基于 OT 的乘法三元组生成

关诚

2021 年 3 月 31 日

1 ABY[DSZ15] 中乘法元组生成算法

ℓ 比特串的 2 选 1 不经意传输记为 OT_ℓ，ℓ 次 OT_ℓ 的调用记为 OT_ℓ^ℓ. 如下的协议 C-OT_ℓ^ℓ 实现 $x+y=a\cdot b$，这里 a 和 x 分别是 P_0 的输入和输出，b 和 y 分别是 P_1 的输入和输出。

C-OT_ℓ^ℓ

Input：Bob 的输入是 b，Alice 的输入是 a.
Output：Bob 的输出是 y，Alice 的输出是 x.

1. Bob 选择 ℓ 个独立的随机值 $s_0, s_1, ..., s_{\ell-1}$，令 $t_i^0 = s_i, t_i^1 = 2^i b + s_i$，得到 ℓ 对元组 $(t_0^0, t_0^1), ..., (t_{\ell-1}^0, t_{\ell-1}^1)$.
2. Alice 和 Bob 并行执行 ℓ 次 OT_ℓ. 在第 i 次 OT_ℓ 中，Alice 以 $a[i]$(a 的第 i 个比特) 作为选择比特，Bob 以 (t_i^0, t_i^1) 作为输入元组，Alice 得到 $t_i^{a[i]}$.
3. Alice 计算 $x = \sum_{i=0}^{\ell-1} t_i^{a[i]}$, Bob 计算 $y = -\sum_{i=0}^{\ell-1} s_i$.

正确性：$x + y = a \cdot b$.

证：因为 $a=\sum_{i=0}^{\ell-1}a[i]\cdot 2^i$，且 $t_i^{a[i]}=a[i]\cdot 2^i b+s_i$ 则

$$\begin{aligned}
x+y &= \sum_{i=0}^{\ell-1} t_i^{a[i]} - \sum_{i=0}^{\ell-1} s_i \\
&= \sum_{i=0}^{\ell-1}(a[i]\cdot 2^i b+s_i) - \sum_{i=0}^{\ell-1} s_i \\
&= b\sum_{i=0}^{\ell-1}(a[i]\cdot 2^i) \\
&= a\cdot b
\end{aligned}$$

乘法元组生成

观察到：$(a_0+a_1)(b_0+b_1)=a_0b_0+a_0b_1+a_1b_0+a_1b_1$，如果 P_0 本地生成 a_0,b_0，P_1 本地生成 a_1,b_1，那么 a_0b_0 和 a_1b_1 可以分别在 P_0 和 P_1 本地计算。对于交叉项 a_0b_1，将其共享为 $x_0+y_0=a_0b_1$，对于交叉项 a_1b_0，将其共享为 $x_1+y_1=a_1b_0$，那么

$$a_0b_0+a_0b_1+a_1b_0+a_1b_1 = a_0b_0+x_0+y_0+x_1+y_1+a_1b_1 \tag{1}$$
$$= (a_0b_0+x_0+y_1)+(a_1b_1+x_1+y_0) \tag{2}$$
$$= c_0+c_1 \tag{3}$$

现在，P_0 只要拿到 x_0,y_1 就可以计算出 $c_0=a_0b_0+x_0+y_1$，P_1 只要拿到 x_1,y_0 就可以计算出 $c_1=a_1b_1+x_1+y_0$. 利用如下两个算法，P_0 和 P_1 可得到需要的值

算法-1

Input：P_0 输入 a_0，P_1 输入 b_1
Output：P_0 得到 x_0，P_1 得到 y_0

1. P_0 和 P_1 调用 C-OT$_\ell^\ell$，其中 P_0 充当 Alice 得到 x_0，P_1 充当 Bob 得到 y_0.
2. 输出 x_0,y_0.

算法-2

Input：P_0 输入 b_0，P_1 输入 a_1

Output：P_0 得到 y_1，P_1 得到 x_1

1. P_0 和 P_1 调用 C-OT_ℓ^ℓ，其中 P_0 充当 Bob 得到 y_1，P_1 充当 Alice 得到 x_1.
2. 输出 x_1, y_1.

2 [FKOS15] 中乘法元组生成算法

Protocol-$\Pi_{\mathrm{BitTriples}}$

协议的目的是生成 ℓ 个 $\mathbb{F}_2$ 上的元组 $\langle x_h\rangle, \langle y_h\rangle, \langle z_h\rangle, h = 1, ..., \ell$ 使得 $z_h = x_h \cdot y_h$，假设可以访问一个随机 oracle(哈希函数)$H : \{0,1\}^* \to \{0,1\}$.

ate **Initialize**：每个参与者 P_i 采样一个随机 MAC 密钥共享 Δ^i 以及一个值 $\widetilde{\Delta}^i \in \mathbb{F}_2^k$，置 $\hat{\Delta}^i = (\widetilde{\Delta}^i || \Delta^i) \in \mathbb{F}_2^{2k}$.

COTe.Extend-[IKNP03]：

1. P_1 以输入 $x^1 = (x_1^1, x_2^1, ..., x_\ell^1) \in \mathbb{F}_2^\ell$ 充当接收方和 P_0 充当发送方运行 $\mathrm{COTe}^{2k,\ell}$，P_1 接收 $\{\hat{t}_h^{(1,0)}\}_{h\in[\ell]}$，$P_0$ 接收 $\hat{q}_h^{(0,1)} = \hat{t}_h^{(1,0)} + x_h^1 \cdot \hat{\Delta}^1, h \in [\ell]$. 其中 $\hat{t}_h^{(1,0)} = (\widetilde{t}_h^{(1,0)} || t_h^{(1,0)}) \in \mathbb{F}_2^{2k}$，$\hat{q}_h^{(0,1)} = (\widetilde{q}_h^{(0,1)} || q_h^{(0,1)}) \in \mathbb{F}_2^{2k}$.
2. P_0 以输入 $x^0 = (x_1^0, x_2^0, ..., x_\ell^0) \in \mathbb{F}_2^\ell$ 充当接收方和 P_1 充当发送方运行 $\mathrm{COTe}^{2k,\ell}$，P_0 接收 $\{\hat{t}_h^{(0,1)}\}_{h\in[\ell]}$，$P_1$ 接收 $\hat{q}_h^{(1,0)} = \hat{t}_h^{(0,1)} + x_h^0 \cdot \hat{\Delta}^1, h \in [\ell]$. 其中 $\hat{t}_h^{(0,1)} = (\widetilde{t}_h^{(0,1)} || t_h^{(0,1)}) \in \mathbb{F}_2^{2k}$，$\hat{q}_h^{(1,0)} = (\widetilde{q}_h^{(1,0)} || q_h^{(1,0)}) \in \mathbb{F}_2^{2k}$.

Triple generation：

1. 每个参与者 P_i 生成 ℓ 个随机比特 $y_h^i \in \mathbb{F}_2$.

2. P_0：(P_1 也执行此步骤，所有角色互换)

   (a) 使用哈希函数 $H:\{0,1\}^* \to \{0,1\}$，打乱 **COTe.Extend** 阶段的数据关系。P_0 本地计算 $H(\widetilde{t}_h^{(0,1)}) = w_h^{(0,1)}$，$P_1$ 本地计算 $H(\widetilde{q}_h^{(1,0)}) = v_{0,h}^{(1,0)}, H(\widetilde{q}_h^{(1,0)} + \widetilde{\Delta}^1) = v_{1,h}^{(1,0)}, \forall h \in [\ell]$.

   (b) 参与者创造新的关系：

      - P_1 发送一个向量 $s^{(1,0)} \in \mathbb{F}_2^\ell$ 给 P_0，满足 $s_h^{(1,0)} = v_{0,h}^{(1,0)} + v_{1,h}^{(1,0)} + y_h^1$.
      - P_0 计算 $n_h^{(0,1)} = w_h^{(0,1)} + x_h^0 \cdot s^{(1,0)} = v_{0,h}^{(1,0)} + x_h^0 \cdot y_h^1$.

3. P_0 计算 $z_h^0 = n_h^{(0,1)} + x_h^0 \cdot y_h^0 + v_{0,h}^{(0,1)}$.
   P_1 计算 $z_h^1 = n_h^{(1,0)} + x_h^1 \cdot y_h^1 + v_{0,h}^{(1,0)}$

2.1 IKNP-OTs[IKNP03]

3 [FLNW17] 中乘法元组生成算法

参考文献

[DSZ15] Daniel Demmler, Thomas Schneider, and Michael Zohner. ABY - A framework for efficient mixed-protocol secure two-party computation. In *22nd Annual Network and Distributed System Security Symposium, NDSS 2015, San Diego, California, USA, February 8-11, 2015*. The Internet Society, 2015.

[FKOS15] Tore Kasper Frederiksen, Marcel Keller, Emmanuela Orsini, and Peter Scholl. A unified approach to MPC with preprocessing using OT. In Tetsu Iwata and Jung Hee Cheon, editors, *Advances in Cryptology - ASIACRYPT 2015 - 21st International Conference on the Theory and Application of Cryptology and Information Security, Auckland, New Zealand, November 29 - December 3, 2015, Proceedings, Part I*, volume 9452 of *Lecture Notes in Computer Science*, pages 711–735. Springer, 2015.

[FLNW17] Jun Furukawa, Yehuda Lindell, Ariel Nof, and Or Weinstein. High-throughput secure three-party computation for malicious adversaries and an honest majority. In Jean-Sébastien Coron and Jesper Buus Nielsen, editors, *Advances in Cryptology - EUROCRYPT 2017 - 36th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Paris, France, April 30 - May 4, 2017, Proceedings, Part II*, volume 10211 of *Lecture Notes in Computer Science*, pages 225–255, 2017.

[IKNP03] Yuval Ishai, Joe Kilian, Kobbi Nissim, and Erez Petrank. Extending oblivious transfers efficiently. In Dan Boneh, editor, *Advances in Cryptology - CRYPTO 2003, 23rd Annual International Cryptology Conference, Santa Barbara, California, USA, August 17-21, 2003, Proceedings*, volume 2729 of *Lecture Notes in Computer Science*, pages 145–161. Springer, 2003.