ĐẠI HỌC QUỐC GIA HÀ NỘI
TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN

o0o

Nguyễn Kiều Linh

BÀI TOÁN TÌM BAO LỒI CỦA TẬP ĐIỂM HỮU HẠN VÀ ỨNG DỤNG

Chuyên ngành: Toán ứng dụng
Mã số: 62 46 01 12

DỰ THẢO TÓM TẮT LUẬN ÁN TIẾN SĨ TOÁN HỌC

Hà Nội - 2018

Công trình được hoàn thành tại:

TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN - ĐẠI HỌC QUỐC GIA HÀ NỘI

Tập thể hướng dẫn khoa học:

HD1: TS. HOÀNG NAM DŨNG
HD2: PGS. TS. PHAN THÀNH AN

Phản biện ....

Phản biện ....

Phản biện ....

Luận án sẽ được bảo vệ trước Hội đồng cấp Đại học Quốc gia chấm luận án tiến sĩ họp tại: TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN - ĐẠI HỌC QUỐC GIA HÀ NỘI

Vào hồi ... giờ ... ngày ... tháng ... năm 20......

Có thể tìm hiểu luận án tại:

- Thư viện Quốc gia Việt Nam
- Trung tâm Thông tin - Thư viện, Đại học Quốc gia Hà Nội.

Mở đầu

Bài toán tìm bao lồi của tập hữu hạn điểm là một trong những bài toán đặc biệt quan trọng trong lĩnh vực hình học tính toán bởi các ứng dụng đa dạng của nó, chẳng hạn như đồ họa máy tính, nhận dạng mẫu, xử lý hình ảnh, tìm đường đi ngắn nhất cho robot, số liệu thống kê, v.v... Hơn nữa, bài toán tìm bao lồi thường được sử dụng như một bài toán phụ, một bước tiền xử lý quan trọng trong các bài toán hình học khác như tìm tam giác phân Delaunay, tính biểu đồ Voronoi, tìm đường kính của một tập hợp, tìm các lớp lồi của một tập hợp, tìm đường đi ngắn nhất, v.v... Các ứng dụng quan trọng cũng như các ý nghĩa thực tiễn của bài toán tìm bao lồi đã thu hút được rất nhiều nhà khoa học nghiên cứu và đưa ra các thuật toán giải bài toán này. Điển hình như sự phát hiện của D. R. Chand và S. S. Kapur vào năm 1970 và R. A. Jarvis vào năm 1973 với thuật toán gói quà (gift-wrapping), R. Graham vào năm 1972 với thuật toán quét Graham (Graham scan), W. Eddy năm 1977 và A. Bykat năm 1978 với thuật toán Quickhull, F. P. Preparata và S. J. Hong năm 1977 với thuật toán chia để trị (devide-and-conquer), M. Kallay năm 1984 với thuật toán tăng dần (incremental), T. Chan vào năm 1993 với thuật toán Chan, v.v...

Hiện nay có rất nhiều đề xuất cải tiến để tăng tốc cho các thuật toán tìm bao lồi nhằm đáp ứng các yêu cầu của cuộc sống hiện đại như xử lí các vấn đề ở tốc độ cao cho các dữ liệu lớn. Năm 2010, P. T. An cải tiến thuật toán quét Graham tìm bao lồi cho tập hữu hạn điểm trong không gian $\mathbb{R}^2$. Năm 2013, P. T. An và L. H. Trang tiếp tục áp dụng phương pháp đường định hướng để tăng tốc cho thuật toán gói quà tìm bao lồi của tập hữu hạn điểm trong không gian $\mathbb{R}^3$, v.v...

Bài toán tìm bao lồi nhưng có dữ liệu đầu vào là tập các hình tròn cũng là một bài toán quan trọng trong hình học tính toán xuất hiện từ những năm 1990 trở lại đây. Đầu tiên là năm 1992, D.

Rappaport giới thiệu một thuật toán $O(n \log n)$ được lấy ý tưởng từ thuật toán chia để trị nhằm tính bao lồi cho tập hình tròn. Trong bài báo này, tác giả cũng đã chỉ ra ứng dụng của bài toán trong một số vấn đề hình học khác như xác định bán kính của tập các hình tròn, tìm ô Voronoi xa nhất, tính miền stabbing và xây dựng miền giao của tập các hình tròn. Năm 1995, O. Devillers và M. Golin đề xuất một cải tiến từ thuật toán tăng dần với tập các hình tròn đầu vào được sắp xếp theo thứ tự độ lớn bán kính của chúng. Đến năm 1998, W. Chen và các cộng sự đã trình bày một phương pháp song song để tìm bao lồi của tập các hình tròn. Năm 2004, D. -S. Kim và các cộng sự đã sử dụng bao lồi của tập các hình tròn như một bài toán phụ để giải quyết bài toán tìm đường đi ngắn nhất tránh các vật cản là các đĩa.

Nhận ra tầm quan trọng và sự cần thiết của việc tăng tốc cho bài toán tìm bao lồi, luận án "Bài toán tìm bao lồi của tập điểm hữu hạn và ứng dụng" đề xuất những phương pháp cải tiến cho một số thuật toán tiêu biểu tìm bao lồi cho tập điểm. Nội dung nghiên cứu chính của luận án bao gồm:

1. Đề xuất một số kỹ thuật cải tiến cho thuật toán Quickhull tìm bao lồi trong không gian $\mathbb{R}^2$, thuật toán tìm bao lồi dưới trong không gian $\mathbb{R}^3$ và thuật toán gói quà trong không gian $\mathbb{R}^d$ (với $d \geq 2$). Mỗi đề xuất chúng tôi đều tính toán thực nghiệm để chỉ ra sự hiệu quả so với các thuật toán hiện có.
2. Giới thiệu ứng dụng của thuật toán tìm bao lồi của tập điểm như một bước tiền xử lý quan trọng cho thuật toán dưới vi phân giải quyết bài toán tối ưu không trơn. Chúng tôi cũng thực hiện các thử nghiệm số để cho thấy sự hiệu quả khi sử dụng bước tiền xử lý này.
3. Đề xuất thuật toán Quickhull tìm bao lồi cho tập hữu hạn các hình tròn đồng thời chứng minh sự đúng đắn của thuật toán, tính độ phức tạp của thuật toán trong trường hợp xấu nhất, trung bình và theo nghĩa smoothed analysis. Các thử nghiệm số để minh họa thuật toán cũng được trình bày ở nội dung này.

Chương 1

Kiến thức chuẩn bị

1.1 Sự định hướng và một số kiến thức liên quan

Một *siêu phẳng có hướng* (orientation hyperplane), cũng có thể được gọi là *siêu phẳng định hướng*, $(x^1x^2\dots x^d)$ là một siêu phẳng chứa d điểm độc lập affin $x^1, x^2, \dots, x^d$ trong không gian $\mathbb{R}^d$ với thứ tự các điểm được hiểu theo nghĩa là

$$(x^1x^2\dots x^d) = (x^2x^3\dots x^dx^1) = \dots = (x^dx^1\dots x^{d-1})$$

và

$$(x^1x^2\dots x^d) \neq (x^2x^1\dots x^d)\dots$$

Trong $\mathbb{R}^d$, cho trước d điểm độc lập affin $x^i = (x^i_1, x^i_2, \dots, x^i_d)$, trong đó $i = 1, 2, \dots, d$, và điểm $t = (t_1, t_2, \dots, t_d)$. Định nghĩa

$$\operatorname{orient}(x^1, x^2, \dots, x^d, t) := \det\begin{pmatrix} x^1_1 & x^1_2 & \dots & x^1_d & 1 \\ x^2_1 & x^2_2 & \dots & x^2_d & 1 \\ \dots & & & & \\ x^d_1 & x^d_2 & \dots & x^d_d & 1 \\ t_1 & t_2 & \dots & t_d & 1 \end{pmatrix}. \quad (1.1)$$

Định nghĩa 1.1.1. Điểm t được gọi là *ở phía dương* (tương ứng *ở phía âm, thuộc*) siêu phẳng $(x^1x^2\dots x^d)$ nếu $\operatorname{orient}(x^1, x^2\dots x^d, t) > 0$ (tương ứng, $\operatorname{orient}(x^1, x^2, \dots, x^d, t) < 0$, $\operatorname{orient}(x^1, x^2, \dots, x^d, t) = 0$).

Ta có

$$\operatorname{orient}(x^1, x^2, \dots, x^d, t) \quad := \quad \det\begin{pmatrix} x^1_1 & x^1_2 & \dots & x^1_d & 1 \\ x^2_1 & x^2_2 & \dots & x^2_d & 1 \\ \dots & & & & \\ x^d_1 & x^d_2 & \dots & x^d_d & 1 \\ t_1 & t_2 & \dots & t_d & 1 \end{pmatrix}$$

$$= \sum_{j=1}^{d} (-1)^{d+1+j} t_j |M_{d+1,j}|,$$

trong đó $|M_{d+1,j}|$ là định thức con của orient$(x^1, x^2, \ldots, x^d, t)$ được xác định bằng cách bỏ dòng thứ $d+1$ và cột thứ j. Nếu đặt

$$\nu_j = (-1)^{d+1+j} |M_{d+1,j}|, \; j = 1, 2, \ldots, d \quad \text{và} \quad \nu_{d+1} = |M_{d+1,d+1}|, \tag{1.2}$$

và ký hiệu $\overrightarrow{\nu} = (\nu_1, \nu_2, \ldots, \nu_d)$ thì

$$\text{orient}(x^1, x^2, \ldots, x^d, t) = \sum_{i=1}^{d} \nu_i t_i + \nu_{d+1}. \tag{1.3}$$

Dễ dàng thấy rằng biểu thức orient có dạng một phương trình siêu phẳng qua d điểm $x^1, x^2, \ldots, x^d$ với vectơ pháp tuyến là $\overrightarrow{\nu} = (\nu_1, \nu_2, \ldots, \nu_d)$ và hệ số tự do là $\nu_{d+1} = -\sum\limits_{i=1}^{d} \nu_i z_i$. Tức là, siêu phẳng định hướng $(x^1 x^2 \ldots x^d)$ có phương trình

$$\sum_{i=1}^{d} \nu_i x_i + \nu_{d+1} = 0. \tag{1.4}$$

Định nghĩa 1.1.2. Gọi H là siêu phẳng định hướng có phương trình cho bởi (1.4) với vectơ $\overrightarrow{\nu}$ được xác định bởi (1.2). *Nửa không gian dương* của H, ký hiệu bởi H^+, được định nghĩa là nửa không gian nằm ở phía dương và bị chặn bởi siêu phẳng H. Nửa không gian mở còn lại được gọi là *nửa không gian âm* của H, ký hiệu bởi H^-. Vectơ $\overrightarrow{\nu}$ được gọi là một *vectơ định hướng dương* của siêu phẳng định hướng H.

Nhận xét 1.1.3. Nếu z là một điểm bất kỳ của H và một điểm t nằm ở không gian dương H^+ thì $\overrightarrow{\nu}\overrightarrow{zt} > 0$, điểm t nằm ở nửa không gian âm H^- thì $\overrightarrow{\nu}\overrightarrow{zt} < 0$, điểm t thuộc H nếu $\overrightarrow{\nu}\overrightarrow{zt} = 0$.

1.2 Bài toán tìm bao lồi và ứng dụng

1.2.1 Bài toán tìm bao lồi cho tập hữu hạn điểm

Từ định nghĩa của bao lồi ta dễ dàng thấy rằng bao lồi của tập P là tập lồi nhỏ nhất chứa P. Bao lồi của một tập hữu hạn điểm $P \subset \mathbb{R}^d$ là một đa diện lồi trong $\mathbb{R}^d$. Một đa diện lồi chỉ có hữu hạn các diện rời nhau, mỗi diện cũng là một đa diện lồi. Như vậy, *bài toán tìm bao lồi* của tập P là bài toán xác định tất cả các diện của $\mathrm{conv}(P)$ với một thứ tự mà theo đó ta có thể dựng lại được $\mathrm{conv}(P)$. Ta có dữ liệu đầu vào của bài toán là tập hợp P hữu hạn gồm n điểm $p_1, p_2, \ldots, p_n$ và đầu ra là tập các diện của bao lồi $\mathrm{conv}(P)$.

1.2.2 Bài toán tìm bao lồi cho tập hình tròn

Từ định nghĩa bao lồi của một tập bất kỳ, ta cũng có bao lồi của tập hợp $D = \{d_1, d_2, \ldots, d_n\} \subset \mathbb{R}^2$ gồm n hình tròn có tâm $c_i(c_{ix}, c_{iy})$ và bán kính $r_i \geq 0$, ký hiệu bởi $\mathrm{conv}(D)$, là tập lồi nhỏ nhất chứa D. Dữ liệu đầu vào của bài toán là tập hợp D gồm n hình tròn $d_1, d_2, \ldots, d_n$ và đầu ra là tập các đường tròn cực biên của bao lồi $\mathrm{conv}(D)$.

Chương 2

Bài toán tìm bao lồi cho tập điểm

Một số cải tiến của thuật toán Quickhull tìm bao lồi của tập điểm 2D trong chương này nhằm mục đích giảm số lượng phép toán căn bản (orient) trong tính toán, giảm không gian tìm nghiệm và giảm kích thước dữ liệu đầu vào. Những kết quả tính toán đã chỉ ra rằng các cải tiến đã thực sự mang lại hiệu quả về tốc độ và khả năng tính toán cho thuật toán Quickhull với độ tăng tốc gấp khoảng 3 lần so với phiên bản hiện có.

2.1 Cải tiến thuật toán Quickhull 2D

2.1.1 Thuật toán Quickhull

Thuật toán Quickhull xác định bao lồi của tập hữu hạn điểm P. Thuật toán bắt đầu bởi việc tìm hai điểm cực đặc biệt (ta thường chọn điểm tận cùng trên trái và điểm tận cùng dưới phải), ta gọi là p và q, và lưu hai điểm này vào tập đỉnh H của $\text{conv}(P)$. Đường thẳng pq chia $(n-2)$ điểm còn lại vào hai tập con P_1 và P_2, trong đó P_1 chứa các điểm của P và nằm ở phía dương đường thẳng pq, P_2 là tập các điểm của P và nằm ở phía âm đường thẳng pq. Sau đó, trong tập P_1 ta tìm điểm r có khoảng cách đến pq là xa nhất. Thêm điểm r vào tập đỉnh H. Ba điểm p, q và r chia tập P_1 thành ba tập con S_0, S_1 và S_2, trong đó S_0 chứa các điểm nằm bên trong tam giác prq, S_1 chứa các điểm nằm ở phía dương pr và S_2 chứa các điểm nằm ở phía dương rq. Ta thay pq bởi pr và rq và tiếp tục lặp lại thuật toán. Áp dụng các bước thực hiện của P_1 cho tập P_2 ta sẽ được bao lồi của tập P.

2.1.2 Hạn chế các phép tính orient

Ở đây chúng tôi trình bày một đề xuất cải tiến cho thuật toán Quickhull nhằm giảm số lượng phép toán cơ bản orient của nó. Với đề xuất này thời gian tính toán trên các thử nghiệm số của thuật toán giảm khoảng 30% so với phiên bản Quickhull hiện tại.

2.1.3 Sử dụng vectơ định hướng

Xuất phát từ ý tưởng phương pháp đường định hướng được giới thiệu bởi H. X. Phú, chúng tôi đề xuất một kỹ thuật mới sử dụng "vectơ định hướng" (orienting vectors) để giảm số lượng phép tính orient của thuật toán Quickhull tính bao lồi cho tập điểm 2D. Áp dụng cải tiến này thuật toán Quickhull giảm khoảng 23% so với phiên bản ban đầu.

2.1.4 Tiền xử lý và chia nhỏ bài toán

Chúng tôi sử dụng tính chất của một số điểm cực đặc biệt để giảm bớt số lượng điểm đầu vào, đồng thời chia nhỏ bài toán và kết hợp với ý tưởng vừa nêu trong Mục 2.1.2 để cải tiến thuật toán Quickhull. Áp dụng ý tưởng cải tiến này thời gian tính toán của thuật toán Quickhull giảm khoảng 48% so với phiên bản hiện thời.

2.1.5 Thử nghiệm số

Để so sánh các thuật toán với nhau chúng tôi đã tạo ngẫu nhiên một vài kiểu dữ liệu cho các tập điểm đầu vào. Với mỗi loại dữ liệu chúng tôi tạo 27 ví dụ, trong đó số điểm ở các ví dụ thay đổi từ 10.000 đến 30.000.000. Các thuật toán được thực thi bằng chương trình C và chạy trên máy tính Core 2Duo 2*2.0 GHz với 2GB RAM. Chúng tôi thử nghiệm cho bốn phiên bản của thuật toán Quickhull. Khi kết hợp cả ba kỹ thuật thì thời gian tính toán nhận được tăng khoảng từ $2,5$ lần (với các tập điểm kiểu hình tròn rỗng) tới 4 lần (cho các tập điểm dữ liệu hình vuông).

2.2 Cải tiến thuật toán gói quà

Thuật toán gói quà là một thuật toán tìm bao lồi cơ bản, hiệu quả và dễ thực thi trong không gian $\mathbb{R}^d$. Thuật toán này xác định bao lồi conv(P) của một tập P hữu hạn điểm bắt đầu với việc tìm một mặt con (subfacet) đầu tiên E trong tập $\mathcal{E}$ các mặt con của conv(P). Bước tiếp theo ta xác định một mặt (facet) F của conv(P) qua E. Qua các mặt con của mặt F ta tiếp tục tìm các mặt khác của bao lồi cho tới khi tất cả các điểm của tập P được "gói lại". Do vậy, một thủ tục chính của thuật toán là tìm một mặt của bao lồi conv(P) qua một mặt con $E \in \mathcal{E}$ cho trước. Ở trong nội dung này chúng tôi đưa ra một kỹ thuật giới hạn để cải tiến thủ tục này.

2.2.1 Kỹ thuật miền hạn chế

Cho

$$P = \{(p_1^i, p_2^i, \ldots, p_d^i) \in \mathbb{R}^d,\ i = 1, 2, \ldots, n,\ n > d\}.$$

Và đặt

$$x_j^M := \max_{1 \le i \le m} \{p_j^i : p^i(p_1^i, p_2^i, \ldots, p_d^i) \in P\}, \tag{2.1}$$

$$x_j^m := \min_{1 \le i \le n} \{p_j^i : p^i(p_1^i, p_2^i, \ldots, p_d^i) \in P\}, \tag{2.2}$$

trong đó $j = 1, 2, \ldots, d$.

Cho $\mathcal{H}$ là siêu hộp được xác định bởi giao của $2d$ nửa không gian đóng chứa P và bị chặn bởi $2d$ siêu phẳng $x_j = x_j^M$ và $x_j = x_j^m$ trong đó $j = 1, 2, \ldots, d$. Khi đó ta có conv$(P) \subset \mathcal{H}$.

Hình chiếu song song với trục tọa độ Ox_d của siêu hộp $\mathcal{H}$ trên siêu phẳng toạ độ $Ox_1x_2\ldots x_{d-1}$ được gọi là *siêu hộp cơ sở* (foundation hyperrectangle) $\mathcal{H}'$. Giả sử rằng $\nu_d \neq 0$ và D_1, D_2 lần lượt là giao của siêu phẳng $(a^1a^2\ldots a^{d-1}t)$ với các siêu phẳng $x_d = x_d^M$, $x_d = x_d^m$ (D_1, D_2 là các $d-2$ diện của $\mathcal{H}$). Chúng tôi ký hiệu D_1', D_2' là các hình chiếu của D_1, D_2 trên siêu phẳng tọa độ $Ox_1x_2\ldots x_{d-1}$. Miền bị giới hạn bởi $\mathcal{H}', D_1', D_2'$, ký hiệu là

$(a^1a^2\ldots a^{d-1}t)_{Ox_1x_2\ldots x_{d-1}}$, được gọi là *miền hạn chế* (the restricted area) của siêu phẳng $(a^1a^2\ldots a^{d-1}t)$ trên $Ox_1x_2\ldots x_{d-1}$.

Mệnh đề 2.2.1. *Trong siêu phẳng tọa độ $Ox_1x_2\ldots x_{d-1}$, D'_2 nằm trong nửa không gian D'^{+}_1 nếu và chỉ nếu $\nu_d > 0$. Ngược lại, D'_2 nằm trong nửa không gian D'^{-}_1 nếu và chỉ nếu $\nu_d < 0$.*

Bổ đề 2.2.2. i) *Nếu $\nu_d > 0$ và nếu điểm p' nằm ở phía dương (tương ứng âm) của D'_2 (tương ứng D'_1) trong siêu phẳng tọa độ $Ox_1x_2\ldots x_{d-1}$ thì $p^*_d < p_d$ (tương ứng $p^*_d > p_d$).*

ii) *Nếu $\nu_d < 0$ và nếu điểm p' nằm ở phía dương (tương ứng âm) của D'_1 (tương ứng D'_2) trong siêu phẳng tọa độ $Ox_1x_2\ldots x_{d-1}$ thì $p^*_d > p_d$ (tương ứng $p^*_d < p_d$).*

Thủ tục tìm một mặt của bao lồi qua một mặt con của conv(P) sử dụng Mệnh đề 2.2.3 dưới đây.

Mệnh đề 2.2.3. i) *Nếu $\nu_d > 0$ và nếu $p \in P$ sao cho p' nằm trong D'^{+}_2 (tương ứng D'^{-}_1) trong siêu phẳng tọa độ $Ox_1x_2\ldots x_{d-1}$, thì p phải nằm trong nửa không gian $(a^1a^2\ldots a^{d-1}t)^+$ (tương ứng $(a^1a^2\ldots a^{d-1}t)^-$).*

ii) *Nếu $\nu_d < 0$ và nếu $p \in P$ sao cho p' nằm trong D'^{+}_1 (tương ứng D'^{-}_2) trong siêu phẳng tọa độ $Ox_1x_2\ldots x_{d-1}$, thì điểm p phải nằm trong nửa không gian $(a^1a^2\ldots a^{d-1}t)^+$ (tương ứng $(a^1a^2\ldots a^{d-1}\ t)^-$).*

2.2.2 Miền hạn chế tốt nhất

Gọi S'_{Ox_d} là diện tích của $(a^1a^2\ldots a^{d-1}t)_{Ox_1x_2\ldots x_{d-1}}$, và S_{Ox_d} diện tích của $\mathcal{H}'$.

Định nghĩa 2.2.4. *Tỉ số $\mathcal{R}_{Ox_d} = S'_{Ox_d}/S_{Ox_d}$ được gọi là tỉ số hạn chế (restricted ratio) của $(a^1a^2\ldots a^{d-1}t)$ trên $Ox_1x_2\ldots x_{d-1}$.*

Định nghĩa 2.2.5. *Miền hạn chế của siêu phẳng $(a^1a^2\ldots a^{d-1}t)$ ứng với giá trị nhỏ nhất trong n tỉ số hạn chế được gọi là miền hạn chế tốt nhất (best restricted area) của siêu phẳng $(a^1a^2\ldots a^{d-1}t)$.*

Tìm miền hạn chế tốt nhất có nghĩa ta sẽ đi tìm miền hạn chế có tỉ số hạn chế nhỏ nhất trong n miền hạn chế của cùng một siêu phẳng trên các siêu phẳng tọa độ.

2.2.3 Một số kết quả tính toán

Chúng tôi thử nghiệm cho dữ liệu trong không gian 2D và 3D. Các thuật toán được thực thi bởi chương trình C và chạy trên PC Core i5 1.6 GHz 3M với 4 GB RAM. Dũ liệu đầu vào của các thuật toán là tập điểm được tạo ngẫu nhiên trong hình lập phương và trên mặt cầu. Chúng tôi thử nghiệm trên các tập điểm được tạo ngẫu nhiên trong không gian $\mathbb{R}^2$ và $\mathbb{R}^3$. Các thử nghiệm số chỉ ra rằng thời gian thực thi thuật toán của chúng tôi giảm trung bình khoảng 40% so với thuật toán gói quà ban đầu và khoảng 35% so với phiên bản cải tiến mới nhất của nó.

2.3 Bài toán tìm bao lồi dưới của tập điểm hữu hạn trong $\mathbb{R}^3$

Bao lồi dưới cũng là một trong những cấu trúc quan trọng và có nhiều ứng dụng trong lính vực hình học tính toán. Một ứng dụng tiêu biểu là được sử dụng để xác định tam giác phân Delaunay và biểu đồ Voronoi. Vì vậy trong mục này chúng tôi trình bày một kỹ thuật để tăng tốc cho thuật toán tính bao lồi dưới được dùng trong lớp bài toán tính tam giác phân Delaunay.

2.3.1 Bao lồi dưới của một tập điểm

Định nghĩa 2.3.1. *Mặt dưới* của bao lồi là một tam giác có các đỉnh thuộc tập P sao cho tất cả các điểm của P thuộc vào hoặc ở phía trên mặt phẳng đi qua ba đỉnh này. *Bao lồi dưới* (lower convex hull), được ký hiệu bởi $\mathrm{conv}_L(P)$, chứa tất cả các mặt dưới của bao lồi. Một mặt của $\mathrm{conv}(P)$ không phải mặt dưới được gọi là một mặt trên của bao lồi (upper facet).

Mệnh đề 2.3.2. *Cho P là một tập hữu hạn điểm trong $\mathbb{R}^3$ và $a, b, p \in P$, giả sử rằng $\overrightarrow{n} = (n_x, n_y, n_z) = \overrightarrow{ab} \times \overrightarrow{ap}$ và $n_z \neq 0$. Khi đó ta*

i) *Nếu* (abp) *là một mặt của* $\text{conv}(P)$ *và* $n_z < 0$ *thì nó là một mặt dưới của* $conv_L(P)$*.*
ii) *Nếu* $n_z > 0$ *thì* (abp) *không phải là một mặt dưới của* $conv_L(P)$*.*

2.3.2 Kỹ thuật hạn chế tính bao lồi dưới

Trong nội dung này chúng tôi sẽ giới thiệu một ý tưởng hạn chế không gian tính toán nhằm tăng tốc cho thủ tục $\mathbf{LF}(e,p)$ trong lớp bài toán tìm bao lồi dưới với mục đích tính tam giác phân Delaunay. Cụ thể, ta có thể xác định lưới tam giác phân Delaunay của tập điểm $P^* = \{p_1^*, p_2^*, \ldots, p_d^*\}$ trong $\mathbb{R}^2$ thông qua việc tính bao lồi dưới trong $\mathbb{R}^3$ như sau:
Bước 1: Tạo tập $P = \{p_i(x_i, y_i, z_i) \in \mathbb{R}^3, i = 1, 2, \ldots, n-1\}$ sao cho cao độ của p_i thoả mãn $z_i = (x_i - q_x)^2 + (y_i - q_y)^2$, trong đó $q(q_x, q_y)$ là điểm trung bình của tập hợp P^*.
Bước 2: Tính bao lồi dưới $\text{conv}_L(P)$ của tập P.
Bước 3: Chiếu tất cả các mặt của bao lồi dưới $\text{conv}_L(P)$ theo phương song song với trục Oz lên mặt phẳng Oxy (mặt chứa tập điểm P^*) ta sẽ nhận được tam giác phân Delaunay tương ứng của P^*.

Nhắc lại rằng bài toán toán của ta là tìm bao lồi dưới của tập

$$P = \{p_i = (x_i, y_i, z_i) \in \mathbb{R}^3, z_i = (x_i - q_x)^2 + (y_i - q_y)^2, i = 1, \ldots, n\},$$

trong đó $q(q_x, q_y)$ là điểm trung bình của tập P. Như vậy tập điểm P được phân bố trên bề mặt của một paraboloid $(\mathcal{P})$ có phương trình

$$z = (x - q_x)^2 + (y - q_y)^2. \qquad (\mathcal{P})$$

Giả sử mặt phẳng (abp) với $a, b, p \in P$ có phương trình

$$n_x x + n_y y + n_z z = d.$$

Vì $a, b, p \in P$ nên (abp) luôn cắt paraboloid $(\mathcal{P})$ theo một đường ellipse $(\mathtt{E})$. Ta sẽ tìm các giá trị lớn nhất, nhỏ nhất của hoành độ, tung độ và cao độ của các điểm thuộc $(\mathtt{E})$. Dựa vào những giá trị này ta có thể có xét vị trí tương đối của một điểm thuộc P với mặt phẳng (abp) một cách thuận lợi và đơn giản hơn. Đặt

$$\min x_{\mathtt{E}} := \min\{x : (x, y, z) \in (\mathtt{E})\}, \quad \max x_{\mathtt{E}} := \max\{x : (x, y, z) \in (\mathtt{E})\},$$

$$\min y_{\mathtt{E}} := \min\{y : (x, y, z) \in (\mathtt{E})\}, \quad \max y_{\mathtt{E}} := \max\{y : (x, y, z) \in (\mathtt{E})\},$$

$$\min z_{\mathtt{E}} := \min\{z : (x, y, z) \in (\mathtt{E})\}, \quad \max z_{\mathtt{E}} := \max\{z : (x, y, z) \in (\mathtt{E})\}.$$

Ta có $(\mathcal{C})$ có dạng phương trình đường tròn trong mặt phẳng Oxy và đường tròn này là hình chiếu của $(\mathtt{E})$ theo phương song song với trục Oz lên mặt phẳng toạ độ Oxy.

$$(x - q_x - \frac{\alpha}{2})^2 + (y - q_y - \frac{\beta}{2})^2 = \alpha q_x + \beta q_y + \frac{\alpha^2}{4} + \frac{\beta^2}{4} + \gamma. \quad (\mathcal{C})$$

Đường tròn $(\mathcal{C})$ có tâm là điểm c có toạ độ

$$\begin{cases} c_x = q_x + \frac{\alpha}{2} \\ c_y = q_y + \frac{\beta}{2}. \end{cases} \tag{2.3}$$

và bán kính là

$$r = \sqrt{\alpha q_x + \beta q_y + \frac{\alpha^2}{4} + \frac{\beta^2}{4} + \gamma}. \tag{2.4}$$

Dựa vào đặc điểm của đường tròn $(\mathcal{C})$ ta suy ra

$$\min x_{\mathtt{E}} = c_x - r, \ \max x_{\mathtt{E}} = c_x + r, \ \min y_{\mathtt{E}} = c_y - r, \ \max y_{\mathtt{E}} = c_y + r.$$

Bổ đề 2.3.3. *Ta có*

i) $\min z_{\mathtt{E}} = (cq - r)^2$,

ii) $\max z_{\mathtt{E}} = (cq + r)^2$.

Gọi hình chiếu của các điểm a, b, p là a', b', p', khi đó $a', b', c' \in (\mathcal{C})$. Ta có mệnh đề sau.

Mệnh đề 2.3.4. *Những điểm thuộc tập P mà có hình chiếu nằm ngoài đường tròn $(\mathcal{C})$ thì nằm trên mặt phẳng (abp). Ngược lại, những điểm có hình chiếu nằm ở bên trong đường tròn $(\mathcal{C})$ thì nằm dưới mặt phẳng (abp).*

Mệnh đề 2.3.5 và 2.3.6 dưới đây giúp ta xác định nhanh một điểm nằm trên hay dưới một mặt phẳng nhờ phép so sánh toạ độ.

Mệnh đề 2.3.5. *Một điểm $t(t_x, t_y, t_z) \in P$ thoả mãn một trong những điều kiện sau*

i) $t_x < \min x_{\mathtt{E}}$,

ii) $t_x > \max x_{\mathtt{E}}$,

iii) $t_y < \min y_{\mathtt{E}}$,

iv) $t_y > \max y_{\mathtt{E}}$,

v) $t_z > \max z_{\mathtt{E}}$.

thì nằm trên mặt phẳng (abp).

Mệnh đề 2.3.6. *Một điểm $t(t_x, t_y, t_z) \in P$ thoả mãn $t_z < \min z_{\mathtt{E}}$ và $cq < r$ thì t nằm dưới mặt phẳng (abp).*

2.3.3 Một số kết quả tính toán

Trong nội dung này chúng tôi thử nghiệm số cho một thuật toán được P. T. An và Đ. T. Giang giới thiệu năm 2015 và thuật toán vừa được trình bày ở mục trước để so sánh tốc độ của chúng. Các thuật toán này được thực thi bởi chương trình C và chạy trên PC Core i5 1.6 GHz 3M với 4 GB RAM. Thuật toán áp dụng cải tiến của chúng tôi tăng tốc khá nhanh, gấp từ khoảng 1,58 đến 2,24 lần so với thuật toán được giới thiệu năm 2015.

Chương 3

Một ứng dụng của bài toán tìm bao lồi của tập điểm

Tìm vị trí tối ưu là một bài toán thường gặp trong thực tế. Bài toán được xét trong chương này là xác định một điểm x trong một tập lồi đóng cho trước $D \subset \mathbb{R}^d$ (thông thường $d \in \{2, 3\}$) sao cho khoảng cách xa nhất từ x tới các điểm của một tập hữu hạn $C \subset \mathbb{R}^d$ là ngắn nhất.

Bài toán được xét trong mục này có thể được mô hình hóa như một bài toán tối ưu lồi không trơn với hàm mục tiêu lồi mạnh. Chúng tôi đề xuất một cải tiến cho thuật toán dưới vi phân để giải quyết bài toán tối ưu không trơn. Thông thường, trong thực tế thì lực lượng của tập C rất lớn. Tuy nhiên, do tính lồi của hàm khoảng cách, ta có thể thay thế tập C bởi tập V_C chứa các đỉnh của bao lồi của C. Trong rất nhiều các trường hợp thực tế, số phần tử của tập V_C nhỏ hơn đáng kể so với số phần tử của tập C. V. Damerow và C. Sohler đã chỉ ra rằng, lực lượng trung bình của V_C là $O(\log^{d-1} n)$. Do đó, tìm đỉnh của bao lồi của C là một bước tiền xử lý quan trọng trước khi giải bài toán xác định vị trí tối ưu. Một số kết quả tính toán thực nghiệm của chúng tôi đã chỉ ra rằng việc tính bao lồi trước khi thực hiện bài toán tăng tốc đáng kể so với việc không sử dụng kỹ thuật này.

3.1 Phát biểu bài toán và các tính chất

Bài toán tìm vị trí tối ưu được phát biểu cụ thể như sau.

Bài toán 3.1.1. *Cho tập lồi đóng $D \subset \mathbb{R}^n$ và một tập hữu hạn điểm C. Tìm một điểm x trong một tập D sao cho khoảng cách Euclide xa nhất từ x tới các điểm của tập C là ngắn nhất.*

Khoảng cách từ một điểm x tới một điểm y được định nghĩa bởi $\|x-y\|$. Đặt $\theta(x,C) := \max_{y\in C} \|x-y\|^2$, khi đó Bài toán 3.1.1 được biểu diễn dưới dạng toán học như sau

$$\min_{x\in D} \theta(x,C). \tag{P}$$

Bổ đề 3.1.2. *Ký hiệu V_C là tập các đỉnh của* conv*(C). Khi đó ta có*

i) $V_C \subseteq C$.

ii) $\theta(x,C) = \max\{\|x-y\|^2 \mid y \in V_C\}$.

Bổ đề 3.1.3. *Cho $v^1, v^2, \ldots, v^m$ là các phần tử của V_C và $d_j(x,C) := \|x-v^j\|^2$ với mỗi $j \in J := \{1,2,\ldots,m\}$. Khi đó ta có*

i) *$\theta(x,C)$ là hàm lồi mạnh với mô đun 2.*

ii) $\partial\theta(\cdot,C)(x) = \operatorname{conv}\left(\bigcup_{j\in J(x)} \partial\theta_j(\cdot,C)(x)\right)$*, trong đó $\partial\theta_j(\cdot,C)(x)$ là khả dưới vi phân của hàm lồi $\theta_j(\cdot,C)$ tại x và $J(x) = \{j \in J \mid \theta(x,C) = \theta_j(x,C)\}$.*

3.2 Thuật toán và sự hội tụ

Định lý 3.2.1. i) *Nếu thuật toán 3.1 dừng ở một vài bước lặp k, thì x^k là nghiệm của bài toán (P).*

ii) *Nếu thuật toán 3.1 không dừng thì dãy $\{x^k\}$ hội tụ đến nghiệm x^* của bài toán (P).*

Để chứng minh Định lý 3.2.1 ta cần các khẳng định sau.

Khẳng định 3.2.2. Ta có

$$\|x^{k+1} - x^k\| \le \beta_k, \ \forall k \in \mathbb{N}.$$

Khẳng định 3.2.3. Dãy $\{\|x^k - x^*\|^2\}$ hội tụ với mọi k.

Khẳng định 3.2.4. Ta có

$$\limsup_{k\to+\infty}(\theta(x^k,C) - \theta(x^*,C)) = 0. \tag{3.2}$$

Thuật toán 3.1 THUẬT TOÁN TÌM NGHIỆM TỐI ƯU x^k

Khởi tạo. Chọn $x^0 \in D$, cố định một tham số $\rho > 0$ và chọn một dãy $\{\beta_k\}$ số dương thỏa mãn điều kiện

$$\sum_{k=0}^{\infty} \beta_k = +\infty, \ \sum_{k=0}^{\infty} \beta_k^2 < \infty. \tag{3.1}$$

Cho $k := 0$.
Bước 1. Tìm $v^k \in V_C$ sao cho

$$v^k \in \arg\max\{\|x^k - v\|^2 : v \in V_C\}.$$

Bước 2. Đặt $g^k := 2(x^k - v^k)$ là dưới đạo hàm của $\|x^k - v^k\|^2$ tại x^k.

Trường hợp 2a: Nếu $g^k = 0$ thì thuật toán dừng, x^k là nghiệm tối ưu của (P).

Trường hợp 2b: Nếu $g^k \neq 0$, tính

$$\alpha_k := \frac{\beta_k}{\max\{\rho, \|g^k\|\}} \quad \text{và} \quad x^{k+1} := P_D(x^k - \alpha_k g^k),$$

trong đó P_D là phép chiếu mêtric trên D.

Bước 3. Nếu $x^{k+1} = x^k$ thì thuật toán dừng, x^k là nghiệm tối ưu của bài toán (P). Ngược lại, đặt $k := k + 1$ và quay lại Bước 1.

Nhận xét 3.2.5. Nếu $g^k = 0$ hoặc $x^{k+1} = x^k$ thì x^k là một nghiệm chính xác của bài toán. Trong tính toán số, cho một nghiệm xấp xỉ ta có thể dừng thuật toán nếu $\|g^k\| \leq \varepsilon$ hoặc $\|x^{k+1} - x^k\| \leq \max\{\|x^k\|, 1\}\epsilon$, trong đó $\epsilon > 0$ là sai số cho trước.

3.3 Một số kết quả tính toán

Trong mục này ta thảo luận về một số kết quả thực nghiệm của không gian hai chiều. Giả sử rằng lực lượng của tập C những người sử dụng dịch vụ là rất lớn (điều này thường xảy ra trong mô hình thực tế) và tập D (nơi đặt vị trí cơ sở để phục vụ cho các đối tượng trong tập C) là một tập lồi.

Như đã trình bày ở trên, để tối thiểu hàm $\theta(x, C)$ ta chỉ cần tính trên các đỉnh của bao lồi của tập C. Chúng tôi áp dụng cải tiến của thuật toán Quickhull được trình bày ở Chương 2 để tìm $\text{conv}(C)$. Thuật toán tính bao lồi được thực hiện bởi chương trình C, thuật toán 3.1 tìm nghiệm tối ưu được tính toán bằng chương trình matlab. Các thuật toán được chạy trên PC 1.8GHz Intel Core i5 với 8 GB RAM. Các thử nghiệm số của chúng tôi chỉ ra rằng việc tìm các đỉnh của bao lồi của tập điểm C là một bước tiền xử lý cần thiết. Với bước thực hiện này giúp cho thuật toán của chúng tôi giảm được đáng kể kích thước của tập C. Thời gian tính toán của bài toán khi sử dụng bước tiền xử lý tăng tốc rất đáng kể so với thời gian tính toán khi không sử dụng kỹ thuật này.

Chương 4

Bài toán tìm bao lồi cho tập hữu hạn các hình tròn

4.1 Sự định hướng cho các hình tròn trong $\mathbb{R}^2$

Định nghĩa 4.1.1. Một hình tròn d được gọi là nằm *ở phía dương* (tương ứng *phía âm*) một đường thẳng có hướng Δ nếu hình tròn đó có nhiều nhất một điểm không nằm ở phía dương (tương ứng ở phía âm) đường thẳng Δ.

Để đơn giản, khi ta viết "hình tròn d đi qua điểm q" có nghĩa là "hình tròn d có biên $\partial(d)$ đi qua điểm q".

Định nghĩa 4.1.2. Một hình tròn của tập D được gọi là một *hình tròn cực biên* nếu nó đi qua một điểm cực biên của D.

Ta biểu diễn kết quả của bài toán tìm bao lồi của tập D bởi $\mathrm{CH}(D) = \{d_1, d_2, \ldots, d_h, d_{h+1}\}$, trong đó $d_1 = d_{h+1}$, sao cho $t(d_i, d_{i+1})$ là một cạnh của $\mathrm{conv}(D)$ với $i = 1, 2, \ldots, h$. Ký hiệu $\mathrm{next}(d_h)$ là hình tròn cực biên liền sau d_h trong $\mathrm{CH}(D)$ và $\mathrm{prev}(d_h)$ là hình tròn cực biên liền trước d_h trong $\mathrm{CH}(D)$. Tức là $\mathrm{next}(d_h)$ $= d_{h+1}$ và $\mathrm{prev}(d_h) = d_{h-1}$.

4.2 Giới thiệu thuật toán

Thuật toán 4.1 mô tả thuật toán Quichkull tìm bao lồi của tập D hữu hạn các hình tròn. Đầu ra của thuật toán là tập các hình tròn cực biên $CH(D)$.

Thuật toán 4.1 THUẬT TOÁN QUICKHULL CHO TẬP HÌNH TRÒN

Đầu vào: Cho tập hợp $D = \{d_1, d_2, \ldots, d_n\}$ gồm n hình tròn có tâm $c_i(c_{ix}, c_{iy})$ và bán kính $r_i \geq 0$. Giả sử $n \geq 2$.
Đầu ra: CH(D).

1. Tìm hình tròn d_p và d_q tương ứng chứa điểm tận cùng trái trên p và tận cùng phải dưới q. Gán CH(D) := $\{d_p\}$. D_1 chứa d_p, d_q và tất cả các hình tròn của $D \setminus \{d_p, d_q\}$ không nằm ở phía dương của pq. D_2 chứa d_p, d_q và tất cả các hình tròn của $D \setminus \{d_p, d_q\}$ không nằm ở phía dương của qp.

2. Gọi Findhull (D_1, pq, d_p, d_q). Gán CH(D) := CH(D) $\cup \{d_q\}$. Gọi Findhull (D_2, qp, d_q, d_p).

Findhull(D_k, pq, d_p, d_q).

1. **if** $D_k = \emptyset$ **then return;**

2. **else** tìm $d_r \in S_k$ đi qua điểm r sao cho r có khoảng cách xa nhất tới pq.
   a. **if** ($d_r \neq d_p$ **and** $d_r \neq d_q$) **then**
      i. Đặt P_1 (tương ứng P_2) chứa d_p, d_r (tương ứng d_r, d_q) và tất cả các hình tròn của $D_k \setminus \{d_p, d_r\}$ (tương ứng $D_k \setminus \{d_r, d_q\}$) không nằm ở phía dương của pr (tương ứng rq).
      ii. Gọi *Findhull*(P_1, pr, d_p, d_r). Gán CH(D) := CH(D) $\cup$ $\{d_r\}$. Gọi *Findhull*(P_2, rq, d_r, d_q).
   b. **else if** $d_r = d_p$**then** tìm hình tròn next(d_p). Gọi p' là tiếp điểm trên next(d_p) của tiếp tuyến phải của hai hình tròn d_p và next(d_p).
      i. Đặt P_1 chứa $d_{p'}, d_q$ và tất cả các hình tròn thuộc $D_k \setminus \{d_{p'}, d_q\}$ không nằm ở phía dương của $p'q$.
      ii. Gán CH(D) := CH(D) $\cup$ $\{\text{next}(d_p)\}$. Gọi *Findhull*$(P_1, p'q, \text{next}(d_p), d_q)$.
   c. **else if** $d_r = d_q$ **then** tìm hình tròn prev(d_q). Gọi q' là tiếp điểm trên prev(d_q) của tiếp tuyến phải của hai hình tròn prev(d_p) và d_q.
      i. Đặt P_1 chứa $d_p, d_{q'}$ và tất cả các hình tròn của $D_k \setminus \{d_p, d_{q'}\}$ không nằm ở phía dương của pq'.
      ii. Gọi *Findhull*$(P_1, pq', d_p, \text{prev}(d_q))$. Gán CH($D$) := CH($D$) $\cup$ $\{\text{prev}(d_q)\}$.

4.3 Sự đúng đắn của thuật toán

Định lý 4.3.1. *Đầu ra của Thuật toán 4.1 là danh sách các hình tròn cực biên của D và được sắp xếp theo thứ tự ngược chiều kim đồng hồ.*

4.4 Độ phức tạp tính toán

4.4.1 Độ phức tạp tính toán trong trường hợp xấu nhất

Mỗi phép gọi *Findhull* luôn tìm ra một đường tròn cực biên. Việc tìm ra một hình tròn cực biên có độ phức tạp tính toán xấu nhất là $O(n)$. Khi mỗi hình tròn cực biên được chọn, có thể nó được xuất hiện nhiều nhất hai lần nữa trong các phép gọi *Findhull* ở bước tiếp theo (một phép gọi được đặt kề trước và một phép gọi được đặt kế sau). Do vậy thuật toán dừng sau không quá $3h$ phép gọi *Findhull*. Từ đó ta có thể kết luận rằng, độ phức tạp tính toán của thuật toán trong trường hợp xấu nhất là $O(n^2)$.

4.4.2 Độ phức tạp tính toán trung bình

Định lý dưới đây đánh giá số hình tròn cực biên trung bình của một tập hợp các hình tròn.

Định lý 4.4.1. *Cho tập D gồm n hình tròn có tâm thuộc tập $\mathcal{C}$ được chọn theo phân bố xác suất Δ bất kỳ. Khi đó trung bình số hình tròn cực biên là $O\left(\log^2 n\right)$.*

Định lý 4.4.2. *Độ phức tạp tính toán trung bình của Thuật toán 4.1 cho tập D gồm n hình tròn có tâm thuộc tập $\mathcal{C}$ được chọn theo phân bố xác suất Δ bất kỳ là $O(n \log^2 n)$.*

4.4.3 Độ phức tạp tính toán theo nghĩa smoothed analysis

Định nghĩa 4.4.3. Cho một thuật toán A và dữ liệu vào x. Khi đó ký hiệu $\mathcal{C}_A(x)$ là một phép đo độ phức tạp (complexity measure) của thuật toán A trên bộ dữ liệu x. Cho X_n là tập các bộ dữ liệu đầu vào có kích thước n của thuật toán A. Ta nói thuật toán A có độ phức tạp smoothed analysis $f(n,s)$ nếu

$$\max_{x \in X_n} E_s[\mathcal{C}_A(\tilde{x})] = f(n,s),$$

trong đó $\tilde{x} = x + s$ với s là biến ngẫu nhiên nào đó (thành phần gây nhiễu cho bộ dữ liệu đầu vào x), E_s là kỳ vọng của $\mathcal{C}_A(\tilde{x})$.

Định lý 4.4.4. *Độ phức tạp tính toán theo nghĩa smoothed analysis T_S của Thuật toán 4.1 cho tập D cho hai trường hợp như sau:*

i) *Nếu tập tâm của D được cho bởi phân bố chuẩn $N(0,\sigma)$ thì*

$$T_S = O\left(n\left(\frac{1}{\delta}\right)^2 \log^2 n\right)$$

ii) *Nếu tập tâm của D được cho bởi phân bố đều trong hình vuông kích thước 2ϵ, tức là*

$$\varphi(x) = \begin{cases} \frac{1}{2\epsilon} \text{ nếu } x \in [-\epsilon, \epsilon] \\ 0 \text{ nếu ngược lại,} \end{cases} \tag{4.1}$$

thì

$$T_S = O\left(n\left(\frac{n\log n}{\epsilon}\right)^{2/3}\right).$$

4.5 Một số kết quả tính toán

Các thử nghiệm số của chúng tôi chỉ ra rằng thuật toán Quickhull chạy nhanh hơn đáng kể so với thuật tăng dần. Hệ số tăng tốc của thuật toán Quickhull so với thuật toán gia tăng là từ 3,16 đến 4,03 lần.

Kết luận

Luận án trình bày một số vấn đề liên quan đến bài toán tìm bao lồi của tập hữu hạn điểm và tập hữu hạn các hình tròn đạt được các kết quả chính như sau.

- Đề xuất được một số kỹ thuật để tăng tốc cho thuật toán Quickhull 2D. Các tính toán chỉ ra thuật toán áp dụng các kỹ thuật này tăng tốc khá hiệu quả, tăng khoảng 3 lần so với phiên bản hiện có.
- Giới thiệu một kỹ thuật giới hạn không gian tìm kiếm để cải tiến thuật toán gói quà tìm bao lồi của tập điểm hữu hạn trong $\mathbb{R}^d$ với $d \geq 2$. Một số thử nghiệm tính toán đã chỉ ra rằng thuật toán áp dụng kỹ thuật này giảm được khoảng 40% so với thuật toán gói quà ban đầu và khoảng 35% so với một phiên bản cải tiến thuật toán này năm 2013.
- Đề xuất thuật toán cải tiến tìm bao lồi dưới cho lớp bài toán ứng dụng tìm tam giác phân Deulaunay và biểu đồ Voronoi.
- Trình bày một ứng dụng của bải toán tìm bao lồi giải một bài toán tìm vị trí tối ưu. Một số kết quả tính toán thực nghiệm của chúng tôi đã chỉ ra rằng, việc tính bao lồi trước khi thực hiện bài toán là hiệu quả.
- Đề xuất thuật toán Quickhull tìm bao lồi của tập hữu hạn các hình tròn trong mặt phẳng dựa vào ý tưởng của thuật toán Quickhull tính bao lồi cho bộ điểm. Các tính toán của chúng tôi ở phần cuối chương này chỉ ra rằng thuật toán Quickhull chạy nhanh hơn gấp khoảng 3,8 lần so với thuật toán gia tăng.

Trong thời gian tiếp theo chúng tôi sẽ tiếp tục tìm hiểu và mở rộng bài toán tìm bao lồi cho tập các hình cầu, tập các đoạn thẳng, tập các đa giác lồi, tập các đa diện lồi, v.v.... Hơn nữa, chúng tôi cũng muốn tiếp tục tìm hiểu các ứng dụng của bài toán tìm bao lồi trong các bài toán hình học khác như tìm đường đi ngắn nhất, tính biểu đồ voronoi của tập hình tròn, v.v...

Danh mục các công trình khoa học của tác giả liên quan đến luận án

1. N. D. Hoang and N. K. Linh (2015), "Quicker than quickhull", *Vietnam Journal of Mathematics* **43**, pp. 57–70.
2. N. K. Linh and L. D. Muu (2015), "A convex hull algorithm for solving a location problem", *RAIRO-Operations Research* **49**, pp. 589–600.
3. N. K. Linh, C. Song, P. T. An, N. D. Hoang, D.-S. Kim (2018), "QHullDisk: A Faster Convex Hull Algorithm for Disks", *Submitted to ACM Transactions on Mathematical Software (TOMS).*