

## ỨNG DỤNG THUẬT TOÁN TIẾN HÓA ĐA MỤC TIÊU TRONG THIẾT KẾ TỐI ƯU KIẾN TRÚC MẠNG VIỄN THÔNG

ThS. Hoàng Ngọc Thanh <sup>1</sup>, Dương Tuấn Anh <sup>2</sup>

<sup>1</sup> Khoa CNTT, Trường Đại học Bà Rịa – Vũng Tàu

<sup>2</sup> Trường Đại học Bách khoa Thành phố Hồ Chí Minh

### Tóm tắt

Bài viết này đề xuất cách tiếp cận sử dụng thuật toán tiến hóa đa mục tiêu (MOEA) để giải quyết bài toán thiết kế tối ưu kiến trúc mạng viễn thông (TND) với nhiều ràng buộc phức tạp, các mục tiêu của bài toán gồm các yếu tố chi phí và độ tin cậy. Mỗi cá thể trong quần thể là biểu diễn của một mô hình mạng (topology) có yếu tố chi phí được xác định nhờ thuật toán đơn hình trong bài toán quy hoạch tuyến tính (LP) và độ tin cậy được xác định nhờ thuật toán Monte Carlo. Các MOEA khác nhau như Nondominated Sorting Genetic Algorithm (NSGA), Strength Pareto Evolutionary Algorithm (SPEA), Fast Non-dominated Sorting Genetic Algorithm (NSGA-II), ... đã được hiện thực để so sánh và đánh giá kết quả.

### Abstract

This paper proposes to apply Multi-Object Evolutionary Algorithm (MOEA) to solve the problem for the optimal design of the telecommunication network architecture (TND) with more complicated constraints and the objectives of the problem including costs and reliability. Each individual in the population is represented by a model of the network (topology) having the costs, which is determined by simplex algorithm in linear planning problem (LP) and the reliability is determined by Monte Carlo algorithm. The different MOEAs such as Nondominated Sorting Genetic Algorithm (NSGA), Strength Pareto Evolutionary Algorithm (SPEA), Fast Non-dominated Sorting Genetic Algorithm (NSGA-II), ... have been implemented to compare and evaluate the results.

## 1. GIỚI THIỆU

Trong thiết kế mạng viễn thông, các nút trung tâm cho các tổng đài hoặc các trung tâm chuyên mạch, cần được kết nối với nhau theo một cách tối ưu nhất (theo nghĩa chi phí truyền tải phải là tối thiểu, trong khi độ tin cậy phải là tối đa) nhằm điều khiển các lưu lượng điểm - điểm mong đợi. Các ràng buộc khác nhau trên mô hình mạng, dung lượng nút và liên kết phải được tôn trọng. Đây là dạng bài toán tối ưu đa mục tiêu có tính phi tuyến cao, mà cho đến nay, việc tìm kiếm một phương pháp chính xác để giải quyết vẫn còn để ngỏ. Mấy năm gần đây, một số tác giả đã giải quyết bài toán nêu trên theo hướng dùng thuật giải di truyền (GA) để tối ưu một trong hai mục tiêu

đã nêu hoặc bỏ qua một số các ràng buộc của bài toán; một số tác giả khác giải quyết hạn chế ở một vài cấu trúc mạng đặc thù. Chẳng hạn như trong [6], K.T Ko, K.S. Tang et al. có đề cập đến vấn đề “Using Genetic Algorithms to Design Mesh Networks”; trong [7] các tác giả L. Berry, B. Murtagh, S. Sugden và G. McMahon có đề cập đến vấn đề “Application of a Genetic-based Algorithm for Optimal Design of Tree-structured Communications Networks”. Trong nước cũng đã có nhiều nơi xem xét ứng dụng GA như: Viện Công nghệ thông tin, Trường ĐH Khoa học tự nhiên, Trường ĐH Bách khoa Tp.HCM, Phân viện CNTT tại Tp.HCM, ... Tuy nhiên, việc ứng dụng MOEA để giải quyết một vấn đề, đặc biệt trong lĩnh vực viễn thông, rất ít được đề

cập đến. Trong tạp chí Bưu chính Viễn thông số 197 (12/2002) tác giả Lương Hồng Khanh cũng có bài viết về việc “Ứng dụng thuật toán tiến hóa trong việc tối ưu hóa các tham số chất lượng mạng” [3]. Ở đây, chúng tôi nghiên cứu tiếp cận bài toán thiết kế tối ưu kiến trúc mạng viễn thông theo hướng tối ưu đa mục tiêu sử dụng một số các MOEA như NSGA, NSGAI, SPEA,... trên cơ sở tôn trọng các ràng buộc và mục tiêu thực tế, không đơn giản hóa hoặc bỏ qua các ràng buộc, tối ưu đồng thời nhiều mục tiêu. Kết quả đạt được có thể vận dụng được cho các mạng viễn thông có cấu trúc không đặc thù.

## 2. BÀI TOÁN

Mạng được mô hình hóa dưới dạng một đồ thị với các nút mạng được thể hiện là các đỉnh và các liên kết là các cạnh trong đồ thị. Cạnh của đồ thị có các trọng số tương ứng với loại của liên kết. Các liên kết cho phép dòng thông tin đi theo hai chiều. Vì vậy đồ thị ở đây là đồ thị vô hướng và có trọng số. Xét đồ thị  $G(V,E)$  với tập nút  $V$  và tập cung  $E$  thuộc tập đồ thị vô hướng  $S$ . Ta biểu diễn  $G$  bằng nửa trên của một ma trận kề nút - nút  $B$  với các phần tử  $b_{ij}$  ( $b_{ij}$  biểu diễn loại của liên kết  $(i,j)$  có giá trị trong khoảng  $[0, t]$ ; 0 tương ứng với không có liên kết). Bài toán của chúng ta là tìm một đồ thị  $G^*$  có chi phí truyền tải lưu lượng tối thiểu, độ tin cậy tối đa; đồng thời đảm bảo các ràng buộc về độ trễ, dung lượng nút mạng, dung lượng liên kết, bậc của nút và giới hạn số nút trung gian.

Định nghĩa:

$F_{pq}$  là tổng băng thông yêu cầu trên các kết nối giữa các cặp nút nguồn - đích  $(p,q)$ ,  $F_{pq}$  có thể được biểu diễn bằng một phần tử trong ma trận lưu lượng. Băng thông này có thể được xem là tương đương với dung lượng. Và  $F_{avg,pq}$  là lưu lượng trung bình dự báo.

Với mỗi liên kết  $(i,j)$ , có  $t$  loại liên kết, tương ứng với độ tin cậy là  $rt,ij$  và chi phí cho từng đơn vị băng thông là  $ct,ij$ .

Băng thông riêng phần của một đường thứ  $r$  từ nút  $p$  đến nút  $q$  được biểu thị là  $F_r$ . Chi phí

cho từng đơn vị băng thông trên đường này là

Rõ ràng ta có:  $h_r^{\#} \geq 0$

Khi đó tổng băng thông của kết nối  $(p,q)$

$$\text{là: } F_{pq} = \sum_r h_r^{\#}$$

Gọi  $a_{ij,r}$  là phần tử  $(i,j)$  của ma trận kề cho cặp  $(p,q)$  trên đường thứ  $r$ ;  $r = 1$  hoặc 0, tương ứng với việc có hoặc không liên kết  $(i,j)$  trên đường thứ  $r$  cho cặp nguồn đích  $(p,q)$ , ta có:

$$C_r^{\#} = \sum_{(i,j)} a_{ij,r}^{\#} c_{t,ij}$$

Chi phí của kết nối  $(p,q)$  là:  $\sum_r C_r^{\#} h_r^{\#}$

Và tổng chi phí truyền tải lưu lượng là:

$$\sum_{p=1}^n \sum_{q>p} \sum_r C_r^{\#} h_r^{\#}$$

Khi đó, tổng băng thông trên liên kết  $(i,j)$  sẽ

$$\text{là: } f_j = \sum_{p=1}^n \sum_{q>p} \sum_r a_{ij,r}^{\#} h_r^{\#}$$

Nếu  $f_j$  là dung lượng cực đại cho phép trên liên

kết  $(i,j)$ , ta có:  $0 \leq f_j \leq f_j^{\max}$

Nếu  $H_{\max}$  là cận trên của số liên kết trong

một chuỗi các liên kết, ta có:  $\sum_{(i,j)} a_{ij,r}^{\#} \leq H^{\max}$

Gọi  $u_i$  là lưu lượng tổng tại nút  $i$  với  $u_{i\max}$  là cận trên, dễ dàng chứng minh được:

$$u_i \equiv \frac{1}{2} \left[ \sum_p (F_p + F_p) + \sum_{j \neq i} f_j \right] \leq u_i^{\max}$$

Giả sử nút  $i$  trong  $G$  có bậc là  $d_i$  và các bậc cận trên và dưới là  $d_i^{\max}$  và  $d_i^{\min}$ , ta có:

$$d_i^{\min} \leq d_i \equiv \sum_{j=1}^n (b_j + b_j) \leq d_i^{\max}$$

Gọi  $f_{avg,ij}$  là tổng lưu lượng trung bình trên liên kết  $(i,j)$ , ta có:

$$f_{avg,j} = \sum_{p=1}^n \sum_{q>p} \sum_r a_{j,r}^{\#} h_r^{\#} \frac{F_{avg,\#}}{F_{\#}}$$

Gọi  $\gamma$  là tổng lưu lượng trên mạng, vậy:

$$\gamma = \sum_{p=1}^n \sum_{q>p} F_{\#}$$

Gọi  $T_{max}$  là độ trễ gói trung bình cực đại cho phép, ta có (xem [10]):

$$T = \frac{1}{\gamma} \sum_{(i,j) \in E} \frac{f_{avg,j}}{f_j - f_{avg,j}} \leq T_{max}$$

Bài toán thiết kế mạng có thể được tóm tắt:

$$\min_{G \in S} \sum_{p=1}^n \sum_{q>p} \sum_r C_r^{\#} h_r^{\#} \quad (r1)$$

$$F_{\#} = \sum_r h_r^{\#} \quad (r2)$$

$$C_r^{\#} = \sum_{(i,j)} a_{j,r}^{\#} c_{i,j} \quad (r3)$$

$$f_j = \sum_{p=1}^n \sum_{q>p} \sum_r a_{j,r}^{\#} h_r^{\#} \quad (r4)$$

$$u_i \equiv \frac{1}{2} \left[ \sum_p (F_p + F_p) + \sum_{j \neq i} f_j \right] \leq u_i^{max} \quad (r5)$$

$$0 \leq f_j \leq f_j^{max} \quad (r6)$$

$$\sum_{(i,j)} a_{j,r}^{\#} \leq H^{max} \quad (r7)$$

$$d_i^{min} \leq d_i \equiv \sum_{j=1}^n (b_j + b_j) \leq d_i^{max} \quad (r8)$$

$$f_{avg,j} = \sum_{p=1}^n \sum_{q>p} \sum_r a_{j,r}^{\#} h_r^{\#} \frac{F_{avg,\#}}{F_{\#}} \quad (r9)$$

$$\gamma = \sum_{p=1}^n \sum_{q>p} F_{\#} \quad (r10)$$

$$T = \frac{1}{\gamma} \sum_{(i,j) \in E} \frac{f_{avg,j}}{f_j - f_{avg,j}} \leq T_{max} \quad (r11)$$

$$h_r^{\#} \geq 0 \quad (r12)$$

Một đồ thị  $G^*$  có (r1) tối thiểu, độ tin cậy tối đa và thỏa các ràng buộc từ (r2) đến (r12) là một mạng tối ưu.

### 3. TỐI ƯU ĐA MỤC TIÊU & CÁC MOEA

#### 3.1 Tối ưu đa mục tiêu

Không mất tính tổng quát, giả thuyết tất cả các mục tiêu cần được tối thiểu hóa - một mục tiêu loại tối thiểu hóa có thể được chuyển thành loại tối đa hóa bằng cách nhân cho -1. Bài toán tối thiểu hóa K mục tiêu được định nghĩa như sau: cho 1 vectơ biến quyết định n chiều  $x = \{x_1, \dots, x_n\}$  trong không gian giải pháp X, tìm vectơ  $x^*$  mà nó tối thiểu tập K hàm mục tiêu đã cho  $z(x^*) = \{z_1(x^*), \dots, z_K(x^*)\}$ . Không gian giải pháp X nói chung bị hạn chế bởi 1 chuỗi các ràng buộc có dạng  $g_j(x^*) = b_j$  ( $j=1, \dots, m$ ).

Một giải pháp khả thi  $x$  được gọi là vượt trội giải pháp  $y$  ( $\cdot$ ), nếu và chỉ nếu,  $z_i(x) \leq z_i(y)$  ( $i=1, \dots, K$ ) và  $z_j(x) < z_j(y)$  ở ít nhất một mục tiêu  $j$ . Một giải pháp được nói là tối ưu Pareto nếu nó không bị vượt trội bởi 1 giải pháp nào trong không gian giải pháp. Tập tất cả các giải pháp khả thi không bị vượt trội trong X được gọi là tập tối ưu Pareto. Với tập tối ưu Pareto đã cho, các giá trị hàm mục tiêu tương ứng trong không gian mục tiêu được gọi là Pareto Front. Mục tiêu của các thuật toán tối ưu đa mục tiêu là xác định các giải pháp trong tập tối ưu Pareto. Thực tế, việc chứng minh một giải pháp là tối ưu thường không khả thi về mặt tính toán. Vì vậy, một tiếp cận thực tế với bài toán tối ưu đa mục tiêu là tìm kiếm tập các giải pháp là thể hiện tốt nhất có thể của tập tối ưu Pareto, một tập các giải pháp như vậy được gọi là tập Best-known Pareto.

#### 3.2. Các MOEA

GA là hướng tiếp cận dựa trên quần thể, đặc biệt phù hợp để giải quyết các bài toán tối ưu đa mục tiêu. Các GA truyền thống có thể được biến đổi để tìm kiếm tập Best-known Pareto trong bài toán tối ưu đa mục tiêu.

MOEA đầu tiên được biết là Vector Evaluated Genetic Algorithm (VEGA) được đề nghị bởi Schaffer [11]. Sau đó, nhiều MOEA khác đã được phát triển gồm Multi-objective Genetic Algorithm (MOGA) [12], Niche Pareto Genetic Algorithm (NPGA) [13], Weight-Based Genetic Algorithm (WBGA) [14], Random Weight Genetic Algorithm (RWGA) [15], Nondominated Sorting Genetic Algorithm (NSGA) [16], Strength Pareto Evolutionary Algorithm (SPEA) [17], SPEA cải tiến (SPEA2) [18], Pareto-Archived Evolution Strategy (PAES) [19], Pareto Enveloped-based Selection Algorithm (PESA) [20], Region-based Selection in Evolutionary Multiobjective Optimization (SPEA-II) [21], Fast Non-dominated Sorting Genetic Algorithm (NSGA-II) [22], Rank-Density Based Genetic Algorithm (RDGA) [23] và Dynamic Multi-Objective Evolutionary Algorithm (DMOEA) [24],... Điểm khác biệt giữa các MOEA nằm ở cách gán độ thích nghi, cách duy trì quần thể ưu tú và các tiếp cận nhằm đa dạng hóa quần thể.

**4. GIẢI PHÁP THỰC HIỆN**

Một cách tổng quát, việc thiết kế mạng bao gồm việc tìm mô hình mạng và xác định lưu lượng cho các đường liên kết. Trong đó, mô hình mạng cần tìm phải liên thông và thỏa ràng buộc về bậc của nút; lưu lượng cho các đường liên kết phải bảo đảm có tổng lưu lượng cung cấp cho từng cặp nguồn - đích bằng với giá trị lưu lượng yêu cầu, cũng như thỏa các ràng buộc về độ trễ, dung lượng nút mạng, dung lượng liên kết và giới hạn số trạm trung gian. Theo hướng tiếp cận của bài báo, thuật toán tiến hóa có nhiệm vụ tìm mô hình mạng. Với mỗi nhiễm sắc thể (NST) - mô hình mạng đã tìm được, thuật toán Monte Carlo được sử dụng để xác định độ tin cậy và thuật toán LP được sử dụng để ấn định lưu lượng tối ưu cho các đường liên kết, từ đó tính ra được chi phí truyền tải của từng mô hình. Một thuật toán

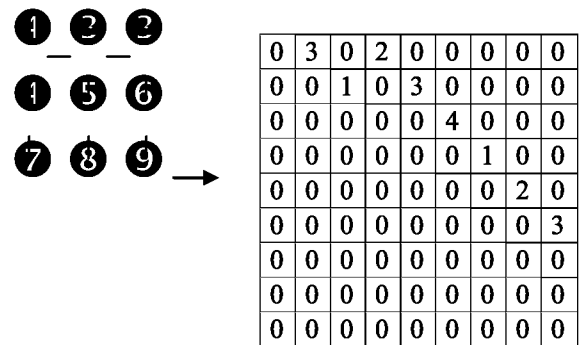
sửa chữa cũng được sử dụng với các mô hình mạng không đáp ứng được ràng buộc về độ trễ gói cực đại cho phép (r11).

**4.1 Biểu diễn NST**

Một đồ thị bất kỳ có thể được biểu diễn duy nhất bằng một ma trận kề nút-nút. Các phần tử của ma trận nhận các giá trị trong khoảng [0, t] tương ứng với loại liên kết (=0 tương ứng với không có liên kết) giữa từng cặp nút hàng-cột. Vì các liên kết là hai chiều, nên chỉ cần xét phần tam giác trên của ma trận. Chọn một thứ tự đọc ma trận tùy ý (ở đây ta chọn đọc theo thứ tự từ trái sang phải, từ trên xuống dưới), ma trận có thể được chuyển thành vector mà không làm mất thông tin (xem hình 1).

Tổng quát, nếu n là số nút trong đồ thị, thì chiều dài NST là:  $n(n-1)/2$  và không gian tìm

kiếm của bài toán là:  $(t+1)^{\frac{n(n-1)}{2}}$



302000001030000004000001000020003000

**Hình 1: Ví dụ biểu diễn của một NST (t=4)**

**4.2 Khởi tạo quần thể**

Quần thể ban đầu được khởi tạo ngẫu nhiên theo nhiều phương pháp khác nhau, các cá thể chỉ được chọn khi chúng là biểu diễn của một mạng liên thông và thỏa ràng buộc về bậc của nút. Phần lớn các cá thể được tạo theo thuật giải:

*begin algorithm*  
*{đi là thứ bậc của nút i, dimax là cận trên của di}*  
 $L = \{ \}$   
 Chọn một nút giữa ngẫu nhiên  $i$  trong  $N$   
 Gọi thủ tục *start\_from\_node(i)*  
**end algorithm**  
*procedure start\_from\_node(j)*  
*while (dj < djmax)*  
 Chọn một nút ngẫu nhiên  $m \in N, m \neq j, (j,m) \notin L$   
 If ( $dm < dmax$ )  
 Thêm cạnh  $(m,j)$  vào  $L$   
 Gọi thủ tục *start\_from\_node(m)*  
*endif*  
*end while*  
*end procedure*  
 Một số cá thể khác có thể được tạo theo cách tạo cây phủ tối thiểu ngẫu nhiên:  
*begin algorithm*  
 $L = \{ \}$   
 Chọn 1 nút bắt đầu ngẫu nhiên  $i \in N$   
 $C = \{ i \}$   
*repeat*  
 Chọn một nút ngẫu nhiên  $d \notin C$   
 Chọn một nút ngẫu nhiên  $c \in C$  thỏa  $dc < dmax$   
 $L = L \cup \{ (c,d) \}$   
 $C = C \cup \{ d \}$   
*until C = N*  
*end algorithm*

#### 4.3 Tính toán giá trị của các hàm mục tiêu (GTTN)

Được thực hiện gồm các bước:

- B1: Xây dựng mô hình mạng từ NST đã cho;  
 B2: Dùng thuật giải Monte Carlo (xem [8]) tính toán độ tin cậy của NST: các NST có độ tin cậy được đánh giá dựa vào xác suất NST vẫn duy trì được tính liên thông khi loại bỏ một hoặc nhiều các liên kết được lựa chọn một cách ngẫu nhiên. Các liên kết có độ tin cậy cao hơn sẽ có xác suất được lựa chọn để loại bỏ thấp hơn;  
 B3: Tính toán chi phí truyền tải lưu lượng

(r1), gồm 2 bước con:

B3a: Dùng thuật giải ở [5] tìm r đường đi ngắn nhất giữa từng cặp nút thỏa (r7);

B3b: Dùng LP (xem [4]) phân bổ lưu lượng theo các đường đi đã tìm ở bước B3a, thỏa các ràng buộc về lưu lượng của bài toán, đồng thời tối thiểu hóa tổng chi phí truyền tải lưu lượng (r1);

B4: Dùng thuật toán sửa chữa, sau đó lặp lại các bước B2 và B3 nếu mô hình mạng không thỏa ràng buộc về độ trễ gói cực đại cho phép (r11).

#### 4.4 Chọn lọc (select)

MOEA chọn lọc các NST cho việc sinh sản ngẫu nhiên, cơ hội được chọn tùy thuộc vào GTTN của chúng. Mỗi MOEA có cách chọn lọc khác nhau:

- Chọn lọc dựa vào tỷ lệ: từ tập các NST và các GTTN, ta có thể tạo ra một bộ chọn lọc ngẫu nhiên tương tự như một bánh xe rulét (xem [1]), các NST có GTTN tốt hơn ánh xạ tương ứng với phần lớn hơn.
- Chọn lọc dựa vào thứ hạng Pareto (Pareto-ranking): các NST có thứ hạng Pareto thấp sẽ có cơ hội được chọn lọc cao hơn.
- Chọn lọc dựa vào đấu loại trực tiếp: hai NST được chọn lựa ngẫu nhiên để đấu loại, NST có GTTN tốt hơn sẽ là người chiến thắng.

#### 4.5 Lai tạo (crossover)

Thuật giải ở đây dùng phép lai đồng dạng để lai tạo quần thể, hai mạng cha mẹ được chọn để tạo ra một mạng con mới theo cách: nếu cả cha và mẹ cùng sở hữu một liên kết, mạng con cũng sẽ có liên kết đó; nếu cả cha và mẹ cùng không có, mạng con cũng sẽ không có; nếu chỉ một trong cha hoặc mẹ có liên kết thì mạng con cũng sẽ có với xác suất 50%. Phép lai tạo này đảm bảo mạng con sẽ thừa hưởng các đặc tính chung của cả cha và mẹ. Các mô hình mạng sau khi lai tạo được kiểm tra tính hợp lệ và sửa chữa để đảm bảo chỉ những mô hình liên thông và thỏa ràng buộc về bậc của nút được đưa sang thế hệ kế tiếp.

#### 4.6 Đột biến (mutation)

Trong biểu diễn NST, các gen tượng trưng cho loại liên kết, gen có giá trị 0 nếu không có liên kết. Trong quá trình đột biến, việc bỏ đi một liên kết sẽ không bao giờ cải thiện được GTTN của NST, vì phép toán tuyến tính phát sinh sẽ trở nên ràng buộc chặt chẽ hơn. Vì vậy, chúng ta chọn giải pháp: chọn ngẫu nhiên 1 gen trong NST; nếu gen có giá trị 0 ta sẽ thiết lập gen là một số ngẫu nhiên có giá trị trong khoảng  $[1, t]$ . Một lần nữa việc kiểm tra tính hợp lệ và sửa chữa các NST lại được thực hiện.

#### 4.7 Sửa chữa (repair)

Không phải tất cả các NST được khởi tạo ngẫu nhiên, lai tạo hay đột biến là biểu diễn của một mạng liên thông hoặc thỏa các ràng buộc về độ trễ và bậc của nút, vì vậy quá trình sửa chữa là cần thiết. Khi sửa chữa, mục đích của ta là tạo ra một giải pháp hợp lệ bằng một vài thay đổi. Tính liên thông dễ dàng được kiểm tra với chi phí không quá lớn bằng cách dùng thuật toán tìm kiếm theo chiều sâu trước (Depth First Search) (xem [2]). Nếu một đồ thị không liên thông, các liên kết ngẫu nhiên được bổ sung giữa các thành phần liên thông cho đến khi đồ thị liên thông hoàn toàn. Nếu bậc của một nút nhỏ hơn cận dưới của ràng buộc, một hoặc nhiều hơn các liên kết sẽ được bổ sung. Nếu bậc của một nút lớn hơn cận trên của ràng buộc, một hoặc nhiều hơn các liên kết sẽ được bỏ đi, nhưng vẫn phải đảm bảo tính liên thông của đồ thị.

Với một mạng đã liên thông và thỏa ràng buộc về bậc của nút ( $r_7$ ), tất cả các đường đi thỏa ràng buộc về giới hạn số trạm trung gian ( $r_6$ ) được tạo ra, thủ tục LP được sử dụng để tìm phân bổ lưu lượng trên các đường đi thỏa các ràng buộc về dung lượng của nút ( $r_4$ ) và liên kết ( $r_5$ ), cũng như đảm bảo sao cho tổng lưu lượng phân bổ cho từng cặp nút phải bằng với ma trận lưu lượng yêu cầu ( $r_2$ ), đồng thời tối thiểu hóa tổng chi phí truyền tải lưu lượng ( $r_1$ ). Nếu LP không tìm ra giải pháp, chúng cũng được sửa chữa. Để sửa chữa, một thủ tục nhỏ được gắn liền với thủ tục LP nhằm tìm ra các liên kết hoặc nút quá tải. Nếu liên kết ( $i,j$ )

bị quá tải, một liên kết thứ hai giữa nút  $i$  và nút  $j$  được tạo ra. Điều này được thực hiện bằng cách chọn ngẫu nhiên một nút thứ ba  $k$  và bổ sung vào các liên kết ( $i,k$ ) và ( $k,j$ ). Nếu nút  $i$  bị quá tải, một liên kết vòng qua  $i$  được tạo ra bằng cách chọn hai nút  $j$  và  $k$  nằm liền kề  $i$ , tức đồ thị đã tồn tại các liên kết ( $i,j$ ) và ( $i,k$ ), sau đó bổ sung vào liên kết ( $j,k$ ).

Với một mạng có độ trễ gói trung bình cao hơn mức mong muốn ( $T_{max}$ ), điều này đồng nghĩa với việc có một vài liên kết nào đó có lưu lượng trung bình xấp xỉ với lưu lượng cho phép. Trong trường hợp như vậy, độ trễ gói trung bình của mạng có thể được cải thiện bằng cách thêm vào 1 liên kết nhằm chia tải với các liên kết bị quá tải. Để tìm ra liên kết ứng thí tốt nhất, lần lượt các liên kết bị quá tải được loại bỏ khỏi mạng cho đến khi mạng được tách thành 2 mạng con riêng biệt  $G_1$  và  $G_2$  (tức là  $V_1 \cap V_2 = \emptyset$ ). Các liên kết bị loại bỏ thiết lập thành tập  $S$ . Liên kết ứng thí là liên kết có chi phí nhỏ nhất  $\{i,j\}$  thỏa  $i \in V_1, j \in V_2$  và  $\{i,j\} \notin S$ . Tuy nhiên, thủ tục này đôi khi thất bại trong việc tìm ra một liên kết như vậy, đặc biệt khi mạng có kết nối dày đặc. Trong trường hợp này, thuật toán sẽ tìm kiếm đường dẫn với chi phí truyền tải cao nhất trong mạng, liên kết ứng thí là liên kết trực tiếp giữa 2 nút ở cuối đường dẫn vừa tìm.

#### 4.8 Phát triển các tầng lớp ưu tú (elitism)

Do phép chọn lọc và lai tạo được thực hiện một cách ngẫu nhiên, không đảm bảo các NST không bị vượt trội sẽ hiện hữu trong thế hệ kế tiếp. Cách giải quyết phổ biến là chọn giữ lại những NST không bị vượt trội được sản sinh của mỗi thế hệ.

#### 4.9 Đảm bảo quần thể đa dạng và nhỏ

Ta chọn cách thức: sau khi lai tạo, tất cả các NST được so sánh với nhau. Vì các NST giống nhau không thêm bất kỳ thông tin nào. Nên ta có thể loại bỏ chúng mà không ảnh hưởng đến sự tiến triển của quần thể.

#### 4.10 So sánh trước - kiểm tra sau

Điểm yếu của LP là tốn nhiều thời gian tính toán. Lợi dụng đặc điểm các quần thể GA thường có độ hội tụ cao, nên trước khi tính toán GTTN của một NST, ta so sánh nó với

tất cả các thành viên đã được tính ở các thể hệ trước (số thể hệ tiền sử được lưu trữ tùy thuộc vào dung lượng bộ nhớ). Các NST giống nhau sẽ có cùng GTTN, nên việc tính lại là không cần thiết.

4.11. Lược bỏ

Quá trình sửa chữa và phép đột biến thường thêm vào các liên kết. Quần thể sẽ hướng tới một đồ thị liên thông hoàn toàn (với ràng buộc bậc của nút cho phép). Vì vậy chất liệu di truyền dư thừa sẽ được sản sinh qua các thế hệ tương lai. Giải pháp được chọn là tìm các liên kết không cần thiết và lược bỏ chúng.

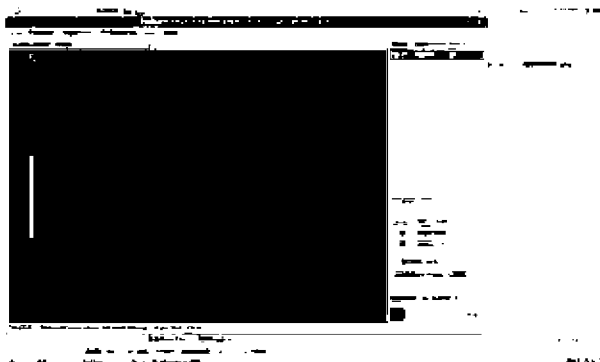
4.12. Khả năng tương tác

Việc cho phép tinh chỉnh các thông số trong thời gian thực có thể cải tiến hiệu năng của hệ thống. Bằng cách thay đổi các thông số và sử dụng các toán tử lai tạo, đột biến và sinh sản ngẫu nhiên trong quần thể, ta có thể nghiên cứu giá trị của các chiến lược mới mà không cần thay đổi mã chương trình. Việc tương tác cũng cho phép thu thập các thông tin cần quan tâm ở bất kỳ giai đoạn nào.

5. KẾT QUẢ THỰC NGHIỆM

Chúng tôi đã xây dựng phần mềm trên cơ sở dùng ngôn ngữ C++ để thể hiện các thuật toán và hình ảnh đồ họa 3 chiều của tập các giải pháp tối ưu trong không gian mục tiêu (hình 2). Theo hướng tiếp cận Pareto các MOEA được hiện thực gồm: NSGA, NSGAIL, NSGAIIC, SPEA; theo hướng tiếp cận HGA các MOEA được hiện thực gồm: PMA, IMMOGLS, MOMGLS,...

Các kết quả được trình bày ở đây có được từ việc chạy các MOEA khác nhau để thiết kế tối ưu một mạng viễn thông có: 24 nút, 55 liên kết, 396 lưu lượng yêu cầu giữa từng cặp nút nguồn - đích (đây là dữ liệu và mô hình mẫu có mã hiệu ta1--U-U-L-N-C-A-Y-N do Telekom Austria đề xuất, được lấy từ thư viện chứa các mẫu kiểm thử dành cho các cộng đồng nghiên cứu trên thế giới nhằm chuẩn hóa việc kiểm tra benchmark, đánh giá và so sánh giữa các mô hình và thuật toán thiết kế tối ưu mạng viễn thông cố định được đặt tại website <http://sndlib.zib.de>).



Hình 2: Hình ảnh đồ họa 3 chiều của tập các giải pháp tối ưu trong không gian mục tiêu

MOEA	N <sub>E</sub>	Cost	Reliable	Time
NSGA	1	3295	0.7910	75p15g
	2	3190	0.7777	
	3	2965	0.7595	
	4	2960	0.7413	
	5	2870	0.7294	
NSGAIL	1	3095	0.7819	18p29g
	2	2965	0.7735	
	3	2945	0.7357	
	4	2910	0.7350	
NSGAIIC	1	3105	0.7749	20p46g
	2	3075	0.7651	
	3	2915	0.7490	
SPEA	1	3220	0.7854	35p37g
	2	3095	0.7840	
	3	3000	0.7721	
	4	2990	0.7378	
	5	2870	0.7364	

Bảng 1: Kết quả thực nghiệm với các MOEA

Tóm tắt kết quả thử nghiệm của chúng tôi với các MOEA khác nhau được thể hiện ở bảng 1 và bảng 2. Trong bảng 1, cột NE thể hiện số thứ tự của các cá thể trong tập Best-known Pareto, cột Cost thể hiện chi phí truyền tải lưu lượng, cột Reliable thể hiện độ tin cậy và cuối cùng cột Time thể hiện thời gian thực thi của từng thuật toán. Trong bảng 2, cột N thể hiện số giải pháp trong tập Best-known Pareto tổng có được bằng cách: hợp các tập Best-known Pareto của mỗi MOEA và chọn ra các giải pháp không bị vượt trội (là các giải pháp được tô đậm trong bảng 1); ứng với từng

thuật toán, cột N1 thể hiện số giải pháp trong tập Best-known Pareto, cột N2 thể hiện số giải pháp trong tập Best-known Pareto có trong tập Best-known Pareto tổng, cột N3 thể hiện số giải pháp trong tập Best-known Pareto bị vượt trội so với các giải pháp trong tập Best-known Pareto tổng và cuối cùng cột N4 thể hiện khoảng cách Euclid trung bình giữa các giải pháp trong tập Best-known Pareto. Qua so sánh kết quả giữa các MOEA ta nhận thấy: NSGA có thời gian xử lý chậm nhất, chất lượng giải pháp thấp và phân bố không rộng trong không gian mục tiêu; SPEA cho kết quả tốt hơn và tốt nhất cả về tiêu chí thời gian lẫn chất lượng giải pháp thuộc về NSGAII và NSGAIII.

Bảng 2: So sánh kết quả giữa các MOEA

MOEA	N <sub>1</sub>	N <sub>2</sub>	N <sub>3</sub>	N <sub>4</sub>	Time
NSGA	5	1	4	7	0.74142 75p15g
NSGAII	4	2	2		0.88188 18p29g
NSGAII C	3	2	1		0.95697 20p46g
SPEA	5	2	3		0.80313 37p37g

## 6. KẾT LUẬN

Phương pháp tối ưu đa mục tiêu dùng thuật toán tiến hóa là vấn đề mới trong phân tích và thiết kế mạng với nhiều ràng buộc phức tạp. Kết quả đạt được đã chứng minh tính hiệu quả và đúng đắn của phương pháp tối ưu này. Tuy nhiên, vẫn còn một số vấn đề cần tiếp tục nghiên cứu sâu hơn:

- Bài toán chưa đặt vấn đề thiết kế mạng với khả năng dự phòng và cấu hình lại khi có sự cố làm mất một hoặc nhiều liên kết,...
- Bài toán chưa được tiếp cận theo hướng dùng một số các thuật toán đa mục tiêu khác MOEA thuộc lớp meta-heuristic như: thuật toán mô phỏng luyện kim (SA),...
- Đối với từng bài toán, hoặc mỗi giai đoạn nhất định trong quá trình giải quyết bài toán, việc chọn thuật toán meta-heuristic hoặc thay đổi các tham số phù hợp để có được kết quả tối ưu đóng vai trò quan trọng, bài báo chưa đặt ra vấn đề này.

- Cần xem xét thêm trường hợp có sự kết hợp với việc định tuyến và điều khiển tối ưu,... để giải bài toán một cách tổng thể.

## TÀI LIỆU THAM KHẢO

- [1] Hoàng Kiếm, Lê Hoàng Thái, 2000, *Thuật giải di truyền*, Nxb Giáo dục.
- [2] T.H. Cormen, C.E. Leiserson, R.L. Rivest, 1997, *Algorithms*, MIT Press.
- [3] Lương Hồng Khanh, 2002, Tạp chí Bưu chính viễn thông, số 197, *Ứng dụng thuật toán tiến hóa trong việc tối ưu hóa các tham số chất lượng mạng*, trang 42-45.
- [4] Đặng Hân, 1994, *Quy hoạch tuyến tính*, Trường Đại học kinh tế Tp.HCM.
- [5] Martins, Pascoal, Santos, 1998, *The k shortest paths problem*, Universidade de Coimbra, PORTUGAL.
- [6] K.T Ko, K.S. Tang et al., 1997, *Computer Journal*, No. 30, *Using Genetic Algorithms to Design Mesh Networks*.
- [7] L. Berry, B. Murtagh, S. Sugden và G. McMahon, 1995, *Application of a Genetic-based Algorithm for Optimal Design of Tree-structured Communications Networks*, Proceedings of the Regional Teletraffic Engineering Conference of the International Teletraffic Congress, South Africa, September 1995, pp. 361-370.
- [8] Greg Kochanski, 2005, *Monte Carlo Simulation*. <http://kochanski.org/gpk>.
- [9] S. Duarte, B. Barán and D. Benítez, 2001, *Telecommunication network design with parallel multi-objective evolutionary algorithms*. In Proceedings of XXVII Conferencia Latinoamericana de Informática CLEI'2001, Merida, Venezuela, 2001.
- [10] Konak A. and Smith A., 1999, *A Hybrid Genetic Algorithm Approach for Backbone Design of Communication Networks*. Proceedings of the 1999 Congress on Evolutionary Computation, Washington D. C., IEEE, 1999.
- [11] Schaffer JD. Multiple objective optimization with vector evaluated genetic algorithms. In: Proceedings of the international conference on genetic algorithm and their applications, 1985.
- [12] Fonseca CM, Fleming PJ. Multiobjective ...