

# Bài toán tối ưu hóa

Bách khoa toàn thư mở Wikipedia

Trong khoa học máy tính và toán học, **bài toán tối ưu hóa** là bài toán tìm kiếm *lời giải tốt nhất* trong tất cả các lời giải khả thi. Bài toán tối ưu hóa có thể được chia thành hai loại tùy thuộc vào việc các biến là liên tục hay rời rạc. Bài toán tối ưu hóa với các biến rời rạc còn được gọi là một bài toán tối ưu hóa tổ hợp. Trong một bài toán tối ưu hóa tổ hợp, chúng ta tìm kiếm một đô i tượng như là một số nguyên, hoán vị hay đồ thị từ một tập hợp hữu hạn (hoặc có thể là vô hạn đếm được). Bài toán với các biến liên tục bao gồm bài toán hạn chế và bài toán đa phương thức.

Mục lục

Bài toán tối ưu hóa liên tục

Bài toán tối ưu hóa tổ hợp

Bài toán tối ưu hóa NP

Tham khảo

Xem thêm

## Bài toán tối ưu hóa liên tục

Dạng tiêu chuẩn của một bài toán tối ưu hóa (liên tục) là <sup>[1]</sup>

minimize

$x$

subject to

$f(x)$

$g_i(x) \leq 0, \quad i = 1, \dots, m$

$h_i(x) = 0, \quad i = 1, \dots, p$

trong đó

- $f(x) : \mathbb{R}^n \rightarrow \mathbb{R}$ ,
- $g_i(x) \leq 0$  được gọi là ràng buộc bất bình đẳng, và
- $h_i(x) = 0$  được gọi là ràng buộc bình đẳng.

Theo quy ước, dạng tiêu chuẩn xác định một **bài toán cực tiểu hóa**. **Bài toán cực đại hóa** có thể được giải bằng cách phủ định hàm mục tiêu.

## Bài toán tối ưu hóa tổ hợp

Chính thức, là một bài toán tối ưu hóa tổ hợp **A** là một bộ tứ **(I, f, m, g)**, trong đó

- I** là một tập hợp các trường hợp;
- đưa ra một ví dụ  $x \in I$ , **f(x)** là tập hợp của các lời giải khả thi;
- đưa ra một ví dụ  $x$  và một lời giải khả thi **y** theo  $x$ , **m(x, y)** biểu thị số đo y, đó thường là một số thực dương.

- ***g*** là hàm mục tiêu, và là một trong hai **min** hoặc **max**.

Mục tiêu là sau đó tìm ra một số trường hợp ***x*** một lời giải tối ưu, có nghĩa là, là một lời giải khả thi ***y***

$$m(x, y) = g\{m(x, y') \mid y' \in f(x)\}.$$

Đối với mỗi bài toán tối ưu hóa tổ hợp, có một bài toán quyết định tương ứng yêu cầu cho dù đó có là một lời giải khả thi đối với một số biến pháp cụ thể ***m*<sub>0</sub>**. Ví dụ, nếu có một đồ thị ***G*** chứa các đỉnh ***u*** và ***v***, một bài toán tối ưu hóa có thể là "tìm một đường đi từ ***u*** tới ***v*** sử dụng các cạnh ít nhất". Bài toán này có thể có một câu trả lời, đó là, 4. Một bài toán quyết định tương ứng sẽ là "một đường đi từ ***u*** tới ***v*** mà sử dụng 10 cạnh hoặc ít hơn?" Bài toán này có thể có được câu trả lời đơn giản hoặc là 'có' hoặc là 'không'.

Trong lĩnh vực thuật toán xấp xỉ, các thuật toán được thiết kế để tìm các lời giải gần tối ưu cho các bài toán khó. Phiên bản quyết định bình thường sau đó là một định nghĩa không đầy đủ về bài toán này kể từ khi nó chỉ xác định các lời giải chấp nhận được. Mặc dù chúng ta có thể giới thiệu các bài toán quyết định phù hợp, bài toán này được mô tả đặc điểm tự nhiên hơn như là một bài toán tối ưu hóa.<sup>[2]</sup>

## Bài toán tối ưu hóa NP

Bài toán tối ưu hóa NP (NPO-"*NP optimization*") là bài toán tối ưu hóa tổ hợp với các điều kiện bổ sung sau.<sup>[3]</sup> Lưu ý rằng các đa thức dưới đây là các hàm kích thước của đầu vào của các hàm tương ứng, không phải là kích thước của một số tập hợp ảnh của các trường hợp đầu vào.

- Kích thước của mỗi lời giải khả thi  $y \in f(x)$  là đa thức bị chặn trong kích thước của ví dụ được đưa ra ***x***,
- $\{x \mid x \in I\}$  và  $\{(x, y) \mid y \in f(x)\}$  có thể được ghi nhận trong thời gian đa thức, và
- ***m*** là có thể tính được thời gian đa thức.

Điều này ngụ ý rằng bài toán quyết định tương ứng thì nằm trong NP. Trong khoa học máy tính, các bài toán tối ưu hóa thú vị thường có những đặc tính trên và cho nên đó là những bài toán NPO. Một bài toán ngoài ra còn được gọi là một bài toán tối ưu hóa-P (PO), nếu có tồn tại một thuật toán mà tìm các lời giải tối ưu trong thời gian đa thức. Thông thường, khi đối phó với lớp NPO, thứ được quan tâm trong các bài toán tối ưu hóa mà các phiên bản quyết định là NP-đầy đủ. Lưu ý rằng các quan hệ độ cứng luôn đối với một số phép suy giảm nào đó. Do sự kết hợp giữa các thuật toán xấp xỉ và các bài toán tối ưu hóa máy tính, các suy giảm duy trì xấp xỉ trong một số khía cạnh là dành cho đối tượng này được ưu tiên hơn so với mức giảm Turing và Karp thông thường. Một ví dụ về việc giảm như vậy sẽ giảm L. Vì lý do này, vấn đề tối ưu hóa với các phiên bản quyết định hoàn chỉnh NP không được gọi là hoàn chỉnh NPO.<sup>[4]</sup>

NPO được chia thành các phân lớp sau tùy theo tính xấp xỉ được của chúng:<sup>[3]</sup>

- ***NPO(I)***: Tương đương với FPTAS. Chứa bài toán xếp ba lô.
- ***NPO(II)***: Tương đương với PTAS. Chứa bài toán lịch Makespan.
- ***NPO(III)***: Lớp của các bài toán NPO có các thuật toán thời gian đa thức sẽ tính toán các lời giải với chi phí ở hầu hết chi phí tối ưu *c* lần (đối với các bài toán cực tiểu hóa) hoặc một chi phí tối thiểu bằng  $1/c$  chi phí tối ưu (đối với các bài toán cực đại hóa). Trong cuốn sách của Hromkovič, đã loại trừ trong lớp này đều là các bài toán NPO (II) được lưu nếu  $P = NP$ . Nếu không có sự loại trừ, tương đương với APX. Chứa MAX-SAT và số liệu TSP (bài toán người bán hàng).

- ***NPO(IV)***::Lớp các bài toán NPO với các thuật toán thời gian đa thức xấp xỉ lời giải tối ưu bởi một tỷ lệ đó là đa thức trong một logarit của kích thước đầu vào. Trong cuốn sách của Hromkovic, tất cả các bài toán NPO (III) được loại trừ lớp này trừ khi  $P = NP$ . Chứa bài toán bao gói tập hợp.
- ***NPO(V)***::Lớp của các bài toán NPO với các thuật toán thời gian đa thức xấp xỉ lời giải tối ưu bởi một tỷ lệ giới hạn bởi một số hàm trên  $n$ . Trong cuốn sách của Hromkovic, tất cả các bài toán NPO (IV) được loại trừ lớp này trừ khi  $P = NP$ . Chứa hai bài toán TSP và Nhóm cực đại.

Một lớp quan tâm khác là NPOPB, NPO với các hàm chi phí đa thức bị chặn. Các bài toán với điều kiện này có nhiều tính chất mong muốn.

## Tham khảo

- <sup>↑</sup> Boyd, Stephen P.; Vandenberghe, Lieven (2004).
- <sup>↑</sup> Ausiello, Giorgio; et al. (2003), *Complexity and Approximation* (Corrected ed.
- <sup>↑</sup> <sup>*a*</sup> <sup>*b*</sup> Hromkovic, Juraj (2002), *Algorithmics for Hard Problems*, Texts in Theoretical Computer Science (2nd ed.
- <sup>↑</sup> Kann, Viggo (1992), *On the Approximability of NP-complete Optimization Problems*, Royal Institute of Technology, Sweden, ISBN 91-7170-082-X

## Xem thêm

Mục tiêu là sau đó tìm ra một số trường hợp *x* một lời giải tối i ưu, có nghĩa là, là một lời giải khả thi *y*

- Semi-infinite programming
- Bài toán quyết định
- Bài toán tìm kiếm
- Counting problem (complexity)
- Function problem

Lấy từ "[https://vi.wikipedia.org/w/index.php?title=Bài\\_toán\\_tối\\_ưu\\_hóa&oldid=65288275](https://vi.wikipedia.org/w/index.php?title=Bài_toán_tối_ưu_hóa&oldid=65288275)"

**Trang này được sửa đổi lần cuối vào ngày 31 tháng 7 năm 2021 lúc 04:03.**

Văn bản được phát hành theo Giấy phép Creative Commons Ghi công–Chia sẻ tương tự; có thể áp dụng điều khoản bổ sung. Với việc sử dụng trang web này, bạn chấp nhận Điều khoản Sử dụng và Quy định quyền riêng tư. Wikipedia® là thương hiệu đã đăng ký của Wikimedia Foundation, Inc., một tổ chức phi lợi nhuận.