o \quad (5)$$

trong đó cửa sổ thời gian $T_w + T_o$ là khoảng thời gian giữa 2 thời điểm hoàn thành chòm liên tiếp. nếu bộ đếm thời gian đạt đến ngưỡng thời gian T_a trước, $T_w = T_a$ và như vậy mô hình của Liu là tương đương với mô hình tập hợp chòm dựa trên ngưỡng thời gian truyền thống (Hình 3a). Trong trường hợp ngưỡng độ dài tối thiểu L_{min} đạt đến trước, như được chỉ ra trong hình 3d, các gói tin được tập hợp trong chòm hiện thời giảm được một độ trễ là $t_1 + T_o - T_a$.

Với đề xuất của Jiang [8], 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 bù đắp T_o , chính là thời gian tập hợp chòm T_a , và độ dài chòm L_e đã được ước tính tính toán trong lần tập hợp chòm sau cùng nhất. Mô hình tập hợp chòm của Jiang là một mô hình lai, 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 ($T(i)$, $1 \leq i \leq M$) bằng công thức sau:

$$T_a = \frac{\sum_{i=1}^M T(i)}{M} \quad (6)$$

Tương tự, độ dài chùm ước tính L_e của lần tập hợp chùm 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):

$$L_e = L_{\min} + step \times (L_{\max} - L_{\min}) / N \quad (7)$$

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 mô hình của Jiang có thể được triệt tiêu ($E = 0$) 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 như 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 $E = L - L_e$.

Một so sánh giữa các mô hình tập hợp chùm nêu trên được thể hiện ở trong Bảng 1.

Bảng 1. So sánh các giải thuật tập hợp chùm giảm độ trễ đã được đề xuất

	Hashiguchi [6]	Sui[12]	Mikoshi [7]	Fukushima [11]	Liu [3]	Jiang [8]
Giải thuật 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 của 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

Để xem xét rõ hơn về lỗi ước tính của các đề xuất chúng tôi đã tiến hành mô phỏng lại các giải thuật, kết quả cho thấy ở trong phần 4.

4. SO SÁNH VÀ ĐÁNH GIÁ DỰA TRÊN MÔ PHÒNG

Mô phỏng được triển khai trên một PC of 2.4 GHz Intel Core 2 CPU, 2G RAM. Các gói tin đến tại hàng đợi của một nút biên mạng OBS có phân bố Poisson với kích thước của chúng thay đổi ngẫu nhiên từ 500 đến 1000 bytes. Các lưu lượng tải được xem xét từ 0.1 đến 0.9 Erlang. Mô phỏng được thực hiện trong thời gian 1 giây. Dữ liệu được trích xuất từ NS2. Các tham số tập hợp chòm bao gồm: $T_a = 6$ ms, $T_o = 2$ ms. Các mục tiêu mô phỏng là:

– So sánh tỉ lệ lỗi ước tính trung bình của các giải thuật đã đề xuất, được tính bởi công thức:

$$R_E = \frac{\sum_{i=1}^M (|L(i) - L_e(i)| / L(i))}{M} \quad (8)$$

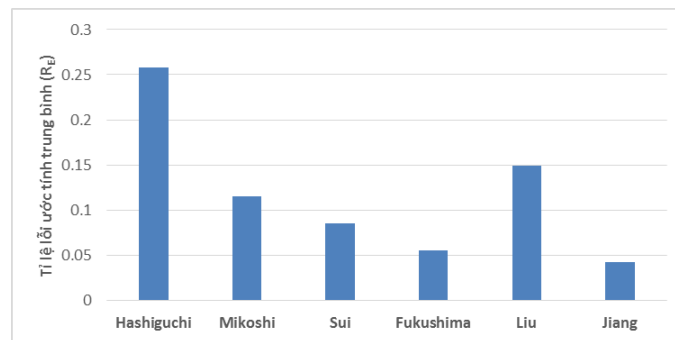
trong đó M là số lần tập hợp chòm, $L(i)$ là kích thước thật của chòm thứ i và $L_e(i)$ là kích thước dự đoán được của chòm thứ i .

– So sánh số gói tin thừa được chuyển cho chòm tiếp theo trong 100 lần tập hợp chòm liên tiếp của các giải thuật.

– Phân tích cách chọn ngưỡng của giải thuật Jiang, giải thuật tốt nhất trong số các mô hình đã đề xuất.

4.1. So sánh tỉ lệ lỗi ước tính trung bình của các giải thuật đã đề xuất

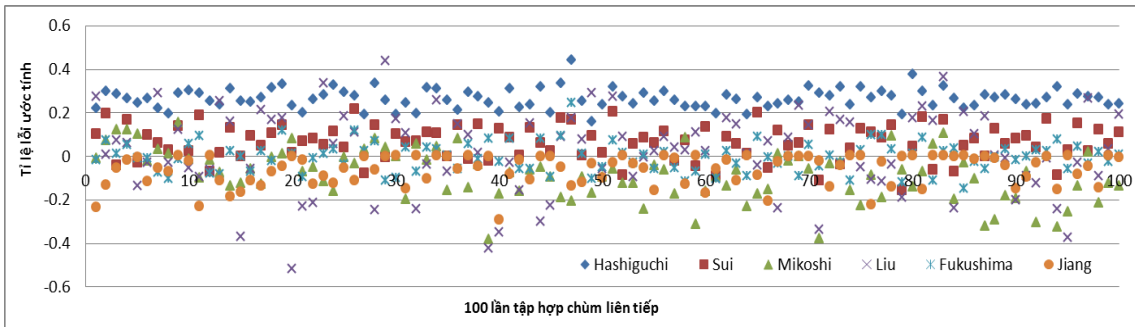
Mô phỏng được thực hiện với các thông số sau: tải đến là 0.5 Erlang, cặp ngưỡng $L_{min}=130000$ byte và $L_{max}=170000$ byte đối với giải thuật Jiang (cặp ngưỡng này được xác định dựa trên kích thước chòm tối thiểu và tối đa được sinh ra trong các giải thuật còn lại). Hình 4 mô tả một so sánh về lỗi ước tính trung bình giữa các giải thuật đã đề xuất.



Hình 4. So sánh tỉ lệ lỗi ước tính trung bình của các giải thuật với tải đến 0.5 Erlang.

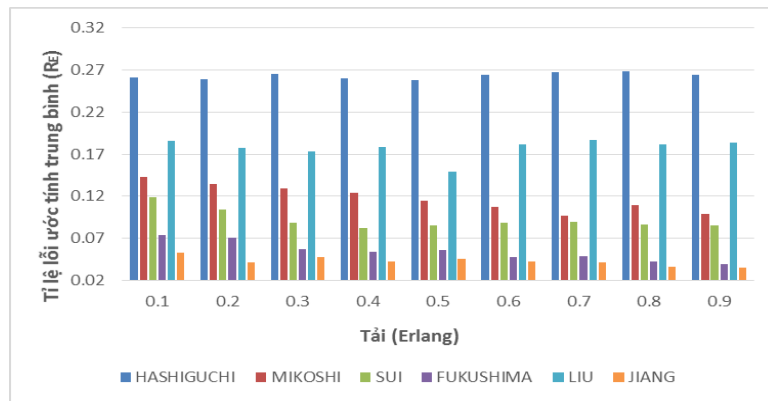
Kết quả mô phỏng trên Hình 4 cho thấy rằng lỗi ước tính trung bình của các giải thuật có sử dụng phương pháp thống kê như của Jiang, Fukushima và Sui cho lỗi ước tính thấp hơn so với các giải thuật còn lại. Để tìm hiểu sâu hơn về kết quả này, chúng tôi xem xét sự phân bố lỗi ước tính của 100 lần tập hợp chòm liên tiếp.

Phân tích các giải thuật tập hợp chòm giảm độ trễ tại nút biên mạng OBS



Hình 5. Phân bố tỉ lệ lỗi ước tính của của các giải thuật trong 100 lần tập hợp chòm liên tiếp.

Như mô tả trong Hình 5, lỗi ước tính phân bố xung quanh giá trị 0. Thực tế, lỗi ước tính xảy ra một trong 2 trường hợp sau: (1) kích thước của chòm hoàn thành lớn hơn kích thước chòm ước tính, nên các gói tin thừa sẽ được chuyển sang cho lần tập hợp kế tiếp và (2) kích thước của chòm hoàn thành nhỏ hơn kích thước chòm ước tính, lúc này có một sự lãng phí băng thông do tài nguyên được đặt trước nhiều hơn yêu cầu sử dụng của chòm. Như mô tả trong Hình 5, giải thuật Jiang cho kết quả tập hợp tốt nhất với phân bố lỗi ước tính gần với giá trị 0.

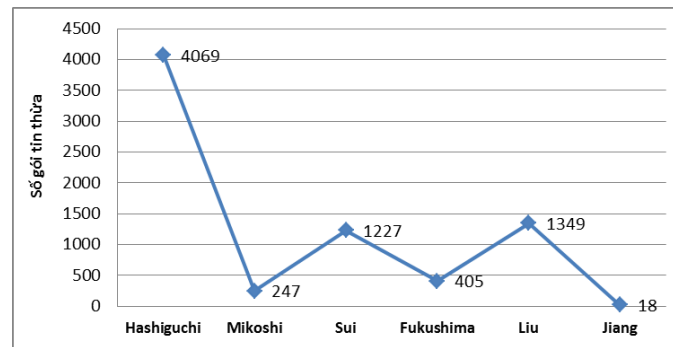


Hình 6. Tỉ lệ lỗi ước tính trung bình gần như không đổi đối với tải từ 0.1 đến 0.9 Erlang.

Vậy thì liệu tải lưu lượng đến có tác động đến lỗi ước tính hay không ? Chúng tôi xem xét trường hợp tải thay đổi từ 0.1 đến 0.9 Erlang và kết quả trong Hình 6 chỉ ra rằng lỗi ước tính gần như không phụ thuộc vào các cấp độ tải lưu lượng đến đến.

4.2. So sánh số gói tin thừa được chuyển cho chòm tiếp theo

Như đã phân tích trong Mục 4.1, 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 này cũng đồng nghĩa với việc làm giảm đáng kể độ trễ trung bình của toàn mạng OBS. Hình 7 so sánh số gói tin thừa của các giải thuật trong 100 lần tập hợp chòm liên tiếp, trong đó giải thuật Jiang có số gói tin thừa được chuyển cho chòm tiếp theo là thấp nhất.



Hình 7. So sánh số gói tin thừa trong 100 chòm sinh ra đầu tiên.

4.3. Phân tích cách chọn ngưỡng của giải thuật Jiang

Như được mô tả trong các hình trên, giải thuật Jiang 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, 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. Ảnh hưởng cặp giá trị ngưỡng (L_{\min} , L_{\max}) đến lỗi ước tính với tải đến 0.5 Erlang

Cặp ngưỡng (Bytes)	Lỗi ước tính trung bình	Số chòm sinh ra	Số gói thừa trong 100 chòm đầu tiên	Số chòm thừa trong 100 chòm đầu tiên	Tổng gói trong 100 chòm 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

4.4. Nhận xét

Tóm lại, các sơ đồ tập hợp chòm nêu trên đều cố gắng giảm độ trễ đệm chòm về thành độ trễ tập hợp. Tuy nhiên, các mô hình này vẫn bộc lộ các tồn tại sau:

(1) Lỗi ước tính là đáng kể, như được chỉ ra trong Hình 4. Lỗi ước tính sẽ gây ra lãng phí băng thông đặt trước nếu độ dài chòm ước tính dài hơn độ dài chòm được hoàn thành. Trong trường hợp kích thước chòm ước tính ít hơn tổng gói tin đến thực tế tại một hàng đợi, các gói

vượt quá sẽ được tập hợp trong chòm tiếp theo. Kết quả là, các gói này phải chịu một độ trễ thêm bằng độ trễ đệm chòm.

(2) Đa số các mô hình được công bố chưa sử dụng hiệu quả lỗi ước tính cho các lần tập hợp chòm tiếp theo. Trong [8], Mikoshi sử dụng lỗi ước tính để điều chỉnh trực tiếp độ dài ước tính cho lần tập hợp chòm tiếp theo. Tuy nhiên cách làm này là không tron, 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 chòm được hoàn thành, một số mô hình dựa trên thống kê độ dài chòm đo được (trong đề xuất của Sui) hay các ngưỡng thời gian trong M lần tập hợp chòm sau cùng nhất (như đề xuất của Jiang) hay tốc độ trung bình của M gói tin đến sau cùng nhất (của Fukushima). 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) Đa số các mô hình đã công bố sử dụng kỹ thuật tập hợp chòm dựa trên ngưỡng thời gian cố định (T_a) và sau đó cố gắng dự đoán độ dài chòm của lần tập hợp chòm hiện thời (L_e). Liu 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, Jiang 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 của các ngưỡng thời gian $T(i)$ của M lần tập hợp chòm trước đó, nên nó không phản ánh đúng xu hướng gần đây nhất của lưu lượng đến; hơn nữa việc điều chỉnh từng bước L_e sẽ không đáp ứng kịp với lưu lượng bùng phát và short peaks. Một vấn đề khác mà giải thuật Jiang phải đối mặt là việc chọn cặp giá trị L_{min} và L_{max} cho phù hợp như đã được chỉ ra trong Bảng 2.

5. KẾT LUẬN

Bài báo này đã tóm lược, phân tích và so sánh các sơ đồ tập hợp chòm giảm độ trễ đã được công bố, trong đó đề xuất của Jiang cho kết quả tập hợp chòm tốt nhất với lỗi ước tính tối thiểu. Tuy nhiên, các đề xuất này cũng đã bộc lộ các hạn chế về phương pháp ước tính, chi phí tính toán và cách xác định các giá trị ngưỡng cố định, mà cần có các nghiên cứu thêm và cải tiến để khắc phục các hạn chế này.

TÀI LIỆU THAM KHẢO

- [1]. A. Ge, F. Callegati, and L. Tamil (2000). On optical burst switching and self-similar traffic, *IEEE Communication Letters.*, vol. 4, issue. 3, pp. 98 – 100.
- [2]. C. Qiao and M. Yoo (1999). Optical burst switching (OBS)—a new paradigm for an optical internet, *Journal of High Speed Networks.*, Vol.8, pp. 69 – 84.
- [3]. H. Liu and S. Jiang (2012). A mixed-length and time threshold burst assembly algorithm based on traffic prediction in OBS network, *Int. Journal of Sensing, Computing & Control.*, Vol.2, No.2, pp. 87-93.
- [4]. Peterson and L. Larry (1996). Computer networks: a system approach, *Morgan Kaufmann*, pp. 552.
- [5]. S. Oh, and M. Kang (2002). A burst assembly algorithm in optical burst switching networks, *Proc. of Optical Fiber Communication Conference and Exhibit*, pp. 771 – 773.
- [6]. T. Hashiguchi et al. (2003). Burst assembly mechanism with delay reduction for OBS networks, *Proc. of Conf. on the Optical Internet.*, pp. 664-666.
- [7]. T. Mikoshi and T. Takenaka (2008). Improvement of burst transmission delay using offset time for burst assembly in optical burst switching, *Proc. of IEICE.*, pp. 13-18.
- [8]. X. Jiang, N. Zhu, and L. Yuan (2013). A novel burst assembly algorithm for OBS networks based on burst size and assembly time prediction, *Journal of Computational Information Systems.*, Vol.9, No.2, pp. 463–475.
- [9]. X. Yu, Y. Chen, and C. Qiao (2002). A study of traffic statistics of assembled burst traffic in optical burst switched networks, *Proc. of Opticomm.*, pp. 149 – 159.
- [10]. Yuan Chi, Huang Junbin, Li Zhenbin, and Xu Anshi (2005). A Novel Burst Assembly Algorithm for OBS networks CBased On Data-length Time-lag Product, *Proc. of Asia-Pacific Conf. on Communications.*, pp. 319 – 323.
- [11]. Y. Fukushima et al.(2009). A burst assembly method to reduce end-to-end delay in optical burst switching networks, *WSEAS Transactions on Communications*, Vol.8, Issue 8, pp. 894-903.
- [12]. Z. Sui, Q. Zeng, and S. Xiao (2005). An offset differential assembly method at the edge of OBS network, *Proc. of SPIE Optical Transmission, Switching and Subsystems III.*, Vol. 6021, pp. 1-6.

ANALYSIS OF THE ALGORITHMS OF BURST ASSEMBLY FOR DELAY REDUCTION AT THE EDGE NODE OF OBS NETWORKS

Le Van Hoa^{1*}, Vo Viet Minh Nhat¹, Nguyen Hoang Son²

¹ Department of Information Technology, Hue University College of Sciences

² Department of Mathematics, Hue University College of Sciences

*Email: levanhoa@hueuni.edu.vn

ABSTRACT

Burst assembly plays an important role in reducing the end-to-end delay of the packets when they are transported through optical burst switching (OBS) networks. There have been several proposals to reduce the delay of the packets assembled at ingress OBS nodes. The main idea of these proposals is to send the control packet early before the burst completion. However, since this information must be carried in the control packet, these proposals have given varied approaches for estimating the length of the completed burst. This paper analyzes and evaluates the algorithms of burst assembly for delay reduction based on the reduced delay rate and the estimation error between the completed burst length and the estimated burst length.

Keywords: *OBS network, ingress node, burst assembly, delay reduction, estimation error.*