

ISSN 0866 708X

VIỆN HÀN LÂM KHOA HỌC VÀ CÔNG NGHỆ VIỆT NAM

VIETNAM ACADEMY OF SCIENCE AND TECHNOLOGY

TẠP CHÍ

# KHOA HỌC VÀ CÔNG NGHỆ

JOURNAL OF SCIENCE AND TECHNOLOGY



BỘ THÔNG TIN VÀ TRUYỀN THÔNG  
HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG

**Tập 52 - Số 6C**

Chuyên san năm thứ 5

**2014**

**CÁC CÔNG TRÌNH NGHIÊN CỨU VỀ ĐIỆN TỬ,  
VIỄN THÔNG VÀ CÔNG NGHỆ THÔNG TIN**



# TẠP CHÍ KHOA HỌC VÀ CÔNG NGHỆ

## Tổng biên tập

**Nguyễn Xuân Phúc**  
Viện Khoa học vật liệu, Hà Nội  
[phucnx@ims.vast.ac.vn](mailto:phucnx@ims.vast.ac.vn)

## Phó Tổng biên tập

**Nguyễn Dũng**, Viện Cơ học và Tin học ứng dụng, Tp Hồ Chí Minh; [nguyendung@vast-hcm.ac.vn](mailto:nguyendung@vast-hcm.ac.vn)  
**Thái Hoàng**, Viện Kỹ thuật nhiệt đới, Hà Nội; [thoang@vnd.vast.ac.vn](mailto:thoang@vnd.vast.ac.vn)  
**Nguyễn Ngọc San**, Học viện Công nghệ Bưu chính Viễn thông, Hà Nội; [khoaqtsdh@hn.vnn.vn](mailto:khoaqtsdh@hn.vnn.vn)

## Ủy viên Hội đồng biên tập

**Lê Huy Bá**, Trường Đại học Công nghiệp Tp Hồ Chí Minh, Tp Hồ Chí Minh; [lehuyba@hui.edu.vn](mailto:lehuyba@hui.edu.vn)  
**Lê Thanh Bình**, Viện Công nghệ sinh học, Hà Nội; [ltbinh24@yahoo.com](mailto:ltbinh24@yahoo.com)  
**Phạm Quang Bắc**, Tổng Công ty Dầu khí, Hà Nội  
**Nguyễn Hoài Châu**, Viện Công nghệ Môi trường, Hà Nội; [nhchau.cet@gmail.com](mailto:nhchau.cet@gmail.com)  
**Nguyễn Hữu Dũng**, Trường Đại học Bách khoa Hà Nội, Hà Nội; [dung-sdh@mail.hut.edu.vn](mailto:dung-sdh@mail.hut.edu.vn)  
**Nguyễn Công Định**, Học viện Kỹ thuật Quân sự, Hà Nội; [dinhnc@mta.edu.vn](mailto:dinhnc@mta.edu.vn)  
**Nguyễn Văn Gia**, Viện Cơ học và Tin học ứng dụng, Tp Hồ Chí Minh; [nvg40@yahoo.com](mailto:nvg40@yahoo.com)  
**Nguyễn Công Hào**, Viện Công nghệ Hóa học, Tp Hồ Chí Minh; [nguyenconghao@yahoo.com](mailto:nguyenconghao@yahoo.com)  
**Phan Bùi Khôi**, Trường Đại học Bách khoa HN, Hà Nội; [phanbuikhoi@mail.hut.edu.vn](mailto:phanbuikhoi@mail.hut.edu.vn)  
**Phạm Hữu Lý**, Viện Hóa học, Hà Nội; [huuly@yahoo.com.vn](mailto:huuly@yahoo.com.vn)  
**Phạm Quốc Long**, Viện Hóa học Các hợp chất thiên nhiên, Hà Nội; [mar.biochem@fpt.vn](mailto:mar.biochem@fpt.vn)  
**Trần Đình Long**, Trường Đại học Bách khoa Hà Nội; [evnbk@hn.vnn.vn](mailto:evnbk@hn.vnn.vn)  
**Châu Văn Minh**, Viện Hóa sinh Biển, Hà Nội; [cvminh@vast.ac.vn](mailto:cvminh@vast.ac.vn)  
**Hoàng Minh**, Học viện Công nghệ Bưu chính Viễn thông, Hà Nội; [hoangminh@ptit.edu.vn](mailto:hoangminh@ptit.edu.vn)  
**Nguyễn Xuân Nguyên**, Viện Hóa học Các hợp chất thiên nhiên, Hà Nội; [nxnguyen\\_ctc@yahoo.com](mailto:nxnguyen_ctc@yahoo.com)  
**Lê Văn Nhung**, Trường Đại học Bách khoa Hà Nội, Hà Nội; [levannhuongtc@yahoo.com](mailto:levannhuongtc@yahoo.com)  
**Nguyễn Hữu Phương**, Trường ĐH Khoa học tự nhiên, Tp Hồ Chí Minh; [nhphuong@hcmuns.edu.vn](mailto:nhphuong@hcmuns.edu.vn)  
**Phạm Văn Thiêm**, Trường Đại học Bách khoa Hà Nội, Hà Nội; [thiempv-edc@mail.hut.edu.vn](mailto:thiempv-edc@mail.hut.edu.vn)  
**Hồ Sĩ Thoảng**, Tổng Công ty Dầu khí, Tp Hồ Chí Minh; [hosithoang@gmail.com](mailto:hosithoang@gmail.com)  
**Phan Minh Tân**, Sở Khoa học Công nghệ Tp HCM, Tp Hồ Chí Minh; [tanpm@tphcm.gov.vn](mailto:tanpm@tphcm.gov.vn)  
**Phan Đình Tuấn**, Trường Đại học Bách khoa Tp HCM, Tp Hồ Chí Minh; [pdtuan@hcmut.edu.vn](mailto:pdtuan@hcmut.edu.vn)

**Thư kí Toà soạn:** Lê Thị Thiên Hương; [tc\\_khen@yahoo.com](mailto:tc_khen@yahoo.com)

**Địa chỉ:** Tòa soạn và Trụ sở: Tầng 1, Nhà A16, 18 Hoàng Quốc Việt, Cầu Giấy, Hà Nội;

**Điện thoại/Fax:** (84)-4-9422825; **Email:** [v-jst@vjs.ac.vn](mailto:v-jst@vjs.ac.vn); **Website:** <http://vjs.ac.vn/index.php/jst>.

## **BAN BIÊN TẬP SỐ CHUYÊN SAN**

### **Trưởng ban**

Hoàng Minh, *Tạp chí Khoa học và Công nghệ*

### **Phó Trưởng ban**

Vũ Văn San, *Học viện Công nghệ Bưu chính Viễn thông*

Nguyễn Ngọc San, *Tạp chí Khoa học và Công nghệ*

Michael E. Kress, *Trường Đại học CSI, Đại học Thành phố NewYork, Hoa Kỳ*

Alexeev Evgeniy B, *Trường Đại học Bưu điện Matxcova, Liên bang Nga*

### **Các thành viên**

Vũ Tuấn Lâm, *Học viện Công nghệ Bưu chính Viễn thông*

Tân Hạnh, *Học viện Công nghệ Bưu chính Viễn thông*

Lê Hữu Lập, *Học viện Công nghệ Bưu chính Viễn thông*

Zhanyang Zhang, *Trường Đại học CSI, Đại học Thành phố NewYork, Hoa Kỳ*

Eun Park, *Trường Đại học CSI, Đại học Thành phố NewYork, Hoa Kỳ*

Ievlev Oleg P, *Trường Đại học Bưu điện Matxcova, Liên bang Nga*

Cao Tiến Huỳnh, *Viện Khoa học Kỹ thuật và Công nghệ, Bộ Quốc phòng*

Đào Khắc An, *Viện Hàn lâm Khoa học và Công nghệ Việt Nam*

Nguyễn Bình, *Học viện Công nghệ Bưu chính Viễn thông*

Từ Minh Phương, *Học viện Công nghệ Bưu chính Viễn thông*

Trần Đình Quế, *Học viện Công nghệ Bưu chính Viễn thông*

Bùi Trung Hiếu, *Học viện Công nghệ Bưu chính Viễn thông*

Hoàng Đăng Hải, *Trung tâm VNCERT Việt Nam, Bộ Thông tin và Truyền thông*

**Trần Công Hùng**, *Học viện Công nghệ Bưu chính Viễn thông*

Phạm Ngọc Anh, *Học viện Công nghệ Bưu chính Viễn thông*

Nguyễn Tiến Ban, *Học viện Công nghệ Bưu chính Viễn thông*

Đặng Hoài Bắc, *Học viện Công nghệ Bưu chính Viễn thông*

Lê Nhật Thăng, *Học viện Công nghệ Bưu chính Viễn thông (Thư ký)*

## MỘT CÁCH TIẾP CẬN HEURISTIC CHO BÀI TOÁN ĐỊNH TUYẾN ĐẢM BẢO BĂNG THÔNG VÀ ĐỘ TRỄ

Cao Thái Phương Thanh<sup>1,\*</sup>, Hà Hải Nam<sup>2</sup>, Trần Công Hùng<sup>3</sup>

<sup>1</sup>Đại học Sài Gòn, 273 An Dương Vương, Q. 5, Tp. Hồ Chí Minh

<sup>2</sup>Học viện Công nghệ Bưu chính Viễn thông, 122 Hoàng Quốc Việt, Cầu Giấy, Hà Nội

<sup>3</sup>Học viện Công nghệ Bưu chính Viễn thông Cơ sở Tp. Hồ Chí Minh, 11 Nguyễn Đình Chiểu, Quận 1, Tp. Hồ Chí Minh

\* Email: [ctpthanh@sgu.edu.vn](mailto:ctpthanh@sgu.edu.vn)

Đến Tòa soạn: 19/8/2014; Chấp nhận đăng: 29/12/2014

### TÓM TẮT

Bài báo này giới thiệu một thuật toán định tuyến đảm bảo hai điều kiện chất lượng dịch vụ: băng thông tối thiểu và độ trễ tối đa. Mục tiêu của thuật toán là đáp ứng càng nhiều yêu cầu định tuyến càng tốt. Nhằm lựa chọn đường đi linh hoạt theo yêu cầu, thuật toán tính giá trị trọng số liên kết dựa trên băng thông. Sau đó áp dụng một cách tiếp cận heuristic vào thuật toán Dijkstra để tìm đường đi thỏa yêu cầu độ trễ và có trọng số nhỏ. Kết quả thực nghiệm trên các sơ đồ mạng khác nhau với tham số và yêu cầu định tuyến khác nhau cho thấy thuật toán được đề xuất có tỉ lệ chấp nhận yêu cầu và thời gian tính toán trung bình tốt hơn các giải pháp đã công bố.

*Từ khóa:* định tuyến đảm bảo băng thông và độ trễ, kỹ thuật lưu lượng, thuật toán heuristic.

### 1. GIỚI THIỆU

Ứng dụng mạng, đặc biệt là các ứng dụng đa phương tiện như truyền thanh, truyền hình trực tuyến yêu cầu chất lượng dịch vụ (Quality of Service - QoS) ngày càng tốt hơn. Những yêu cầu này được định nghĩa thành các điều kiện chất lượng dịch vụ trong hợp đồng giữa nhà cung cấp mạng và khách hàng. Trong quá trình vận hành mạng, nhà cung cấp không chỉ cần đảm bảo điều kiện chất lượng dịch vụ mà còn muốn sử dụng tài nguyên mạng một cách hiệu quả nhất. Vì vậy, thuật toán định tuyến kỹ thuật lưu lượng với điều kiện đảm bảo chất lượng dịch vụ là một lĩnh vực quan trọng đã và đang được quan tâm nghiên cứu [1]. Mục tiêu chính của thuật toán là đáp ứng càng nhiều yêu cầu định tuyến càng tốt với điều kiện đường đi được thiết lập cho mỗi yêu cầu phải đảm bảo một hoặc một số điều kiện chất lượng dịch vụ.

Điều kiện chất lượng dịch vụ phổ biến nhất trong các công trình định tuyến đã công bố là điều kiện đảm bảo băng thông [2] của đường đi lớn hơn một giới hạn yêu cầu (băng thông tối thiểu). Một cách tổng quát, các thuật toán định tuyến tính giá trị trọng số liên kết, sau đó áp dụng thuật toán tìm đường có trọng số nhỏ nhất (ví dụ Dijkstra) để xác định đường đi cho mỗi yêu cầu định tuyến. Trong thuật toán tìm đường, điều kiện băng thông được đảm bảo bằng cách không

xét các liên kết có băng thông nhỏ hơn băng thông yêu cầu. Việc đảm bảo điều kiện băng thông đơn giản vì băng thông được xét độc lập trên từng liên kết. Một điều kiện chất lượng dịch vụ phổ biến khác là đảm bảo độ trễ của đường đi nhỏ hơn một giới hạn yêu cầu (độ trễ tối đa). Khác với băng thông, điều kiện độ trễ phải xét theo tổng độ trễ của tất cả liên kết trên đường đi. Vì vậy, các thuật toán định tuyến đảm bảo băng thông không thể áp dụng để đảm bảo yêu cầu độ trễ. Trong sự hiểu biết của tác giả, số lượng thuật toán định tuyến đảm bảo đồng thời hai yêu cầu chất lượng dịch vụ băng thông và độ trễ ít hơn hẳn số thuật toán đảm bảo một yêu cầu. Bài báo này giới thiệu một cách tiếp cận heuristic mới cho bài toán định tuyến đảm bảo băng thông và độ trễ với hai đóng góp: (1) tính trọng số liên kết một cách đơn giản nhưng hiệu quả dựa trên băng thông và (2) điều chỉnh thuật toán Dijkstra để tìm đường đi thỏa điều kiện độ trễ và có trọng số nhỏ.

Bài báo được trình bày gồm năm phần. Sau phần giới thiệu, phần 2 định nghĩa bài toán và trình bày các công trình liên quan. Phần 3 đề xuất thuật toán định tuyến đảm bảo băng thông và độ trễ với tên gọi enhanced-Heuristic Routing Algorithm with Bandwidth Delay Constraints (eHRABDC). Hiệu quả định tuyến của thuật toán eHRABDC được so sánh, đánh giá với các thuật toán khác trong phần 4 theo hai tiêu chí tỉ lệ chấp nhận yêu cầu và thời gian tính toán trung bình. Cuối cùng, phần 5 trình bày kết luận và hướng phát triển.

## 2. ĐỊNH NGHĨA BÀI TOÁN VÀ CÁC CÔNG TRÌNH LIÊN QUAN

### 2.1. Định nghĩa bài toán

Sơ đồ mạng được biểu diễn như một đồ thị  $G(N, L)$ .  $N$  là tập các nút mạng và  $L$  là tập các liên kết. Mỗi liên kết  $l \in L$  có ba thông số: dung lượng (capacity -  $c(l)$ ), băng thông còn lại (residual bandwidth -  $b(l)$ ), và độ trễ (delay -  $d(l)$ ). Tương tự như phần lớn công trình định tuyến đảm bảo băng thông và độ trễ đã công bố [3 – 5] bài báo này giới hạn độ trễ của liên kết là độ trễ lan truyền. Như vậy, dung lượng và độ trễ của liên kết có giá trị cố định, trong khi băng thông còn lại có giá trị thay đổi theo lượng băng thông đang được bảo đảm trên liên kết. Một yêu cầu định tuyến được kí hiệu  $r(i, e, \beta, \delta)$  sẽ yêu cầu thiết lập một đường đi  $p_{ie}$  từ nút vào  $i$  đến nút ra  $e$ . Đường đi  $p_{ie}$  cần thỏa hai điều kiện: mỗi liên kết trên đường đi có băng thông còn lại lớn hơn hoặc bằng  $\beta$  và tổng độ trễ của các liên kết nhỏ hơn hoặc bằng  $\delta$ . Hơn nữa, nếu  $p_{ie}$  được thiết lập, băng thông còn lại của các liên kết thuộc đường đi này sẽ giảm  $\beta$  đơn vị (đảm bảo băng thông).

Trong phạm vi nghiên cứu, yêu cầu định tuyến được giả thiết đến tuần tự và các cặp nút vào – ra trong sơ đồ mạng được biết trước. Ngoài ra, thuật toán định tuyến được thực hiện tập trung với giả thiết các thông số liên kết như dung lượng, băng thông còn lại, và độ trễ đã sẵn sàng cho quá trình xử lí. Mục tiêu của thuật toán định tuyến là đáp ứng tối đa số yêu cầu định tuyến. Vì vậy, tỉ lệ chấp nhận yêu cầu được sử dụng làm độ đo hiệu quả định tuyến.

$$\text{tỉ lệ chấp nhận} = \frac{\text{số yêu cầu được chấp nhận}}{\text{tổng số yêu cầu}} * 100\% \quad (1)$$

### 2.2. Công trình liên quan

Thuật toán định tuyến đảm bảo băng thông và độ trễ được giới thiệu đầu tiên là Least Delay Algorithm (LDA) [3]. Đối với mỗi yêu cầu, LDA đánh dấu bỏ qua các liên kết không thỏa điều kiện băng thông và áp dụng thuật toán Dijkstra với trọng số là độ trễ của các liên kết còn lại. Độ trễ (nhỏ nhất) của đường đi tìm được sẽ được so sánh với độ trễ cần đảm bảo  $\delta$  để quyết định có chấp nhận yêu cầu hay không. Tuy đơn giản nhưng LDA luôn chọn cùng một đường đi (có độ

trễ nhỏ nhất) cho một cặp nút vào – ra. Kết quả là một phần sơ đồ mạng thuộc các đường đi có độ trễ nhỏ bị sử dụng nhiều, dễ tạo ra hiện tượng tắc cổ chai và cuối cùng làm giảm tỉ lệ chấp nhận yêu cầu định tuyến.

Thuật toán Minimum Interference Routing Algorithm (MIRA) [6] giới thiệu ý tưởng lựa chọn đường đi sao cho hạn chế ảnh hưởng đến các yêu cầu định tuyến trong tương lai. Sự ảnh hưởng này được đánh giá dựa trên tính chất maxflow – mincut và được thể hiện qua giá trị trọng số liên kết. Cụ thể, liên kết thuộc tập mincut được cho là có nhiều ảnh hưởng vì nếu được sử dụng sẽ làm giảm maxflow của một cặp vào – ra. Vì vậy, liên kết thuộc tập mincut được gọi là tới hạn và có trọng số cao. Ý tưởng hạn chế ảnh hưởng được kế thừa và phát triển với nhiều phương pháp tính trọng số heuristic khác nhau trong nhiều công trình định tuyến đảm bảo băng thông đã được công bố [2].

Các thuật toán đảm bảo băng thông có thể mở rộng thêm yêu cầu đảm bảo độ trễ bằng cách áp dụng Dijkstra mở rộng [7]. Cụ thể, thuật toán Dijkstra mở rộng tìm đường đi có trọng số nhỏ nhất và thỏa thêm yêu cầu tổng một thông số nguyên dương của các liên kết trên đường tìm được nhỏ hơn hoặc bằng một giá trị cho trước. Cụ thể với yêu cầu thỏa độ trễ  $\delta$ , Dijkstra mở rộng duy trì ở mỗi nút mạng một danh sách gồm  $(\delta + 1)$  mục  $EN_n = \{en_n^0, \dots, en_n^\delta\}$ . Mỗi mục  $en_n^k = \{d_n^k, w_n^k, m_n^k\}$  ở nút  $n$  có thông tin về nút liền trước ( $m_n^k$ ), trọng số ( $w_n^k$ ), và độ trễ ( $d_n^k$ ) của quá trình tìm đường đi từ nút vào  $i$  đến nút  $n$ . Ngoài ra, giá trị độ trễ của mỗi mục được gán trước  $d_n^k = k$ ,  $k = [0, 1, \dots, \delta]$  và được sử dụng như là chỉ số danh sách các mục. Dựa trên danh sách  $(\delta + 1)$  mục này mà thuật toán Dijkstra mở rộng tìm được đường đi có trọng số nhỏ nhất thỏa điều kiện độ trễ nhỏ hơn hay bằng  $\delta$ . Thuật toán 1 trình bày chi tiết thuật toán Dijkstra mở rộng tìm đường đi có trọng số nhỏ nhất thỏa yêu cầu độ trễ. Lưu ý, vì sử dụng độ trễ làm chỉ số của danh sách  $EN_n$ , thuật toán Dijkstra mở rộng chỉ có thể áp dụng với giá trị độ trễ nguyên dương.

Áp dụng Dijkstra mở rộng, các tác giả của [4] đề xuất hai thuật toán đảm bảo băng thông và độ trễ: Maximum Delay Weighted Capacity Routing Algorithm (MDWCRA) và Modified Maximum Delay Weighted Capacity Routing Algorithm (M-MDWCRA). Hai thuật toán này định nghĩa giá trị DWC (delay-weighted capacity) tỉ lệ thuận với khả năng đáp ứng yêu cầu của các cặp nút vào – ra theo công thức sau:

$$DWC_{ie} = \sum_{LP'_{ie} \in LP_{ie}} \frac{B'_{ie}}{D'_{ie}} \quad (2)$$

trong đó,  $LP_{ie} = \{LP'_{ie}, \dots, LP'_{ie}, \dots, LP'_{ie}\}$  là tập các đường đi có độ trễ nhỏ nhất của cặp  $ie$  (được xác định bằng Dijkstra với trọng số là độ trễ). Hai thuật toán MDWCRA và M-MDWCRA khác nhau ở cách xác định đường đi trong tập  $LP_{ie}$ . Cụ thể, thuật toán MDWCRA áp dụng Dijkstra để tìm đường đi có độ trễ thấp nhất thứ  $LP'_{ie}$  sau khi bỏ tất cả liên kết của đường đi đã tìm được trước đó  $LP'_{ie}{}^{l-1}$ . Trong khi thuật toán M-MDWCRA chỉ bỏ liên kết có băng thông còn lại nhỏ nhất trong đường  $LP'_{ie}{}^{l-1}$ . Ngoài ra, trong công thức 2,  $B'_{ie}$  là băng thông còn lại và  $D'_{ie}$  là tổng độ trễ của đường  $LP'_{ie}$ . Do DWC tỉ lệ thuận với khả năng đáp ứng yêu cầu nên thuật toán lựa chọn đường đi sao cho giá trị DWC ít giảm. Đối với mỗi đường  $LP'_{ie}$ , liên kết có băng thông còn lại nhỏ nhất ( $b(l) = B'_{ie}$ ,  $l \in LP'_{ie}$ ) nếu được sử dụng sẽ làm giảm DWC nhiều nhất, vì vậy liên kết này được gọi là liên kết tới hạn và được tăng trọng số. Tập hợp các liên kết này được gọi là tập liên kết tới hạn theo độ trễ. Sau khi xử lý tập  $LP$  cho mọi cặp vào – ra (cập nhật trọng số liên kết trong mỗi bước tìm  $LP'$ ), MDWCRA và M-MDWCRA áp dụng Dijkstra mở rộng để xác định đường đi có trọng số nhỏ nhất thỏa yêu cầu độ trễ.

Gần đây, thuật toán đảm bảo băng thông và độ trễ Minimum Delay and Maximum Flow (MDMF) [8] kết hợp ý tưởng tập liên kết tới hạn theo độ trễ của M-MDWCRA [4] và tập liên

**Thuật toán 1** Dijkstra mở rộng thỏa điều kiện độ trễ [7]

<p><b>Input:</b>                  Mạng <math>G(N, L)</math>                  Yêu cầu định tuyến <math>r(i, e, \beta, \delta)</math></p> <p><b>Output:</b>                  Đường đi có trọng số nhỏ nhất thỏa yêu cầu độ trễ <math>\delta</math>                  hoặc NULL // không có đường đi</p> <pre> 1: function EXTENDED_DJKSTRA(<math>G, r</math>) 2:   INIT(<math>G</math>) 3:   <math>Q = \{en_n^k   n \in N, k \in [0, \delta]\}</math> // <math>en_n^k</math> là mục thứ <math>k</math> của nút <math>n</math> 4:   while <math>Q \neq \emptyset</math> do 5:     tìm <math>en_n^k \in Q : w_n^k = \min\{w\}_{en \in Q}</math> 6:     <math>Q = Q - en_n^k</math> 7:     if <math>w_n^k = \infty</math> then 8:       break 9:     end if 10:    for liên kết <math>l</math> từ nút <math>n</math> đến nút liền kề <math>o</math> do 11:      RELAX(<math>Q, en_n^k, l, o, \delta</math>) 12:    end for 13:  end while 14:  GETPATH(<math>G</math>) 15: end function  16: function INIT(<math>G</math>) 17:   for nút <math>n \in N</math> do 18:     for mục <math>en_n^k   k \in [0.. \delta]</math> do                 </pre>	<pre> 19:       <math>d_n^k = k</math> // độ trễ 20:       <math>w_n^k = \infty</math> // trọng số 21:       <math>m_n^k = NULL</math> // nút trước 22:     end for 23:   end for 24:   for mục <math>en_i^k   k \in [0.. \delta]</math> do // nút vào <math>i</math> 25:     <math>w_i^k = 0</math> 26:   end for 27: end function 28: function RELAX(<math>Q, en_n^k, l, o, \delta</math>) 29:   <math>new\_d = d_n^k + d(l)</math> 30:   if <math>en_o^{new\_d} \in Q</math> và <math>new\_d \leq \delta</math> then 31:     <math>new\_w = w_n^k + w(l)</math> 32:     if <math>new\_w &lt; w_o^{new\_d}</math> then 33:       <math>w_o^{new\_d} = new\_w</math> 34:       <math>m_o^{new\_d} = n</math> 35:     end if 36:   end if 37: end function 38: function GETPATH(<math>G</math>) 39:   // xét các mục của nút ra <math>e</math> 40:   tìm mục <math>en_e^k : w_e^k \neq \infty</math> và <math>w_e^k = \min_{h \in [0, \delta]} \{w_e^h\}</math> 41:   if tìm được <math>en_e^k</math> then 42:     trả về đường đi dựa trên nút trước <math>m_e^k</math> 43:   end if 44:   trả về NULL 45: end function                 </pre>
---	---

kết tới hạn theo maxflow - mincut của MIRA [6] để tính trọng số liên kết (công thức 3). Ngoài ra, MDMF áp dụng mô hình latency-rate server [9] để tính độ trễ theo băng thông. Từ đó, MDMF giới thiệu một phương pháp heuristic tăng lượng băng thông đảm bảo để làm giảm độ trễ cho đường đi nhằm cố gắng thỏa điều kiện độ trễ.

$$w(l) = \frac{1 + \mu_{flow}CM + v_{delay}CD}{b(l)} \quad (3)$$

$$CM = \sum_{(i,e) \in IE} v_l$$

$$v_l = \begin{cases} 1 & \text{nếu } l \text{ thuộc tập mincut của } (i, e) \\ 0 & \text{nếu } l \text{ không thuộc tập mincut của } (i, e) \end{cases}$$

$$CD = \sum_{(i,e) \in IE} w_l$$

$$w_l = \begin{cases} 1 & \text{nếu } l \text{ thuộc tập liên kết tới hạn theo độ trễ của } (i, e) \\ 0 & \text{nếu } l \text{ không thuộc tập liên kết tới hạn theo độ trễ của } (i, e) \end{cases}$$

$IE$  là tập các cặp nút vào - ra

$\mu_{flow}, v_{delay}$  là các tham số điều khiển

