دانشگاه فردوسی مشهد

الگوریتم پیشرفته

۲۲ آذر

دکتر حامد فهیمی

دانشکده علوم ریاضی

مقدمه‌ای بر پیچیدگی محاسباتی و مسائل NP

در این جلسه، به بررسی دسته‌ای از مسائل می‌پردازیم که حل آن‌ها در زمان چندجمله‌ای (Polynomial Time) بسیار دشوار به نظر می‌رسد. این مسائل به عنوان مسائل NP-Complete و NP-Hard شناخته می‌شوند و بخش مهمی از علوم کامپیوتر نظری را تشکیل می‌دهють.

کلاس‌های پیچیدگی P و NP

برای شروع، دو کلاس مهم از مسائل تصمیم‌گیری را تعریف می‌کنیم:

- **کلاس P:** شامل مسائلی است که می‌توان آن‌ها را با یک الگوریتم قطعی (Deterministic) در زمان چندجمله‌ای حل کرد. به عبارت دیگر، اگر اندازه ورودی n باشد، زمان اجرا از مرتبه $O(n^k)$ برای یک ثابت k است. این مسائل، مسائلی هستند که به طور کارآمد قابل حل هستند.
- **کلاس NP:** شامل مسائلی است که اگر یک "گواه" یا "راه‌حل کاندید" برای آن‌ها به ما داده شود، می‌توانیم صحت آن را در زمان چندجمله‌ای بررسی کنیم. نام NP مخفف Nondeterministic Polynomial time (زمان چندجمله‌ای غیرقطعی) است. به طور شهودی، این‌ها مسائلی هستند که "بررسی" راه‌حل آن‌ها آسان است، حتی اگر "پیدا کردن" راه‌حل دشوار باشد.

واضح است که $P \subseteq NP$ است، زیرا اگر بتوانیم مسئله‌ای را در زمان چندجمله‌ای حل کنیم، می‌توانیم یک گواه را نادیده گرفته و مسئله را حل کنیم تا صحت آن را بررسی نماییم. سوال بزرگ و حل‌نشده در علوم کامپیوتر این است که آیا $P = NP$ است یا خیر.

مسئله مسیر و دور همیلتونی (Hamiltonian Path and Cycle)

یکی از مسائل کلاسیک در کلاس NP، مسئله **دور همیلتونی** است.

> **تعریف دور همیلتونی:** در یک گراف G=(V, E)، یک دور همیلتونی، دوری است که از هر رأس گراف دقیقاً یک بار عبور می‌کند و به رأس شروع بازمی‌گردد.

مسئله تصمیم‌گیری مرتبط با آن به این صورت تعریف می‌شود:

HAM-CYCLE: آیا گراف G دارای دور همیلتونی است؟

مسئله‌ی نزدیک به آن، **مسیر همیلتونی** است.

> **تعریف مسیر همیلتونی:** در یک گراف G=(V, E)، یک مسیر همیلتونی، مسیری است که از هر رأس گراف دقیقاً یک بار عبور می‌کند (اما لزومی ندارد به نقطه شروع بازگردد).

هر دو مسئله مسیر و دور همیلتونی، **NP-Complete** هستند. این به این معناست که نه تنها در NP قرار دارند، بلکه از سخت‌ترین مسائل این کلاس نیز هستند. اثبات NP-Complete بودن آن‌ها معمولاً از طریق کاهش (Reduction) از یک مسئله NP-Complete دیگر مانند SAT-3 انجام می‌شود.

خلاصه‌ای از نکات کلیدی این بخش در یادداشت‌های زیر آمده است:

> **نکات اصلی از یادداشت‌ها:**
> - اثبات NP-Complete بودن یک مسئله (مانند دور همیلتونی) معمولاً از طریق کاهش از یک مسئله شناخته‌شده دیگر (مانند SAT-3) انجام می‌شود.
> - پیدا کردن مسیر همیلتونی یک مسئله NP-Complete است.
> - برای اینکه ثابت کنیم مسئله مسیر همیلتونی در P است، باید یک الگوریتم زمان چندجمله‌ای برای آن پیدا کنیم که برای هر ورودی، وجود یا عدم وجود مسیر را مشخص کند. این کار در زمان چندجمله‌ای امکان‌پذیر به نظر نمی‌رسد.
> - اگر گرافی دارای رأس برشی (Cut Vertex) باشد، و حذف آن گراف را به بیش از دو مؤلفه همبندی بشکند، آن گراف دور همیلتونی ندارد.
> - مسائل NP، مسائلی هستند که می‌توان جواب آن‌ها را (در صورت وجود) به سادگی و در زمان چندجمله‌ای تأیید (Verify) کرد.

کلاس Co-NP

یک کلاس مهم دیگر در نظریه پیچیدگی، **Co-NP** است. این کلاس به صورت تنگاتنگی با NP مرتبط است.

> **تعریف Co-NP:** یک زبان L در کلاس Co-NP قرار دارد اگر و تنها اگر مکمل آن ($\bar{L}$) در کلاس NP باشد.

به عبارت دیگر، مسائل Co-NP مسائلی هستند که برای نمونه‌های "نه" (No instances) آن‌ها، یک گواه کوتاه و قابل بررسی در زمان چندجمله‌ای وجود دارد.

برای مثال، مسئله **TAUTOLOGY** را در نظر بگیرید. این مسئله می‌پرسد: "آیا یک فرمول بولی معین، یک توتولوژی است (یعنی به ازای تمام مقادیر ورودی، درست است)؟"

مکمل این مسئله، مسئله SAT است (آیا فرمولی وجود دارد که قابل صدق باشد؟). از آنجایی که SAT در NP قرار دارد، مسئله TAUTOLOGY در Co-NP است.

روابط بین کلاس‌های P، NP و Co-NP از سوالات اساسی حل نشده در این حوزه است:

- می‌دانیم که $P \subseteq NP \cap Co\text{-}NP$ است. زیرا P تحت مکمل‌گیری بسته است (اگر بتوان L را در زمان چندجمله‌ای حل کرد، مکمل آن را نیز می‌توان حل کرد).
- اما نمی‌دانیم که آیا $P = NP \cap Co\text{-}NP$ است یا خیر.
- همچنین نمی‌دانیم که آیا $NP = Co\text{-}NP$ است یا خیر.

یادداشت‌های زیر، به درک بهتر این روابط کمک می‌کنند:

> **نکات اصلی از یادداشت‌ها:**
> - مسئله "**توتولوژی**" (TAUTOLOGY) یک مثال از مسائل کلاس Co-NP است. این مسئله می‌پرسد آیا یک گزاره همواره درست است؟
> - مکمل مسئله TAUTOLOGY، مسئله SAT است که در کلاس NP قرار دارد. از این رو TAUT در Co-NP است.
> - دیاگرام‌ها سناریوهای ممکن برای رابطه بین P، NP و Co-NP را نشان می‌دهند:
> 1. **سناریوی اول (باور عمومی):** کلاس P یک زیرمجموعه محض از اشتراک NP و Co-NP است و این دو کلاس نیز با هم برابر نیستند. یعنی $P \subset (NP \cap Co\text{-}NP)$ و $NP \neq Co\text{-}NP$. 2. **سناریوی دوم:** کلاس‌های NP و Co-NP با هم برابرند، اما P همچنان زیرمجموعه محض آن‌هاست. یعنی $P \subset NP = Co\text{-}NP$. 3. **سناریوی سوم (کمترین احتمال):** هر سه کلاس با هم برابرند: $P = NP = Co\text{-}NP$.

راهکارهای مقابله با مسائل دشوار

از آنجایی که الگوریتم کارآمدی برای مسائل NP-Complete وجود ندارد (مگر اینکه P=NP باشد)، در عمل از راهکارهای دیگری برای مقابله با آن‌ها استفاده می‌شود. یکی از این راهکارها، الگوریتم‌های **انشعاب و حد (Branch-and-Bound)** است.

این روش، که نسخه‌ی بهبودیافته‌ای از **عقبگرد (Backtracking)** برای مسائل بهینه‌سازی است، فضای حالت مسئله را به صورت یک درخت جستجو می‌کند. ایده اصلی آن به این شرح است:

1. برای هر گره در درخت جستجو، یک **حد (Bound)** بر روی بهترین جواب ممکن که از آن گره قابل دستیابی است، محاسبه می‌شود (مثلاً حد پایین برای مسائل کمینه‌سازی).
2. مقدار بهترین جوابی که تا کنون پیدا شده، نگهداری می‌شود.
3. اگر حد یک گره، از بهترین جواب یافت‌شده تا آن لحظه بدتر باشد، آن شاخه از درخت جستجو هرس (Prune) می‌شود، زیرا نمی‌تواند به جواب بهینه منجر شود.

این روش تضمین می‌کند که جواب بهینه را پیدا می‌کند و در بسیاری از موارد، با هرس کردن بخش‌های بزرگی از فضای جستجو، بسیار سریع‌تر از جستجوی کامل عمل می‌کند.

موفق باشید.