

# نظریه پیچیدگی \*

علیرضا سلطانی نشان

۶ اسفند ۱۴۰۲

## فهرست مطالب

۱	مجاز	۱
۱	کلاس‌های پیچیدگی	۲
۲	۱.۲ کلاس P	۱۰.۲
۲	۱.۱.۲ ویژگی‌های کلاس P	۱۰.۱.۲
۲	۲.۱.۲ کاربردها	۲۰.۱.۲
۲	۲.۲ کلاس NP	۲۰.۲
۲	۱.۲.۲ ویژگی‌های کلاس NP	۱۰.۲.۲
۳	۲.۲.۲ کاربرد	۲۰.۲.۲
۳	۳.۲ کلاس Co-NP	۳۰.۲
۳	۱.۳.۲ مسائل	۱۰.۳.۲
۳	۴.۲ کلاس NP-Hard	۴۰.۲
۳	۱.۴.۲ ویژگی‌ها	۱۰.۴.۲
۴	۲.۴.۲ کاربرد	۲۰.۴.۲
۴	۵.۲ کلاس NP-Complete	۵۰.۲
۴	۱.۵.۲ کاربرد	۱۰.۵.۲
۴	۶.۲ قضیه $P = NP$	۶۰.۲

## ۱ مجوز

به فایل license همراه این برگه توجه کنید. این برگه تحت مجوز GPLV۳ منتشر شده است که اجازه نشر و استفاده (کد و خروجی/pdf) را رایگان می‌دهد.

## ۲ کلاس‌های پیچیدگی

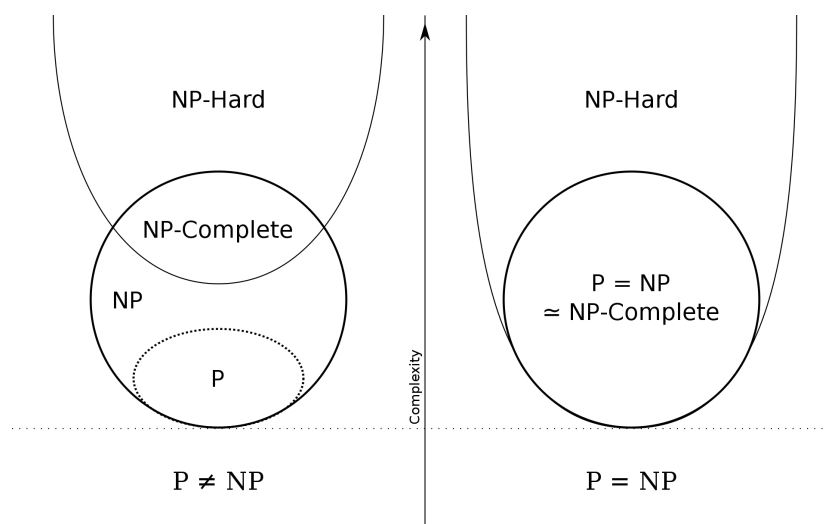
سطح دشواری یک مسئله را مشخص می‌کند. برای رسیدن به پاسخ یک مسئله چه دشواری‌هایی را خواهیم داشت. برای حل هر مسئله به تعداد پیش‌نیازها احتیاج داریم:  
منابع اصلی برای حل مسئله:

۱. زمان: مدت زمانی که می‌توانیم یک مسئله را دریافت کنیم و به پاسخ آن برسیم.

۲. فضا: برای حل مسئله چه پارامترهایی بیان شده است؟ چه میزان حافظه برای رسیدن به پاسخ نیاز داریم؟

پیچیدگی زمانی یا همان مرتبه اجرایی الگوریتم به منظور تعیین تعداد مراحل مورد نیاز برای حل مسئله می‌باشد. علاوه بر این، مرتبه اجرایی به منظور تعیین مدت زمان مورد نیاز برای تایید جواب استفاده می‌شود.  
درجه سختی سوال در نظریه پیچیدگی مطرح می‌شود، اینکه چه مسئله‌ای قابل حل است و چه مسئله‌ای را نمی‌توان حل کرد. معروف‌ترین کلاس‌های پیچیدگی P و NP می‌باشد که مسئله را از نظر زمان مورد نیاز تقسیم می‌کنند. بقیه کلاس‌ها را می‌توانید در شکل ۱ مشاهده کنید.

\*Computational complexity theory



شکل ۱: دیاگرام اوایلر مسائل کلاس‌های پیچیدگی؛ از پایین به بالا ساده، متوسط، سخت، سخت‌ترین

## ۱.۲ کلاس P

P یک کلاس پیچیدگی است که مجموعه‌ای از تصمیم‌گیری‌هایی که می‌توان در زمان حل یک معادله چند جمله‌ای داشت را نشان می‌دهد. معمولاً پاسخ این مسائل به صورت بلی یا خیر بیان می‌شود و توسط ماشین قطعی و در زمان چندجمله‌ای قابل حل است. برای مثال مسئله زیر می‌تواند مطرح شود:

گراف G را داریم، آیا می‌توان رئوس این گراف را به گونه‌ای رنگ آمیزی کرد که هیچ لبه‌ای تک رنگ نباشد؟ الگوریتم آن در نظریه گراف: از یک راس دلخواه شروع می‌کنیم. آن را قرمز می‌کنیم و تمام رئوس همسایه را آبی می‌کنیم و ادامه می‌دهیم. این عمل را تا جایی انجام می‌دهیم که یال کشیده شده بین رئوس جدید تک رنگ نباشد.

### ۱.۱.۲ ویژگی‌های کلاس P

۱. یافتن راه‌حل برای مسائل P آسان است.
۲. مسائل P اغلب قابل حل و رام‌پذیر (Tractable) هستند. مسئله رام‌پذیر مسئله‌ای است که هم به صورت نظری هم به صورت عملی قابل حل می‌باشد. اما مسئله غیر رام‌پذیر دقیقاً مسئله‌ای است که تنها در تئوری قابل حل و اثبات می‌باشد اما در عمل غیر قابل حل

## ۲.۱.۲ کاربردها

۱. محاسبه بزرگ‌ترین مقسوم علیه مشترک
۲. یافتن حداکثر تطابق
۳. نسخه‌های تصمیم برنامه‌ریزی خطی

## ۲.۲ کلاس NP

NP<sup>۱</sup> یک کلاس پیچیدگی است که مجموعه از پاسخ‌ها را به همراه مدارک و اثبات حل مسئله را در صورتی که پاسخ مسئله بله باشد را ارائه می‌دهد. این مسائل توسط ماشین غیرقطعی در زمان چند جمله‌ای قابل حل می‌باشد. برای مثال مسئله زیر می‌تواند مطرح شود: فرض کنید که یک لغتنامه با تعداد کلمات محدود دارید. می‌خواهیم با استفاده از ترکیب این لغات به یک کلمه ۱۰ حرفی مشخص برسیم. آیا این ترکیب امکان پذیر است؟ اگر راه‌حل این مسئله مشخص است، آن را می‌نویسیم، اما اگر مشخص نیست تماماً باید با مدارک و شواهد ثابت گردد که این لغتنامه محدود مناسب ساخت این ترکیب نیست.

### ۱.۲.۲ ویژگی‌های کلاس NP

۱. یافتن جواب برای مسائل کلاس NP سخت است، زیرا توسط ماشین غیرقطعی حل می‌شوند، اما جواب‌ها به سادگی قابل تایید هستند.
۲. مسائل NP را می‌توان توسط ماشین تورینگ در زمان چندجمله‌ای تایید کرد.

<sup>۱</sup> Non-Deterministic Polynomial Time

به عنوان مثالی دیگر در مورد کلاس NP می‌توان به مورد زیر اشاره کرد:  
تصور کنید که یک شرکت کلاً ۱۰۰۰ کارمند دارد که می‌خواهد آن‌ها را در ۲۰۰ اتاق تقسیم کند. مدیر عامل این شرکت اطلاعات برخی از کارمندان را دارد که می‌داند به دلیل شخصی برخی کارمندان نمی‌توانند در یک اتاق کار کنند.  
بررسی اینکه ترکیب کارمندان در یک اتاق برای هر یک از همکاران در آن اتاق رضایت‌بخش است یا نه، کار آسانی است. اما ایجاد چنین فهرستی از صفر آنقدر سخت است که کاملاً غیرعملی خواهد بود.

## ۲.۲.۲ کاربرد

این کلاس شامل بسیار از مسائل از جمله موارد زیر است:

۱. مسئله رضایتمندی بولی (SAT)

۲. مسئله مسیر همیلتونی

۳. رنگ آمیزی گراف

## ۳.۲ کلاس Co-NP

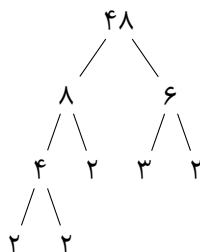
کلاس Co-NP<sup>۲</sup> به معنی مکمل کلاس NP است. اگر پاسخ یک مسئله در این کلاس خیر باشد، بنابراین اثباتی وجود دارد که می‌توان آن را در یک زمان چندجمله‌ای بررسی کرد.

تشخیص این کلاس نسبت به NP ساده می‌باشد. اگر در یک مسئله X در کلاس NP باشد، در این صورت مکمل آن  $\overline{NP}$  در کلاس CoNP قرار خواهد داشت. برای یک مسئله NP و  $\overline{NP}$  نیازی به تایید همه پاسخ‌ها به طور همزمان در زمان چندجمله‌ای نیست، صرفاً باید پاسخی معین یعنی بله یا خیر در زمان چندجمله‌ای تایید شود.

## ۱.۳.۲ مسائل

۱. بررسی عدد اول

۲. تجزیه اعداد صحیح به اعداد اول (Integer Factorization) شکل ۱.۳.۲



شکل ۲: تجزیه اعداد صحیح به اعداد اول  $48 = 2x2x2x2x3$

## ۴.۲ کلاس NP-Hard

کلاس NP-Hard سخت‌ترین سطح مسئله در کلاس NP می‌باشد. کلاس NP به گونه‌ای است که هر مسئله در NP به NP Hard قابل تبدیل و کاهش پذیر است.

## ۱.۴.۲ ویژگی‌ها

۱. همه مسائل NP Hard در کلاس NP وجود ندارند.

۲. بررسی مسئله NP Hard زمان زیادی می‌برد. این بدان معناست که اگر جوابی برای یک مسئله NP Hard ارائه شود، بررسی درستی یا نادرستی آن زمان زیادی طول می‌کشد.

۳. اگر مسئله L واقع در کلاس NP را بتوان در زمان چندجمله‌ای به مسئله A تبدیل کرد (تجزیه و کاهش داد) در این صورت مسئله A یک مسئله NP Hard محسوب می‌شود (خورد کردن و شکستن مسئله).

<sup>۲</sup> Complement of NP Class

## ۲.۴.۲ کاربرد

۱. مسئله توقف Halting problem

۲. فرمول بولی کمی واقعی True Quantified Boolean Formula

۳. دور هملیتونی در گراف جهت‌دار

## ۵.۲ کلاس NP-Complete

یک مسئله زمانی NP Complete است که هم در کلاس NP باشد و هم در کلاس NP Hard. مسائل NP Complete مسائل سختی در NP محسوب می‌شوند.

## ۱.۵.۲ کاربرد

۱. مسئله کوله پشتی

۲. مسئله دور هملیتونی

۳. رضایت‌پذیری (Satisfiability)

۴. پوشش گره‌ای (Vertex cover)

## ۶.۲ قضیه $P = NP$

این قضیه یکی از برترین و برجسته‌ترین مسائل ریاضی و کامپیوتری است به گونه‌ای که موسسه Clay یک میلیون دلار برای یافتن یک جواب صحیح برای این مسئله تعیین کرده است. واقعیت این است که می‌دانیم مسائل P زیر مجموعه‌ای از کلاس NP است. سوالی که ممکن است پیش آید این است که آیا مسائل NP راه‌حل‌های زمانی چند جمله‌ای قطعی دارند یا غیرقطعی؟ تا حد زیادی اعتقادات بر این است که به این شکل نمی‌باشد. وضعیت این سوال را می‌توانید در این سایت بررسی کنید.