2. Lý thuyết độ phức tạp (Complexity Theory)

Ngày 22 tháng 7 năm 2025

1 Khái quát

Lý thuyết độ phức tạp nghiên cứu lượng tài nguyên cần thiết để giải một bài toán tính toán, thường là thời gian (*time complexity*) hoặc bộ nhớ (*space complexity*).

Mối quan hệ giữa tài nguyên cần và kích thước đầu vào n là trọng tâm phân loại bài toán.

Mục tiêu: xác định liệu bài toán có thể giải được trong thời gian hợp lý hay không khi kích thước dữ liệu đầu vào tăng lên.

2 Các lớp độ phức tạp chính

2.1 Lớp P (Polynomial time)

Là tập hợp các bài toán quyết định (*decision problems*) có thể giải được trong thời gian đa thức theo kích thước đầu vào n.

Tồn tại thuật toán chạy trong thời gian $O(n^k)$ với một số hằng số k.

Ví dụ: kiểm tra tính liên thông của đồ thị, sắp xếp mảng.

2.2 Lớp NP (Nondeterministic Polynomial time)

Bao gồm các bài toán quyết định mà nếu có câu trả lời "có" (*yes instance*), thì ta có thể xác minh nó trong thời gian đa thức.

Không nhất thiết có thuật toán tìm lời giải nhanh, nhưng nếu được cung cấp "chứng minh" (*witness*), ta có thể kiểm tra tính đúng của lời giải.

Ví dụ: bài toán tìm bộ con kích thước k trong đồ thị (clique), hoặc bài toán giải mã khóa bí mật với bản rõ biết trước.

2.3 Lớp NP-Complete

Là tập các bài toán trong NP mà mọi bài toán NP khác đều có thể *giảm quy* (reduce) về bài toán này trong thời gian đa thức.

Nếu giải được một bài toán NP-Complete trong thời gian đa thức, thì tất cả các bài toán NP đều có thể giải nhanh.

Đây là các bài toán "khó nhất" trong lớp NP.

Ví dụ: bài toán chu trình Hamilton, bài toán đóng tập tối thiểu.

2.4 Lớp NP-Hard

Gồm các bài toán khó ít nhất như các bài toán NP-Complete, nhưng không nhất thiết phải thuộc NP.

Có thể là các bài toán không phải quyết định, như bài toán tối ưu hoặc thậm chí không có lời giải "có/không".

Ví dụ: một số trò chơi như *Tetris*, *Super Mario* được chứng minh là NP-hard.

3 Các bài toán quyết định (Decision Problems)

Là các bài toán đưa ra câu trả lời đơn giản "có" hoặc "không" dựa trên đầu vào.

Ví dụ: "Đồ thị G có chu trình clique kích thước k hay không?"

Các bài toán quyết định giúp phân loại các bài toán phức tạp khác thông qua khả năng giải quyết trong thời gian đa thức.

4 Giao thức tương tác trong ZKP và các lớp phức tạp mở rộng

Giao thức tương tác gồm một **prover** (người chứng minh) và một **verifier** (người kiểm chứng).

- Prover: có sức mạnh tính toán không giới hạn.
- Verifier: giới hạn ở tính toán đa thức và có thể sử dụng xác suất.

Các lớp phức tạp mở rộng với tính ngẫu nhiên:

- **Arthur–Merlin (AM) Protocols:** Arthur (verifier) là máy tính đa thức ngẫu nhiên; Merlin (prover) có tài nguyên vô hạn. Kiểu giao thức này cho phép verifier kiểm tra chứng minh với độ tin cậy cao qua vài bước tương tác.
- **MA (Merlin–Arthur):** Phiên bản đơn giản của AM, verifier chỉ nhận chứng minh một lần và kiểm tra với độ ngẫu nhiên.

Đặc điểm:

- *Completeness (Đầy đủ):* Nếu lời khẳng định đúng, prover có thể thuyết phục verifier chấp nhận với xác suất ít nhất $\frac{2}{3}$.
- *Soundness (Độ tin cậy):* Nếu lời khẳng định sai, không prover nào có thể thuyết phục verifier chấp nhận với xác suất lớn hơn $\frac{1}{3}$.

Public coin vs Private coin:

- **Public coin:** lựa chọn ngẫu nhiên của verifier được công khai.
- **Private coin:** lựa chọn ngẫu nhiên được giữ bí mật.

5 Ý nghĩa trong Zero Knowledge Proofs (ZKP)

ZKP là các giao thức tương tác để chứng minh một khẳng định đúng mà không tiết lộ bất kỳ thông tin nào ngoài việc khẳng định là đúng.

ZKP thường sử dụng các khái niệm trong lý thuyết độ phức tạp để chứng minh rằng verifier có thể xác minh một cách nhanh chóng dựa trên bằng chứng nhỏ gọn (certificate) hoặc đánh giá đa thức.

Tính hiệu quả của ZKP dựa trên khả năng verifier chạy thuật toán đa thức, trong khi prover có thể sử dụng sức mạnh tính toán không giới hạn.

Các lớp phức tạp mở rộng như AM, MA giúp mô hình hóa các loại giao thức tương tác trong ZKP.