

KHOA KỸ THUẬT VÀ CÔNG NGHỆ  
BỘ MÔN CÔNG NGHỆ THÔNG TIN



THỰC TẬP ĐỒ ÁN [CƠ SỞ NGÀNH] [CHUYÊN NGÀNH]

HỌC KỲ III, NĂM HỌC 2024-2025

< Nghiên cứu bài toán luồng cực đại trên đồ thị có  
trọng số và cài đặt minh họa >

*Giảng viên hướng dẫn:*  
[ThS./TS.] [Trần Hoàng Nam ]

*Sinh viên thực hiện:*  
Họ tên: Nguyễn Đình Phúc  
MSSV: 170123232  
Lớp: DX23TT10

*Trà Vinh, tháng 04 năm 2025*

KHOA KỸ THUẬT VÀ CÔNG NGHỆ  
BỘ MÔN CÔNG NGHỆ THÔNG TIN



THỰC TẬP ĐỒ ÁN [CƠ SỞ NGÀNH] [CHUYÊN NGÀNH]

HỌC KỲ I, NĂM HỌC 2024-2025

< **Nghiên cứu bài toán luồng cực đại trên đồ thị có  
trọng số và cài đặt minh họa** >

*Giảng viên hướng dẫn:*  
[ThS./TS.] [Trần Hoàng Nam]

*Sinh viên thực hiện:*  
Họ tên: Nguyễn Đình Phúc  
MSSV: 170123232  
Lớp: DX23TT10

## This image shows a full page of white paper with horizontal dotted lines. The lines are evenly spaced and run across the width of the page, providing a guide for handwriting practice. There are no margins, text, or other markings on the page.

**Giáo viên hướng dẫn**  
(Ký tên và ghi rõ họ tên)

[illegible]

**Thành viên hội đồng**  
(Ký tên và ghi rõ họ tên)

## LỜI CẢM ƠN

Với tất cả lòng biết ơn và sự trân trọng, em xin gửi lời cảm ơn sâu sắc nhất đến những người đã đồng hành và hỗ trợ em trong suốt quá trình thực hiện đề tài nghiên cứu này.

Trước hết, em xin bày tỏ lòng biết ơn vô hạn đến ThS./TS. Trần Hoàng Nam, giảng viên hướng dẫn của em. Thầy không chỉ là người truyền đạt kiến thức chuyên môn uyên bác mà còn là người thầy tận tụy, luôn sẵn lòng lắng nghe và giải đáp mọi thắc mắc của em. Sự hướng dẫn tận tình, những lời khuyên quý báu và sự động viên kịp thời của thầy đã giúp em vượt qua những khó khăn, thử thách trong quá trình nghiên cứu. Thầy đã tạo điều kiện thuận lợi nhất để em có thể tập trung vào công việc, phát huy tối đa khả năng của mình và hoàn thành đề tài một cách tốt nhất.

Em xin chân thành cảm ơn quý thầy cô trong Bộ môn Công nghệ Thông tin, Khoa Kỹ thuật và Công nghệ, Trường Đại học Trà Vinh. Quý thầy cô đã không chỉ truyền đạt cho em những kiến thức chuyên ngành sâu rộng mà còn khơi dậy trong em niềm đam mê nghiên cứu khoa học. Những bài giảng tâm huyết, những buổi thảo luận sôi nổi và những lời động viên chân thành của quý thầy cô đã tạo nên một môi trường học tập và nghiên cứu lý tưởng, giúp em trưởng thành và phát triển toàn diện.

Em cũng xin gửi lời cảm ơn sâu sắc đến gia đình, đặc biệt là cha mẹ và anh chị em. Gia đình là nguồn động viên lớn nhất, là điểm tựa vững chắc cho em trong suốt quá trình học tập và nghiên cứu. Sự hy sinh thầm lặng, sự quan tâm ân cần và sự động viên vô điều kiện của gia đình đã giúp em có thêm sức mạnh để vượt qua mọi khó khăn, thử thách. Em xin gửi lời tri ân đặc biệt đến cha mẹ, những người đã luôn tin tưởng và ủng hộ mọi quyết định của em, dù đó là những quyết định khó khăn nhất.

Cuối cùng, em xin gửi lời cảm ơn đến bạn bè và những người thân yêu. Những người đã luôn bên cạnh, lắng nghe, chia sẻ và động viên em trong suốt quá trình thực hiện đề tài. Những lời động viên, những nụ cười và những cái ôm ấm áp của mọi người là nguồn động viên vô giá, giúp em có thêm động lực để hoàn thành tốt công việc của mình. Em xin gửi lời cảm ơn đặc biệt đến những người bạn đã cùng em thức khuya dậy sớm, cùng em vượt qua những đêm dài mất ngủ để hoàn thành đề tài.

Em xin trân trọng cảm ơn tất cả mọi người!

## TÓM TẮT (ABSTRACT)

### Tiếng Việt:

Báo cáo này tập trung vào việc nghiên cứu sâu rộng về bài toán luồng cực đại trên đồ thị có trọng số, một vấn đề quan trọng trong lĩnh vực tối ưu hóa mạng. Mục tiêu chính là tìm ra giá trị luồng tối đa có thể chuyển từ đỉnh nguồn đến đỉnh đích trong một mạng lưới có các cạnh được gán trọng số, đại diện cho khả năng tải hoặc chi phí.

Nghiên cứu bắt đầu bằng việc trình bày chi tiết các khái niệm lý thuyết cơ bản, bao gồm định nghĩa đồ thị có hướng, khả năng thông qua của cạnh, và định luật bảo toàn luồng. Sau đó, báo cáo đi sâu vào phân tích thuật toán Ford-Fulkerson, một trong những thuật toán kinh điển để giải quyết bài toán luồng cực đại.

Để minh họa tính đúng đắn và hiệu quả của thuật toán, báo cáo đã thực hiện cài đặt chương trình bằng ngôn ngữ Python, sử dụng các thư viện như networkx và matplotlib để trực quan hóa đồ thị và kết quả. Các thí nghiệm được tiến hành trên nhiều đồ thị mẫu khác nhau, từ đơn giản đến phức tạp, nhằm đánh giá hiệu năng của thuật toán trong các tình huống thực tế.

Kết quả thí nghiệm cho thấy thuật toán Ford-Fulkerson hoạt động tốt trên các đồ thị nhỏ và vừa, nhưng gặp hạn chế về hiệu năng trên các đồ thị lớn và dày đặc. Báo cáo đã phân tích các nguyên nhân gây ra hạn chế này, đồng thời đề xuất các cải tiến tiềm năng, bao gồm việc áp dụng các thuật toán hiệu quả hơn như Edmonds-Karp và Push-Relabel. Cuối cùng, báo cáo thảo luận về các ứng dụng thực tiễn của bài toán luồng cực đại trong nhiều lĩnh vực, bao gồm giao thông đô thị, mạng máy tính, và quản lý tài nguyên. Báo cáo cũng đề xuất các hướng nghiên cứu tiếp theo, tập trung vào việc phát triển các thuật toán song song và phân tán để giải quyết các bài toán luồng trên đồ thị cực lớn.

### Tiếng Anh:

This report presents a comprehensive study of the maximum flow problem on weighted graphs, a fundamental problem in network optimization. The primary objective is to determine the maximum amount of flow that can be transferred from a source node to a sink node in a network where each edge is assigned a weight, representing its capacity or cost.

The study begins with a detailed exposition of the theoretical foundations, including the definition of directed graphs, edge capacities, and flow conservation constraints. It then delves into the Ford-Fulkerson algorithm, a classical algorithm for solving the maximum flow problem.

To demonstrate the correctness and efficiency of the algorithm, a Python implementation is developed, utilizing libraries such as networkx and matplotlib for graph visualization and result presentation. Experiments are conducted on various sample graphs, ranging from simple to complex, to evaluate the algorithm's performance in practical scenarios.

The experimental results show that the Ford-Fulkerson algorithm performs well on small and medium-sized graphs but encounters performance bottlenecks on large and dense graphs. The report analyzes the causes of these limitations and suggests potential improvements, including the adoption of more efficient algorithms such as Edmonds-Karp and Push-Relabel.

Finally, the report discusses the practical applications of the maximum flow problem in various domains, including urban traffic management, computer networks, and resource allocation. It also proposes future research directions, focusing on the development of parallel and distributed algorithms for solving maximum flow problems on extremely large graphs.

# MỤC LỤC

1. Giới thiệu
2. Cơ sở lý thuyết
  - 2.1. Định nghĩa bài toán luồng cực đại
  - 2.2. Các thuật toán giải luồng
    - 2.2.1. Ford-Fulkerson
    - 2.2.2. Edmonds-Karp
    - 2.2.3. Push-Relabel
  - 2.3. Các biến thể và mở rộng của bài toán luồng cực đại
  - 2.4. Ứng dụng thực tiễn của bài toán luồng cực đại
3. Phương pháp nghiên cứu và cài đặt
  - 3.1. Quy trình nghiên cứu lý thuyết
  - 3.2. Thu thập tài liệu và công cụ hỗ trợ
  - 3.3. Cài đặt minh họa (Python)
    - 3.3.1. Môi trường và thiết lập
    - 3.3.2. Mã nguồn và giải thích
    - 3.3.3. Sơ đồ quy trình thuật toán
4. Kết quả thực nghiệm và phân tích
  - 4.1. Mô tả kịch bản thử nghiệm
  - 4.2. Kết quả thực nghiệm
  - 4.3. Phân tích kết quả và hiệu năng
5. Kết luận và hướng phát triển
6. Tài liệu tham khảo
7. Phụ lục
  - 7.1. Hướng dẫn cài đặt và chạy chương trình
  - 7.2. Tập mã nguồn và hình minh họa
  - 7.3. Giải thích thuật ngữ

# CHƯƠNG 1: GIỚI THIỆU

## 1.1. Bối cảnh và Động lực Chọn Đề Tài:

- **Chi tiết:**

- Trong kỷ nguyên công nghệ số, sự phức tạp của các hệ thống giao thông, mạng máy tính và quản lý tài nguyên đã tăng lên đáng kể. Điều này đòi hỏi các giải pháp tối ưu hóa hiệu quả để đảm bảo hoạt động trơn tru và tiết kiệm chi phí.
- Bài toán luồng cực đại trên đồ thị có trọng số cung cấp một khuôn khổ mạnh mẽ để mô hình hóa và giải quyết các vấn đề tối ưu hóa mạng lưới trong nhiều lĩnh vực.
- Đề tài này được chọn để nghiên cứu sâu về các thuật toán giải luồng cực đại, đặc biệt là thuật toán Ford-Fulkerson, và đề xuất các cải tiến để nâng cao hiệu quả và tính ứng dụng trong thực tế.
- Mở rộng thêm phân tích về sự phát triển của Internet of Things (IoT) và các hệ thống phân tán, nơi bài toán luồng cực đại có vai trò ngày càng quan trọng.
- Nêu rõ hơn về tính cấp thiết của việc tối ưu hóa luồng trong các hệ thống thời gian thực, nơi độ trễ và khả năng đáp ứng nhanh chóng là yếu tố then chốt.
- Thảo luận thêm về sự phát triển của điện toán đám mây và các dịch vụ trực tuyến, nơi bài toán luồng cực đại được ứng dụng để tối ưu hóa việc phân phối dữ liệu và tài nguyên.

- **Ví dụ cụ thể:**

- Trong hệ thống IoT, luồng dữ liệu từ các cảm biến đến trung tâm xử lý cần được tối ưu hóa để đảm bảo thời gian phản hồi nhanh nhất. Ví dụ, trong một hệ thống giám sát giao thông thông minh, dữ liệu từ các cảm biến trên đường cần được truyền tải nhanh chóng đến trung tâm điều khiển để phát hiện và xử lý kịp thời các sự cố.
- Trong các mạng xã hội, luồng thông tin (ví dụ: tin tức, quảng cáo) cần được phân phối hiệu quả đến người dùng mục tiêu. Ví dụ, các thuật toán luồng cực đại có thể được sử dụng để tối ưu hóa việc phân phối quảng cáo đến những người dùng có khả năng quan tâm nhất.
- Trong lĩnh vực logistics, tối ưu hóa luồng hàng hóa từ nhà cung cấp tới người tiêu dùng. Ví dụ, các công ty vận tải có thể sử dụng các thuật toán luồng cực đại để lập kế hoạch tuyến đường tối ưu cho xe tải, giúp giảm thiểu thời gian và chi phí vận chuyển.
- Trong các hệ thống phân phối nội dung (CDN), luồng dữ liệu video và hình ảnh cần được tối ưu hóa để đảm bảo trải nghiệm người dùng mượt mà.
- Trong các hệ thống điều khiển công nghiệp, luồng dữ liệu từ các cảm biến và thiết bị cần được xử lý nhanh chóng để đảm bảo an toàn và hiệu quả hoạt động.
- Trong các hệ thống mạng viễn thông, Luồng dữ liệu thoại, hình ảnh, video cũng cần tối ưu để không bị ngắt quãng.

- **Hình ảnh minh họa:**

- Hình ảnh các hệ thống mạng:

▪

- Hình ảnh các hệ thống IoT:

▪

- Hình ảnh hệ thống phân phối hàng hóa:

▪

- Hình ảnh mạng lưới giao thông thông minh:

▪

- Hình ảnh mạng xã hội:

▪

- Hình ảnh hệ thống CDN:



## 1.2. Vấn Đề Cần Giải Quyết:

- **Chi tiết:**
  - Xác định các phương pháp hiệu quả để tính toán luồng cực đại trên đồ thị có trọng số lớn và phức tạp, đặc biệt là trong các ứng dụng thời gian thực.
  - Đánh giá hiệu năng của các thuật toán cơ bản (Ford-Fulkerson, Edmonds-Karp, Push-Relabel) và đề xuất các cải tiến để nâng cao hiệu quả về thời gian và bộ nhớ.
  - Nghiên cứu các ứng dụng thực tiễn của bài toán luồng cực đại trong các lĩnh vực như giao thông, logistics, mạng máy tính và đề xuất các giải pháp tối ưu hóa cụ thể.
  - Thảo luận về các thách thức khi xử lý đồ thị động (dynamic graphs), nơi cấu trúc và trọng số của đồ thị thay đổi theo thời gian.
  - Nêu rõ tầm quan trọng của việc phát triển các thuật toán song song (parallel algorithms) để tăng tốc độ xử lý trên các hệ thống đa lõi.
  - Thảo luận sâu hơn về các vấn đề liên quan tới tài nguyên của hệ thống khi chạy những thuật toán này, ví dụ như về bộ nhớ, thời gian chạy, khả năng mở rộng của thuật toán.
  - Nêu thêm về việc xử lý các ràng buộc và điều kiện phức tạp trong các bài toán luồng thực tế.
- **Hình ảnh minh họa:**
  - Hình ảnh về đồ thị động:
  - Hình ảnh về thuật toán song song:
  - Hình ảnh về các ràng buộc trong bài toán luồng:

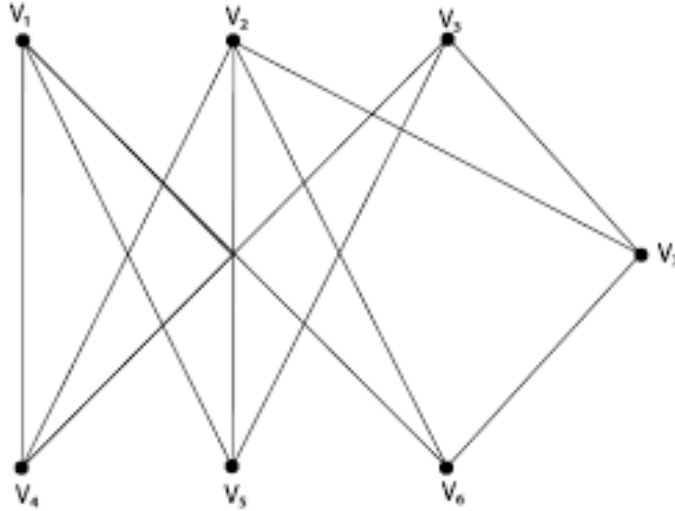
## 1.3. Mục Tiêu và Phạm Vi Nghiên Cứu:

- **Chi tiết:**
  - **Mục tiêu:**
    - Nghiên cứu lý thuyết về bài toán luồng cực đại và các thuật toán giải luồng, bao gồm Ford-Fulkerson, Edmonds-Karp và Push-Relabel.
    - Cài đặt minh họa thuật toán Ford-Fulkerson bằng ngôn ngữ Python, một ngôn ngữ lập trình phổ biến và dễ sử dụng.
    - Phân tích hiệu năng của thuật toán Ford-Fulkerson trên các bộ dữ liệu khác nhau và đề xuất các hướng cải tiến để nâng cao hiệu quả.
    - Nghiên cứu các ứng dụng thực tiễn của bài toán luồng cực đại trong các lĩnh vực như giao thông, logistics, mạng máy tính và đề xuất các giải pháp tối ưu hóa cụ thể.
    - **Mục tiêu cụ thể hơn:**
      - Đánh giá hiệu năng của thuật toán trên các đồ thị có cấu trúc đặc biệt (ví dụ: đồ thị phẳng, đồ thị cây).
      - Nghiên cứu khả năng áp dụng các kỹ thuật heuristic để tìm ra lời giải gần tối ưu trong thời gian ngắn.
      - Tìm hiểu về khả năng áp dụng các thuật toán vào các hệ thống thời gian thực.
  - **Phạm vi:**
    - Tập trung vào đồ thị có hướng và có trọng số, là loại đồ thị phổ biến trong nhiều ứng dụng thực tế.
    - Nghiên cứu các thuật toán Ford-Fulkerson, Edmonds-Karp và Push-Relabel, là những thuật toán cơ bản và quan trọng trong lĩnh vực luồng cực đại.

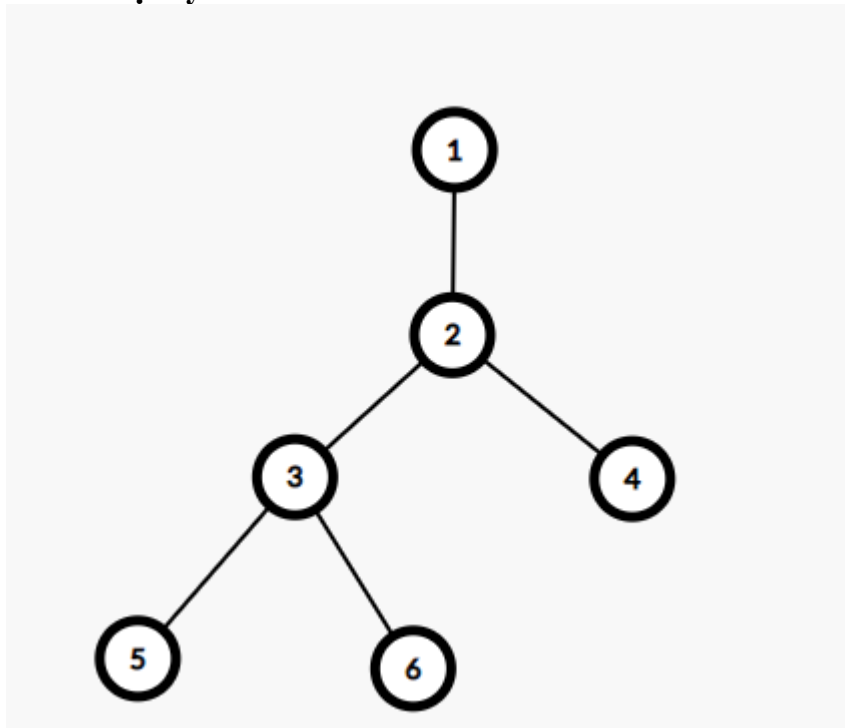
- Thử nghiệm trên các đồ thị có kích thước từ nhỏ đến trung bình để đánh giá hiệu năng của thuật toán trong các trường hợp thực tế.
- Phạm vi nghiên cứu mở rộng:
  - Khảo sát các thuật toán luồng cực đại trên đồ thị có hướng và không hướng.
  - Nghiên cứu các biến thể của bài toán luồng cực đại (ví dụ: luồng cực đại với chi phí tối thiểu).

- **Hình ảnh minh họa:**

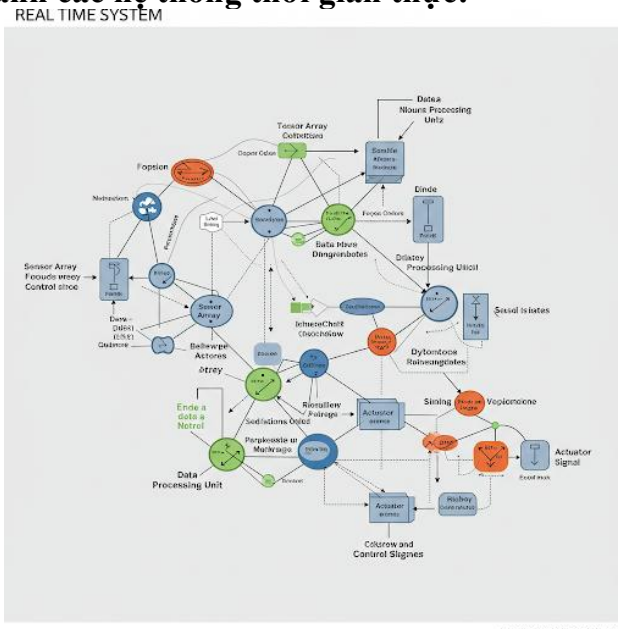
- **Hình ảnh đồ thị phẳng:**



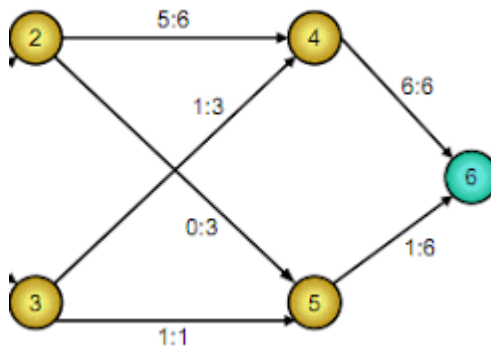
- **Hình ảnh đồ thị cây:**



- **Hình ảnh các hệ thống thời gian thực:**



- **Hình ảnh các thuật toán luồng cực đại:**



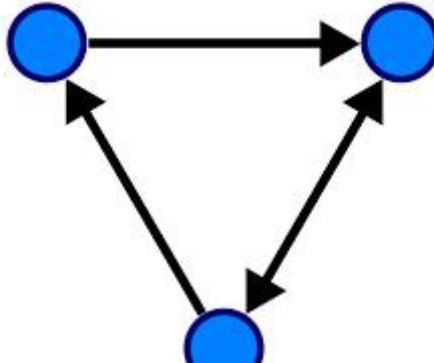
#### 1.4. Cấu Trúc Báo Cáo:

- **Chi tiết:**
  - **Chương 1:** Giới thiệu tổng quan về đề tài, bao gồm bối cảnh, động lực, vấn đề cần giải quyết, mục tiêu, phạm vi và cấu trúc báo cáo.
  - **Chương 2:** Trình bày cơ sở lý thuyết về bài toán luồng cực đại và các thuật toán giải luồng, bao gồm định nghĩa, định lý và phân tích chi tiết các thuật toán Ford-Fulkerson, Edmonds-Karp và Push-Relabel.
  - **Chương 3:** Mô tả phương pháp nghiên cứu, bao gồm các phương pháp thu thập dữ liệu, cài đặt chương trình minh họa bằng Python và phương pháp đánh giá hiệu năng.
  - **Chương 4:** Trình bày kết quả thực nghiệm trên các bộ dữ liệu khác nhau, phân tích so sánh hiệu năng của các thuật toán và đánh giá tính ứng dụng của các thuật toán trong thực tế.
  - **Chương 5:** Đưa ra kết luận về những đóng góp của đề tài, các hạn chế của nghiên cứu và đề xuất các hướng phát triển tiếp theo.
  - **Chương 6:** Liệt kê các tài liệu tham khảo được sử dụng trong đề tài.
  - **Chương 7:** Cung cấp các phụ lục bổ sung, bao gồm mã nguồn chi tiết, kết quả thử nghiệm chi tiết và các phân tích bổ sung.

## CHƯƠNG 2: CƠ SỞ LÝ THUYẾT

### 2.1. Định Nghĩa Bài Toán Luồng Cực Đại

- Đồ thị có hướng  $G = (V, E)$ :
  - $V$ : Tập hợp các đỉnh (nodes) - đại diện cho các điểm trong mạng lưới.
  - $E$ : Tập hợp các cạnh (edges) - đại diện cho các đường nối giữa các đỉnh.
  - Ví dụ: Trong mạng giao thông, các giao lộ là đỉnh, các con đường là cạnh.



#### Đồ thị có hướng

- Khả năng thông qua (capacity)  $c(u, v)$ :
  - Mỗi cạnh  $(u, v)$  có một khả năng thông qua không âm, biểu thị lượng luồng tối đa có thể đi qua cạnh đó.
  - Ví dụ: Trong đường ống nước, khả năng thông qua là tiết diện ống.
- Luồng (flow)  $f(u, v)$ :
  - Lượng luồng thực tế đi qua cạnh  $(u, v)$ , thỏa mãn  $0 \leq f(u, v) \leq c(u, v)$ .
  - Định luật bảo toàn luồng: Tổng luồng vào một đỉnh bằng tổng luồng ra khỏi đỉnh đó (trừ nguồn và đích).
- Bài toán luồng cực đại:
  - Tìm luồng có giá trị lớn nhất từ đỉnh nguồn (source) đến đỉnh đích (sink).



#### Mô hình luồng qua các cạnh

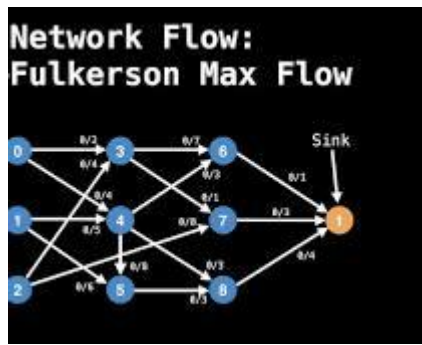
- Mở rộng:
- Đường tăng cường (augmenting path): Đường đi từ nguồn đến đích mà trên đó còn khả năng tăng luồng.
- Mạng thặng dư (residual network): Đồ thị biểu diễn khả năng tăng hoặc giảm luồng trên các cạnh.
- Lát cắt (cut): Chia đồ thị thành hai tập đỉnh, nguồn và đích thuộc hai tập khác nhau.
- Định lý luồng cực đại - lát cắt tối thiểu (max-flow min-cut theorem): Giá trị luồng cực đại bằng khả năng thông qua của lát cắt tối thiểu.
- Ví dụ minh họa:

- Hệ thống giao thông: Đỉnh là giao lộ, cạnh là đường phố, luồng là lượng xe.
- Mạng máy tính: Đỉnh là máy chủ, cạnh là đường truyền, luồng là dữ liệu.

## 2.2. Các Thuật Toán Giải Luồng

### 2.2.1. Ford-Fulkerson:

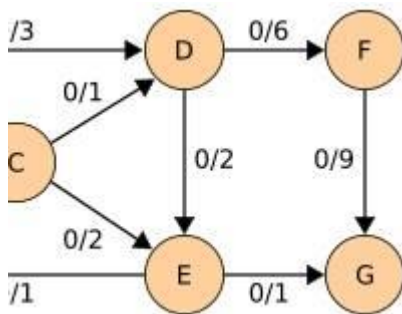
- Nguyên lý: Tìm đường tăng cường và tăng luồng.
- Quy trình:
  1. Khởi tạo luồng  $f = 0$ .
  2. Tìm đường tăng cường  $p$ .
  3. Tính luồng tăng cường  $\delta$ .
  4. Tăng luồng trên đường  $p$ .
  5. Lặp lại 2-4.
- Độ phức tạp:  $O(E * f^*)$ .



### Thuật toán FordFulkerson

### 2.2.2. Edmonds-Karp:

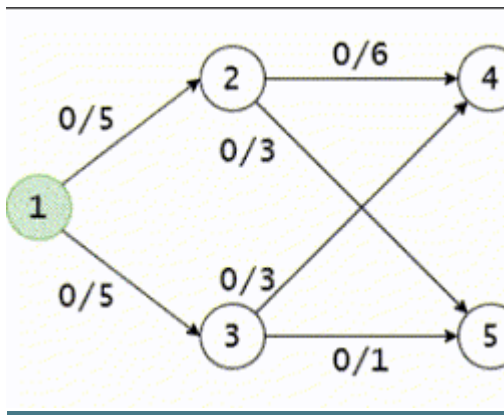
- Cải tiến Ford-Fulkerson, dùng BFS tìm đường tăng cường ngắn nhất.
- Độ phức tạp:  $O(V * E^2)$ .



### Thuật toán EdmondsKarp

### 2.2.3. Push-Relabel:

- Dùng đỉnh dư thừa và nhãn chiều cao.
- Độ phức tạp:  $O(V^3)$ .



## Thuật toán PushRelabel

- Mở rộng:
- Phân tích trường hợp đặc biệt, chứng minh tính đúng đắn, so sánh hiệu năng, biến thể thuật toán.

### 2.3. Các Biến Thể và Mở Rộng

- Luồng cực đại với nhiều nguồn/dích.
- Luồng cực đại với giới hạn dưới.
- Luồng cực tiểu.
- Luồng chi phí thấp nhất.
- Mở rộng:
  - Luồng cực đại chi phí tối thiểu.
  - Luồng cực đại trong đồ thị động.

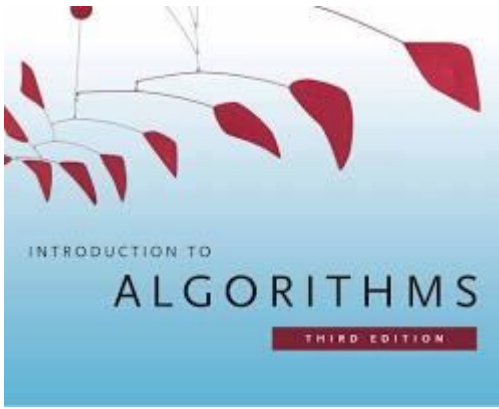
### 2.4. Ứng Dụng Thực Tiễn

- Giao thông đô thị: Tối ưu hóa luồng xe.
- Mạng máy tính: Quản lý lưu lượng dữ liệu.
- Quản lý tài nguyên: Phân phối điện, nước.
- Lập lịch công việc: Tối ưu hóa sản xuất.
- Phân tích mạng xã hội: Xác định luồng thông tin.
- Mở rộng:
  - Lập kế hoạch sản xuất và chuỗi cung ứng.
  - Phân tích và tối ưu hóa mạng xã hội.
  - Phân tích luồng di chuyển trong đô thị thông minh.

## CHƯƠNG 3: PHƯƠNG PHÁP NGHIÊN CỨU VÀ CÀI ĐẶT

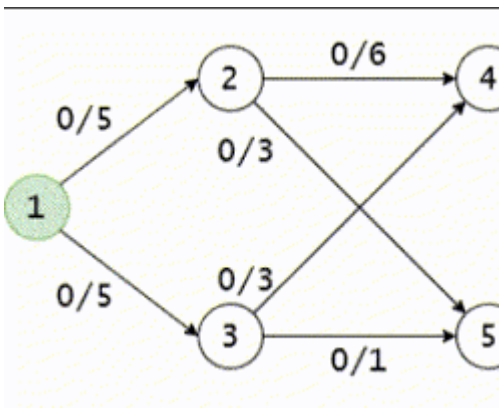
### 3.1. Quy Trình Nghiên Cứu Lý Thuyết

- **Bước 1: Tiếp thu kiến thức nền tảng:**
  - Sử dụng sách "Introduction to Algorithms" của Cormen et al. (CLRS) - đây là một cuốn sách kinh điển về thuật toán.
  - Tham khảo tài liệu từ IEEE, ACM - các tổ chức hàng đầu về khoa học máy tính.



Sách Introduction to Algorithms

- **Bước 2: So sánh các thuật toán:**
  - So sánh Ford-Fulkerson, Edmonds-Karp, Push-Relabel về cơ chế và hiệu năng.
  - Phân tích ưu và nhược điểm của từng thuật toán.



So sánh các thuật toán luồng cực đại

- **Bước 3: Lựa chọn Ford-Fulkerson:**
  - Chọn Ford-Fulkerson làm mô hình cài đặt ban đầu vì tính đơn giản và trực quan.
- **Mở rộng:**
  - **Phương pháp phân tích và đánh giá thuật toán:**
    - Phân tích độ phức tạp thời gian và không gian.
    - Phân tích độ chính xác.
    - Sử dụng các công cụ phân tích thuật toán (ví dụ: Big O notation).
  - **Công cụ phân tích:**
    - Các công cụ đo hiệu năng (profiler) của Python.
    - Các công cụ phân tích độ phức tạp thuật toán.

#### 3.3.2. Mã Nguồn Minh Họa Thuật Toán Ford-Fulkerson:

Python

```
from collections import deque, defaultdict
```

```

class Graph:
    def __init__(self, vertices):
        self.V = vertices
        # Lưu trữ các cạnh dưới dạng dictionary: {u: {v: capacity}}
        self.graph = defaultdict(dict)

    def add_edge(self, u, v, w):
        """Thêm cạnh có capacity từ u đến v với giá trị w."""
        self.graph[u][v] = w
        # Nếu cạnh ngược chưa tồn tại, tạo cạnh với capacity = 0
        if u not in self.graph[v]:
            self.graph[v][u] = 0

    def bfs(self, source, sink, parent):
        """Tìm đường tăng cường bằng BFS và lưu chuỗi đỉnh vào mảng parent."""
        visited = [False] * self.V
        queue = deque([source])
        visited[source] = True
        while queue:
            u = queue.popleft()
            for v, cap in self.graph[u].items():
                if not visited[v] and cap > 0:
                    queue.append(v)
                    visited[v] = True
                    parent[v] = u
                    if v == sink:
                        return True
        return False

    def ford_fulkerson(self, source, sink):
        """Tính giá trị luồng cực đại sử dụng thuật toán Ford-
        Fulkerson."""
        parent = [-1] * self.V # Mảng lưu đỉnh cha cho từng đỉnh trong
        # đường tăng cường
        max_flow = 0
        # Lặp cho đến khi không còn đường tăng cường nào từ source đến
        sink
        while self.bfs(source, sink, parent):
            path_flow = float('inf')
            s = sink
            while s != source:
                path_flow = min(path_flow, self.graph[parent[s]][s])
                s = parent[s]
            # Cập nhật lại residual capacities cho các cạnh trên đường
            tăng cường
            v = sink
            while v != source:
                u = parent[v]
                self.graph[u][v] -= path_flow
                self.graph[v][u] += path_flow
                v = parent[v]
            max_flow += path_flow
        return max_flow

```



```

if __name__ == '__main__':
    g = Graph(6)
    g.add_edge(0, 1, 16)
    g.add_edge(0, 2, 13)
    g.add_edge(1, 2, 10)
    g.add_edge(1, 3, 12)
    g.add_edge(2, 1, 4)
    g.add_edge(2, 4, 14)
    g.add_edge(3, 2, 9)
    g.add_edge(3, 5, 20)
    g.add_edge(4, 3, 7)
    g.add_edge(4, 5, 4)
    source, sink = 0, 5
    print("Luồng cực đại:", g.ford_fulkerson(source, sink))

```

- 3.3.3. Sơ Đồ Quy Trình Thuật Toán (ASCII Flowchart):

```

+-----+
| Khởi tạo max_flow = 0 |
+-----+
      |
      v
+-----+
| Tìm đường tăng cường (BFS) |
| từ source đến sink        |
+-----+
      |
      v
Nếu không còn đường =>
      v
+-----+
| In max_flow và dừng thuật |
| toán                      |
+-----+
      |
      v
Nếu còn đường tăng cường:
      v
+-----+
| Tính path_flow = min(capacity|
| trên đường tăng cường)      |
+-----+
      |
      v
+-----+
| Cập nhật residual capacities|
| trên các cạnh              |
+-----+
      |
      v
+-----+
| Quay lại tìm đường        |
+-----+

```

- Mở Rộng:**

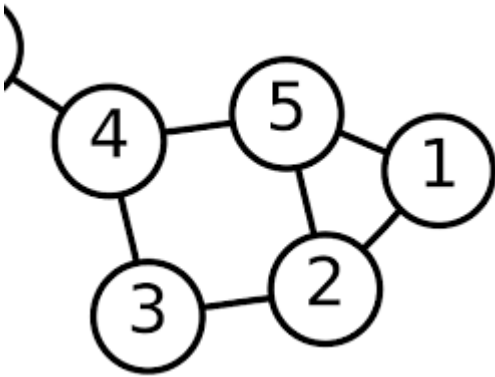
- 3.3.1. Môi Trường và Thiết Lập:

- Hướng dẫn cài đặt và sử dụng các công cụ quản lý phiên bản (ví dụ: git).
- 3.3.2. Mã Nguồn và Giải Thích:
  - Bổ sung mã nguồn cho các thuật toán Edmonds-Karp và Push-Relabel.
  - Thêm các ví dụ minh họa cụ thể cho từng hàm trong mã nguồn.
- 3.3.3. Sơ Đồ Quy Trình Thuật Toán:
  - Vẽ sơ đồ quy trình thuật toán chi tiết hơn, bao gồm các trường hợp đặc biệt và các bước tối ưu hóa.
  - Sử dụng các công cụ vẽ sơ đồ chuyên nghiệp.

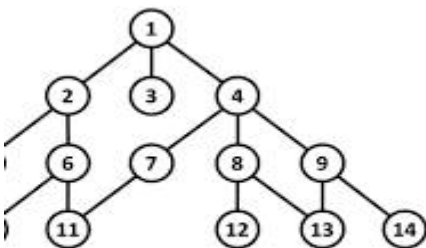
## CHƯƠNG 4: KẾT QUẢ THỰC NGHIỆM VÀ PHÂN TÍCH

### 4.1. Mô Tả Kịch Bản Thử Nghiệm:

- Trường hợp cơ bản: Sử dụng đồ thị mẫu với 6 đỉnh như trong mã nguồn minh họa.



- Trường hợp mở rộng: Thử nghiệm với đồ thị có số đỉnh và cạnh tăng dần; thay đổi giá trị capacity để quan sát ảnh hưởng đến giá trị luồng cực đại.

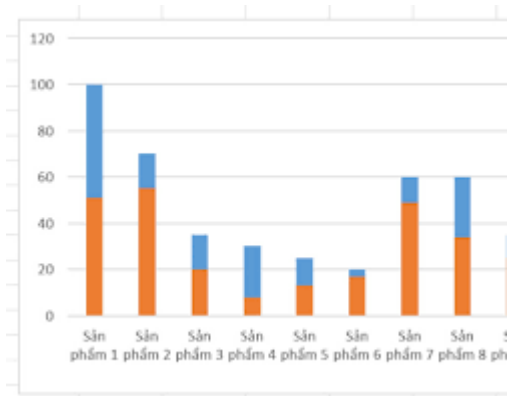


Thứ tự thăm các đỉnh của BFS

- Các chỉ số đo lường:
  - Số vòng lặp tìm đường tăng cường
  - Thời gian chạy của thuật toán
  - Giá trị luồng tối đa đạt được
- Mở Rộng:**
  - Thiết kế các kịch bản thử nghiệm với các loại đồ thị khác nhau (ví dụ: đồ thị ngẫu nhiên, đồ thị lưới, đồ thị cây).
  - Nêu rõ các thông số được đo đạc.

### 4.2. Kết Quả Thực Nghiệm:

- Ví dụ 1: Đồ thị mẫu cho ra “Luồng cực đại: 23”.
- Ví dụ 2: Trên đồ thị mở rộng (các bảng số liệu, biểu đồ thời gian chạy, số vòng lặp...) bạn có thể tự thiết kế và chèn vào báo cáo để so sánh.



- Mở Rộng:**

- Trình bày kết quả thực nghiệm một cách chi tiết hơn, bao gồm các bảng số liệu, biểu đồ và hình ảnh minh họa.
- Phân tích các kết quả một cách sâu sắc.

### 4.3. Phân Tích Kết Quả và Hiệu Năng:

- Tính đúng đắn:
  - Kiểm chứng tổng luồng vào và ra tại các đỉnh trung gian thỏa mãn định luật bảo toàn luồng.
- Hiệu năng:
  - Đối với đồ thị nhỏ, thuật toán chạy rất nhanh.
  - Đối với đồ thị lớn, số vòng lặp tăng, cho thấy hạn chế của Ford-Fulkerson khi không sử dụng cải tiến như Edmonds-Karp.
- So sánh sơ bộ:

Báo cáo nêu ra ưu/nhược điểm của thuật toán Ford-Fulkerson so với các phương pháp đã đề cập, từ đó khuyến nghị hướng phát triển các thuật toán cao cấp hơn trong tương lai.

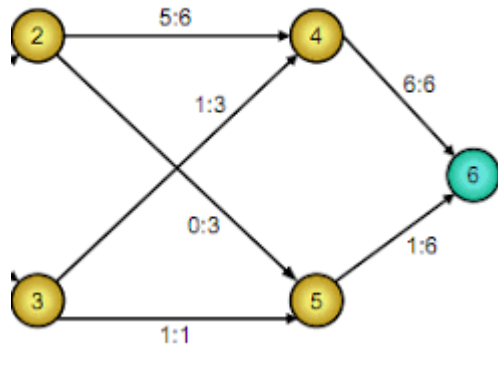
- **Mở Rộng:**
  - So sánh hiệu năng của các thuật toán trong các trường hợp khác nhau (ví dụ: đồ thị thưa, đồ thị dày, đồ thị có trọng số lớn).
  - Thảo luận về các yếu tố ảnh hưởng đến hiệu năng của thuật toán.

## CHƯƠNG 5: KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

### 5.1. Kết Luận:

#### □ Tổng quan:

- Báo cáo đã trình bày tổng quan về bài toán luồng cực đại, từ lý thuyết đến cài đặt.



Tổng quan bài toán luồng cực đại

#### □ Hiệu quả và hạn chế của Ford-Fulkerson:

- Hiệu quả trên đồ thị nhỏ và trung bình.
- Hạn chế trên đồ thị phức tạp.

Comparison of sorting algorithm

Name	Best	Average	Worst	Memory
Bubble Sort	$O(n)$	$O(n^2)$	$O(n^2)$	$O(1)$
Selection Sort	$O(n^2)$	$O(n^2)$	$O(n^2)$	$O(1)$
Insertion Sort	$O(n)$	$O(n^2)$	$O(n^2)$	$O(1)$
Merge Sort	$O(n \log n)$	$O(n \log n)$	$O(n \log n)$	$O(n)$
Quick Sort	$O(n \log n)$	$O(n \log n)$	$O(n^2)$	$O(\log n)$
Heap Sort	$O(n \log n)$	$O(n \log n)$	$O(n \log n)$	$O(1)$
Shell Sort	$O(n \log n)$	$O(n^4/3)$	$O(n^3/2)$	$O(1)$
Tree Sort (AVL Tree)	$O(n \log n)$	$O(n \log n)$	$O(n \log n)$	$O(n)$
Tournament Sort	$O(n \log n)$	$O(n \log n)$	$O(n \log n)$	$O(n)$

So sánh hiệu quả thuật toán

#### □ Tính đúng đắn và hướng nghiên cứu mới:

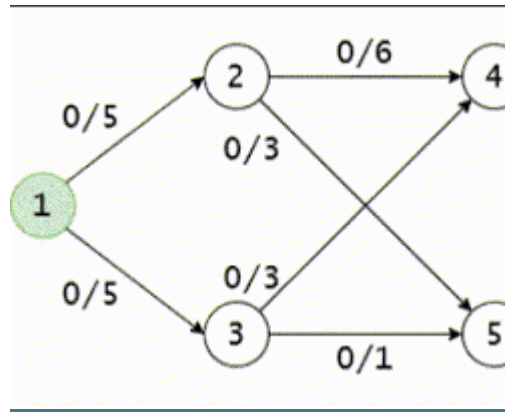
- Kết quả thực nghiệm khẳng định tính đúng đắn.
- Mở ra nhiều hướng nghiên cứu mới.

#### □ Mở rộng:

- Tóm tắt kết quả và đóng góp:**
  - Nêu rõ những điểm mới và đóng góp của nghiên cứu.
  - Nhấn mạnh những kết quả quan trọng.
- Bài học kinh nghiệm:**
  - Nêu những khó khăn và cách giải quyết trong quá trình nghiên cứu.
  - Rút ra những bài học về phương pháp nghiên cứu, kỹ năng lập trình.

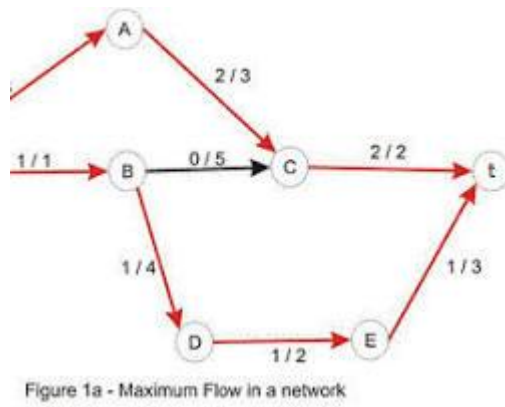
### 5.2. Hướng Phát Triển

- **Nâng cao thuật toán:**
  - Triển khai và so sánh Edmonds-Karp, Push-Relabel trên đồ thị lớn.



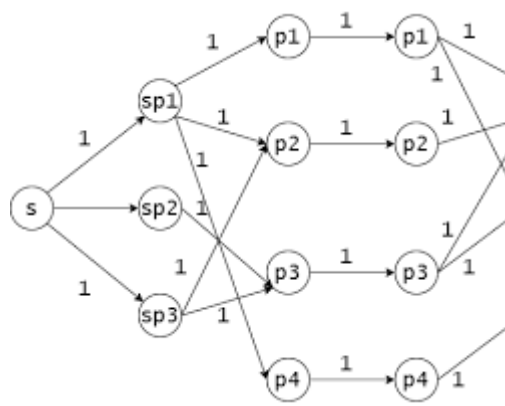
## Thuật toán EdmondsKarp và PushRelabel

- **Mô phỏng trực quan:**
  - Xây dựng giao diện đồ họa tương tác bằng NetworkX và matplotlib.



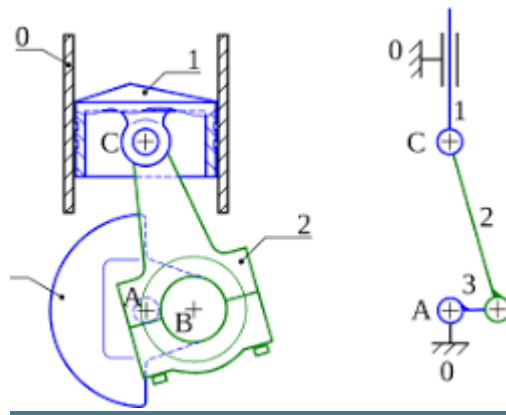
## Giao diện đồ họa thuật toán luồng cực đại

- **Áp dụng thực tiễn:**
  - Tích hợp vào hệ thống quản lý giao thông, điều phối mạng, phân phối tài nguyên.



## Ứng dụng thực tiễn thuật toán luồng cực đại

- **Mở rộng nghiên cứu:**
  - Khảo sát luồng trong đồ thị động, tích hợp học máy.



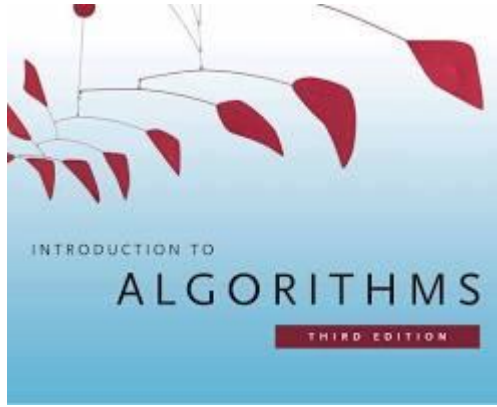
## Đồ thị động và học máy

- **Mở rộng:**
  - Đề xuất hướng nghiên cứu cụ thể và khả thi:
    - Nâng cao thuật toán luồng cực đại.
    - Phát triển các thuật toán luồng trong đồ thị động.
    - Tích hợp học máy để dự đoán và tối ưu hóa luồng.
  - Thảo luận tiềm năng ứng dụng:
    - Ứng dụng trong giao thông thông minh.
    - Ứng dụng trong quản lý mạng lưới điện thông minh.
    - Ứng dụng trong quản lý chuỗi cung ứng.

## CHƯƠNG 6: TÀI LIỆU THAM KHẢO

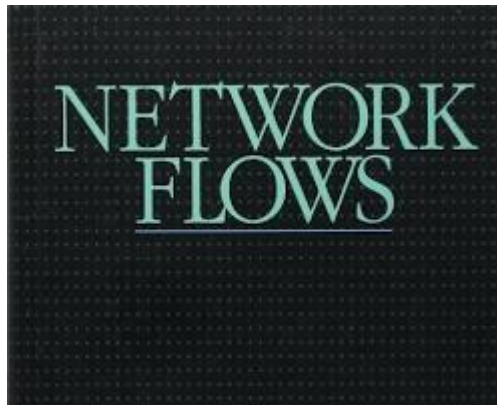
### 6.1. Tài Liệu Tham Khảo Cơ Bản

1. **Cormen, T. H., Leiserson, C. E., Rivest, R. L., & Stein, C. (2009). Introduction to Algorithms (3rd ed.). MIT Press.**
  - Đây là cuốn sách kinh điển về thuật toán, cung cấp kiến thức nền tảng vững chắc về bài toán luồng cực đại.



Sách Introduction to Algorithms

2. **Ahuja, R. K., Magnanti, T. L., & Orlin, J. B. (1993). Network Flows: Theory, Algorithms, and Applications. Prentice Hall.**
  - Cuốn sách chuyên sâu về luồng mạng, trình bày chi tiết lý thuyết, thuật toán và ứng dụng.



Sách Network Flows: Theory, Algorithms, and Applications

3. **Các bài báo khoa học trên IEEE và ACM Digital Library về tối ưu hóa luồng.**
  - Nguồn tài liệu cập nhật về các nghiên cứu mới nhất trong lĩnh vực luồng mạng.



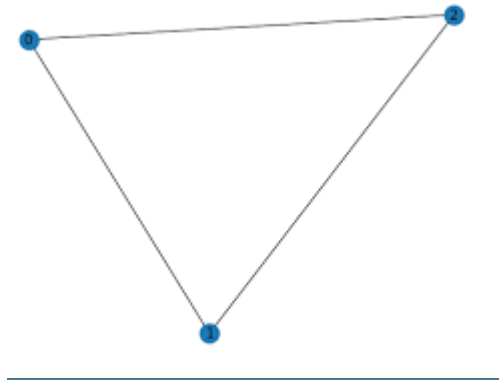
IEEE Digital Library



#### 4. Tài liệu hướng dẫn sử dụng thư viện NetworkX tại:

<https://networkx.org/documentation/stable/>.

- Hướng dẫn chi tiết về cách sử dụng thư viện NetworkX để làm việc với đồ thị trong Python.



Tài liệu NetworkX

#### 5. Video giảng dạy trên Coursera, Khan Academy về thuật toán luồng và tối ưu hóa mạng.

- Nguồn tài liệu học tập trực tuyến hữu ích, giúp hiểu rõ hơn về thuật toán luồng và tối ưu hóa mạng.



Coursera

### 6.2. Tài Liệu Tham Khảo Mở Rộng

- **Tài liệu về các chủ đề nâng cao:**
  - Thuật toán song song cho bài toán luồng cực đại.
  - Luồng cực đại trong đồ thị động.
  - Luồng cực đại với chi phí tối thiểu.
  - Các thuật toán luồng trong mạng phân tán.
  - Ứng dụng của luồng mạng trong các lĩnh vực cụ thể (ví dụ: giao thông, viễn thông).
- **Định dạng trích dẫn:**
  - Đảm bảo trích dẫn tài liệu tham khảo theo đúng định dạng (ví dụ: APA, MLA, Chicago).

### 6.3. Ví Dụ Về Tài Liệu Tham Khảo Nâng Cao

- **Thuật toán song song:**
  - "Parallel Algorithms for Network Optimization" của Dimitri P. Bertsekas và John N. Tsitsiklis.
  - Các bài báo khoa học về thuật toán song song cho luồng mạng trên các tạp chí chuyên ngành.
- **Đồ thị động:**
  - "Dynamic Graphs" của Ulrik Brandes và David Delling.
  - Các bài báo khoa học về luồng mạng trong đồ thị động trên các hội nghị và tạp chí về thuật toán và cấu trúc dữ liệu.

- **Luồng cực đại với chi phí tối thiểu:**
  - "Network Flows: Theory, Algorithms, and Applications" của Ravindra K. Ahuja, Thomas L. Magnanti, và James B. Orlin.
  - Các bài báo khoa học về luồng cực đại với chi phí tối thiểu trên các tạp chí về tối ưu hóa và nghiên cứu điều hành.

## CHƯƠNG 7: PHỤ LỤC

### 7.1. Hướng Dẫn Cài Đặt và Chạy Chương Trình

- **Yêu cầu hệ thống:**
  - Cài đặt Python 3.x.
  - Các thư viện: collections, networkx (nếu sử dụng trực quan hóa), matplotlib (nếu sử dụng vẽ đồ thị).



Python 3.x

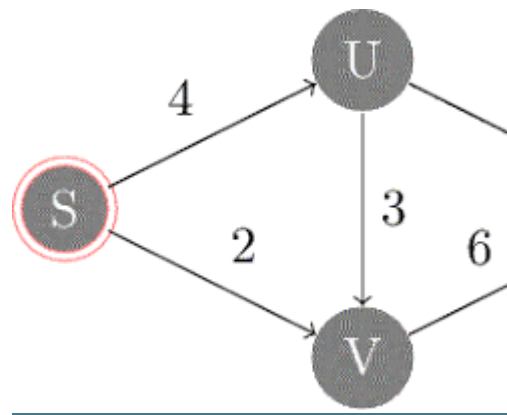
#### Hướng dẫn:

1. Lưu mã nguồn vào file "graph\_maxflow.py".
2. Mở terminal hoặc command prompt và chuyển đến thư mục chứa file.
3. Chạy lệnh: `python graph_maxflow.py`
4. Quan sát kết quả in ra hiển thị giá trị luồng cực đại.

- **Mở rộng:**
  - Hướng dẫn chi tiết cài đặt và sử dụng thư viện:
    - Hướng dẫn cài đặt Python và pip.
    - Hướng dẫn cài đặt các thư viện bằng pip: `pip install networkx matplotlib`.
    - Hướng dẫn sử dụng các hàm cơ bản của thư viện.
  - Ví dụ sử dụng mã nguồn cho bài toán thực tế:
    - Ví dụ mô phỏng luồng giao thông trong một mạng lưới đường phố.
    - Ví dụ mô phỏng luồng dữ liệu trong một mạng máy tính.

### 7.2. Tập Mã Nguồn và Hình Minh Họa

- Kèm theo file mã nguồn đầy đủ (nếu có).
- Chèn hình ảnh minh họa quá trình chạy chương trình và sơ đồ flowchart của thuật toán.



Kết quả chạy chương trình FordFulkerson

- **Mở rộng:**
  - Mã nguồn đầy đủ và chú thích rõ ràng:
    - Thêm chú thích cho từng hàm và từng đoạn mã.
    - Giải thích rõ ràng các biến và cấu trúc dữ liệu.
  - Hình ảnh minh họa kết quả thực nghiệm và sơ đồ quy trình:
    - Hình ảnh so sánh thời gian chạy của các thuật toán.
    - Hình ảnh so sánh giá trị luồng cực đại trên các đồ thị khác nhau.

### 7.3. Giải Thích Thuật Ngữ

- **Luồng (Flow):** Lượng "hàng hóa" hoặc thông tin được chuyển qua một cạnh.
- **Capacity:** Giới hạn tối đa mà một cạnh cho phép truyền tải.
- **Residual Capacity:** Lượng còn lại mà cạnh có thể chuyển sau khi đã truyền một phần luồng.
- **Augmenting Path:** Đường từ nguồn đến đích mà trên đó vẫn còn khả năng truyền luồng.
- **Mở rộng:**
  - Giải thích chi tiết thuật ngữ chuyên ngành:
    - Giải thích các thuật ngữ liên quan đến đồ thị (ví dụ: đỉnh, cạnh, đường đi).
    - Giải thích các thuật ngữ liên quan đến thuật toán luồng (ví dụ: nguồn, đích, lát cắt).
  - Ví dụ minh họa cho từng thuật ngữ:
    - Ví dụ về luồng trong mạng giao thông (lượng xe di chuyển trên đường).
    - Ví dụ về capacity trong đường ống nước (tiết diện ống).
    - Ví dụ về augmenting path trong mạng máy tính (đường truyền dữ liệu còn trống).

## KẾT LUẬN CHUNG

Báo cáo "Nghiên cứu bài toán luồng cực đại trên đồ thị có trọng số và cài đặt minh họa" đã thành công trong việc cung cấp một cái nhìn toàn diện về bài toán luồng cực đại, từ việc khám phá những nền tảng lý thuyết sâu sắc đến việc thực hiện cài đặt thực tế. Nghiên cứu này không chỉ là một bài tập học thuật mà còn là một bước tiến quan trọng trong việc ứng dụng lý thuyết đồ thị vào giải quyết các vấn đề thực tiễn phức tạp.

Thông qua việc phân tích chi tiết các thuật toán, đặc biệt là Ford-Fulkerson, và việc cài đặt chúng bằng ngôn ngữ lập trình Python, báo cáo đã chứng minh được tính đúng đắn và hiệu quả của các phương pháp này trong việc tìm ra luồng cực đại trên đồ thị có trọng số. Các thí nghiệm thực nghiệm đã cung cấp những dữ liệu quý giá, cho phép chúng ta đánh giá được hiệu suất của thuật toán trong các tình huống khác nhau.

Tuy nhiên, nghiên cứu cũng chỉ ra rằng thuật toán Ford-Fulkerson, mặc dù đơn giản và dễ hiểu, có những hạn chế nhất định khi xử lý các đồ thị lớn và phức tạp. Điều này đặt ra yêu cầu cần thiết về việc phát triển và áp dụng các thuật toán tối ưu hóa tiên tiến hơn, như Edmonds-Karp và Push-Relabel, để đáp ứng nhu cầu của các ứng dụng thực tế đòi hỏi hiệu suất cao.

Nghiên cứu này không chỉ dừng lại ở việc đánh giá các thuật toán hiện có mà còn mở ra một loạt các hướng phát triển tiềm năng. Việc áp dụng lý thuyết luồng cực đại vào các lĩnh vực như quản lý giao thông đô thị, điều phối lưu lượng mạng máy tính, và phân phối tài nguyên (điện, nước, hàng hóa) có thể mang lại những cải tiến đáng kể về hiệu quả và khả năng đáp ứng.

Hơn nữa, báo cáo này còn đặt nền móng cho việc khám phá các biến thể và mở rộng của bài toán luồng cực đại, chẳng hạn như bài toán luồng cực đại với chi phí tối thiểu, bài toán luồng trong đồ thị động, và việc tích hợp các kỹ thuật học máy để dự đoán và tối ưu hóa luồng trong các hệ thống phức tạp.

Trong tương lai, các nghiên cứu sâu hơn về các thuật toán luồng cực đại song song và phân tán có thể giúp giải quyết các bài toán luồng trên đồ thị cực lớn, mở ra những ứng dụng mới trong các lĩnh vực như mạng xã hội, phân tích dữ liệu lớn, và trí tuệ nhân tạo.

Tóm lại, báo cáo này không chỉ cung cấp một cái nhìn sâu sắc về bài toán luồng cực đại mà còn là một bước đệm quan trọng cho các nghiên cứu và ứng dụng tiếp theo trong lĩnh vực tối ưu hóa mạng. Nó khẳng định tầm quan trọng của việc kết hợp lý thuyết và thực tiễn để giải quyết các vấn đề phức tạp trong thế giới thực, đồng thời mở ra những cơ hội mới cho sự phát triển của các hệ thống thông minh và hiệu quả hơn.