第八讲 计算复杂性理论
下

本讲提要

- 非确定图灵机与NP问题
- NP完全与典型的NP完全问题
- 计算复杂性理论在密码学中的应用

1.1非确定图灵机与NP问题

许多问题目前尚未找到多项式复杂性的解决方法，但是很容易设计出一个时间复杂性为多项式级的算法来验证问题解答的正确性。为此，我们需要修改确定图灵机为非确定图灵机来描述这类问题的算法。

1.1非确定图灵机与NP问题 (续)

例子1背包问题

给定一个集合$S=\{a_1,a_2,\cdots,a_n\}$和一个目标T，

其中$a_i \geq 0$，是否存在一个选择$V=[v_1,v_2,\cdots,v_n]$，

其中$v_i=0$或1，使得

$$\sum_{i=1}^{n} a_i v_i = T。$$

例如，集合S为{4,7,1,12,10}。对于目标和T=17存在一个解，即$17=4+1+12$。对应的选择向量$V=[1,0,1,1,0]$。而当$T=25$时无解。

1.1非确定图灵机与NP问题(续)

定义**1**非确定k带图灵机程序M由如下一个七元组构成

$M=(Q, q_0, q_f, T, I, b, \delta)$

其中，Q，q_0，q_f，T，I，b，δ的定义与上一讲确定图灵机的定义相同，δ是从$Q\times T^k$到$Q\times(T\times\{L, R, S\})^k$的一个映射，而且其中至少有一个映射是一对多的映射。

1.2 确定图灵机与RAM模型(续)

一个确定性k带图灵机程序(Program)可以详细描述为一个七元组：$TM=(Q, q_0, q_f, T, I, b, \delta)$,其中，$Q$是一个有限状态集合；$q_0 \in Q$称为初始状态；$q_f \in Q$称为终止状态或接收状态。$T$是带格中符号的有限集合；$I$是输入字符集，且$I \subset T$；$b$是$T$中的唯一空符，有$b \in T-I$；$\delta$称做图灵机的下移函数或有限状态控制函数，是从$Q \times T^k$的某一个子集到$Q \times (T \times \{L, R, S\})^k$的映射函数，即对由一个当前状态和$k$条带上扫描到的当前符号所构成的一个$k+1$元组，它唯一地给出一个新的状态和$k$个序偶，而每一个序偶由一个新的带符号和带头移动方向组成。

1.2 确定图灵机与RAM模型(续)

假定某台图灵机的下移函数表中有一个定义式为
$\delta(q_1, a_1, a_2, \ \ldots, a_k)=[q', (a'_1, d_1), (a'_2, d_2), \ldots, (a'_k, d_k)]$。

当图灵机处于状态q且对一切$1\leq i\leq k$，第i条带的带头扫描当前方格中的符号正好是a_i时，图灵机就按这个下移函数定义式所规定的内容进行如下工作：

（1）把i条带头下当前方格中的符号a_i清除并写上新的带符号a'_i，$1\leq i\leq k$。

（2）按d_i指出的方向移动各带的带头。这里，$d_i=L$表示带头往左移一格，$d_i=R$表示带头往右移一格，$d_i=S$表示带头不动。

（3）将图灵机的当前状态q改为q'。

1.1非确定图灵机与NP问题 (续)

一个一对多的映射意味着，从当前状态q和当前扫描的k个带符号x_1，x_2,…，x_k，可以选择新状态、新的带符号和带头移动的多种组合，即可以定义

$$\delta(q,x_1,x_2,\cdots,x_k)=\left\{\begin{array}{c}(q_1,(a_{11},d_{11}),(a_{12},d_{12}),\cdots,(a_{1k},d_{1k}))\\(q_2,(a_{21},d_{21}),(a_{22},d_{22}),\cdots,(a_{2k},d_{2k}))\\\vdots\\(q_r,(a_{r1},d_{r1}),(a_{r2},d_{r2}),\cdots,(a_{rk},d_{rk}))\end{array}\right\}$$

式中$r\geq 2$。图灵机执行时，每次可以选择（猜测）这r种新状态、新带符号和带头移动的某一固定组合。

1.1非确定图灵机与NP问题(续)

定义2非确定图灵机的时间复杂性$T(n)$是指，假若对于任何长度为n的可接收的输入ω，都存在着一条导致接收状态的计算序列，该序列至多有$T(n)$步。带头移动过程中任何一条带上被扫描到的不同方格数的总和不超过$S(n)$，则$S(n)$定义为该台非确定图灵机的空间复杂性。

1.1非确定图灵机与NP问题(续)

定理1 设$L(M)$是一台非确定图灵机M所可以接收的语言，M的时间复杂性是$T(n)$。那么，必存在一台确定图灵机M'，它所接收的语言$L(M')=L(M)$且时间复杂性是$O(C^{T(n)})$，其中C是某个正常数。

若存在一个多项式$p(x)$，使得对于所有的$n>1$，有$T(n)\leq p(n)$，则称非确定图灵机程序M为一个多项式时间非确定图灵机程序。

1.1非确定图灵机与NP问题(续)

定义**3**NP类为：

NP=$\{L$：存在一个多项式时间非确定图灵机程序M，使得L能被M识别，即$L=L_M\}$。

定理2若$\Pi \in$NP，则存在一个多项式p，使得Π可以用一个时间复杂性为$O(2^{p(n)})$的确定性算法来求解。

公开困难问题：P问题是否等于NP问题？

1.2 NP完全与典型的NP完全问题

定义**4**称语言L_1在多项式时间内可以归约为语言L_2，写做$L_1 \leq_p L_2$，是指存在一个多项式时间可计算的函数f：$\{0, 1\}^* \rightarrow \{0, 1\}^*$，满足对所有的$x \in \{0, 1\}^*$，都有

$$x \in L_1 \text{当且仅当} x \in L_2$$

称函数f为归约函数，计算f的多项式时间算法F称为归约算法。

引理**1**如果L_1，$L_2 \subseteq \{0, 1\}^*$且满足$L_1 \leq_p L_2$的语言，如果$L_2 \in \mathrm{P}$则$L_1 \in \mathrm{P}$。

1.2 NP完全与典型的NP完全问题(续)

定理3多项式归约具有如下基本性质：如果$L_1 \leq_p L_2$，$L_2 \leq_p L_3$，则$L_1 \leq_p L_3$。

定理3证明.

令f_1：$\{0, 1\}^* \to \{0, 1\}^*$为$L_1$到$L_2$的一个多项式变换，而$f_2$：$\{0, 1\}^* \to \{0, 1\}^*$为$L_2$到$L_3$的一个多项式变换。那么对于所有的$x \in L_1$，由$f(x)=f_2(f_1(x))$所定义的函数$f$：$\{0, 1\}^* \to \{0, 1\}^*$，就是从$L_1$到$L_3$所需要的变换，显然$f(x) \in L_3$当且仅当$x \in L_1$，而$f$的值可以由一个多项式时间确定图灵机程序来计算。

1.2 NP完全与典型的NP完全问题(续)

语言$L \subseteq \{0, 1\}^*$，如果对一个$L' \in$NP，有$L' \leq_p L$，则L也是NP困难的。

定义**5**语言$L \in$NP，如果对每一个$L' \in$NP，有$L' \leq_p L$，则称L是NP完全的。

定理4 如果任何NP完全问题是多项式时间可求解的，则P=NP。等价地，如果NP中的任何问题不是多项式时间可求解的，则所有NP完全问题都不是多项式时间可求解的。

1.2 NP完全与典型的NP完全问题(续)

定理4证明.

假定$L \in$P，并且$L \in$NP完全问题。对任意的$L' \in$NP，由定义5有，$L' \leq_p L$。这样，根据引理1，就有$L' \in$P。这样就证明了本定理的第一个结论。第二个结论是第一个结论的逆否命题。因此，第二个结论也得以证明。

$\leq_p$符号的说明.

（1）通过$\leq_p$符号，一个算法复杂度猜想是：P类<NP类<NP完全类。

（2）$\leq_p$可为问题建立等价类和偏序关系。

1.2 NP完全与典型的NP完全问题(续)

定理5若L_1和L_2都属于NP，L_1是NP完全的，且$L_1 \leq_p L_2$，那么L_2必是NP完全的。

定理5证明.

因为$L_2 \in$NP，所以下面只需要证明对每个$L' \in$NP，有L'可通过多项式归约到L_2。考虑任一$L' \in$NP，因为L_1是NP完全的，故必有L'多项式归约到L_1即$L' \leq_p L_1$。那么，根据定理3的传递性，有$L' \leq_p L_2$，所以L_2是NP完全的。

1.2 NP完全与典型的NP完全问题(续)

定理6（Cook定理）布尔表达式的可满足性问题是NP完全的。

于是Cook定理之后证明一个问题Q的NP完全性可由以下三步组成：

（1）证明问题Q属于NP；

（2）选择一个已知的NP完全问题Q′；

（3）构造从Q′到Q的多项式变换函数*f*。

1.2 NP完全与典型的NP完全问题(续)

在布尔表达式中，一个文字是指变量或者变量的“非”；一个子句是一个或多个文字的“或”。多个子句进行“与”，所得到的布尔表达式称为合取范式 (Conjunctive Normal Form, CNF)。如果公式中每个子句恰好有三个不同的变量，则该公式称为3-合取范式，即 3-CNF 。例如，$(x_1 \vee \neg x_1 \vee x_2) \wedge (x_3 \vee x_2 \vee x_4) \wedge (\neg x_3 \vee \neg x_2 \vee \neg x_4)$ 。

定理7 3-合取范式形式的布尔表达式的可满足性问题是NP完全的。

1.2 NP完全与典型的NP完全问题(续)

无向图$G=(V, E)$中的团是一个顶点子集$V' \subseteq V$，其中每一对顶点之间都由E中的一条边相连。换句话说，一个团是G的一个完全子图。团的规模是指它所包含的顶点数。团问题就是关于寻找图中规模最大的团的最优化问题。一个重要的判断问题是：在图中是否存在一个给定规模为k的团，其定义为

CLIQUE=$\{<G, k>$：G是具有规模为k的团的图$\}$。

定理8 团问题是NP完全的。

1.2 NP完全与典型的NP完全问题(续)

无相图$G=(V, E)$的顶点覆盖(Vertex Cover)是指子集$V'\subseteq V$，满足对任意$(\mu, \nu)\in E$，则$\mu\in V'$或$\nu\in V'$成立。亦即，每个顶点“覆盖”与其关联的边，G的顶点覆盖E中所有边的顶点组成的集合，顶点覆盖的规模是指它所包含的顶点数目。

顶点覆盖问题(Vertex Cover Problem)是指在给定的图中，找出具有最小规模的顶点覆盖。把这一最优化问题重新表述为判定问题，即确定一个图是否具有一个给定规模k的顶点覆盖。其定义为

VERTEX-COVER={<G, k>：G是具有规模为k的顶点覆盖}

定理9顶点覆盖问题是NP完全的。

1.2 NP完全与典型的NP完全问题(续)

其它NP完全问题

（1）哈密顿回路问题。

（2）货郎担问题。

1.3计算复杂性理论在密码学中的应用

1.3.1 基于计算复杂度安全的基本思想

（1）仅仅在对抗“有效”敌手时，安全存在，“有效”是指在可行的计算时间内运行。

（2）敌手成功的概率非常小（小到可以不关心它是否真的会发生）。

1.3计算复杂性理论在密码学中的应用（续）

1.3.1 基于计算复杂度安全的基本思想（续）

采用渐进方法来刻画安全的概念，于是有：

（1）将“可行的策略”或者“有效的算法”的概念，和在以n为参数的多项式时间内运行的概率算法等同看待，这里可用PPT表示概率多项式时间。这意味着，对于常量a、c，安全参数为n，该算法的运行时间为$a \cdot n^c$。要求诚实方在多项式时间内运行方案，并且仅关心多项式时间的敌手。而且，如果是超多项式时间的攻击策略，则不被认为是在现实中有威胁的。

1.3计算复杂性理论在密码学中的应用（续）

1.3.1 基于计算复杂度安全的基本思想（续）

（2）将“小的成功概率”的概念和成功概率小于任何以n为参数的多项式倒数等同看待，这意味着，对于每个常量c，当n的值足够大，敌手成功的概率小于n^{-c}。比任何多项式的倒数增长得都慢的函数被称为可忽略的（negligible）。

1.3计算复杂性理论在密码学中的应用（续）

1.3.1 基于计算复杂度安全的基本思想（续）

例子**2** 假设有一个安全方案，运行了n^3分钟的敌手能够成功“攻破该方案”的概率为$2^{40}\cdot 2^{-n}$（这是一个关于参数n可以忽略的函数）。当$n\leq 40$，意味着一个运行40^3分钟（大约6周）的敌手能够攻破该方案的概率为1，所以那样的n值不能用。甚至当$n=50$，一个运行50^3分钟（大约3个月）的敌手能够攻破该方案的概率约为1/1000，这也是不能接受的。而当$n=500$时，一个运行时间超过200年的敌手攻破该方案的概率仅仅为2^{-500}。

1.3计算复杂性理论在密码学中的应用（续）

1.3.1 基于计算复杂度安全的基本思想（续）

有效计算：前面已经讨论有效计算是指能够在PPT内执行的计算。一个算法A被认为在多项式时间内运行是指：如果存在一个多项式$p(\cdot)$，使得对于每个输入$x\in\{0,\ 1\}^*$，$A(x)$的计算最多在$p(|x|)$个步骤内终止（这里$|x|$表示的是字符串x的长度）。

1.3计算复杂性理论在密码学中的应用（续）

1.3.1 基于计算复杂度安全的基本思想（续）

定义**6**函数f为可忽略的，如果对于每个多项式$p(\cdot)$，存在一个N，使得对于所有的整数$n>N$，都满足$f(n)<1/p(n)$。

可忽略的成功概率：对某个多项式p，如果一个敌手能够成功地攻破该方案的概率为$1/p(n)$，则该方案被认为是不安全的。但是，对任意多项式p，如果攻破该方案的概率渐进小于$1/p(n)$，则认为该方案是安全的。

定义**7**如果每个PPT敌手仅以可忽略的概率成功攻破一个方案，那么该方案是安全的。

1.3计算复杂性理论在密码学中的应用（续）

1.3.1 基于计算复杂度安全的基本思想（续）

例子3函数$2^{-n}, 2^{-\sqrt{n}}$，$n^{-\log_2 n}$都是可忽略的。但是，它们接近零的速度是不同的。下面计算什么样的n值，会使得每一函数的值都小于10^{-6}。

(1) $2^{20} = 1048576$，因此对于$n \geq 20$，有$2^{-n} < 10^{-6}$。

(2) $2^{\sqrt{400}} = 1048576$，因此对于$n \geq 400$，有$2^{-\sqrt{n}} < 10^{-6}$。

(3) $32^5 = 33554432$，因此对于$n \geq 32$，有$n^{-\log_2 n} < 10^{-6}$。

1.3计算复杂性理论在密码学中的应用（续）

1.3.1 基于计算复杂度安全的基本思想（续）

定理10令$negl_1$和$negl_2$为可忽略函数，则：

（1）如果函数$negl_3$定义为$negl_3(n)=negl_1(n)+negl_2(n)$，则该函数也是可忽略的。

（2）对任何正多项式p，如果函数$negl_4$定义为$negl_4(n)=p(n)\cdot negl_1(n)$，则该函数也是可忽略的。

定义8如果每个PPT敌手A执行某个特定类型的攻击，对于每个多项式$p(\cdot)$，存在一个整数N，对于$n>N$，A在这次攻击中成功的概率小于$1/p(n)$，则认为该方案是安全的。

1.3计算复杂性理论在密码学中的应用(续)

1.3.2基于计算安全的加密方案

安全论证（约减）的具体方法：

（1）假定某个PPT敌手A攻击方案Π，将敌手的成功概率表示为$\varepsilon(n)$。

（2）构造一个叫做“归约”的有效算法A'，该算法将敌手A作为子程序来使用，试图解决难题X。这里指定难题X的某个输入实例x，有效算法A'将会对敌手A模拟出一个Π的实例，满足敌手A与Π交互。更准确地说，当敌手A作为子程序被有效算法A'运行时，它的分布应该和当A自身与Π交互时的分布是相同的；如果敌手A成功攻破了由A'模拟的Π的实例，则这将允许A'解决给出的难题实例x，成功的概率至少为多项式的倒数，即$1/p(n)$。

1.3计算复杂性理论在密码学中的应用(续)

1.3.2基于计算安全的加密方案(续)

（3）如果$\varepsilon(n)$不是可忽略的，则A'解决难题X的概率为不可忽略的概率$\varepsilon(n)/p(n)$。这与X是难题的假设矛盾。

（4）因而结论是，给定一个关于X是难题的假设，不存在PPT的敌手A能够以不可忽略的概率成功攻破Π，即Π是计算上安全的。

1.3计算复杂性理论在密码学中的应用(续)

1.3.2基于计算安全的加密方案(续)

定义9 一个对称密钥加密方案是PPT算法(Gen, Enc, Dec)的三元组。

（1）密钥生成算法Gen的输入为安全参数1^n，输出为密钥k：记为$k \leftarrow \text{Gen}(1^n)$（Gen是一个随机算法）。不失一般，假设任何由$\text{Gen}(1^n)$输出的密钥$k$，都满足$|k| \geq n$。

（2）加密算法Enc将密钥k和明文消息$m \in \{0,1\}^*$作为输入，并且输出一个密文c。因为Enc可能是随机化的算法，可记为$c \leftarrow \text{Enc}_k(m)$。

1.3计算复杂性理论在密码学中的应用(续)

1.3.2基于计算安全的加密方案(续)

定义9（续）

（3）解密算法Dec将密钥k和密文c作为输入，输出消息m。假设Dec是确定性的，可记为$m:=\mathrm{Dec}_k(c)$。

如果方案(Gen, Enc, Dec)满足：对每个n、由$\mathrm{Gen}(1^n)$输出的密钥k，算法Dec_k只对消息$m\in\{0,1\}^{l(n)}$有定义，则可以说(Gen, Enc, Dec)是一个消息长度为定长$l(n)$的对称密钥加密方案。

1.3计算复杂性理论在密码学中的应用(续)

1.3.2基于计算安全的加密方案(续)

窃听不可区分实验$\mathrm{PrivK}^{\mathrm{eav}}_{\mathrm{A},\Pi}(n)$：

(1) 输入1^n给敌手A，A输出一对长度相等的消息m_0，m_1。

(2) 运行$\mathrm{Gen}(1^n)$生成一个密钥k，选择一个随机比特b，$b \leftarrow \{0,1\}$。计算一个密文$c \leftarrow \mathrm{Enc}_k(m_b)$并且给$A$。这里$c$叫挑战密文。

(3) A输出一个比特b'。

(4) 该实验的输出定义为：如果$b = b'$，输出为1；否则为0。如果$\mathrm{PrivK}^{\mathrm{eav}}_{\mathrm{A},\Pi}(n) = 1$，则$A$成功。

1.3计算复杂性理论在密码学中的应用(续)

1.3.2基于计算安全的加密方案(续)

定义10 如果对于所有的PPT敌手A，存在一个可忽略函数$negl$使得

$$\Pr(\mathrm{PrivK}^{\mathrm{eav}}_{\mathrm{A},\Pi}(n)=1)\le 1/2+negl(n),$$

则一个对称密钥加密方案$\Pi=(\mathrm{Gen},\ \mathrm{Enc},\ \mathrm{Dec})$具有选择明文不可区分性。其中Pr是概率符号，而概率的来源是A的随机性及实验的随机性（选择密钥、随机比特数b，以及在加密过程中引入的任何随机性）。

1.3计算复杂性理论在密码学中的应用(续)

1.3.2基于计算安全的加密方案(续)

定义11（伪随机发生器）令$l(\cdot)$为多项式，令G为确定多项式时间算法。G满足对任意输入$s\in\{0,1\}^n$，输出一个长度为$l(n)$的字符比特串。如果该算法还满足下面两个条件，则称G是一个伪随机发生器。

（1）对于每个n来说，满足$l(n)>n$。

（2）对所有的PPT区分器D来说，存在一个可忽略函数$negl$，满足

$$|\Pr(D(r)=1)-\Pr(D(G(s))=1)|\leq negl(n),$$

式中，r是从$\{0,1\}^{l(n)}$中均匀随机选择的，种子s是从$\{0,1\}^n$中均匀随机选择的，并且概率源为D的随机性和对r、s的选择。

1.3计算复杂性理论在密码学中的应用(续)

1.3.2基于计算安全的加密方案(续)

可归约安全加密方案

一个可归约安全的对称加密方案可构造如下：

（1）Gen：输入1^n，均匀随机选择$k \leftarrow \{0,1\}^n$，并将其作为密钥输出。

（2）Enc：输入一个密钥$k \in \{0,1\}^n$，以及消息$m \in \{0,1\}^{l(n)}$，输出密文$c := G(k) \oplus m$。

（3）输入一个密钥$k \in \{0,1\}^n$，以及一个密文$c \in \{0,1\}^{l(n)}$，输出消息$m = G(k) \oplus c$。

谢谢！