P vs. NP

计算复杂性之一瞥

毛昕渝 2021/10/30

衡量计算的效率

“时间就是金钱，效率就是生命。”

——改革开放初期出现在深圳蛇口工业区的宣传标语

时间资源与空间资源

设有函数 $T: \mathbb{N} \rightarrow \mathbb{N}$.
我们说图灵机 $\mathbb{M}$在 T时间内计算了 f，如果对于任意输入$x \in \{0,1\}^*$, $\mathbb{M}$ 在不超过$T(|x|)$步内停机.

- $T = n$ 和 $T = 2n$ 区别大吗?
- 我们更关心运行时间的渐进增长
- $T(n) = O(n), T(n) = O(2^n)$ ...

设有函数 $S: \mathbb{N} \rightarrow \mathbb{N}$.
我们说图灵机 $\mathbb{M}$在S空间内计算了 f，如果对于任意输入$x \in \{0,1\}^*$, $\mathbb{M}$访问过的格子不超过$S(|x|)$个.

计算模型的影响

定理1. 如果图灵机$\mathbb{M}$ 在时间 T 内计算了问题f, 并且使用的符号表是Σ, 那么有一台图灵机$\widetilde{\mathbb{M}}$ 在T $4\log|\Sigma|\ T(n)$ 时间内计算f, 并且使用的符号表为 $\{0,1,\lhd,\Box\}$.

符号表不影响效率

定理2. 运行时间在$T(n)$内的k带图灵机可以被单带图灵机模拟，且运行时间为$O(T(n)^2)$.

多条带子对效率影响不大

定理3. 存在通用图灵机$\mathbb{U}$, 如果$\mathbb{M}_\alpha$运行时间为T，则$\mathbb{U}(\alpha,x)$在 $c_\alpha T^2$内停机.
其中c_α是只和α有关的常数.
*更高效的模拟： $c_\alpha T\log T$. (Hennie and Stearns [HS66])

存在高效的通用图灵机

Cobham-Edmonds Thesis

Church-Turing thesis

任何计算过程都可以被图灵机模拟。

对于可计算性而言，我们只需要考虑图灵机。

Cobham-Edmonds Thesis(also known as strong Church-Turing Thesis)

任何计算过程都可以被图灵机模拟，运行时间是原来运行时间的多项式。

- 如果某模型的运行时间为T，图灵机可以在$p(T)$的时间内模拟它，其中p是个多项式。
- 对于效率而言，我们只需要考虑图灵机。
- 问题有内在的复杂性，与计算模型无关。

P 和 PSPACE

- 我们只考虑判定问题。
- $\mathbf{TIME}\big(T(n)\big)$ 和 $\mathbf{SPACE}\big(S(n)\big)$
  - 如果存在图灵机$\mathbb{M}$在 $cT(n)$ 时间内判定语言L，则$L \in \mathbf{TIME}\big(T(n)\big)$.
  - 如果存在图灵机$\mathbb{M}$在 $cS(n)$ 空间内判定语言L，则$L \in \mathbf{SPACE}\big(T(n)\big)$.
- $\mathbf{P} \coloneqq \bigcup_{i=1}^{\infty} \mathbf{DTIME}\big(n^i\big)$
  - 一般认为，**P**是能高效解决的问题的集合。
- $\mathbf{PSPACE} \coloneqq \bigcup_{i=1}^{\infty} \mathbf{SPACE}(n^i)$.
- 显然，$\mathbf{P} \subseteq \mathbf{PSPACE}$.

P vs. NP

NP：能高效验证的问题的集合

NP 的定义

$\mathbb{V}$称为验证机

对于语言L , $L \in$ **NP** 当且仅当存在满足下列条件的图灵机$\mathbb{V}$和多项式p：
- $\mathbb{V}$ 在多项式时间内运行。
- 对任意的$x \in \{0\,,1\}^*$,

$$x \in L \text{ 当且仅当 } \exists y \in \{0\,,1\}^{p(|x|)}\ \mathbb{V}(x, y) = 1.$$

若 $\mathbb{V}(x, y) = 1$，则 y称是x的 证明（proof）或证书（certificate）

- 显然，**P** $\subseteq$ **NP**.
- NP语言可以在指数时间内判定：输入 x, 枚举所有$y \in \{0\,,1\}^{p(|x|)}$，检查是否 $\mathbb{V}(x, y) = 1$.
- **P** vs. **NP**：验证比解决问题（给出证明）更难吗?

Examples of **NP** languages

- $\texttt{SAT} \coloneqq \{\langle\varphi\rangle : \varphi \text{ is a satisfiable CNF}\}$.
  - Conjunction Normal Form
  - The certificate is a satisfying assignment.
- $\texttt{THEOREM} \coloneqq \{(\langle\varphi\rangle, 1^n) : \varphi \text{ has a proof of length} \le n\}$.
  - This is the finite version of Hilbert's *Entscheidungsproblem*.
- $\texttt{SubsetSum}$: Given n numbers $A_1, \ldots, A_n$ and a number T , decide if there is a subset of the numbers that sums up to T .
  - The certificate is the list of members in such a subset.
- ……

Design an algorithm determining whether a given mathematical statement has a proof.

NP完备性

归约的定义.

设有两个语言L, L'. 如果存在多项式时间内可计算的函数f 满足

$$x \in L \text{ 当且仅当 } f(x) \in L', \forall x \in \{0,1\}^*$$

我们说L可以Karp-归约到L'，记作$L \leq_K L'$.

NP完备问题

如果对任意的$L' \in \mathbf{NP}$ 都有$L' \leq_K L$，则称L是NP难的。
如果 L 既在 NP中，又是 NP难的，则称L是NP完备的。

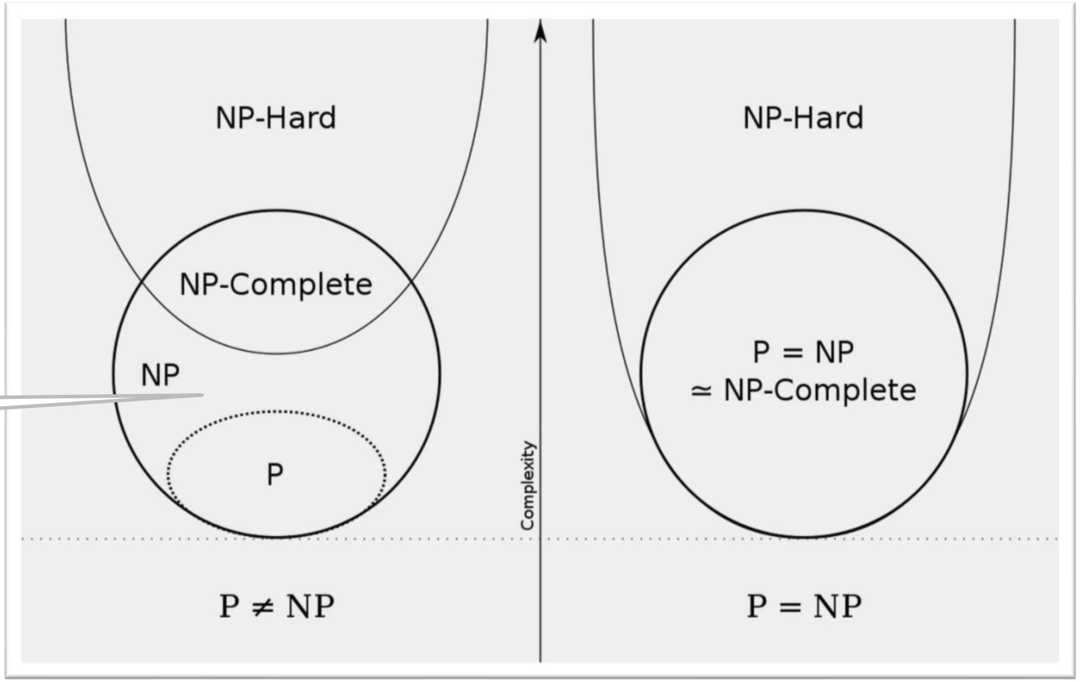
NP完备问题是NP中“最难的问题”。

Cook-Levin定理

SAT是 NP完备的。

NP 完备性与 P vs. NP

这中间有东西！



Ladner 定理

如果 $\mathbf{P} \neq \mathbf{NP}$, 则存在 $L \in \mathbf{NP} \setminus \mathbf{P}$ 不是NP完备的。

P = NP的世界

- 计算机可以完成大量的数学证明工作 ☺
- 最优的芯片设计 ☺
- 我们不再需要随机算法 ☺
- 无所不能的 AI ☺
- 密码学不复存在。这是一个没有隐私的世界 ☹
- ……

验证，证明与NP问题

什么是证明?

“任何一个一元复系数多项式方程都至少有一个复数根。”

真的是这样吗?

当然是这样，
愚蠢的人类。

《高等代数》

这些都是证明吗?

我还是去上线性
代数课吧....

Google

https://zh.wikipedia.org › zh-hans · Translate this page

代数基本定理- 维基百科，自由的百科全书

代数基本定理说明，任何一个一元複系数多项式方程都至少有一个複数根。也就是说，複數域是代数封闭的。 有时这个定理表述为：任何一个非零的一元n次複系数多项式，都 ...

证明（系统）=完备+无误

- 我们对于“证明”的要求是什么?
  - 完备（Completeness）：如果是真的，你总有办法说服我相信。
  - 无误（Soundness）：如果是假的，你无论如何也骗不过我。
  - 验证过程是高效的。

NP 的重新解释1

对于语言L，$L\in \mathbf{NP}$ 当且仅当存在满足下列条件的图灵机$\mathbb{V}$和多项式p:
- $\mathbb{V}$ 在多项式时间内运行。
- （完备）$\forall x\in L, \exists y\in\{0,1\}^{p(|x|)}$ $\mathbb{V}(x,y)=1.$
- （无误）$\forall x\notin L, \forall y\in\{0,1\}^{*}$ $\mathbb{V}(x,y)=0.$

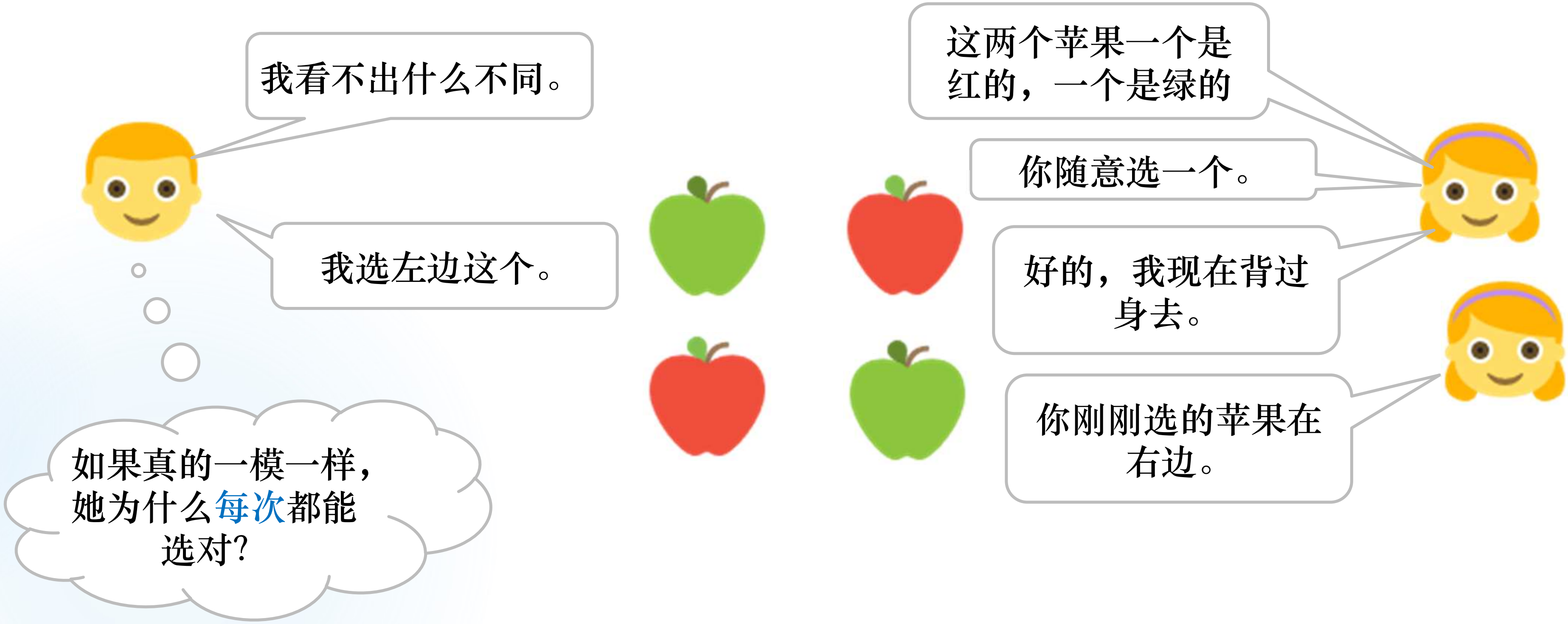
NP 的重新解释2

对于语言L，$L\in \mathbf{NP}$ 当且仅当存在满足下列条件的图灵机 $\mathbb{P},\mathbb{V}$:
- $\mathbb{V}$ 在多项式时间内运行。
- （完备）$\forall x\in L$, $\langle \mathbb{P},\mathbb{V}\rangle(x)=1.$
- （无误）$\forall x\notin L$, $\forall$图灵机 $\widetilde{\mathbb{P}}$, $\langle \widetilde{\mathbb{P}},\mathbb{V}\rangle(x)=0.$

如果$\mathbb{P},\mathbb{V}$的交互包含随机性呢?

$\mathbb{P}$的计算能力没有限制

如何让色盲相信苹果的颜色不同?



交互式证明（Interactive Proof）

你一定能从证明中学到东西吗?
这个证明是零知识（Zero-knowledge）的!

尾声

Don't think twice, it's alright.

Bob Dylan

最后的回望

- 中心问题：为什么有的问题难，而有的问题容易?
- 不知道…
  - 复杂性理论主要的成功在于问题的分类，类似化学中的元素周期表
  - 关于问题的复杂性、困难性的原因和本质，我们知之甚少
- 为什么这些很自然的问题这么难回答?
  - 或许是因为，这些问题体现了自然和生活本身的复杂和混沌
  - 这就是生活 ☺

> If people do not believe that mathematics is simple,
> it is only because they do not realize how complicated life is.
> John von Neumann

Thanks for listening. ☺