

Giải pháp tối ưu truyền thông multicast với mã mạng

The Optimal Solution for Multicast with Network Coding

Đặng Hùng Vĩ, Lê Văn Sơn

Abstract: Currently, in complex and large systems as distributed systems, resource allocation in communications has to be ensured high throughput, timely and exactly. One of the current approaches to the multicast transmission has achieved some certain results compared with unicast ones. However, the multicast transmission has fundamental restrictions such as data overlap in the destination set, which affect performance of the communications system. In order to solve this problem, we propose multicast transmission combined with network coding in which based on studies of the algorithms for building network topology so that the throughput in the destination set reaches the optimum value. The results of the study are compared with operating schemes to determine feasibility of solution proposed.

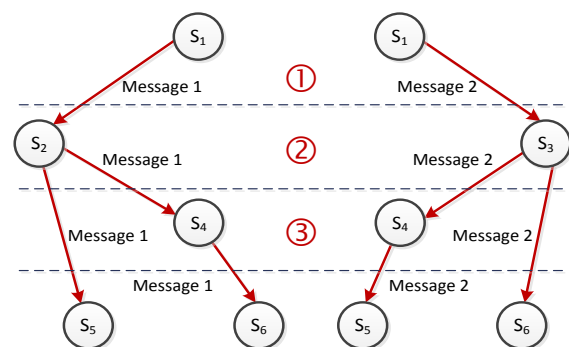
Keywords: Network Coding, multicast, Distributed System

I. GIỚI THIỆU

Truyền thông điểm đến điểm (point-to-point) là phương thức truyền unicast trong hệ thống mạng [1]. Phương thức truyền này có hai hạn chế cơ bản là dễ gây tắc nghẽn và bị mất kết nối giữa các điểm với nhau [2]. Các nghiên cứu phương thức truyền thông thay thế truyền unicast trong các ứng dụng phân tán bằng truyền multicast đang được tiến hành và mang lại nhiều kết quả khả quan [3-9]. Tuy nhiên, nhược điểm của phương thức truyền multicast là các gói tin truyền đến tập đích có thể bị trùng lặp dẫn đến lãng phí tài nguyên truyền thông [10, 11]. Chính vì thế, giải pháp khắc phục tình trạng nêu trên là một trong những xu hướng nghiên cứu tất yếu. Một trong các hướng nghiên cứu để giải quyết giải pháp này là mã mạng.

Mã mạng ban đầu được đề xuất cho kết nối multicast đơn, những nghiên cứu sau này đã đưa ra những thuận lợi mà mã mạng cung cấp tài nguyên truyền thông khả quan hơn [12].

Mục đích nghiên cứu kỹ thuật mã mạng của chúng tôi là thiết lập các kết nối multicast và tránh trùng lặp thông tin tại tập đích. Tuy nhiên, hiệu quả của mã mạng phụ thuộc vào topology mạng hiện hành.



Hình 1. Cây multicast

Việc nghiên cứu thiết kế và xây dựng mạng hoàn chỉnh bao gồm: tính toán ma trận lưu lượng, xây dựng topology và quản lý [13]. Trong đó, xây dựng topology với chi phí tối ưu là một trong những khía cạnh quan trọng của thiết kế mạng. Đối với bài toán tối ưu xây dựng topology, các nghiên cứu trước đây đã đưa ra giải pháp giải quyết bài toán dựa trên mô hình cây [14-17].

Hình 1 mô tả cây multicast với nút nguồn là S_1 , các nút trung gian S_2, S_3, S_4 và tập đích: S_5 và S_6 ; tập đích nhận được gói tin ở nút trung gian gần nhất là cấp 3 theo sự phân cấp của cây multicast [18]. Các vấn đề xử lý trên cây là khả năng khôi phục lỗi khi xảy ra sự cố và giới hạn băng thông giữa các nút với nhau. Khi xảy ra sự cố, các nút ở nhánh không thể dự đoán được nguyên nhân lỗi xảy ra, các nút nhánh này càng gần

đích thì hiệu quả truyền cao hơn. Giới hạn băng thông là vấn đề cần phải xem xét trong các ứng dụng lớn, xảy ra tắc nghẽn tại nhánh nào đó trong cây sẽ làm giảm hiệu năng trong quá trình truyền.

Giải pháp tối ưu truyền thông multicast với mã mạng là một trong những nghiên cứu của chúng tôi sao cho thông tin tại tập đích tránh trùng lặp và đạt thông lượng tối ưu. Giải quyết giải pháp này trên cây multicast là xây dựng lại tô pô kết hợp kỹ thuật mã mạng để thông lượng đạt cực đại tại tập đích.

Để đánh giá giải pháp đề xuất, chúng tôi thực nghiệm dựa trên công cụ tạo tô pô BRITE [19] để tạo ra tập tin chứa các nút và các liên kết ban đầu. Tô pô này thực hiện trên mô phỏng hệ phân tán Distributed System Simulator (DSSim) [20]. DSSim kết hợp với mã mạng qua công cụ Network Coding Utilities (ncutils) [21], các công cụ nêu trên được xây dựng trên ngôn ngữ Java. Sau khi kết hợp các công cụ nêu trên, thuật toán được xây dựng để minh chứng tính tối ưu truyền thông qua ba phương thức truyền: unicast, multicast và mã mạng. Qua mô phỏng, có thể đánh giá hiệu năng và chứng minh tính hiệu quả của kỹ thuật mã mạng trong chiến lược cung cấp tài nguyên truyền thông cho các hệ thống lớn, phức tạp.

II. CÁC YÊU CẦU VỀ THÔNG LƯỢNG VÀ XÂY DỰNG TÔ PÔ MẠNG

Hệ thống mạng (theo Hình 1) có thể mô tả dưới dạng đồ thị $G=(U,V)$, 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 kế trong hệ thống. Ký hiệu nút mạng thứ j là S_j , $j=0..U$. Gọi I là tập các phiên multicast truyền thông điệp trong hệ. Đối với mỗi phiên $i \in I$ ta có $S_0 \in U$ và $S_{|U|-1} \in U - \{S_0\}$ tương ứng là nguồn và đích. Đối với hệ thống đang xét, ta có tập đích D thì $S_{|U|-1} \subset D$.

Thông lượng cực đại của một mạng U có chứa phiên truyền đơn i ký hiệu là $thl(U)$. Chúng ta sẽ so sánh $thl(U)$ với các tham số khác đã được xác định như: *gói*, *độ bền* và *kết nối* để mô tả các kết nối hoặc trọng số của một mạng truyền thông. Nghiên cứu tập trung so sánh các thông số tương ứng cho truyền unicast, multicast và mã mạng.

Gói đề cập đến các tính toán số lượng gói tin trên thời gian theo cặp cạnh rời rạc các cây con của G , số lượng gói của một mạng truyền thông U được ký hiệu là $g(U)$ và bằng với thông lượng cực đại mà không mã.

Độ bền là một loại kết nối vùng của mạng được ký hiệu là $db(U)$, đó là tỷ lệ cực tiểu của công thức (1):

$$db(U) = \frac{|lk_{tp}|}{(tp - 1)} \quad (1)$$

trong đó tp là số của các thành phần truyền thông nhóm được tách lập, lk_{tp} là tập các liên kết liên thành phần và vùng được yêu cầu phải có ít nhất một nút nguồn hoặc đích trong mỗi thành phần.

Kết nối được ký hiệu là $kn(U)$ đề cập đến các cạnh kết nối giữa một cặp nút trong truyền thông nhóm ký hiệu là $NM\{S_0, S_j\} \subseteq U$. Nghiên cứu tập trung vào việc định tuyến phân chia và định tuyến tích hợp. Đối với định tuyến tích hợp, tất cả các trọng số liên kết và tỷ lệ lưu lượng có giá trị nguyên. Đối với định tuyến phân chia, trọng số liên kết có thể được phân chia theo cả hai hướng và lưu lượng có thể được phân chia và sáp nhập ở quy mô tùy ý.

Trong một mạng với một phiên unicast $U = \{G, ts, NM\}$, số lượng gói $g(U)$ trở thành số đường dẫn cạnh rời rạc $S_0 \rightarrow S_{|U|-1}$. Thông lượng $thl(U)$ là tỷ lệ lưu lượng thông tin cực đại có thể đạt được trong việc truyền tải $S_0 \rightarrow S_{|U|-1}$. Độ bền $db(U)$ bây giờ được cực tiểu trong tất cả các lát cắt đơn giản tách S_0 và $S_{|U|-1}$. Kết nối trở thành cạnh kết nối giữa S_0 và $S_{|U|-1}$, tức là giá trị cực tiểu trọng số cạnh một trong những thành phần cần loại bỏ khỏi mạng, để tách S_0 và $S_{|U|-1}$.

Đối với truyền unicast với tỷ lệ $tl_{S_0, S_{|U|-1}}$ từ nút nguồn S_0 đến nút đích $S_{|U|-1}$, lượng lưu lượng unicast $du(\cdot)$ vào một nút phải bằng lưu lượng unicast này ra khỏi nút này, trừ khi nút này là nguồn hoặc đích. Đối với truyền unicast trong một mạng U , các tham số cân bằng thể hiện qua công thức (2):

$$g(U) = thl(U) = db(U) = kn(U) \quad (2)$$

Cây Steiner cơ bản truyền multicast với tập nút $S_t = \{S_{t,0}, S_{t,1}, \dots, S_{t,|U|-1}\}$ là một sự kết hợp đặc biệt của

truyền unicast $|U|-1$. Sự khác biệt giữa cây Steiner cơ bản multicast với nút tập nút S_t và truyền unicasts $|U|-1$ từ nút $S_{t,0}$ đến mỗi nút trong $\{S_{t,1}, \dots, S_{t,|U|-1}\}$ đó là: trong cách truyền trước đây tài nguyên sử dụng của mỗi liên kết (i,j) là giá trị cực đại một trong những $du_{S_{i,j}}^{(S_{t,0}, S_{t,1})}, \dots, du_{S_{i,j}}^{(S_{t,0}, S_{t,|U|-1})}$; trong khi ở cách truyền sau này, tài nguyên sử dụng của mỗi liên kết (i,j) là tổng của $du_{S_{i,j}}^{(S_{t,0}, S_{t,1})}, \dots, du_{S_{i,j}}^{(S_{t,0}, S_{t,|U|-1})}$. Sự khác biệt này là tính hiệu quả của cây Steiner cơ bản multicast trong việc sử dụng nguồn tài nguyên truyền thông. Bảo toàn lưu lượng truyền cây multicast cơ bản có thể được mô tả qua công thức (3) như sau:

$$\sum_{\{j:(i,j) \in V\}} du_{S_{i,j}}^{(S_{t,0}, S_{t,l})} - \sum_{\{j:(i,j) \in V\}} du_{S_{j,i}}^{(S_{t,0}, S_{t,l})} = \begin{cases} -tl_l & \text{nếu } i=S_{t,l}, \\ tl_l & \text{nếu } i=S_{t,0}, \\ 0 & \text{ngược lại} \end{cases}$$

$$\forall i \in U, l \in \{1, \dots, |U|-1\}, \quad (3a)$$

$$\sum_{\{j:(i,j) \in V\}} F\left(du_{S_{i,j}}^{(S_{t,0}, S_{t,l})}\right) \leq 1, \forall i \in U, \quad (3b)$$

$$\sum_{\{j:(i,j) \in V\}} F\left(du_{S_{j,i}}^{(S_{t,0}, S_{t,l})}\right) \leq 1, \forall i \in U, \quad (3c)$$

Hàm F nhận giá trị 1 khi giá trị các biến bên trong lớn hơn 0, ngược lại F nhận giá trị 0. Khi mã mạng được áp dụng, vấn đề của việc thiết lập truyền multicast với tập nút S_t và tỷ lệ lưu lượng tl_{S_t} tương đương với hai vấn đề cơ bản tách lập: Một là xác định các đồ thị con trong mạng hiện tại, và thứ hai là xác định mã để sử dụng qua đồ thị con [22].

Trong nghiên cứu [22] về giải pháp điều khiển tỷ lệ nguồn với mã mạng, toán tử logic XOR được sử dụng tính toán số học trên trường hữu hạn. Vì vậy, trọng số multicast đạt được với mã mạng tuyến tính trên trường hữu hạn dựa vào xây dựng thuật toán tính toán hiệu quả để tìm tập các trọng số đạt được của hệ số mã mạng tuyến tính.

Như vậy, multicast là một sự kết hợp đặc biệt của $|U|-1$ unicast. Tuy nhiên, khác với trường hợp của cây Steiner cơ bản multicast, có thể có nhiều đường định tuyến một thông điệp đồng thời cho mỗi unicast từ nguồn $S_{t,0}$ đến đích (không có ràng buộc (3b) và (3c)). Định tuyến truyền multicast mã mạng cơ bản cũng được ứng dụng để cân bằng tải mạng. Với truyền multicast, mục đích của việc áp dụng định tuyến mã mạng cơ bản thay vì định tuyến cây Steiner cơ bản là để đạt được định tuyến tối ưu làm giảm đáng kể mức tiêu thụ băng thông của mỗi kết nối [18]. Vì vậy, giải pháp truyền này giảm tiêu thụ tài nguyên truyền thông trong một mạng. Giống như trường hợp ở truyền multicast Cây Steiner cơ bản, tiêu thụ băng thông của mỗi liên kết (i,j) là giá trị cực đại một trong số

$$\sum_{\{j:(i,j) \in V\}} dt_{S_{i,j}}^{(S_{t,0}, S_{t,1})}, \dots, \sum_{\{j:(i,j) \in V\}} dt_{S_{i,j}}^{(S_{t,0}, S_{t,|U|-1})} \text{ và } \sum_{\{j:(i,j) \in V\}} dt_{S_{j,i}}^{(S_{t,0}, S_{t,|U|-1})},$$

thay cho tổng của chúng. Đối với truyền multicast trong một mạng U , các tham số thể hiện qua công thức (4):

$$1/2kn(Z) \leq g(Z) \leq thl(Z) \leq db(Z) \leq kn(Z) \quad (4)$$

Trong nghiên cứu ở [23] nêu ra vấn đề chi phí cực tiểu truyền multicast mã mạng cơ bản hiệu quả hơn truyền multicast cây Steiner cơ bản.

Ký hiệu $cp_{S_{i,j}}$ là chi phí cho mỗi đơn vị lưu lượng trên liên kết (i,j) . Chi phí cực tiểu kết nối multicast với tập nút S_t có thể được công thức hóa ở (5) và (6) sau:

$$\text{Tính giá trị cực tiểu: } \sum_{(i,j) \in V} cp_{S_{i,j}} \cdot b_{S_{i,j}}$$

Áp dụng cho:

$$b_{S_{i,j}} \geq dt_{i,j}^{(S_{t,0}, S_{t,l})}, \forall (i,j) \in U, l \in \{1, \dots, |U|-1\},$$

$$\sum_{\{j:(i,j) \in U\}} dt_{S_{i,j}}^{(S_{t,0}, S_{t,l})} - \sum_{\{j:(i,j) \in U\}} dt_{S_{j,i}}^{(S_{t,0}, S_{t,l})} = \begin{cases} -tl_l & \text{nếu } i = S_{t,l}, \\ tl_l & \text{nếu } i = S_{t,0}, \\ 0 & \text{ngược lại} \end{cases} \quad (5)$$

$$\forall i \in U, l \in \{1, \dots, |U|-1\}.$$

$$ts_{S_{i,j}} \geq dt_{S_{i,j}}^{(S_{t,0}, S_{t,l})}, \forall (i,j) \in V, l \in \{1, \dots, |U|-1\}. \quad (6)$$

Đây là bài toán lập trình tuyến tính với các thuật toán thời gian đa thức để có được giải pháp tối ưu.

Trong thuật toán xây dựng tô pô, chúng tôi quan tâm khoảng cách $kc_{i,j}$ cũng như $cp_{i,j}$ và xây dựng các kết nối multicast chi phí cực tiểu cho từng yêu cầu multicast.

Trong mạng G tổng lưu lượng unicast và lưu lượng multicast trên một liên kết (i,j) nhỏ hơn hoặc bằng $ts_{i,j}$. Ràng buộc này có thể được mô tả qua công thức (7) như sau:

$$\sum_{i_1=1}^u \sum_{i_2=1}^u du_{S_{i,j}}^{(S_{i_1}, S_{i_2})} + \sum_{t=1}^u \max_{l \in \{1, \dots, |S_t|-1\}} dt_{S_{i,j}}^{(S_{t,0}, S_{t,l})} \leq ts_{i,j} \quad (7)$$

$$\forall (i, j) \in V$$

Số hạng đầu tiên ở phía bên trái của (7) là tổng số của lưu lượng unicast trên liên kết (i,j) và số hạng thứ hai là tổng lưu lượng truyền multicast mã mạng cơ bản trên liên kết (i,j) . Đối với truyền multicast thứ t , tổng lưu lượng lưu lượng trên liên kết (i,j) là giá trị cực đại một trong $|U|-1$ dòng unicast, có nghĩa là, tính giá trị $\max_{l \in \{1, \dots, |U|-1\}} dt_{S_{i,j}}^{(S_{t,0}, S_{t,l})}$ thay cho tổng của $|U|-1$ lưu lượng unicast trên liên kết (i,j) .

Bài toán xây dựng tô pô được xây dựng như sau:

1. Số nút U và ma trận khoảng cách $(kc_{i,j})_{U \times U}$
2. Ma trận yêu cầu unicast $(yc_{i,j})_{U \times U}$
3. Tập nút $\{S_{t,0}, S_{t,1}, \dots, S_{t,|U|-1}\}$ và tỷ lệ lưu lượng tl_i của yêu cầu multicast thứ i ($i=1, 2, \dots, Y$)
4. Trọng số $ts_1, \dots, ts_K$, chi phí cố định $cp_1, \dots, cp_K$ và chi phí trên một đơn vị chiều dài $p_1, \dots, p_K$ các loại đường truyền khác nhau
5. k -nút kết nối
6. Tính giá trị tối đa liên kết sử dụng $e_{\max}$.

Tính giá trị cực tiểu

$$\sum_{i=1}^{u-1} \sum_{j=i+1}^u kc_{i,j} = \sum_{i=1}^{u-1} \sum_{j=i+1}^u \sum_{t=1}^K S_{i,j}^t (cp_t + kc_{i,j} \cdot p_t) \quad (8)$$

qua các biến

$$S_{i,j}^1, \dots, S_{i,j}^K \in \mathbb{N} : 1 \leq i \leq U-1, i+1 \leq j \leq U$$

$$du_{S_{i,j}}^{(S_0, S_{|U|-1})} \geq 0 : 1 \leq i, j, S_0, S_{|U|-1} \leq U (i \neq j, S_0 \neq S_{|U|-1})$$

$$dt_{S_{i,j}}^{(S_{t,0}, S_{t,l})} \geq 0 : t \in \{1, \dots, M\}, l \in \{1, \dots, |S_t|-1\},$$

$$1 \leq i, j \leq u (i \neq j)$$

Áp dụng cho

1. Ràng buộc bảo toàn lưu lượng:

$$\sum_{\substack{1 \leq j \leq u \\ j \neq i}} du_{S_{i,j}}^{(S_0, S_{|U|-1})} - \sum_{\substack{1 \leq j \leq u \\ j \neq i}} du_{S_{j,i}}^{(S_0, S_{|U|-1})} = \begin{cases} -tl_{S_0, S_{|U|-1}} & \text{nếu } i = S_0, \\ tl_{S_0, S_{|U|-1}} & \text{nếu } i = S_{|U|-1}, \\ 0 & \text{ngược lại} \end{cases} \quad (9)$$

$$\forall 1 \leq i, S_0, S_{|U|-1} \leq u$$

2. Ràng buộc bảo toàn lưu lượng multicast:

$$\sum_{\substack{1 \leq j \leq u \\ j \neq i}} dt_{S_{i,j}}^{(S_{t,0}, S_{t,l})} - \sum_{\substack{1 \leq j \leq u \\ j \neq i}} dt_{S_{j,i}}^{(S_{t,0}, S_{t,l})} = \begin{cases} -tl_i & \text{nếu } i = S_{t,l}, \\ tl_i & \text{nếu } i = S_{t,0}, \\ 0 & \text{ngược lại} \end{cases} \quad (10)$$

$$\forall t \in \{1, \dots, M\}, l \in \{1, \dots, |S_t|-1\}, 1 \leq i \leq U.$$

3. Ràng buộc sử dụng liên kết và yêu cầu độ trễ:

$$du_{S_{i,j}} = \sum_{i_1=1}^z \sum_{i_2=1}^z du_{S_{i,j}}^{(S_{i_1}, S_{i_2})} + \sum_{t=1}^Y \max_{\substack{l \in \{1, \dots, |S_t|-1\} \\ i_2 \neq i_1}} dt_{S_{i,j}}^{(S_{t,0}, S_{t,l})} \leq e_{\max} \cdot ts_{i,j} \quad (11)$$

$$\forall 1 \leq i, j \leq U (i \neq j).$$

So với vấn đề xây dựng tô pô truyền thống, có thêm một ràng buộc bảo toàn lưu lượng truyền mã mạng trên cơ sở multicast. Khi so sánh với bài toán bảo toàn, ràng buộc (3) có thêm số hạng mô tả đặc tính của mã mạng.

III. CÁC KỸ THUẬT XỬ LÝ DÒNG THÔNG TIN

Một trong những đặc điểm của hệ thống mạng truyền unicast là mạng có thể thay đổi động. Mô hình hóa mạng động bằng cách thêm/xóa các luật phối hợp giữa các nút; do đó, việc xóa một nút được mô hình hóa bằng cách xóa tất cả các luật phối hợp liên quan đến nút này. Đối với yêu cầu thêm/xóa các nút với các luật phối hợp được dễ dàng nhận biết được với giả định rằng tất cả các nút được thiết lập ngay từ đầu và chỉ có luật phối hợp được thay đổi.

Chúng tôi xác định một hoạt động thay đổi tô pô mạng truyền unicast diễn ra như sau:

- AddLink(i, j , luật, id): thêm liên kết với luật phối

hợp từ nút j đến nút i . id là giá trị định danh duy nhất cho một cặp nút.

- DeleteLink(i, j, id): xóa các luật phối hợp giữa các nút i, j và id .

Truyền unicast giữa hai điểm và vấn đề triển khai tô pô ban đầu với hai thuật toán thêm/xóa liên kết để tạo ra tô pô hoạt động truyền thông điệp giữa các nút mạng tính chất cơ bản vừa nêu.

Trong các hệ phân tán phức tạp, các nút mạng phối hợp trao đổi thông điệp qua lại nên truyền unicast có nhiều hạn chế. Trên cơ sở đó, triển khai truyền multicast cho hệ phân tán trở nên cần thiết và là điều kiện để phát triển ứng dụng lớn. Phần trình bày tiếp theo mô tả thuật toán thêm/xóa liên kết truyền multicast và tối ưu cục bộ. Sau khi hình thành tô pô con kết nối multicast sẽ kết hợp với mã mạng để tập đích nhận được các gói tin truyền với thông lượng tối ưu.

III.1. Yêu cầu thuật toán

Bổ đề 1: Bài toán xây dựng tô pô của mạng unicast có khả năng tự phục hồi là bài toán NP-khó.

Chứng minh: Bài toán xây dựng tô pô này là NP-khó ngay cả khi yêu cầu lưu lượng $tl_{S_i,j}(i, j \in U, i \neq j)$

là rất nhỏ, để trọng số nhỏ nhất ts là đủ cho mỗi liên kết được gán, bởi nó chứa các tham số đã biết của bài toán NP-khó [11].

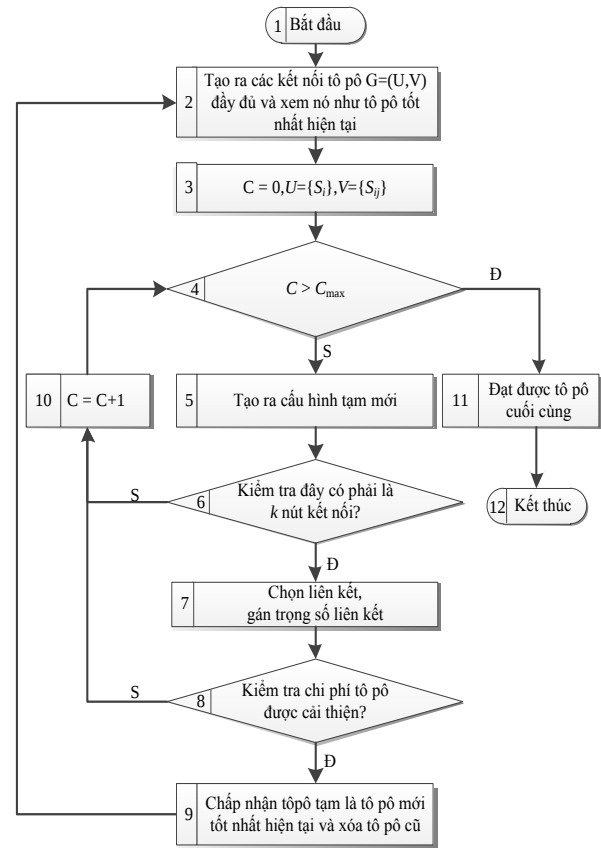
Định lý 1: Bài toán xây dựng tô pô của mã mạng multicast cơ sở có khả năng tự phục hồi là NP-khó.

Chứng minh: Bài toán xây dựng tô pô mới của truyền multicast mã mạng cơ sở có khả năng tự phục hồi bao gồm bài toán xây dựng hướng unicast như trường hợp nghiên cứu đặc biệt; do đó, đây cũng là bài toán NP-khó.

Hiện tại, không có thuật toán thời gian đa thức có sẵn đạt được giải pháp tối ưu của bài toán tối ưu NP-khó. Trong phần này, chúng tôi đưa ra hai thuật toán cho bài toán xây dựng tô pô này: thuật toán xóa liên kết và thuật toán thêm liên kết.

Thuật toán thêm liên kết thực thi đồ thị là tập đỉnh với các cặp đỉnh sắp thứ tự, đó là tô pô nguyên thủy

không có liên kết; đối với thuật toán xóa liên kết phải thực thi trên đồ thị có hướng, đó là tô pô kết nối đầy đủ.



Hình 2. Pha đầu trong thuật toán xóa liên kết

Hai thuật toán đề xuất đều bao gồm hai pha, bắt đầu tạo tô pô và tiến trình tối ưu hóa cục bộ: Trong pha đầu tiên của thuật toán xóa liên kết, bằng cách xóa từng liên kết từ tô pô kết nối đầy đủ cho đến khi không còn kết nối nào có thể bị xóa nữa, tô pô kết nối k nút với chi phí thấp bắt đầu được tạo ra. Trong pha đầu tiên của thuật toán thêm liên kết, bằng cách thêm các liên kết một bởi một tô pô nguyên thủy không có liên kết cho đến khi không còn liên kết nào nữa được thêm vào, tô pô k nút kết nối với chi phí thấp bắt đầu được tạo ra. Trong pha thứ hai của cả hai thuật toán, hoán đổi liên kết được lặp đi lặp lại cục bộ được thực hiện từng bước một để cải tiến tô pô ban đầu.

III.1.1. Thuật toán xóa liên kết trong multicast

Mục đích của pha đầu là để tạo ra một tô pô k nút kết nối có chi phí tương đối thấp. Sơ đồ của pha này

được thể hiện trong Hình 2.

Theo thuật toán trong Hình 2, ở khối 2 là các bước tạo ra các tô pô đầy đủ kết nối và xem nó như tô pô tốt nhất hiện tại. Sau đó, có được một cấu hình tạm bằng cách xóa một liên kết cụ thể trong cấu hình hiện tại được xử lý ở các khối 6,7,8. Nếu cấu hình tạm này đáp ứng một số điều kiện cụ thể, có nghĩa là, dựa trên cấu hình tạm này, một tô pô khả thi mới với chi phí thấp hơn có thể đạt được. Chấp nhận tô pô có tính khả thi mới này là tô pô tốt nhất hiện tại mới, loại bỏ tô pô cũ thể hiện trong khối 9. Tham số C trong khối 3 đó là tham số bộ đếm sử dụng để đếm số lần thất bại liên tục, trở lại bằng không. Nếu cấu hình tạm thời này không đáp ứng tất cả những điều kiện, loại bỏ nó và tăng C lên một được thực hiện ở khối 10. Nếu giá trị của C vượt quá một giá trị nhất định $C_{\max}$, kết thúc thuật toán, và các tô pô tốt nhất hiện tại là tô pô cuối cùng của pha này thể hiện ở khối 11. Ngược lại, đạt được một cấu hình tạm khác và kiểm tra nó. Bằng cách này, xóa liên kết được thực hiện lặp đi lặp lại đến khi không còn liên kết thích hợp có thể bị xóa nữa.

Xác định một ma trận $mt_{i,j}$ trên mỗi liên kết (i,j) bởi giá trị $mt_{i,j}=kc_{i,j}/(cp_{i,j}+cp_{j,i})$.

Tiến trình này bao gồm các bước cụ thể như sau:

1. Chọn số nút i từ nút 1 đến U và tạo ra các cấu hình kết nối đầy đủ. Sau đó, chọn định tuyến cho từng yêu cầu và xác định trọng số liên kết. Xem kết quả tô pô này là tô pô tốt nhất hiện tại.

2. Thiết lập bộ đếm tham số t về không và khởi tạo E , chứa các liên kết ứng viên để xóa, với tập chứa tất cả các liên kết trong tô pô tốt nhất hiện tại.

3. Kiểm tra xem giá trị của C là lớn hơn $C_{\max}=[U \cdot k/2]$. Nếu đúng, đi đến bước 7.

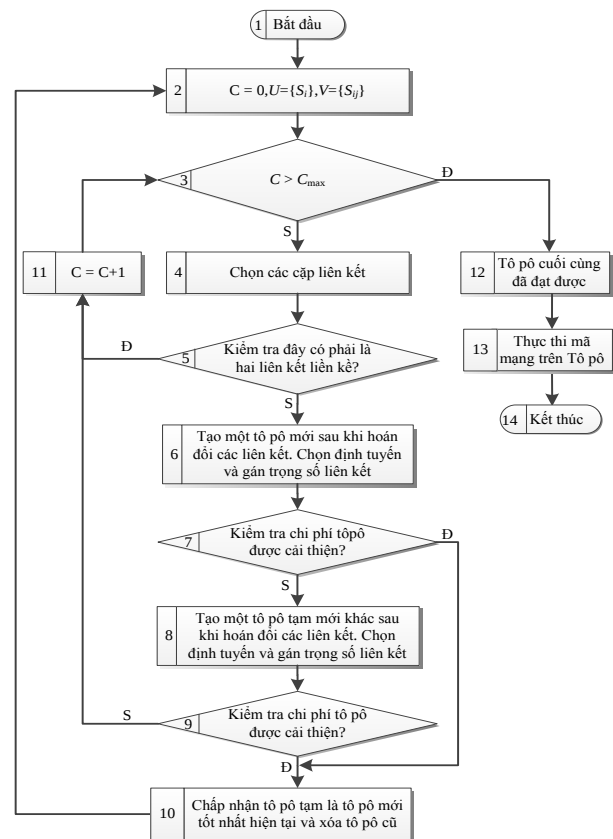
4. Từ E , chọn liên kết l có giá trị ma trận hiệu suất là lớn nhất. Chứa cấu hình tạm bằng cách loại bỏ liên kết l từ cấu hình hiện tại. Kiểm tra xem cấu hình tạm này là kết nối k nút. Nếu không, loại bỏ nó, tăng C lên một, và loại bỏ liên kết l từ liên kết ứng viên tập E . Sau đó, quay trở lại bước 3.

5. Gán lại định tuyến chỉ dành cho những yêu cầu

unicast và multicast có định tuyến qua liên kết l trong tô pô tốt nhất hiện tại.

6. Tính toán tổng chi phí của tất cả các liên kết. Nếu chi phí tô pô được cải thiện, chấp nhận tô pô tạm này là tô pô tốt nhất hiện tại. Sau đó, quay trở lại bước 2. Nếu không phải, loại bỏ các cấu hình tạm thời, tăng C lên một và loại bỏ liên kết l từ liên kết ứng viên tập E . Sau đó, quay trở lại bước 3.

7. Thoát ra và trả về tô pô tốt nhất hiện tại.



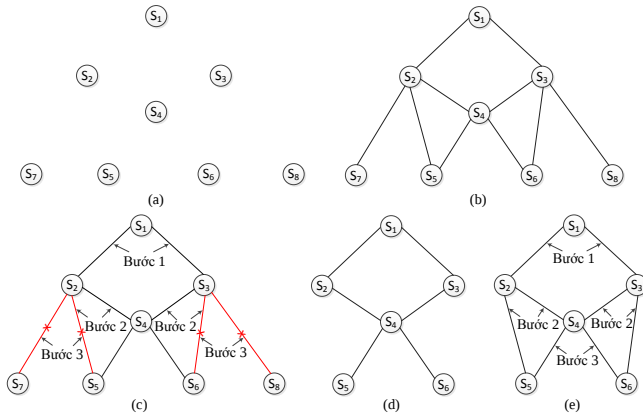
Hình 3. Thuật toán tối ưu cục bộ

Trong bước 3, lý do tại sao chúng ta để cho $C_{\max}=[U \cdot k/2]$ cho mỗi tô pô tốt nhất hiện tại để k nút kết nối có ít nhất $[U \cdot k/2]$ liên kết.

Thuật toán của quá trình tối ưu hóa cục bộ được thể hiện trong Hình 3.

Các bước thực hiện theo thuật toán thể hiện theo Hình 4 được mô tả. Hình 4(a) là tập các nút mạng ban đầu trong tập tô pô, Hình 4(b) thể hiện tô pô đầu đủ các kết nối. Khi thực hiện các bước theo thuật toán, các bước 1 và 2 trong Hình 4(c) bị loại bỏ bởi vì phạm

tính chất k nút kết nối trong multicast và kết quả đạt được của thuật toán xóa liên kết thể hiện trong Hình 4(d) đạt được tô pô tối ưu và đảm bảo k nút kết nối.



Hình 4. Các bước thực hiện thuật toán xóa/thêm liên kết

III.1.2. Thuật toán thêm liên kết trong multicast

Thuật toán này cũng bao gồm hai pha, pha bắt đầu tạo tô pô và tiến trình tối ưu hóa cục bộ, và giai đoạn thứ hai là giống như của các thuật toán xóa liên kết. Do đó, chúng tôi chỉ mô tả pha đầu.

Ý tưởng chính của pha này là tạo ra một cấu hình k nút kết nối có khả năng là một tô pô chi phí thấp và sau đó xây dựng một tô pô dựa trên cấu hình này.

Pha này bao gồm các bước cụ thể như sau:

1. Chọn số nút i từ 1 đến U .
2. Xác định các nút với cặp nút sắp thứ tự. Gọi nút này X . Nếu có một số nút ứng viên, chọn một trong các số thứ tự nhỏ nhất. Xác định các nút với nút có số thứ tự nhỏ nhất chưa kết nối với X . Gọi nút này Y . Nếu có một số nút ứng viên, chọn một trong số đó gần nhất đến X . Thêm liên kết $\{X, Y\}$.
3. Lặp lại bước 2 cho đến khi liên kết của mỗi nút là nhỏ hơn hoặc bằng k .
4. Kiểm tra xem cấu hình hiện tại là k nút kết nối. Nếu đúng, đi đến bước 6.
5. Kiểm tra xem các kết nối của cấu hình hiện tại có thể được tăng lên bằng cách chỉ thêm một liên kết. Nếu nó có thể được, bổ sung các liên kết ngắn nhất mà bổ sung này có thể tăng khả năng kết nối. Nếu không,

loại bỏ các cấu hình hiện tại và quay trở lại bước 1. Lặp lại các hoạt động nêu trên cho đến khi cấu hình hiện tại là k nút kết nối hoặc cho đến khi kết nối của cấu hình hiện tại không thể tăng liên kết nào nữa.

6. Sau đó, chọn định tuyến cho từng yêu cầu và xác định trọng số liên kết.

7. Thoát ra và trả về tô pô tốt nhất hiện tại.

Ở bước 5, nếu có một liên kết phải thêm vào liên kết khác để tăng kết nối; như vậy, các quy tắc là khá phức tạp để xác định liên kết có thích hợp được bổ sung vào để đảm bảo rằng các tô pô kết quả có một chi phí thấp.

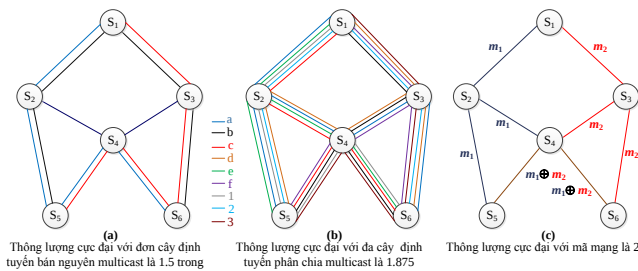
Theo Hình 4(e), các bước thực hiện thuật toán thêm liên kết đạt được tô pô cuối cùng. Tuy nhiên, khi so sánh với kết quả của thuật toán xóa liên kết ta nhận thấy rằng tập đích sẽ nhận được thông tin truyền có thể bị trùng lặp bởi có hai kết nối đến chúng. Nhưng kết quả trong Hình 4(e) vẫn đảm bảo được các bước khi thực hiện thuật toán và bảo toàn kết nối là k nút.

Chi phí để đạt được tô pô tốt hơn nếu kỹ thuật mã mạng được sử dụng hỗ trợ truyền multicast, chúng tôi nghiên cứu sự khác biệt chi phí tô pô giữa ba trường hợp sau: 1. Mỗi yêu cầu multicast được coi là đa yêu cầu unicast. 2. Các yêu cầu multicast được xem xét riêng rẽ từ yêu cầu unicast, và thuật toán cây Steiner được sử dụng để xây dựng định tuyến multicast. 3. Thuật toán truyền multicast mã mạng cơ sở với chi phí cực tiểu được sử dụng để xây dựng định tuyến multicast.

III.2. Tỷ lệ lưu lượng trong cây multicast với mã mạng

Sau khi thực thi tô pô ban đầu của mạng U để đạt được một tô pô tối ưu đề cập trong phần 3.1, chi phí khi thực hiện truyền trong tô pô với trường hợp có mã mạng và không có mã mạng thể hiện qua Hình 5. Theo kết quả ở (4), $1/2kn(Z) \leq g(Z)$ và $thl(Z) \leq kn(Z)$ có nghĩa là định tuyến phân chia được cho phép, do đó $1/2thl(Z) \leq g(Z)$ là ưu điểm mã mạng, như vậy ta có tỷ lệ lưu lượng $thl(Z)/g(Z) \leq 2$. Các liên kết được gán nhãn với cùng ký tự trong một cây Steiner. Thực thi thông

lượng trong [11] bằng cách truyền một lưu lượng dọc theo mỗi cây Steiner với tỷ lệ lưu lượng tương đương với trọng số cây. Kết quả thể hiện trong Hình 5(a) và 5(b), thông lượng đạt được tương ứng là 1,5 đối với định tuyến bán nguyên, 1,875 đối với định tuyến phân chia cả hai trường hợp đều không có mã mạng. Thông lượng tối đa với mã mạng theo điều kiện lý tưởng là 2 thể hiện trong Hình 5(c). Dựa vào kết quả trên, tỷ lệ giữa mã mạng và đa cây multicast là 1,067.



Hình 5. Ưu điểm của mã mạng trong cải tiến thông lượng multicast từ nguồn đến tập đích

III.3. Kiểm nghiệm thuật toán

Hiệu năng của một thuật toán xây dựng tô pô có thể được đánh giá thông qua so sánh với các thuật toán có sẵn [24] trên cùng một bài toán hoặc đo khoảng cách giữa các chi phí tô pô thu được bằng thuật toán này và thấp hơn ràng buộc về chi phí của các tô pô tối ưu. Nhưng ở đây, tác giả không tìm ra thuật toán có sẵn và các ràng buộc khác, chỉ biết đến trường hợp đơn giản như bài toán xây dựng tô pô hướng unicast và đa cây multicast theo Hình 5.

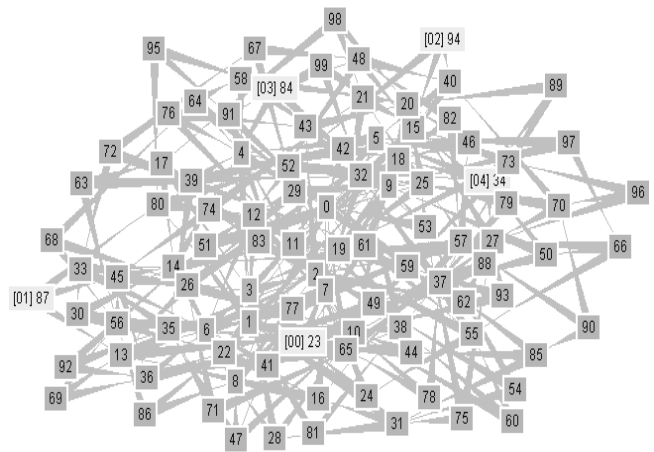
BriteConfig

```
BeginModel
  Name= 1          #Bộ định tuyến, Hệ thống tự quản Waxman
                  #Router Waxman=1, AS Waxman=3)
  N=100           #Số nút trong đồ thị
  HS=1000         #Kích thước của mặt phẳng chính
  LS=100          #Kích thước của mặt phẳng bên trong
  NodePlacement=1 #Các nút được thay thế trong mặt phẳng như thế nào
                  # (Random=1, Heavy Tailed=2)
  GrowthType=1    #Các nút tham gia vào tô pô bằng cách nào
                  # (Incremental=1, All=2)
  m=2             #Số kết nối trên nút mới
  alpha=0.15      #Thông số Waxman (0<alpha<=1)
  beta=0.2        #Thông số Waxman (0<beta<=1)
  BWDist=1        #Bảng thông được gán cho các liên kết
                  # (Constant=1, Uniform=2, HeavyTailed=3, Exponential=4)
  BWMin=10.0      # Giá trị bảng thông liên kết nhỏ nhất
  BWMax=1024.0    # Giá trị bảng thông liên kết lớn nhất
EndModel

BeginOutput
  BRITE=1         #Lưu dưới dạng BRITE (0=không, 1=có)
  OTTER=0         #Lưu dưới dạng Otter (0=không, 1=có)
EndOutput
```

Hình 6. Các tham số trong tập tin cấu hình BRITE

Để đánh giá hiệu năng của các thuật toán, chúng tôi xét mạng Z gồm 100 nút, k nút kết nối là 2 ($m=2$) theo Hình 7. Các tham số trong tập tin cấu hình của BRITE cho mô phỏng theo Hình 6. Trong tập tin BRITE, các tham số để tạo tô pô bao gồm bộ định tuyến được chọn là Router Waxman nằm trong mặt phẳng hiển thị là 1000 ($HS=1000$), tổng số nút mạng ($N=100$) được phân bố đều trên mặt phẳng trong là 100 ($LS=100$) và các nút thay thế được chọn ngẫu nhiên. Hai thành phần quan trọng để xuất tập tin cho chương trình phân tán thực thi đó là bảng thông được gán cho các liên kết được chọn đồng nhất ($BWDist=1$) với giá trị bằng thông nhỏ nhất là 10,0MB ($BWMin=10,0$).



Hình 7. Tô pô với 100 nút vật lý và 6 nút logic

Để đánh giá thuật toán trên, công cụ tạo tô pô BRITE tạo ra tập dữ liệu ban đầu với các tham số đầu vào mô tả trong Hình 6, kết quả đầu ra là tập 100 nút S_j , $j=0..99$, và 200 liên kết; đây chính là đồ thị $G=(100,200)$ có hướng ban đầu được tạo [19].

Để tính toán việc truyền thông điệp giữa các nút trong mạng, tô pô được thực hiện trên mô phỏng hệ phân tán [20] theo thuật toán thêm liên kết. Trên thực tế, tô pô mạng có thể thay đổi do các nút bị sự cố rời khỏi hệ thống, khi đó thuật toán được thực thi lại với số nút mới để tạo ra tô pô mới đảm bảo chi phí tối ưu và yêu cầu ràng buộc truyền multicast là k nút.

Hình 7 thể hiện chương trình đọc dữ liệu từ tập tin và xây dựng tô pô với liên kết các nút $u \in U$ đang hoạt động để truyền thông điệp được tô màu xám. Trong

tập liên kết này, chương trình lấy ra số nút logic được tô màu xám nhạt theo khai báo, chọn ngẫu nhiên số lượng nút này trong tô pô và hoạt động theo nguyên lý của token. Các nút hoạt động truyền token với sự hỗ trợ của mã mạng qua công cụ Network Coding Utilities [21], mỗi liên kết ra sẽ kết nối với 2 nút trong liên kết hoạt động để tìm đường đi ngắn nhất trong tô pô đến tập đích là số lượng nút logic ngoại trừ nút logic nguồn có giá trị 0.

Hiệu năng của kỹ thuật mã mạng trong xây dựng tô pô được trình bày bằng cách so sánh thiết kế mã mạng cơ sở với hướng unicast và đa cây multicast. Trong mô phỏng, tỷ lệ thiết lập yêu cầu unicast $tl_{s_i,j}$ ($i < j$) được chọn đồng nhất từ 10,0 đến 60,0 theo thông số đầu vào của tô pô cho các nút kết nối theo dạng token tương ứng với số nút mạng logic từ 6 đến 16 theo Bảng 1.

Bảng 1. Kết quả thực thi tô pô với 3 phương thức truyền

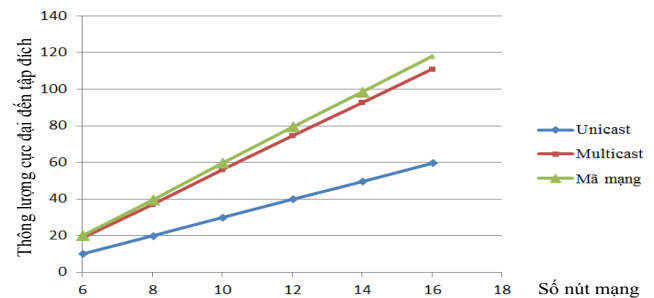
Lưu lượng	Mạng		Phương thức truyền		
	Nút logic	Liên kết logic	Unicast	Multicast	Mã mạng
10,0	6	8	10,0	18,75	20,00
20,0	8	12	20,0	37,50	40,00
30,0	10	16	30,0	56,25	60,00
40,0	12	20	40,0	75,00	80,00
50,0	14	24	50,0	93,75	100,00
60,0	16	28	60,0	112,50	120,00

Trong đó, nút nguồn phân chia tỷ lệ truyền, các cụm nút trung gian được chọn đồng nhất về chi phí đường truyền. Dựa vào tô pô này, chúng tôi sẽ nghiên cứu hiệu năng của hai thuật toán được đề xuất theo phân chia tỷ lệ lưu lượng. Kết quả thể hiện ở Bảng 1, tập nút nguồn sau khi phân chia tỷ lệ gói truyền trên tô pô với số nút và cạnh tương ứng trong mạng con của G , các phương thức truyền được đánh giá lần lượt là multicast đơn (Unicast), đa cây multicast (Multicast) và mã mạng.

Qua Bảng 1 và biểu đồ ở Hình 8 cho thấy, khi số lượng nút và cạnh tăng lên tương ứng với lưu lượng mạng, các tỷ lệ giữa ba phương thức truyền không có

sự thay đổi lớn. Tập nút đích nhận lưu lượng truyền với mã mạng đạt được cao hơn so với hai phương thức truyền còn lại, tỷ lệ giữa Multicast và Unicast xấp xỉ 1,875, tỷ lệ giữa Mã mạng và Multicast xấp xỉ 1,067. Vì vậy, với kết nối đa cây multicast nhất định, định tuyến chi phí cực tiểu mã mạng cơ sở có thông lượng tại tập đích cực đại.

Bước tiếp theo của nghiên cứu là dựa vào các số liệu mô phỏng trên cho thực nghiệm mạng diện rộng để đánh giá tính tối ưu của hai thuật toán đồng thời xây dựng giải pháp điều khiển đa nguồn, đa tỷ lệ với sự hỗ trợ của kỹ thuật mã mạng trong trường hợp số nút kết nối lớn hơn 2 và lưu lượng giữa các cạnh không đồng nhất. Qua đó, đánh giá thực nghiệm để rút ra kết quả trong nghiên cứu, nếu kết quả này tương đồng với mô phỏng sẽ đem lại nhiều thuận lợi cho các ứng dụng lớn.



Hình 8. Biểu đồ thể hiện thông lượng truyền đến tập đích qua 3 phương thức truyền

IV. KẾT LUẬN

Trong bài báo này, chúng tôi nghiên cứu đưa ra giải pháp xây dựng tô pô truyền multicast kết hợp với mã mạng. Chúng tôi xây dựng hai thuật toán tạo tô pô là thêm liên kết, xóa liên kết và thực thi chúng trong cùng ngữ cảnh. Các tô pô con được tạo từ tập kết nối ban đầu dựa trên các ràng buộc về thông lượng và yêu cầu xây dựng tô pô. Thuật toán triển khai truyền multicast kết hợp kỹ thuật mã mạng đảm bảo tập đích nhận thông tin không bị trùng lặp và đạt được thông lượng cực đại. Giải pháp đưa ra được thử nghiệm tính đúng đắn và có thể áp dụng cho các hệ thống lớn, phức tạp. Trong hướng nghiên cứu tiếp theo của chúng

tôi là tính toán độ phức tạp của hai thuật toán và độ hội tụ của bài toán khi tăng dần k nút kết nối.

TÀI LIỆU THAM KHẢO

- [1] R., NAKAO HAIDER A., POTTER, A., *Challenges in Resource Allocation in Network Virtualization*, in *In 20th ITC Specialist Seminar*, Hoi An, Vietnam, 2009.
- [2] LARRY L. PETERSON, BRUCE S. DAVIE, *"Computer Networks, Fifth Edition: A Systems Approach"*, Morgan Kaufmann Publishers, 2011.
- [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, 2013.
- [6] R. GALLAGER, *"A Minimum Delay Routing Algorithm Using Distributed Computation"*, IEEE Transactions on Communications, 25(1), 1977, pp.73-85.
- [7] WEIJIA JIA, WANLEI ZHOU, *"Distributed Network Systems: From Concepts to Implementations (Network Theory and Applications)"*, Springer-Verlag New York, 2006.
- [8] S. DEB, R. SRIKANT, *"Congestion control for fair resource allocation in networks with multicast flows"*, Networking, IEEE/ACM Transactions on, 12(2), 2004, pp.274-285.
- [9] SASWATI SARKAR, LEANDROS TASSIULAS, *"A framework for routing and congestion control for multicast information flows"*, IEEE Transactions on Information Theory, 48(10), 2002, pp.2690-2708.
- [10] SHUO-YEN ROBERT LI, RAYMOND W. YEUNG, NING CAI, *"Linear network coding"*, IEEE Transactions on Information Theory, 49(2), 2003, pp.371-381.
- [11] ZONGPENG LI, BAOCHUN LI, *Network Coding in Undirected Networks*, in *Conference on Information Sciences and Systems*, 2004.
- [12] Z. LI, B. LI, *Network coding: The case for multiple unicast sessions*, 2004.
- [13] AARON KERSHENBAUM, *"Telecommunications network design algorithms"*, McGraw-Hill, 1993.
- [14] YINING LI, JINGUO QUAN, XUELEI TAN, HUI LI, *"A Layered Multi-Tree IP Multicast Protocol With Network Coding"*, INFOCOMP 2012, The Second International Conference on Advanced Communications and Computation, 2012, pp.171 - 175.
- [15] CHRISTINA FRAGOULI, et al., *"Network coding: an instant primer"*, SIGCOMM Comput. Commun. Rev., 36(1), 2006, pp.63-68.
- [16] STEFAN BIRRER, et al., *FatNemo: Building a Resilient Multi-source Multicast Fat-Tree*, in *Web Content Caching and Distribution*, Chi-Hung Chi, Maarten Van Steen, and Craig Wills, Editors, Springer Berlin Heidelberg, 2004, pp.182-196.
- [17] SHEE ENG TAN, et al., *Minimizing Network Coding Nodes in Multicast Tree Construction via Genetic Algorithm*, in *Proceedings of the 2012 Fourth International Conference on Computational Intelligence, Communication Systems and Networks*, IEEE Computer Society, 2012, pp.399-404.
- [18] LEE SI-HYEON, CHUNG SAE-YOUNG, *"Capacity of a Class of Multicast Tree Networks"*, Information Theory, IEEE Transactions on, 59(6), 2013, pp.3848-3857.
- [19] BRITE, <http://www.cs.bu.edu/brite>.
- [20] Hermes, <http://hermes.soft112.com>.
- [21] Network Coding Utilities, <https://code.google.com/p/ncutils>.
- [22] ĐẶNG HÙNG VĨ, LÊ VĂN SƠN, *"Một giải pháp điều khiển tỷ lệ nguồn với mã mạng"*, Hội thảo quốc gia lần thứ XVI: Một số vấn đề chọn lọc của Công nghệ thông tin và truyền thông, Việt Nam, 2013, pp.193-198.
- [23] R. AHLWEDE, et al., *"Network information flow"*, IEEE Trans. Inf. Theor., 46(4), 2006, pp.1204-1216.
- [24] G. J. ZUURMOND M. BOTH, A. E. KRZESINSKI, *"An Implementation of the MENTOR Algorithm for Random Network Generation"*, in *Southern African Telecommunication Networks and Applications Conference*, 2002.

Nhận bài ngày: 01/06/2015

SƠ LƯỢC VỀ TÁC GIẢ

ĐẶNG HÙNG VĨ



Sinh ngày 07/03/1980 tại Đà Nẵng.

Tốt nghiệp ĐH tại Trường ĐH Bách khoa - ĐH Đà Nẵng năm 2003 chuyên ngành CNTT. Nhận bằng Thạc sỹ Khoa học Máy tính

năm 2008 tại ĐH Đà Nẵng. Hiện đang là nghiên cứu sinh tại Đại học Đà Nẵng.

Hiện công tác tại Trường ĐH Sư phạm, ĐH Đà Nẵng.

Lĩnh vực nghiên cứu: Mạng máy tính, mã mạng, hệ phân tán, điện toán đám mây.

Điện thoại: 090 5 294 998

Email: dhungvi@ued.udn.vn

LÊ VĂN SƠN



Sinh năm 1948 tại Điện Bàn, Quảng Nam.

Tốt nghiệp ĐH năm 1978, bảo vệ TS năm 1997 tại Trường ĐH Tổng hợp Donesk, Ucraina, được phong Phó Giáo sư năm 2004.

Hiện công tác tại Khoa Tin học, Trường ĐH Sư phạm, ĐH Đà Nẵng.

Lĩnh vực nghiên cứu: Hệ điều hành, mạng máy tính, hệ phân tán, tính toán đám mây.

Điện thoại: 090 5 827 499

E-mail: levansupham2004@yahoo.com