概率加密*

SHAFI GOLDWASSER 和 SILVIO MICALI

麻省理工学院计算机科学实验室，剑桥，马萨诸塞州 02139

收到时间：1983年2月3日；修订时间：1983年11月8日

本文介绍了一种新的数据加密概率模型。对于这个模型，在适当的复杂性假设下，证明对于一个具有多项式有界计算资源的对手而言，从密文中提取关于明文的任何信息平均来说是困难的。该证明适用于具有任何概率分布的任意消息空间。本文提出了该模型的第一个实现。在判定二次剩余性对于因子分解未知的合数的难解性假设下，证明了该实现的安全性。

1. 引言

本文提出了一种具有以下性质的加密方案：

给定密文后关于明文能高效计算的任何信息，在没有密文的情况下也能高效计算。

我们的加密方案的安全性基于复杂性理论。因此，当我们说对手从密文计算关于明文的任何信息是"不可能的"时，我们指的是在计算上是不可行的。

相对年轻的复杂性理论领域尚未能为哪怕一个自然的NP完全问题证明一个非线性下界。与此同时，尽管付出了巨大的数学努力，数论中的一些问题几个世纪以来一直拒绝任何"驯服"。因此，为了具体实现我们的方案，我们假设了数论中一些问题的难解性，例如因子分解或判定关于合数模的二次剩余性。在这个背景下，证明一个问题困难意味着证明其等价于上述提到的某个问题。换句话说，对我们加密方案具体实现的任何安全威胁，都将导致判定关于合数整数的二次剩余性的高效算法。

- 这项研究是在两位作者都是加州大学伯克利分校学生时完成的，并部分由NSF基金MCS 82-04506支持。本手稿的准备工作完成时，第一作者在麻省理工学院计算机科学实验室，由Bantrell奖学金和IBM教师发展奖支持；第二作者在多伦多大学计算机科学系。

1.1. 确定性加密：陷门函数模型

我们的加密方案受益于 Diffie 和 Hellman [9]、Rivest、Shamir 和 Adleman [21] 以及 Rabin [20] 的思想。

Diffie 和 Hellman [9] 引入了公钥密码系统的思想，它基于某些基础计算问题的难解性。直观地说，这个想法是找到一个加密函数 E，它易于计算但难以求逆，除非知道一些秘密信息，即陷门。这样的函数被称为陷门函数。为了加密消息 m，任何人只需计算 $E(m)$，但只有那些知道陷门信息的人才能从 $E(m)$ 计算出 m。

对于本文来说，最相关且最具启发性的两个陷门函数实现是 Rivest、Shamir 和 Adleman 提出的 RSA 函数 [21]，以及 Rabin [20] 建议的其特化版本。

1.2. 对陷门函数模型的基本反对意见

我们指出这种方法存在的两个基本弱点：

(1) f 是一个陷门函数，这一事实并不能排除当 x 具有特殊形式时，从 $f(x)$ 计算 x 的可能性。通常，消息并非由随机选择的数字组成，而是具有更多结构。这种结构信息可能有助于解码。例如，一个在通用输入上难以求逆的函数 f，在英语句子的 ASCII 表示上可能容易求逆。

(2) f 是一个陷门函数，这一事实并不能排除从 $f(x)$ 轻松计算关于 x 的某些部分信息（甚至是 x 的每一位）的可能性。以确保所有部分信息的保密性的方式加密消息，是密码学的一个重要目标。假设我们想使用加密通过电话玩纸牌游戏。如果一张牌的花色或颜色可能被泄露，那么整个游戏应被视为无效。事实上，Lipton [17] 曾指出，在 SRA 的 Mental Poker [22] 实现中，可以轻松计算出一个本应隐藏的关于纸牌的比特信息。

尽管没有人知道如何破解 RSA 或 Rabin 方案，但在这些方案中，都没有证明在不对消息空间做任何假设的情况下解码是困难的。Rabin 证明，在他的方案中，解码对于对手是困难的，前提是可能的消息集合具有某种密度属性。我们将在第 2 节进一步讨论这一点。

1.3. 概率加密：新模型

在本文中，我们从确定性框架转向概率框架。这使我们能够处理陷门函数模型中出现的问题，而无需对我们想要发送的消息施加任何概率结构。

我们用不可近似陷门谓词的概念取代了陷门函数的概念。简而言之，谓词 B 是陷门且不可近似的，如果任何人都可以选择一个 x 使得 $B(x) = 0$，或选择一个 y 使得 $B(y) = 1$，但只有知道陷门信息的人才能在给定 z 的情况下计算 $B(z)$ 的值。当陷门信息未知时，具有多项式有界计算资源的对手无法以比随机猜测更好的方式决定 $B(z)$ 的值（正式定义见第 3 节）。

我们用单比特的概率加密替代了确定性的分组加密，其中有许多不同的"1"编码和许多不同的"0"编码。为了加密每个消息，我们使用一枚公平的硬币。因此，每个消息的编码将取决于消息加上一系列硬币投掷的结果。更具体地说，二进制消息将按比特加密如下："0"通过随机选择一个满足 $B(x) = 0$ 的 x 来编码，"1"通过随机选择一个满足 $B(x) = 1$ 的 x 来编码。因此，每个消息都有许多可能的编码。然而，消息总是可以被唯一解码的。

新模型的两个特性是：

(1) 对于知道陷门信息的合法消息接收者来说，解码是容易的，但对对手来说则是可证明困难的。因此，陷门函数的精神得以保留。此外，在我们的方案中，我们对消息空间没有任何限制。该方案的安全性对于属于任何消息空间、具有任何概率分布的消息都得到了证明。

(2) 对手无法从加密消息中获得任何信息。

设 $g \colon M \to V$ 是一个非常数函数。假设消息空间 M 具有某种概率分布。相应地，对于每个 $v \in V$，令 $p_v = \mathrm{prob}(g(m) = v \mid m \in M)$，并令 $\bar{v} \in V$ 满足 $p_{\bar{v}} = \max_{v \in V} p_v$。那么，无需任何特殊能力，一个给定密文的对手总是可以猜测 g 在明文上的值，并且以概率 $p_{\bar{v}}$ 正确。我们证明，对于一个概率加密方案，给定密文的对手无法以优于 $p_{\bar{v}}$ 的概率猜测 g 在明文上的值。注意，g 不必是多项式时间可计算的，甚至不必是递归的。因此，我们的加密模型通过了 Shannon 完美保密定义的多项式有界版本；见第 7.3 小节。

这个特性使 Goldwasser 和 Micali [11] 能够设计一个 Mental Poker 方案，在该方案下，根据二次剩余性假设，无法轻松计算出应隐藏的纸牌的任何部分信息。

1.4. 新模型的具体实现

我们引入对于因子分解未知的合数模的二次剩余性（精确定义见第 6 节），作为不可近似陷门谓词的第一个例子。因此，我们引入了一个新的概率公钥密码系统，当且仅当判定关于合数模的二次剩余性是困难的时，该系统在非常强的概率意义上是安全的（见第 4 节）。该公钥密码系统提供的安全性扩展到加密消息的所有部分信息、所有可能的消息空间以及消息空间的所有可能概率分布（安全性的正式定义见第 5 节）。

此类谓词的另一个例子出现在 Goldwasser、Micali 和 Tong [12] 以及 Goldwasser [13] 的论文中。他们提出的谓词是难以近似的，当且仅当因子分解合数是困难的。利用第 4 节的构造，我们可以基于他们提出的谓词构建一个公钥密码系统。同样，对这个密码系统的任何安全威胁，都将导致一个高效的因子分解算法。

在 [26] 中，Yao 证明如果一一对应的陷门函数存在，则不可近似的陷门谓词存在。

1.5. 相关工作

Blum 和 Micali 在 [5] 中展示了第一个不可近似谓词的例子，该谓词不是陷门的。他们的谓词是难以近似的，当且仅当离散对数问题是困难的。

二次剩余性谓词不仅是不可近似陷门谓词的一个例子，还具有其他使其在协议设计中特别有吸引力的特性。自从我们在 [10] 中首次提出它以来，它已被广泛使用。第一个使用这个谓词的协议是由 Goldwasser 和 Micali 在 [11] 中提出的。他们设计了一个协议，让两名玩家通过电话玩 Mental Poker，使得任何玩家都无法获得不在其手中的纸牌的任何部分信息。其他证明此谓词有用的工作有：Blum、Blum 和 Shub 的密码学强伪随机比特生成器实现 [4]（基于 [5]），Brassard 的认证标签实现 [7]，Luby、Micali 和 Rackoff 的同时交换秘密比特的方法 [19]，以及 Vazirani 和 Vazirani 的单比特披露实现 [25]。

2. 基于陷门函数的公钥密码系统概述

本节使用的所有数论符号将在第 3 节定义。

2.1. 什么是公钥密码系统？

公钥密码系统的概念是由 Diffie 和 Hellman 在他们开创性的论文 [9] 中提出的。设 M 为有限消息空间，$\{A, B, \ldots\}$ 为用户，$m \in M$ 表示消息。设 $E_A \colon M \to M$ 为 A 的加密函数，理想情况下是双射的，D_A 为 A 的解密函数，满足对所有 $m \in M$ 有 $D_A(E_A(m)) = m$。在公钥密码系统中，E_A 被放置在公开文件中，用户 A 将 D_A 保密。在只知道 E_A 的情况下，D_A 应该难以计算。为了将消息 m 发送给 A，B 从公共文件中获取 E_A，计算 $E_A(m)$ 并将此消息发送给 A。A 轻松计算 $D_A(E_A(m))$ 以获得 m。

2.2. RSA 方案和 Rabin 方案

这类加密函数 E_{A} 的两个实现是 Rivest 等人的 RSA 函数 [21] 和 Rabin 函数 [20]。

RSA 方案和 Rabin 方案的关键思想在于选择一个合适的数论陷门函数。在 RSA 方案中，用户 A 选择 n，即两个大的不同素数 p_1 和 p_2 的乘积，以及一个数字 s，使得 s 和 $\varphi(n)$ 互质，其中 φ 是欧拉函数。A 将 n 和 s 放入公共文件，并将 n 的因式分解保密。令 $Z_n^* = \{x \in N : 1 \leqslant x \leqslant n - 1 \text{ 且 } x \text{ 与 } n \text{ 互质}\}$。对于每个消息 $m \in Z_n^*$，$E_A(m) = m^s \bmod n$。显然，计算模 n 的 s 次根的能力意味着解码能力。知道 n 因式分解的 A 可以轻松计算模 n 的 s 次根。当 n 的因式分解未知时，尚无已知的有效方法来计算模 n 的 s 次根。

Rabin 建议通过选择 $s = 2$ 来修改 RSA 方案。因此，对于所有用户 A，$E_A(x) = x^2 \mod n$。注意，E_A 是一个 4 对 1 的函数，因为我们的 n 是两个素数的乘积。事实上，模 n 的每个二次剩余，即每个满足 $q \equiv x^2 \mod n$ 对于某个 $x \in Z_n^*$ 的 q，有四个模 n 的平方根：$\pm x \mod n$ 和 $\pm y \mod n$。由于 A 知道 n 的因式分解，在接收到加密消息 $m^2 \mod n$ 时，她可以轻松计算其四个平方根并获得消息 m。（A 可以通过先计算模 p_1 和 p_2 的平方根，然后通过中国剩余定理组合它们来计算模 n 的平方根。）为了消除解码中的歧义，可以提出以下启发式方法：发送消息 m 时，发送 $m^2 \mod n$ 以及 m 的最后 20 位。这种额外信息实际上无助于解码：人们总是可以猜测 m 的最后 20 位数字。（为了避免公开 m 的最后 20 位数字，只需选择一个 20 位的随机整数 r，然后发送 $(m2^{20} + r)^2 \mod n$ 以及 r。）

下面的定理显示了反转 Rabin 函数 $x^{2} \mod n$ 有多困难。

定理 (Rabin)。如果对于模 n 的二次剩余 q 中的 $1 / \log n$ 部分，人们可以找到 q 的一个平方根，那么人们可以在随机多项式时间内分解 n。

该定理源于引理 1，我们在此陈述而不加证明。

引理 1。给定 $x, y \in Z_n^*$ 满足 $x^2 \equiv y^2 \mod n$ 且 $x \neq \pm y \mod n$，则存在一个多项式时间算法来分解 n。（实际上，n 与 $x \pm y$ 的最大公约数是 n 的一个因子。）

Rabin 定理的非正式证明。假设我们有一个神奇盒子 MB，当给定模 n 的二次剩余 q 时，对于 q 中的 $1 / \log n$ 部分，它能输出 $q \mod n$ 的一个平方根。那么我们可以通过迭代以下步骤来分解 n：

在 Z_{n}^{*} 中随机选取 i，并计算 $q = i^{2} \mod n$。将神奇盒子 MB 与 q 作为输入。如果 M 输出 q 的一个不同于 i 或 $-i \mod n$ 的平方根，那么（根据引理 1）分解 n。

预期的迭代次数很低，因为在每一步，我们有 $1/2 \log n$ 的机会分解 n。

2.3. 对基于陷门函数的密码系统的反对意见

以下问题可能出现在 RSA 和 Rabin 方案中，更一般地说，出现在任何其他基于陷门函数的公钥密码系统中：

(1) f 是一个陷门函数，这一事实并不能排除当 x 具有特殊形式时，从 $f(x)$ 计算 x 的可能性。

(2) f 是一个陷门函数，这一事实并不能排除从 $f(x)$ 轻松计算关于 x 的某些部分信息的可能性。

2.3.1. 对反对意见 1 的讨论

有人可能认为 Rabin 的公钥密码系统与因子分解一样难以破解，理由如下：任何能够从其加密 $m^2 \mod n$ 在 $1 / \log n$ 的时间内获取消息 m 的人，实际上实现了 Rabin 定理中的神奇盒子，因此可以高效地分解 n。

我们希望指出以下事实。

声称。如果 M，即消息空间，在 Z_{n}^{*} 中是"稀疏的"，那么能够解码所有消息中的 $1 / \log n$ 部分并不能产生一个随机多项式时间的因子分解算法。

所谓"稀疏"，我们指的是对于一个随机选择的 $x \in Z_n^*$，x 是一条消息的概率实际上为 0。

令 $f(x) = x^{2} \mod n$。假设我们只能在 $f(M)$ 上反转函数 f。那么，我们将拥有一个神奇盒子 MB，在输入 $m^{2} \mod n$（其中 $m \in M$）时，它输出 m；而在输入 $q \notin \{m^{2} \mod n \mid m \in M\}$ 时，对于 q 中可忽略的部分，它输出正确答案。使用这样的神奇盒子，我们可以解码，但不能高效地分解 n。让我们看看上面 Rabin 定理的非正式证明，使用这个 MB。如果我们选择 $m \in M$ 并将 $m^{2} \mod n$ 输入 MB，那么我们得到 m 返回，无法分解。如果我们选择 $i \notin M$ 并将 $i^{2} \mod n$ 输入 MB，那么 $i^{2} \mod n$ 的任何不同于 i 的平方根属于 M 的概率实际上为 0，我们得不到答案。

我们得出结论，对于 Rabin 函数，当且仅当合法消息在 Z_{n}^{*} 中是稠密的（例如，$M = Z_{n}^{*}$ 且所有消息等概率），人们才能解码当且仅当能够分解。

2.3.2. 对反对意见 2 的讨论

加密算法的一个理想特性是，对手不应该能够从密文中获得关于明文的任何部分信息。

例如，设 f 是定义在消息空间 M 上的哈希函数或非常量谓词。令 $m \in M$。如果，给定 m 的加密，对手能够高效地计算 $f(m)$，那么我们说关于 m 的信息可以从 m 的加密中获得。

注意，如果加密算法 E 是一个陷门函数，那么关于明文的部分信息就无法隐藏。事实上，定义在明文上的以下谓词 B 易于从密文中求值：$B(x) =$ 真，当且仅当 $E(x)$ 是偶数。我们可以使用概率加密来避免此类问题。

现在让我们讨论一个由 Brassard [6] 提出的与部分信息安全密切相关的关键问题：如何在公钥密码系统中安全地发送单个比特。

2.3.3. 在基于陷门函数的公钥密码系统中安全发送单个比特的尝试

假设用户 B 想要向用户 A 以高度保密的方式发送一个单比特消息。该比特是 0 或 1 的概率相等。B 不希望任何对手能够以 51% 的概率正确猜测他的消息。B 知道用户 A 的公共加密函数 E_A 难以求逆，并尝试以下列方式利用这一事实。

想法 1。系统中的所有用户约定一个整数 i。用户 B 在 M 中随机选择 r，除了 r 的第 i 位将是他的消息。B 将 $E_A(r)$ 发送给 A。

A 可以解码从而获得所需的比特。但对手能做什么呢？

危险。令 $y = E_A(x)$，其中 E_A 是一个单向函数。那么，给定 y，计算 x 可能很困难，但计算 x 的某个特定比特可能不难。

例子。令 p 为一个大素数，使得 $p - 1$ 至少有一个大素因子。令 g 为 Z_p^* 的生成元。那么 $y \equiv g^x \bmod p$ 被认为是一个单向函数。但是，尽管从 $g^x \bmod p$ 计算 x 很困难（索引查找问题），获取 x 的最后一位却很容易。事实上，x 以 0 结尾当且仅当 y 是模 p 的二次剩余，并且存在概率多项式时间算法来测试数字是否是模素数 p 的二次剩余（见第 3.1 小节）。

以下想法由 Donald Johnson 提出。

想法 2。B 构造一个 100 位的整数 x 如下：他随机选择 $8 \leqslant i \leqslant 100$，并将 x 的第 i 位设置为他想要通信的比特。x 的剩余 92 位随机选择，除了 x 的前 7 位指定了位置 i。B 将 $E_A(x)$ 发送给 A。

危险。E_{A} 可能是一个陷门函数，然而人们可能能够从 $E_{A}(x)$ 轻松计算 x 的前 7 位和 x 的最后 93 位中的某一位。如果是这种情况，人们可以以概率 $\frac{1}{92} + \frac{1}{2} \cdot \frac{91}{92}$ 正确计算 B 的消息 x。

总而言之，有很多方法可以将单个比特"嵌入"到一个二进制数 x 中。取 x 所有数字的"异或"只是另一个例子。然而，给定 $y = E_A(x)$，能够发现嵌入在 x 中的单个比特并不违背计算 x 很困难这一事实。那么，什么是发送单个比特的安全方法呢？不可近似陷门谓词将为这个问题提供一个解决方案。

3. 不可近似陷门谓词

在第 4 节中，我们介绍了概率公钥密码系统的模型。我们证明该模型具有高度安全性。我们的模型从分组加密切换到逐比特加密。为此目的，我们必须放弃陷门函数的概念，转而采用新的不可近似陷门谓词概念。

定义 (ε-近似)。如果一个电路 $C[\cdot]$ 满足 $C[x] = B[x]$ 对于至少 $\frac{1}{2} + \varepsilon$ 部分的 $x \in \Omega$ 成立，则称电路 $C[\cdot]$ ε-近似了谓词 $B: \Omega \to \{0, 1\}$。

我们接着形式化地定义不可近似陷门谓词。

令 N 表示自然数集合，N' 是 N 的无限子集。对于每个 $k \in N'$，令 S_k 表示 k 比特整数的子集；对于每个 $i \in S_k$，令 Ω_i 表示至多 k 比特的整数的子集。设

是一个由大小为 k 的整数索引的谓词集合，并且

我们称 $\mathbf{B}$ 是一个**不可近似陷门谓词 (UTP)**，如果：

(1) **(B 是不可近似的)**：固定多项式 P_{1} 和 P_{2}。令 $k \in N'$。令 c_{k} 表示最小规模电路 $C[\, , \,]$ 的规模，使得对于至少 $1 / P_{2}(k)$ 部分的 $i \in S_{k}$，$C[\, , i]$ $(1 / P_{1}(k))$-近似 B_{i}。我们称 $\mathbf{B}$ 是不可近似的，如果 c_{k} 的增长速度比 k 的任何多项式都快。

(2) **(B 是陷门的)**：对于 $v \in \{0,1\}$，令 $\Omega_i^v = \{x \in \{\Omega_i\} \mid B_i(x) = v\}$。我们称 B 是陷门的，如果：

(a) 存在一个在 k 上为概率多项式时间的图灵机 T_{1}，在输入 (i, v)（其中 $i \in S_{k}$ 且 $v \in \{0, 1\}$）时，以均匀概率选择 $x \in \Omega_{i}^{v}$。

(b) 存在一个函数 $\sigma: \bigcup_{k \in N'} S_k \to N$，使得对于某个多项式 Q，对于所有 x，$|\sigma(x)| < Q(|x|)$，以及一个多项式时间图灵机 T_2，满足 $T_2[i, \sigma(i), x] = B_i(x)$ 对于所有 $i \in S_k$ 和所有 $x \in \Omega_i$。我们称 $\sigma(i)$ 为 i 的秘密。

(c) **(可构造条件)**：对于所有 $k \in N'$，可以在概率多项式（在 k 内）时间内以概率 $1 / |S_k|$ 选择任何一对 $(i \in S_k, \sigma(i))$。

条件 (2c)，即可构造条件，保证了如果有人选择一对 $(i,\sigma(i))$，其中 $i \in S_k$，并公开 i，那么计算 $B_i(x)$ 将是困难的。否则，假设可以高效选择的配对 $(i,\sigma(i))$，$i \in S_k$，仅占所有可能配对的极小部分。那么，对手可以从公开的 i 出发，通过反复选择配对 $(j,\sigma(j))$ 直到 $j = i$ 来找出 $\sigma(i)$。

备注 3.1。注意，如果 B 是一个不可近似谓词，P_{1}, P_{2} 是多项式，那么对于所有足够大的 k，对于 $1 - (1 / P_{1}(k))$ 部分的 $i \in S_{k}$，$|\Omega_{i}^{0}| / |\Omega_{i}|$ 和 $|\Omega_{i}^{1}| / |\Omega_{i}|$ 都大于 $\frac{1}{2} - (1 / P_{2}(k))$。否则，要么总是输出 0 的平凡电路 C_{k}，要么总是输出 1 的平凡电路，将在至少 $1 / P_{1}(k)$ 部分的 $i \in S_{k}$ 上 $(1 / P_{2}(k))$-近似 B_{i}。

3.1. 二次剩余性作为 UTP

我们在二次剩余性问题（QRP）的难解性假设下，展示了一个不可近似陷门谓词集合的例子。如果需要，可以在第 7 节找到数论定义。

令 $k \in N$。令 p_1 和 p_2 表示素数。设，

并令 Z_{n}^{1} 表示 Z_{n}^{*} 中包含雅可比符号为 $+1$ 的元素的子集。对于所有 $x \in Z_{n}^{1}$，定义 Q_{n} 为：

令 $k \in N$。令 x 和 y 为二进制字符串。我们用 $x \neq y$ 表示 x 和 y 的连接。定义 $S_{4k} = \{n \neq y \mid n \in H_k$ 且 $y \in Z_n^1$ 是模 n 的二次非剩余。定义 $\Omega_{n \neq y} = Z_n^1$，并对于每个 $x \in Z_n^1$ 设 $Q_{n \neq y}(x) = Q_n(x)$。那么 $Q^\# = \{Q_{n \neq y} \mid n \neq y \in S_{4k}\}$ 是一个谓词集合。二次非剩余 y 的存在将用于展示 $Q^\#$ 的陷门性质。

(1) $Q^{\#}$ **是不可近似的**：这在二次剩余性假设下，由定理 2（第 7 节）证明。

(2) $Q^{\#}$ **是陷门的**：令 $\sigma_{4k}(n \neq y)$ 为 n 的因式分解，则 $Q^{\#}$ 是一个陷门谓词集合。实际上，如果知道 n 的因式分解，可以在 $O(k^3)$ 时间内计算 $Q_n(\cdot)$。此外，给定 y，一个模 n 的二次非剩余，我们可以通过在 Z_n^* 中随机选择 x 并计算 $r = yx^2 \mod n$，以均匀概率在概率多项式（在 k 内）时间内生成模 n 的二次非剩余。

(3) $Q^{\#}$ **是可构造的**：考虑以下算法，它选择一个元素 $n \neq y \in S_{4k}$，其中 $n \in H_k$ 且 $y \in Z_n^{+1}$ 是模 n 的二次非剩余。

步骤 1。抛 $4k$ 次公平硬币。

步骤 2。检查前 k 个结果和第二组 k 个结果是否分别构成一个大小为 k 的素数 p_1 和 p_2 的二进制表示。如果是，令 $n = p_1p_2$，并检查最后 $2k$ 位是否构成一个模 n 的二次非剩余 y。如果是，则停止：已选择 $p_1 \cdot p_2 \neq y$。否则转到步骤 1。

由于 S_{4k} 中的每个元素恰好可以由一个长度为 $4k$ 的硬币投掷序列生成，上述算法以均匀概率选择 S_{4k} 中的元素。根据素数定理以及存在用于素性检验的随机多项式时间算法，上述算法在随机 $\text{poly}(k)$ 时间内运行。

我们得出结论，在 QRA 下，$Q^{\#}$ 是一个不可近似陷门谓词。

4. 公钥密码系统和概率公钥密码系统

在上一节中，我们定义了 UTP。现在我们准备介绍我们的概率加密模型。在第 4.2 小节中，我们形式化地定义了由安全参数参数化的公钥密码系统（PKC）的概念。在第 4.3 小节中，我们定义了我们概率公钥密码系统（PPKC）的模型。在第 4.4 小节中，我们基于 QRA（二次剩余性问题的难解性假设）提出了该模型的具体实现。

4.1. 初步记号

以下记号将在本文的其余部分使用：令 Γ 为一个概率图灵机。我们用 $\Gamma[\beta]$ 表示 Γ 在输入 β 上的可能输出集合。我们赋予 $\Gamma[\beta]$ 以下概率分布：如果 $\alpha \in \Gamma[\beta]$，那么 α 的概率是 Γ 在输入 β 上输出 α 的概率。

令 T_{1} 和 T_{2} 为图灵机。通过说 T_{1} 被输入到（被输出自）T_{2}，我们指的是 T_{1} 的标准编码被输入到（被输出自）T_{2}。

4.2. 公钥密码系统

非正式地说，我们将 PKC 视为一个服务器。系统中的每个用户带着其消息空间的描述和一个公共安全参数 k 来到 PKC。在这样的输入下，PKC 产生一对算法：一个加密算法（可能是概率的）和一个解密算法。加密算法和解密算法的描述都应该是短的（k 的多项式）。此外，两种算法都应在多项式时间内停止。用户将（描述）加密算法存储在公共文件中，并将（描述）解密算法保密。

我们继续形式化定义什么是 PKC。

我们令 k 表示一个参数，它将以一元形式呈现给本文中的所有算法。令 $U = \{A, B, \ldots\}$ 为用户有限集合。

一个消息生成器是一个概率多项式时间图灵机 MG，它在输入 k 时输出一个被称为消息的字符串。

定义。一个**公钥密码系统**是一个概率多项式时间图灵机 Π，它在输入 k 和 MG 时输出两个算法 E 和 D 的描述，使得

(1) 对于某些常数 c，在大小为 n 的输入上，E 和 D 都在 n^c 步内停止，并且

(2) 对于所有 $m \in \mathbf{MG}[k]$，$D(E(m)) = m$。

我们称 E 为由 Π 生成的**加密算法**，D 为由 Π 生成的**解密算法**。由 Π 生成的加密算法可能是概率的。

备注。让我们再次强调 Π 是一个概率图灵机，因此在相同的输入对 $(k, \mathbf{MG})$ 上，它可能输出许多不同的（加密算法，解密算法）对。当我们只对 Π 在输入 k 和 $\mathbf{MG}$ 时生成的加密算法 E 感兴趣时，我们将写作 $E \in \Pi(k, \mathbf{MG})$。

4.3. 概率公钥密码系统

令 $\mathbf{B} = \bigcup_{k\in N'}\mathbf{B}_k$，其中 $\mathbf{B}_k = \{B_i;\Omega_i\to \{0,1\} |i\in S_k\}$，是一个不可近似陷门谓词。一个具有 UTP B 的**概率公钥密码系统 (PPKC)** 是一个 PKC $\varPi$，它接受安全参数 k 和消息生成器 MG 作为输入，并输出一对 $(i,\sigma (i))$，其中 $i\in S_k$ 且 $\sigma (i)$ 是 i 的秘密。这可以通过 B 的可构造性实现。

Π 的输出 $i \in S_k$ 指定一个加密算法 E 如下：E 接受一个 l 位二进制消息 $m = m_1m_2\dots m_l$ 作为输入。对于 m 的二进制表示中的每个 m_j，E 随机选择一个元素 $x_{j} \in \Omega_{l}$，使得 $B_{i}(x_{j}) = m_{j}$，并输出 l 元组 $(x_{1},\dots,x_{l})$。根据 B 的陷门性质，这可以在概率多项式（在 k 和 l 内）时间内完成。E 的输出以 $O(kl)$ 为界。

一般来说，考虑二进制字符串 $b = b_{1} \cdots b_{l}$，其中 $b_{j} \in \{0, 1\}$。我们称任何 l 元组 $(x_{1},\ldots ,x_{l})$ 为使用谓词 B_{i} 对 b 的**概率加密**，如果 $x_{j}\in \Omega_{i}$ 且 $B_{i}(x_{j}) = b_{j}$ 对所有 $1\leqslant j\leqslant l$ 成立。因此，注意与基于陷门函数（如 RSA）的 PKC 相比，在概率公钥密码系统中，每个消息 m 都有许多可能的概率加密。

Π 的输出 $\sigma(i)$ 指定一个解密算法 D 如下：令 T 为一个概率多项式时间图灵机，在输入 $i \in S_k$，$x \in \Omega_i$ 和 $\sigma(i)$ 时计算 $B_i(x)$。根据 B 的陷门性质，这样的 T 存在。那么 D 使用 T 作为子程序，如下所示：令 D 的输入由 l 元组 $(x_1, \ldots, x_l)$ 组成，其中 $x_j \in \Omega_i$ 对于每个 $1 \leqslant j \leqslant l$。那么对于每个 $1 \leqslant j \leqslant l$，$D$ 使用输入 i，$\sigma(i)$，x_j 调用 T 来计算 $B_i(x_j)$，并将 T 的 l 个答案中的每一个写入其输出磁带。由于 T 在多项式时间内运行，所以 D 也是。

4.4. 基于二次剩余性的 PPKCS 实现

让我们明确描述基于二次剩余性问题的 PPKC 实现。

示例 1。令 $Q^{\#}$ 为上一节中定义的不可近似陷门谓词。回想一下，$Q^{\#} = \{Q_{n\# y} \mid n \neq y \in S_{4k}\}$，其中 $n \in H_k$ 且 $y \in Z_n^1$ 是模 n 的二次非剩余。

令 Π 为一个基于不可近似陷门谓词 $Q^{\#}$ 的概率公钥密码系统。用户将安全参数 k 输入 Π。在输入 k 和消息生成器 MG 时，Π 工作如下：

(1) 它随机选择两个 k 位素数 p_1 和 p_2。

(2) 令 $n = p_1, p_2$，

(3) 选择 $\nu \in Z_{n}^{1}$ 使得 ν 是模 n 的二次非剩余，

(4) 输出作为加密算法的是对 (n, y) 的某种标准编码，作为解密算法的是对 (p_1, p_2) 的某种标准编码。

用户 C 公开 (n,y) 对，将 (p_1,p_2) 对保密。

如何加密

假设用户 B 想向用户 C 发送一个二进制字符串 $b = b_{1}\dots b_{l}$。那么，

对于每个 $b_{i}\in b$，
B 在 Z_n^* 中随机选择 x，
如果 $b_{i} = 1$，B 令 $e_i = yx^2\bmod n$，
否则 $\mathbf{B}$ 令 $e_i = x^2\bmod n$。
B 向 C 发送 l 元组 $(e_1,\dots,e_l) = E_n(b)$。

编码一个 l 比特消息 b 需要 $O(lk^2)$ 时间。通常，明文的 1 比特被扩展为 k 比特密文。

如何解密

假设用户 C 接收到 $(e_1, \dots, e_l)$，即消息 b 的加密。那么，

对于每个 $e_i \in e$，

（注意：由于 C 知道 n 的因式分解，他可以计算 $Q_{n}(x)$）

从其加密计算 b（$|b| = l$）需要 $O(lk^3)$ 时间。

5. 公钥密码系统的安全性

我们继续讨论公钥密码系统的安全性概念。显然，公钥密码系统中的安全性概念取决于对手可能的行为模型。在本文中，对手是被动的线路窃听者。这个对手知道消息空间及其概率分布，知道加密算法，被给予密文，并试图通过计算来检索明文。

5.1. 多项式安全性

非正式设定

令消息寻找者 F 和线路窃听者 T 是你最喜欢的具有多项式有界计算资源的计算模型。这样的 F 和 T 可以是多项式时间图灵机、概率多项式时间图灵机、"小"电路等。直观地说，我们说一个公钥密码系统是**多项式安全的**，如果对于所有具有任何概率分布的消息空间 M，由服务器产生的加密算法将满足：多项式有界的消息寻找者 F 无法找到 M 中的两个消息 m_{1} 和 m_{2}，使得它们的加密能被多项式有界的线路窃听者 T 区分开来。也就是说，给定 α（m_{1} 或 m_{2} 的加密），T 在理解 α 正在编码两个消息中的哪一个方面不应有任何优势。注意，很可能存在一对消息，它们的加密能被 T 区分，但多项式有界的 F 不可能找到这样的一对。注意，产生确定性加密算法（例如 RSA）的 PKC 不可能是多项式安全的。

在本文中，消息寻找者和线路窃听者被选择为电路。

正式设定

令 π 为一个 PKC。令 MG 为一个消息生成器。我们用 M_{k} 表示 $\mathbf{MG}[k]$。不失一般性，我们假设所有 $m\in M_k$ 具有相同的长度 $l_{k} = Q(k)$，其中 Q 为某个多项式。

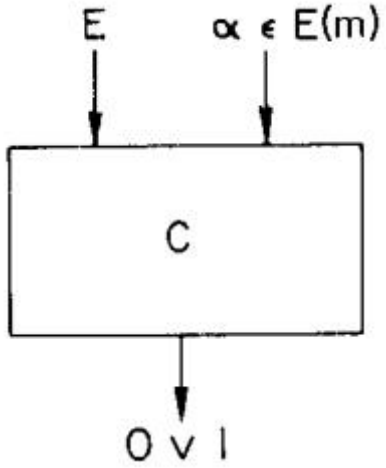

图 1

一个 **k 线路窃听者**是一个电路 C，具有一个布尔输出和足够多的布尔输入，以接收（加密算法的描述）$E \in \Pi(k, \mathrm{MG})$ 和 $\alpha \in E(m)$，其中 $m \in M_k$（见图 1）。令 $m_1, m_2 \in M_k$。令 p_1^E 为 C 在输入 $E \in \Pi(k, \mathrm{MG})$ 和 $\alpha \in E(m_1)$ 时输出 1 的概率，p_2^E 为 C 在输入 $E \in \Pi(k, \mathrm{MG})$ 和 $\alpha \in E(m_2)$ 时输出 1 的概率。我们说 C **P-区分** m_1 和 m_2 关于 E，如果 $|p_1^E - p_2^E| > 1 / P(k)$。

一个 **k 消息寻找者**是一个电路 C，具有 $2l_{k}$ 个布尔输出和足够多的布尔输入来描述一个 $E \in \Pi[k, \mathbf{MG}]$。在输入 E 时，C 输出两个消息 $m_{1}, m_{2} \in M_{k}$（见图 2）。

注意 F_{k} 可能内置了 MG 的描述。

定义（多项式安全的公钥密码系统）。令 Q，P_1，P_2 为多项式。令 Π 为一个公钥密码系统，MG 为一个消息生成器。令 $T = \{T_k\}$，其中 T_k 是一个 k 线路窃听者，其门数少于 $Q(k)$。令 s_k^T 为最小规模的消息寻找者 F 的规模，使得以大于 $1 / P_1(k)$ 的概率在输入 $E \in \Pi(k, \mathbf{MG})$ 和 MG 时，输出两个消息 m_1 和 m_2 在 M_k 中，使得 T_k 和 P_2-区分 m_1 和 m_2。我们说 Π 关于 MG **是多项式安全的**，如果对于任何线路窃听者序列 T，s_k^T 的增长速度比 k 的任何多项式都快。我们说 Π **是多项式安全的**，如果对于任何消息生成器 MG，Π 关于 MG 是多项式安全的。

备注。注意，在多项式安全的公钥密码系统的定义中，我们没有对 m_{1} 和 m_{2} 的概率施加任何约束。因此，

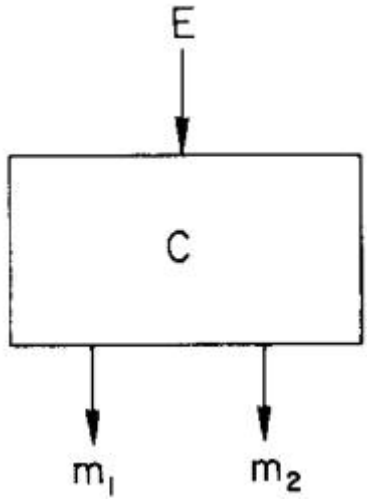

图 2

即使两个消息出现的可能性很小且能被 T_{k} 区分，也不能被轻易找到。

直观上，并且将被正式证明，多项式安全性意味着更传统的安全性概念。非正式地说，如果一个公钥密码系统是多项式安全的，那么没有多项式有界的线路窃听者 T 能够从密文中检索明文或其任何部分信息。

我们首先展示新引入的概率 PKC 确实是多项式安全的。

关于定理 5.1 的备注。定理 5.1 证明的基本思想是采样游走。假设 d 维超立方体 C 中的每个顶点 v 都被标记为 0 到 1 之间的实数 $\lambda(v)$，并且易于找到两个顶点 u 和 v 使得 $|\lambda(u) - \lambda(v)| > \varepsilon$。那么就易于找到两个相邻的顶点 s 和 t 使得 $|\lambda(s) - \lambda(t)| > \varepsilon / d$：只需找到 C 中的顶点 u 和 v 使得 $|\lambda(u) - \lambda(v)| > \varepsilon$；然后考虑 $(\omega_0, \dots, \omega_k)$，一条从 u 到 v 的最短顶点游走，并查看 (ω_l, ω_{l+1}) 对。

在我们的例子中，超立方体的每个顶点 v 都是一个 d 位字。标签 $\lambda(v)$ 是线路窃听者在 v 的概率加密上输出 1 的频率。我们通过采样快速近似这些频率。然后我们找到两个相邻的字 s 和 t，它们的关联频率有跳跃，并使用 s 和 t 来近似系统所基于的 UTP。

定理 5.1。每个概率公钥密码系统都是多项式安全的。

定理 5.1 的证明。令

是一个不可近似陷门谓词。令 Π 为一个 PPKC，在输入 k 和 MG 时以概率 $1 / |S_k|$ 输出 $i\in S_k$ 和 $\sigma (i)$。这指定了一个概率加密算法 E，如第 4.3 小节所述。回想一下，T_{k}，线路窃听者，是一个 $\text{poly}(k)$ 规模的电路，在接收到作为输入的 i 和 M_{k} 中 m 的使用 B_{i} 的概率编码后，输出 0 或 1。

令 $f_{i,m}$ 为当 T_{k} 被输入 m 使用 B_{i} 的所有概率编码时输出 1 的频率。

令 P_{1} 和 P_{2} 为多项式。对于 $k \in N$，设

并令 F_{k} 为一个消息寻找者。令 N'' 为 N' 的无限子集。假设对于 $1 / \eta_{k}$ 部分的 $i \in S_{k}$，F_{k} 输出两个消息 m_{1}^{i} 和 m_{2}^{i} 使得

那么我们将证明，对于所有 $k \in N''$，存在一个具有预言机 F_k 和 T_k 的概率 $\text{poly}(k, \delta^{-1})$ 时间图灵机 G，以概率 $1 - \delta$，对于 $1 / \eta_k$ 部分的 $i \in S_k$，$(\varepsilon_k / 5l_k)$-近似 B_i。

因此，由于 T_{k} 的规模以 k 的多项式为界，如果 F_{k} 的规模也以 k 的多项式为界，那么对于每个 $k \in N''$，G 可以很容易地转换成一个 $\text{poly}(k)$ 规模的电路 C_{k}，该电路对于至少 $1 / \eta_{k}$ 部分的 $i \in S_{k}$ $(\varepsilon_{k} / 5l_{k})$-近似 B_{i}。这将与 B 的不可近似性相矛盾。因此，F_{k} 的规模必须比 k 的任何多项式增长得都快，并且 Π 是多项式安全的。

a 和 $b \in \{0, 1\}^{l_k}$ 之间的**汉明距离**是 a 和 b 不同的比特数，如果它们之间的距离为 1，我们说 a 和 b 是**相邻的**。

我们继续构造图灵机 G。令 $\Omega_{k}^{l_{k}}$ 表示 Ω_{i} 元素的所有长度为 l_{k} 的序列的集合。在输入 $i \in S_{k}$ 和 $y \in \Omega_{i}$ 时，G 猜测 $B_{i}(y)$ 如下：

第 1 部分。它用输入 i 调用预言机 F_{k} 以找到 M_{k} 中的 m_{1}^{i} 和 m_{2}^{i}，使得

令 Δ 为 m_1^i 和 m_2^i 之间的距离。令 $a_0, a_1, \ldots, a_\Delta$ 为一个长度为 l_k 的二进制字符串序列，使得 $a_0 = m_1$，$a_\Delta = m_2$ 且 a_j 与 a_{j+1} 相邻，对于 $0 \leqslant j < \Delta$。由于 $|f_{i,m_1^i} - f_{i,m_2^i}| > \varepsilon_k$，必须存在 $x, 0 \leqslant x \leqslant \Delta - 1$，使得 $|f_{i,a_x} - f_{i,a_{x+1}}| > \varepsilon_k / l_k$。

赋予 Ω_{i} 和 $\Omega_{i}^{l_{k}}$ 均匀概率分布。根据 B 的陷门性质，这样的 a_{x} 和 a_{x+1} 可以通过蒙特卡洛实验以大于 $1 - \delta$ 的概率在概率 $\text{poly}(k, \delta^{-1})$ 时间内正确找到。为了记号的方便，令 $s = a_{x}$，$t = a_{x+1}$。计算 $f_{i,s}$ 和 $f_{i,t}$。

由于 $s = (s_1, \dots, s_{I_k})$ 和 $t = (t_1, \dots, t_{I_k})$ 是相邻的，它们恰好在一个位置不同。称这个位置为 d。

第 2 部分。不失一般性，假设 $f_{i,s} > f_{i,t}$。

情况 1。$s_d = 1, t_d = 0$。

那么，

在 $\Omega_i^{l_k}$ 中随机选取 $x = (x_{1},x_{2},\dots,x_{l_{k}})\in \Omega_{i}^{l_{k}}$，从所有满足 $B_{i}(e_{j}) = s_{j} = t_{j}$ 对于 $j\neq d$ 且 $e_d = y$ 的元素 $e = (e_1,\dots,e_{l_k})$ 中选取。
（回想一下，y 是 G 的输入。）

如果 $T_{k}(x) = 1$ 则 $G[y] = 1$
否则如果 $T_{k}(x) = 0$ 则 $G[y] = 0$。

情况 2。$s_d = 0$ 且 $t_d = 1$。

按照情况 1 进行，但令 $G[y] = 1 - T_k[x]$。这完成了对 G 的描述。

让我们证明，如果 s 和 t 被正确找到，对于 S_k 中 $1 / \eta_k$ 部分的 i，对于 $y \in \Omega_i$，

备注 5.1。由于 B 是不可近似的，根据备注 3.1，对于所有足够大的 k，对于 $1 - (\eta_k / 2)$ 部分的 $i \in S_k$，$|\Omega_i^0| / |\Omega_i| > \frac{1}{2} - (\varepsilon_k / 4l_k)$ 且 $|\Omega_i^1| / |\Omega_i| > \frac{1}{2} - (\varepsilon_k / 4l_k)$。因此，对于大于 $\eta_k(1 - (\eta_k / 2)) > (\eta_k / 2)$ 部分的 $i \in S_k$，F_k 输出 m_1^i 和 m_2^i 使得 $|f_{i,m_1^i} - f_{i,m_2^i}| > \varepsilon_k$；并且 $|\Omega_i^0| / |\Omega_i|$ 和 $|\Omega_i^1| / |\Omega_i|$ 都大于 $\frac{1}{2} - (\varepsilon_k / 4l_k)$。

i-签名 (x)，其中 $x = (x_{1},\ldots ,x_{l_{k}})\in \Omega_{i}^{l_{k}}$，将表示二进制字符串 $B_{i}(x_{1})\dots B_{i}(x_{l_{k}})$。那么，对于这样的 i，在情况 1 中，

在情况 2 中，遵循类似的证明，G 也将 $(\varepsilon_k / 5l_k)$-近似 B_i。

5.2. 语义安全性

在本节中，我们定义公钥密码系统的第二个安全标准，称为**语义安全性**。非正式地说，如果一个系统是语义安全的，那么无论窃听者在给定密文的情况下能计算关于明文的什么，他在没有密文的情况下也能计算。我们证明每个多项式安全的公钥密码系统都是语义安全的。因此，概率 PKC 是语义安全的。因此，我们的加密方案通过了 Shannon [23] 完美保密定义的多项式有界版本：将我们的注意力限制在具有多项式有界资源用于分析截获消息的对手上，截获的密码代表各种消息的后验概率与截获前相同消息的先验概率相同。

非正式设定

令 f 为定义在消息空间 M 上的任何函数。因此 f 不必是快速可计算的，甚至不必是递归的。我们说 $f(m)$ 构成了关于消息 $m \in M$ 的**信息**。在实践中，感兴趣的典型 f 是恒等函数、布尔谓词、哈希函数等。

我们希望从消息的编码中提取关于消息的任何信息应该是困难的，即使已知与消息空间相关的概率分布。

令 M 为一个消息空间，f 为定义在 M 上的函数。对于所有 $m \in M$，令 $p_m = \operatorname{Prob}(x = m \mid x \in M)$。考虑像 $f(M)$。定义 $p^M = \max_{v \in V} (\sum_{m \in f^{-1}(v)} p_m)$，并令 v^M 为 $f(M)$ 中达到最大概率的值。令 E 为一个加密算法。考虑以下三个游戏。假设对手知道 E。

游戏 1。在 M 中随机选取 m（每个 $x \in M$ 被选取的概率为 p_x）。在这个游戏中，要求对手在不被告知 m 是什么的情况下猜测 $f(m)$ 的值。

如果对手总是猜测 v^M，他将以概率 p^M 正确。对手没有策略可以给他更好的获胜概率。

游戏 2。在 M 中随机选取 m。计算一个加密 $\alpha \in E(m)$。将 α 给对手。现在，要求对手猜测 $f(m)$。

游戏 3。让对手选择一个定义在 M 上的函数 f_{E}。在 M 中随机选取 m。计算一个加密 $\alpha \in E(m)$。将 α 给对手。现在，要求对手猜测 $f_{E}(m)$。

非正式地，我们说 π 是一个**语义安全的公钥密码系统**，如果对手在游戏 3 中的胜率不能高于游戏 1。

正式设定

定义（语义安全的公钥密码系统）。令 Π 为一个公钥密码系统。令 MG 为一个消息生成器。如前所述 $M_{k} = \mathrm{MG}[k]$。对于所有 $m \in M_{k}$，p_{m} 将表示 MG 在输入 k 时输出 m 的概率。令 $f_{\mathrm{MG}} = \{f_{E} : M_{k} \to V \mid E \in \Pi(k, \mathrm{MG}), k \in N\}$ 为 MG 上的一组函数。对于每个 $E \in \Pi(k, \mathrm{MG})$，令 $p_{E} = \max_{v \in V} (\sum_{m \in f_{E}^{-1}} p_{m})$。

令 C 为一个电路，在输入 $E \in \Pi(k, \mathbf{M}G)$ 和 $\alpha \in E(m)$（其中 $m \in M_k$）时输出一个字符串 y。令 P, Q 为多项式。我们说 C **(P, Q, k)-计算** $f_{\mathbf{MG}}$ 从 Π，如果 $\operatorname{Prob}(y = f_E(m) \mid m \in M_k, \alpha \in E(m)) > p_E + (1 / Q(k))$ 对于所有属于子集 $S \subseteq \Pi(k, \mathbf{M}G)$ 的 E 成立，其中 S 的概率至少为 $1 / P(k)$。

令 P, Q 为多项式。令 $C_k^{Q,P}$ 表示能够 (P, Q, k)-计算 $f_{\mathbf{MG}}$ 从 Π 的最小规模电路 C 的规模。

我们说 Π 是**语义安全的**，如果对于所有 $\mathbf{MG}$，所有 $f_{\mathbf{MG}}$，所有 P, Q，$C_k^{P,Q}$ 的增长速度比 k 的任何多项式都快。

定理 5.2。每个多项式安全的公钥密码系统都是语义安全的。

证明。令 Π 为一个多项式安全的公钥密码系统。

假设矛盾地，Π 不是语义安全的。那么存在一个消息生成器 $\mathbf{MG}$，$\mathbf{MG}$ 的一组函数 $f_{\mathbf{MG}} = \{f_E\}$，多项式 P_1, P_2 和 Q，一个无限子集 $N' \subseteq N$ 以及一个电路序列 $\{C_k\}$，使得：

(1) C_k 的门数少于 $P_2(k)$，

(2) 子集 $S_{k}\subseteq \Pi (k,\mathbf{M}G)$ 的概率大于 $1 / P(k)$，且

(3) 对于所有 $E \in S_k$，在输入 E 和 $\alpha \in E(m)$（其中 $m \in \mathbf{MG}[k]$）时，C 将以（关于输入 α 的）大于 $p_E + (1 / Q(k))$ 的概率输出 $f_E(m)$。

在证明的剩余部分，k 将属于 N'，i 属于 S_k。令 $\varepsilon_k = 1 / Q(k)$，$p_E = \max_{v \in V} \sum_{m \in f_E^{-1}(v)} p_m$。

令 $r_{m,y}^{E}$ 表示 C_k 在输入 E 和 $\alpha \in E(m)$ 时输出 y 的概率。那么，$r_{m,f_E(m)}^{E}$ 是 C_k 在输入 E 和 $\alpha \in E(m)$ 时正确求值 f_E 的概率。

因此，我们为矛盾而假设的条件可以表示为

从 M_{k} 中选取 μ 并在证明的剩余部分固定它。定义 $\bar{M} \subseteq M_{k}$ 为满足以下条件的消息 m 的集合：

我们观察到以下两个引理。

引理 A。对于所有常数 $c > 0$，存在一个概率 $\text{poly}(k)$ 时间算法，在输入 $i \in S_k$ 和 $\xi \in \overline{M}$ 时，以概率 $1 - (1 / k^c)$ 找到一个 $v \in V$ 使得

证明。使用加密算法 E 构造消息 ξ 的编码的随机样本。令 $\{x_{1},\ldots,x_{s}\}$ 表示这个样本。对于 $1 \leqslant j \leqslant s$ 计算 $C_{k}[E,x_{j}]$。令

并对所有 $v \in V$（使得 $C[E, x_{j}] = v$ 对于某个 j 在 1 到 s 之间）设 $\alpha_{v} = \sum_{1 \leqslant j \leqslant s} I_{v}(x_{j}) / s$。至多有 s 个 V 中的值使得这个频率非零。

类似地，使用加密算法 E_{i} 构造消息 μ 的编码的随机样本。令 $\{y_1,\dots,y_s\}$ 表示这个样本。对所有 $v\in V$（使得 $C_k[E,y_j] = v$ 对于某个 j 在 1 到 s 之间）设 $\beta_{v} = \sum_{1\leqslant j\leqslant s}I_{v}(y_{j}) / s$。检查两个列表（每个大小小于 s）的 α_{v} 和 β_{v}。如果存在 $\bar{v}$ 至少在其中一个列表中，使得 $|\alpha_{\bar{v}} - \beta_{\bar{v}}| > 3\varepsilon_k^2 / 40$，则输出 $\bar{v}$。

我们声称，对于样本大小 s 的适当选择，这个输出正确的概率为 $1 - 1 / k^{c}$。推理如下。设 $s = 1 / (4[1 / 2k^c][\varepsilon_k^2 /80]^2)$。那么，对于满足 $|r_{\mu ,v}^{E} - r_{\xi ,v}^{E}| > \varepsilon_{k}^{2} / 10$ 的 v。（记住这样的 v 存在，因为 $\xi \in \widetilde{M}$），弱大数定律保证，

且

最后，

反之，对于一个满足 $|\alpha_v - \beta_v| > 3\varepsilon_k^2 /40$ 的 v，有

引理 B。$\sum_{m\in \overline{M}}p_m > \varepsilon_k / 10.$

证明。令 $V_{3} = \{v \in V | r_{\mu, v} > \varepsilon_{k} / 6\}$，$V_{4} = \{v \in V | r_{\mu, v} \leqslant \varepsilon_{k} / 6\}$，以及分别地，$M_{3} = \{m \in M_{k} - \overline{M} | r_{\mu, f_{E}(m)} > \varepsilon_{k} / 6\}$ 和 $M_{4} = M_{k} - \overline{M} - M_{3}$。$M_{3}$ 包括所有 $m \notin \overline{M}$ 使得 $f_{E}(m) \in V_{3}$，M_{4} 包括所有 $m \notin \overline{M}$ 使得 $f_{E}(m)$ 不在 V_{3} 中。显然，$l = |V_{3}| < 6 / \varepsilon_{k}$。将 V_{3} 中的值表示为 $\{v_{1}, \dots, v_{l}\}$。那么，

（由于 $\forall m \notin \overline{M}$，$|r_{m, f_E(m)}^E - r_{\mu, f_E(m)}^E| < \varepsilon_k^2 / 10$）这小于或等于

重新排列等式两边后，我们得到 $\sum_{m\in \widetilde{M}}p_m > \varepsilon_k / 10$

引理 A 和 B 意味着对于所有 $k \in N'$，存在一个 $\text{poly}(k)$ 电路 F_{k}，使得在输入 $E \in S_{k}$ 时，F_{k} 产生 M_{k} 中的两个消息 m_{1} 和 m_{2} 以及 $f^{-1}(M_{k})$ 中的一个值 v，使得 $|r_{m_{1},v}^{E} - r_{m_{2},v}^{E}| > \varepsilon_{k}^{2}/20$。

F_{k} 工作如下。在输入 E 时，它在 M_{k} 中随机选取一个 μ。然后，它在 M_{k} 中随机生成一个元素 ξ。（根据引理 B，以至少 $\varepsilon_{k} / 10$ 的概率，$\xi \in \bar{M}$；如果不是，不用担心。）然后使用引理 A 寻找 $v \in V$，使得以高概率 $|r_{\xi, v} - r_{\mu, v}| > \varepsilon_{k}^{2} / 20$。如果没有找到这样的 v，可能是因为 ξ 本来就不在 $\bar{M}$ 中，我们选择另一个 ξ，直到在预期的多项式次尝试后成功。如果 v 被找到，设 $m_{1} = \xi$，$m_{2} = \mu$。

现在，定义 $T_{k}[i,x] = 1$ 如果 $C_k[i,x] = v$，否则为 0。那么 T_{k} 是一个 $\text{poly}(k)$ 线路窃听者，$(\varepsilon_{k}^{2} / 20)$-区分 F_{k} 找到的两个消息 m_{1} 和 m_{2}。这与 Π 是多项式安全的公钥密码系统的假设相矛盾。

6. 二次剩余性问题 (QRP)

我们引入一个新的基于二次剩余性假设的陷门数论谓词。

令 x 和 y 为整数。符号 (x, y) 将表示 x 和 n 的最大公约数。符号 $\operatorname{Prob}(X)$ 将表示事件 X 的概率。令 N 表示正整数集合，$n \in N$。令 $Z_{n}^{*} = \{x \mid 1 \leqslant x \leqslant n - 1$ 且 $(x, n) = 1\}$。

6.1. 背景和记号

给定 $q \in Z_n^*$，$q \equiv x^2 \mod n$ 是否可解？如果 n 是素数，那么这个问题很容易计算 [16]：如果 $q^{(n-1)/2} \mod n = 1$，则是；如果 $q^{(n-1)/2} \mod n = -1$，则否。如果解存在，q 被称为模 n 的**二次剩余**。否则，q 被称为模 n 的**二次非剩余**。在本节中，p_1 和 p_2 将是不同的奇素数，$n = p_1p_2$。那么，$q \equiv x^2 \mod n$ 可解当且仅当 $q \equiv x^2 \mod p_1$ 和 $q \equiv x^2 \mod p_2$ 都可解。因此，如果知道 n 的因式分解，$q \equiv x^2 \mod n$ 的可解性很容易判定。

引理 1。给定一个合数 n 的素因子分解，判定 $q \in Z_n^*$ 是否是模 n 的二次剩余可以在 $O(|n|^3)$ 时间内完成。

当 n 的因式分解未知时，可以从雅可比符号获得关于判定一个数是否是模 n 的二次剩余的一些信息。令 p 为奇素数，$q \in Z_p^*$，那么雅可比符号 (q / p) 等于 1 如果 q 是模 p 的二次剩余，否则为 -1。雅可比符号 (q / n) 定义为 $(q / n) = (q / p_1)(q / p_2)$。尽管雅可比符号 (q / n) 是通过 n 的因式分解定义的，但 (q / n) 即使在不知道 n 的因式分解时也是多项式时间可计算的！

从上述定义容易看出，如果 $(q / n) = -1$，那么 q 必须是模 n 的二次非剩余。事实上，q 必须是模 p_1 或模 p_2 的二次非剩余。然而，如果 $(q / n) = +1$，那么要么 q 是模 n 的二次剩余，要么 q 是模 n 的两个素因子的二次非剩余。

在本文中，我们对那些雅可比符号为 $+1$ 的 Z_{n}^{*} 元素感兴趣。因此我们引入集合，

让我们计算 Z_{n}^{+1} 的元素个数。证明见 [16]。

事实 1。令 p 为奇素数。那么 Z_p^* 是一个循环群。

事实 2。令 g 为 Z_p^* 的生成元，那么 $g^s \bmod p$ 是二次剩余当且仅当 s 是偶数。

推论 3。$\mathbb{Z}_p^*$ 中一半的数是二次剩余，一半是二次非剩余。

事实 4。令 $n = p_1p_2$（p_1 和 p_2 是不同的奇素数）。那么 Z_n^* 中一半的数的雅可比符号等于 -1，因此是二次非剩余。其余数的雅可比符号为 1。这些数中恰好一半是模 n 的二次剩余。

6.2. 二次剩余性假设

令 n 为合数，q 为 Z_{n}^{+1} 的元素。具有参数 q 和 n 的**二次剩余性问题**是判定 q 是否是模 n 的二次剩余。如果不知道 n 的因式分解，那么没有已知的有效过程来解决具有参数 n 和 Z_{n}^{+1} 中的 q 的二次剩余性问题。这个判定问题是数论中一个著名的难题。它是高斯 [8] 在他的《算术研究》（1801）中讨论的四个主要算法问题之一。它的多项式解将意味着数论中其他开放问题的多项式解。一个例子是判定一个合数 n 是 2 个还是 3 个素数的乘积（见 Adleman [2] 中的开放问题 9 和 15）。

为了正式陈述二次剩余性问题的难解性假设，让我们引入谓词 Q_{n} 和困难合数集合 H_{k}。对于所有 $x \in Z_{n}^{1}$，谓词 Q_{n} 定义为：

H_{k} 将表示困难合数整数的集合：令 p_1 和 p_2 表示素数。

H_{k} 的元素构成了任何已知因子分解算法的最困难输入。

二次剩余性假设 (QRA)

令 P_{1} 为固定多项式。对于每个整数 k，令 C 为具有两个 $2k$ 位输入和一个布尔输出的电路。令 C_{k} 为电路 C 的最小规模，使得对于 $1 / P_{1}(k)$ 部分的 $n \in H_{k}$，$C[n, x] = Q_{n}(x)$ 对于所有 $x \in Z_{n}^{+1}$ 成立。那么，对于所有多项式 Q，对于所有足够大的 k：$C_{k} > Q(k)$。

接下来，我们证明在 QRA 下，计算 $Q_{n}(x)$ 不仅对于某些特殊的 $x \in Z_{n}^{1}$ 是困难的，而且在平均意义上也是困难的。

6.3. 一个数论结果

我们回忆一下，电路 $C[\cdot]$ 是谓词 $B: \Omega \to \{0,1\}$ 的近似，如果 $C[x] = B[x]$ 对于至少 $\frac{1}{2} + \varepsilon$ 部分的 $x \in \Omega$ 成立。

让我们回忆弱大数定律：

弱大数定律

令 $y_{1},y_{2},\dots,y_{r}$ 为 r 个独立的 0-1 变量，使得 $y_{i} = 1$ 的概率为 p，并且 $S_{r} = \sum_{i = 1}^{r}y_{i}$，那么对于实数 $\psi, \delta > 0$，$r \geqslant 1 / 4\delta \psi^2$ 意味着 $\mathrm{Prob}(|(S_r / r) - p| > \psi) < \delta$。注意 r 以 ψ^{-1} 和 δ^{-1} 的多项式为界。

关于定理 1 的备注。定理 1 表明，判定模 n 的二次剩余性要么是"处处困难"，要么是"处处容易"。这个定理的主要思想是"如何收集随机优势"，即如何将一个能正确回答大多数问题但你不知道是哪些问题的预言机，变成一个能以任意高概率正确回答每个问题的预言机。

定理 1。固定多项式 P_1 和 P_2，令 $O[\cdot, \cdot] \colon N \times N \to \{0,1\}$ 为一个预言机。令 S 为困难整数 n 的集合，使得 $O[\cdot, n]$ 对于 Q_n 是 $(1 / P_1(|n|))$ 近似的。那么存在一个具有预言机 O 的概率 $\text{poly}(|n|)$ 算法，对于任何 $n \in S$ 和任何 $x \in Z_n^1$，以大于 $1 - (1 / P_2(|n|))$ 的概率正确判定 x 是否是模 n 的二次剩余。

证明。令 $n \in S$。取 Z_{n}^{1} 具有均匀概率分布。为了记号的简便，令 $\varepsilon = 1 / P_{1}(|n|)$，$\delta = 1 / P_{2}(|n|)$。那么，$\operatorname{Prob}(O[q,n] = Q_{n}(q) | q \in Z_{n}^{1}) > \frac{1}{2} + \varepsilon$。令 $\alpha = \operatorname{Prob}(O[q,n] = 1 | Q_{n}(q) = 1)$，$\beta = \operatorname{Prob}(O[q,n] = 1 | Q_{n}(q) = 0)$。

$\operatorname{Prob}(O[q, n] = Q_n(q) \mid q \in Z_n^1) = \frac{1}{2}\alpha + \frac{1}{2}(1 - \beta) > \frac{1}{2} + \varepsilon$。因此，$\alpha - \beta \geqslant 2\varepsilon$，但 α 可以远小于 $\frac{1}{2} + \varepsilon$。我们首先需要获得 α 的良好估计。

在 Z_{n}^{*} 中构造 r 个随机选择的二次剩余的样本（r 的值将在后面定义）。这可以通过在 Z_{n}^{*} 中随机选择 $s_{1},\ldots ,s_{r}$ 并对它们取平方模 n 来轻松完成。将计数器 C 初始化为 0。

对于 $i = 1$ 到 r，向预言机询问值 $O[s_i^2 \bmod n, n]$。每次预言机回答 1（即"二次剩余"）时递增 C。

令 $\psi = \varepsilon / 2$。如果 r 被选择得足够大，$r = 1 / \delta \psi^2$，弱大数定律保证 C / r 是 α 的一个良好的 $(\varepsilon / 2)$-估计：

即，C / r 很好地近似了如果输入仅为二次剩余时预言机"猜测" Q_{n} 的效果。

现在我们准备描述一个用于确定 Z_{n}^{1} 中任何元素的二次剩余性的过程。令 q 为我们想要测试二次剩余性的 Z_{n}^{1} 中的一个元素。随机生成 r 个二次剩余，$x_{1},\ldots ,x_{r}$，在 Z_{n}^{1} 中，并计算 $y_{i}\equiv qx_{i}\bmod n$，对于 $i = 1,\dots,r$。注意

(1) 如果 q 是二次剩余，那么 y_{i} 是随机二次剩余，

(2) 如果 q 是 Z_{n}^{1} 中的二次非剩余，那么 y_{i} 是随机二次非剩余。

让我们推迟 (1) 和 (2) 的证明，并暂时假设它们是正确的。将计数器 $\overline{C}$ 初始化为 0。对于 $i = 1$ 到 k，调用预言机获取值 $O[y_i, n]$。每次预言机回答 1 时递增 $\overline{C}$。如果 $|(C / r) - (\overline{C} / r)| < \varepsilon$ 则输出"q 是模 n 的二次剩余"，否则输出"q 是模 n 的二次非剩余"。

由于

且

那么

Prob(回答 q 是二次非剩余 $\mid q$ 是二次非剩余)

因此，任何 $q \in Z_n^1$ 的二次剩余性以大于 $1 - \delta$ 的概率被正确判定。

我们仍然需要证明 (1) 和 (2)。为了证明 (1)，只需证明给定任何二次剩余 q，Z_{n}^{*} 中的任何其他二次剩余 y 可以唯一地写成 $y = qx \mod n$，其中 x 也是模 n 的二次剩余。令 g_{1} 和 g_{2} 分别为 $Z_{p_{1}}^{*}$ 和 $Z_{p_{2}}^{*}$ 的生成元。令 a 和 b 满足 $a \equiv g_{1} \mod p_1$，$a \equiv 1 \mod p_2$，且 $b \equiv 1 \mod p_1$ 和 $b \equiv g_{2} \mod p_2$。根据中国剩余定理，这样的 a 和 b 存在。那么，Z_{n}^{*} 中的任何元素可以唯一地写成 $a^{i}b^{j} \mod n$，其中 $1 \leqslant i \leqslant p_{1} - 1$ 且 $1 \leqslant j \leqslant p_{2} - 1$。此外，$q$ 是模 n 的二次剩余当且仅当它可以写成 $q = a^{2i}b^{2j} \mod n$，其中 $1 \leqslant 2i \leqslant p_{1} - 1$ 且 $1 \leqslant 2j \leqslant p_{2} - 1$。因此，如果 $y = a^{2s}b^{2t} \mod n$ 是任何二次剩余，则存在唯一的 x，模 n 的二次剩余，$x = a^{2(s - i)}b^{2(t - j)}$，使得 $y = qx \mod n$。这证明了 (1)；(2) 以类似方式证明。

推论 1。固定多项式 P_{1} 和 P_{2}。令 $k \in N$。令 C_k 为最小规模电路 C 的规模，使得对于 $1 / P_1(k)$ 部分的 $n \in H_k$，C $(1 / P_2(k))$-近似 Q_n。在 QRA 下，对于所有多项式 Q，对于所有足够大的 k：$C_k > Q(k)$。

证明。假设矛盾地，存在多项式 P_1, P_2 和 Q 以及一个无限 $\bar{N} \subseteq N$，使得对于所有 $k \in \bar{N}$：$C_k < Q(k)$。那么，对于每个 $k \in \bar{N}$，令 S_k 包含 H_k 中 $1 / P_1(k)$ 部分的元素，并且 $\bar{C}_k$ 是一个规模为 C_k 的电路，使得对于所有 $n \in S_k$，$\bar{C}_k[x,n] = QR_n(x)$ 对于至少 $\frac{1}{2} + (1 / P_2(k))$ 的 Z_n^{+1} 元素成立。

对于每个 $k \in \overline{N}$，选择定理 1 中的预言机 O 为 $\bar{C}_k$。即，设 $O[x,n] = \bar{C}_k[x,n]$ 对于所有 $n \in S_k$ 和所有 $x \in Z_n^1$。那么，根据定理 1，对于所有 $k \in \overline{N}$，所有 $n \in S_k$，所有 $x \in Z_n^1$，以及所有多项式 P_3，存在一个具有预言机 $\bar{C}_k$ 的概率多项式（在 k 内）时间算法，以大于 $1 - (1 / P_3(k))$ 的概率正确判定 $x \mod n$ 的二次剩余性。由于 C_k 的规模小于 $Q(k)$，对于所有 $k \in \overline{N}$，这样的算法可以转换成一个 k 的多项式规模电路，该电路对于所有 $n \in S_k$ 正确判定模 n 的二次剩余性。由于 $|S_k| > (1 / P_1(k))|H_k|$，这与 QRA 矛盾。

令 n 为一个因式分解未知的合数。我们想研究，当我们被额外告知一个特定的 $y \in Z_n^1$ 是模 n 的二次非剩余时，判定模 n 的二次剩余性的难度会发生什么变化。

关于定理 2 的备注。当 n 的因式分解是秘密的时，没有已知的选择模 n 的二次非剩余的有效算法。因此，揭示，比如，Z_{n}^{1} 中最小的二次非剩余，可能会危及 n 的因式分解的秘密，或者使得判定模 n 的二次剩余性变得容易。

定理 2 表明，如果揭示一个随机选择的模 n 的二次非剩余，二次剩余性问题的复杂性保持不变。换句话说：假设对于 Z_n^1 中多项式部分的二次非剩余 x，知道 x 确实是模 n 的二次非剩余将导致模 n 的二次剩余性的有效判定过程。那么，即使没有这种额外的帮助，模 n 的二次剩余性也本可以被有效判定。

定理 2。令 P_{1} 和 P_{2} 为固定多项式。对于每个 $k \in N$，令 $E_{k} \subseteq H_{k}$ 包含 H_{k} 中 $1 / P_{1}(k)$ 部分的整数。对于每个 $n \in E_{k}$，令 S_{n} 包含 Z_{n}^{1} 中 $1 / P_{2}(k)$ 部分的二次非剩余。令 C_{k} 为最小规模电路 $C[\cdot, \cdot, \cdot]$ 的规模，使得对于所有 $n \in E_{k}$，所有 $s \in S_{n}$，以及所有 $x \in Z_{n}^{1}$，$C[n, s, x] = Q_{n}(x)$。那么，对于所有多项式 Q，对于所有足够大的 k：$C_{k} > Q(k)$。

证明。令 $k \in N$。固定多项式 P_{1} 和 P_{2}。令 $C[\cdot, \cdot, \cdot]$ 为一个规模为 C_{k} 的电路，使得 $C[n, y, q] = Q_{n}(q)$ 对于所有 $n \in E_{k}, y \in S_{n}, q \in Z_{n}^{1}$ 成立。证明分为 3 部分：

(1) 存在一个具有预言机 $C[\cdot, \cdot, \cdot]$ 的概率算法 A_{1}，在输入 $n \in E_{k}$ 时，输出 $x \in Z_{n}^{1}$，使得以大于 $1 - (1 / P_{2}(k))$ 的概率，$C[n, x, \cdot]$ $(1 / P_2(k))$-近似 $Q_n(\cdot)$。算法 A_{1} 在期望时间内终止，该时间是 k 的多项式。

(2) 算法 A_{1} 可以转换成一个规模为 k 和 C_{k} 的多项式的电路 $C_{1}[\cdot, \cdot]$，使得对于所有 $n \in E_{k}, q \in Z_{n}^{1}, C_{1}[n, q] = Q_{n}(q)$。

(3) 根据 QRA，对于所有足够大的 k，C_1 的规模超过任何给定的 k 的多项式。因此，再次对于足够大的 k，对于任何给定的多项式 Q，$C_k > Q(k)$。

我们继续证明第 (1) 部分。在输入 $n \in E_k$ 时，定义算法 A_1 如下：

重复

(1) 从 Z_{n}^{1} 中随机选择 x。

(2) 从 Z_n^1 中随机选择 k 个元素 $e_1, \ldots, e_k$。（注释：这可以在概率 $\text{poly}(k)$ 时间内完成，通过以均匀概率选择 $[1, n]$ 中的元素 r，并检查是否 $r \in Z_n^*$ 且 $(r / n) = 1$）。

（注释：以大于 $1 - (1/2^k)$ 的概率，其中一个 e_i 是模 n 的二次非剩余。）

(3) 令 $e_0 = 1$

(4) 对于 $i = 0,\dots,n, j = 1,\dots,k$

(5) 选择模 n 的随机二次剩余的样本 $x_{1},\ldots ,x_{k}$，并计算 $y_{i,j} = e_i x_j \mod n$。

（注释：由于 $e_0 = 1$，$\{y_{0,1},\dots,y_{0,k}\}$ 是模 n 的随机二次剩余的样本。以大于 $1 - (1 / 2^k)$ 的概率，对于某个 $i > 0$，$\{y_{i,1},\dots,y_{i,k}\}$ 是 Z_n^1 中二次非剩余的样本。）

(6) 对于 $i = 0,\dots ,k$

(7) 设 $f_{i}^{x} = (\sum_{j=0}^{k} C[n, x, y_{i,j}] / k)$。

（注释：f_{i}^{x} 估计了 $C[n, x, \cdot]$ 在 Z_{n}^{1} 中那些二次特征与 e_{i} 相同的元素上输出 1 的概率。）

直到 $f_0^x = 1$ 且 $f_i^x = 0$ 对于某个 $i \geqslant 1$。

输出 x

我们现在证明，以大于 $1 - (1 / P_{2}(k))$ 的概率，算法 A_{1} 计算出的 x 使得 $C[n,x,\cdot]$ $(1 / P_{2}(k))$-近似 $Q_{n}(\cdot)$。令 $\alpha_{x} = \operatorname{Prob}(C[n,x,q] = 0 \mid Q_{n}(q) = 0)$，$\beta_{x} = \operatorname{Prob}(C[n,x,q] = 0 \mid Q_{n}(q) = 1)$。那么，由于 $f_{0}^{x} = 1$ 且对于某个 $i \geqslant 1$ 有 $f_{i}^{x} = 0$，那么对于所有足够大的 k，弱大数定律向我们保证 $|\alpha_{x} - \beta_{x}| > (1 / 2P_{2}(k))$。根据定理 1，这意味着 $C[n,x,\cdot]$ $P_{2}(k)$-近似 $Q_{n}(\cdot)$。

最后，关于 A_{1} 的运行时间。注意，如果在算法的一次迭代中，我们从 S_{n} 中抽取一个 x，并且其中一个 e_{i} 是二次非剩余，那么 $f_{0}^{x} = 1$ 且 $f_{i}^{x} = 0$，算法终止。因此，算法 A_{1} 执行的期望迭代次数为

由于每次迭代可以在概率 $\text{poly}(k)$ 时间内执行，A_{1} 在期望多项式（在 k 内）时间内运行。这证明了第 (1) 部分。

第 (2) 部分由推论 1 以及将概率算法转换为电路的标准方法得出。第 (3) 部分很容易从第 (2) 部分得出。

推论 2。令 P_1, P_2 和 P_3 为固定多项式。对于每个 $k \in N$，令 $E_k \subseteq H_k$ 包含 H_k 中 $1 / P_1(k)$ 部分的整数。对于每个 $n \in E_k$，令 S_n 为 Z_n^1 中 $1 / P_2(k)$ 部分的二次非剩余。令 C_k 为最小规模电路 $C[\cdot, \cdot, \cdot]$ 的规模，使得在输入 $n \in E_k$ 和 $s \in S_n$ 时，$(1 / P_3(k))$-近似 Q_n。那么，对于所有多项式 Q，对于所有足够大的 k：$C_k > Q(k)$。

这个推论说明的是，假设 QRA，当用户 B 被呈现 (n, y)，其中 $n \in H_k$ 且 y 是 Z_n^1 中的二次非剩余，以及 $x \in Z_n^1$，他无法以大于 $\frac{1}{2}$ 的概率猜测 $Q_n(x)$。

6.4. 二次剩余性的一个特殊性质

令 $n \in H_k$，$\alpha = (x_1, \ldots, x_k)$ 为使用谓词 Q_n 对 k 比特消息 m 的概率加密。给定 α，任何人，即使不知道 n 的因式分解，也可以重新加密 m。事实上，他可以通过简单地将每个 x_i 乘以一个不同的、随机选择的模 n 的二次剩余，以均匀概率选择 m 的另一个概率加密。

这个性质已被 Luby、Micali 和 Rackoff 在 [19] 中用于公平交换秘密比特。

7. 最终备注

7.1. 电路与图灵机

令 A 为公钥密码系统中的用户，k 为 A 放在公共文件中的加密算法 E_A 描述中的比特数。假设有人（最终）证明，对于所有多项式时间图灵机 M，存在一个常数 k_M，使得对于所有 $k > k_M$，在某个消息空间上反转 E_A 需要 $\Omega(2^{\sqrt{k}})$ 步。由于被动窃听者有权在 E_A 放入公共文件后选择 M，A 应该选择什么 k？

正是为了消除这个困难，我们选择了电路复杂性作为复杂性度量。应该注意，证明我们的定理并不需要这样的选择。原本可以假设关于概率多项式时间图灵机的难解性，并以基本相同的方式证明所有定理。

7.2. 其他类型的对手

在公钥密码系统中，通过窃听获取密文并试图通过计算解密，是最明显的攻击。然而，这并不是唯一的攻击！Goldwasser、Micali 和 Tong [9] 展示了在 Diffie 和 Hellman 的公钥密码系统模型中，对手如何通过通信（作为用户）破坏方案的安全性。他们提出了 Diffie 和 Hellman 模型的修改，并展示了新模型对于线路窃听者甚至是选择密文攻击是安全的。

7.3. Shannon 完美保密定义与语义安全性之间的关系

让我们描述 Shannon 在 [23] 中"完美保密"的定义。考虑一个具有无限时间和人力用于分析截获密码的对手。令所有可能消息的集合是有限的。这些消息具有先验概率并被编码和发送。当对手截获一个编码消息时，他可以计算各种消息的后验概率。如果对于所有编码消息，后验概率等于先验概率，则实现了完美保密。因此截获消息不会给对手任何信息。在本文中，我们定义了 Shannon 完美保密的多项式有界版本，称为语义安全性。语义安全性意味着当对手只有多项式有界资源可用时，截获编码消息不会给他任何新信息。此外，不存在对手在截获编码消息后可以计算的、定义在消息集合上的函数，是他不截获消息时无法计算的。进一步讨论见 [26]。

致谢

我们最诚挚的感谢归于 Manuel Blum 和 Richard Karp，他们指导了这项研究，感谢他们的鼓励和他们如此乐意与我们分享的绝妙想法。我们特别感谢 Zvi Galil、Mike Luby、Charles Rackoff 和 Ron Rivest 在澄清本文中的思想和表达方面提供的慷慨帮助。还要感谢 Steve Cook、Faith Fich、Jeff Shallit、Mike Sipser 以及审稿人提供的许多关于形式和内容的想法、评论和批评。Vijai Vazirani 在第 2.3.1 小节的声称中提供了帮助。

参考文献

1. L. ADLEMAN, K. MANDERS, AND G. MILLER, On taking roots in finite fields, in “Proceedings of the 18th Annual IEEE Symposium on Foundations of Computer Science,” pp. 175–177, 1977.
2. L. ADLEMAN, On distinguishing prime numbers from composite numbers, in “Proceedings of the 21st IEEE Symposium on the Foundations of Computer Science,” pp. 387–408, Syracuse, N.Y., 1980.
3. M. BLUM, Coin flipping by telephone, in "Proceedings of the IEEE, Spring Comp-Con, pp. 133-137, 1982.
4. L. BLUM, M. BLUM, AND M. SHUB, “A Simple Secure Pseudo-Random Number Generator,” CRYPTO, 1982.
5. M. BLUM AND S. MICALI, How to generate cryptographically strong sequences of pseudo random bits, in "Proceedings of the 23rd IEEE on the Foundations of Computer Science," Chicago, Ill., 1982.
6. G. BRASSARD, Relativized cryptography, in “Proceedings of the 20th IEEE Symposium on the Foundations of Computer Science,” pp. 383-391, San Juan,

Puerto Rico, 1979.
7. G. BRASSARD, On computationally secure authentication tags requiring short secret shared keys, CRYPTO, 1982.
8. C. F. GAUSS, “Disquisitiones Arithmeticae,” 1801, translated by A. Arthur and S. J. Clark, Yale Univ. Press, New Haven, 1966.
9. W. DIFFIE AND M. E. HELLMAN, New direction in cryptography, IEEE Trans. Inform. Theory IT-22 (6) (1976), 644-654.
10. S. GOLDWASSER AND S. MICALI, “A Bit by Bit Secure Public Key Cryptosystem,” Memorandum No. UCB/ERALM81/88, University of California, Berkeley, December 1981.
11. S. GOLDWASSER AND S. MICALI, Probabilistic encryption & how to play mental poker, keeping secret all partial information, in "Proceeding of 14th STOC Conference," San Francisco, 1982.
12. S. GOLDWASSER, S. MICALI, AND P. TONG, Why and how to establish a private code in a public network, in “Proceedings of the 23rd Symposium on Foundations of Computer Science,” Chicago, Ill., 1982.
13. S. GOLDWASSER, "Probabilistic Encryption: Theory and Applications," Ph.D. thesis, Univ. of California at Berkeley, 1983.
14. S. GOLDWASSER, S. MICALI, AND A. YAO, Strong signature schemes and authentication, in "Proceedings, 15th STOC," Boston, Mass., 1983.
15. K. R. Guy, How to factor a number, in “Proceedings of Fifth Manitoba Conference on Numerical Math,” pp. 49-89, 1975.
16. D. KNUTH, “The Art of Computer Programming,” Vol. 2, 2nd ed., Addison-Wellesley, Reading, Mass., 1981.
17. R. LIPTON, How to cheat at mental poker, in “Proceeding of the AMS Short Course on Cryptology,” January 1981.
18. G. MILLER, Riemann's hypothesis and tests for primality, Ph.D. thesis, U.C. Berkeley, 1975.
19. M. LUBY, S. MICALI, AND C. RACKOFF, How to simultaneously exchange a secret bit by flipping a symmetrically-biased coin, FOCS 1983.
20. M. RABIN, Digitalized signatures and public-key functions as intractable as factorization, MIT/ LCS/TR-212, Technical Memo MIT, 1979.
21. R. RIVEST, A. SHAMIR, AND L. ADLEMAN, A method for obtaining digital signatures and public key cryptosystems, Communications of the ACM, February 1978.
22. A. SHAMIR, R. RIVEST, AND L. ADLEMAN, "Mental Poker," MIT Technical Report, 1978.
23. C. E. SHANNON, Communication theory of secrecy systems, Bell System Tech. J. 28 (1949), 656-715.
24. D. SHANKS, "Solved and Unsolved Problems in Number Theory," Chelsea, New York, 1978.
25. V. VAZIRANI AND U. VAZIRANI, Secure one-bit disclosures using a pseudo random number generator, in "Proceedings, FOCS," 1983.
26. A. YAO, On the theory and application of trapdoor functions, in “Proceedings of the 23rd Symposium on the Foundations of Computer Science,” Chicago, Ill., November 1982.