如何生成与交换秘密

（扩展摘要）

Andrew Chi-Chih Yao*

计算机科学系

普林斯顿大学

新泽西州普林斯顿市 08544

摘要

本文我们引入一种新的工具，用于控制密码协议设计中的知识转移过程。该工具被应用于解决一大类问题，涵盖了文献中大多数双方密码学问题。

具体而言，我们展示了双方 A 和 B 如何交互式地生成一个随机整数 $N = p \cdot q$，使得其秘密（即素因子 (p, q)）对任何一方单独隐藏，但如果需要可以共同恢复。这可用于为拥有私有值 i 和 j 的两方提供一个协议，以计算任何多项式可计算函数 $f(i, j)$ 和 $g(i, j)$，实现最小的知识转移和强公平性。作为一个特例，A 和 B 可以交换一对秘密 s_A, s_B（例如，一个整数的分解和一个图中的哈密顿回路），其方式使得 s_A 在且仅在 s_B 对 A 可计算时对 B 可计算。所有这些结果的证明仅基于大整数分解问题在计算上是难解的假设。

1. 引言

协议 $\mathcal{M} = (M_A, M_B)$ 是一对通信的概率图灵机，每台机器都有一个特殊的“发送-接收”带。给定形式为 (n, i_A) 和 (n, i_B) 的输入，两台机器将交替使用发送-接收带发送和接收消息字符串；每台机器在接收到消息字符串后，将作为标准图灵机执行计算，包括计算要发送的下一条消息。最终，两台机器在 n 的某个多项式步数内停机，在输出带上留下字符串 u_A 和 u_B。关于详细描述，参见例如 [GMR] [GHY]；在我们的情况下，我们允许每台机器拥有自己的私有输入带。对于协议的任何一次运行 σ，令 $\Delta_A(\sigma)$ 表示从 A 的角度看到的运行历史，即 M_A 的一系列瞬时描述；类似地，$\Delta_B(\sigma)$ 表示从 B 的角度看到的该次运行历史。

我们感兴趣的问题类如下。给定输入 (n, i_A) 和 (n, i_B)，其中 (i_A, i_B) 是根据概率分布 h_n 在 $\{0, 1\}^* \times \{0, 1\}^*$ 上随机生成的一对字符串，我们希望设计一个协议 $\mathcal{M} = (M_A, M_B)$，使得输出 (u_A, u_B) 根据某个特定的概率分布 w_{n,i_A,i_B} 在 $\{0, 1\}^* \times \{0, 1\}^*$ 上分布。分布序列 $(h_1, h_2, \ldots, h_n, \ldots)$ 被假定为多项式合奏，即存在一个概率图灵机，给定输入 n，能在 n 的多项式

时间内生成一个随机字符串 x，其分布在计算上不可与 h_n 区分；类似地，我们假设 w_{n,i_A,i_B} 是一个多项式合奏，即当给定参数 n, i_A, i_B 时，可以在 n 的多项式时间内生成一个随机样本点。让我们称此为（双方）交互式计算问题 $\langle h_n, w_n \rangle$。特别有趣的情况是当概率分布 w_{n,i_A,i_B} 仅在一点非零，记该点为 $(f_n(i_A, i_B), g_n(i_A, i_B))$，在这种情况下我们可以将此问题视为评估一对函数 $f_n(i_A, i_B), g_n(i_A, i_B)$；将其写为 $\langle h_n, (f_n, g_n) \rangle$。

除了上述的有效性要求外，我们还希望协议具有某些隐私性和公平性。粗略地说，“隐私性”意味着如果 A 按照协议行事，那么 B 对于 i_A, u_A 的值不会比在由预言机为 B 完成计算并仅交给 B 一个值 u_B 的情况下获得更多信息。以隐私性为主要关注点的经典例子包括不经意传输（Rabin [R], Fischer et al [FRMW]）、抛币（Blum [B1], Cleve [C]）、心理扑克（Shamir, Rivest, and Adleman [SRA], Goldwasser and Micali [GM]）。关于该问题的一般性研究在 Yao [Y1, Y2] 中给出。一个相关问题是由 Goldwasser, Micali, and Rackoff [GMR] 以及 Galil, Haber, and Yung [GHY] 提出的交互式证明系统，其中玩家 A 希望说服玩家 B 一个字符串 τ 属于某个特定语言 L。最近，Goldreich, Micali, and Wigderson [GMW] 证明了，对于具有多项式时间计算能力的玩家，任何 NP 语言都存在一个最小知识的交互式证明，假设存在合适的单向函数。这后一个问题，在我们的表述中，与计算一对函数 $f_n(i_A, i_B), g_n(i_A, i_B)$ 的情况密切相关，其中 $i_A = (s, \tau)$，$i_B = \tau$，$g_n(i_A, i_B) = 1$ 当且仅当 s 是 $\tau \in L$ 的一个简短证明，并且 $f_n(i_A, i_B) \equiv 1$。

“公平性”要求意味着作弊者不应能够在否认另一方获得适当输出的同时，获得其期望的输出。交换秘密的问题（Blum [B2], Luby, Micali, and Rackoff [LMR], Vazirani and Vazirani [VV]）以此要求为主要关注点。

在本文中，我们将给出一个执行计算 $\langle h_n, w_n \rangle$ 的协议，在假设大整数分解是计算上难解的前提下，实现定理 3 所述的**有效性、隐私性和公平性**。许多提到的现有结果是此定理的特例。一个特别有趣的特例（陈述为定理 2）是它允许双方交换秘密，例如一个公开整数的分解或一个公开大图的哈密顿回路，其方式使得成功作弊的概率可以任意小。令人惊讶的是，被交换的两个秘密可能在表观复杂度上非常不同，并且人们可能会认为在交换秘密的过程中维持公平的比特比率是困难的。之前 Blum [B2] 给出了一个基于若干假设的交换素因子的协议（参见 Hastad and Shamir [HS] 的讨论）；Luby, Micali, and Rackoff [LMR] 以及 Vazirani and Vazirani [VV] 给出了交换单比特长秘密的协议。

定理 3 的证明依赖于两方能够生成一个随机整数 $N = p \cdot q$ 的能力，其秘密 (p, q) 对每一方都隐藏，但可以在后期通过共同努力恢复。这个结果本身具有独立的意义，并应作为密码协议设计的另一个有用工具。此结果的一般形式作为定理 1 给出。

与处理相同一般问题的 [Y1,Y2] 相比，**公平性**属性的加入是导致本项工作的新动机。需要指出的是，本文中考虑的所有用于解决交互式计算问题 $\langle h_n, w_n \rangle$ 的协议都将独立于 h_n，正如在考虑密码协议时的传统做法。主要结果（定理 1,2 和 3）均在以下假设下证明。令 W_n 为所有形式为 $p \cdot q$ 的整数 N 的集合，其中 $p \equiv q \equiv 3 \mod 4$ 是素数。

分解的难解性假设（IAF）：令 $k > 0$ 为任意固定数。对于任何多项式时间的概率整数分解算法，当输入从 W_{n} 中随机选取的整数时，对所有足够大的 n，其成功概率小于 $1 / n^{k}$。

2. 术语

令 $S = (S_{1}, S_{2}, \ldots)$ 和 $S' = (S_{1}', S_{2}', \ldots)$ 为多项式合奏，其中 $S_{n} = (X_{n}, Y_{n})$ 和 $S_{n}' = (X_{n}', Y_{n}')$ 各是 $\{0, 1\}^{*} \times \{0, 1\}^{*}$ 上的概率分布。进一步假设两个合奏 $(X_{1}, X_{2}, \ldots)$ 和 $(X_{1}', X_{2}', \ldots)$ 对于多项式时间计算是不可区分的。

令 $\mathcal{D} := (d_1, d_2, \ldots)$ 为一列谓词 $d_n: \{0, 1\}^* \to \{0, 1\}$，其中给定 d 和 n，$d_n(x)$ 可以在 n 的多项式时间内概率地计算。定义**猜测算法** Q_B 为一个概率算法，它以 (n, y) 作为输入，其中 $y \in \{0, 1\}^*$，并在 n 的多项式时间内输出 0 或 1。

考虑随机选取 (x_{n},y_{n}) 服从分布 (X_{n},Y_{n}) 的实验。假设观察者看到值 y_{n} 并尝试使用猜测算法 Q 来猜测 $d_{n}(x_{n})$ 的值；让我们用 $r(d_n,X_n,Y_n,Q)$ 表示做出正确猜测的概率。我们使用符号 o(poly-small) 表示任何序列 $(b_{1},b_{2},\ldots)$ 具有性质：对于所有固定的 k，有 $b_{n} = o(1 / n^{k})$。

定义 我们记 $I_{n}(X_{n} \mid Y_{n}) \preceq I_{n}(X_{n}^{\prime} \mid Y_{n}^{\prime})$，如果 $\forall D$ 和 $Q \ni Q^{\prime}$ 使得 $r(d_{n}, X_{n}^{\prime}, Y_{n}^{\prime}, Q^{\prime}) - r(d_{n}, X_{n}, Y_{n}, Q) \geq o(\text{poly - small})$。

定义 我们记 $I_{n}(X_{n} \mid Y_{n}) \approx I_{n}(X_{n}^{\prime} \mid Y_{n}^{\prime})$，如果 $I_{n}(X_{n} \mid Y_{n}) \preceq I_{n}(X_{n}^{\prime} \mid Y_{n}^{\prime})$ 且 $I_{n}(X_{n}^{\prime} \mid Y_{n}^{\prime}) \preceq I_{n}(X_{n} \mid Y_{n})$。

一个**谜题合奏** $\mathcal{P} = (L, \mathcal{F})$ 包含一个语言 $L \in \mathrm{BPP}$ 和一个多项式时间合奏 $\mathcal{F} = (F_1, F_2, \ldots)$，其中每个 F_n 是 $\{0, 1\}^* \times \{0, 1\}^*$ 上的分布；我们进一步要求，随机选取的服从 F_n 分布的 (s, τ) 将以概率 1-o(poly-small) 满足 $(n, s, \tau) \in L$；我们将称 s 为文本 τ 的**秘密**。令 $T_{\mathcal{P}, n}$ 表示从随机 (s, τ)（服从 F_n 分布）的第二个分量中随机取出的 τ 的分布。我们称 $\mathcal{P}$ 为**难解的**，如果对于每个概率多项式时间算法 S，当输入一对 (s, τ)，其中 τ 服从 $T_{\mathcal{P}, n}$ 分布时，算法将以概率 1-o(poly-small) 无法产生一个满足 $(n, s, \tau) \in L$ 的 s。

例如，令 $L = \{(n; p, q, N) \mid N = p \cdot q, p, q \text{ are } n$ 比特素数$\}$，且 F_{n} 为集合 W_{n} 上的均匀分布。在分解难解性假设下，**分解谜题合奏** $\mathcal{P} = (L, \mathcal{F})$ 是一个难解谜题合奏。

3. 生成一个秘密

令 $\mathcal{P} = (L, \mathcal{F})$，其中 $\mathcal{F} = (F_1, F_2, \ldots, F_n, \ldots)$，为一个难解谜题合奏。我们希望设计一个协议 $\mathcal{M} = (M_A, M_B)$，对于任意给定的 n，具有以下属性：1) $\mathcal{M}$ 隐式地生成一对服从 F_n 分布的 (s, τ)，2) 文本 τ 将被 M_A 和 M_B 作为其输出发现，以及 3) 秘密 s，虽然在协议执行结束时可以基于双方拥有的信息由 A 和 B 共同计算，但完全对每一方自身隐藏，即使其中一方在执行协议期间作弊。

为了简化结果的表述，我们限制自己于具有唯一秘密的谜题。让我们称一个谜题合奏 $\mathcal{P} = (L,\mathcal{F})$ 为**唯一可译的**，如果对于每个 n,τ，至多存在一个 s 满足 $(n,s,\tau)\in L$。例如，分解谜题合奏是唯一可译的。对于一个唯一可译的难解谜题合奏，我们可以将其写为 $\mathcal{P} = (\alpha ,\mathcal{D})$，其中 $\alpha =

(a_{1},a_{2},\ldots)$ 和 $\mathcal{D} = (D_1,D_2,\ldots)$ 由 $a_{n}(\tau) = s$ 和 $D_{n} = T_{p,n}$ 给出。

令 $\mathcal{P} = (\alpha, \mathcal{D})$ 为一个唯一可译的难解谜题合奏，其中 $\alpha = (a_1, a_2, \ldots)$ 和 $\mathcal{D} = (D_1, D_2, \ldots)$。正式地，我们为 $\mathcal{P}$ 生成秘密的协议 $\mathcal{M} = (M_A, M_B)$ 定义我们的要求如下。（A 或 B 有时会输出 u_A 或 $u_B = \text{CHEATING}$；非正式地说，我们说 A 或 B 检测到另一方作弊。）

有效性

如果 A 和 B 都遵循协议，那么

(i) 以概率 $1 - o(\text{poly - small})$，有 $u_A = u_B$，且它们的公共值 τ 服从一个分布，该分布在多项式时间计算下与 D_n 不可区分；

(ii) 对 $j \in \{A, B\}$，有 $I_n^{(J)}(a_n(\tau) \mid \tau, \Delta_j) \approx I_n^{(\mathcal{L})}(a_n(\tau) \mid \tau)$，其中 J 是由协议 $\mathcal{M} = (M_A, M_B)$ 的执行所诱导的随机过程，而 $\mathcal{L}$ 是根据 D_n 获取 τ 的随机过程。

(iii) 存在一个协议 $\mathcal{N} = (N_A, N_B)$，给定输入 (Δ_A, Δ_B)，计算输出 v_A, v_B，并具有性质：以概率 1-o(poly-small) 有 $v_A = v_B = a_n(\tau)$。

为了讨论当一方（例如 B）可能行为不端时的有效性概念。令 d_{n} 为运行中 $v_{A} \neq \text{CHEATING}$ 的概率，并令 D_{n}' 为 τ 在限制于此类运行时的概率分布。如果 d_{n} 可忽略，则 A 几乎总能捕获 B 作弊，无需进一步要求。另一方面，如果对于某个固定的 $t > 0$，有 $d_{n} = \Omega(1/n^{t})$，那么我们需要以下两个约束为真：

(iv) $\mathcal{P}' = (\alpha, \mathcal{D}')$ 是一个唯一可译的难解谜题合奏，其中 $\mathcal{D}' = (D_1', D_2', \ldots)$。

(v) $I_{n}^{(J)}(a_{n}(\tau) \mid \tau, \Delta_{B}) \approx I_{n}^{(\mathcal{L})}(a_{n}(\tau) \mid \tau)$，其中 J 是由协议 $\mathcal{M} = (M_A, M_B')$ 的执行所诱导的随机过程，而 $\mathcal{L}$ 是根据 D_{n}' 获取 τ 的随机过程。

为发展公平性概念，考虑执行协议 $\mathcal{M}$ 然后 $\mathcal{N}$。如果 A 遵循协议，我们要求 B 获得 s 而 A 无法恢复 s 的概率很小。

公平性

(i) [B 可能作弊。] 假设协议 $\mathcal{M} = (M_A, M_B)$ 和 $\mathcal{N} = (N_A, N_B)$ 与机器对 (M_A, M_B')、(N_A, N_B') 一起运行。存在一个多项式时间的概率算法 Y（依赖于 M_B' 和 N_B'），它将 $\mathcal{M}$ 和 $\mathcal{N}$ 的历史对作为输入，并输出一个字符串 w。我们要求：如果 $\mathbf{A}$ 遵循协议 $\mathcal{M}$ 和 $\mathcal{N}$ 然后运行 Y，则 $u_A = \tau$、$v_B = a_n(\tau)$ 同时 $w \neq a_n(\tau)$ 的概率是 o(poly-small)。

(ii) [A 可能作弊。]（将 (i) 中 A 和 B 的角色互换。）

定理 1. 令 $\mathcal{P}$ 是一个唯一可译的难解谜题合奏。存在一个从 $\mathcal{P}$ 生成秘密的协议 $\mathcal{M} = (M_A, M_B)$，该协议实现有效性和公平性。

4. 秘密交换

令 $\mathcal{P}_A = (L_A, \mathcal{F}_A)$ 和 $\mathcal{P}_B = (L_B, \mathcal{F}_B)$ 为两个难解谜题合奏。令 $(s_A, \tau_A), (s_B, \tau_B)$ 为根据 $F_{A,n}$ 和 $F_{B,n}$ 分布的随机谜题；将 (n, s_A, τ_A, τ_B) 作为输入给 A，(n, s_B, τ_A, τ_B) 给 B。我们希望设计一个协议 $\mathcal{M} = (M_A, M_B)$，使 A 和 B 能够交换他们的秘密 s_A 和 s_B，且双方都不会被欺骗。我们在下面陈述协议的准则。

有效性

如果双方都遵循协议，则以概率 $1 - o(\text{poly-small})$，有 $u_{B} = s_{A}$ 且 $u_{A} = s_{B}$。

公平性

(i) [如果 B 得到 s_A，那么 A 可以计算 s_B。]
如果 A 遵循协议而 B 没有（即 $M_B' \neq M_B$），那么对于任意固定的 k，存在一个概率多项式时间算法 S，它以 (n, Δ_A) 作为输入并输出一个数字 v，使得以下为真：

(ii) [如果 A 得到 s_B，那么 B 可以计算 s_A。]
（将 (i) 中的 A 和 B 互换。）

定理 2. 令 $\mathcal{P}_A$ 和 $\mathcal{P}_B$ 是任意两个难解合奏。存在一个在 $\mathcal{P}_A$ 和 $\mathcal{P}_B$ 之间交换秘密的协议 $\mathcal{M} = (M_A, M_B)$，该协议实现有效性和公平性。

5. 通用计算

考虑第 1 节中定义的交互式计算问题 $\langle h_n, w_n \rangle$。我们将展示存在一个协议来解决该问题，并满足一些强的隐私性和公平性约束。在这个扩展摘要中，我们限制自己于 w_n 表示一对待评估函数 (f_n, g_n) 的情况，即给定输入 i_A, i_B，A 和 B 希望计算 $f_n(i_A, i_B)$ 和 $g_n(i_A, i_B)$。这些结果可以推广到一般情况。

我们将考虑两个变体，它们因输入-输出格式而异。模型 I 是自然模型，其中输入和输出如前一段所述。然而，由于通常 A 无法控制 i_B 的值，不诚实的 B 可以假装 i_B 是任意值 y。结果，公平性约束至多只能强制 B 为某个 y 计算 $f_n(i_A, y)$ 的值。在模型 II 中，A 将获得输入 $(i_A, p_A, q_A, N_A, N_B, E_{N_B}(i_B))$，其中 $N_A = p_A \cdot q_A$ 是两个大素数的乘积，N_B 是另一个由两个大素因子组成的整数，用于将 B 的参数 i_B 编码为 $E_{N_B}(i_B)$；B 拥有输入 $(i_B, p_B, q_B, N_B, N_A, E_{N_A}(i_A))$。整数 N_A 和 N_B 是从某些分布生成的，使得分解这些数在计算上是难解的；E_N 可以是任何在分解难解性假设下被证明安全的概率加密方案（例如，Alexi, Chor, Goldreich and Schnorr [ACGS] 或 Blum and Goldwasser [BS] 中的方案）。在输出中，A 和 B 分别获得 (u_A, v_A, w_A) 和 (u_B, v_B, w_B)。当双方都诚实行为时，$u_A = f_n(i_A, i_B)$，$v_A = (p'_A, q'_A, N'_A)$，$w_A = (N'_B, E_{N'_B}(g_n(i_A, i_B)))$，其中 N'_A 和 N'_B 是两个大素因子的乘积；类似地对于 B 的输出。这种格式在协议级联中很自然地出现。此模型中的隐私性和公平性约束比第一个模型更强。

模型 I

我们首先定义当双方都遵循协议时的约束。

有效性

以概率 1-o(poly-small)，有 $u_{A} = f_{n}(i_{A}, i_{B})$ 和 $u_{B} = g_{n}(i_{A}, i_{B})$。

隐私性

(i) $I_n^{(J)}(i_A,i_B,u_A\mid \Delta_B,i_B,u_B)\approx I_n^{(\mathcal{L})}(i_A,i_B,u_A\mid i_B,u_B)$，其中 J 是使用来自 h 的输入运行 $\mathcal{M}$ 的随机过程，而 $\mathcal{L}$ 是使得 (i_A,i_B) 根据 h 分布且 $u_{A} = f_{n}(i_{A},i_{B})$、$u_{B} = g_{n}(i_{A},i_{B})$ 的随机过程。
(ii)（将 (i) 中的 A 和 B 互换。）

我们现在定义当 A 遵循协议而 B 可能作弊（即 B 使用任意 M_B' 执行协议）时的约束。由于 M_B' 是任意的，B 总是可以生成一个 i_B' 并表现得好像这是输入值 i_B。因为 A 永远无法检测到这种作弊方式，A 必须合作并让 B 知道 $g_{n}(i_{A}, i_{B}^{\prime})$ 的值。这使得 B 能够控制 A 的输出 u_{A}，并在某种程度上探测 i_{A} 的值；如果我们想要保持上述针对诚实方的有效性条件，这无法阻止。我们将要求 B 不能做更多的事。让我们用 Z 表示集合 $\{0,1\}^{*} \cup \{\text{CHEATING}\}$，并通过 $f_{n}(i_{A}, y) = \text{CHEATING}$ 如果 $y = \text{CHEATING}$ 来扩展函数 f_{n}。

令 M_B' 为任意通信图灵机，并令 U_n 为当使用 (M_A, M_B') 执行协议时对应 u_A 的分布。

有效性

存在一个概率多项式时间算法 S，它以 (n, i_B) 作为输入并产生一个随机的 $y \in Z$，使得 U_n 与对应于 $f_n(i_A, y)$ 的分布不可区分。

令 J 为使用根据 h_n 分布的输入 (i_A, i_B) 运行 (M_A, M_B') 的随机过程。令 S 为所有概率多项式时间算法 S 的集合，这些算法以 (n, i_B) 作为输入并产生随机的 $y \in Z$。对于任何 $S \in S$，令 $\mathcal{L}(S)$ 为根据 h_n 生成 (i_A, i_B) 的随机过程，在输入 (n, i_B) 上运行 S 以产生随机的 y，并定义 $u_A = f_n(i_A, y)$，$u_B = g_n(i_A, y)$。直观地说，B 可以使用 S 来生成一个随机的 y，之后，表现得好像 y 是 i_B 的值，并遵循协议；$\mathcal{L}(S)$ 显然是 B 信息量较少的过程，其中 A 仅告诉 B 值 $u_B = g_n(i_A, y)$，而没有其他通信发生。

隐私性

对于任何 M_B'，存在一个 $S \in S$ 使得 $I_n^{(J)}(i_A, i_B, u_A \mid \Delta_B, i_B, u_B) \preceq I_n^{(\mathcal{L}(S))}(i_A, i_B, u_A \mid i_B, u_B)$。

公平性概念涉及获得应得的信息。假设 B 希望知道 $g_{n}(i_{A},y)$ 对于某个不同于 i_{B} 的 y 的值。正如我们之前提到的，B 可以通过假装 $i_{B} = y$ 并遵循协议来成功；在这个过程中，A 也将获得 $f_{n}(i_{A},y)$ 的值。隐私性约束规定，除了 $g_{n}(i_{A},y)$ 的值之外，没有其他信息从 A 传递给 B。公平性约束关注的是 B 是否可能在某一点停止，一旦 B 拥有关于 $g_{n}(i_{A},y)$ 的信息，并拒绝让 A 知道 $f_{n}(i_{A},y)$。以下表述不是此约束的最强可能版本，但对于某些应用（例如交换足够长的秘密）是足够的。令 L_{n} 表示当 (i_{A},i_{B}) 根据 h_{n} 分布时，$f_{n}(i_{A},i_{B})$ 的所有可能值的集合。

定义. 对于 $\mathbf{A}$ 的一个**恢复算法** R 是一个概率多项式时间算法，它以 (n, Δ_A) 作为输入，并输出一个字符串 v。

考虑一对根据 h_n 分布的 (i_A, i_B)。令 G 表示一个概率多项式时间图灵机，它以 i_B 作为输入并输出 z；令 $\beta_n(G)$ 表示 $z = g_n(i_A, i_B)$ 的概率。

公平性

对于任何固定的 k，存在一个恢复算法 R（依赖于 M_B'），其输出 v 满足以下条件：

对于某个 G。

为了完成定义，我们在上述讨论中互换 A 和 B 的角色，从而得到当 B 遵循协议而 A 作弊时，有效性、隐私性和公平性的约束。

定义. 一个协议被称为**实现有效性、隐私性和公平性**，如果满足所有上述约束。

定理 3. 对于任何交互式计算问题 $\langle h_n,(f_n,g_n)\rangle$，存在一个协议 $\mathcal{M}$ 实现有效性、隐私性和公平性。

在模型 II 中，约束更简单。输入-输出格式如本节开头所述。

模型 II

当 A 和 B 都遵循协议时，要求如下。

有效性

以概率 1-o(poly-small)，有 $u_A = f_n(i_A, i_B)$，$v_A = (p'_A, q'_A, N'_A)$ 且 $w_A = E_{N'_B}(g_n(i_A, i_B))$，其中 N'_A 是 n 比特素数 p'_A 和 q'_A 的乘积；N'_A 的分布使得分解它是难解的。对 B 也有双重约束要求。

隐私性

(i) $I_n^{(J)}(i_A,i_B,u_A\mid \Delta_B,i_B,u_B,v_B,w_B)\approx I_n^{(\mathcal{L})}(i_A,i_B,u_A\mid i_B,u_B)$，其中 $\pmb{J}$ 是使用来自 h 的输入运行 $\mathcal{M}$ 的随机过程，而 $\mathcal{L}$ 是使得 (i_A,i_B) 根据 h 分布且 $u_{A} = f_{n}(i_{A},i_{B})$、$u_{B} = g_{n}(i_{A},i_{B})$ 的随机过程。
(ii)（将 (i) 中的 A 和 B 互换。）

如果 A 遵循协议，但 B 可能作弊并执行某个 M_B^t。令 d_{n} 为运行中 $v_{A}\neq \mathrm{CHEATING}$ 的概率，并令 U_{n,i_B} 为 u_{B} 在限制于此运行且 i_B 作为 B 的输入时的概率分布。如果 d_{n} 可忽略，则 A 几乎总能捕获 B 作弊，无需进一步要求。另一方面，如果对于某个固定的 $t > 0$，有 $d_{n} = \Omega (1 / n^{t})$，那么我们需要以下有效性和隐私性约束。

有效性

以概率 $d_n - o(\text{poly - small})$，有 $u_A = f_n(i_A, i_B)$，$v_A = (p'_A, q'_A, N'_A)$ 且 $w'_A = E_{N'_B}(g_n(i_A, i_B))$，其中 N'_A 是 n 比特素数 p'_A 和 q'_A 的乘积；N'_A 的分布使得分解它是难解的。

隐私性

$I_{n}^{(J)}(i_{A},i_{B},u_{A}\mid \Delta_{B},i_{B},u_{B},v_{B},w_{B})\approx I_{n}^{(\mathcal{L})}(i_{A},i_{B},u_{A}\mid i_B,u_B)$，其中 J 是由协议 $\mathcal{M} = (M_A,M_B')$ 的执行所诱导的随机过程，而 $\mathcal{L}$ 是给定根据 U_{n,i_B} 分布的 u_{B} 值的随机过程。

如同在模型 I 中，令 G 表示一个概率多项式时间图灵机，它以 i_B 作为输入并输出 z；令 $\beta_n(G)$ 表示 $z = g_n(i_A, i_B)$ 的概率。

公平性

对于任何固定的 k，存在一个恢复算法 R（依赖于 $M_B^{\prime}$），其输出 s 满足以下条件：

对于某个 G。

当 B 遵循协议而 A 可能作弊时，我们有一组通过交换上述要求中 A 和 B 的角色而得到的要求。

定理 3 对于模型 II 也成立。

6. 联系

我们选择了隐私性的语义定义，即通信不会使得一方能够更准确地计算任何多项式时间谓词。在最近的文献中，[GMR]（并在 [GHV] 中推广）引入了一个优雅的最小知识转移协议概念来捕捉无意外信息泄露的概念。我们可以基于此概念为我们的问题定义隐私性约束。我们可以证明用于证明这些定理的协议也满足最小知识转移要求。

定理 4. 对于任何交互式计算问题 $\langle h_n, (f_n, g_n) \rangle$，存在一个实现有效性和公平性的最小知识转移协议 $\mathcal{M}$。

这些定理的证明很长，将在完整论文中给出。

参考文献

[ACGS] W. Alexi, B. Chor, O. Goldreich, and C. P. Schnorr, "RSA/Rabin bits are $1/2 + 1/2$ poly(log n) secure," Proceedings of 25th Annual IEEE Symposium on Foundations of Computer Science, 1984, 449-457.

[B1] M. Blum, "Coin flipping by phone," COMPCON (1982), 133-137.

[B2] M. Blum, "How to exchange (secret) keys," ACM Transactions on Computer Systems 1(1983), 175-193.

[BS] M. Blum and S. Goldwasser, "An efficient probabilistic PKCS as secure as factoring," Proceedings of Crypto 84, 1984.

[C] R. Cleve, "Limits on the security of coin flips when half of the processors are faulty," Proceedings of 18th Annual ACM Symposium on Theory of Computing, 1986, 364-369.

[FMRW] M. Fischer, S. Micali, C. Rackoff, and D. Wittenberg, "An oblivious transfer protocol," 1985, to appear.

[GHY] Z. Galil, S. Haber, and M. Yung, "A private interactive test of a Boolean predicate and minimum-knowledge public-key cryptosystems," Proceedings of 26th Annual IEEE Symposium on Foundations of Computer Science, 1985, 360-371.

[GM] S. Goldwasser and S. Micali, "Probabilistic encryption and how to play mental poker keeping secret all partial information," Proceedings of 14th Annual ACM Symposium on Theory of Computing, 1982, 365-377.

[GMR] S. Goldwasser, S. Micali, and C. Rackoff, "The knowledge complexity of interactive proof systems," Proceedings of 17th Annual ACM Symposium on Theory of Computing, 1985, 291-304.

[GMW] O. Goldreich, S. Micali, and A. Wigderson, "Proofs that yield nothing but their validity and a methodology of cryptographic protocol design," Proceedings of 27th Annual IEEE Symposium on Foundations of Computer Science, 1986.

[HS] J. Hastad and A. Shamir, "The cryptographic security of truncated linearly related variables," Proceedings of 17th Annual ACM Symposium on Theory of Computing, 1985, 356-362.

[LMR] M. Luby, S. Micali, and C. Rackoff, "How to simultaneously exchange a secret bit by flipping a symmetrically-based coin," Proceedings of 24th Annual IEEE Symposium on Foundations of Computer Science, 1985, 11-22.

[R] M. Rabin, "How to exchange secrets," 1981, unpublished manuscript.

[SRA] A. Shamir, R. Rivest, and L. Adleman, "Mental Poker," MIT Technical Report, 1978.

[T] T. Tedrick, "How to exchange half a bit," Crypto '88.

[VV] U. Vazirani and V. Vazirani, "Trapdoor pseudo-random number generators, with applications to protocol design," Proceedings of 24th Annual IEEE Symposium on Foundations of Computer Science, 1985, 23-30.

[Y1] A. Yao, "Protocols for secure computations," (extended abstract) Proceedings of 21st Annual IEEE Symposium on Foundations of Computer Science, 1982.

[Y2] A. Yao, "Protocols for secure computations," in preparation.

[Y3] A. Yao, "Theory and applications of trapdoor functions," Proceedings of 21st Annual IEEE Symposium on Foundations of Computer Science, 1982.