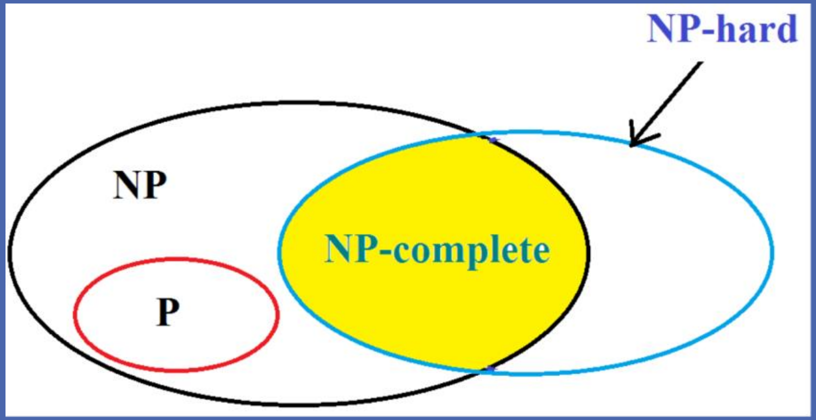
به نام خدا

نظریه پیچیدگی محاسباتی

Computational Complexity Theory



دانشکده مهندسی کامپیوتر – دانشگاه اصفهان
دکتر مرجان کائدی

مسائل تصمیم گیری (Decision making)

- هر مساله که جواب آن بله یا خیر باشد، یک مساله تصمیم گیری نامیده می شود.
- روی این مسائل تمرکز می کنیم چون اکثر مسائل دیگر، یک مساله تصمیم گیری معادل دارند.
- مثلا مسائل بهینه سازی را می توان در قالب مسائل تصمیم گیری سازماندهی کرد.

تقسیم مسائل به چهار کلاس

- مسائل تصمیم گیری به چهار کلاس (دسته) تقسیم می شوند:
  - کلاس P
  - کلاس NP
  - کلاس NP-hard
  - کلاس NP-complete

کلاس P

کلاس P (Polynomial)

- مسائلی که برای حل آنها الگوریتم هایی معین با مرتبه چند جمله ای موجود است.

- الگوریتم معین یا قطعی یعنی الگوریتمی که در آن نتیجه هر عمل به طور قطعی مشخص است.

نمونه هایی از مسائل دسته P

- جستجو
- مرتب سازی
- ضرب ماتریس
- درخت پوشای کمینه

کلاس NP

کلاس NP (Non-Polynomial)

- برای تعریف این کلاس، نیاز داریم که ابتدا الگوریتم نامعین را معرفی کنیم:

- دستور نامعین: دستوری که نتیجه آن از قبل قابل پیش بینی نیست (مثلا انتخاب عدد تصادفی بین ۰ و ۱)

- الگوریتم های نامعین: الگوریتم هایی که برای ماشین های نامعین طراحی می شوند.

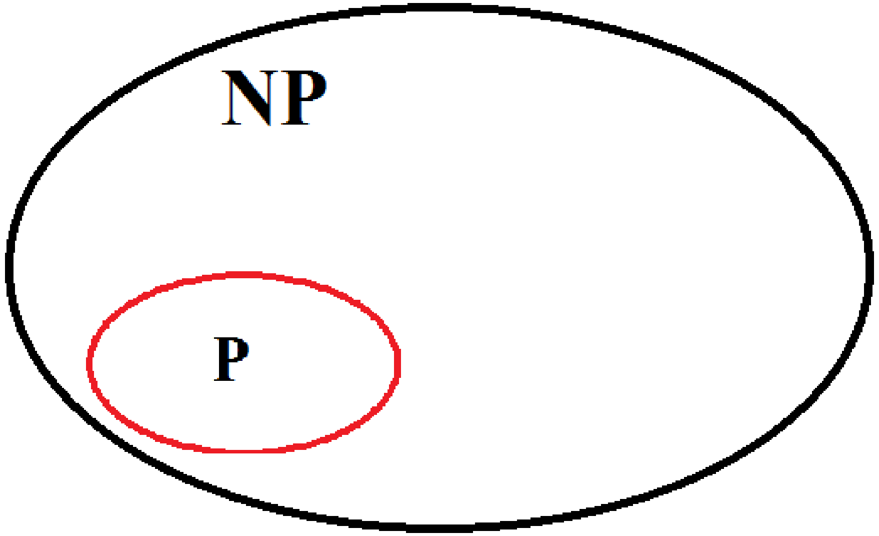
مثالی از یک الگوریتم نامعین

```
void  nd-search(A[], n, x)
{int i;
i=choice(1...n);
if (A[i]==x)
     {print (i);
     success();}
print(-1);
failure();
}
```

مرتبه این الگوریتم $O(1)$ است.

تعریف کلاس NP

- مسائل تصمیم گیری که به وسیله یک الگوریتم نامعین، با مرتبه زمانی چندجمله ای قابل حل هستند، در دسته NP قرار دارند.



بدیهی است که کلاس P زیرمجموعه NP است: P C NP

یک نمونه مسالهNP: مساله صدق پذیری دودویی (binary satisfiability)

- یک عبارت منطقی داده شده است.
- می خواهیم تعیین کنیم که آیا این عبارت منطقی به ازای مقادیری از متغیرها، true خواهد شد یا خیر.

- مثلا:

- (x1 AND x2) OR (X1′ AND X2′)
- (x1 OR x2) AND (X1′ OR X2′)
- (x1 AND x1′) OR (x2 AND X2′)

- الگوریتمی که مساله صدق پذیری را به صورت معین حل کند از مرتبه 2^n است که n تعداد متغیرها است.

- ولی الگوریتم نامعین برای این منظور از مرتبه n است.

یک الگوریتم نامعین برای مساله صدق پذیری

```
Void nd-satisfiability(E, n)
{boolean x[];
for (i=1; i<=n; i++)
    X[i]=choice (false, true);
if (E(x)=true)
    success();
else
    failure();
}
```

- چندین نسخه از این الگوریتم را می توان به طور موازی روی یک سیستم چند پردازنده ای اجرا کرد.

- اگر تعداد پردازنده ها به سمت بی نهایت برود، نتیجه اجرا مانند یک الگوریتم معین است.

نمونه هایی از مسائل دسته NP

- دور هامیلتونی
- رنگ آمیزی گراف
- صدق پذیری دودویی
- و .....

کلاس NP-hard

کلاس NP-hard

- برای تعریف این کلاس، ابتدا مفهوم کاهش پذیری را بیان می کنیم.
- اگر M1 وM2 دو مساله باشند، می گوییم مساله M1به مساله M2 کاهش می یابد (M1 ➔ M2) اگر:
- چنانچه برای مساله M2 الگوریتمی معین با مرتبه چندجمله ای یافت شود، برای M1 هم الگوریتمی معین با مرتبه چند جمله ای یافت خواهد شد.
- (به زبان ساده: یعنی M1 ساده تر از M2 است).
- مسائل NP-hard مسائلی هستند که مسائل صدق پذیری به آنها کاهش پیدا می کند
- (یعنی مسائل NP-hard مسائلی هستند که از صدق پذیری سخت تر هستند).

نمونه هایی از مسائل دسته NP-hard

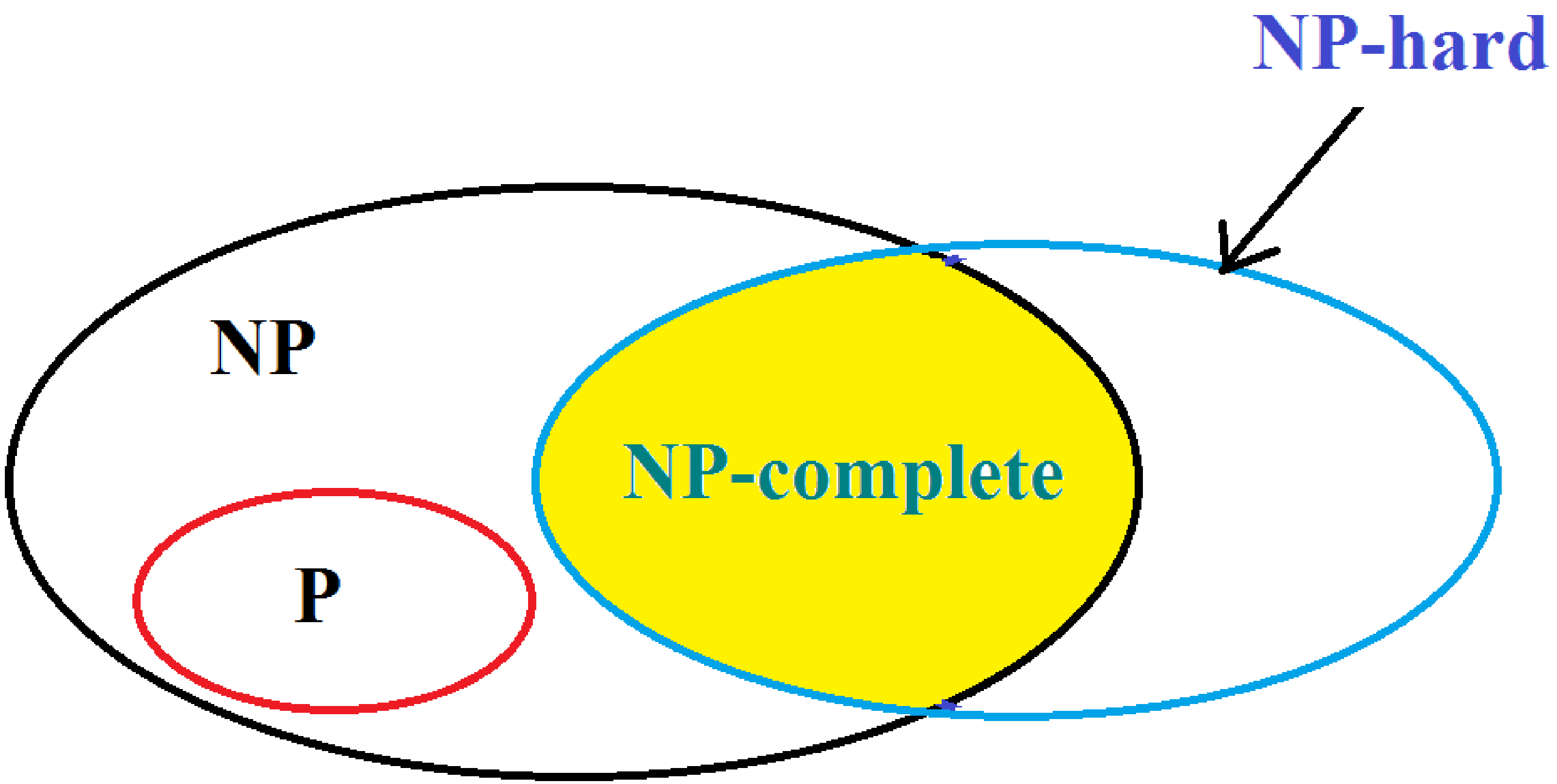
- فروشنده دوره گرد
- جمع زیرمجموعه ها
- مساله توقف پذیری (halt problem)
- و ....

کلاس NP- complete

کلاس NP- complete

- کلاس NP-complete فصل مشترک کلاس های NP و NP-hard است.

- یعنی مسائلی که از صدق پذیری سخت تر هستند ولی الگوریتم نامعینی با مرتبه چندجمله ای برای آنها وجود دارد.


NP-hard
NP
NP-complete
P

نمونه هایی از مسائل دسته NP-complete

- دور هامیلتونی
- کوله پشتی
- رنگ آمیزی گراف
- جمع زیرمجموعه ها
- همریختی دو گراف
- صدق پذیری دودویی
- و .....

پایان