

MỘT CÁI TIẾN VỀ ĐIỀU KHIỂN CHẤP NHẬN LẬP LỊCH TRONG MẠNG OBS CÓ XÉT ĐẾN QoS

Phạm Trung Đức

Trường Đại học Khoa học, Đại học Huế

Email: ptduc1988@gmail.com

Ngày nhận bài: 15/9/2017; ngày hoàn thành phản biện: 21/11/2017; ngày duyệt đăng: 8/01/2018

TÓM TẮT

Mạng chuyển mạch chùm quang (mạng OBS) được xem là công nghệ hứa hẹn cho Internet thế hệ tiếp theo, trong đó lập lịch là một trong những hoạt động có ảnh hưởng lớn đến hiệu năng của mạng OBS. Hiện đã có nhiều giải thuật lập lịch được đề xuất mà có thể được phân thành các loại như lập lịch trực tiếp, lập lịch nhóm, lập lịch kết hợp với các kỹ thuật xử lý tắc nghẽn. Trong mạng OBS có xét đến QoS, việc lập lịch cần phải xem xét thêm một số điều kiện ưu tiên và đã có một số công bố về điều khiển chấp nhận lập lịch. Trong bài viết này chúng tôi đề xuất một cải tiến về điều khiển chấp nhận lập lịch trong mạng OBS có xét đến QoS. Các phân tích và đánh giá hiệu quả dựa trên mô phỏng sẽ khẳng định ưu điểm của mô hình được cải tiến này.

Từ khoá: mạng OBS, QoS, điều khiển chấp nhận lập lịch.

1. GIỚI THIỆU

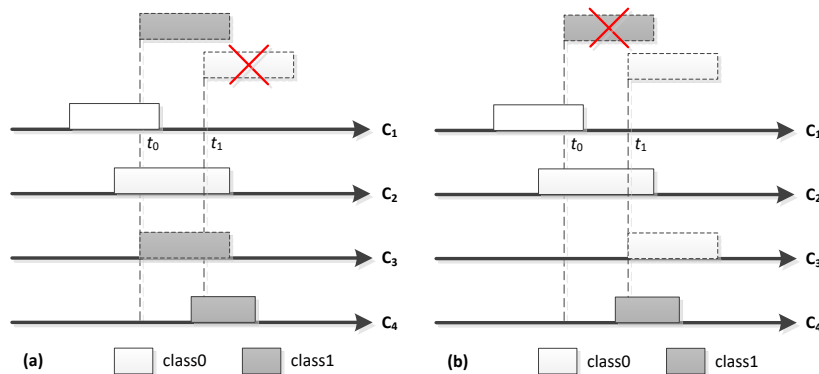
Chuyển mạch chùm quang (*Optical Burst Switching, OBS*) [1] được xem là một giải pháp chuyển mạch gói quang khả thi cho mạng Internet thế hệ tiếp theo, trong đó bằng thông các sợi quang có thể được khai thác một cách có hiệu quả nhờ vào công nghệ ghép phân chia kênh bước sóng (*Wavelength Division Multiplexing, WDM*) [2].

Có 2 loại nút mạng được phân biệt trong mạng OBS: nút biên và nút lõi. Các nút biên, bao gồm nút biên vào và nút biên ra, được xem là giao diện giữa miền điện tử và miền quang. Nút biên vào có nhiệm vụ nhận dữ liệu đến từ các mạng truy cập (chẳng hạn các gói tin IP), tập hợp chúng thành các chùm dữ liệu (*burst*) và truyền chúng vào mạng lõi. Một gói điều khiển sẽ được truyền đi trước một khoảng thời gian bù đắp (*offset time*) nhằm đặt trước tài nguyên và cấu hình các chuyển mạch tại các nút trung gian (nút lõi); nhờ đó chùm dữ liệu sẽ được chuyển mạch toàn quang khi đến tại mỗi nút trung gian này cho đến khi đạt đến nút biên ra của nó. Nút biên ra thực hiện

một hoạt động ngược lại với nút biên vào là khôi phục lại các dữ liệu (các gói IP) ban đầu.

Như các mạng chuyển mạch gói khác, tắc nghẽn cũng xảy ra trong mạng OBS, khi có hai hoặc nhiều chùm đến tại cùng một cổng ra, cùng một thời điểm và yêu cầu cùng một bước sóng. Trong trường hợp một chùm đến yêu cầu tài nguyên (bước sóng) tại một kênh ra mà đang bị chiếm giữ bởi một chùm khác đã lập lịch trước đó, đó cũng được xem là một trường hợp khác của tắc nghẽn. Các loại tắc nghẽn này là dễ dàng được xử lý trong mạng điện bằng cách sử dụng các bộ đệm (chẳng hạn, bộ nhớ RAM), nhưng nó trở nên khó khăn trong mạng quang vì công nghệ quang hiện nay chưa cho phép sản xuất các bộ đệm quang. Do vậy vấn đề tắc nghẽn trong mạng OBS thường được xử lý thông qua hoạt động lập lịch.

Việc lập lịch các chùm đến thông thường được thực hiện tuần tự, theo kiểu đến trước, phục vụ trước (*first come, first served*). Tuy nhiên trong môi trường có xét đến QoS, việc lập lịch thành công một chùm trước (có QoS thấp) có thể gây tắc nghẽn cho một chùm đến sau (có QoS cao hơn). Lấy ví dụ đối với trường hợp được nêu ra như trong Hình 1a, nếu chùm QoS thấp đến tại thời điểm t_0 được lập lịch và chùm QoS cao đến ngay sau đó (tại thời điểm t_1) thì chùm QoS cao sẽ không thể được lập lịch do hết tài nguyên. Nên để ưu tiên tài nguyên đối với chùm QoS cao, chùm QoS thấp đến tại t_0 cần được đánh rơi. Do đó, lập lịch với điều khiển chấp nhận là cần thiết nhằm để dành nhiều tài nguyên cho chùm QoS cao, trong khi hạn chế tài nguyên đối với các chùm QoS thấp.



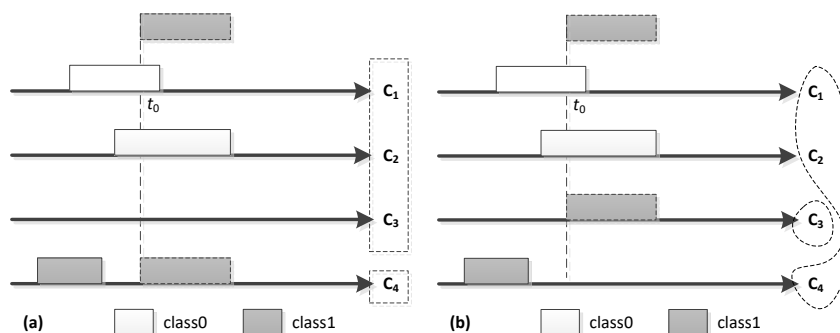
Hình 1. Một ví dụ về việc áp dụng điều khiển lập lịch trong đó chùm QoS thấp bị đánh rơi để dành tài nguyên cho chùm QoS cao đến sau (b) so với lập lịch đến trước, phục vụ trước (a).

Đã có một số kỹ thuật điều khiển lập lịch được đề xuất đối với mạng OBS có xét đến QoS, như kỹ thuật phân phối tài nguyên tĩnh (*Static Admission Control, SAC*) [3], phân phối tài nguyên động (*Dynamic Admission Control, DAC*) [3] và phân phối tài nguyên dựa trên mức tải (*Load-Level Admission Control, LLAC*) [4]. Tuy nhiên, các kỹ thuật này vẫn bộc lộ một số nhược điểm sau: đối với SAC và DAC, (1) tỉ lệ phân bổ các kênh bước sóng ra đối với các luồng chùm QoS cao và QoS thấp là cố định và mang

tính chủ quan của các tác giả đề xuất (2) hiệu quả sử dụng băng thông thấp, khi mà lưu lượng luồng chùm QoS cao đến thấp, trong khi lưu lượng luồng chùm QoS thấp đến cao; và (3) thông tin cần phải lưu trữ tại mỗi nút về trạng thái các kênh bước sóng là lớn, cần nhiều không gian nhớ. Đối với LLAC, mặt dù thông tin cần lưu trữ là ít hơn, nhưng (4) tỉ lệ tải được thiết lập cho các luồng chùm QoS cao và QoS thấp là cố định và không có căn cứ; hơn nữa (5) cách thiết lập tùy tiện này lại gây ra tỉ lệ đánh rơi chùm QoS thấp cao hơn so với DAC. Bài viết này sẽ đề xuất một mô hình cải tiến về điều khiển chấp nhận lập lịch mà phần nào giảm nhẹ được tất cả các nhược điểm trên.

2. CÁC NGHIÊN CỨU LIÊN QUAN

Các tác giả trong [3] đã đề xuất hai cơ chế điều khiển chấp nhận lập lịch cho mạng OBS có xét đến QoS, bao gồm kỹ thuật phân phối tài nguyên tĩnh và động. Cụ thể, xét một cổng ra có W kênh bước sóng khả dụng, kỹ thuật phân phối tĩnh (kỹ thuật SAC) sẽ phân bổ W_0 ($W_0 < W$) bước sóng cho các chùm QoS cao và W_1 ($W_1 < W$) bước sóng cho các chùm QoS thấp. Lưu ý rằng, $W_0 > W_1$ và $W_0 + W_1 = W$. Như ví dụ được chỉ ra trong Hình 2a, với $W = 4$ bước sóng ở cổng ra, kỹ thuật SAC phân bổ $W_0 = 3$ bước sóng gồm C_1 , C_2 , và C_3 dành riêng cho các chùm QoS cao và $W_1 = 1$ bước sóng còn lại, kênh C_4 , cho các chùm QoS thấp; nên một chùm QoS thấp đến tại thời điểm t_0 sẽ được lập lịch vì kênh C_4 rồi.



Hình 2. Ví dụ về kỹ thuật SAC (a) và DAC (b).

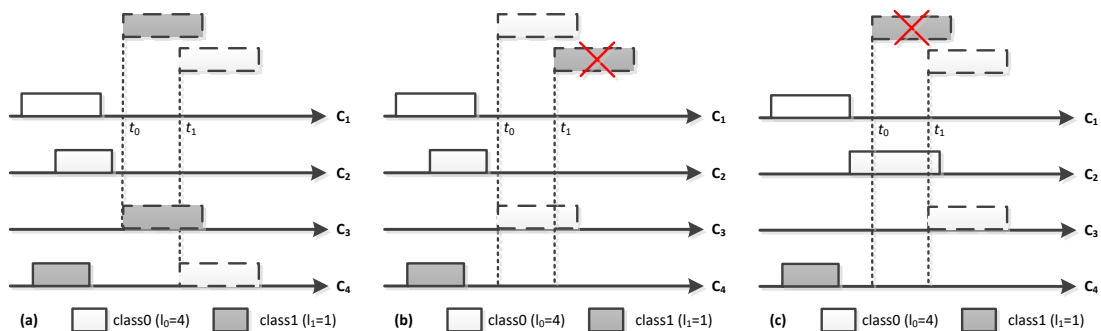
Ưu điểm của kỹ thuật SAC là đơn giản, đảm bảo dành riêng các kênh bước sóng cho các chùm QoS cao nên đảm bảo chất lượng dịch vụ. Tuy nhiên, trong trường hợp luồng các chùm QoS cao là thấp, trong khi luồng các chùm QoS thấp là cao, một sự lãng phí tài nguyên sẽ xuất hiện đối với các kênh bước sóng dành cho các chùm QoS cao, trong khi các chùm QoS thấp bị đánh rơi vì thiếu tài nguyên. Để linh hoạt hơn trong vấn đề phân bổ tài nguyên, các tác giả trong [3] đề xuất tiếp một kỹ thuật phân phối động.

Khác với kỹ thuật SAC, là các kênh bước sóng được dành riêng tuyệt đối (về mặt vật lý) cho mỗi loại chùm QoS cao và QoS thấp, kỹ thuật DAC chỉ dựa trên lượng

phân bố kênh bước sóng về mặt định lượng được cấp phát chủ quan ban đầu. Cụ thể là, xét tại thời điểm đến (t_0) của một chùm đến, nếu số chồng lắp (chiếm dụng chùm cùng thời điểm chùm đến trên các kênh bước sóng) của chùm này với các chùm cùng loại (ω hay ω_1) mà nhỏ hơn lượng bước sóng được phân bố ($\omega < W_0$ hay $\omega_1 < W_1$), việc lập lịch là được chấp nhận. Như ví dụ được chỉ ra trong Hình 2b, một chùm QoS thấp đến tại thời điểm t_0 vẫn có thể được lập lịch lên kênh C_3 hoặc C_4 vì số chồng lắp đối với loại chùm này là $\omega_1 = 0$, trong khi $W_1 = 1$, nên $\omega_1 < W_1$.

Ưu điểm của kỹ thuật DAC là linh hoạt hơn so với SAC trong việc sử dụng tài nguyên, nhưng cũng như SAC, kỹ thuật DAC cũng gây lãng phí tài nguyên đối với các kênh bước sóng dành cho các chùm QoS cao nếu mật độ các chùm này đến thấp, trong khi các chùm QoS thấp bị đánh rơi vì thiếu tài nguyên. Cả hai kỹ thuật SAC và DAC đều yêu cầu lưu lại thông tin trạng thái của tất cả các kênh bước sóng bị chiếm bởi chùm đến nên cần nhiều bộ nhớ cho việc lưu trữ.

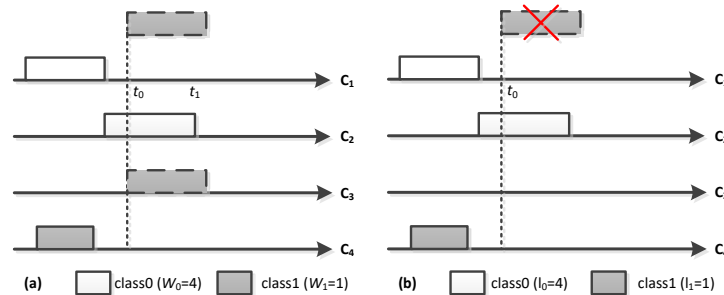
Để cải thiện về mặt sử dụng băng thông của các kênh bước sóng dành riêng và không gian lưu trữ, các tác giả trong [4] đã đề xuất một kỹ thuật phân phối dựa trên tải LLAC, trong đó chỉ có thông tin mức tải của mỗi lớp và tổng số bước sóng bị chiếm được lưu trữ. Các mức tải (l_0 hay l_1) sẽ được qui định trước cho mỗi lớp QoS cao/thấp; nếu số chồng lắp của một chùm đến với bất kỳ loại chùm nào đã được lập lịch trên các kênh ra (ω) là nhỏ hơn mức tải ($\omega < l_0$ hay $\omega < l_1$) chùm đến sẽ được lập lịch. Như ví dụ được chỉ ra trong Hình 3a, với hai mức tải $l_0 = 4$ và $l_1 = 1$ được thiết lập đối với 2 loại chùm QoS cao và QoS thấp, khi có một chùm QoS thấp đến tại thời điểm t_0 , nó được lập lịch vì $\omega < l_1$ ($\omega = 0$). Tại thời điểm t_1 , một chùm QoS cao đến và nó được lập lịch vì $\omega < l_0$ (lúc này $\omega = 1$). Tuy nhiên trong Hình 3b, với một chùm QoS cao đến tại thời điểm t_0 , nó được lập lịch vì $\omega < l_0$ ($\omega = 0$). Tại thời điểm t_1 , một chùm QoS thấp đến và nó không được lập lịch vì $\omega = l_1$ (lúc này $\omega = 1$). Tương tự với Hình 3c, một chùm QoS thấp đến tại thời điểm t_0 sẽ không được lập lịch vì $\omega = l_1$ ($\omega = 1$), trong khi một chùm QoS cao đến tại thời điểm t_1 vẫn được lập lịch vì $\omega < l_0$ ($\omega = 1$).



Hình 3. Ví dụ về cách thức hoạt động của LLAC.

Tuy nhiên, kỹ thuật LLAC có xu hướng đánh rơi các chùm không có QoS nhiều hơn so với kỹ thuật DAC. Như ví dụ trong Hình 4, kỹ thuật DAC sẽ không đánh rơi

chùm không có QoS khi nó đến tại thời điểm t_0 (Hình 4a); trong khi kỹ thuật LLAC sẽ đánh rơi nó vì $\omega = l_1$ ($\omega = 1$).



Hình 4. So sánh giữa DAC (a) và LLAC (b).

Tóm lại, cả 3 kỹ thuật SAC, DAC và LLAC có các ưu và nhược điểm như được chỉ ra trong Bảng 1.

Bảng 1. Ưu nhược điểm kỹ thuật SAC, DAC, LLAC

Kỹ thuật	Ưu điểm	Nhược điểm
SAC	- Đơn giản.	- Lưu nhiều thông tin trạng thái; - Lãng phí tài nguyên dành riêng cho QoS cao; - Tỷ lệ phân bổ kênh cố định, không có căn cứ.
DAC	- Phân bổ kênh linh hoạt hơn SAC; - Sử dụng bảng thông hiệu quả hơn SAC.	- Lưu nhiều thông tin trạng thái; - Tỷ lệ phân bổ kênh cố định, không có căn cứ.
LLAC	- Lưu ít thông tin trạng thái hơn SAC và DAC.	- Tỷ lệ xác định tải cố định, không có căn cứ; - Đánh rơi nhiều chùm không QoS hơn SAC và DAC.

Rõ ràng có một nhu cầu cần cải tiến các phương pháp phân phối tài nguyên này, mà sau đây sẽ mô tả chi tiết một cải tiến nhằm giảm bớt các nhược điểm của chúng.

3. ĐIỀU KHIỂN LẬP LỊCH DỰA TRÊN TỐC ĐỘ CHÙM ĐẾN

Phần này sẽ mô tả một thuật toán điều khiển lập lịch cải tiến (*improved Scheduling Admission Control, iSAC*) cho mạng OBS có hỗ trợ QoS. Ý tưởng cải tiến được chúng tôi đề xuất như sau:

- Các chùm QoS cao được lập lịch trên bất kỳ bước sóng nào tại một cổng ra ($W_0 = W$) và được lập lịch trực tiếp có lấp đầy khoảng trống;
- Số bước sóng mà các chùm QoS thấp có thể sử dụng ($W_1 \leq W$) được điều

Một cải tiến về điều khiển chấp nhận lập lịch trong mạng obs có xét đến QoS

chỉnh tăng/giảm định kỳ tùy thuộc vào tốc độ đến của chúng trong khoảng thời gian quan sát (w) $100\mu s$.

Cụ thể, gọi r_0 là tốc độ luồng các chùm QoS cao và r_1 là tốc độ luồng các chùm QoS thấp, số bước sóng cấp phát cho luồng các chùm QoS thấp (W_1) được xác định là phần nguyên cận trên của công thức sau:

$$W_1 = \left\lceil W_0 \times \frac{r_1}{r_0 + r_1} \right\rceil$$

Việc có được công thức trên lý do bởi tỷ lệ $\frac{r_1}{r_0 + r_1} \leq 1$ luôn tồn tại, thể hiện tỷ lệ tốc độ đến của lớp 1 trong toàn mạng và việc xác định số kênh dành riêng cho lớp ưu tiên thấp W_1 bằng tổng số kênh được cấp phát ($W_0=W$) nhân tỷ lệ tốc độ đến của lớp 1 trên toàn mạng rồi lấy phần nguyên cận trên là cách xác định thỏa mãn cho việc điều chỉnh giá trị W_1 phù hợp. Trong đó tốc độ các chùm (QoS cao và QoS thấp) được xác định dựa trên tốc độ chùm trung bình r_i^{avg} với giá trị khởi tạo ban đầu $r_i^{avg} = 0.5$, $i = \{0,1\}$, và tốc độ chùm hiện thời r_i^{cur} đến trong cửa sổ quan sát w . Mục đích của việc xem xét cả 2 tốc độ chùm trung bình và tốc độ chùm hiện thời là nhằm quan tâm đến các trường hợp có thay đổi đột ngột về tốc độ.

Tốc độ các chùm (QoS cao và QoS thấp) tiếp đó được tính như sau:

$$\begin{aligned} r_0 &= \alpha \times r_0^{avg} + (1 - \alpha) \times r_0^{cur} \\ r_1 &= \beta \times r_1^{avg} + (1 - \beta) \times r_1^{cur} \end{aligned}$$

, trong đó α, β ($0 \leq \alpha, \beta \leq 1$) là các trọng số thể hiện mức độ quan trọng giữa tốc độ trung bình và tốc độ hiện thời đối với việc dự đoán tốc độ chùm trong thời gian kế tiếp. Thông thường hệ số này được gán một cách chủ quan hoặc dựa trên kinh nghiệm, trong mô phỏng ở Phần 4, các trọng số này được chọn là $\alpha = \beta = 0.3$. Việc định ra một cửa sổ quan sát w nhằm để giảm khối lượng tính toán.

Lưu ý rằng, tốc độ chùm trung bình r_i^{avg} của lần tính toán tiếp theo chính là tốc độ chùm được dự đoán, tức là $r_i^{avg}(t+1) = r_i^{cur}(t)$. Tốc độ chùm hiện thời r_i^{cur} sẽ được xác định bằng $r_i^{cur} = \frac{n_i}{w}$, trong đó n_i là số chùm (QoS cao hay QoS thấp) đến trong một cửa sổ quan sát w .

Với cách tiếp cận nêu trên, rõ ràng thuật toán iSAC đảm bảo ưu tiên cho các chùm QoS cao, phân bổ linh hoạt hơn lượng bước sóng có thể sử dụng cho các chùm QoS thấp, giảm được số lượng thông tin lưu trữ số chùm tương ứng của mỗi loại bị chồng lấp trên các kênh đã phân bổ và không phụ thuộc vào việc phân bổ tải theo chủ quan của tác giả đề xuất hoặc dựa trên kinh nghiệm như trong SAC và LLAC. Độ phức

tạp thuật toán iSAC là $O(n \log_2 n)$.

Sau đây là mô tả chi tiết về thuật toán iSAC.

Thuật toán iSAC

Đầu vào: - Tập các chùm đến $B_{ub} = \{ub_i | i = 1 \dots N\}$, $ub_i = \{s_{ub}, e_{ub}, prio_{ub}\}$, trong đó s_{ub} và e_{ub} là thời gian đến và kết thúc, $prio_{ub} = \{0, 1\}$ xác định chùm ub là QoS cao (0) hay thấp (1).

Đầu ra: - Tập các chùm QoS cao được lập lịch, S_0 , và bị đánh rơi, D_0 ;
- Tập các chùm QoS thấp được lập lịch, S_1 , và bị đánh rơi, D_1 ;

Phương pháp

```

1   $time\_last \leftarrow 0; w \leftarrow 100;$  // xác định cửa sổ quan sát ( $\mu s$ )
2   $ch \leftarrow -1; \alpha \leftarrow 0.3; \beta \leftarrow 0.3;$ 
3  while ( $B_{ub} \neq \emptyset$ ) do; //
4     $ub \leftarrow$  chùm đầu tiên từ tập  $B_{ub}$ 
5     $B_{ub} \leftarrow B_{ub} \setminus \{ub\};$  // duyệt chùm đến tiếp theo
6    if ( $prio_{ub} = 0$ ) then
7       $n_0 \leftarrow n_0 + 1;$  // đếm số chùm QoS cao.
8       $ch \leftarrow BFVF(ub, W);$ 
9      if ( $ch \neq -1$ ) then
10        $S_0 \leftarrow S_0 + 1;$  // Lập lịch chùm  $ub$  trên kênh  $ch$  và tăng  $S_0$ 
11     else
12        $D_0 \leftarrow D_0 + 1;$  // Loại bỏ chùm  $ub$ 
13     end if
14   end if
15   if ( $prio_{ub} = 1$ ) then
16      $n_1 \leftarrow n_1 + 1;$  // đếm số chùm QoS thấp
17      $time\_cur \leftarrow$  thời gian hiện thời;
18     if ( $time\_cur - time\_last = w$ ) then
19        $r_0^{cur} \leftarrow \frac{n_0}{w};$ 
20        $r_1^{cur} \leftarrow \frac{n_1}{w};$ 
21        $r_0 \leftarrow \alpha \times r_0^{avg} + (1 - \alpha) \times r_0^{cur};$ 
22        $r_1 \leftarrow \beta \times r_1^{avg} + (1 - \beta) \times r_1^{cur};$ 

```

Một cải tiến về điều khiển chấp nhận lập lịch trong mạng obs có xét đến QoS

```

23       $W_1 \leftarrow \left\lceil W_0 \times \frac{r_1}{r_0 + r_1} \right\rceil$ 
24       $n_0 \leftarrow 0; n_1 \leftarrow 0;$ 
25       $time\_last \leftarrow time\_cur;$ 
26       $r_0^{avg} \leftarrow r_0; r_1^{avg} \leftarrow r_1;$ 
27  end if
28   $ch \leftarrow BFVF(ub, W_1);$ 
29  if ( $ch \neq -1$ ) then
30       $S_1 \leftarrow S_1 + 1;$ 
31  else
32       $D_1 \leftarrow D_1 + 1;$ 
33  end if
34  end if
35   $time\_wait = time\_cur;$ 
36  end while

```

Hàm BF-VF(ub, W) có nhiệm vụ lập lịch chùm ub lên một trong W kênh: nếu lập lịch thành công, hàm BF-VF [5] trả về kênh được chọn để lập lịch cho ub ; nếu việc lập lịch không thành công, hàm BF-VF trả về -1.

Hàm BFVF(ub, W)

Đầu vào: - $ub(s_{ub}, e_{ub})$ chùm đến;

- W kênh bước sóng

Đầu ra: - Kênh được lựa chọn $best_channel$.

Phương pháp

```

1   $best\_utilisation \leftarrow \infty;$ 
2   $best\_channel \leftarrow -1;$ 
3  for each  $k \in W$  do
4       $e_{0,k} \leftarrow 0;$ 
5       $SB_k \leftarrow$  tập các chùm đã lập lịch trên kênh  $k$ 
6      for each  $sb_{i,k} \in SB_k$  do
7          if ((( $s_{ub} \geq e_{i,k}$ )  $\wedge$  ( $s_{i+1,k} \geq e_{ub}$ )  $\wedge$  ( $s_{i+1,k} - e_{i,k} < best\_utilisation$ ))) then
8               $best\_utilisation \leftarrow s_{i+1,k} - e_{i,k};$ 
9               $best\_channel \leftarrow k;$ 
10         end if;
11     end for;

```


12 end for;

13 return *best_channel*;

4. MÔ PHÒNG VÀ PHÂN TÍCH KẾT QUẢ

Các thuật toán SAC, DAC, LLAC và iSAC được cài đặt máy tính có CPU 2,4 GHz Intel Core 2, 2G RAM. Xét có 2 loại luồng đến, luồng QoS cao (r_0) và luồng QoS thấp (r_1). Giả thiết các chùm đến có phân phối Poisson và kích thước các chùm có phân phối hàm số mũ. Các chùm này được sinh ra từ gói mô phỏng mạng obs 0.9a [7] của hệ mô phỏng NS2.

Mục tiêu mô phỏng là so sánh hiệu suất của các thuật toán SAC, DAC, LLAC và iSAC với trường hợp lưu lượng tải tăng của luồng chùm QoS thấp đến tại một cổng ra của một nút mạng OBS với chùm QoS cao và băng thông sử dụng của các giải thuật để chỉ ra ưu điểm của đề xuất. Lưu ý rằng, tải chuẩn hóa được tính xác định như sau:

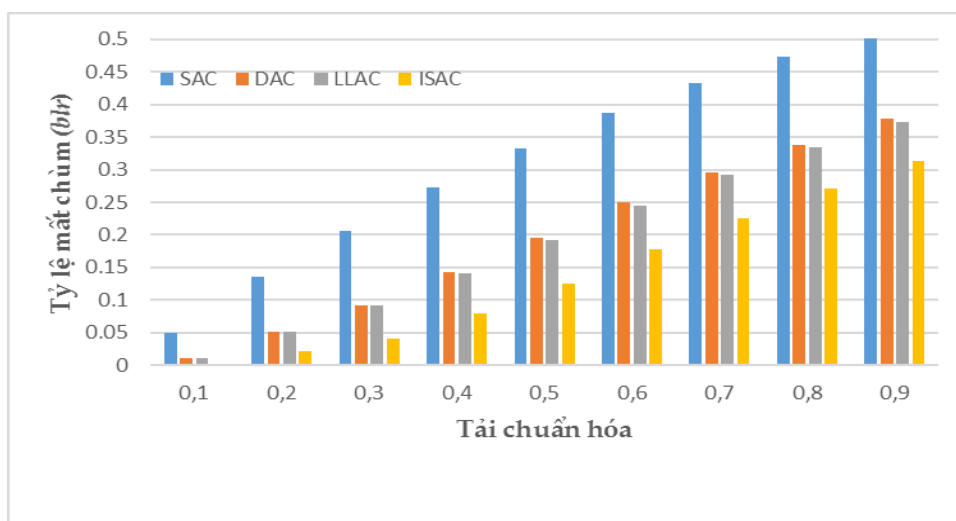
Tải chuẩn hóa = (tốc độ chùm đến $\times$ độ dài chùm trung bình) / băng thông các kênh ra.

Như vậy Tải chuẩn hóa $\in [0,1]$.

Hiệu suất của các thuật toán được đo bằng tỉ lệ mất chùm (*burst loss rate, blr*), được xác định như sau:

$blr = \text{số chùm bị đánh rơi} / \text{tổng số chùm đến}$.

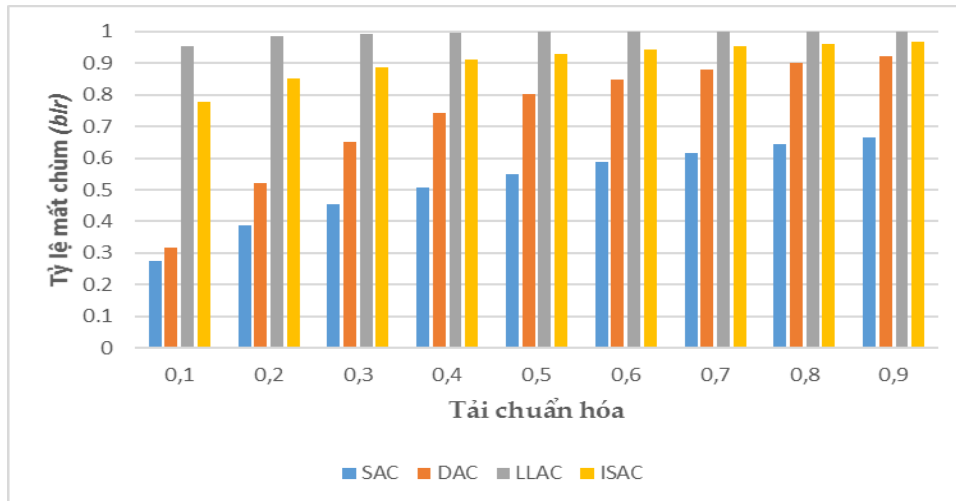
Như mô tả trong Hình 5 về tỉ lệ mất chùm của lớp QoS cao, thuật toán SAC cho kết quả mất chùm cao nhất, trong khi thuật toán đề xuất của chúng tôi, iSAC cho tỉ lệ mất chùm thấp nhất. Kết quả mô phỏng này cũng phản ánh được các cài đặt đã được thực hiện trong [4], trong đó LLAC cho kết quả lập lịch tốt hơn so với DAC và SAC.



Hình 5. So sánh tỉ lệ mất chùm lớp QoS cao giữa SAC, DAC, LLAC và iSAC.

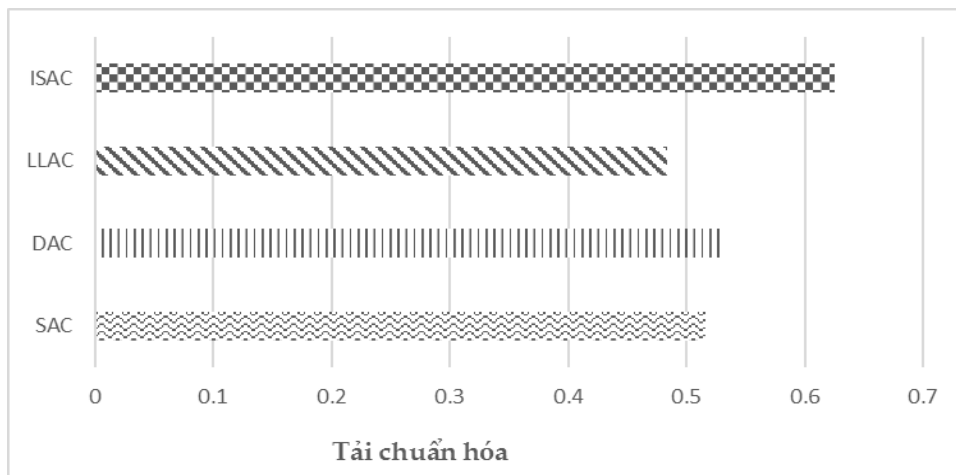
Một cải tiến về điều khiển chấp nhận lập lịch trong mạng obs có xét đến QoS

Xét đối với lớp QoS thấp, tỉ lệ mất chùm của iSAC có thấp hơn LLAC như tại Hình 6, nhưng cao hơn so với SAC và DAC. Thực tế luôn có một sự hài hòa giữa tỉ lệ mất chùm, độ phức tạp giải thuật và lượng thông tin cần lưu trữ. Như đã được phân tích trong [4], lượng thông tin cần lưu trữ của LLAC ít hơn so với SAC và DAC; nhưng đối với iSAC, như đã phân tích trong Mục 3, thuật toán được đề xuất của chúng tôi có tỉ lệ mất chùm và lượng thông tin cần lưu trữ thấp hơn so với LLAC, nên cũng chứng tỏ được ưu điểm của iSAC.



Hình 6. So sánh tỉ mất chùm QoS thấp giữa SAC, DAC, LLAC và iSAC.

Cuối cùng, khi xét đến hiệu quả của việc sử dụng băng thông trên tải toàn mạng thì giải thuật đề xuất cũng cho kết quả dùng băng thông tốt hơn nhiều so với các giải thuật trước đó gồm có: SAC, DAC và LLAC như phân tích trong Hình 7, càng khẳng định được ưu điểm của giải thuật được đề xuất.



Hình 7. Băng thông sử dụng trung bình của các giải thuật SAC, DAC, LLAC và iSAC.

5. KẾT LUẬN

Trong bài báo này, một thuật toán lập lịch thích nghi iSAC đã được đề xuất. iSAC dựa trên thuật toán lập lịch tối ưu khoảng trống BF-VF [5] và việc cấp phát tài nguyên cho chùm ưu tiên thấp dựa trên sự thay đổi tương ứng trong một khoảng thời gian quan sát tốc độ đến của các loại chùm đến. Dựa trên kết quả mô phỏng và phân tích, iSAC thể hiện được các ưu điểm thông qua tỉ lệ mất chùm thấp hơn và lượng thông tin cần lưu trữ (không lưu trữ số chồng lấp của từng loại chùm trên các kênh đã cấp phát) ít hơn so với giải thuật đã công bố trước đây.

TÀI LIỆU THAM KHẢO

- [1]. Costa, L.H.M.K., Fdida, S., Duarte (2006). O.C.M.B.: Incremental service deployment using the Hop By Hop multicast routing protocol. *IEEE/ACM Trans. Netw.* 14(3), 543–556.
- [2]. C. Qiao et al (1999). Optical burst switching (OBS) - A new paradigm for an optical Internet, *Journal of High Speed Networks*, vol.8, pp.69-84.
- [3]. Igor M. Moraes et al (2008). An efficient admission control mechanism for optical burst-switched networks, *Springer Science+Business Media*, LLC.
- [4]. Igor M. Moraes and O. C. M. B. Duarte (2010). Using the Network Load for Admission Control in OBS Networks: A Multilink Approach, *J. Opt. Commun. Netw*, Vol 2, No. 3.
- [5]. M. Nandi, A. K. Turuk, D. K. Puthal and S. Dutta (2009). Best Fit Void Filling Algorithm in Optical Burst Switching Networks, *IEEE Second International Conference on Emerging Trends in Engineering and Technology*, 609-614.
- [6]. Qiao Zhang (2004). Absolute QoS Differentiation in Optical Burst-Switched Network, *IEEE*, Vol 22, No 9.
- [7]. The package of NS-obs0.9a: <http://www.wine.icu.ac.kr/obsns/>.

AN IMPROVEMENT OF SCHEDULING ADMISSION CONTROL IN QoS SUPPORTING OBS NETWORKS

Pham Trung Duc

University of Sciences, Hue University

Email: ptduc1988@gmail.com.

ABSTRACT

Optical Burst Switched Networks (OBS network) is considered as a promising technology for the next generation Internet, in which scheduling is one of the most influential activities to efficiency of the OBS network. There have been already many proposed scheduling algorithms that can be classified into categories such as direct scheduling, group scheduling, scheduling combined with congestion processing techniques. In the OBS network, if there is referred to QoS, scheduling must be considered further about some priority conditions and therefore there have been some releases of scheduling admission controls. In this article, we propose an improvement about scheduling admission controls in OBS networks including in QoS. The effective analyses and evaluations based on simulation will confirm the strengths of this proposed new model.

Keywords: admission control scheduling, OBS Networks, QoS.



Phạm Trung Đức sinh ngày 10/08/1988 tại Thừa Thiên Huế. Năm 2010, ông tốt nghiệp Cử nhân Công nghệ thông tin tại Trường Đại học Khoa học, Đại học Huế. Năm 2012, ông nhận bằng Thạc sĩ chuyên ngành Khoa học máy tính tại Trường Đại học Khoa học, Đại học Huế. Từ tháng 12/2016 đến nay, ông là nghiên cứu sinh tại Trường Đại học Khoa học, Đại học Huế.

Lĩnh vực nghiên cứu: Mạng chuyển mạch chùm quang OBS, đa dạng dịch vụ QoS, điều khiển chấp nhận.