

# GIẢI PHÁP CUNG CẤP TÀI NGUYÊN TRUYỀN THÔNG PHÂN TÁN

Đặng Hùng Vĩ<sup>1</sup>, Lê Văn Sơn<sup>1</sup>

<sup>1</sup> Trường Đại học Sư phạm – Đại học Đà Nẵng

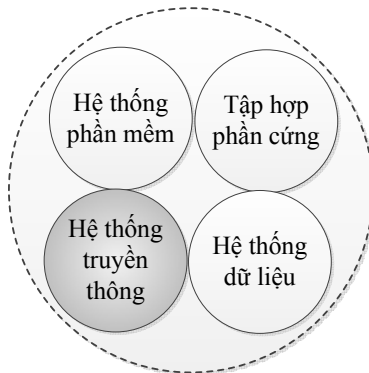
dhungvi@ued.udn.vn, levansupham2004@yahoo.com

**TÓM TẮT** - Hiện nay, hệ phân tán đã và đang được nghiên cứu, triển khai trong các hệ thống ảo hóa. Các giải pháp kỹ thuật đảm bảo tính gắn bó dựa vào hợp lực giữa các máy chủ bên trong hệ. Hợp lực trong hệ phân tán chủ yếu sử dụng cơ chế truyền thông điệp (message passing) trong mạng. Nhược điểm của vấn đề hợp lực là các thông điệp được tiếp nhận không theo trật tự bởi độ trễ truyền thông gây ảnh hưởng đến xử lý trong hệ. Bên cạnh đó, thông điệp truyền trong môi trường truyền thông có thể bị mất, phân mảnh và trùng lặp. Do đó, cần phải xây dựng bộ cung cấp tài nguyên truyền thông để đảm bảo trật tự và tối ưu truyền thông điệp. Vì vậy, chúng tôi đề xuất cải tiến các thuật toán cung cấp tài nguyên truyền thông phân tán trong nghiên cứu của mình áp dụng cho hệ phân tán.

**Từ khóa** - Hệ phân tán, lưu lượng cực đại, cung cấp tài nguyên truyền thông, truyền thông điệp.

## I. ĐẶT VẤN ĐỀ

Hệ phân tán và các ứng dụng của hệ được nghiên cứu và triển khai trên với quy mô lớn, đáp ứng số lượng lớn người dùng truy cập. Bốn thực thể của hệ phân tán thể hiện Hình 1, hệ thống này bao gồm các máy chủ kết nối qua hệ thống truyền thông dưới sự điều hành thống nhất của một hệ điều hành gọi là hệ điều hành phân tán [1].



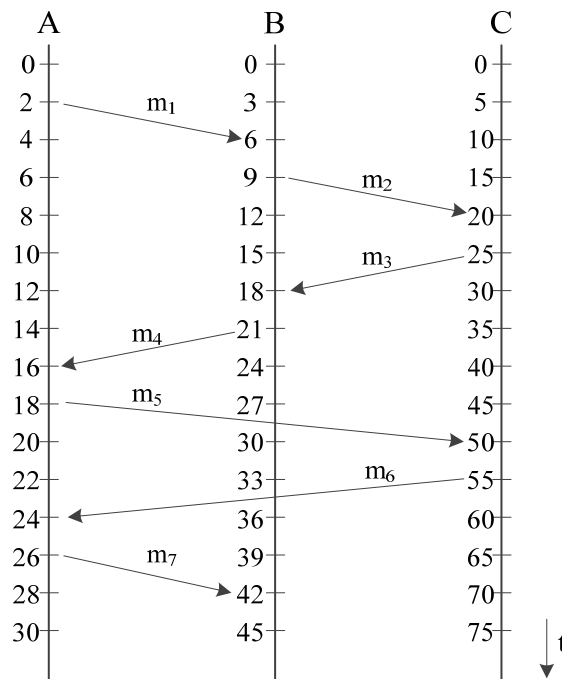
Hình 1. Bốn thực thể của hệ tin học phân tán

Các hoạt động của hệ thống phân tán bao gồm tập các tiến trình cung cấp tài nguyên dùng chung được trao đổi bởi môi trường truyền thông. Độ trễ trong mạng truyền thông là hữu hạn nhưng không thể đoán trước và các bộ vi xử lý không chia sẻ một bộ nhớ chung toàn cục; giao tiếp duy nhất trong hệ bằng cách truyền thông điệp qua mạng [2]. Do đó, môi trường truyền thông có thể cung cấp các thông điệp không theo trật tự; thông điệp có thể bị mất, bị phân mảnh, hoặc trùng lặp do độ trễ, hết thời gian chờ và yêu cầu phát lại. Bên cạnh đó, các tiến trình có thể thất bại và các liên kết truyền thông có thể mất kết nối. Hệ phân tán không có đồng hồ vật lý toàn cục để các tiến trình có thể truy cập tức thời, do đó hệ phải sử dụng đồng hồ riêng là đồng hồ lô gic để gán giá trị và đồng bộ tiến trình trên các máy chủ hay còn gọi là nhãn thời gian lô gic. Tuy nhiên, các nhãn đồng hồ này phải được cập nhật và nhất quán trên tất cả các trạm, nếu không việc xử lý thông điệp sẽ bị sai và hoạt động của hệ bị sai trật tự theo lý thuyết trật tự như Hình 2 [1].

Qua đó cho thấy, thực thể hệ thống truyền thông theo Hình 1 đóng vai trò quan trọng trong quá trình thiết kế các ứng dụng và điều khiển hệ phân tán. Các thực thể khác như hệ thống phần mềm, dữ liệu đóng vai trò điều hành và đảm bảo tính gắn bó giữa các máy chủ thông qua hợp lực. Các đặc tính của hệ phân tán thể hiện như sau:

- **Đáp ứng truy cập tài nguyên số lượng lớn** là hệ thống đảm bảo tính toán số lượng truy cập người dùng, hệ thống này sẽ cảnh báo tắc nghẽn và chuyển truy cập người dùng sang máy chủ khác có khối lượng xử lý ít hơn để đảm bảo truy cập. Hệ thống cảnh báo này gọi là bộ phân phối tải.
- **Tính trong suốt phân tán** thể hiện ở đặc tính người dùng chỉ quan tâm là tài nguyên mình đăng ký truy cập có dễ dàng và có được chấp nhận hay không mà không cần quan tâm đến những hoạt động diễn ra bên trong hệ.
- **Tính gắn bó** là một trong những đặc điểm quan trọng của hệ, một tài nguyên dùng chung chỉ cung cấp duy nhất và trạng thái này phải giống nhau trên tất cả các máy chủ. Để đảm bảo tính gắn bó này, sự hợp lực giữa các máy chủ phải giải quyết được tương tranh, phòng tránh bế tắc và thống nhất giữa các máy chủ, đồng thời thực hiện trên cùng một giải thuật.
- **Tính mở** là khả năng tăng các máy chủ xử lý khi số lượng truy cập của các máy chủ đã đạt tới hạn có khả năng xảy ra tắc nghẽn bởi các máy chủ không thể tự tăng tài nguyên của mình lên được nữa do giới hạn về khả năng phân cứng.

- **Tính chịu lỗi** là khi một hay nhiều máy chủ bị sự cố rời khỏi và vào lại hệ sau khi đã khôi phục thì yêu cầu **tính gắn bó** phải đảm bảo cho toàn hệ. Hệ dùng khi tất cả các máy chủ trong hệ dùng hoạt động.



**Hình 2.** Nhân thời gian thông điệp không theo trật tự

Hệ phân tán phụ thuộc vào môi trường truyền thông có độ trễ là một trong những vấn đề quan tâm đó là trật tự thông điệp tiếp nhận và xử lý trên các máy chủ. Các trật tự này phải đảm bảo nhất quán giữa các máy chủ sao cho quá trình thực thi thông điệp dẫn đến gắn bó trong toàn hệ.

Đối với hệ phân tán phức tạp kết nối ngang hàng chia sẻ tài nguyên dùng chung, khi một tài nguyên được cung cấp cho người dùng thì tất cả máy chủ khác đều nhận biết và cung cấp tài nguyên duy nhất. Dựa trên nguyên lý này, hệ phân tán có thể cố định số lượng hoặc tăng lên tùy thuộc vào nhu cầu truy cập thực tế từ người dùng. Tính phức tạp của hệ tăng lên khi tài nguyên gần tới hạn dẫn đến tương tranh càng lớn, hệ rất dễ xảy ra bế tắc và thiếu thốn tài nguyên vô hạn.

Theo nguyên lý thiết kế các chiến lược đồng bộ hóa các tiến trình trong [1] đưa ra các chiến lược cung cấp tài nguyên trong hệ phân tán nhằm đảm bảo tính gắn bó trên cơ sở nhờ dấu. Để thực thi giải pháp vừa nêu trong hệ, các máy chủ liên lạc với nhau theo nhóm để thực hiện truyền thông thông điệp, đây chính là phương thức truyền multicast [3-6]. Tuy nhiên, hai thách thức cơ bản được đưa ra trong nghiên cứu giải pháp cung cấp tài nguyên truyền thông phân tán là *trùng lặp thông tin tại tập đích* và *trùng lặp đường dẫn định tuyến* [7, 8]. Như vậy, cần thiết phải xây dựng *bộ cung cấp tài nguyên truyền thông* để giải quyết hai hạn chế vừa nêu.

Bộ cung cấp tài nguyên truyền thông phân tán tập trung vào các xử lý cơ bản là cung cấp giá trị đồng hồ lô gic, điều khiển lưu lượng và định tuyến đường dẫn truyền thông điệp. Bộ cung cấp này cho phép các thuật toán được xây dựng và phát triển tham gia vào quá trình điều khiển nhằm tối ưu truyền thông điệp. Truyền thông điệp trong hệ phân tán thuộc bài toán NP-khó [8], do đó các giải pháp tối ưu chỉ mang giá trị đúng cho bài toán đang xét.

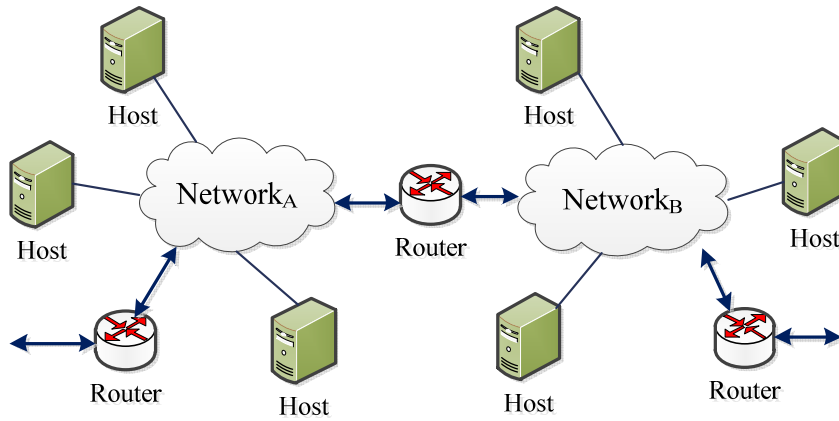
Truyền thông trong hệ phân tán có thể được mô hình hóa như một đồ thị có hướng, trong đó đỉnh đại diện cho máy chủ xử lý các tiến trình vào/ra và các cạnh đại diện cho các kênh truyền thông. Các hướng nghiên cứu của giải pháp tối ưu truyền thông phân tán tập trung vào bài toán tối ưu lưu lượng mạng từ nguồn đến đích. Giải pháp giải quyết cho bài toán này chủ yếu là giải pháp tối ưu dựa trên cây [5]. Hai bài toán cho giải pháp tối ưu trên cây là bài toán tìm luồng cực đại và đường đi ngắn nhất.

Hướng nghiên cứu của chúng tôi là tìm giải pháp đảm bảo lưu lượng cực đại trong cung cấp tài nguyên truyền thông phân tán; bên cạnh đó các thông điệp được gắn dấu nhờ vào bộ cung cấp giá trị đồng hồ lô gic. Bộ cung cấp tài nguyên truyền thông phân tán phát triển dựa trên các thuật toán phân tán Lamport, Ford Fulkerson để cung cấp giá trị đồng hồ lô gic, tính toán luồng cực đại khi truyền thông điệp.

## II. GIẢI PHÁP TỐI ƯU TRUYỀN THÔNG NHÓM TRONG HỆ PHÂN TÁN

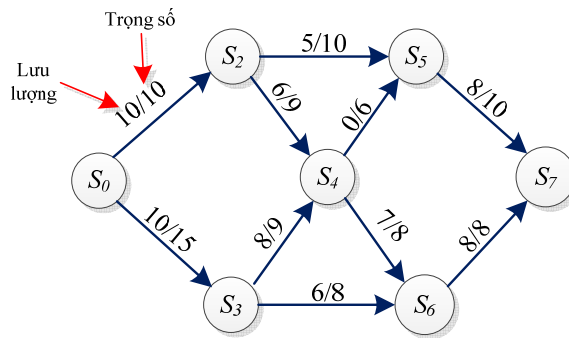
Một hệ thống truyền thông phân tán được mô tả theo Hình 3 là tập các máy chủ (Host) kết nối qua môi trường truyền thông và định tuyến thông qua các router. Mỗi máy chủ là một hệ thống độc lập có bộ nhớ và vi xử lý riêng, sự hợp lực của các máy chủ thông qua truyền thông điệp để thực hiện nhiệm vụ chung. Các thông điệp này cũng có thể

được xét như là các tiến trình di chuyển trong hệ thống mạng. Việc thực thi các tiến trình phụ thuộc vào bộ cung cấp tài nguyên truyền thông để định tuyến từ nguồn đến đích.



Hình 3. Mạng truyền thông phân tán

Hệ thống mạng ở Hình 3 có thể mô tả dưới dạng đồ thị  $G=(U, V)$  theo Hình 4, trong đó  $U$  là tập các  $S_i$  và  $V$  là tập giữa các  $S_{ij}$  trong hệ,  $S_{ij}$  là liên kết giữa hai nút  $S_i$  và  $S_j$  liên kề trong hệ thống. Với mỗi  $S_{ij}$  có trọng số nguyên dương  $ts_{S_{ij}} \geq 0$ . Ký hiệu  $P$  là tập các phiên chuyển thông điệp trong hệ, đối với mỗi phiên  $p \in P$  ta có đường dẫn từ  $S_0 \in U$  đến  $S_{|U|-1} \in U - \{S_0\}$  tương ứng là nguồn và đích. Đối với hệ thống đang xét, ta có tập đích  $D$  thì  $S_{|U|-1} \subset D$ . Ký hiệu lưu lượng  $l^p$  của một phiên  $p$  truyền trong mạng là một giá trị nguyên dương, ta có  $0 \leq l^p \leq ts_{S_{ij}}$ .



Hình 4. Hệ thống mạng biểu diễn dưới dạng đồ thị

Để tối ưu giải pháp truyền thông trong hệ phân tán, truyền thông nhóm được đề xuất để truyền thông điệp từ nguồn đến đích [9-11]. Truyền thông nhóm là một tập hợp các các tiến trình chia sẻ ngữ cảnh và hợp lực trên một nhiệm vụ chung trong miền ứng dụng phân tán. Các thuật toán cụ thể cần phải được thiết kế để cho phép truyền thông và quản lý nhóm hiệu quả, bên cạnh đó các tiến trình có thể tham gia và rời khỏi nhóm tự động. Khi nhiều thông điệp phát đồng thời, những bộ nhận trên các máy chủ có thể nhận được các thông điệp trong trật tự khác nhau; do đó, có thể phá vỡ hoạt động của chương trình phân tán nếu máy chủ xử lý các thông điệp không theo trật tự này. Vì vậy, vai trò bộ cung cấp trật tự cần phải được xây dựng và thực hiện để cung cấp trật tự cho thông điệp đi và đến trong hệ.

Truyền thông nhóm áp dụng cho việc xây dựng chiến lược cung cấp tài nguyên trong hệ thống phân tán với các đặc điểm sau:

- *Dịch vụ chịu lỗi dựa trên bản sao*: dịch vụ bản sao bao gồm một nhóm các máy chủ. Yêu cầu máy chủ là truyền thông điệp multicast cho tất cả các thành viên của nhóm, các thông điệp này thực thi cùng một thuật toán. Khi một số thành viên bị lỗi, máy khách truy cập tài nguyên vẫn có thể được phục vụ.
- *Dịch vụ chuyển kết nối*: thông điệp multicast có thể được sử dụng bởi các máy chủ và máy khách để xác định vị trí dịch vụ trong đăng ký tài nguyên dùng chung. Khi một máy chủ bị sự cố hoặc quá tải, hệ thống sẽ chuyển đổi truy cập từ máy khách sang một máy chủ khác có khả năng xử lý cao hơn để đảm bảo trong quá trình đăng ký tài nguyên không ảnh hưởng đến quá trình xử lý.
- *Hiệu năng cao thông qua nhân bản dữ liệu*: Dữ liệu được nhân bản để tăng hiệu suất của việc cung cấp tài nguyên. Mỗi lần thay đổi dữ liệu, giá trị mới được truyền multicast cho các tiến trình quản lý các bản sao để cùng cập nhật lại.

- *Nhân bản các thông báo sự kiện*: truyền multicast cho một nhóm có thể được sử dụng để thông báo cho tiến trình khi diễn ra các hoạt động trên máy chủ. Điều này quan trọng đối với việc ứng dụng thuật toán Lamport để cung cấp giá trị đồng hồ logic, thiết lập trật tự tổng quát và thực hiện xử lý tuần tự trên các máy chủ.

Truyền thông nhóm thực hiện truyền thông điệp dựa vào đường dẫn định tuyến đã biết để truyền từ nguồn đến đích. Truyền thông nhóm xét hai khía cạnh đó là lưu lượng và các kênh truyền thông. Lưu lượng thể hiện giá trị được cấp phát để truyền thông điệp, trong khi đó các kênh truyền thông được xem xét như là tài nguyên của bộ cung cấp tài nguyên truyền thông để thông điệp đi qua. Đối với các máy chủ có nhiều kênh truyền thông thì hiệu quả truyền tốt hơn, tuy nhiên nó còn phải phụ thuộc vào lưu lượng được cấp phát. Bộ cung cấp tài nguyên truyền thông phân tán trên cơ sở truyền thông nhóm theo Hình 5, tại mỗi máy chủ có hai thành phần đó là hàng đợi và bộ cung cấp tài nguyên truyền thông. Hàng đợi trong các máy chủ nhằm cho phép tiếp nhận các thông điệp đến thể hiện như các tiến trình yêu cầu tài nguyên dùng chung. Các tiến trình này được xử lý và sắp xếp theo giá trị đồng hồ, do đó nó có thể giải quyết vấn đề tranh tài nguyên. Bộ cung cấp tài nguyên truyền thông trên mỗi máy chủ trong hệ đảm nhiệm vai trò vừa yêu cầu, vừa đáp ứng yêu cầu từ các server khác trong hệ. Mục đích của truyền thông là đảm bảo lưu lượng cục đại khi qua các nút để thông tin đến đích chính xác và nhanh chóng. Để giải quyết bài toán này, lý thuyết đồ thị được đề xuất dựa vào tính toán lưu lượng trên các kênh truyền.



Hình 5. Cung cấp tài nguyên phân tán cho cặp yêu cầu/đáp ứng

## A. Các thuật toán truyền thông nhóm

### 1. Thuật toán Lamport

Thuật toán Lamport nhằm cho phép ghi lại các sự kiện của hệ phân tán [1, 11, 12]. Thuật toán tập trung vào nguyên lý sau: mỗi trạm  $s$  đều có trang bị công tơ với các giá trị nguyên gọi là  $H_s$ . Đó chính là đồng hồ lô gic tăng lên giữa hai sự kiện kế tiếp. Trạm  $e$  phát thông điệp ghi dấu  $E$  của mình dựa trên giá trị hiện hành của  $H_e$ . Khi nhận được thông điệp, trạm nhận  $r$  cập nhật đồng hồ  $H_r$  riêng của mình bằng giải thuật rút gọn theo công thức (1):

$$\begin{aligned} &\text{Nếu } H_r \leq E, \text{ thì} \\ &H_r := E + 1 \\ &\text{Chấm dứt nếu} \end{aligned} \quad (1)$$

Một sự kiện  $a$  sinh ra trong trạm  $i$  và được đánh dấu bởi đồng hồ cục bộ gọi là  $H_i(a)$ . Nếu  $a$  và  $b$  đều là hai sự kiện trên hai trạm  $i$  và  $j$ , ta luôn luôn có quan hệ xác định theo công thức (2) như sau :

$$a \rightarrow b \Leftrightarrow H_i(a) < H_i(b) \quad (2)$$

### 2. Thuật toán Ford Fulkerson

Thuật toán Ford Fulkerson là thuật toán cho phép tính toán luồng cực đại trong một mạng truyền thông [13]. Thuật toán này cũng thường được sử dụng cho thuật toán Edmonds-Karp, một trường hợp đặc biệt của thuật toán Ford-Fulkerson.

Cách giải quyết của thuật toán là tồn tại một đường dẫn từ nguồn đến đích với điều kiện tất cả các cạnh trên đường dẫn đó vẫn còn khả năng thông qua, chúng sẽ tìm lưu lượng gửi đi dọc theo đường đi đó, sau đó chúng tìm một đường đi khác và lặp lại thuật toán. Một đường dẫn còn khả năng thông qua là một đường dẫn có khả năng mở rộng thêm hay một đường dẫn mà lưu lượng qua đó còn khả năng tăng thêm. Kết thúc thuật toán ta đạt được một đường dẫn với lưu lượng có giá trị lớn nhất.

Thuật toán Ford-Fulkerson có hai bước chính. Việc đầu tiên là một tiến trình ghi nhãn để tìm kiếm cho một đường dẫn tăng lưu lượng tức là, một đường đi từ  $S_0$  đến  $S_{|U|-1}$  với ràng buộc  $l < t_s$  dọc theo tất cả các cạnh về phía trước và  $l > 0$  dọc theo tất cả các cạnh ngược lại. Nếu bước này tìm thấy một đường dẫn lưu lượng tăng, bước thứ hai thay đổi lưu lượng phù hợp. Nếu không, không có đường dẫn tăng lưu lượng tồn tại, sau đó chúng nhận được lưu lượng tối đa. Thuật toán bắt đầu với bất kỳ lưu lượng khả thi nào ( $l = 0$ ), một nút ở một trong ba trạng thái: không gắn nhãn, gắn nhãn và đã duyệt, hoặc gắn nhãn và chưa duyệt. Khi thực hiện bước 1, tất cả các nút không gắn nhãn. Bước đầu tiên chỉ ra rằng nút nguồn được gắn nhãn và chưa duyệt. Các bước chi tiết như sau:

- *Bước 1*: Khởi tạo, gắn nhãn nguồn ( $S_0$ ;  $nhãn(S_0) = \infty$ );

- *Bước 2*: Chọn bất kỳ nút  $S_i$ , có gắn nhãn và chưa duyệt (Nếu không có các nút được gắn nhãn và chưa duyệt, thì lưu lượng hiện tại là lưu lượng tối đa). Đối với tất cả các nút  $S_j$ , nếu  $S_j$  không gắn nhãn thì:

Nếu  $S_{ij} \in V$  và  $ll_{S_{ij}} < ts_{S_{ij}}$  thì gắn nhãn  $(S_i; +; nhan(S_j))$  cho nút  $S_j$  theo công thức (3a) như sau:

$$nhan(S_j) = \min(nhan(S_i), ts_{S_{ij}} - ll_{S_{ij}}); \quad (3a)$$

Nếu  $S_{ji} \in V$  và  $ll_{S_{ji}} > 0$  thì gắn nhãn  $(S_i; -; nhan(S_j))$  cho nút  $S_j$  theo công thức (3b) như sau:

$$nhan(S_j) = \min(nhan(S_i), ll_{S_{ji}}); \quad (3b)$$

Sau đó, hãy để  $S_i$  được gắn nhãn và đã duyệt, trong khi đó để cho  $S_j$  được gắn nhãn và chưa duyệt. Nếu nút  $S_{|U|-1}$  được gắn nhãn thì đi đến bước 3, ngược lại trở lại bước 2.

- *Bước 3*: Hãy để  $S_x = S_{|U|-1}$ , sau đó thực hiện công việc tính lưu lượng theo công thức (4a) hoặc (4b) cho đến khi  $S_x = S_0$ .

Nếu nhãn của  $S_x$  là  $(S_y; +; nhan(S_x))$  thì đặt

$$ll_{S_{yx}} = ll_{S_{yx}} + nhan(S_{|U|-1}) \quad (4a)$$

Nếu nhãn của  $S_x$  là  $(S_y; -; nhan(S_x))$  thì đặt

$$ll_{S_{xy}} = ll_{S_{xy}} - nhan(S_{|U|-1}) \quad (4b)$$

Đặt  $S_x = S_y$

Sau đó đi đến bước 1.

Yêu cầu ràng buộc đối với bài toán luồng cực đại là tìm giá trị cực đại dựa trên (5), (6) và (7):

$$\text{- Ràng buộc trọng số (lưu lượng trên mỗi liên kết } S_{ij} \text{ không quá } ts_{S_{ij}}): 0 \leq ll_{S_{ij}}^p \leq ts_{S_{ij}} \quad (5)$$

$$\text{- Đối xứng lưu lượng (} \forall S_i \neq S_0, S_{|U|-1}, \text{ lưu lượng dòng vào bằng với lưu lượng dòng ra): } ll_{S_{ij}}^p = -ll_{S_{ji}}^p \quad (6)$$

$$\text{- Ràng buộc lưu lượng (} \forall S_i \neq S_0, S_{|U|-1}): \sum_{j \in U} ll_{S_{ij}} = 0 \quad (7)$$

## B. Giải pháp cải tiến thuật toán truyền thông nhóm

### 1. Trật tự toàn phần với đồng hồ lô gic Lamport

Dựa vào truyền thông nhóm, thuật toán Lamport được cải tiến lại theo cách giải quyết sau: Mỗi trạm  $S$  đều có trang bị công tơ với các giá trị nguyên gọi là  $H_s$ , đồng hồ lô gic tăng lên khi có sự kiện  $sk_j$  diễn ra trên một trạm bất kỳ. Một trạm  $i$  khi phát sinh sự kiện gửi thông điệp yêu cầu được cung cấp giá trị đồng hồ đến tất cả các trạm còn lại theo thủ tục:

$$Get\_Lamport(S_i, H_{S_i}, sk_j)$$

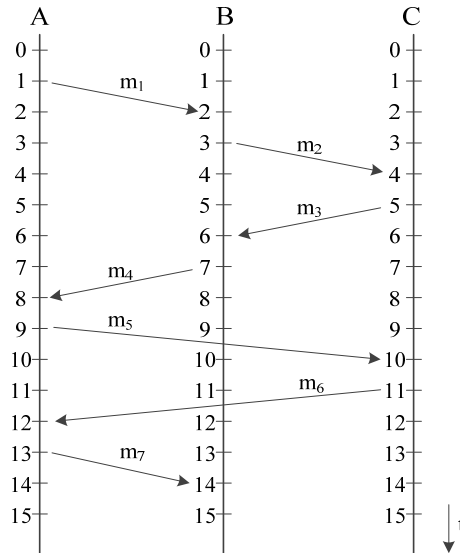
tất cả các trạm sau khi nhận được thông điệp trên kiểm tra, so sánh với giá trị hiện tại  $H_{S_k}$  của trạm mình ( $H_{S_i} \leq H_{S_k}$ ) và phản hồi thông điệp chấp nhận giá trị với thủ tục:

$$Accept\_Lamport(S_k, H_{S_k}, sk_j, true)$$

Sau khi tiếp nhận thông điệp phản hồi từ các trạm còn lại, nếu tham số thứ 4 nhận được đều có giá trị *true* thì tiến hành lấy  $\max(H_{S_k})$ , cập nhật lại giá trị đồng hồ theo  $\max(H_{S_k})$  và tăng giá trị lên 1 để gắn cho sự kiện. Sau khi gắn giá trị cho sự kiện và cập nhật lại đồng hồ theo giá trị mới, một thông điệp gửi đồng hồ đến các trạm theo thủ tục để xác nhận giá trị đồng hồ đã được gắn cho sự kiện  $sk_i$ :

$$Lamport(S_i, H_{S_i}, sk_j)$$

Tất cả các trạm nhận so sánh và cập nhật lại giá trị đồng hồ theo giá trị đồng hồ hiện tại của trạm  $i$  cung cấp, đồng thời ghi nhận nhãn thời gian sự kiện  $sk_j$  cho trạm phát yêu cầu. Kết quả của thuật toán là tất cả các trạm cùng thực hiện cập nhật giá trị đồng hồ và tuân thủ theo trật tự toàn phần như Hình 6. Kết quả này là điều kiện hết sức quan trọng đối với các giải pháp tiếp theo để phát triển ứng dụng phân tán.



**Hình 6.** Trật tự tổng quát các thông điệp theo thuật toán Lamport

Sau khi đã gắn nhãn cho thông điệp, bộ cung cấp tài nguyên truyền thông sẽ tính toán luồng cực đại trong hệ thống mạng và tiến hành chuyển thông điệp đến tập đích.

## 2. Thuật toán tìm luồng cực đại

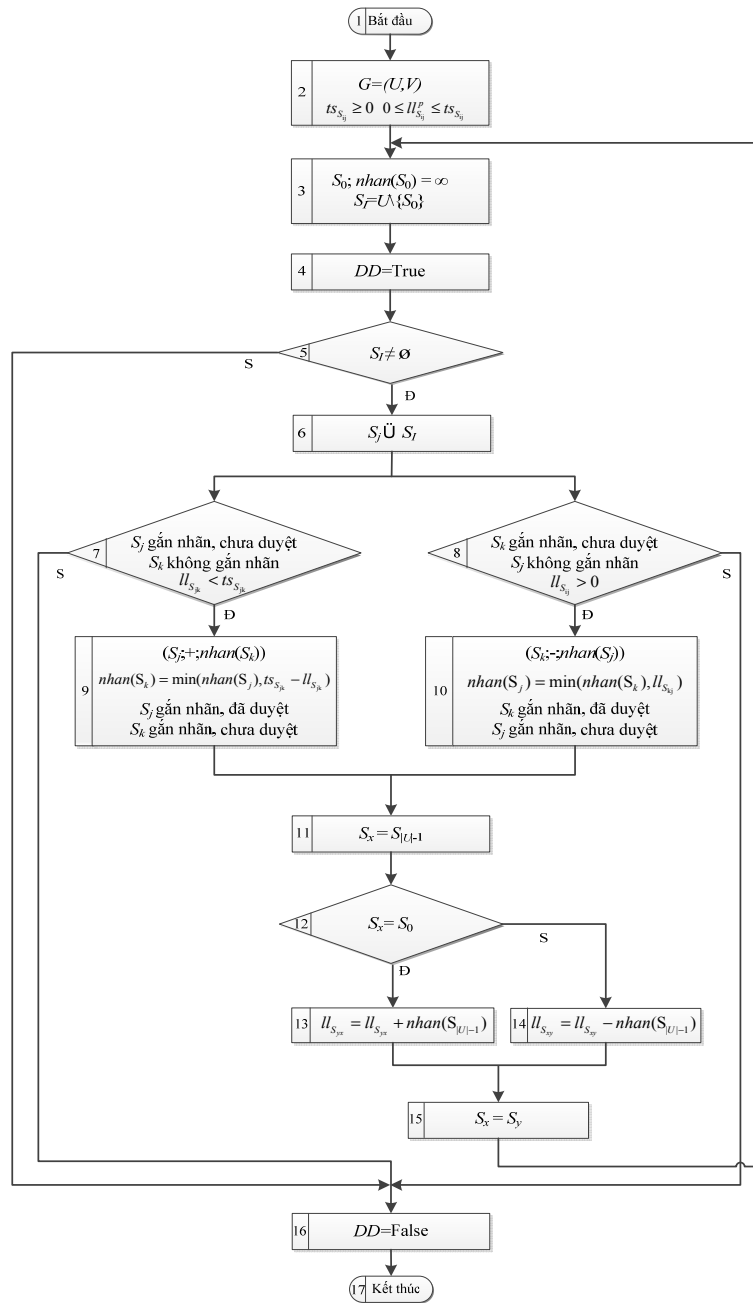
Cải tiến thuật toán Ford Fulkerson tập trung vào giải pháp truyền thông nhóm trong thuật toán theo Hình 7, thể hiện xử lý này trong bước 2. Giải pháp này xử lý tất cả các nút được gắn nhãn và chưa duyệt với tất cả  $k$ -nút kết nối, cải tiến bước 2 thể hiện như sau:

Đối với mỗi cạnh  $S_{jk} \in V$ , một trong hai tình huống có thể có thể xảy ra:

- . Tại khối 7,  $S_j$  đã được gắn nhãn và chưa duyệt,  $S_k$  không gắn nhãn và  $ll_{S_{jk}} < ts_{S_{jk}}$ . Nếu tình huống này xuất hiện thì khối 9 thực hiện gắn nhãn ( $S_j; +; nhan(S_k)$ ) cho nút  $S_k$ , trong đó  $nhan(S_k) = \min(nhan(S_j), ts_{S_{jk}} - ll_{S_{jk}})$  và đặt  $S_j$  đã được gắn nhãn và đã duyệt và  $S_k$  được gắn nhãn và chưa duyệt;
- . Tại khối 8,  $S_k$  được gắn nhãn và chưa duyệt,  $S_j$  không gắn nhãn và  $ll_{S_{jk}} > 0$ . Nếu tình huống này xuất hiện thì khối 10 thực hiện gắn nhãn ( $S_k; -; nhan(S_j)$ ) cho nút  $S_j$ , trong đó  $nhan(S_j) = \min(nhan(S_k), ll_{S_{jk}})$  và đặt  $S_k$  đã được gắn nhãn và đã duyệt và  $S_j$  đã được gắn nhãn và chưa duyệt;

Nếu nút  $S_{|U|-1}$  được gắn nhãn thì đi đến bước 3, ngược lại trở lại bước 2.

Trong giải pháp xử lý đồ thị tuyến tính cho các ứng dụng phân tán sử dụng phương thức truyền multicast thì đường dẫn từ nguồn đến đích có thể tồn tại nhiều đường đi khác nhau và phụ thuộc vào  $k$ -nút kết nối. Đối với bài toán  $k$ -nút kết nối là 2 theo Hình 4 thì từ nguồn ta có 2 đường dẫn đến đích theo phân chia tỷ lệ lưu lượng, do đó thuật toán Ford Fulkerson sẽ thực hiện theo các ràng buộc và tham số khác nhau trong nghiên cứu của chúng tôi.



Hình 7. Sơ đồ thuật toán cải tiến thuật toán Ford Fulkerson

### III. KẾT QUẢ THỰC NGHIỆM

Để đánh giá thực nghiệm, chúng tôi dựa vào công cụ tạo mô phỏng mạng BRITE để tạo tập dữ liệu ban đầu. Số nút mạng tăng dần từ 100 đến 500, số cạnh tạo tương ứng với số nút, ở đây chúng tôi chọn  $k$ -nút kết nối là 2 và các tham số khác đồng dạng. Các tham số cho thể hiện trong Hình 8.

```

BriteConfig
BeginModel
    Name = 1           #Router Waxman = 1, AS Waxman = 3
    N = 500           #Số nút trong đồ thị
    HS = 1000         #Kích thước trong mặt phẳng chính
    LS = 100          #Kích thước trong mặt phẳng bên trong
    NodePlacement = 1 #Random = 1, Heavy Tailed = 2
    GrowthType = 1    #Incremental = 1, All = 2
    m = 2             #k-nút kết nối
    alpha = 0.15      #Waxman Parameter
    beta = 0.2         #Waxman Parameter
    BWDist = 3        #Constant = 1, Uniform =2, HeavyTailed = 3, Exponential =4
    BWMin = 10.0
    BWMax = 1024.0
EndModel
    
```

Hình 8. Tham số đầu vào để tạo mô phỏng

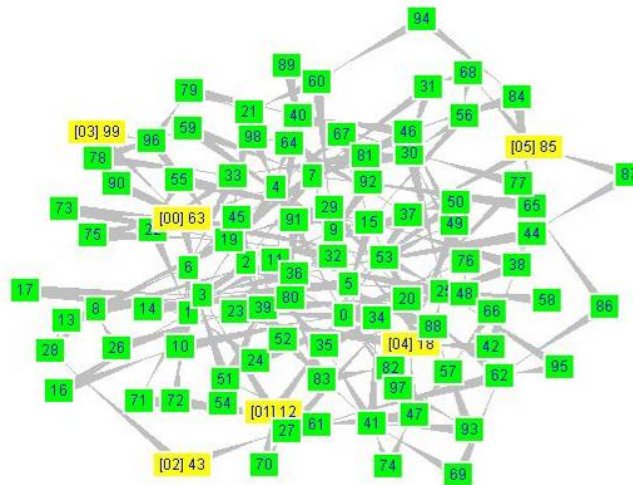
Trong Hình 8,  $N$  là số nút trong mạng được chọn từ 100 đến 500 nút, băng thông cung cấp từ 10M đến 1024M và được lựa chọn trải đều trên các cạnh. Đầu ra của chương trình là tập tin brite với số nút và cạnh được tạo tương ứng theo Hình 9.

```

Topology: ( 100 Nodes, 200 Edges )
Model ( 5 ): 1 -1 0 0
AS Level: Model ( 3 ): 5 1000 100 1 1 2 0.15 0.2 1 10 1024
Router Level: Model ( 1 ): 20 1000 100 1 1 2 0.15 0.2 1 10 1024
Nodes: ( 100 )
0 536 233 11 11 -1 RT_NODE
1 346 720 11 11 -1 RT_NODE
2 534 624 7 7 -1 RT_NODE
...
98 893 229 2 2 -1 RT_NODE
99 661 284 2 2 -1 RT_NODE

Edges: (200):
0 2 1 211.09239683134018 0.7041284435225525 173.3547070775228 -1 -1 RT_NODE
1 2 0 391.0051150560565 1.3042526742152283 46.7503610865091 -1 -1 RT_NODE
2 3 2 211.482859825566 0.7054308878763255 20.969351692142233 -1 -1 RT_NODE
...
198 1 72 290.31879029783795 0.9683992460472036 13.127006715125411 -1 -1 RT_NODE
199 1 71 298.10736320996836 0.9943791288103997 103.7707730144837 -1 -1 RT_NODE
    
```

Hình 9. Dữ liệu của tập tin brite



Hình 10. Kết quả mô phỏng với 100 nút kết nối và 6 nút lô gic hoạt động bên trong hệ

Tập tin brite được thi trên công cụ Distributed System Simulator (DSSim), chương trình khởi tạo đồ thị chứa các nút, tính toán lưu lượng truyền dựa vào số nút lô gic để truyền thông điệp được chọn ngẫu nhiên trong tổng số nút mạng được tạo. Hình 10 thể hiện kết quả mô phỏng với 100 nút kết nối và 6 nút lô gic hoạt động bên trong hệ.

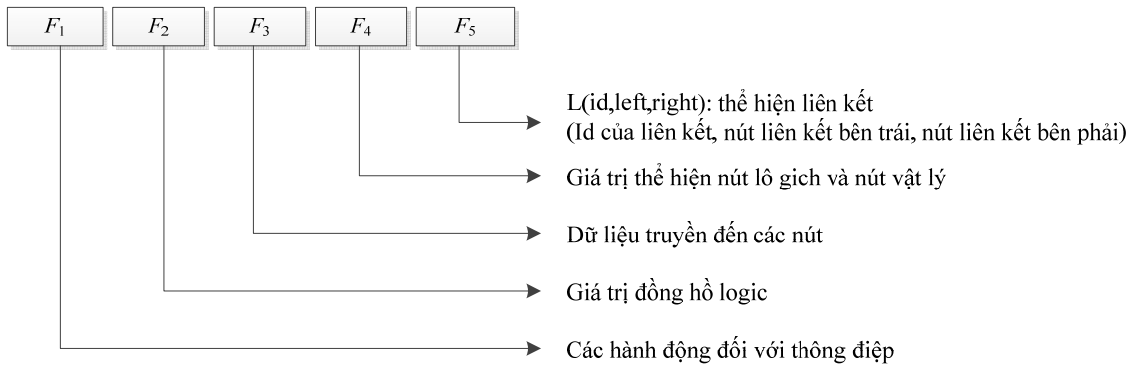
Đối với giá trị đồng hồ lô gic, các sự kiện phát sinh trên mỗi trạm đều được cập nhật và tăng dần giá trị đồng hồ. Các hành động diễn ra trong sự kiện thể hiện qua Bảng 1 như sau:

Bảng 1. Các hành động diễn ra trong sự kiện của hệ

TT	Tên sự kiện	Thuyết minh
1	MSG NODE SEND	Nút gửi thông điệp
2	MSG NODE RECV	Nút nhận thông điệp
3	MSG ROUTE SEND	Định tuyến gửi thông điệp
4	MSG ROUTE RECV	Định tuyến nhận thông điệp
5	ROUTE_BEGIN	Định tuyến bắt đầu
6	ROUTE_END	Định tuyến kết thúc
7	NODE_BEGIN	Nút bắt đầu
8	NODE_END	Nút kết thúc
9	MONITORING_LLINK_CREATION	Giám sát tạo liên kết
10	MONITORING_LLINK_REMOVAL	Giám sát xóa liên kết
11	MONITORING_PROP_CHANGED	Giám sát thay đổi

Dựa vào các hành động trong Bảng 1, hệ phân tán có thể điều khiển và giám sát toàn bộ hoạt động thông qua thông điệp. Cấu trúc một thông điệp được mô tả ở Hình 11.





Hình 11. Cấu trúc thông điệp

Giá trị đồng hồ là 0 thường là trạng thái khởi tạo ban đầu và 1 là bắt đầu các hoạt động trong hệ. Tương ứng với mỗi sự kiện được tạo, một giá trị đồng hồ được gán cho sự kiện đó. Hệ hoạt động song song, do đó giá trị gửi và nhận các sự kiện có thể có sự cách biệt. Quá trình hoạt động diễn ra trên một nút thể hiện qua Hình 12 như sau:

```
Running simulation...
Current time = 0
[su kien = NODE_CREATION, nhan thoi gian = 0, data = null,node logic = [?] node physic = 63(-1), Link = null]
[su kien = MONITORING_PROP_CHANGED, nhan thoi gian = 1, data = {knownHosts=[[00]], myAddress=[00]},node logic = [00] node physic = null, Link = null]
nut: [01]
nextPNode = 27(-1), destPNode = 12(-1)
[su kien = MSG_NODE_SEND, nhan thoi gian = 1, data = [04]->[03], "[04]", node logic = [04] node physic = 18(-1), Link = null]
[su kien = MSG_ROUTE_RECV, nhan thoi gian = 209, data = [02]->[01], "[02]", node logic = null node physic = 27(-1), Link = L(82,43(-1),27(-1))]
[su kien = MONITORING_PROP_CHANGED, nhan thoi gian = 1, data = {knownHosts=[[01], [00]], myAddress=[01]}, node logic = [01] node physic = null, Link = null]
[su kien = MONITORING_LLINK_CREATION, nhan thoi gian = 1, data = null, node logic = [01] node physic = null, Link = null]
```

Hình 12. Các sự kiện hoạt động trên nút 1

Dựa vào tô pô đã tạo, thuật toán Ford Fulkerson tiến hành găn nhãn và xây dựng tập lát cắt cực tiểu từ nguồn 0 đến các đích có thể trong tô pô. Chương trình sẽ tính toán luồng cực đại từ nguồn đến đích thông qua định tuyến giữa các nút. Kết quả tính giá trị luồng cực đại từ nút nguồn 0 đến các nút đích trong tập lát cắt cực tiểu thể hiện ở Bảng 2:

Bảng 2. Giá trị luồng cực đại từ nút nguồn 0 đến các nút đích trong tập lát cắt cực tiểu

1	2	3	4	5	6	7	9	10	11	12	15	18	20	25	27	30	39	47	48	51	54	61	71	76	82
27.44	27.44	27.44	13.57	10.30	23.29	27.44	24.05	13.13	24.25	27.44	14.14	27.44	27.44	10.30	10.26	10.30	22.56	12.05	10.30	10.15	27.12	13.13	27.44	10.30	17.14

Thuật toán Ford Fulkerson xử lý song song để tính giá trị luồng cực đại, tuy nhiên định tuyến của thông điệp chưa phải là đường đi ngắn nhất. Chương trình thể hiện giá trị tính toán luồng cực đại từ nguồn 0 đến đích 71 qua Hình 13.

```
Max flow from 0 to 71
0->82 17.137780840129547/17.137780840129547
0->76 10.300250523302278/10.300250523302278
1->71 27.438031363431826/103.7707730144837
2->1 0.13807224556338937/173.3547070775228
3->1 6.940867236199129/22.399929844294924
6->1 10.162178277738889/27.268578834024947
7->6 10.162178277738889/10.162178277738889
7->2 0.13807224556338937/13.405771039819596
12->3 6.940867236199129/40.173380307117824
18->1 10.196913603930419/10.196913603930419
27->12 6.940867236199129/41.17808718558616
30->7 10.300250523302278/10.33726885201113
47->27 6.940867236199129/10.262745710257473
76->30 10.300250523302278/29.347007500104446
82->47 6.940867236199129/12.054926436918477
82->18 10.196913603930419/11.9729766215762
Min cut: 0
Max flow value = 27.438031363431826
```

Hình 13. Tính toán luồng cực đại từ nguồn 0 đến đích 71

### III. KẾT LUẬN

Giải pháp cung cấp tài nguyên phân tán là một trong những giải pháp nhằm tối ưu truyền thông điệp. Trong nghiên cứu này, chúng tôi đưa ra hai giải pháp để cải tiến xử lý truyền thông điệp đó là thuật toán Lamport nhằm cho phép ghi nhận các sự kiện diễn ra trong hệ phân tán và thuật toán Ford Fulkerson tìm luồng cực đại truyền thông điệp từ nguồn đến đích. Hướng chính của giải pháp là dựa vào truyền thông nhóm để cải tiến hai thuật toán trên đảm bảo đạt được trật tự toàn phần khi truyền thông điệp trong hệ và các thông điệp được chia thành nhiều hướng khác nhau để đến đích một cách nhanh chóng, chính xác dựa vào tính toán lưu lượng cực đại trong quá trình truyền. Tuy nhiên, trong giải pháp cung cấp tài nguyên phân tán, trong quá trình truyền các thông điệp có thể bị trùng lặp khi phân chia tại các nút. Như vậy, thông điệp nhận tại tập đích có thể bị dư thừa dẫn đến lãng phí tài nguyên truyền thông. Một trong những hướng nghiên cứu để giải quyết cho vấn đề trên là lĩnh vực mã mạng (Network Coding), chúng tôi sẽ trình bày trong nghiên cứu tiếp theo.

### IV. TÀI LIỆU THAM KHẢO

- [1] Lê Văn Sơn, Hệ tin học phân tán, Nhà xuất bản Đại học Quốc gia Tp. Hồ Chí Minh, 2002.
- [2] Michel Raynal, Distributed Algorithms for Message-Passing Systems, Springer Publishing Company, Incorporated, 2013.
- [3] Valmir C. Barbosa, An introduction to distributed algorithms, MIT Press, 1996.
- [4] Gerard Tel, Introduction to distributed algorithms, Cambridge University Press, 1994.
- [5] Kayhan Erciyes, Distributed Graph Algorithms for Computer Networks, Springer Publishing Company, Incorporated, 2013.
- [6] Weijia Jia and Wanlei Zhou, Distributed Network Systems: From Concepts to Implementations (Network Theory and Applications), Springer-Verlag New York, Inc., 2006.
- [7] Shuo-yen Robert Li, Raymond W. Yeung, and Ning Cai, "Linear network coding", IEEE Transactions on Information Theory, 49(2), pp. 371-381, 2003.
- [8] Zongpeng Li and Baochun Li, "Network Coding in Undirected Networks", in Conference on Information Sciences and Systems, 2004.
- [9] Vijay K. Garg, Roger Wattenhofer, and Kishore Kothapalli, "Distributed Computing and Networking", in International Conference of Distributed Computing and Networking, 2009.
- [10] Ajay D. Kshemkalyani and Mukesh Singhal, Distributed Computing: Principles, Algorithms and Systems, Cambridge University Press, 2008.
- [11] Nancy A. Lynch, Distributed Algorithms, Morgan Kaufmann, 1996.
- [12] Andrew S. Tanenbaum and Maarten van Steen, Distributed Systems: Principles and Paradigms (2nd Edition), Prentice-Hall, Inc., 2006.
- [13] Harold S. Stone, "Multiprocessor Scheduling with the Aid of Network Flow Algorithms", Software Engineering, IEEE Transactions on, SE-3(1), pp. 85-93, 1977.

## A METHOD OF RESOURCE ALLOCATION FOR DISTRIBUTED COMMUNICATION

Dang Hung Vi, Le Van Son

**ABSTRACT** - Nowadays, distributed systems have been researched and developed for virtualization systems. The technical solutions establish coherence status that based on cooperating between the servers of system, cooperation in distributed systems primarily use message passing mechanism. The disadvantages of cooperation are the received messages not in the order because communication latency influences on the system processing. Besides, the messages may be lost, fragmentation and duplication in communication environment. Therefore, in order to ensure the order and optimal transmission of messages, allocator communication resources should be built. In the paper, we propose to improve the distributed communication resource allocation algorithms that are able to effectively apply for distributed systems.