

Tiếp cận Heuristic giải bài toán Clique lớn nhất trên đơn đồ thị vô hướng không có trọng số

Phan Thành Huân¹, Huỳnh Thị Châu Ái², Châu Lê Sa Lin³

¹ Đại học Quốc gia Tp. Hồ Chí Minh

² Khoa Kỹ thuật Công nghệ, Trường Đại học Văn Hiến

³ Khoa Công nghệ Thông tin - truyền thông, Trường Cao đẳng Kinh tế - Kỹ thuật Cần Thơ

Tác giả liên hệ: Phan Thành Huân, huanphan@hcmussh.edu.vn

Ngày nhận bài: 15/04/2021, ngày sửa chữa: 01/06/2021, ngày duyệt đăng: 12/06/2021

Định danh DOI: 10.32913/mic-ict-research.v2021.n1.964

Tóm tắt: Bài toán Clique lớn nhất (Maximum Clique Problem) là bài toán tìm tập con lớn nhất của tập đỉnh trong đơn đồ thị vô hướng, sao cho hai đỉnh phân biệt trong nó luôn kề nhau. Đây là bài toán nổi tiếng thuộc lớp NP-complete, được ứng dụng nhiều trong các lĩnh vực khai thác dữ liệu, phân tích mạng, truy xuất thông tin, y học, giáo dục và nhiều lĩnh vực khác liên quan đến mạng lưới toàn cầu. Có nhiều cách tiếp cận giải bài toán Clique lớn nhất như quy hoạch động, nhánh-cận, heuristic hay meta-heuristic – cho lời giải chính xác hay xấp xỉ. Trong bài báo này, nhóm tác giả phân tích hai thuật giải tiếp cận heuristic gần đây và đề xuất các heuristic tăng độ chính xác của lời giải cho bài toán Clique lớn nhất. Phần thực nghiệm, nhóm tác giả so sánh chất lượng lời giải của thuật giải đề xuất trên 10 bộ dữ liệu từ DIMACS.

Từ khóa: clique lớn nhất, tiếp cận Heuristic, NP-complete.

Title: Heuristic Approach for Solving the Maximum Clique Problem on Simple Undirected and Unweighted Graph

Abstract: The Maximum Clique Problem is the problem of finding the largest subset of vertices in a simple undirected graph, such that two distinct vertices in it are always adjacent. This is a well-known NP-complete problem, widely applied in the fields of data mining, network analysis, information retrieval, medicine, education and many other fields related to World Wide Web. There are many approaches to solving the Maximum Clique problem such as dynamic programming, branch and cut, heuristic or metaheuristic – for exact or approximate solutions. This article, the authors analyze two recent heuristic approaches and propose heuristics to increase the accuracy of the solution for the Maximum Clique problem. The experimental section, the authors compare the solution quality of the proposed algorithm on 10 datasets from the DIMACS.

Keywords: maximum clique, Heuristic, NP-complete.

I. GIỚI THIỆU

Bài toán clique lớn nhất (Maximum Clique Problem – MCP) là một bài toán kinh điển của lý thuyết đồ thị thuộc lớp NP-complete và có nhiều ứng dụng lĩnh vực khoa học máy tính, sinh học, tài chính,... được giải quyết thông qua bài toán tìm kiếm các Clique trong đồ thị. Trong ứng dụng thực tế, các bài toán thường được mô hình hoá bằng đồ thị rất lớn và cần phải có bộ nhớ lưu trữ đủ lớn cho quá trình thực hiện các thuật toán. Nhóm tác giả Đàm Thanh Phương và đồng sự [16] đã thảo luận những thách thức tính toán từ lý thuyết đến thực hành của bốn ứng dụng cụ thể - cho thấy khó khăn trong thiết kế các thuật toán phù hợp mang hiệu quả cao.

Trong vài năm trở lại đây, có nhiều công trình nghiên cứu về MCP và các ứng dụng không liên quan đến mạng lưới toàn cầu như: xác định các cấu trúc protein tương đồng [2] – sinh học; theo dõi đa mục tiêu [9] – thị giác máy tính; phân cụm điện não đồ [12] – y học; xếp hạng các trường đại học [13] – giáo dục;... Nhóm tác giả dựa vào công trình khảo sát [10], có thể phân loại các nghiên cứu giải bài toán MCP theo hai nhóm chiến lược tiếp cận:

Chiến lược tìm kiếm lời giải chính xác: Thuật toán quy hoạch động [1], thuật toán nhánh-cận [2, 4, 8]. Chiến lược này chỉ thích hợp thực hiện trên các đồ thị có kích thước nhỏ và cho thời gian thực thi là cao. Trong kỷ nguyên bùng nổ dữ liệu của các lĩnh vực – Đây thực sự là thách thức lớn

trong lĩnh vực nghiên cứu lý thuyết tối ưu tổ hợp. Chiến lược này còn là cơ sở cho đánh giá độ chính xác của các giải thuật cho lời giải gần đúng.

Chiến lược tìm kiếm lời giải gần đúng: Đây là giải pháp lựa chọn cho các bài toán có thời gian thực thi là cao và có thể chấp nhận lời giải gần đúng; chiến lược này thường dựa vào kinh nghiệm để đưa ra các heuristic cho từng bài toán hoặc dữ liệu cụ thể – không chắc hiệu quả cho tất cả; gọi là các giải thuật heuristic. Giải thuật heuristic làm giảm đáng kể thời gian thực thi – ưu điểm nổi bật; một số giải thuật heuristic giải bài toán MCP: [3, 5, 14, 18, 19], v.v.. Ngoài ra, chiến lược này còn sử dụng kết hợp nhiều ưu điểm của từng giải thuật heuristic – gọi là meta-heuristic. Gần đây cũng có nhiều nghiên cứu sử dụng các giải thuật meta-heuristic giải bài toán MCP – giải thuật bầy ong [11], giải thuật tối ưu đàn kiến NACO [6], KLS_improve [17], v.v..

Năm 2019, nhóm tác giả Phan Tấn Quốc và Huỳnh Thị Châu Ái đã đề xuất heuristic cải tiến thuật giải MAXCLIQUEHEU1 [15] và MAXCLIQUEHEU2 [5] cho lời giải chất lượng chưa tốt ở một số bộ dữ liệu từ hệ thống DIMACS (10 bộ dữ liệu ở Bảng 2). Vì vậy, nhóm tác giả chọn nghiên cứu đề xuất các heuristic hiệu quả cho việc gia tăng chất lượng của lời giải.

Trong nghiên cứu này, nhóm tác giả tập trung phân tích hai giải thuật heuristic [18] và đề xuất các heuristic hiệu quả làm gia tăng chất lượng lời giải cho bài toán MCP dựa trên các tính toán thống kê phù hợp. Trong Phần 2, trình bày các khái niệm cơ bản về bài toán Clique lớn nhất và phân tích một số thuật giải gần đây. Phần 3, trình bày thuật giải đề xuất MCP-HEU* xác định nhanh Clique lớn nhất của đơn đồ thị vô hướng không có trọng số. Kết quả thực nghiệm được trình bày trong Phần 4 và kết luận ở Phần 5.

II. KHẢO SÁT MỘT SỐ HEURISTIC GIẢI BÀI TOÁN CLIQUE LỚN NHẤT

Trong phần này, nhóm tác giả trình bày các khái niệm liên quan bài toán Clique lớn nhất và đồng thời phân tích 2 giải thuật gần đây của nhóm tác giả Phan Tấn Quốc và Huỳnh Thị Châu Ái [18, 19].

1. Bài toán Clique lớn nhất

Cho $G = (V, E)$ là đơn đồ thị vô hướng không có trọng số, có V là tập chứa các đỉnh của đồ thị, E là tập chứa các cạnh trong đồ thị. Tập các đỉnh liền kề với đỉnh v_i , ký hiệu $\text{adj}(v_i)$; bậc của đỉnh v_i là lực lượng của $\text{adj}(v_i)$, ký hiệu $\text{deg}(v_i)$ và $\text{deg}(v_i) = |\text{adj}(v_i)|$.

Định nghĩa 1: (Complete Graph) Đồ thị đầy đủ n đỉnh là đơn đồ thị vô hướng mà giữa 2 đỉnh bất kỳ trên đồ thị luôn có cạnh nối, ký hiệu K_n .

Định nghĩa 2: (Clique) $\forall K_{n \geq 1} \subseteq G = (V, E)$, các tập đỉnh của đồ thị $K_{n \geq 1}$ được gọi là các Clique của đồ thị G . Ký hiệu, C_G là một Clique của đồ thị G .

Tính chất 1: Số lượng đỉnh (kích thước) của một Clique C của đồ thị G , ký hiệu là $|C|$. Nếu $v_i \in |C|$ thì $|C| < \text{deg}_G(v_i) + 1$.

Định nghĩa 3: (Maximal Clique - Clique cực đại) C_G là một Clique của đồ thị G được gọi là một Clique cực đại, nếu không tồn tại $C'_G \supset C_G$ và $|C'_G| \geq |C_G|$. Ký hiệu, C_{Maximal} là Clique cực đại của đồ thị G .

Định nghĩa 4: (Maximum Clique - Clique lớn nhất) C_G là một Clique của đồ thị G được gọi là một Clique lớn nhất, nếu không tồn tại $C'_G \supset C_G$ và $|C_G| \geq |C'_G|$. Ký hiệu, C_{Maximum} là Clique lớn nhất của đồ thị G .

Định nghĩa 5: (Clique number - Chỉ số Clique) là số lượng đỉnh của C_{Maximum} Clique lớn nhất của đồ thị G . Chỉ số Clique của đồ thị G , ký hiệu là $\omega(G)$.

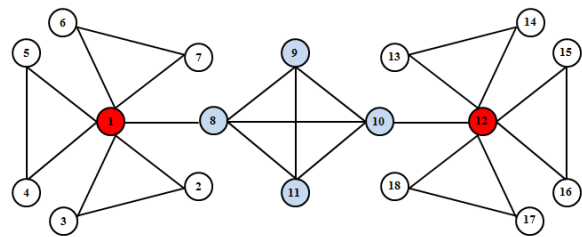
Tính chất 2: (dựa vào định nghĩa 3 và 4) Một Clique lớn nhất là một Clique cực đại, nhưng một Clique cực đại chưa chắc đã là một Clique lớn nhất.

Tính chất 3: (dựa vào định nghĩa 4) Đồ thị có thể có nhiều Clique có kích thước bằng Clique lớn nhất.

Bài toán tìm Clique lớn nhất

Cho $G = (V, E)$ là đơn đồ thị vô hướng không có trọng số. Bài toán tìm Clique lớn nhất (Maximum Clique Problem - MCP) là bài toán tìm một Clique lớn nhất (theo định nghĩa 4) trong đồ thị đã cho.

Cho G là đơn đồ thị vô hướng không có trọng số, có 18 đỉnh và 26 cạnh như ở Hình 1.



Hình 1. Đơn đồ thị vô hướng G cho ví dụ

Bảng I, cho thấy đồ thị G có số lượng phần tử của tập Clique, Clique cực đại và Clique lớn nhất lần lượt là 37, 9 và 1 – số lượng các Clique rất lớn so với Clique cực đại và Clique lớn nhất. Đồng thời cho thấy Clique lớn nhất của đồ thị là $\{8, 9, 10, 11\}$ và chỉ số Clique $\omega(G) = 4$.

2. Một số giải thuật giải bài toán Clique lớn nhất

Năm 2019, nhóm tác giả Phan Tấn Quốc cùng đồng sự đã cải tiến thuật toán MAXCLIQUEHEU1 [15] của nhóm tác giả Vũ Đình Hòa và MAXCLIQUEHEU2 [5] của

Bảng I
TẬP CLIQUE, CLIQUE CỰC ĐẠI VÀ CLIQUE LỚN NHẤT TRÊN ĐỒ THỊ G.

Tập Clique (#C=37)	Tập Clique cực đại (#C _{Maximal} =9)	Tập Clique lớn nhất (#C _{Maximum} =1)
{1, 2}, {1, 3}, {1, 4}, {1, 5}, {1, 6}, {1, 7}, {1, 8}, {2, 3}, {4, 5}, {6, 7}, {8, 9}, {8, 10}, {8, 11}, {9, 10}, {9, 11}, {10, 11}, {10, 12}, {12, 13}, {12, 14}, {12, 15}, {12, 16}, {12, 17}, {12, 18}, {13, 14}, {15, 16}, {17, 18}, {1, 2, 3}, {1, 4, 5}, {1, 6, 7}, {8, 9, 10}, {8, 9, 11}, {8, 10, 11}, {9, 10, 11}, {12, 13, 14}, {12, 15, 16}, {12, 17, 18}, {8, 9, 10, 11}	{1, 8}, {10, 12}, {1, 2, 3}, {1, 4, 5}, {1, 6, 7}, {12, 13, 14}, {12, 15, 16}, {12, 17, 18}, {8, 9, 10, 11}	{8, 9, 10, 11}

Algorithm 1: HEU1 - xác định kích thước Clique lớn nhất của đồ thị.

```

1 Inputs: Đồ thị  $G = (V, E)$ .
2 Outputs: Kích thước Clique lớn nhất -  $\omega_G$ .
3 begin
4    $\omega_{Best} = 0$ ;
5    $C_{Maximum} = \{\emptyset\}$ ;
6   while điều kiện dừng chưa thỏa do
7      $U = \{\emptyset\}$ ;
8     while  $V \neq \{\emptyset\}$  do
9        $S = \{k \text{ đỉnh thuộc } V \text{ có bậc lớn nhất}\}$ ;
10      Chọn ngẫu nhiên đỉnh  $v \in S$ ;
11       $U = U \cup \{v\}$ ;
12       $V = V - \{v\}$ ;
13      Xóa các đỉnh không kề với đỉnh  $v$  ra khỏi  $V$ ;
14      Cập nhật bậc các đỉnh trong  $V$ 
15    end
16    if  $|U| \geq \omega_{Best}$  then
17       $\omega_{Best} = |U|$ ;
18      Cập nhật  $C_{Maximum} = U$ ;
19    end
20  end
21  return  $\omega_{Best}$ ;
22 end

```

nhóm tác giả Pattabiraman cùng đồng sự; thuật toán cải tiến lần lượt có tên gọi là MAXCLIQUEHEU1_improve và MAXCLIQUEHEU2_improve [18] – để thuận tiện trong việc phân tích, chúng tôi sử dụng tên tắt cho 2 thuật toán cải tiến trên là HEU1 và HEU2. Các giải thuật cải tiến HEU1, HEU2 được trình bày:

Phần mã giả thuật giải HEU1 cải tiến từ MAXCLIQUE-HEU1 [15] - Thuật giải 1

Trong giải thuật HEU1, nhóm tác giả đã bổ sung dòng 9 và 10 – chọn k đỉnh thuộc V có bậc lớn nhất (dòng 9), sử dụng chiến lược chọn ngẫu nhiên (dòng 10) và giải thuật được lặp lại nhiều lần (điều kiện dừng – dòng 6) cho việc chọn lời giải tốt hơn.

Ưu điểm: Dễ hiểu, đơn giản khi sử dụng chiến lược tham lam – “đỉnh có bậc lớn nhất thì có nhiều khả năng là đỉnh thuộc Clique lớn nhất”;

Nhược điểm: Chọn tập S gồm k đỉnh có bậc lớn nhất – cần xác định k là bao nhiêu cho phù hợp từng bộ dữ liệu,

Algorithm 2: HEU2 - xác định kích thước Clique lớn nhất của đồ thị.

```

1 Inputs: Đồ thị  $G = (V, E)$ .
2 Outputs: Kích thước Clique lớn nhất -  $\omega_G$ .
3 begin
4    $\omega_{Best} = 0$ ;
5    $C_{Maximum} = \{\emptyset\}$ ;
6   while điều kiện dừng chưa thỏa do
7     Chọn ngẫu nhiên đỉnh  $v \in S$ ;
8     if  $\deg(v_i) \geq \max$  then
9        $U = \{\emptyset\}$ ;
10      foreach  $v_j \in \text{adj}(v_i)$  do
11        if  $\deg(v_j) \geq \max$  then
12           $U = U \cup \{v_j\}$ ;
13        end
14      while  $V \neq \{\emptyset\}$  do
15         $S = \{k \text{ đỉnh } V \text{ có bậc lớn nhất từ } V\}$ ;
16        Chọn ngẫu nhiên đỉnh  $v \in S$ ;
17         $U = U - \{v\}$ ;
18        Chỉ giữ lại đỉnh  $u \in U$ : ( $u$  kề với đỉnh  $v$  và  $\deg(u) \geq \max$ );
19         $\max = \max + 1$ ;
20      end
21    end
22    if  $|U| \geq \max$  then
23       $\max = |U|$ ;
24    end
25    if  $\max \geq \omega_{Best}$  then
26       $\omega_{Best} = \max$ ;
27      Cập nhật  $C_{Maximum} = U$ ;
28    end
29  end
30  return  $\omega_{Best}$ ;
31 end

```

điều này cho ta thấy chất lượng của lời giải phụ thuộc vào tập đỉnh có bậc lớn nhất được lựa chọn.

Phần mã giả thuật giải HEU2 cải tiến từ MAXCLIQUE-HEU2 [5] - Thuật giải 2

Tương tự, ở giải thuật HEU2, nhóm tác giả đã bổ sung dòng 16 – chọn k đỉnh thuộc U có bậc lớn nhất và chiến lược chọn ngẫu nhiên (dòng 16) và giải thuật được lặp lại nhiều lần (điều kiện dừng – dòng 6) cho việc chọn lời giải tốt hơn. Phần ưu và nhược điểm cũng tương tự như ở thuật

giải HEU1.

Trong cả 2 giải thuật HEU1 và HEU2, nhóm tác giả đã lựa chọn $k = 10\%$ đỉnh có bậc lớn nhất cùng chiến lược chọn ngẫu nhiên và số lần lặp lại là 30 lần (điều kiện dừng) – chất lượng của lời giải phụ thuộc vào k .

Phần thực nghiệm của cả 2 giải thuật HEU1 và HEU2 trên 37 bộ dữ liệu từ hệ thống thách thức DIMACS cho chất lượng của lời giải tốt hơn cả 2 giải thuật MAXCLIQUE-HEU1 và MAXCLIQUEHEU2. Tuy nhiên, chất lượng lời giải từ HEU1 và HEU2 chỉ đạt từ 69,00% đến 100,00% và từ 72,00% đến 100,00% so với kết quả tối ưu (ω_{Best}).

Bảng IV, các đỉnh của đơn đồ thị G vô hướng không có trọng số được sắp xếp giảm dần theo bậc.

Bảng II, mô tả kết quả thực nghiệm trên 10 bộ dữ liệu từ hệ thống DIMACS (10 bộ dữ liệu cho lời giải có chất lượng thấp trong 37 bộ dữ liệu) - gồm các thông tin sau: Tên bộ dữ liệu, $|V|$ là số đỉnh, $|E|$ là số cạnh, deg_{min} là bậc cực tiểu, deg_{max} là bậc cực đại, median là giá trị trung vị của bậc các đỉnh, iqr là độ trải giữa của bậc các đỉnh ($Q3-Q1$), ω_{Best} là chỉ số Clique tốt nhất hiện nay [7], ω_{Heu1} và ω_{Heu2} .

Minh họa thuật giải HEU1 theo đồ thị G ở Hình 1.

Bảng III, bậc của các đỉnh trên đơn đồ thị G vô hướng không có trọng số theo thứ tự các đỉnh.

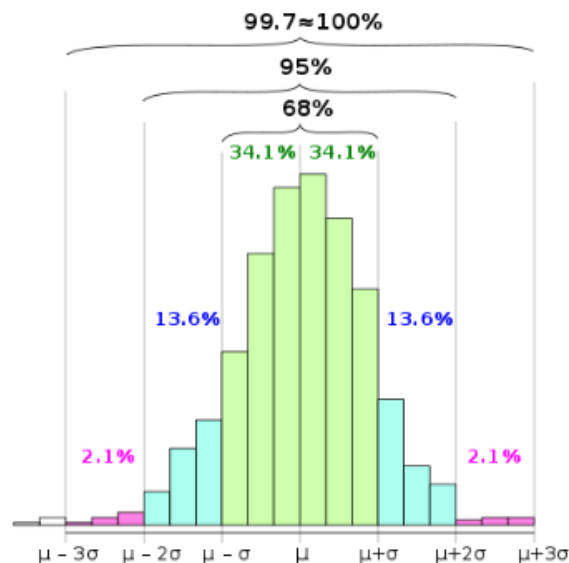
Từ Bảng III, thuật giải HEU1 và HEU2 chọn $k = 10\%$ đỉnh có bậc cao nhất trên đồ thị G. Khi đó, tập $S = \{1, 12\}$: giả sử chọn ngẫu nhiên đỉnh 1 – kết quả trả về chỉ số Clique $\omega_{Best} = 3$; dựa vào Bảng I, ta thấy với $\omega_{Best} = 3$ chưa phải là lời giải tốt ($\omega(G) = 4$). Từ minh họa và phân tích trên, nhóm tác giả chỉ ra rằng khi chọn $k = 10\%$ đỉnh có bậc cao nhất và chọn ngẫu nhiên – đây chỉ là các lựa chọn mang tính chủ quan của nhóm tác giả [18]. Qua đó, cho thấy để giải bài toán Clique lớn nhất cho lời giải chất lượng tốt không chỉ dựa vào yếu tố ngẫu nhiên và cảm tính. Rất cần thiết cho việc đề xuất heuristic phù hợp hơn cho bài toán tìm Clique lớn nhất.

III. ĐỀ XUẤT HEURISTIC GIẢI BÀI TOÁN CLIQUE LỚN NHẤT

Phần đề xuất thuật giải hiệu quả cho bài toán tìm Clique lớn nhất, nhóm tác giả dựa trên nền tảng thống kê và phân tích đề xuất các heuristic phù hợp cho bài toán tìm Clique lớn nhất.

1. Một số đại lượng mô tả sự phân bố của dữ liệu trong thống kê

Quy tắc ba Sigma 68-95-99,7 (Empirical Rule). Đây là quy tắc đã kiểm chứng thường được sử dụng trong thống kê để dự báo các kết quả cuối cùng.

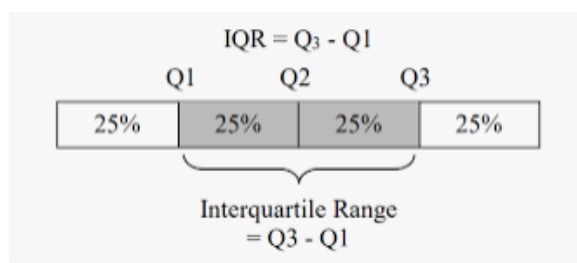


Hình 2. Quy tắc ba Sigma - giá trị trung bình μ và độ lệch chuẩn σ

Hình 2, theo quy tắc ba Sigma có 68% quan sát nằm trong độ lệch chuẩn đầu tiên ($\mu \pm \sigma$), 95% quan sát nằm trong hai độ lệch chuẩn đầu tiên ($\mu \pm 2\sigma$) và 99,7% nằm trong ba độ lệch chuẩn đầu tiên ($\mu \pm 3\sigma$). Ngoài ra, quy tắc ba Sigma còn cho thấy 15,70% các giá trị lớn nhất thuộc $[\mu + \sigma, \mu + 3\sigma]$.

Hình 3, minh họa các phân vị (Quartiles) và độ trải giữa (Interquartile Range - IQR). Các phân vị là đại lượng mô tả sự phân bố và sự phân tán của tập dữ liệu, gồm có 3 giá trị: phân vị thứ nhất ($Q1$), thứ hai ($Q2$) và thứ ba ($Q3$) – chia tập dữ liệu thành 4 phần có số lượng quan sát đều nhau; độ trải giữa của tập dữ liệu là hiệu số giữa phân vị thứ ba và thứ nhất, công thức tính theo phương trình sau:

$$IQR = Q3 - Q1 \quad (1)$$



Hình 3. Các phân vị (Quartiles) và độ trải giữa (InterQuartile Range)

Một vài quan sát điển hình từ Bảng II: ta thấy bộ dữ liệu "hamming10-4" có 1.024 đỉnh và các đỉnh có bậc bằng nhau 848, lời giải của cả 2 giải thuật đều cho chất lượng

Bảng II
KẾT QUẢ THỰC NGHIỆM CỦA HEU1 VÀ HEU2 TRÊN 10 BỘ DỮ LIỆU TỪ DIMACS

Dữ liệu	$ V $	$ E $	deg_{min}	deg_{max}	median	iqr	ω_{Best}	ω_{Heu1}	ω_{Heu2}
brock400 2	400	59.786	274	328	299	10,00	29	24	24
brock800 2	800	208.166	472	566	512	18,00	24	19	20
brock800 4	800	207.643	481	565	519	18,25	26	19	19
C2000.9	2.000	1.799.532	1.751	1.848	1.800	18,00	80	66	62
C4000.5	4.000	4.000.268	1.895	2.123	2.001	42,00	18	16	16
gen400 p0.9 55	400	71.820	334	375	360	13,25	55	50	50
gen400 p0.9 65	400	71.820	333	378	361	14,00	65	48	50
gen400 p0.9 75	400	71.820	335	380	359	13,00	75	52	54
hamming10-4	1.024	434.176	848	848	848	0,00	40	34	33
keller6	3.361	4.619.898	2.690	2.952	2.724	50,00	59	43	49

Bảng III
BẬC CỦA CÁC ĐỈNH TRÊN ĐỒ THỊ G THEO THỨ TỰ CÁC ĐỈNH.

Đỉnh	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
Bậc	7	2	2	2	2	2	2	4	3	4	3	7	2	2	2	2	2	2

Bảng IV
CÁC ĐỈNH TRÊN ĐỒ THỊ G ĐƯỢC SẮP XẾP GIẢM DẦN THEO BẬC.

Đỉnh	1	12	8	10	9	11	2	3	4	5	6	7	13	14	15	16	17	18
Bậc	7	7	4	4	3	3	2	2	2	2	2	2	2	2	2	2	2	2

Bảng V
KẾT QUẢ THỰC NGHIỆM TRÊN 10 BỘ DỮ LIỆU TỪ DIMACS

Dữ liệu	$ V $	$ E $	deg_{min}	deg_{max}	ω_{Best}	ω_{Heu1}	ω_{Heu2}	ω_{Heu1*}	ω_{Heu2*}
brock400 2	400	59.786	274	328	29	24	24	26	28
brock800 2	800	208.166	472	566	24	19	20	22	24
brock800 4	800	207.643	481	565	26	19	19	23	24
C2000.9	2.000	1.799.532	1.751	1.848	80	66	62	75	77
C4000.5	4.000	4.000.268	1.895	2.123	18	16	16	17	18
gen400 p0.9 55	400	71.820	334	375	55	50	50	52	53
gen400 p0.9 65	400	71.820	333	378	65	48	50	61	63
gen400 p0.9 75	400	71.820	335	380	75	52	54	67	71
hamming10-4	1.024	434.176	848	848	40	34	33	36	38
keller6	3.361	4.619.898	2.690	2.952	59	43	49	53	57

xấp xỉ nhau 85% và 82,25% ; bộ dữ liệu "gen400 p0.9 75" có 400 đỉnh và độ trải là 55 (Range = $deg_{min} - deg_{max} = 380 - 335 = 55$) cho thấy các đỉnh tương đồng rất nhiều – cả 2 giải thuật đều cho lời giải chất lượng thấp 69,33% và 72%.

2. Thuật giải HEU* cho bài toán tìm Clique lớn nhất

Trong phần này, nhóm tác giả trình bày thuật giải đề xuất tìm Clique lớn nhất dựa vào một số heuristic đề xuất như sau:

Heuristic_0: (dựa vào tính chất 1) Điều kiện ngắt trong quá trình tìm Clique lớn nhất, nếu chỉ số Clique $\omega_{Best} >$

Algorithm 3: MCP-HEU* - xác định kích thước Clique lớn nhất của đồ thị.

```

1 Inputs: Đồ thị  $G = (V, E)$ .
2 Outputs: Kích thước Clique lớn nhất -  $\omega_G$ .
3 begin
4    $\omega_{Best} = 0$ ;
5   Chọn tập  $S$  theo heuristic_1 hoặc heuristic_2;
6   foreach  $v_i \in S$  do
7      $U = \{\emptyset\}$ ;
8     foreach  $v_j \in adj(v_i)$  do
9       If  $deg(v_j) \geq \omega_{Best}$   $U = U \cup \{v_j\}$ ;
10      while  $V \neq \{\emptyset\}$  do
11        Chọn  $v \in U$  có bậc lớn nhất;
12         $U = U - \{v\}$ ;
13        Chỉ giữ lại đỉnh  $u \in U$ : ( $u$  kề với đỉnh  $v$ 
        và  $deg(u) \geq \omega_{Best}$ );
14         $\omega_{Best} = \omega_{Best} + 1$ ;
15      end
16      if  $|U| \geq \omega_{Best}$  then
17         $\omega_{Best} = |U|$ 
18      end
19    end
20  end
21  return  $\omega_{Best}$ ;
22 end

```

$deg(v_i) + 1$ (v_i là các đỉnh còn lại trong tập S).

Dựa vào quy tắc ba Sigma, nhóm tác giả đề xuất heuristic sau:

Heuristic_1: Chọn $k = 16\%$ (làm tròn từ 15,70%) đỉnh có bậc lớn nhất - nghĩa là chọn từ đỉnh có bậc lớn nhất đến đỉnh có bậc xấp xỉ giá trị $(\mu + \sigma)$ được làm tròn xuống (lấy phần nguyên);

Ví dụ 1: theo đồ thị G có 18 đỉnh và 26 cạnh, với $k = 16\%$ tương ứng tập S có 3 đỉnh từ đỉnh có bậc lớn nhất đến đỉnh có bậc xấp xỉ $(\mu + \sigma) = 4$ ($\mu = 2,89; \sigma = 1,64$).

Tương tự, dựa vào giá trị trung vị median và độ trải giữa iqr (xác định $Q1, Q2, Q3$) - nhóm tác giả đề xuất heuristic cho việc chọn đỉnh tiềm năng trong tập S :

Heuristic_2: Chọn $k = 16\%$ đỉnh có bậc lớn nhất theo tỷ lệ 8 : 5 : 3 trên từng khoảng phân vị - chọn ngẫu nhiên 8% đỉnh có bậc thuộc $[Q3, degmax]$; 5% thuộc $[Q2, Q3]$ và 3% thuộc $[Q1, Q2]$. Tỷ lệ 8 : 5 : 3 có thể tùy biến.

Ví dụ 2: theo đồ thị G có 18 đỉnh và 26 cạnh, median = 2, iqr = 1 ($Q1=2, Q2=2, Q3=3$), degmin = 1, degmax = 7. Khi đó, chọn ngẫu nhiên 8% đỉnh có bậc thuộc $[3, 7]$; 5% đỉnh có bậc thuộc $[2, 3]$ và 3% đỉnh có bậc thuộc $[2, 2]$ - kết quả nhận được tổng số đỉnh thuộc tập S có thể lớn hơn $k = 16\%$ do làm tròn (8% chọn 2 đỉnh; 5% chọn 1 đỉnh và 3% chọn 1 đỉnh - có 4 đỉnh được chọn cho tập S).

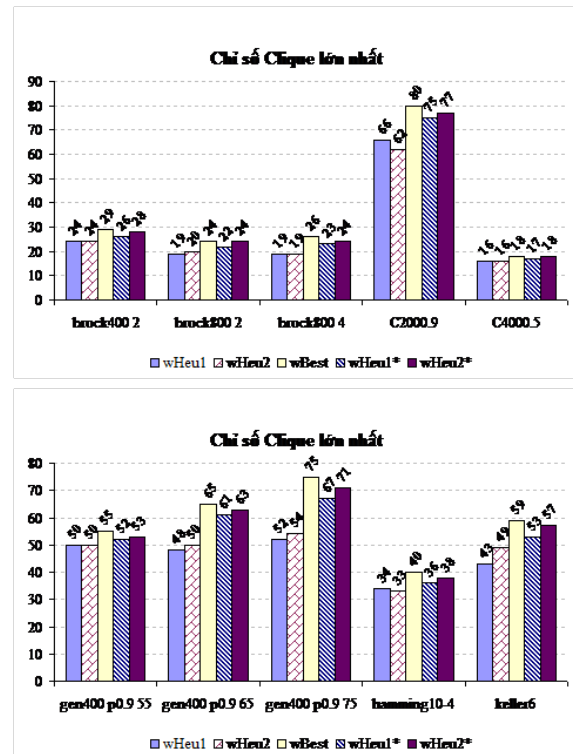
Trong nghiên cứu này, nhóm tác giả chọn giải thuật HEU2 cho thực nghiệm cải tiến thay thế các heuristic vừa trình bày ở trên.

Thuật giải MCP-HEU1* là thuật giải ở dòng 2 sử dụng heuristic_1; thuật giải MCP-HEU2* là thuật giải dùng heuristic_2 ở dòng 2. Ở cả 2 thuật giải, tập S được chọn trước và đây là tập chứa các đỉnh tiềm năng thuộc Clique lớn nhất.

IV. THỰC NGHIỆM VÀ ĐÁNH GIÁ

Thực nghiệm trên máy tính Core i7-3540M CPU 3.0 GHz, 4Gb RAM, thuật toán cài đặt trên C#, Microsoft Visual Studio 2015. Dữ liệu thực nghiệm. Dữ liệu được chọn từ hệ thống DIMACS cho thách thức giải bài toán Clique lớn nhất gồm có 37 bộ dữ liệu [7]. Tuy nhiên, nhóm tác giả chỉ chọn 10 bộ dữ liệu (giải thuật HEU1_improve, HEU2_improve cho lời giải chất lượng chưa tốt, xấp xỉ 70%) theo Bảng V cho phần thực nghiệm so sánh độ chính xác cùng với 2 giải thuật HEU1_improve, HEU2_improve [18] - thuận tiện cho việc so sánh kết quả thực nghiệm, nhóm tác giả viết tắt tên của 2 giải thuật trên là HEU1 và HEU2.

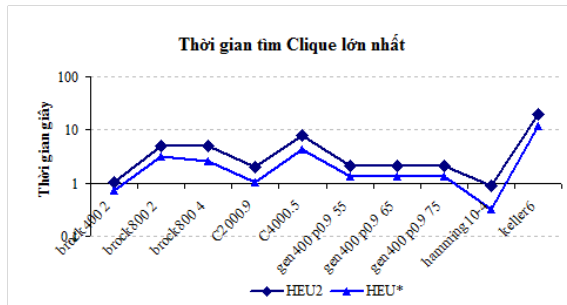
Bảng V, mô tả 10 bộ dữ liệu thực nghiệm từ hệ thống DIMACS - gồm các thông tin sau: Tên bộ dữ liệu, $|V|$ là số đỉnh, $|E|$ là số cạnh, degmin là bậc cực tiểu, degmax là bậc cực đại, ω_{Best} là chỉ số Clique tốt nhất hiện nay [7], ω_{Heu1} , ω_{Heu2} , ω_{Heu1*} và ω_{Heu2*} .



Hình 4. Chỉ số Clique của các thuật giải trên 10 bộ dữ liệu từ DIMACS

Hình 4, cho thấy chất lượng lời giải của 2 thuật giải HEU1* và HEU2* từ 88,46% đến 100% tốt hơn HEU1,

HEU2. Ngoài ra, thuật toán cũng cần thực nghiệm thêm trên nhiều bộ dữ liệu có mật độ khác nhau, cũng như trên nhiều dữ liệu cỡ lớn.



Hình 5. Thời gian (giây) tìm MCP của các thuật giải trên 10 bộ dữ liệu

Hình 5, cho thấy thời gian xác định chỉ số Clique của thuật giải HEU* nhanh hơn HEU2 rất nhiều – do heuristic_1 và heuristic_2 được chọn trước với tỷ lệ phù hợp trên các khoảng.

Tóm lại, các heuristic được đề xuất mang lại hiệu quả cao về chất lượng lời giải và thời gian thực hiện nhanh hơn các giải thuật gần đây.

V. KẾT LUẬN

Nhóm tác giả đã phân tích ưu và nhược điểm của một số thuật giải gần đây và đề xuất các heuristic tìm Clique lớn nhất với chất lượng lời giải tốt hơn. Tỷ lệ ở heuristic_2 có thể điều chỉnh tùy biến và mở rộng trên bốn khoảng phân vị.

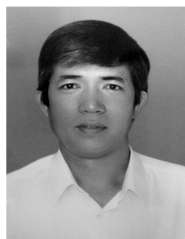
Dựa trên kết quả đạt được từ thực nghiệm: Tương lai, nhóm tác giả mở rộng giải thuật để có thể giải bài toán Clique lớn nhất trên đồ thị vô hướng có trọng số cạnh và đỉnh; đồng thời nghiên cứu đề xuất các heuristic cũng như kết hợp các heuristic nhằm nâng cao chất lượng lời giải.

TÀI LIỆU THAM KHẢO

- [1] W. Myrvold, P. Tania, W. Neil, "A Dynamic Programming Approach for Timing and Designing Clique Algorithms", *Proc of ALEX98*, 88-95, 1998.
- [2] Strickland D. M., E. Barnes, J. Sokol, "Optimal Protein Structure Alignment Using Maximum Cliques", *Oper. Res.*, 53, 389-402, 2005.
- [3] D. Kumlander, "Comparing the best maximum clique finding algorithms, which are using heuristic vertex colouring", *10th WSEAS Inter Conf on COMPUTERS*, 932-937, 2006.
- [4] E. Tomita, Y. Sutani, T. Higashi, M. Wakatsuki, "A simple and faster branch-and-bound algorithm for finding a maximum clique with computational experiments", *IEICE Trans. Inf. Syst.*, 96(6), 1286-1298, 2013.
- [5] B. Pattabiraman, M. M. A. Patwary, A. H. Gebremedhin, W. Liao, A. Choudhary, "Fast Algorithms for the Maximum Clique Problem on Massive Graphs with Applications to Overlapping Community Detection. Internet Mathematics", 11(4-5), 421-448, 2015.
- [6] K. Schiff, "An Ant Algorithm for the Maximum Clique Problem in a Special Kind of Graph. Journal of Automation", *Mobile Robotics and Intelligent Systems* 9, 20-23, 2015.
- [7] http://iridia.ulb.ac.be/fmascia/maximum_clique/DIMACS-benchmark, 2015.
- [8] P.S. Segundo, A. Lopez, P.M. Pardalos, "A new exact maximum clique algorithm for large and massive sparse graphs", *Comput. Oper. Res.*, 66, 81-94, 2016.
- [9] C. Li, C. Di, S. Hao, C. Jiahui, Z. Yang, "Heuristic Maximum Clique Based Identity Switches Awareness for Tracking", *Inter Conf ACAAI 2018*, 129-133, 2018.
- [10] F. Fakhfakh, M. Tounsi, M. Mosbah, K. A. Hadj, "Algorithms for Finding Maximal and Maximum Cliques: A Survey", In: *Abraham A., Muhuri P., Mada A., Gandhi N. (eds). ISDA 2017, AISC*, 736, 745-754, 2018.
- [11] S. Fotoohi, S. Saeidi, "Discovering the Maximum Clique in Social Networks Using Artificial Bee Colony Optimization Method", *IJITCS* 11, 1-11, 2019.
- [12] C. Dai, J. Wu, D. Pi, S. Becker, C. Lin, Q. Zhang, B. Johnson, "Brain EEG Time-Series Clustering Using Maximum-Weight Clique", *IEEE transactions on cybernetics*, 1-15, 2020.
- [13] M. O., Edwin, R. A. Mora-Gutiérrez, B. Obregón-Quintana, S. de-los-Cobos-Silva, E. A. Rincón-García, P. Lara-Velázquez and M. A. G. Andrade, "Mexican University Ranking Based on Maximal Clique", 327-395, 2020.
- [14] V. Balash, A. Stepanova, V. Daniil, S. Mironov, F. Alexey, S. Sidorov, "Testing a Heuristic Algorithm for Finding a Maximum Clique on DIMACS and Facebook Graphs", *Wseas Transactions on Systems and Control*, 15, 93-101, 2020.
- [15] Vũ Đình Hòa, Đỗ Trung Kiên, "Thuật toán song song giải bài toán xác định clique cực đại trên đồ thị", *Hội thảo Quốc gia: Một số vấn đề chọn lọc của Công nghệ thông tin và truyền thông*, 426-442, 2009.
- [16] Đàm Thanh Phương, Ngô Mạnh Tường, Khoa Thu Hoài, "Bài toán clique lớn nhất - ứng dụng và những thách thức tính toán", *Tạp chí KHCN - Chuyên san Khoa học Tự nhiên - Kỹ thuật, Đại học Thái Nguyên*, ISSN: 1859-2171, Tập 102, Số 2, 13-17, 2013.
- [17] Huỳnh Thanh Tân, Nguyễn Văn Thành, "Đề xuất thuật toán local search giải bài toán max clique", *Hội nghị KHCN Quốc gia lần thứ X về Nghiên cứu cơ bản và ứng dụng Công nghệ thông tin (FAIR)*, 148-155, 2017.
- [18] Phan Tấn Quốc, Huỳnh Thị Châu Ái, "Cải tiến một số thuật toán heuristic giải bài toán clique lớn nhất", *Hội nghị KHCN Quốc gia lần thứ XII về Nghiên cứu cơ bản và ứng dụng Công nghệ thông tin (FAIR)*, 64-72, 2019.
- [19] Huỳnh Thị Châu Ái, "Đề xuất thuật toán giải gần đúng cho bài toán clique lớn nhất", *Luận văn Thạc sỹ chuyên ngành Khoa học Máy tính, Trường Đại học Sài Gòn*, 2019.

SƠ LƯỢC VỀ TÁC GIẢ

Phan Thành Huân



Nhận bằng Thạc sỹ Đảm bảo Toán học cho Máy tính và Hệ thống tính toán năm 2016 tại trường Đại học Khoa học Tự nhiên, ĐHQG Tp.HCM. Hiện công tác tại trường Đại học Khoa học Xã hội và Nhân văn, ĐHQG Tp.HCM. Lĩnh vực nghiên cứu: Trí tuệ nhân tạo, khai thác dữ liệu, tính toán hiệu năng cao.
Email: huanphan@hcmussh.edu.vn

Huỳnh Thị Châu Ái



Nhận bằng Thạc sỹ Khoa học máy tính năm 2020 tại Trường Đại học Sài Gòn Tp.HCM. Hiện công tác tại Khoa Kỹ thuật Công nghệ, Trường Đại học Văn Hiến. Lĩnh vực nghiên cứu: Tính toán hiệu năng cao, giải thuật tối ưu.
Email: aihtc@vhu.edu.vn
Điện thoại: 0983 764 768

Châu Lê Sa Lin



Nhận bằng Thạc sỹ Hệ thống thông tin năm 2017 tại Trường Đại học Cần Thơ. Hiện công tác tại Khoa Công nghệ thông tin-truyền thông, Trường Cao đẳng Kinh tế - Kỹ thuật Cần Thơ. Lĩnh vực nghiên cứu: Tính toán hiệu năng cao, giải thuật tối ưu.
Email: clsalin@ctec.edu.vn
Điện thoại: 0945 017 660