

TRƯỜNG ĐẠI HỌC BÁCH KHOA HÀ NỘI

**VIỆN ĐIỆN TỬ - VIỄN THÔNG**



BÁO CÁO BÀI TẬP LỚN

**Tổ Chức Quy Hoạch**

**Mạng Viễn Thông**

**Đề tài:**

**Thiết Kế Phần Mềm**

**Tính Toán Mạng Viễn Thông**

Giảng Viên Hướng Dẫn: TS. Nguyễn Xuân Dũng

Học Kì: 20201

Nhóm: 6

|  |  |  |
| --- | --- | --- |
| Nguyễn Xuân Thiện | 20163896 | ĐT 6 – K61 |
| Phạm Quang Tuấn | 20164406 | ĐT 6 – K61 |
| Nguyễn Đình Quang | 20163306 | ĐT 6 – K61 |
| Phạm Đình Linh | 20162474 | ĐT 6 – K61 |

Hà Nội, 12-2020

# Nhận Xét

# Lời Giới Thiệu

Ngày nay, các hệ thống thông tin viễn thông được xem như phương tiện hiệu

quả nhất để trao đổi thông tin và truyền số liệu. Các mạng viễn thông điện tử đã trở

nên ngày càng phức tạp hơn và có khuynh hướng kỹ thuật cao nhằm đáp ứng nhu

cầu về gia tăng các dịch vụ viễn thông tiên tiến, có chất lượng cao. Do đó việc tổ

chức một mạng viễn thông nhằm tạo ra một mạng viễn thông có đủ khả năng đáp

ứng các yêu cầu trên đóng một vai trò rất quan trọng.

Muốn xây dựng một mạng viễn thông thì ta phải thiết lập được mô hình phân

bố (Topology) của các phần tử mạng. Có nhiều thuật toán được sử dụng để thiết kế

Topology mạng. MENTOR (Mesh Network Topology Optimization Routing) là

một thuật toán rất thích hợp cho việc thiết kế các mạng thông tin vì nó không phụ

thuộc vào đặc điểm của bất kỳ một công nghệ hay kiến trúc mạng nào mà chỉ phụ

thuộc vào nguyên tắc thiết kế mạng. MENTOR có thể ứng dụng cho nhiều loại

mạng, điển hình là mạng ATM.

Biết được tầm quan trọng của việc thiết kế Topology mạng, nhóm đã chọn và

hoàn thành đề tài “**Viết một phần mềm tạo topology mạng viễn thông theo thuật**

**toán Mentor**”. Tuy nhiên, do thời gian cũng như kiến thức còn hạn chế nên không

thể tránh khỏi một vài sai sót. Chúng em xin gửi lời cảm ơn tới thầy giáo Nguyễn

Xuân Dũng đã tận tình hướng dẫn giúp chúng em hoàn thành bài tập và mong nhận

được những lời góp ý của Thầy để đề tài của chúng em có thể hoàn thiện hơn.

# Mục Lục

[Nhận Xét 1](#_Toc58279093)

[Lời Giới Thiệu 2](#_Toc58279094)

[Mục Lục 3](#_Toc58279095)

[Danh Mục Hình Ảnh 4](#_Toc58279096)

[Danh Mục Bảng Biểu 4](#_Toc58279097)

[Danh Mục Từ Viết Tắt 4](#_Toc58279098)

[Phân Công Công Việc 5](#_Toc58279099)

[Nội Dung 6](#_Toc58279100)

[1. Dẫn nhập 6](#_Toc58279101)

[1.1. Tổng quan về môn học [1] 6](#_Toc58279102)

[1.2. Thuật toán MENTOR [2] 6](#_Toc58279103)

[1.3. Thuật toán Prim - Dijsktra [3] 6](#_Toc58279104)

[2. Cơ sở lý thuyết 7](#_Toc58279105)

[2.1. Thuật toán MENTOR 7](#_Toc58279106)

[2.2. Thuật toán Prim - Dijsktra 11](#_Toc58279107)

[3. Thiết kế phần mềm 11](#_Toc58279108)

[3.1. Tổng quan chức năng chính 11](#_Toc58279109)

[3.2. Một số hình ảnh chương trình 12](#_Toc58279110)

[4. Kết Luận 15](#_Toc58279111)

[Danh Mục Tài Liệu Tham Khảo 16](#_Toc58279112)

# Danh Mục Hình Ảnh

[Hình 3.1 Sơ đồ khối chức năng 16](file:///D:\Storage\HUST\20182\To%20Chuc%20Quy%20Hoach%20Mang%20Vien%20Thong\BTL\20182_BaoCao_TCQHMVT_Nhom1.docx#_Toc8108508)

[Hình 4.1 Các nút tạo với vị trí ngẫu nhiên 21](file:///D:\Storage\HUST\20182\To%20Chuc%20Quy%20Hoach%20Mang%20Vien%20Thong\BTL\20182_BaoCao_TCQHMVT_Nhom1.docx#_Toc8108509)

[Hình 4.2 Tìm Backbone 21](#_Toc8108510)

[Hình 4.3 Tìm cây truy nhập 22](#_Toc8108511)

[Hình 4.4 Các nút được tạo với vị trí tuần tự 22](#_Toc8108512)

[Hình 4.5 Tìm Backbone 23](#_Toc8108513)

[Hình 4.6 Tìm cây truy nhập 23](#_Toc8108514)

[Hình 4.7 Tìm nút backbone 24](#_Toc8108515)

[Hình 4.8 Tìm cây truy nhập 24](#_Toc8108516)

# Danh Mục Bảng Biểu

[Bảng 3.1 Thuộc tính dữ liệu nút 17](#_Toc8075019)

[Bảng 3.2 Thủ tục thuật toán MENTOR 18](#_Toc8075020)

[Bảng 3.3 Thủ tục thuật toán Esau William 19](#_Toc8075021)

# Danh Mục Từ Viết Tắt

|  |  |  |
| --- | --- | --- |
| Từ Viết Tắt | Tiếng Anh | Tiếng Việt |
| NW | Normalized Weight | Trọng số chuẩn hóa |
| IEEE | Institute of Electrical and Electronics Engineers | Hiệp hội kỹ sư điện điện tử |

# Phân Công Công Việc

|  |  |
| --- | --- |
| Nguời thực hiện | Công việc |
| Phạm Quang Tuấn | Code  Phân chia công việc |
| Nguyễn Đình Quang | Code |
| Nguyễn Xuân Thiện | Code  Thu thập tài liệu lí thuyết |
| Phạm Đình Linh | Làm báo cáo |

# Nội Dung

## Dẫn nhập

### Tổng quan về môn học [1]

Đối với kỹ sư điện tử viễn thông chuyên ngành kỹ thuật thông tin truyền thông, học phần Tổ chức quy hoạch mạng viễn thông sẽ cung cấp các kiến thức sau:

* Cung cấp kiến thức tổng quan về mạng viễn thông, các mạng cung cấp dịch vụ viễn thông, Các công nghệ viễn thông mới, các xu hướng phát triển mạng …
* Miêu tả các kỹ thuật và các giải pháp thích hợp để thiết kế các loại mạng khác nhau
* Đưa ra phương pháp để đánh giá thiết kế mạng

### Thuật toán MENTOR [2]

Thuật toán định tuyến MENTOR là một thuật toán để sử dụng trong việc định tuyến các mạng xương sống. Nó được phát triển bởi Aaron Kershenbaum, Parviz Kermani và George A. Grove và được xuất bản bởi IEEE vào tháng 4 năm 1991 với bài báo tiêu đề *“MENTOR an algorithm for mesh network topological optimization and routing”* trong IEEE Transactions on Communications Journal.

Thuật toán này ra đời đã tạo ra cơ chế xây dựng mạng xương sống với độ phức tạp tính toán giảm xuống, giảm thời gian tính toán thiết kế mạng đi đáng kể (theo nhóm tác giả là gấp hàng trăm lần các phương pháp thời bấy giờ). Nguyên do là năm 1991 các máy vi tính thời ấy chưa cho hiệu năng tính toán mô phỏng cao. Đồng thời, thuật toán này xây dựng thành công mạng có chi phí thấp, hiệu năng cao và các liên kết dung lượng lớn.

### Thuật toán Prim - Dijsktra [3]

Trong khoa học máy tính, thuật toán Prim là một thuật toán tham lam để tìm cây bao trùm nhỏ nhất của một đồ thị vô hướng có trọng số liên thông. Nghĩa là nó tìm một tập hợp các cạnh của đồ thị tạo thành một cây chứa tất cả các đỉnh, sao cho tổng trọng số các cạnh của cây là nhỏ nhất. Thuật toán được tìm ra năm 1930 bởi nhà toán học người Séc Vojtěch Jarník và sau đó bởi nhà nghiên cứu khoa học máy tính Robert C. Prim năm 1957 và một lần nữa độc lập bởi Edsger Dijkstra năm 1959. Do đó nó còn được gọi là thuật toán DJP, thuật toán Jarník, hay thuật toán Prim–Jarník.

## Cơ sở lý thuyết

### Thuật toán MENTOR

Thuật toán MENTOR nhằm xác định nút Backbone và những nút truy nhập của nút Backbone đó. Thực hiện thuật toán qua bốn bước sau [4]:

Bước 1: Xác định nút Backbone dựa trên ngưỡng lưu lượng

Bước 2: Tính giá trị Maxcost

Bước 3: Xác định nút truy nhập của nút Backbone đã tìm được

Bước 4: Xác định nút Backbone và nút truy nhập cho các nút còn lại dựa trên giá trị thưởng

Dưới đây trình bày chi tiết các bước thực hiện.

* 1. Xác định nút Backbone dựa trên ngưỡng lưu lượng
* Trọng số của một nút W(i) là tổng lưu lượng vào ra của nút đó.
* Liên kết dung lượng là C.
* Trọng số chuẩn hóa của nút i là
* Nếu nút i có thì nút i được chọn là nút Backbone, với W là tham số cho trước.
  1. Tính MaxCost
  2. Xác định nút truy nhập của nút Backbone đã tìm được
* Bán kính mạng truy nhập:
* Ta có
* Từ nút Backbone i, vẽ đường tròn bán kính R.Maxcost, phủ được nút nào thì nút ấy sẽ thành nút truy nhập của nút Backbone i.
  1. Xác định nút Backbone và nút truy nhập cho các nút còn lại dựa trên giá trị thưởng
* Ấn định tọa độ của mỗi nút
* Tìm nút trung tâm trọng lực có tọa độ là và
* Tính
* Lấy và
* Tính giá trị thưởng cho tất cả các nút
* Nút i có giá trị thưởng lớn nhất sẽ được chọn là nút Backbone, tìm các nút truy nhập của nút này.
* Lặp lại bước cho tới khi tất cả các nút được xét.

**Bài toán minh họa**: Cho các nút như hình vẽ dưới, với , , , , , , , , , ,

1

2

4

3

5

7

6

8

**Tính toán**:

Sắp xếp và xét các nút theo trọng số giảm dần, do đó nút 1 sẽ được xét trước, trọng số chuẩn hóa của nút 1 được tính như sau:

Nhận thấy trọng số chuẩn hóa của nút 1 lớn hơn trọng số w = 3.2, ta đặt nút 1 làm nút Backbone.

Tìm nút truy nhập cho nút Backbone 1. Dễ thấy khoảng cách Descartes giữa nút 2 và nút 8 là lớn nhất, nên ta tính Maxcost theo công thức:

Từ giá trị Maxcost, ta tính được bán kính phủ cho nút Backbone 1 là:

Phủ được nút 2 và nút 4 do khoảng cách Descartes từ nút 1 đến bằng 1,4. Sau khi đã tìm được nút Backbone đầu và nút truy nhập của nó, tiếp tục tính trung tâm trọng lực để xác định nút Backbone tiếp theo:

Nút trọng tâm sẽ có tọa độ , nhận thấy, nút 3 xa trung tâm trọng lực nhất, ta có Maxdc tính theo công thức:

Tính thưởng từng nút một, nút nào có thưởng cao nhất thì chọn làm nút Backbone, nhìn vào các tham số để tính thưởng, ta lọc ra được nút backbone mới có thể là vào nút 5 hoặc nút 8, dc(5) = 0,9; dc(8) = 1,3.

Thưởng nút 5:

Thưởng nút 8:

Vậy có thêm nút 8 là nút Backbone, bán kính phủ bằng , vậy nút 6 sẽ là nút truy nhập của nút 8. Xét ba nút còn lại, nút 3, 7, 5:

Tìm trung tâm trọng lực:

Trung tâm trọng lực mới sẽ có tọa độ , nhận thấy rằng, nút 3 là nút xa nhất, và nút 5 là gần nhất, nghi vấn nút 5 sẽ là nút backbone tiếp theo và giá trị maxdc tính từ nút 3.

Tính khoảng cách từ nút 5 đến nút trung tâm:

Khoảng cách từ nút 7 đến nút trung tâm:

Tính giá trị thưởng:

Vậy ta có thêm nút 5 là nút backbone, nút 7 nằm trong bán kính phủ, nên nút 7 sẽ là nút truy nhập của nút 5. Vậy kết thúc ta có, nút backbone 1 gồm nút 2 và nút 4 là nút truy nhập, nút backbone 8 có nút truy nhập 6, nút backbone 5 có nút 7 là nút truy nhập, nút 3 sẽ trở thành nút backbone.

1

2

4

3

5

7

6

8

### Thuật toán Prim – Dijsktra

Thuật toán xuất phát từ một cây chỉ chứa đúng một đỉnh và mở rộng từng bước một, mỗi bước thêm một cạnh mới vào cây, cho tới khi bao trùm được tất cả các đỉnh của đồ thị.

Dữ liệu vào: Một đồ thị có trọng số liên thông với tập hợp đỉnh V và tập hợp cạnh E (trọng số có thể âm). Đồng thời cũng dùng V và E để ký hiệu số đỉnh và số cạnh của đồ thị.

* Khởi tạo: Vmới = {x}, trong đó x là một đỉnh bất kì (đỉnh bắt đầu) trong V, Emới = {}
* Lặp lại cho tới khi Vmới = V:
* Chọn cạnh (u, v) có trọng số nhỏ nhất thỏa mãn u thuộc Vmới và v không thuộc Vmới (nếu có nhiều cạnh như vậy thì chọn một cạnh bất kì trong chúng)
* Thêm v vào Vmới, và thêm cạnh (u, v) vào Emới

Dữ liệu ra: Vmới và Emới là tập hợp đỉnh và tập hợp cạnh của một cây bao trùm nhỏ nhất

## Thiết kế phần mềm

### Tổng quan chức năng chính

Chương trình dưới dạng 1 webapp bằng HTML5, CSS & JavaScript, có tham khảo mã nguồn của các bài tập tương tự viết bằng ngôn ngữ khác.

Các chức năng cơ bản:

• Nhập vào số lượng nút (N).

• Alpha : Là hệ số αdùng trong thuật toán Mentor.

• W ( threshold ) : là giá trị ngưỡng để chọn nút Backbone.

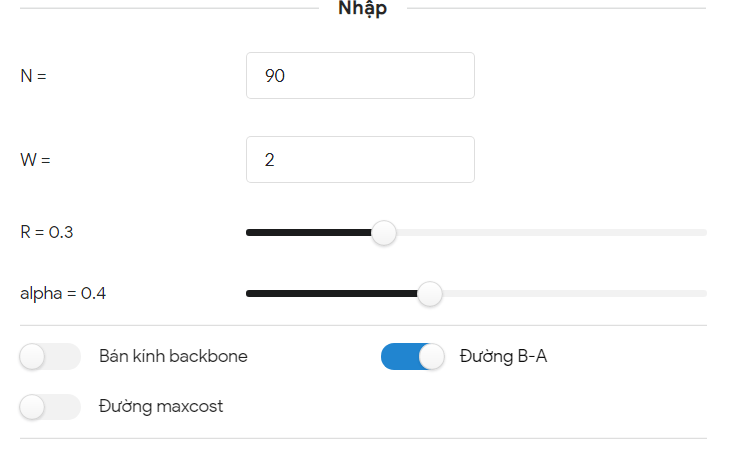
• R : Bán kính của mạng truy nhập.

Sau khi nhập các thông số đầu vào ta nhấn nút Thiết lập mạng để chương trình thực hiện thuật toán MENTOR thiết lập mạng giữa các nút Backbone và mạng truy nhập.

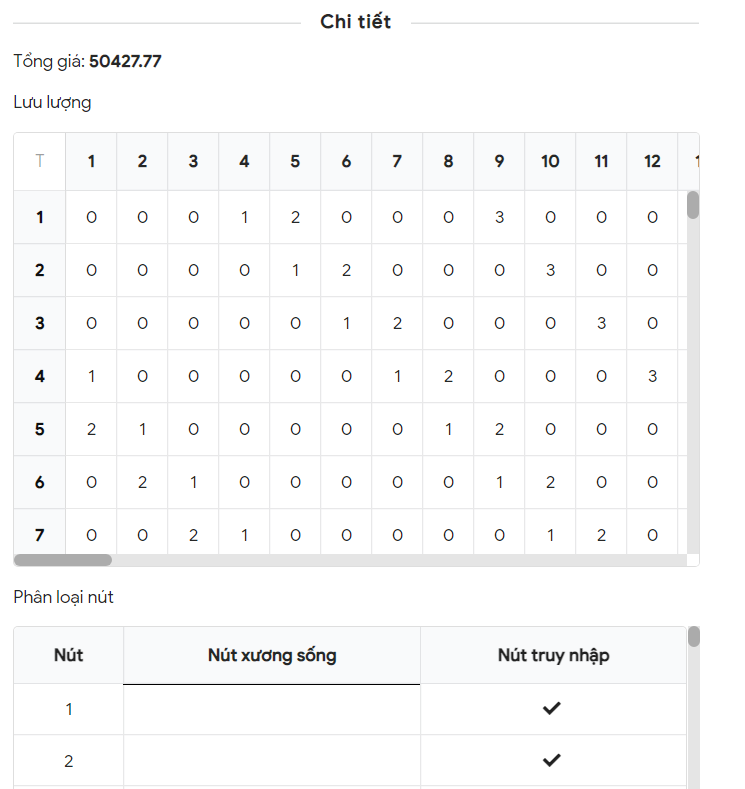
Ta có thể kéo thả các thông số alpha, R và W để thay đổi các thông số cho mạng. Khi kéo thả, mạng sẽ thay đổi theo các thông số mới ngay lập tức để ta có cái nhìn tổng quan hơn.

### Một số hình ảnh chương trình

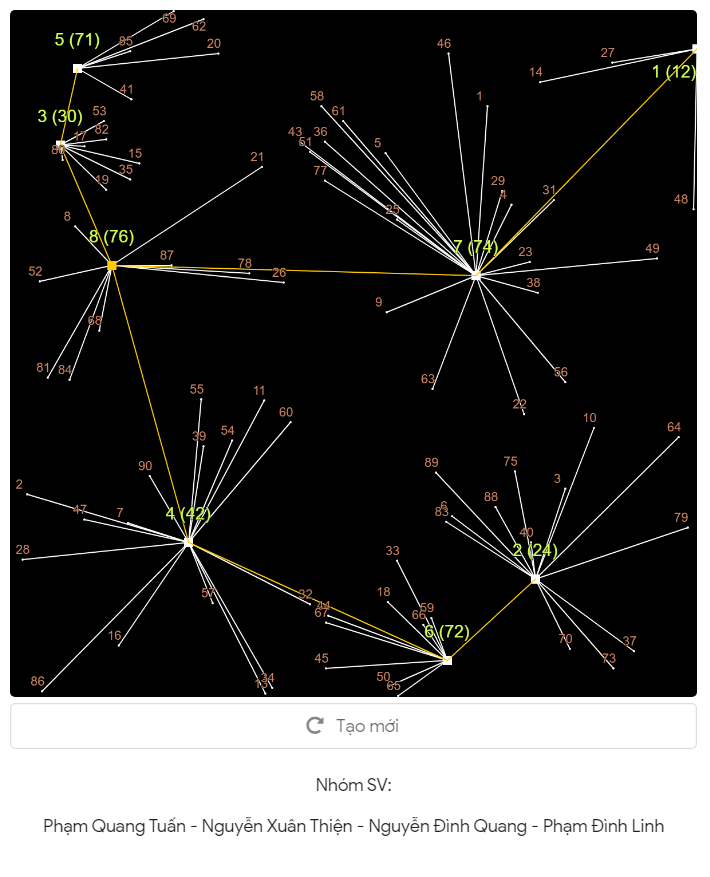
Giao diện nhập số liệu:



Giao diện hiển thị kết quả:



Hiển thị dạng sơ đồ:



## Kết Luận

Qua thời gian làm việc gần một tháng, nhóm đã cơ bản hoàn thành mục tiêu đề tài bài tập lớn. Còn một số vấn đề như giao diện thay đổi tùy biến dữ liệu đầu vào của phần mềm có thể tiếp tục nâng cấp, tiếp tục kiểm thử sửa lỗi cho các thư viện,.. Nếu có thời gian, nhóm sẽ tiếp tục nâng cấp phần mềm trên.

Những kinh nghiệm, kiến thức học được thông qua thời gian thực hiện bài tập lớn sẽ giúp ích cho công việc học tập trên lớp và công việc sau khi tốt nghiệp đại học.

# Danh Mục Tài Liệu Tham Khảo

|  |  |
| --- | --- |
| [1] | Viện Điện Tử Viễn Thông, Đại Học Bách Khoa Hà Nội, Chương Trình Giáo Dục Đại Học, Hà Nội, 2009. |
| [2] | A. Kershenbaum, P. Kermani, G.A. Grover, "MENTOR: an algorithm for mesh network topological optimization and routing," *IEEE Transactions on Communications,* vol. 39, no. 4, pp. 503-513, 1991. |
| [3] | L. R. Esau, K. C. Williams, "A method for approximating the optimal network," *IBM System Journal,* vol. 5, no. 3, pp. 142-147, 1966. |