

**BỘ GIÁO DỤC
VÀ ĐÀO TẠO**

**VIỆN HÀN LÂM KHOA HỌC
VÀ CÔNG NGHỆ VIỆT NAM**

HỌC VIỆN KHOA HỌC VÀ CÔNG NGHỆ



Nguyễn Thị Huệ

**NGHIÊN CỨU THUẬT TOÁN XẤP XỈ CHO
MỘT SỐ BÀI TOÁN TỐI ƯU SUBMODULAR
VỚI RÀNG BUỘC CHI PHÍ**

TÓM TẮT LUẬN ÁN TIẾN SĨ MÁY TÍNH

Ngành: Khoa học máy tính

Mã số: 9 48 01 01

Hà Nội - 2026

Công trình được hoàn thành tại: Học viện Khoa học và Công nghệ, Viện Hàn lâm Khoa học và Công nghệ Việt Nam

Người hướng dẫn khoa học:

1. Người hướng dẫn 1: PGS. TS. Phạm Văn Cảnh
2. Người hướng dẫn 2: PGS. TS. Nguyễn Long Giang

Phản biện 1:

Phản biện 2:

Luận án được bảo vệ trước Hội đồng đánh giá luận án tiến sĩ cấp Học viện họp tại Học viện Khoa học và Công nghệ, Viện Hàn lâm Khoa học và Công nghệ Việt Nam vào hồi giờ, ngày tháng năm

Có thể tìm hiểu luận án tại:

1. Thư viện Học viện Khoa học và Công nghệ
2. Thư viện Quốc gia Việt Nam

MỞ ĐẦU

1. Tính cấp thiết

Các bài toán tối ưu tổ hợp (Combinatorial Optimization – CO) hình thành từ nhu cầu lựa chọn phương án tối ưu trong các hoạt động kinh tế, kỹ thuật và đời sống xã hội. Trong kỷ nguyên trí tuệ nhân tạo, CO đóng vai trò trung tâm trong nhiều ứng dụng khoa học dữ liệu như tối đa hóa ảnh hưởng, tóm tắt thông tin, hệ thống khuyến nghị và tối đa hóa doanh thu. Nhiều bài toán CO có thể được mô hình hóa trên tập hợp, trong đó hàm mục tiêu hoặc ràng buộc mang tính *submodular*. Tính chất này, còn gọi là *lợi nhuận biên giảm dần*, cho phép xây dựng các thuật toán hiệu quả với bảo đảm lý thuyết về chất lượng nghiệm. Lớp bài toán này được gọi chung là tối ưu submodular (Submodular Optimization – SO), đặc biệt phù hợp với các ứng dụng dữ liệu quy mô lớn. Dưới dạng tổng quát nhất, bài toán SO được phát biểu như sau: Cho tập nền hữu hạn $V = \{e_1, e_2, \dots, e_n\}$, với hàm mục tiêu $f : 2^V \mapsto \mathbb{R}_+^1$ có tính chất submodular, nghĩa là với mọi $A \subseteq B \subseteq V$ và mọi phần tử $u \in V$, ta có $f(A \cup \{u\}) - f(A) \geq f(B \cup \{u\}) - f(B)$. Bài toán SO yêu cầu tìm tập con $S \subseteq V$ thỏa mãn ràng buộc $\mathcal{C}$ cho trước sao cho giá trị của f là cực đại hoặc cực tiểu.

Bài toán SO thuộc lớp bài toán NP-khó. Dưới góc độ học thuật, vì lớp bài toán SO tương đối rộng, các nhà nghiên cứu chia bài toán này thành hai dạng cụ thể như sau (Xem hình 0.1).

Bài toán 1: Bài toán phủ submodular với chi phí tối thiểu (MCSC). Mỗi phần tử $e \in V$ có một chi phí $c(e) > 0$. Với hàm mục tiêu là tổng chi phí $f(S) = \sum_{e \in S} c(e)$, cho trước một ngưỡng $T > 0$ và một hàm đánh giá $g : 2^V \rightarrow \mathbb{R}$, có tính chất submodular, tức là $g(A \cup \{e\}) - g(A) \geq g(B \cup \{e\}) - g(B)$, bài toán yêu cầu tìm

$$\arg \min_{S \subseteq V} f(S),$$

thỏa mãn: $g(S) \geq T$.

Bài toán 2: Bài toán tối đa hóa hàm submodular với ràng buộc (SM). Với ràng buộc $\mathcal{C}$ cho trước, bài toán yêu cầu tìm

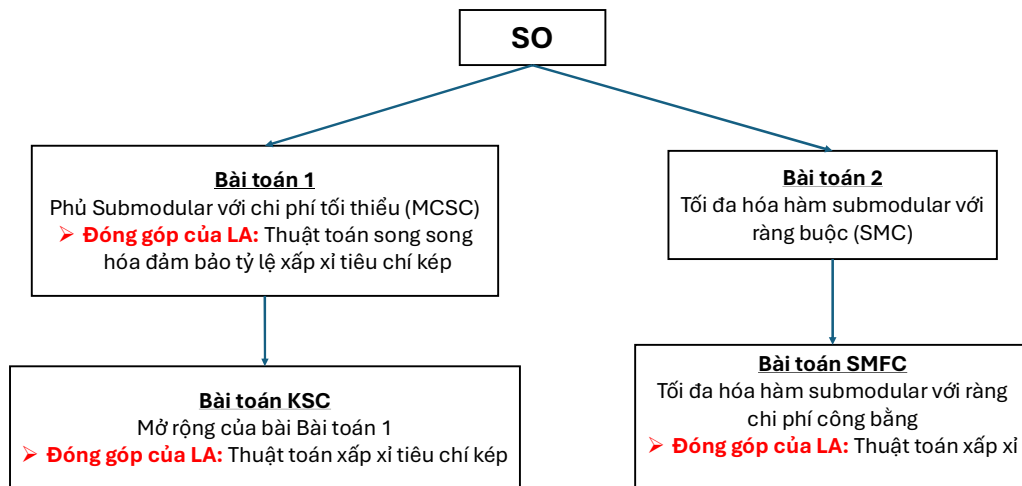
$$\max_{S \subseteq V} f(S)$$

thỏa mãn: $S \in \mathcal{C}$.

Dạng bài toán này bao trùm nhiều bài toán điển hình trong phân tích dữ liệu và học máy, chẳng hạn như tối đa hóa ảnh hưởng trong mạng xã hội, lựa chọn đặc trưng trong học

¹ 2^V : Tập tất cả các tập con của V bao gồm tập rỗng.

máy, tối đa hóa doanh thu trong mạng xã hội, bài toán đặt cảm biến, và lựa chọn tập đại diện nhằm tối đa hóa lượng thông tin thu được dưới các ràng buộc tài nguyên, v.v. Các ràng buộc trong $\mathcal{C}$ thường gặp trong các bài toán tối ưu tổ hợp bao gồm: ràng buộc kích thước (cardinality constraint), ràng buộc ba lô (knapsack constraint), ràng buộc matroid (matroid constraint), v.v. Tuy đã đạt được nhiều kết quả quan trọng, lớp các bài toán SO



Hình 0.1: Các dạng của bài toán SO và đóng góp của Luận án.

vẫn đối mặt với nhiều thách thức cả về phương diện mô hình hóa lẫn thiết kế thuật toán, như sau:

Thứ nhất, sự gia tăng nhanh chóng của quy mô dữ liệu trong các hệ thống hiện đại đặt ra yêu cầu cấp thiết đối với các thuật toán vừa đảm bảo chất lượng lời giải, vừa có khả năng xử lý trong thời gian phù hợp với yêu cầu thực tế.

Thứ hai, trong thực tế vẫn tồn tại nhiều biến thể của bài toán mang tính ứng dụng cao nhưng chưa có thuật toán hiệu quả Các bài toán thực tế thường đi kèm với nhiều loại ràng buộc phức tạp như yêu cầu công bằng giữa các nhóm đối tượng, giới hạn tài nguyên đa chiều, hoặc các cấu trúc phụ thuộc giữa các phần tử, khiến cho các mô hình chuẩn hiện nay chưa thể phản ánh đầy đủ yêu cầu của ứng dụng. Do đó, việc mở rộng mô hình và xây dựng các thuật toán hiệu quả cho các biến thể mới vẫn là một hướng nghiên cứu quan trọng và còn nhiều tiềm năng phát triển.

2. Mục tiêu của luận án

Luận án tập trung nghiên cứu giải quyết những thách thức trên với các mục tiêu sau: 1. Trong khuôn khổ luận án, luận án nghiên cứu bài toán Bài toán 1 (MCSC) và Bài toán 2 (SM) với các ràng buộc $\mathcal{C}$ có tính thời sự.

2. Đề xuất và phân tích các thuật toán xấp xỉ hiệu quả trong bối cảnh nơi mà việc tìm nghiệm tối ưu chính xác thường không khả thi do độ phức tạp tính toán cao.

3. Kết quả đạt được

Với mục đích trên, luận án đã đạt được các kết quả sau:

Luận án nghiên cứu Bài toán 1 (MCSC) và đề xuất một thuật toán song song hiệu quả đạt được tỷ lệ tiệm cận với thuật toán xấp xỉ tốt nhất hiện nay nhưng giảm rất đáng kể độ phức tạp song song về cả phương diện lý thuyết và thực nghiệm.

Luận án nghiên cứu bài toán Phủ k -Submodular với chi phí tối thiểu (kSC), là một mở rộng của bài toán MCSC. Luận án đề xuất các thuật toán xấp xỉ tiêu chí kép hiệu quả hơn các phương pháp hiện có về cả lý thuyết và thực nghiệm.

Đối với Bài toán 2, luận án nghiên cứu bài toán *Tối đa hóa hàm submodular với ràng buộc chi phí công bằng* (MCFA) và đề xuất các thuật toán xấp xỉ hiệu quả cho bài toán này.

4. Bố cục của luận án

Ngoài phần mở đầu và kết luận, luận án chia thành bốn chương như sau

Chương 1 trình bày tổng quan về các bài toán tối ưu tổ hợp và vai trò của chúng trong các ứng dụng phân tích dữ liệu hiện đại.

Chương 2 trình bày kết quả nghiên cứu bài toán Phủ submodular với chi phí tối thiểu (MCSC).

Chương 3 trình bày kết quả nghiên cứu bài toán Phủ k -Submodular với chi phí tối thiểu (kSC).

Chương 4 trình bày kết quả nghiên cứu bài toán *Tối đa hàm Submodular với ràng buộc chi phí công bằng* (SMFC).

CHƯƠNG 1

TỔNG QUAN VỀ CÁC BÀI TOÁN TỐI ƯU SUBMODULAR

1.1. BÀI TOÁN TỐI ƯU SUBMODULAR

Dưới góc độ mô hình hóa toán học, bài toán SO thường được định nghĩa dựa trên các khái niệm sau:

- *Tập nền* $V = \{e_1, e_2, \dots, e_n\}$: tập hữu hạn gồm n phần tử hữu hạn, trong đó mỗi phần tử e_i đại diện cho một đối tượng ứng viên cần được lựa chọn.
- *Ràng buộc* $\mathcal{C}$: Nhằm mô hình hóa các yêu cầu và hạn chế trong thực tế.
- *Họ các tập con* 2^V : Tập hợp tất cả các tập con có thể của tập V .
- Không gian lời giải hợp lệ: $\Omega = \{S \subseteq V : S \text{ thỏa mãn } \mathcal{C}\}$.
- *Hàm mục tiêu* f , *đánh giá* g : Trên không gian này, ta xác định hàm mục tiêu, ràng buộc nhận giá trị thực không âm: $f, g : 2^V \rightarrow \mathbb{R}_+$, trong đó f đo lường chất lượng hoặc lợi ích của mỗi tập con và mức thỏa mãn ràng buộc $\mathcal{C}$ của lời giải. Hàm $f : 2^V \rightarrow \mathbb{R}$ thỏa mãn tính chất *submodular*, tức là với mọi cặp tập con $A \subseteq B \subseteq V$ và với mọi phần tử $e \in V \setminus B$, điều kiện sau được thỏa mãn:

$$f(A \cup \{e\}) - f(A) \geq f(B \cup \{e\}) - f(B).$$

Ngoài ra, một hàm submodular f được gọi là *đơn điệu* nếu $f(A) \leq f(B), \forall A \subseteq B \subseteq V$.

Một số khái niệm quan trọng:

- *Lời giải hợp lệ (feasible solution)*: Một tập con $S \subseteq \Omega$ được gọi là một lời giải chấp nhận được.
- *Lời giải tối ưu (optimal solution)*: Một lời giải chấp nhận được $O \in \Omega$ được gọi là lời giải tối ưu nếu $O \in \arg \max_{S \in \Omega} f(S)$ (hoặc $O \in \arg \min_{S \in \Omega} f(S)$).

1.1.1. Một số bài toán cụ thể

Dưới đây là một số bài toán tiêu biểu thuộc lớp các bài toán SO thường gặp trong thực tế.

Định nghĩa 1.1 (Bài toán tối đa ảnh hưởng (Influence maximization–IM)). Cho một đồ thị có hướng $G = (V, E)$ mô hình hóa một mạng xã hội và một mô hình lan truyền thông tin cố định $\mathcal{M}$. Bài toán tối đa ảnh hưởng đặt mục tiêu xác định một tập nút khởi tạo (hay còn gọi là tập hạt giống) $S \subseteq V$ sao cho mức độ ảnh hưởng lan truyền từ S lên toàn bộ mạng, được đo lường thông qua hàm ảnh hưởng $f(S)$, đạt giá trị lớn nhất.

Định nghĩa 1.2 (Bài toán Ảnh hưởng ngưỡng [1]). Cho một mạng xã hội $G = (V, E)$, một mô hình lan truyền thông tin $\mathcal{M}$, và một ngưỡng T . Mục tiêu là tìm một tập hạt

giống $S \subseteq V$ sao cho $f(S) \geq T$. với lượng ngân sách là tối thiểu, chẳng hạn như số lượng tập *hạt giống* $|S|$ hoặc tổng chi phí bỏ ra $\sum_{v \in S} c(v)$.

1.1.2. Độ phức tạp của thuật toán

Định nghĩa 1.3 (Độ phức tạp thời gian của thuật toán). Xét một thuật toán $\mathcal{A}$ giải một bài toán với dữ liệu đầu vào có kích thước n . Độ phức tạp thời gian của thuật toán $\mathcal{A}$, ký hiệu $T_{\mathcal{A}}(n)$, được định nghĩa là số bước tính toán cơ bản nhiều nhất mà thuật toán cần thực hiện để xử lý mọi dữ liệu đầu vào có kích thước không vượt quá n .

Định nghĩa 1.4 (Độ phức tạp truy vấn (Query complexity)). Xét một thuật toán $\mathcal{A}$ giải một bài toán với dữ liệu đầu vào có kích thước n . Giả sử có thể truy vấn tới hàm mục tiêu thông qua một phương pháp cho trước (gọi là black-box). Độ phức tạp truy vấn của thuật toán $\mathcal{A}$, ký hiệu $Q_{\mathcal{A}}(n)$, được định nghĩa là truy vấn đến hàm f thông qua phương pháp đó (hay số lần gọi hàm mục tiêu f) nhiều nhất mà thuật toán cần thực hiện đối với mọi dữ liệu đầu vào có kích thước không vượt quá n .

Định nghĩa 1.5 (Độ phức tạp song song). Với một hàm mục tiêu f , độ phức tạp song song của một thuật toán là số vòng lặp tối thiểu cần thiết sao cho trong mỗi vòng lặp, thuật toán thực hiện $O(\text{poly}(n))$ truy vấn độc lập đến hàm mục tiêu, trong đó n là kích thước dữ liệu đầu vào của thuật toán.

1.1.3. Định nghĩa thuật toán xấp xỉ

Định nghĩa 1.6 (Thuật toán xấp xỉ [2]). Cho bài toán tối ưu $\mathcal{P}$ với hàm mục tiêu f , với kích thước dữ liệu đầu vào là n , trong đó mục tiêu có thể là tìm giá trị cực đại hoặc cực tiểu. Một thuật toán $\mathcal{A}$ được gọi là thuật toán xấp xỉ cho bài toán $\mathcal{P}$ với hệ số xấp xỉ $\rho(n)$ nếu: (i) Thuật toán $\mathcal{A}$ chạy trong thời gian đa thức theo kích thước đầu vào n ; (ii) Luôn trả về lời giải hợp lệ cho bài toán, gọi lời giải là S . Gọi S^* lời giải tối ưu, ta có:

$$\begin{aligned} f(S) &\geq \rho(n) \cdot f(S^*), (0 < \rho \leq 1) \quad \text{Đối với bài toán tìm cực đại} \\ f(S) &\leq \rho(n) \cdot f(S^*), (\rho \geq 1) \quad \text{Đối với bài toán tìm cực tiểu} \end{aligned}$$

Nói cách khác, nghiệm của thuật toán không lệch khỏi nghiệm tối ưu quá một hệ số $\rho(n)$.

Định nghĩa 1.7 (Thuật toán xấp xỉ tiêu chuẩn kép [3]). Với bài toán MCSC, một thuật toán được gọi là *thuật toán xấp xỉ tiêu chuẩn kép*, hay thuật toán (α, β) -xấp xỉ, nếu trong thời gian đa thức theo kích thước dữ liệu đầu vào n , nó tìm được một nghiệm S sao cho giá trị của hàm ràng buộc chỉ vi phạm nhiều nhất một hệ số α , và đạt được tỷ lệ xấp xỉ β , tức là: $g(S) \leq \alpha C; f(S) \geq \beta f(S^*)$ với S^* là nghiệm tối ưu và $C = c(S^*)$.

CHƯƠNG 2

BÀI TOÁN PHỦ SUBMODULAR VỚI CHI PHÍ TỐI THIỂU

2.1. ĐỊNH NGHĨA BÀI TOÁN VÀ ĐÓNG GÓP CỦA LUẬN ÁN

Xét một tập nền hữu hạn $V = \{e_1, \dots, e_n\}$ và một hàm tiện ích $g : 2^V \mapsto \mathbb{R}^+$ dùng để đo lường chất lượng của một tập con $S \subseteq V$ thỏa mãn g là *submodular* và đơn điệu (tăng), tức là $g(A) \geq g(B)$. Bên cạnh đó, cho một hàm chi phí $c : 2^V \mapsto \mathbb{R}_+$, trong đó c với $c(S) = \sum_{e \in S} c(e)$. Bài toán *Phủ Submodular với chi phí tối thiểu* (MCSC) yêu cầu tìm tập $S \subseteq V$ sao cho $g(S) \geq T$ với chi phí nhỏ nhất.

Nghiên cứu	Tỷ lệ xấp xỉ	Truy vấn	Độ phức tạp song song
Bài toán Phủ Submodular (MCSC)			
[4]	$1 + \log(\frac{\max_{e \in S} g(e)}{\beta})$	$\Omega(n S)$	$\Omega(S)$
[5]	$(1 + \log \frac{T}{\epsilon}, 1 - \epsilon)$	$\Omega(n S)$	$\Omega(S)$
[6]	$\frac{H(\min\{\max_{e \in S} g(e), T\})}{1 - 5\epsilon}$	$O(\frac{n \log(nT) \log(T)(\log T + \log \log(nT))}{\epsilon^4})$	$O(\frac{\log(nT) \log(T)(\log T + \log \log(nT))}{\epsilon^4})$
LA	$((1 + \epsilon)(1 + \ln \frac{1}{\lambda}), 1 - \lambda)$	$O(\frac{n^2 \log^2(n/\epsilon) \log(\frac{1}{\lambda})}{\epsilon^3})$	$O(\frac{\log(n) \log(1/\lambda)}{\epsilon^2})$
		hoặc $O(\frac{n \log(n) \log^2(n/\epsilon) \log(\frac{1}{\lambda})}{\epsilon^3})$	hoặc $O(\frac{\log^2(n) \log(1/\lambda)}{\epsilon^2})$
Phủ submodular với chi phí đồng nhất			
[7]	$O(\log T)$	$O(n \log(n \log T) \log T)$	$O(\log(n \log T) \log T)$
LA	$((1 + \epsilon)(1 + \ln \frac{1}{\lambda}), 1 - \lambda)$	$O(\frac{n^2 \log^2(n) \log(\frac{1}{\lambda})}{\epsilon^3})$	$O(\frac{\log(n) \log(1/\lambda)}{\epsilon^2})$
		hoặc $O(\frac{n \log^3(n) \log(\frac{1}{\lambda})}{\epsilon^3})$	hoặc $O(\frac{\log^2(n) \log(1/\lambda)}{\epsilon^2})$

Bảng 2.1: Bảng so sánh các thuật toán hiện có cho bài toán MCSC và SC. Ký hiệu (α, β) tương ứng với thuật toán xấp xỉ song tiêu chí (α, β) . Lưu ý rằng các thuật toán được đề xuất trong [7, 6] chỉ áp dụng cho hàm g nhận giá trị nguyên.

Đóng góp của luận án đối với bài toán MCSC và thảo luận về kết quả. Xuất phát từ các hạn chế và khoảng trống nghiên cứu đối với các nghiên cứu liên quan (xem Bảng 2.1). So với thuật toán có độ phức tạp song song thấp tốt nhất, thuật toán đề xuất loại bỏ hoàn toàn sự phụ thuộc vào tham số $\log T$ và đồng thời làm giảm đáng kể sự phụ thuộc vào $\frac{1}{\epsilon}$.

2.2. THUẬT TOÁN SONG SONG AdaptMCSC

Về mặt kỹ thuật, thuật toán AdaptMCSC kết hợp kỹ thuật *lấy mẫu tuần tự* [8] và chiến lược *tham lam theo ngưỡng* được giới thiệu trong [9]. Để hiện thực hóa ý tưởng này, LA đề xuất một thủ tục con ký hiệu là THRESHSEQ, có nhiệm vụ lựa chọn một tập phần tử thỏa mãn một ngưỡng mật độ lợi ích cho trước nhằm lựa chọn các tập phần tử có giá trị vào tập lời giải. Trong thuật toán chính, thuật toán điều chỉnh chiến lược tham lam theo ngưỡng trong [9] theo hướng chỉ thực hiện một số vòng lặp tuần tự hằng số. Tại mỗi vòng, thuật toán gọi THRESHSEQ như một thủ tục con để lựa chọn một tập phần tử có mật độ lợi ích không nhỏ hơn một ngưỡng thích hợp. Cuối cùng, LA sử dụng kỹ thuật

phân hoạch tập nền V nhằm khống chế tỉ lệ $c_{\max}/c_{\min}$, từ đó suy ra các cận chặt cho độ phức tạp truy vấn và độ phức tạp song song của thuật toán.

a. Mô tả thuật toán

Dựa trên ý tưởng trên, thuật toán AdaptMCSC chia làm 2 pha như sau

Pha 1: Tìm lời giải hợp lệ với chặn chi phí tối ưu

Trong pha đầu tiên, thuật toán xử lý tỉ lệ $c_{\max}/c_{\min}$ và phân chia tập nền V thành tập V_1 , V_0 và V' đảm bảo rằng mọi nghiệm tối ưu đều không chứa phần tử thuộc V_1 , trong khi tập V_0 có thể được đưa trực tiếp vào nghiệm cuối cùng với chi phí nhỏ.

Pha 2: Giải các bài toán con

Trong pha thứ hai, thuật toán giải bài toán con trên thể hiện $(g_{V_0}, V', T - g(V_0))$, trong đó $g_{V_0}(S) = g(V_0 \cup S) - g(S)$ bằng cách gọi thuật toán AdaptMCSCopt (một thủ tục phụ của thuật toán khi đã biết ước lượng tối ưu) nhiều lần, tương ứng với các giá trị ước lượng khác nhau của chi phí tối ưu. Gọi opt' là chi phí của nghiệm tối ưu cho bài toán con này. Do ta có $c'_{\min} \leq \text{opt}' \leq nc'_{\max}$, nên một dãy các ước lượng hợp lý cho opt' có thể được xác định dưới dạng $v_i = \frac{c'_{\min}}{(1-\epsilon)^i}$. Với mỗi giá trị v_i , thuật toán AdaptMCSCopt được gọi để tìm ra một nghiệm ứng viên. Trong số các nghiệm ứng viên thu được, thuật toán lựa chọn nghiệm có chi phí nhỏ nhất S thỏa mãn $g(S) \geq (1 - \lambda)(T - g(V_0))$. Cuối cùng, thuật toán AdaptMCSC trả về nghiệm $S \cup V_0$ làm nghiệm đầu ra cho bài toán MCSC ban đầu. Toàn bộ các bước của thuật toán chính AdaptMCSC được mô tả chi tiết trong Thuật toán 2.1.

Định lý 2.1. Với $\delta \in (0, 1)$ và $\epsilon \in (0, 1/2)$, Thuật toán 2.1 trả về một nghiệm S thỏa mãn các tính chất sau $g(S) \geq T(1 - \lambda)$, $\mathbb{E}[c(S)] \leq (1 + \epsilon) \left(1 + \ln \left(\frac{1}{\lambda}\right)\right) \text{opt}$ với độ phức tạp truy vấn: $O\left(\frac{n^2}{\epsilon^3} \log\left(\frac{1}{\lambda}\right) \log^2\left(\frac{n}{\epsilon}\right)\right)$ và độ phức tạp song song (adaptive complexity): $O\left(\frac{1}{\epsilon^2} \log\left(\frac{1}{\lambda}\right) \log\left(\frac{n}{\epsilon}\right)\right)$.

2.3. THỬ NGHIỆM VÀ KẾT QUẢ

LA tiến hành các thực nghiệm đánh giá thuật toán đề xuất với các thuật toán mới nhất (Bảng 2.2) trên bài toán Ảnh hưởng ngưỡng (IT) về các tiêu chí: (1) Chất lượng lời giải, được đo bằng tổng chi phí của tập lời giải thu được để đạt ngưỡng ảnh hưởng yêu cầu; (2) Độ phức tạp song song (adaptive complexity), phản ánh số vòng truy cập dữ liệu theo mô hình song song của thuật toán; (3) Thời gian thực hiện của thuật toán trong các thí nghiệm thực tế. LA sử dụng thư viện OpenMP để song song hóa các thuật toán. Các thuật toán được viết bởi ngôn ngữ C++. Các thực nghiệm trên máy tính hiệu năng cao (HPC) với các thông số: partition=large, #threads(CPU)=64, node=2, max memory = 3,073 GB. $\epsilon = 0.1$ cho tất cả thuật toán. Từ các kết quả thực nghiệm, có thể rút ra rằng

Thuật toán 2.1: Thuật toán AdaptMCSC

Input: $(g, V, T), \epsilon \in (0, 1/2), \delta \in (0, 1)$.

Output: A subset S

1. $S \leftarrow \emptyset, \epsilon' \leftarrow 4\epsilon/23$
 2. Sort $V = \{u_1, u_2, \dots, u_{|V|}\}$ in decreasing cost order
 3. $j \leftarrow \min\{i : g(\{u_1, u_2, \dots, u_i\}) \geq T\}$
 4. $c'_{min} \leftarrow \epsilon' c(u_j)/n, c'_{max} \leftarrow jc(u_j)$
 5. $V_0 \leftarrow \{u \in V : c(u) < c'_{min}\}$
 6. $V_1 \leftarrow \{u \in V : c(u) > c'_{max}\}$
 7. $V' \leftarrow V \setminus (V_0 \cup V_1), T' \leftarrow T - g(V_0), g'(\cdot) \leftarrow g_{V_0}(\cdot)$
 8. **for** $i = 0$ **to** $\lceil \log_{1/(1-\epsilon')} (n^3/\epsilon') \rceil$ **in parallel do**
 9. $v_i = c'_{min}/(1 - \epsilon')^i$
 10. $S(i) \leftarrow \text{AdaptMCSCopt}(V', g', T', \epsilon', \lambda, v_i)$
 11. **end**
 12. $S \leftarrow \arg \min\{c(S(i)) : g(S(i)) \geq T'(1 - \lambda), i = 1, \dots, \lceil \log_{1/(1-\epsilon')} (n^3/\epsilon') \rceil\}$
 13. $S_{final} \leftarrow S \cup V_0$
 14. **return** S_{final}
-

Bảng 2.2: Bảng thống kê các dữ liệu được sử dụng

Tên mạng	Nguồn	Số đỉnh (n)	Số cạnh (m)
EmailEuCore	SNAP	1,005	25,571
EgoFacebook	SNAP	4,039	88,234
NetHEPT	SNAP	15,233	62,774

các thuật toán MINSMC và AdaptMCSC không chỉ duy trì được chất lượng lời giải tương đương với phương pháp tham lam mà còn giảm đáng kể số vòng lặp tuần tự cần thiết. Đặc biệt, thuật toán AdaptMCSC do tác giả đề xuất thường đạt số vòng lặp tuần tự nhỏ nhất trong hầu hết các thiết lập thực nghiệm và vượt trội về mặt thời gian thực thi, đặc biệt trên các bộ dữ liệu có quy mô lớn. Kết quả này phù hợp với các phân tích lý thuyết của thuật toán

CHƯƠNG 3

BÀI TOÁN PHỦ k -SUBMODULAR VỚI CHI PHÍ TỐI THIỂU

3.1. ĐẶT VẤN ĐỀ VÀ PHÁT BIỂU BÀI TOÁN

Trong nhiều ứng dụng thực tiễn, việc chỉ thêm một phần tử vào lời giải S là chưa đủ để mô hình hóa đầy đủ bản chất của bài toán. Chẳng hạn, trong bài toán lan truyền ảnh hưởng theo ngưỡng, mỗi người dùng có thể bắt đầu chịu ảnh hưởng từ những chủ đề khác nhau. Tương tự, trong bài toán bố trí cảm biến, mỗi loại cảm biến có chức năng đo lường các đại lượng khác nhau như nhiệt độ, ánh sáng, độ ẩm, v.v.

Do đó, để mô hình hóa các tình huống này một cách tự nhiên và linh hoạt hơn, cần sử dụng lớp hàm k -submodular, cho phép mỗi phần tử có thể được gán vào một trong k trạng thái hoặc nhãn khác nhau.

Cho tập nền $V = \{e_1, e_2, \dots, e_n\}$ và một số nguyên k . Ký hiệu $[k] = \{1, 2, \dots, k\}$ và đặt

$$(k+1)^V = \{(V_1, V_2, \dots, V_k) \mid V_i \subseteq V, \forall i \in [k], V_i \cap V_j = \emptyset, \forall i \neq j\}$$

là họ các bộ gồm k tập con đôi một rời nhau của V , được gọi là một **k -tập** (k -set). Với $\mathbf{x} = (X_1, X_2, \dots, X_k) \in (k+1)^V$, ta định nghĩa $\text{supp}_i(\mathbf{x}) = X_i$, $\text{supp}(\mathbf{x}) = \bigcup_{i \in [k]} X_i$. Tập X_i được gọi là **tập thứ i của $\mathbf{x}$** . Ngoài ra, ký hiệu $\mathbf{0} = (\emptyset, \dots, \emptyset)$ là k -tập rỗng. Nếu $e \in X_i$ thì ta viết $\mathbf{x}(e) = i$ và nói rằng i là **vị trí** của phần tử e trong $\mathbf{x}$; ngược lại, nếu $e \notin \text{supp}(\mathbf{x})$ thì $\mathbf{x}(e) = 0$. Việc thêm một phần tử $e \notin \text{supp}(\mathbf{x})$ vào tập X_i được ký hiệu bởi $\mathbf{x} \sqcup (e, i)$. Ta cũng có thể biểu diễn

$$\mathbf{x} = \{(e_1, i_1), (e_2, i_2), \dots, (e_t, i_t)\},$$

trong đó $e_j \in \text{supp}(\mathbf{x})$ và $i_j = \mathbf{x}(e_j)$ với mọi $1 \leq j \leq t$. Trong trường hợp $X_i = \{e\}$ và $X_j = \emptyset$ với mọi $j \neq i$, khi đó $\mathbf{x}$ được ký hiệu gọn là (e, i) .

Với hai k -tập $\mathbf{x} = (X_1, X_2, \dots, X_k)$ và $\mathbf{y} = (Y_1, Y_2, \dots, Y_k)$ trong $(k+1)^V$, ta ký hiệu $\mathbf{x} \subseteq \mathbf{y}$ nếu và chỉ nếu $X_i \subseteq Y_i$ với mọi $i \in [k]$.

Để đơn giản, ta giả sử rằng hàm g là không âm, tức là $g(\mathbf{x}) \geq 0$ với mọi $\mathbf{x} \in (k+1)^V$, và được chuẩn hoá, nghĩa là $g(\mathbf{0}) = 0$. Một hàm $g : (k+1)^V \mapsto \mathbb{R}_+$ được gọi là *k -submodular* nếu với mọi $\mathbf{x} = (X_1, X_2, \dots, X_k)$ và $\mathbf{y} = (Y_1, Y_2, \dots, Y_k)$ trong $(k+1)^V$, ta có

$$g(\mathbf{x}) + g(\mathbf{y}) \geq g(\mathbf{x} \sqcap \mathbf{y}) + g(\mathbf{x} \sqcup \mathbf{y}),$$

trong đó

$$\mathbf{x} \sqcap \mathbf{y} = (X_1 \cap Y_1, \dots, X_k \cap Y_k)$$

với

$$\mathbf{x} \sqcup \mathbf{y} = (Z_1, \dots, Z_k), \quad Z_i = X_i \cup Y_i \setminus \bigcup_{j \neq i} (X_j \cup Y_j).$$

Giả sử mỗi phần tử $e \in V$ được gán một chi phí dương $c(e)$, khi đó tổng chi phí của một nghiệm $\mathbf{x}$ được xác định bởi $c(\mathbf{x}) = \sum_{i=1}^k \sum_{e \in X_i} c(e)$. Trong chương này, LA nghiên cứu bài toán Bao phủ k -Submodular (kSC), được định nghĩa như sau.

Định nghĩa 3.1 (Phủ k -Submodular (kSC)). Cho một ngưỡng $T > 0$, bài toán kSC yêu cầu tìm một nghiệm $\mathbf{x} = (X_1, X_2, \dots, X_k) \in (k+1)^V$ có tổng chi phí nhỏ nhất $c(\mathbf{x})$ sao cho $g(\mathbf{x}) \geq T$.

Bài toán kSC có nhiều ứng dụng trọng lan truyền thông tin trên mạng xã hội, tối đa doanh thu được mô tả cụ thể như sau:

Ứng dụng 1: Ảnh hưởng ngưỡng với k chủ đề (ITk). Trong các kịch bản marketing lan truyền và gợi ý sản phẩm, một doanh nghiệp có thể mong muốn triển khai một chiến dịch quảng cáo gồm k nhóm sản phẩm khác nhau và tác động đến ít nhất T người dùng trong mạng xã hội. Để giảm thiểu tổng ngân sách, doanh nghiệp cần lựa chọn một tập các người dùng có ảnh hưởng lớn, đủ khả năng tạo ra mức lan tỏa mong muốn. Mô hình này là sự mở rộng tự nhiên của các mô hình lan truyền thông tin cổ điển cho trường hợp một chủ đề ($k = 1$), đồng thời mở rộng bài toán ngưỡng ảnh hưởng được nghiên cứu rộng rãi.

Ứng dụng 2: Ngưỡng doanh thu với chi phí tối thiểu (Revenue Threshold-RT) Cho đồ thị $G = (V, E)$, trong đó V là tập các đỉnh đại diện cho khách hàng và mỗi cạnh $(u, v) \in E$ có trọng số $w_{uv} \geq 0$ biểu diễn mức độ liên hệ giữa hai khách hàng u và v . Mục tiêu là phân bổ k loại sản phẩm cho khách hàng bằng cách xác định một k -tập $\mathbf{s} = (S_1, S_2, \dots, S_k)$, trong đó $S_i \subseteq V$ là tập khách hàng được phân bổ sản phẩm loại i , và các tập $S_1, \dots, S_k$ đôi một rời nhau (tức là mỗi khách hàng được gán nhiều nhất một loại sản phẩm). Hàm doanh thu của một phương án phân bổ $\mathbf{s}$ được định nghĩa bởi $g(\mathbf{s}) = \sum_{u \in V} \sum_{i=1}^k \left(\sum_{v \in S_i} w_{uv} \right)^{\alpha_{u,i}}$,

Đóng góp của Luận án. Luận án tập trung nghiên cứu bài toán kSC trong hai trường hợp chính: (1) chi phí đồng nhất (*uniform-cost*) và (2) chi phí tổng quát (*general-cost*).
a. Trường hợp bài toán Phủ k -Submodular đồng nhất.

Đối với trường hợp chi phí đồng nhất, luận án đề xuất thuật toán FBiU và phân tích các đảm bảo lý thuyết của thuật toán *xấp xỉ tiêu chí kép*. Cụ thể, thuật toán trả về một nghiệm $\mathbf{s}$ sao cho kích thước nghiệm được không chế so với nghiệm tối ưu và đồng thời giá trị hàm mục tiêu đạt được một tỷ lệ nhất định so với ngưỡng yêu cầu.

Bảng 3.1 trình bày so sánh giữa thuật toán FBIU được đề xuất trong luận án và thuật toán luồng đã được giới thiệu trước đó trong [10]. Kết quả cho thấy thuật toán FBIU đạt được đảm bảo xấp xỉ tiêu chí kép mạnh hơn đáng kể. Đặc biệt, khi đặt $\delta = \epsilon$, thuật toán đảm bảo nghiệm thu được có kích thước $|supp(s)| \leq (1 + \epsilon)(1 + \ln(\frac{1}{\delta}))$, đồng thời đạt giá trị hàm mục tiêu $g(s) \geq \frac{1-\epsilon^2}{2} \text{opt}$, trong đó opt là kích thước nghiệm tối ưu. So với thuật toán luồng trước đây, kết quả này cải thiện đáng kể các đảm bảo lý thuyết trong khi vẫn duy trì độ phức tạp truy vấn đa thức.

Bảng 3.1: Các thuật toán cho bài toán kSC. Bộ (x, y) biểu diễn tỷ lệ xấp xỉ tiêu chí kép (x, y) của một thuật toán, tức là thuật toán trả về nghiệm s sao cho $|supp(s)| \leq y \cdot \text{opt}$ và $g(s) \geq x \cdot T$, với $x, y > 0$. So với thuật toán luồng trong [10], thuật toán FBIU được đề xuất trong luận án đạt được đảm bảo xấp xỉ tiêu chí kép mạnh hơn đáng kể.

Tài liệu	Tỷ lệ xấp xỉ	Độ phức tạp truy vấn
FBIU (Thuật toán 3.3, LA)	$\left((1 + \epsilon)(1 + \ln(\frac{1}{\delta})), \frac{1-\delta}{2}\right)$	$O\left(\frac{nk \log^2(n)}{\epsilon} \log\left(\frac{n}{\delta}\right)\right)$
Thuật toán luồng [10]	$\left(\frac{1-\epsilon}{2\epsilon}, \frac{1-\epsilon^2}{2}\right)$	$O\left(nk \frac{\log(n)}{\epsilon}\right)$

b. Trường hợp bài toán Phủ k-Submodular tổng quát Đối với trường hợp chi phí tổng quát, luận án đề xuất hai thuật toán xấp xỉ tiêu chí kép xác định cho bài toán kSC trong thời gian đa thức, bao gồm thuật toán *Extended Greedy* và thuật toán FBI. Các thuật toán này được thiết kế nhằm đồng thời cải thiện các đảm bảo lý thuyết về chất lượng nghiệm và giảm độ phức tạp truy vấn so với các phương pháp hiện có trong tài liệu.

Cụ thể, thuật toán *Extended Greedy* đạt độ phức tạp truy vấn $O(n^2k)$ và đảm bảo tỷ lệ xấp xỉ tiêu chí kép $(\frac{1}{2}(\ln(\frac{1}{\epsilon}) + \frac{1+\epsilon}{\epsilon}), \frac{1-\epsilon}{2})$ đối với trường hợp hàm lợi ích g là đơn điệu, và $(\frac{1}{3}(\ln(\frac{1}{\epsilon}) + \frac{2+\epsilon}{\epsilon}), \frac{1-\epsilon}{3})$ đối với trường hợp g không đơn điệu.

Bên cạnh đó, luận án đề xuất thuật toán FBI cho trường hợp hàm lợi ích đơn điệu, với độ phức tạp truy vấn $O(\frac{nk}{\epsilon^2} \log(\frac{n}{\epsilon}))$, đồng thời đạt tỷ lệ xấp xỉ tiêu chí kép gần tương đương với *Extended Greedy*.

Bảng 3.2 trình bày so sánh giữa các thuật toán được đề xuất trong luận án với các thuật toán tiên tiến hiện nay cho bài toán kSC. Có thể thấy rằng các thuật toán của luận án đạt được đảm bảo tốt hơn về chi phí nghiệm so với các thuật toán luồng trong [11], đồng thời giảm đáng kể độ phức tạp truy vấn từ dạng không đa thức xuống còn đa thức.

3.2. THUẬT TOÁN CHO TRƯỜNG HỢP CHI PHÍ ĐỒNG NHẤT

Thuật toán FBIU (Thuật toán 3.1) cho trường hợp chi phí đồng nhất được mô tả trong thuật toán 3.1. Thuật toán FBIU trước hết xây dựng tập ứng viên

Thuật toán	Tỷ lệ xấp xỉ	Độ phức tạp truy vấn	Loại thuật toán
Hàm đơn điệu			
Tham lam mở rộng (LA)	$\left(\frac{1}{2}(\log(\frac{1}{\epsilon}) + \frac{1+\epsilon}{\epsilon}), \frac{1-\epsilon}{2}\right)$	$O(n^2k)$	-
FBI (LA)	$\left(\frac{1}{2}(1+\epsilon)(\ln(\frac{1}{\delta}) + \frac{\delta+1}{\delta}), \frac{1-\delta}{2}\right)$	$O(\frac{nk}{\epsilon^2} \log(\frac{n}{\epsilon}))$	-
Thuật toán 2 [11]	$\left(\frac{3-\epsilon}{2\epsilon(1-\epsilon)}, \frac{1-\epsilon}{2}\right)$	$O(nk \frac{\log(\frac{c_{min}}{nc_{max}})}{\log(1-\epsilon)})$	Luồng
Thuật toán 3 [11]	$\left(\frac{3-\epsilon}{2\epsilon(1-\epsilon)}, \frac{1-\epsilon}{2}\right)$	$O\left(nk \frac{\log(\epsilon T(B \max_{e \in V, i \in [k]} g((e, i)))^{-1})}{\log(1-\epsilon)}\right)$	Luồng
Hàm không đơn điệu			
Tham lam mở rộng (LA)	$\left(\frac{1}{3}(\log(\frac{1}{\epsilon}) + \frac{2+\epsilon}{\epsilon}), \frac{1-\epsilon}{3}\right)$	$O(n^2k)$	-
Thuật toán 2 [11]	$\left(\frac{4-\epsilon}{3\epsilon(1-\epsilon)}, \frac{1-\epsilon}{3}\right)$	$O(nk \frac{\log(\frac{c_{min}}{nc_{max}})}{\log(1-\epsilon)})$	Luồng
Thuật toán 3 [11]	$\left(\frac{4-\epsilon}{4\epsilon(1-\epsilon)}, \frac{1-\epsilon}{3}\right)$	$O\left(nk \frac{\log(\epsilon T(B \max_{e \in V, i \in [k]} g((e, i)))^{-1})}{\log(1-\epsilon)}\right)$	Luồng

Bảng 3.2: Các thuật toán cho bài toán kSC. Bộ (x, y) biểu diễn tỷ lệ xấp xỉ tiêu chí kép (x, y) . Các thuật toán được đề xuất trong luận án (Extended Greedy và FBI) là các thuật toán **offline**, tức là giả định có quyền truy cập đầy đủ vào tập nền. Ngược lại, các thuật toán của Wang *et al.* (2023) là các thuật toán luồng, xử lý các phần tử tuần tự với bộ nhớ và số lần duyệt hạn chế.

Thuật toán 3.1: Thuật toán FBIU

Input: Một thể hiện (V, g, k, T) , trong đó f là hàm đơn điệu, $\epsilon \in (0, 1/2)$, $\delta > 0$.

Output: Một nghiệm $\mathbf{s}$.

- 1: $O = \{v \in \mathbb{N} : 1 \leq (1+\epsilon)^i \leq n\}$
 - 2: **foreach** $v \in O$ **do**
 - 3: $\mathbf{s}_v \leftarrow \text{FBIUOpt}(V, g, k, v, \delta)$
 - 4: **end**
 - 5: $\mathbf{s} \leftarrow \arg \max \{g(\mathbf{s}_v) : |\text{supp}(\mathbf{s}_v)| \leq v \log(\frac{1}{\epsilon}), v \in O\}$
 - 6: **return** $\mathbf{s}$
-

$$O = \{(1 + \epsilon)^i : 1 \leq (1 + \epsilon)^i \leq n, i \in \mathbb{N}\}$$

các giá trị v . Với mỗi $v \in O$, thuật toán gọi thủ tục $\text{FBIOpt}(V, f, k, v, \delta)$ để tính một nghiệm ứng viên $\mathbf{s}_v$. Sau khi sinh ra toàn bộ các nghiệm ứng viên, FBIU lựa chọn nghiệm có giá trị hàm mục tiêu lớn nhất trong số các nghiệm thỏa mãn ràng buộc kích thước $|\text{supp}(\mathbf{s}_v)| \leq v \log(\frac{1}{\epsilon})$ (Dòng 5). Cuối cùng, thuật toán xuất ra nghiệm được chọn $\mathbf{s}$.

Định lý 3.1. Với $\epsilon, \delta \in (0, 1)$, với xác suất ít nhất $1 - \delta$, Thuật toán 3.1 trả về một nghiệm xấp xỉ tiêu chí kép $\left((1 + \epsilon) \log(\frac{1}{\delta}), \frac{1 - \delta}{2}\right)$ và có độ phức tạp truy vấn $O\left(\frac{nk \log^2(n)}{\epsilon} \log\left(\frac{n}{\delta}\right)\right)$.

3.2.1. Thực nghiệm

LA tiến hành so sánh thực nghiệm thuật toán FBIU với hai phương pháp cơ sở tiên tiến hiện nay. Việc đánh giá được thực hiện dựa trên ba tiêu chí chính: (i) chất lượng lời giải (số lượng phần tử trong lời giải) (ii) số lượng truy vấn hàm, và (iii) thời gian chạy thuật toán.

Các thí nghiệm được thực hiện trên bài toán ứng dụng **k -type Product Revenue Cover** trên các bộ dữ liệu chuẩn¹.

Các kết quả thực nghiệm cho thấy FBIU vượt trội về chất lượng lời giải và nhanh hơn đáng kể so với các thuật toán còn lại.

3.3. THUẬT TOÁN CHO BÀI TOÁN PHỦ k -SUBMODULAR TỔNG QUÁT

Trong phần này, luận án đề xuất các thuật toán cho bài toán k SC trong trường hợp tổng quát, bao gồm: (i) thuật toán tham lam mở rộng, và (ii) thuật toán FBI cải tiến thời gian chạy của tham lam mở rộng với trường hợp hàm đơn điệu.

3.3.1. Thuật toán tham lam mở rộng

Ý tưởng của thuật toán này được kế thừa từ thuật toán tham lam cổ điển cho bài toán và được tổng quát hóa để xử lý hàm mục tiêu k -submodular. Chi tiết được mô tả trong thuật toán 3.2.

Định lý 3.2. Với mọi $\epsilon \in (0, 1)$, thuật toán 3.2 có độ phức tạp truy vấn $O(n^2 k)$ và đạt được các tỷ lệ xấp xỉ tiêu chí kép sau:

- Nếu g là hàm đơn điệu, thuật toán đạt tỷ lệ xấp xỉ $\left(\frac{1}{2} \left(\ln\left(\frac{1}{\epsilon}\right) + \frac{1+\epsilon}{\epsilon}\right), \frac{1-\epsilon}{2}\right)$.
- Nếu g là hàm không đơn điệu, thuật toán đạt tỷ lệ xấp xỉ $\left(\frac{1}{3} \left(\ln\left(\frac{1}{\epsilon}\right) + \frac{2+\epsilon}{\epsilon}\right), \frac{1-\epsilon}{2}\right)$.

¹<https://snap.stanford.edu/data>

Thuật toán 3.2: Thuật toán tham lam mở rộng (Extended Greedy) cho bài toán kSC

Input: Thể hiện (V, g, k, T) , tham số $\epsilon \in (0, 1)$

Output: Nghiệm s

```

1:  $s \leftarrow \mathbf{0}; \quad \alpha \leftarrow \begin{cases} \frac{1}{2}, & \text{nếu } g \text{ đơn điệu,} \\ \frac{1}{3}, & \text{nếu } g \text{ không đơn điệu} \end{cases}$ 
2: Định nghĩa  $h : (k+1)^V \mapsto \mathbb{R}_+$  bởi  $f(\cdot) \leftarrow \min\{g(\cdot), T\}$ 
3: while  $h(s) < \alpha(1 - \epsilon)T$  do
4:    $(e, i) \leftarrow \arg \max_{e \in V \setminus \text{supp}(s), i \in [k]} \frac{\Delta_{(e,i)}h(s)}{c(e)}$ 
5:    $s \leftarrow s \sqcup (e, i)$ 
6: end
7: return  $s$ 

```

3.3.2. Thuật toán xấp xỉ nhanh FBi

Thuật toán này gồm 2 Pha như sau: Pha 1 (trích chọn tập con): Từ V , thuật toán xây dựng hai tập con V_0 và V' để đảm bảo tìm kiếm nghiệm một cách khả thi cũng như chặn được độ phức tạp phụ thuộc và $c_{\max}/c_{\min}$. Pha 2 (tìm nghiệm ứng viên và chọn tốt nhất): Thuật toán kết hợp gọi FBiOpt (thủ tục con được đề xuất với giả định biết lời giải tối ưu) nhiều lần trên miền V' với các giá trị đoán v_i , thu được các nghiệm ứng viên s_i và trả về nghiệm kết hợp với $\mathbf{x}_0$ tốt nhất.

Định lý 3.3. Với mọi $\epsilon \in (0, 1/2)$ và $\delta \in (0, 1)$, giả sử g là một hàm đơn điệu và k -submodular. Khi đó, Thuật toán 3.3 trả về một nghiệm đạt tỷ lệ xấp xỉ tiêu chí kép

$$\left(\frac{1}{2}(1 + \epsilon) \left(\ln \left(\frac{1}{\delta} \right) + \frac{1 + \delta}{\delta} \right), \frac{1 - \delta}{2} \right),$$

và có độ phức tạp truy vấn

$$O \left(\frac{nk}{\epsilon^2} \log \left(\frac{n}{\epsilon} \right) \right).$$

3.3.3. Thực nghiệm

Để đánh giá hiệu quả của các thuật toán đề xuất trong luận án, bao gồm *Tham lam mở rộng* và FBi, LA tiến hành so sánh các phương pháp này với các thuật toán tiên tiến hiện có được liệt kê trong Bảng 3.2. Việc so sánh được thực hiện theo ba tiêu chí quan trọng: (i) chất lượng lời giải thu được, (ii) số lượng truy vấn tới hàm mục tiêu g , phản ánh chi phí tính toán của thuật toán, và (iii) thời gian chạy thực tế.

Thuật toán 3.3: Thuật toán FBi (Fast Bicriteria)

Input: Thể hiện (V, g, k, T) , g đơn điệu; $\epsilon \in (0, 1/2)$, $\delta \in (0, 1)$

Output: Nghiệm $\mathbf{s}$

- 1: $\mathbf{x} \leftarrow \mathbf{0}$; $\epsilon' \leftarrow \epsilon/8$; $\mathbf{x}_0 \leftarrow \mathbf{0}$
 - 2: $\mathbf{x} \leftarrow$ nghiệm của ukS trên (g, V, k) bằng thuật toán xấp xỉ 1/2 trong [12]
 - 3: **foreach** $e \in V \setminus \text{supp}(\mathbf{x})$ **do**
 - 4: | chọn ngẫu nhiên $i \in [k]$ và $\mathbf{x} \leftarrow \mathbf{x} \sqcup (e, i)$
 - 5: **end**
 - 6: Sắp xếp các phần tử trong $\text{supp}(\mathbf{x}) = \{e_1, \dots, e_n\}$ theo thứ tự chi phí không giảm
 - 7: $j \leftarrow \min \left\{ i : g(\{(e_1, \mathbf{x}(e_1)), \dots, (e_i, \mathbf{x}(e_i))\}) \geq T \right\}$
 - 8: $c_m \leftarrow \epsilon' c(e_j)/n$; $c_M \leftarrow j c(e_j)$
 - 9: $V_0 \leftarrow \{e \in V : c(e) < c_m\}$; $V' \leftarrow \{e \in V : c_m \leq c(e) \leq c_M\}$
 - 10: $\mathbf{x}_0 \leftarrow$ nghiệm của ukS trên (g, V_0, k) bằng thuật toán xấp xỉ 1/2 trong [12]
 - 11: $T_0 \leftarrow g(\mathbf{x}_0)$; định nghĩa $g'(\cdot) \leftarrow g(\mathbf{x}_0 \sqcup \cdot) - g(\mathbf{x}_0)$
 - 12: $T' \leftarrow T - T_0/\alpha$
 - 13: **for** $i = 0$ **to** $\left\lceil \log_{\frac{1}{1-\epsilon'}} \left(\frac{jn^2}{\epsilon'} \right) \right\rceil$ **do**
 - 14: | $v_i \leftarrow c_m/(1 - \epsilon')^i$
 - 15: | $\mathbf{s}_i \leftarrow$ kết quả của FBiOpt với đầu vào $(V', g', k, T', \epsilon', \delta, v_i)$
 - 16: **end**
 - 17: $\mathbf{s} \leftarrow \arg \min \left\{ c(\mathbf{s}_i) : g(\mathbf{s}_i \sqcup \mathbf{x}_0) \geq (1 - \delta)T/2 \right\}$
 - 18: **return** $\mathbf{s} \sqcup \mathbf{x}_0$
-

3.3.3.1. Thiết lập thực nghiệm

Luận án tiến hành thực nghiệm so sánh hai thuật toán đề xuất là *Tham lam mở rộng* và FBI với các phương pháp tiên tiến hiện có trên hai ứng dụng tiêu biểu của bài toán kSC, bao gồm: (i) bài toán *ảnh hưởng ngưỡng k chủ đề-IT* và (ii) bài toán *Ngưỡng doanh thu -RT*. Các thuật toán được sử dụng để so sánh được lựa chọn dựa trên các phương pháp tốt nhất đã được công bố trong tài liệu (xem Bảng 3.2). Trong toàn bộ các thí nghiệm, các tham số được cố định ở mức $\epsilon = 0.1$ và $\delta = 0.1$ với các thiết lập chuẩn theo các nghiên cứu liên quan. Trong phần thực nghiệm, LA sử dụng bốn bộ dữ liệu chuẩn về mạng xã hội tại SNAP².

3.3.3.2. Kết quả thực nghiệm

Các kết quả trên nhiều bộ dữ liệu thực và tổng hợp cho thấy các thuật toán đề xuất luôn duy trì chất lượng lời giải tương đương với các thuật toán tiên tiến hiện có. Đồng thời, các phương pháp của luận án giảm đáng kể số lượng truy vấn và thời gian chạy so với các thuật toán của Wang, qua đó cho thấy hiệu quả tính toán vượt trội và tiềm năng áp dụng cao trên các bài toán quy mô lớn.

²<https://snap.stanford.edu>

CHƯƠNG 4

BÀI TOÁN TỐI ĐA HÀM SUBMODULAR VỚI RÀNG BUỘC CHI PHÍ CÔNG BẰNG

4.1. PHÁT BIỂU BÀI TOÁN VÀ ĐÓNG GÓP CỦA LUẬN ÁN

Trong nhiều ứng dụng thực tế, đặc biệt là bài toán *tối đa ảnh hưởng trong mạng xã hội*, các phần tử trong tập nền có thể được phân chia thành các nhóm người dùng khác nhau, phản ánh các đặc điểm như khu vực địa lý, sở thích hoặc nhân khẩu học. Khi đó, việc lựa chọn tập hạt giống không chỉ chịu ràng buộc ngân sách tổng mà còn cần đảm bảo tính công bằng giữa các nhóm. Bài toán tối đa hàm submodular với ràng buộc chi phí công bằng (SMFC), được định nghĩa như sau:

Định nghĩa 4.1. Cho trước một tập nền V , một hàm submodular đơn điệu $f : 2^V \rightarrow \mathbb{R}_+$, một ngân sách tổng B , một tập gồm K nhóm $C_1, C_2, \dots, C_K$ với $C_i \subseteq V$, và một tỷ lệ công bằng $\alpha \in (0, 1)$. Mỗi phần tử $u \in V$ có một chi phí dương $c(u)$. Mục tiêu là tìm một tập $S \subseteq V$ sao cho giá trị $f(S)$ là lớn nhất, đồng thời thoả mãn các ràng buộc sau:

1. Tổng chi phí các phần tử được chọn trong mỗi nhóm không vượt quá αB , tức là

$$c(S \cap C_i) \leq \alpha B, \forall i \in [K]$$

2. Tổng chi phí của tập được chọn không vượt quá ngân sách tổng, tức là

$$c(S) \leq B$$

Ứng dụng trong tối đa ảnh hưởng. Trong nhiều ứng dụng thực tế của marketing lan truyền và phổ biến thông tin trên mạng xã hội, các người dùng trong mạng thường được phân chia thành nhiều nhóm khác nhau, phản ánh các đặc điểm như khu vực địa lý, độ tuổi, sở thích hoặc nhân khẩu học. Trong các chiến dịch quảng bá hoặc truyền thông cộng đồng, ngoài mục tiêu tối đa hóa phạm vi lan truyền của thông tin, các nhà tổ chức còn mong muốn đảm bảo rằng chiến dịch được phân bổ một cách công bằng giữa các nhóm người dùng khác nhau.

Do đó, bài toán đặt ra là lựa chọn một tập hạt giống có khả năng tạo ra mức lan truyền thông tin lớn nhất trong mạng xã hội, đồng thời đảm bảo ngân sách được phân bổ hợp lý giữa các nhóm người dùng.

Đóng góp của Luận án. LA nghiên cứu bài toán SMFC trong hai trường hợp: (i) *chi phí đồng nhất*, trong đó chi phí của mỗi phần tử đều bằng 1, và (ii) *chi phí tổng quát*, trong đó mỗi phần tử có thể có một chi phí dương bất kỳ.

Bảng 4.1: Tóm tắt các kết quả chính của luận án cho bài toán *Tối đa hàm Submodular* với ràng buộc chi phí công bằng (SMFC)

Thuật toán	Trường hợp	Hệ số xấp xỉ	Độ phức tạp
TGFast	Chi phí đồng nhất ($c(e) = 1$)	$1/2 - \epsilon$	$O\left(\frac{n}{\epsilon} \log \frac{k}{\epsilon}\right)$
<i>Threshold Greedy Fast</i>	Chi phí tổng quát ($c(e) > 0$)	$\frac{\alpha}{\alpha+2} - \epsilon$	$O(n \log n)$

Trong bảng trên, $n = |V|$ là kích thước tập nền, k là ngân sách trong trường hợp chi phí đồng nhất, $\alpha \in (0, 1)$ là tỷ lệ công bằng giữa các nhóm, và $\epsilon > 0$ là tham số chính xác điều khiển sự đánh đổi giữa chất lượng nghiệm và thời gian chạy của thuật toán.

4.2. THUẬT TOÁN ĐỀ XUẤT

4.2.1. Thuật toán cho trường hợp đặc biệt khi chi phí đồng nhất

Đối với thiết lập này, luận án đề xuất thuật toán *Threshold Greedy*, ký hiệu là TGFast. Thuật toán TGFast kế thừa và điều chỉnh ý tưởng của phương pháp tham lam ngưỡng, nhằm lựa chọn các phần tử có đóng góp lớn, đồng thời bảo đảm không vi phạm các ràng buộc phân bổ theo nhóm, được mô hình hoá dưới dạng ràng buộc matroid. Giả mã chi tiết của thuật toán TGFast được trình bày trong Thuật toán 4.2.

Định lý 4.1. *Thuật toán 4.1 có độ phức tạp thời gian $O\left(\frac{n}{\epsilon} \log \frac{k}{\epsilon}\right)$ và trả về một nghiệm có tỷ lệ xấp xỉ $1/2 - \epsilon$ cho bài toán MCFA.*

4.2.2. Thuật toán cho trường hợp tổng quát

Luận án đề xuất thuật toán TGFast với lệ xấp xỉ $\frac{\alpha}{\alpha+2} - \epsilon$, trong đó $\epsilon \in (0, 1)$ là một hằng số tùy ý nhỏ. Ý tưởng của thuật toán TGFast mở rộng thuật toán cho trường hợp chi phí đồng nhất TGFastU với tiêu chí lựa chọn phần tử dựa trên *tỷ lệ mật độ* (density ratio), tức là tỷ lệ giữa độ tăng biên của hàm mục tiêu và chi phí của phần tử. Cách tiếp cận này cho phép lựa chọn các phần tử có đóng góp lớn trong khi vẫn đảm bảo không vi phạm các ràng buộc về chi phí tổng và chi phí theo nhóm trong mỗi vòng lặp.

Thuật toán khởi tạo tập nghiệm $S = \emptyset$ và trước hết tìm một phần tử $e_{\max} = \arg \max_{e \in V} f(e)$ có giá trị đơn lẻ lớn nhất, đồng thời đặt $M = f(e_{\max})$. Phần chính của thuật toán được thực hiện trong một vòng lặp ngoài, trong đó một ngưỡng θ được sử dụng để kiểm soát việc chọn phần tử. Ban đầu, θ được đặt tỉ lệ với M và sau mỗi vòng lặp ngoài, giá trị này được giảm theo hệ số $(1 - \epsilon)$.

Tại mỗi vòng lặp, thuật toán duyệt qua các phần tử trong $V \setminus S$. Một phần tử e được thêm vào tập S nếu việc thêm e không vi phạm ràng buộc ngân sách tổng, không vi phạm ràng buộc chi phí theo nhóm, và đồng thời thoả mãn điều kiện tỷ lệ mật độ

Thuật toán 4.1: Thuật toán TGFastU

Input: Hàm submodular f , tập các nhóm $\mathcal{C} = \{C_1, C_2, \dots, C_K\}$, tỷ lệ công bằng $\alpha \in (0, 1)$, ngân sách k , tham số chính xác $\epsilon > 0$

Output: Tập nghiệm S

```

1:  $S \leftarrow \emptyset$ ;
2:  $M \leftarrow \max_{e \in V} f(\{e\})$ ;
3:  $\theta \leftarrow M$ ;
4: while  $\theta \geq \epsilon M/k$  do
5:   foreach  $e \in V$  do
6:     if  $|S \cap C(e)| < \alpha k$  and  $|S| \leq k$  then
7:       if  $f(S \cup \{e\}) - f(S) \geq \theta$  then
8:          $S \leftarrow S \cup \{e\}$ ;
9:       end
10:    end
11:  end
12:   $\theta \leftarrow (1 - \epsilon)\theta$ ;
13: end
14: return  $S$ ;

```

$f(e | S)/c(e) \geq \theta$. Thuật toán dừng khi ngưỡng θ nhỏ hơn một giá trị biên được xác định trước. Cuối cùng, thuật toán trả về nghiệm tốt hơn giữa tập S thu được và nghiệm đơn phần tử $\{e_{\max}\}$.

Định lý 4.2. Thuật toán 4.2 có độ phức tạp truy vấn $O(\frac{n}{\epsilon} \log(n \frac{\alpha+2}{2\alpha}))$ và trả về một nghiệm hợp lệ có tỷ lệ xấp xỉ $\frac{\alpha}{\alpha+2} - \epsilon$ cho bài toán SMFC, trong đó $\alpha \in (0, 1)$ là hệ số công bằng của bài toán và ϵ là tham số chính xác của thuật toán thỏa mãn $\frac{\alpha}{\alpha+2} \geq \epsilon$.

4.3. THỰC NGHIỆM VÀ KẾT QUẢ

Trong phần này, LA tiến hành các thí nghiệm nhằm đánh giá hiệu năng của thuật toán đề xuất TGFastU và TGFast cho bài toán SMFC trên ứng dụng Tối đa ảnh hưởng. Cụ thể, các thí nghiệm được thiết lập để so sánh thuật toán đề xuất với các phương pháp hiện có trên hai tiêu chí chính: (i) *chất lượng lời giải*, được đo bằng giá trị lan truyền ảnh hưởng đạt được, và (ii) *thời gian chạy* của thuật toán.

Thuật toán 4.2: Thuật toán TGFast

Input: Tập nền V , hàm submodular $f : 2^V \rightarrow \mathbb{R}_+$, tập các nhóm

$\mathcal{C} = \{C_1, \dots, C_K\}$, tỷ lệ công bằng $\alpha \in (0, 1)$, ngân sách tổng B , tham số chính xác $\epsilon > 0$

Output: Tập nghiệm $S \subseteq V$

```

1:  $S \leftarrow \emptyset$ ;
2:  $e_{\max} \leftarrow \arg \max_{e \in V} f(e)$ ;
3:  $M \leftarrow f(e_{\max})$ ;
4:  $\beta \leftarrow \frac{\alpha}{\alpha+2}$ ;
5:  $\theta \leftarrow \frac{nM}{\alpha B}$ ;
6: while  $\theta \geq \frac{\beta M}{\alpha B}$  do
7:   foreach  $e \in V \setminus S$  do
8:     if  $c(S) + c(e) \leq B$  then
9:       if  $c(S \cap C(e)) + c(e) \leq \alpha B$  then
10:        if  $\frac{f(S \cup \{e\}) - f(S)}{c(e)} \geq \theta$  then
11:           $S \leftarrow S \cup \{e\}$ ;
12:        end
13:      end
14:    end
15:  end
16:   $\theta \leftarrow (1 - \epsilon)\theta$ ;
17: end
18:  $S' \leftarrow \arg \max_{X \in \{S, \{e_{\max}\}\}} f(X)$ ;
19: return  $S'$ 

```

4.3.1. Thiết lập thực nghiệm

Các thí nghiệm được thiết lập nhằm so sánh hiệu năng của thuật toán đề xuất với thuật toán tiên tiến nhất hiện nay là OPIM, cho bài toán Tối đa ảnh hưởng (IM). Việc so sánh được thực hiện trên hai tiêu chí chính: *chất lượng lời giải* (giá trị lan truyền ảnh hưởng) và *thời gian chạy*. LA sử dụng các tập dữ liệu mạng xã hội công khai từ SNAP¹.

4.3.2. Kết quả thực nghiệm

Trong **trường hợp chi phí đồng nhất**, thuật toán OPIM đạt giá trị lan truyền ảnh hưởng cao hơn một chút so với các thuật toán còn lại nhưng mức chênh lệch này là khá nhỏ so với thuật toán TGFastU. Tuy nhiên TGFastU duy trì thời gian thực thi thấp và tăng gần như tuyến tính theo ngân sách, nhỏ hơn rất nhiều so với OPIM.

Trong **trường hợp chi phí tổng quát**, thuật toán TGFastU thường đạt giá trị hàm mục tiêu cao nhất. Thời gian thực thi của TGFast gần như không thay đổi đáng kể khi ngân sách tăng và luôn thấp hơn đáng kể so với OPIM. Các kết quả thực nghiệm cho thấy thuật toán đề xuất đạt được chất lượng lời giải tốt nhất trong khi vẫn duy trì thời gian chạy cạnh tranh. Đặc biệt, lợi thế của TGFast trở nên rõ rệt hơn khi kích thước tập dữ liệu tăng lên. Sự cải thiện này chủ yếu đến từ hiệu quả của thuật toán TGFast được tích hợp trong TGFast, vốn có tốc độ nhanh hơn các phương pháp tham lam hiện có.

¹<https://snap.stanford.edu/data/>

KẾT LUẬN VÀ KIẾN NGHỊ

Luận án tập trung nghiên cứu lớp bài toán SO bao ba bài toán: (1) Phủ submodular với chi phí tối thiểu (MCSC), (2) Phủ k -Submodular với chi phí tối thiểu (kSC) và (3) Tối đa hóa hàm submodular dưới ràng buộc chi phí công bằng (SMFC). Đây là một lớp bài toán có khả năng mô hình hóa linh hoạt nhiều ứng dụng thực tiễn, bao gồm tối đa hóa ảnh hưởng trên mạng xã hội, lựa chọn đặc trưng, hệ thống khuyến nghị, phân bổ ngân sách tiếp thị, cũng như các bài toán phân bổ tài nguyên dưới các ràng buộc công bằng. Việc nghiên cứu các mô hình này không chỉ có ý nghĩa về mặt lý thuyết mà còn đóng vai trò nền tảng trong thiết kế các hệ thống phân tích dữ liệu hiện đại. Trong khuôn khổ luận án, các kết quả chính đạt được bao gồm:

1. **Đối với bài toán MCSC:** Luận án đề xuất một thuật toán song song hiệu quả cho bài toán phủ submodular với chi phí tối thiểu. Thuật toán đạt tỷ lệ xấp xỉ tiệm cận với các kết quả tốt nhất hiện nay, đồng thời cải thiện đáng kể độ phức tạp thích nghi và khả năng song song hóa. Các phân tích lý thuyết chứng minh chất lượng nghiệm của thuật toán, trong khi các thực nghiệm trên dữ liệu thực cho thấy ưu thế rõ rệt về thời gian thực thi và khả năng mở rộng. Kết quả này góp phần thu hẹp khoảng cách giữa các đảm bảo lý thuyết và yêu cầu triển khai trong thực tế.

2. **Đối với bài toán kSC:** Luận án mở rộng mô hình phủ submodular sang miền k -submodular và đề xuất các thuật toán xấp xỉ tiêu chuẩn kép cho cả trường hợp chi phí đồng nhất và chi phí tổng quát. Các thuật toán đạt được hệ số xấp xỉ tốt hơn các kết quả trước đây, đồng thời giảm đáng kể độ phức tạp truy vấn hàm mục tiêu. Kết quả thực nghiệm cho thấy các phương pháp đề xuất duy trì được chất lượng nghiệm cao trong khi giảm đáng kể chi phí tính toán, qua đó nâng cao khả năng ứng dụng đối với các bài toán dữ liệu lớn.

3. **Đối với bài toán SMFC:** Luận án nghiên cứu một mô hình tối ưu mới kết hợp giữa hiệu quả tối ưu và yêu cầu công bằng trong phân bổ chi phí. Hai thuật toán xấp xỉ được đề xuất tương ứng với trường hợp chi phí đồng nhất và chi phí tổng quát, kèm theo các bảo đảm lý thuyết về hệ số xấp xỉ. Các kết quả đạt được mở rộng phạm vi nghiên cứu của tối ưu submodular sang các hệ thống có ràng buộc công bằng, góp phần giải quyết các yêu cầu ngày càng quan trọng trong các ứng dụng dữ liệu và mạng xã hội hiện đại.

Mặc dù đã đạt được các kết quả lý thuyết và thực nghiệm quan trọng, các nghiên cứu trong luận án vẫn còn một số hạn chế nhất định. Các thuật toán được đề xuất chủ yếu

được nghiên cứu trong các mô hình tối ưu tĩnh và chưa xem xét đầy đủ các đặc trưng của dữ liệu quy mô rất lớn hoặc dữ liệu thay đổi theo thời gian. Bên cạnh đó, các mô hình được khảo sát mới tập trung vào một số dạng ràng buộc chi phí và công bằng cơ bản, trong khi nhiều ứng dụng thực tế thường đòi hỏi các ràng buộc phức tạp hơn. Trong thời gian tới, các kết quả của luận án có thể được mở rộng theo một số hướng nghiên cứu chính sau.

Thứ nhất, phát triển các thuật toán tối ưu submodular và k -submodular trong các mô hình xử lý dữ liệu quy mô lớn như mô hình luồng (streaming), song song và phân tán. Đồng thời, nghiên cứu các kỹ thuật giảm số lần truy vấn hàm mục tiêu nhằm nâng cao khả năng áp dụng đối với các tập dữ liệu có kích thước rất lớn.

Thứ hai, mở rộng các mô hình tối ưu hiện tại sang các dạng ràng buộc phức tạp hơn, chẳng hạn như nhiều ràng buộc ngân sách đồng thời, các yêu cầu công bằng đa mức hoặc các ràng buộc kết hợp phát sinh trong các bài toán thực tế.

Thứ ba, nghiên cứu các bài toán tối ưu trong môi trường động và bất định, nơi dữ liệu, chi phí hoặc các tham số của hệ thống có thể thay đổi theo thời gian. Đây là hướng nghiên cứu có tiềm năng ứng dụng cao trong các hệ thống mạng xã hội, hệ thống khuyến nghị và các nền tảng dữ liệu hiện đại.

DANH MỤC CÔNG TRÌNH CÔNG BỐ LIÊN QUAN ĐẾN LUẬN ÁN

1. **Hue T. Nguyen**, Dung T.K. Ha, Canh V. Pham. *Efficient Parallel Algorithm for Minimum Cost Submodular Cover Problem with Lower Adaptive Complexity*. Asia-Pacific Journal of Operational Research , 41(6), 2450005 (2024) (SCIE).
2. **Hue T. Nguyen**, Nguyen Long Giang, Canh V. Pham. *Efficient Deterministic Bicriteria Approximation Algorithms for k -Submodular Cover Problem*. Asia-Pacific Journal of Operational Research, (2026) (SCIE).
3. **Hue T. Nguyen**, Nguyen Long Giang, Tran D. Tran, Canh V. Pham. *Thuật toán song song hiệu quả cho bài toán Phủ Submodular với chi phí tối thiểu và ứng dụng*. Kỷ yếu Hội thảo Quốc gia lần thứ XXVII Một số vấn đề chọn lọc về Công nghệ thông tin và Truyền thông, Tr. 622-634, 2024.
4. **Hue T. Nguyen**, Bac D Pham, Uyen T. Tran, Nguyen Long Giang, Canh V. Pham. *Influence Maximization with Fairness Allocation Constraint*. In: proceeding of the 13rd International Symposium on Information and Communication Technology (SoICT), 401-414 (SCOPUS).
5. **Hue T. Nguyen**, Bac D Pham, Dung TK Ha, Nguyen Long Giang, Canh V. Pham. *Influence Maximization with Fairness Cost on Groups in Online Social Networks*. In: proceeding of the 17th Asian Conference Intelligent Information and Database Systems (ACIIDS) 2025, Kitakyushu, Japan, April 23-25, 2025 (SCOPUS, Rank B).
6. **Hue T. Nguyen**, Tan D. Tran, Nguyen Long Giang, Canh V. Pham. *Fast Stochastic Greedy Algorithm for k -Submodular Cover Problem*. In: proceeding of the 14rd International Symposium on Information and Communication Technology (SoICT), pp 173–184 (SCOPUS).

DANH MỤC TÀI LIỆU THAM KHẢO

- [1] Alan Kuhnle, Victoria G. Crawford, My T. Thai, 2017, Scalable and Adaptive Algorithms for the Triangle Interdiction Problem on Billion-Scale Networks, *2017 IEEE International Conference on Data Mining, ICDM 2017, New Orleans, LA, USA, November 18-21, 2017*, 237–246, IEEE Computer Society.
- [2] Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein, 2009, *Introduction to Algorithms*, MIT Press, Cambridge, MA, 3 edition.
- [3] Madhav V Marathe, R Ravi, Ravi Sundaram, S.S Ravi, Daniel J Rosenkrantz, Harry B Hunt, 1998, Bicriteria Network Design Problems, *Journal of Algorithms*, 28, (1), 142–171.
- [4] Laurence A. Wolsey, 1982, An analysis of the greedy algorithm for the submodular set covering problem, *Comb.*, 2, (4), 385–393.
- [5] Amit Goyal, Francesco Bonchi, Laks V. S. Lakshmanan, Suresh Venkatasubramanian, 2013, On minimizing budget and time in influence propagation over social networks, *Social Network Analysis and Mining*, 3, (2), 179–192.
- [6] Yingli Ran, Zhao Zhang, Shaojie Tang, 2022, Improved Parallel Algorithm for Minimum Cost Submodular Cover Problem, *Proc. of the Learning Theory*, 2022, volume 178 of *Proceedings of MLS*, 3490–3502, PMLR.
- [7] Matthew Fahrbach, Vahab S. Mirrokni, Morteza Zadimoghaddam, 2019, Submodular Maximization with Nearly Optimal Approximation, Adaptivity and Query Complexity, *Proc. of SODA 2019*, 255–273.
- [8] Eric Balkanski, Aviad Rubinstein, Yaron Singer, 2019, An optimal approximation for submodular maximization under a matroid constraint in the adaptive complexity model, *Proc. of STOC 2019*, 66–77.
- [9] Ashwinkumar Badanidiyuru, Jan Vondrák, 2014, Fast algorithms for maximizing submodular functions, *Proceedings of the 2014 Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, 1497–1514.
- [10] Phuong N. H. Pham, Quy T. N. Co, Anh N. Su, Phuong T. Pham, Canh V. Pham, Václav Snásel, 2022, k-Submodular Cover Problem: Application and Algorithms, *The 11th International Symposium on Information and Communication Technology, SoICT 2022, Hanoi, Vietnam, December 1-3, 2022*, 442–449, ACM.

- [11] Wenqi Wang, Gregory Z. Gutin, Yaping Mao, Donglei Du, Xiaoyan Zhang, 2023, Streaming Algorithms for the k -Submodular Cover Problem, *CoRR*, abs/2312.03593.
- [12] Satoru Iwata, Shin-ichi Tanigawa, Yuichi Yoshida, 2016, Improved Approximation Algorithms for k -Submodular Function Maximization, Robert Krauthgamer, editor, *Proceedings of the 2016 Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, 404–413, SIAM.