Correlated Pseudorandomness from Expand-Accumulate Codes

Crypto 2022 · `https://ia.cr/2022/1014`

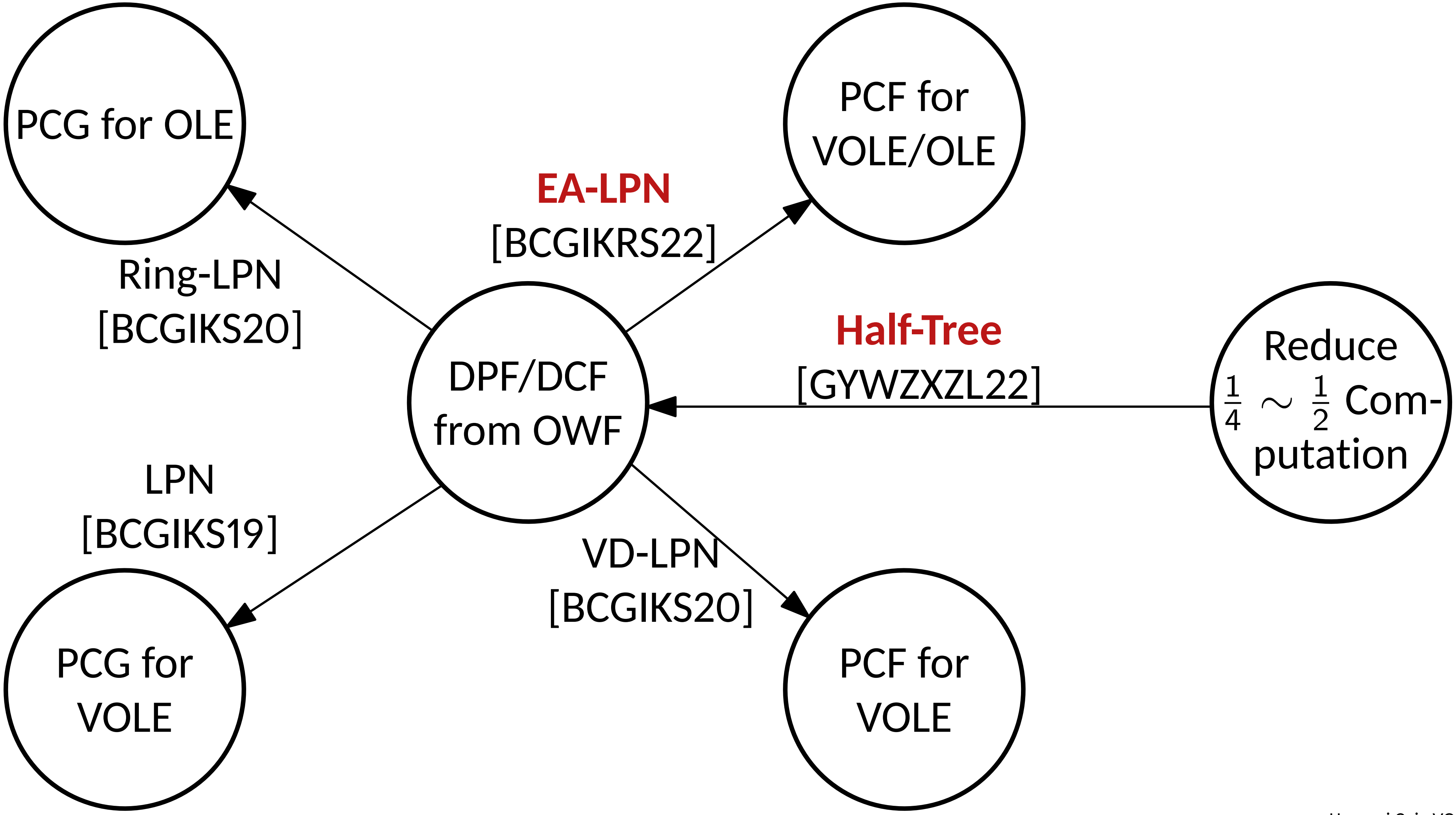
Elette Boyle · Geoffroy Couteau · Niv Gilboa ·
Yuval Ishai · Lisa Kohl · Nicolas Resch · Peter Scholl

January 17, 2023· Presented by Hongrui Cui

* Some acknowledgments?

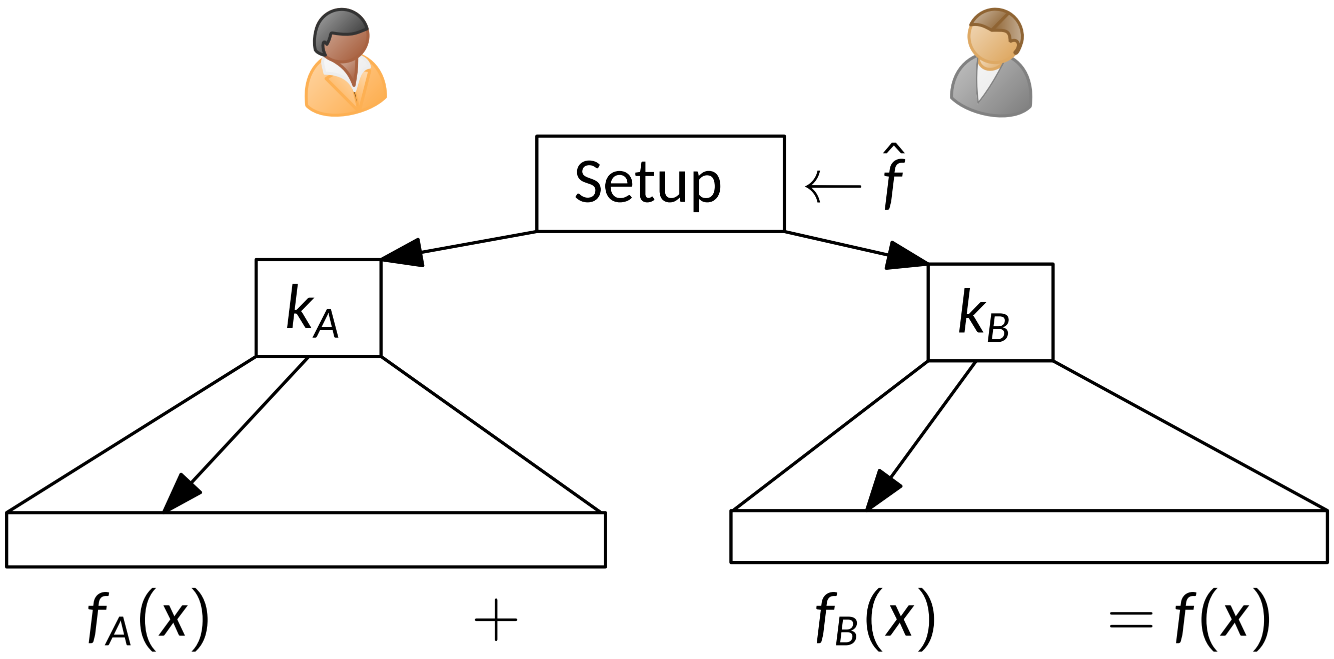
Introduction

- PCG/PCF paradigm = FSS + LPN
- The main contribution is a new LPN variant and FSS optimization



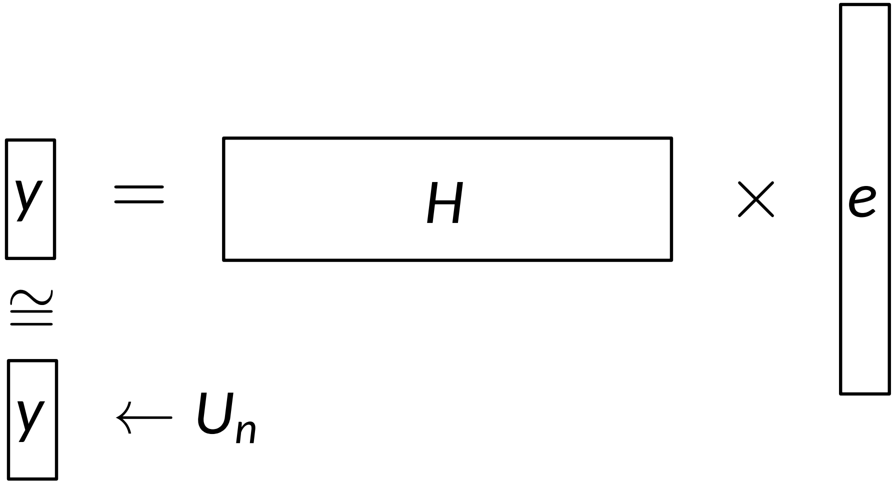
Preliminaries on PCG/PCF

Function Secret Sharing



- Succinctness: $|k_A|, |k_B| \ll 2^{|x|}$
- Efficient FSS exists for point/comparison functions

dual-LPN



- View *e* as seed, *H* is a linear PRG
- PCG idea: generate sparse correlations as seed and expand them using dual-LPN

Example: PCG for VOLE

- KeyGen:

Step 1: $e \leftarrow \chi^N$ $\quad$ $e = \begin{array}{|cccc|}\hline \beta_1 & \beta_2 & \cdots & \beta_\ell \\ \hline \end{array}$

$$\begin{array}{cccc} \alpha_1 & \alpha_2 & \cdots & \alpha_\ell \end{array}$$

Step 2: $(k_0^1, k_1^1) \leftarrow FSS.KeyGen(\alpha_1, \beta_1 \cdot \Delta)$

$\cdots$

$(k_0^\ell, k_1^\ell) \leftarrow FSS.KeyGen(\alpha_\ell, \beta_\ell \cdot \Delta)$

$key_0 := \{k_0^1, ..., k_0^\ell\}, \Delta$ $\qquad$ $key_1 := \{k_1^1, ..., k_1^\ell\}, e$

- Expand:

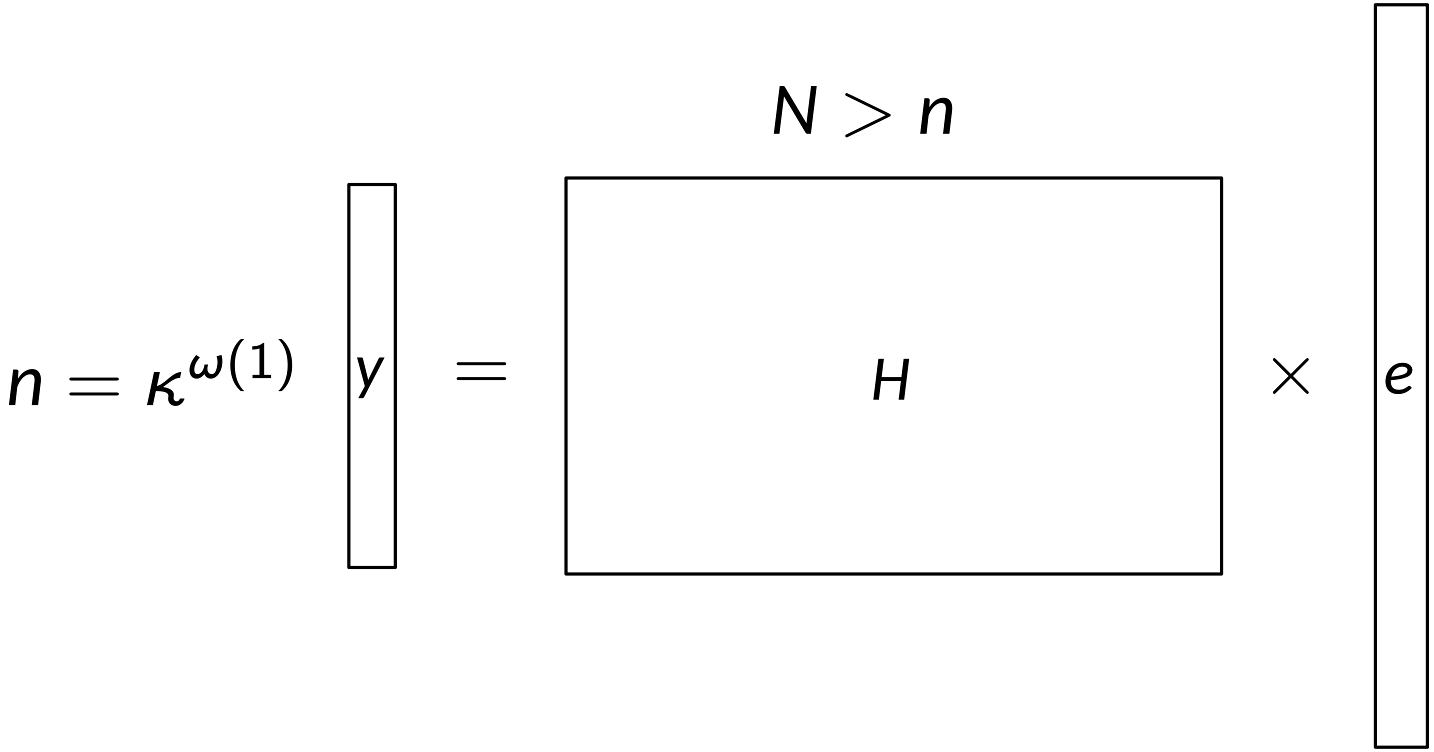
$w := H \cdot (FullEval(k_0^1) + ... + FullEval(k_0^\ell))$

$v := H \cdot (FullEval(k_1^1) + ... + FullEval(k_1^\ell)), u := H \cdot e$

$$w + v = \begin{array}{c} \begin{array}{|l|}\hline \quad \beta_1 \cdot \Delta \qquad\qquad\qquad\qquad\qquad\qquad \\ \hline \end{array} \\ + ... + \\ \begin{array}{|r|}\hline \qquad\qquad\qquad\qquad\qquad\qquad \beta_\ell \cdot \Delta \\ \hline \end{array} \end{array} = H \cdot e \cdot \Delta = u \cdot \Delta$$

From PCG to PCF

- Analogous to the extension from PRG to PRF
- Main problem: N is super-polynomial
- If H has no structure, then evaluating the inner-product is infeasible



Expand-Accumulate LPN

- Structure:

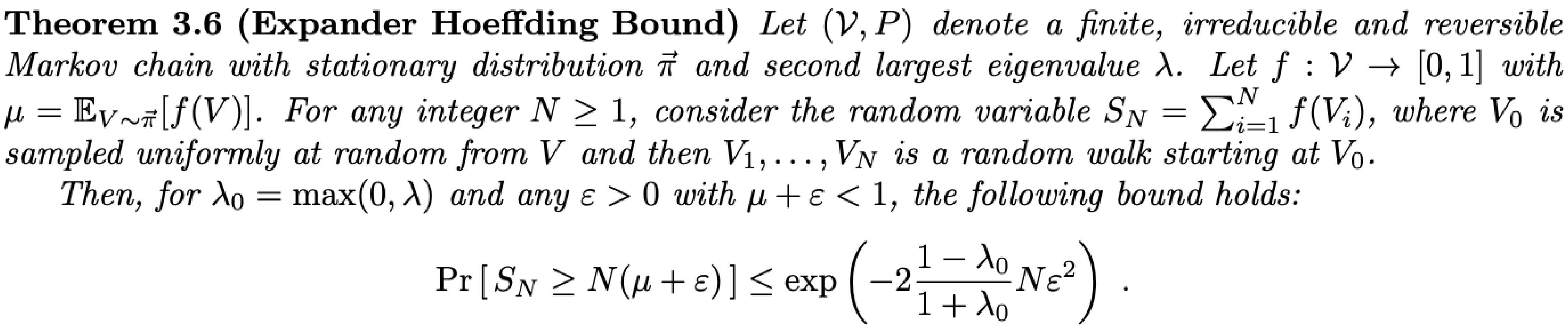
$$\boxed{\qquad H \qquad} = \underset{c_1\,c_2\quad \cdots \quad c_N}{\boxed{\qquad Ber \qquad}} \times \begin{array}{|ccccc|}\hline 1 & 0 & \ldots & 0 & 0 \\ 1 & 1 & \ldots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 1 & 1 & \ldots & 1 & 0 \\ 1 & 1 & \ldots & 1 & 1 \\ \hline \end{array}$$

  - $h_N = c_N$
  - $h_{N-1} = h_N + c_{N-1}$
  - $h_1 = h_2 + c_1$

- Proof strategy: Prove that one row in *H* is high-weight whp.

- Intuition1: Bernoulli noise accumulates to uniform due to pilling up lemma
- Intuition2: Columns of *H* corresponds to random walk

Tools: Random Walk and Markov Chain

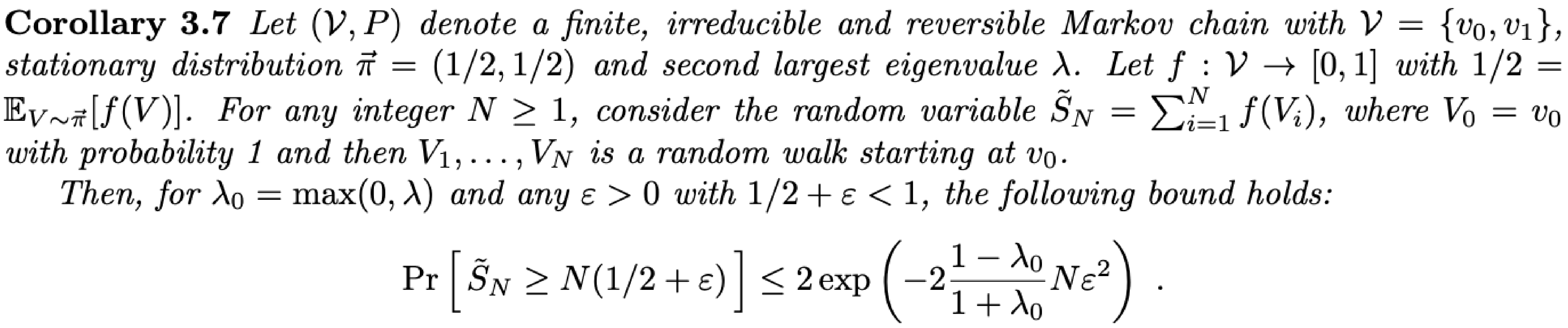


Theorem 3.6 (Expander Hoeffding Bound) *Let $(\mathcal{V}, P)$ denote a finite, irreducible and reversible Markov chain with stationary distribution $\vec{\pi}$ and second largest eigenvalue λ. Let $f : \mathcal{V} \to [0,1]$ with $\mu = \mathbb{E}_{V \sim \vec{\pi}}[f(V)]$. For any integer $N \geq 1$, consider the random variable $S_N = \sum_{i=1}^{N} f(V_i)$, where V_0 is sampled uniformly at random from V and then $V_1, \ldots, V_N$ is a random walk starting at V_0.*

Then, for $\lambda_0 = \max(0, \lambda)$ and any $\varepsilon > 0$ with $\mu + \varepsilon < 1$, the following bound holds:

$$\Pr\left[\, S_N \geq N(\mu + \varepsilon) \,\right] \leq \exp\left(-2 \frac{1 - \lambda_0}{1 + \lambda_0} N \varepsilon^2 \right) \ .$$

- Applying Markov bound

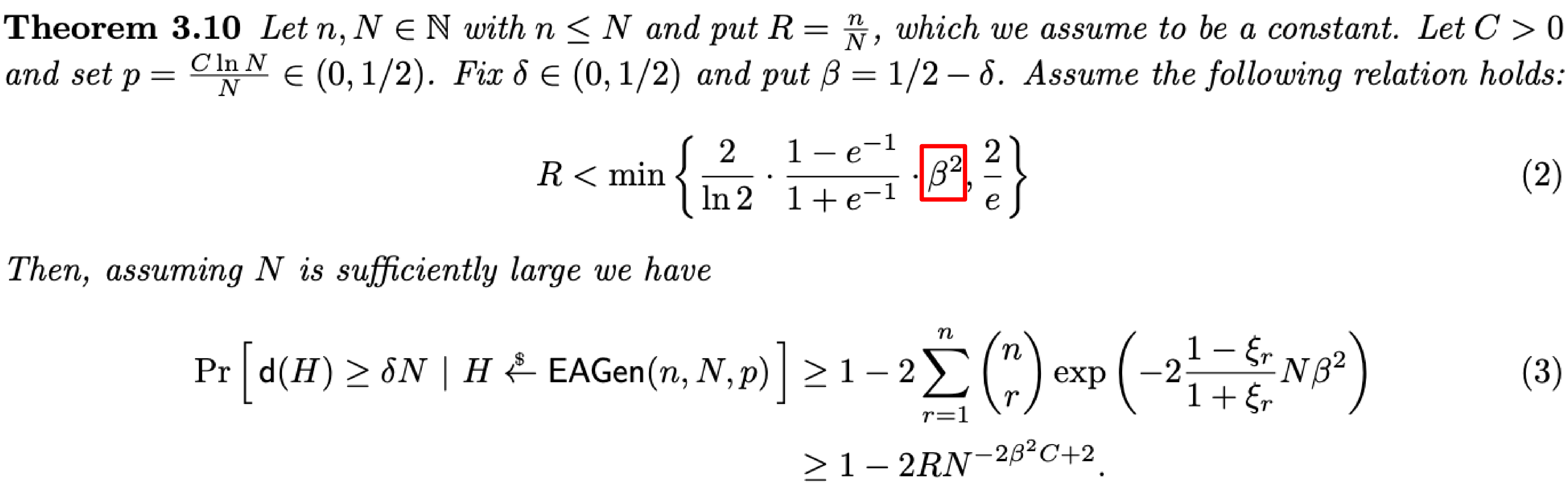


Corollary 3.7 *Let $(\mathcal{V}, P)$ denote a finite, irreducible and reversible Markov chain with $\mathcal{V} = \{v_0, v_1\}$, stationary distribution $\vec{\pi} = (1/2, 1/2)$ and second largest eigenvalue λ. Let $f : \mathcal{V} \to [0,1]$ with $1/2 = \mathbb{E}_{V \sim \vec{\pi}}[f(V)]$. For any integer $N \geq 1$, consider the random variable $\tilde{S}_N = \sum_{i=1}^{N} f(V_i)$, where $V_0 = v_0$ with probability 1 and then $V_1, \ldots, V_N$ is a random walk starting at v_0.*

Then, for $\lambda_0 = \max(0, \lambda)$ and any $\varepsilon > 0$ with $1/2 + \varepsilon < 1$, the following bound holds:

$$\Pr\left[\tilde{S}_N \geq N(1/2 + \varepsilon) \right] \leq 2 \exp\left(-2 \frac{1 - \lambda_0}{1 + \lambda_0} N \varepsilon^2 \right) \ .$$

Security of EA-LPN against Linear Attack



Theorem 3.10 *Let $n, N \in \mathbb{N}$ with $n \leq N$ and put $R = \frac{n}{N}$, which we assume to be a constant. Let $C > 0$ and set $p = \frac{C \ln N}{N} \in (0, 1/2)$. Fix $\delta \in (0, 1/2)$ and put $\beta = 1/2 - \delta$. Assume the following relation holds:*

$$R < \min\left\{\frac{2}{\ln 2} \cdot \frac{1 - e^{-1}}{1 + e^{-1}} \cdot \boxed{\beta^2}, \frac{2}{e}\right\} \tag{2}$$

Then, assuming N is sufficiently large we have

$$\begin{aligned}\Pr\left[\mathsf{d}(H) \geq \delta N \mid H \xleftarrow{\$} \mathsf{EAGen}(n, N, p)\right] &\geq 1 - 2\sum_{r=1}^{n} \binom{n}{r} \exp\left(-2\frac{1 - \xi_r}{1 + \xi_r} N\beta^2\right) \\ &\geq 1 - 2RN^{-2\beta^2 C + 2}.\end{aligned} \tag{3}$$

- (ϵ, η)-security: $\Pr[\mathsf{d}(H) \geq d] > \eta$ and $\max_{|v| \geq d} \mathsf{bias}_v(\chi^N) \leq \epsilon$
- $\mathsf{d}(H) \geq \delta N \rightarrow \mathsf{bias} \leq \frac{1}{2} \cdot (1 - 2 \cdot \frac{t}{N})^{\delta N} \approx \frac{1}{2} \cdot 2^{-2t\delta}$
- For $C = O(1)$, $\eta = 1 - \frac{1}{\mathsf{poly}}$; for $C = \log(N)$, $\eta = 1 - \mathsf{negl}$

Proving Theorem 3.10 Using Random Walk

- Differentiate between different hamming weight of x

$$\boxed{x^T} \times \boxed{H} = \boxed{x^T} \times \boxed{Ber(p)} \times \boxed{\begin{matrix} 1 & 0 & \dots & 0 & 0 \\ 1 & 1 & \dots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 1 & 1 & \dots & 1 & 0 \\ 1 & 1 & \dots & 1 & 1 \end{matrix}}$$

$$= \boxed{Ber'(p')} \times A$$

- Pilling-up lemma: $2 \cdot Bias' = (2 \cdot Bias)^{|x|} \iff \xi_r = \xi^r$ s.t. $|x| = r$
- Applying the Hoeffding bound:

$$\Pr[\mathrm{wt}(Ber') \leq (\tfrac{1}{2} - \beta) \cdot N] \leq 2 \cdot \exp(-2 \cdot \tfrac{1-2 \cdot Bias'}{1+2 \cdot Bias'} \cdot N \cdot \beta^2)$$

- This gives the first inequality in Theorem 3.10

$$\Pr\left[\mathsf{d}(H) \geq \delta N \mid H \xleftarrow{\$} \mathsf{EAGen}(n, N, p)\right] \geq 1 - 2\sum_{r=1}^{n} \binom{n}{r} \exp\left(-2\frac{1-\xi_r}{1+\xi_r} N\beta^2\right)$$

Bounding the Failure Probability

- $r = 1$

$$\begin{aligned}\binom{n}{1}\exp(-2\cdot\frac{1-\xi}{1+\xi}\cdot N\cdot\beta^2) &\le RN\cdot\exp(-2pN\beta^2)\\ &= RN\cdot\exp(-2\frac{C\ln N}{N}N\beta^2)\\ &\le N^{-2C\beta^2+1}\end{aligned}$$

- $2 \le r \le \frac{N}{2C\ln N}$: Equivalent to prove

$$\ln\left(\binom{n}{r}\exp(-2\cdot\tfrac{1-\xi_r}{1+\xi_r}N\beta^2)\right) = -\Omega(\log N)$$

$$-1\cdot\ln\left(\binom{n}{r}\exp(-2\cdot\frac{1-\xi_r}{1+\xi_r}N\beta^2)\right) = 2\cdot\frac{1-\xi_r}{1+\xi_r}N\beta^2 - \ln\binom{n}{r}$$

$$\ge (1-\xi_r)N\beta^2 - r\ln(\frac{eRN}{r}) \ge \ln(N^{2C\beta^2-1})$$

$$R \le \frac{e}{2}$$

Bounding Failure Probability (Continued)

- $r \geq \frac{N}{2C \ln N}$

$$\xi_r = (1 - \tfrac{2C \ln N}{N})^r \leq e^{-1}$$
$$\ln \tbinom{n}{r} \leq \ln(2^{RN}) = RN \ln(2)$$

$$-1 \cdot \ln \left(\binom{n}{r} \exp(-2 \cdot \frac{1-\xi_r}{1+\xi_r} N\beta^2) \right) = 2 \cdot \frac{1-\xi_r}{1+\xi_r} N\beta^2 - \ln \binom{n}{r}$$
$$\geq 2 \cdot \frac{1-e^{-1}}{1+e^{-1}} N\beta^2 - RN \ln(2) > 0$$

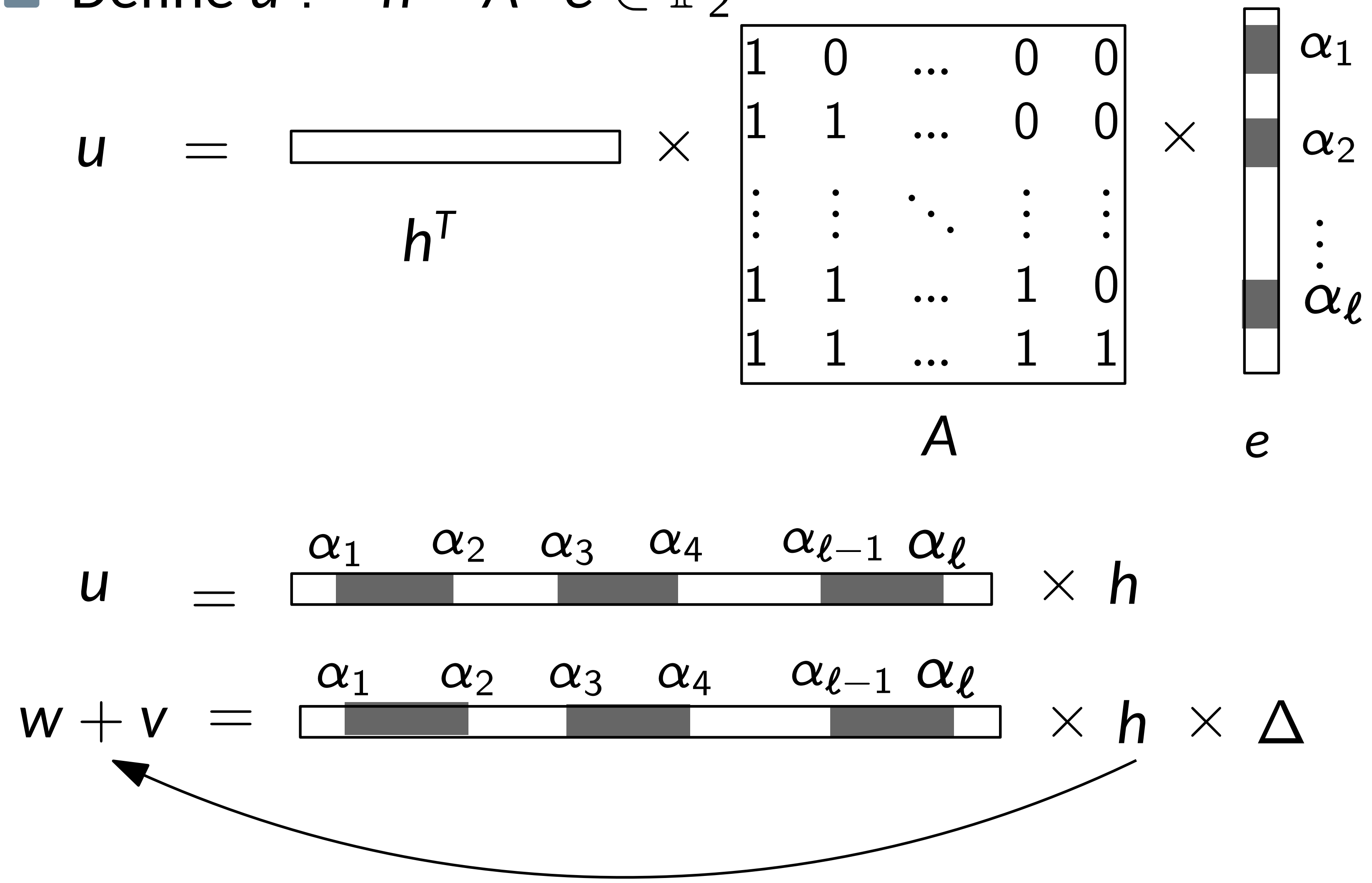
$$R < 2 \cdot \tfrac{1-e^{-1}}{1+e^{-1}} \cdot \tfrac{\beta^2}{\ln(2)}$$

- Summing over $1 \leq r \leq n$:

$$\Pr[\mathrm{Fail}] \leq 2 \cdot n \cdot N^{-2C\beta^2+1} = 2 \cdot R \cdot N^{-2C\beta^2+2}$$

Constructing PCF from EA-LPN

- Sample one row of H: $\mathsf{Samp}(x) \mapsto h^T$
- Define $u := h^T \cdot A \cdot e \in \mathbb{F}_2$



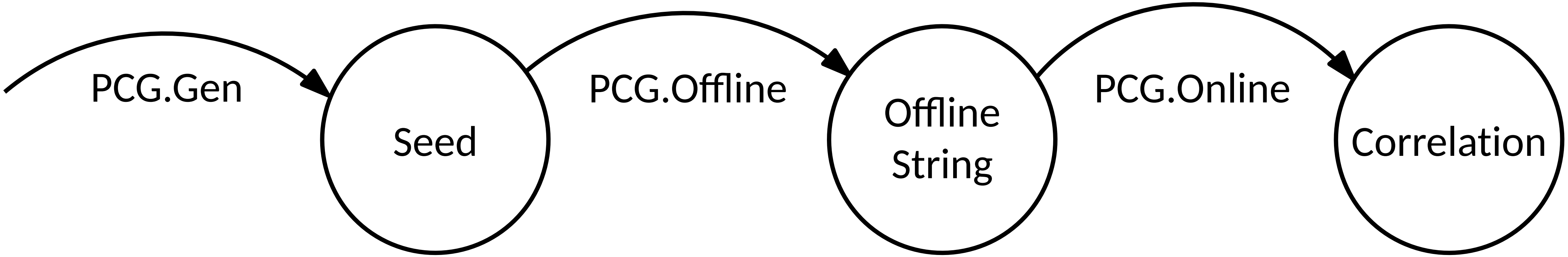
$$u = h^T \times \begin{bmatrix} 1 & 0 & \dots & 0 & 0 \\ 1 & 1 & \dots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 1 & 1 & \dots & 1 & 0 \\ 1 & 1 & \dots & 1 & 1 \end{bmatrix} \times e$$

h^T
A
e
α_1
α_2
α_ℓ
$u =$
α_1 α_2 α_3 α_4 $\alpha_{\ell-1}$ α_ℓ
$\times h$
$w + v =$
α_1 α_2 α_3 α_4 $\alpha_{\ell-1}$ α_ℓ
$\times h \times \Delta$

Run DCF on every (public) non-zero coordinate of h

Generalizations

- Offline/Online PCG

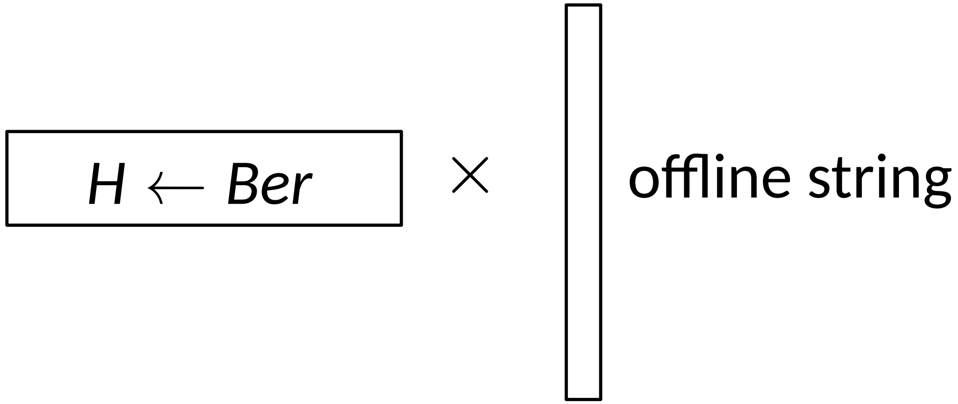


- Motivation: Utilize online idle time to mitigate offline burden

- Expand:

$$w := H \cdot (FullEval(k_0^1) + ... + FullEval(k_0^\ell))$$

$$v := H \cdot (FullEval(k_1^1) + ... + FullEval(k_1^\ell)), u := H \cdot e$$

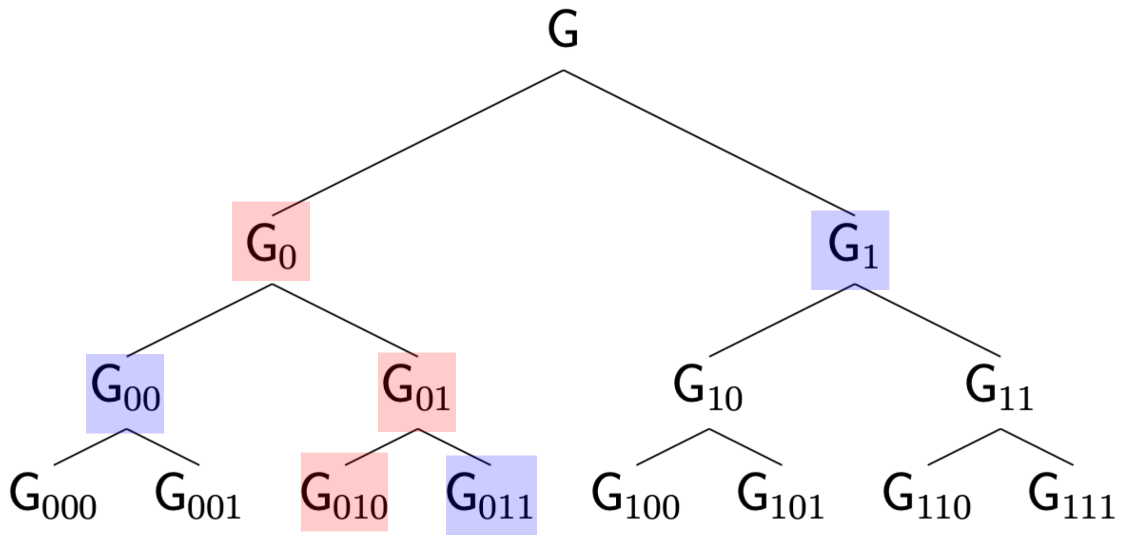


Relaxed Distributed Comparison Function

- RDCF: $f(x) = \begin{cases} 0 & x \leq \alpha \\ \beta & x > \alpha \end{cases}$

$\mathsf{Expand}(k^0) \mapsto \alpha, y^0$ $\mathsf{Expand}(k^0) \mapsto y^1$

- Example: $\alpha = 010$



- $\gamma_1 = H(G), \gamma_2 = H(G_0), \gamma_3 = H(G_{01})$
- $c_1 = \bar{\alpha}_1 \cdot \gamma_1, c_2 = \bar{\alpha}_2 \cdot \gamma_2, c_3 = \bar{\alpha}_3 \cdot \gamma_3$
- $B_i = c_1 + \ldots + c_{i-1} + \alpha_i \cdot \gamma_i + \alpha_i \cdot \beta$

$k^1 := G$

$k^0 := \langle \alpha, \{B_i\}, y = G_{010} + \sum c_i, G_1, G_{00}, G_{011} \rangle$

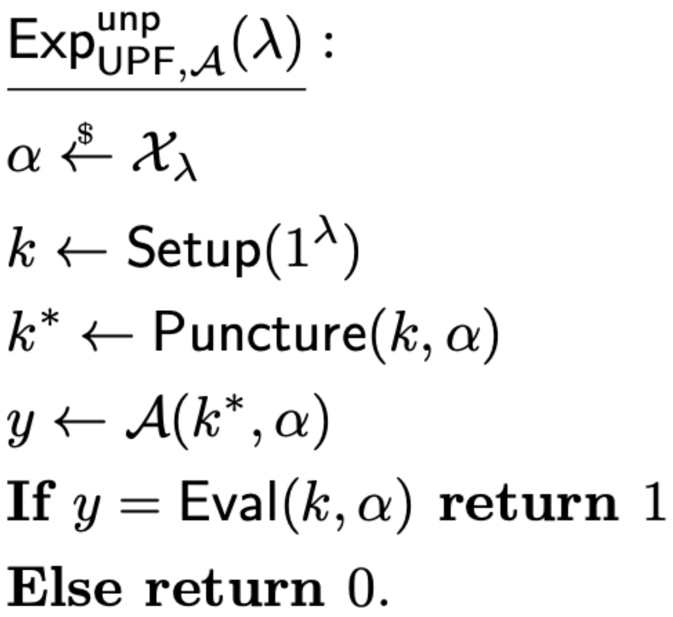
- Eval (x): Define $c^1_1 = \bar{x}_1 \cdot H(G), c^1_2 = \bar{x}_2 \cdot H(G_{x_1}), c^1_3 = \bar{x}_3 \cdot H(G_{x_1 x_2})$

$f^1(x) = G_{x_1 x_2 x_3} + \sum c^1_i$

$f^0(x) = \begin{cases} y & x = \alpha \\ B_j + c^1_{j+1} + \ldots + c^1_m & x \neq \alpha \end{cases}$

Offline Optimization: UPF

- Replace pseudorandomness in PPRF by unpredictability

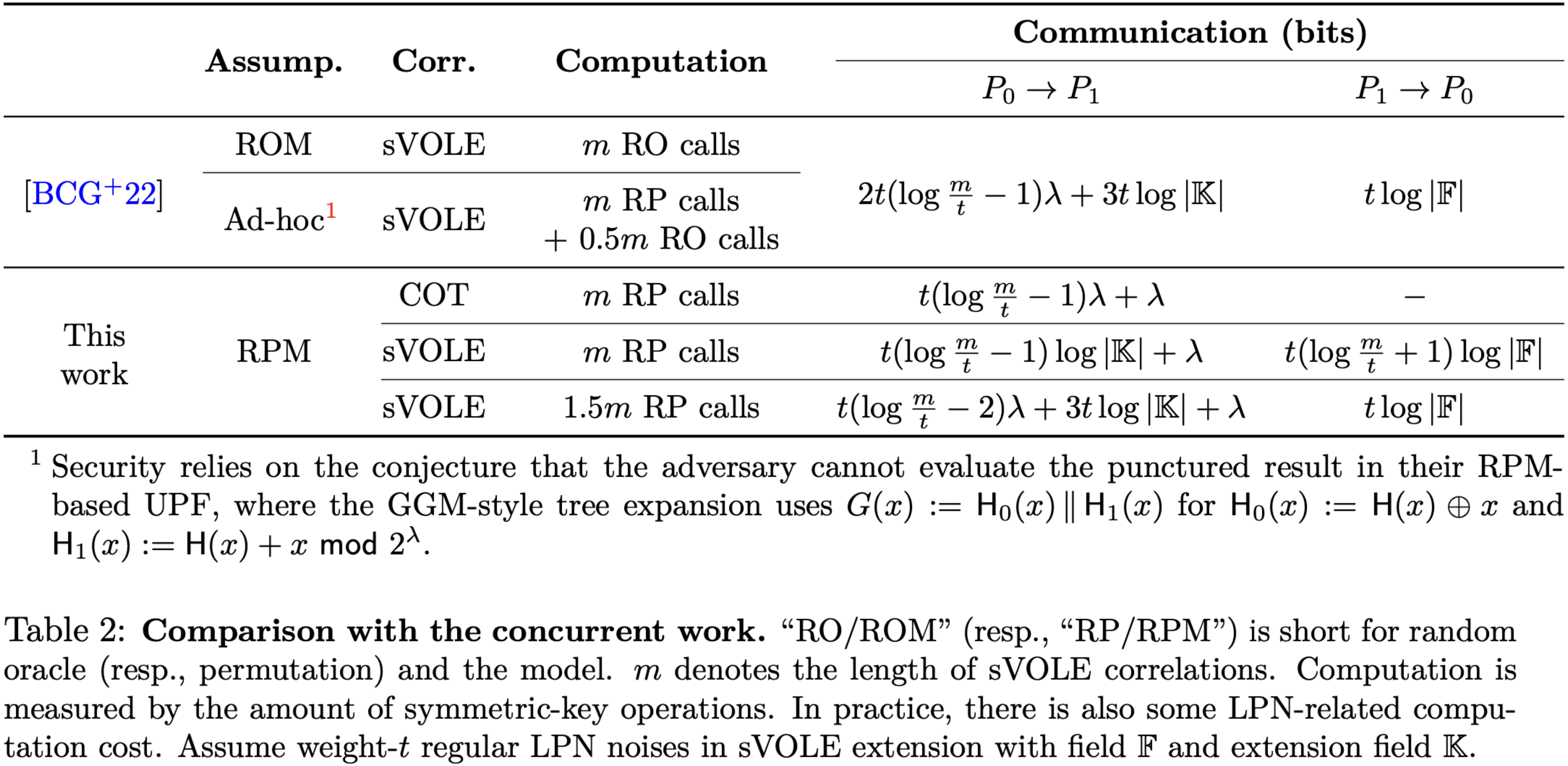


```
$\mathsf{Exp}^{\mathsf{unp}}_{\mathsf{UPF},\mathcal{A}}(\lambda)$ :
$\alpha \xleftarrow{\$} \mathcal{X}_\lambda$
$k \leftarrow \mathsf{Setup}(1^\lambda)$
$k^* \leftarrow \mathsf{Puncture}(k, \alpha)$
$y \leftarrow \mathcal{A}(k^*, \alpha)$
$\textbf{If } y = \mathsf{Eval}(k, \alpha) \textbf{ return } 1$
$\textbf{Else return } 0.$
```

- Step1: a UPF that takes N ROs
- Step2: a PPRF by hashing the left leaves of UPF that takes $N/2$ ROs
- Computation saving: $2N \rightarrow 1.5N$

Summary

- Contribution 1: EA-LPN
- Contribution 2: Offline Optimization (checkout on Half-tree)



	Assump.	Corr.	Computation	Communication (bits)	
				$P_0 \to P_1$	$P_1 \to P_0$
[BCG+22]	ROM	sVOLE	m RO calls	$2t(\log \frac{m}{t} - 1)\lambda + 3t \log \lvert\mathbb{K}\rvert$	$t \log \lvert\mathbb{F}\rvert$
	Ad-hoc[1]	sVOLE	m RP calls $+ \, 0.5m$ RO calls		
This work	RPM	COT	m RP calls	$t(\log \frac{m}{t} - 1)\lambda + \lambda$	–
		sVOLE	m RP calls	$t(\log \frac{m}{t} - 1) \log \lvert\mathbb{K}\rvert + \lambda$	$t(\log \frac{m}{t} + 1) \log \lvert\mathbb{F}\rvert$
		sVOLE	$1.5m$ RP calls	$t(\log \frac{m}{t} - 2)\lambda + 3t \log \lvert\mathbb{K}\rvert + \lambda$	$t \log \lvert\mathbb{F}\rvert$

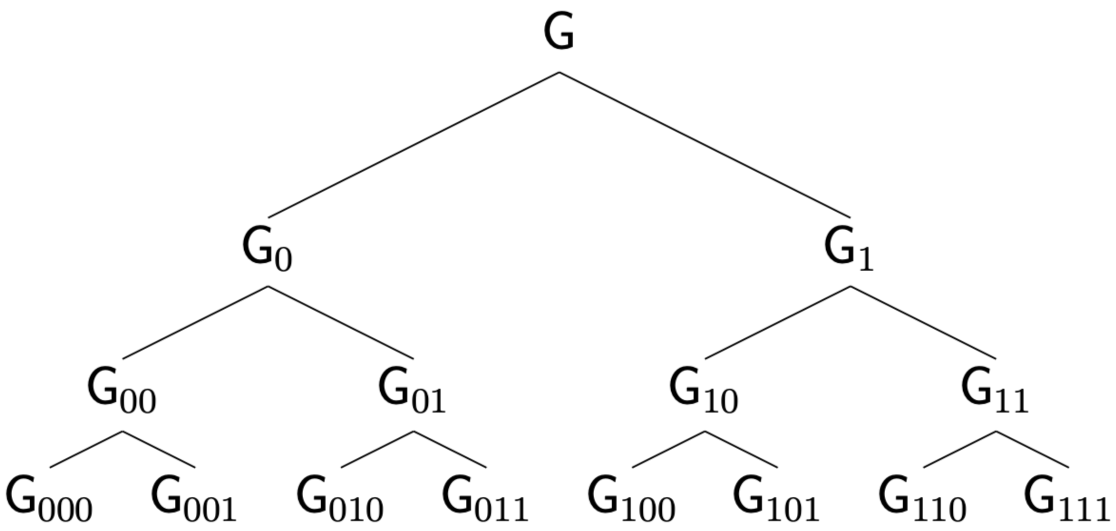
[1] Security relies on the conjecture that the adversary cannot evaluate the punctured result in their RPM-based UPF, where the GGM-style tree expansion uses $G(x) := \mathsf{H}_0(x) \,\|\, \mathsf{H}_1(x)$ for $\mathsf{H}_0(x) := \mathsf{H}(x) \oplus x$ and $\mathsf{H}_1(x) := \mathsf{H}(x) + x \bmod 2^\lambda$.

Table 2: **Comparison with the concurrent work.** "RO/ROM" (resp., "RP/RPM") is short for random oracle (resp., permutation) and the model. m denotes the length of sVOLE correlations. Computation is measured by the amount of symmetric-key operations. In practice, there is also some LPN-related computation cost. Assume weight-t regular LPN noises in sVOLE extension with field $\mathbb{F}$ and extension field $\mathbb{K}$.

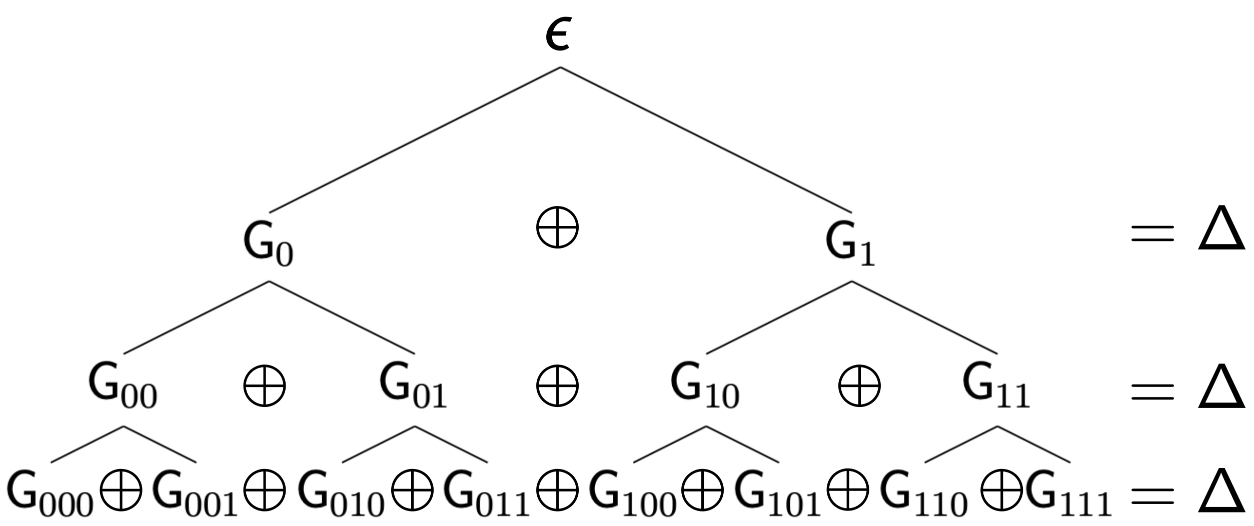
Half-tree Optimization

- Save computation/communication by introducing correlation at each level

GGM Tree

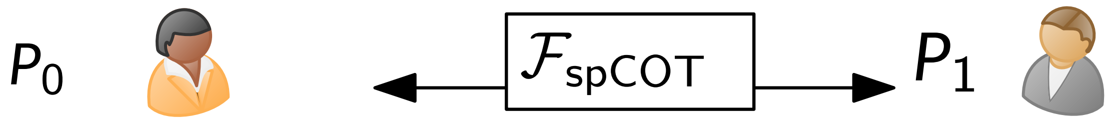

G
G_0 G_1
G_{00} G_{01} G_{10} G_{11}
G_{000} G_{001} G_{010} G_{011} G_{100} G_{101} G_{110} G_{111}

Correlated GGM Tree


ϵ
$G_0 \oplus G_1 = \Delta$
$G_{00} \oplus G_{01} \oplus G_{10} \oplus G_{11} = \Delta$
$G_{000} \oplus G_{001} \oplus G_{010} \oplus G_{011} \oplus G_{100} \oplus G_{101} \oplus G_{110} \oplus G_{111} = \Delta$

- Expansion: $G_{00}||G_{01} = \mathsf{PRG}(G_0)$ $\quad$ $G_{00} = \mathsf{H}(G_0), G_{01} = G_0 \oplus G_{00}$
- Costs: $N\times$ RO or $2N\times$ RP $\quad$ $N\times$ RP
- Initial Setup: $G \leftarrow \mathbb{F}_2^\kappa$ $\quad$ $G_0 = k \leftarrow \mathbb{F}_2^\kappa \quad G_1 = \Delta - k$

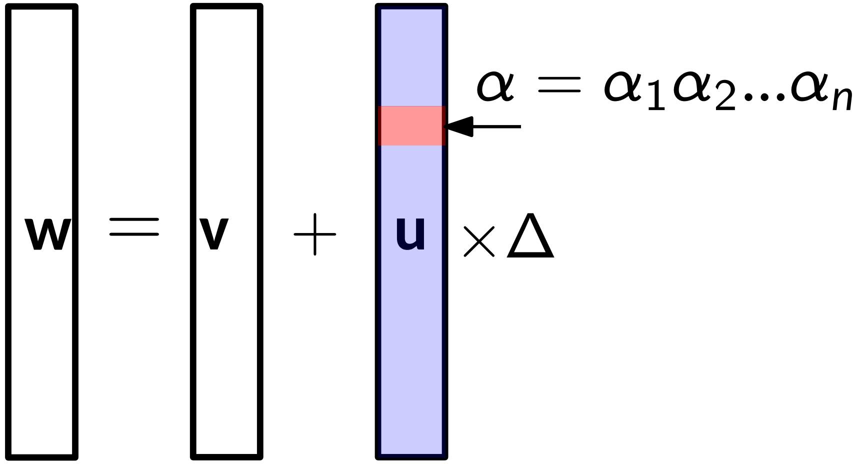
Example 1: Single Point COT



$\Delta \in \mathbb{F}_{2^\kappa}, \mathbf{v} \in \mathbb{F}_{2^\kappa}^N$

$\mathbf{u} \in \mathbb{F}_2^N, \mathbf{w} \in \mathbb{F}_{2^\kappa}^N$

$\|\mathbf{u}\| = 1$



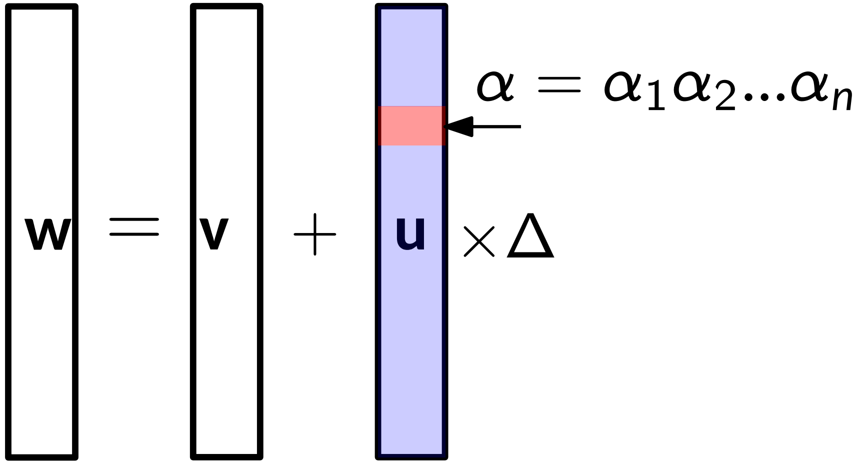
Example 1: Single Point COT

P_0 $\longleftarrow$ $\boxed{\mathcal{F}_{\text{spCOT}}}$ $\longrightarrow$ P_1

$\Delta \in \mathbb{F}_{2^\kappa}, \mathbf{v} \in \mathbb{F}_{2^\kappa}^N$

$\mathbf{u} \in \mathbb{F}_2^N, \mathbf{w} \in \mathbb{F}_{2^\kappa}^N$

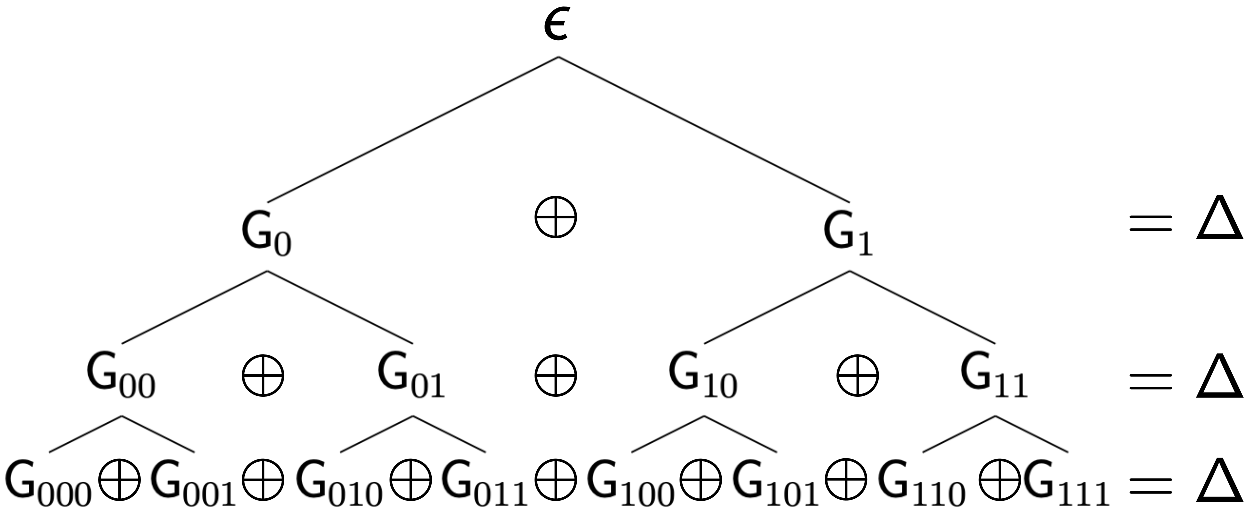
$\|\mathbf{u}\| = 1$



$\mathbf{w} = \mathbf{v} + \mathbf{u} \times \Delta$

$\alpha = \alpha_1\alpha_2 \ldots \alpha_n$

- Setup: $\mathcal{F}_{\text{COT}}$ with Δ global key
- Prepare $[r_1], \ldots, [r_n] \in \mathbb{F}_2^n$



ϵ

$G_0 \oplus G_1 = \Delta$

$G_{00} \oplus G_{01} \oplus G_{10} \oplus G_{11} = \Delta$

$G_{000} \oplus G_{001} \oplus G_{010} \oplus G_{011} \oplus G_{100} \oplus G_{101} \oplus G_{110} \oplus G_{111} = \Delta$

P_0 For $i \in [1, n]$ P_1

$K_0^i := \bigoplus G_{*0}, K_1^i := \bigoplus G_{*1}$

$$\xrightarrow{c_i := K_0^i \oplus \mathsf{K}[r_i]}$$

For $i \in [1, n]$, $\alpha_i := \bar{r}_i$

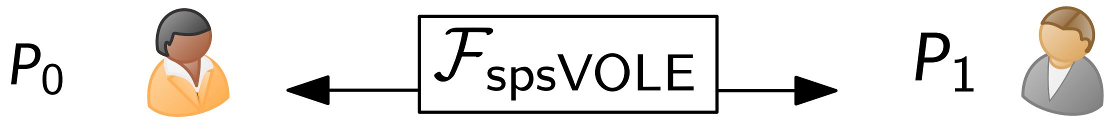
$K_{\bar{\alpha}_i}^i = c_i \oplus \mathsf{M}[r_i]$

For $x \in [1, N]$

$\mathbf{v}[x] = G_x$

$$\mathbf{w}[x] = \begin{cases} G_x & x \neq \alpha \\ -\sum_{x \neq \alpha} G_x & x = \alpha \end{cases}$$

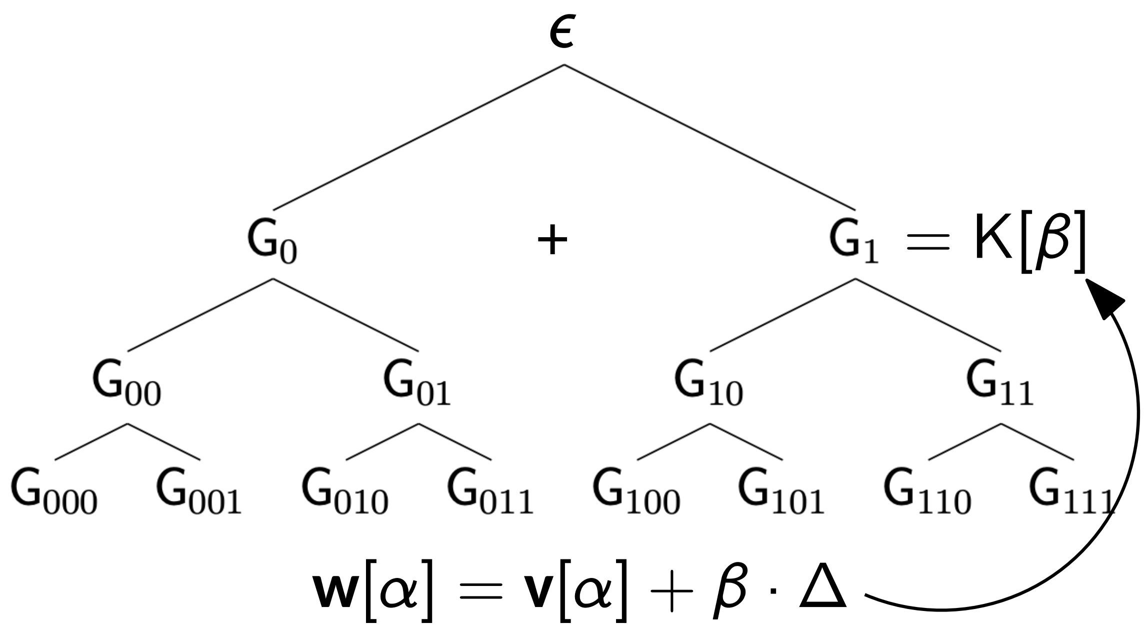
Example 2: Single Point sVOLE



$\mathbf{u} \in \mathbb{F}^N, \mathbf{w} \in \mathbb{K}^N$

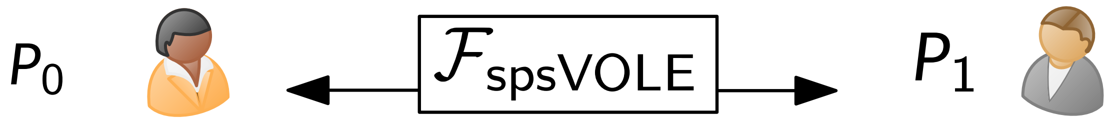
$\Delta \in \mathbb{K}, \mathbf{v} \in \mathbb{K}^N$

$||\mathbf{u}|| = 1, \mathbf{u}[\alpha] = \beta \in \mathbb{F}$



$\mathbf{w}[\alpha] = \mathbf{v}[\alpha] + \beta \cdot \Delta$

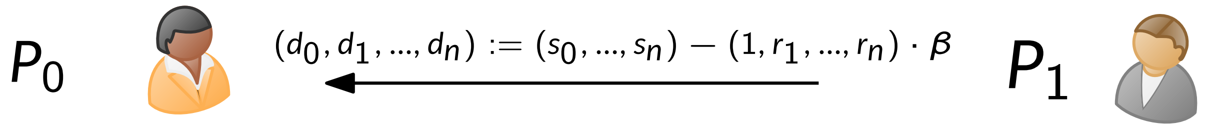
Example 2: Single Point sVOLE



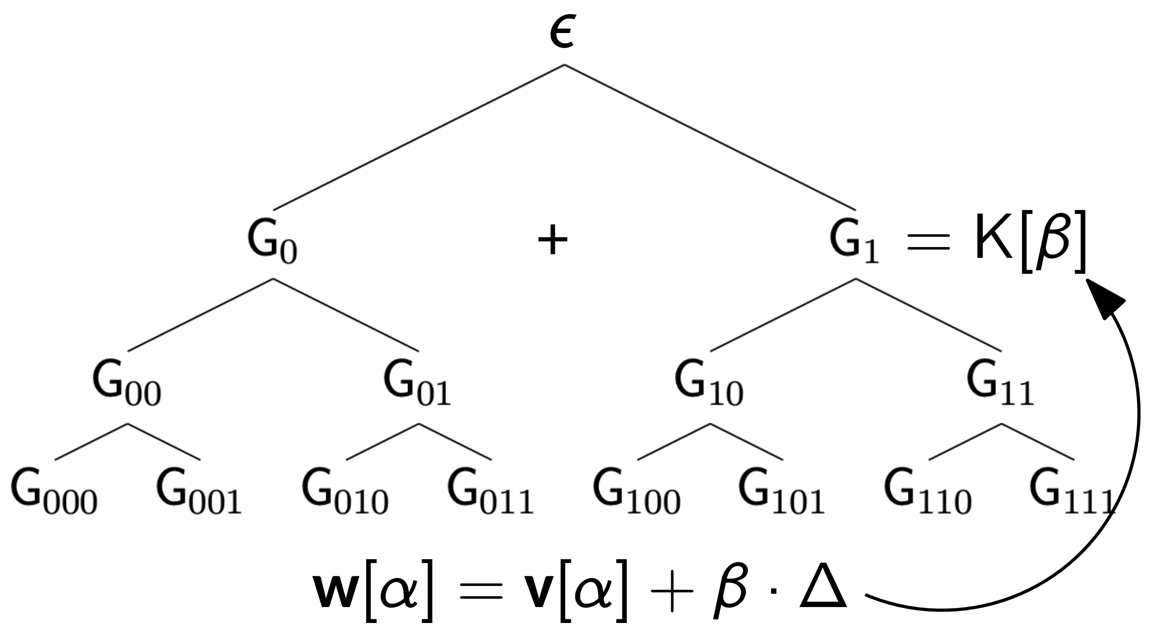
$\mathbf{u} \in \mathbb{F}^N, \mathbf{w} \in \mathbb{K}^N$

$\Delta \in \mathbb{K}, \mathbf{v} \in \mathbb{K}^N$ $\qquad$ $||\mathbf{u}|| = 1, \mathbf{u}[\alpha] = \beta \in \mathbb{F}$

- Setup: $\mathcal{F}_{\text{sVOLE}}$ with Δ global key
- Prepare $[s_0], [s_1], ..., [s_n] \in \mathbb{F}$


P_0
$(d_0, d_1, ..., d_n) := (s_0, ..., s_n) - (1, r_1, ..., r_n) \cdot \beta$
P_1

$\mathsf{K}[\beta] := \mathsf{K}[s_0] + d_0 \cdot \Delta$ $\qquad$ $\mathsf{M}[\beta] = \mathsf{M}[s_0]$



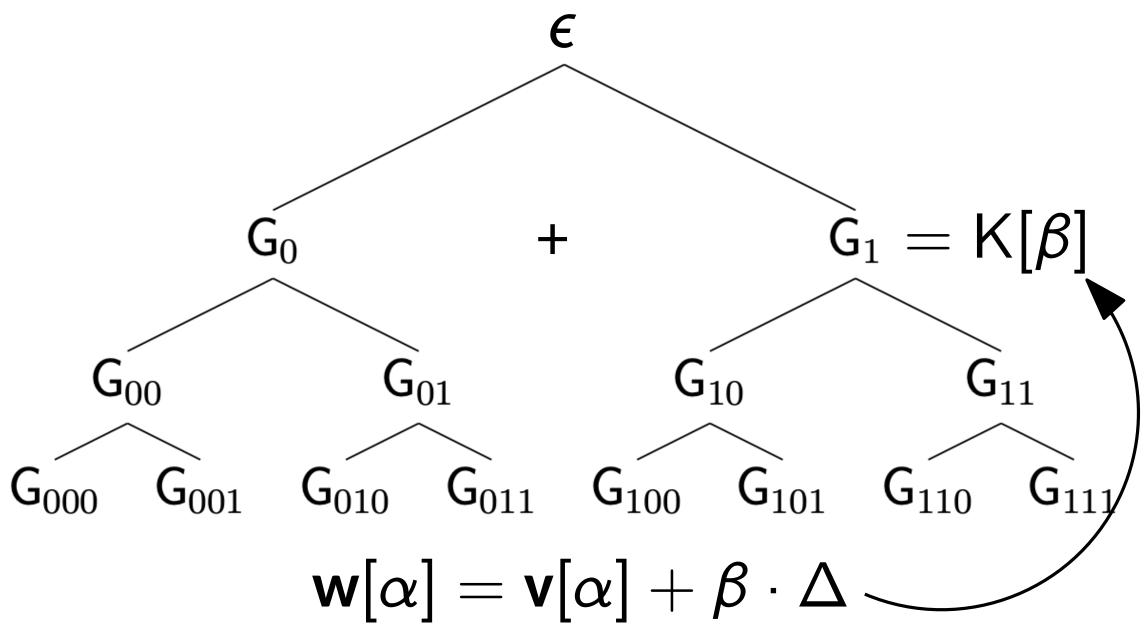
Example 2: Single Point sVOLE

P_0 $\xleftarrow{\quad}$ $\boxed{\mathcal{F}_{\text{spsVOLE}}}$ $\xrightarrow{\quad}$ P_1

$\Delta \in \mathbb{K}, \mathbf{v} \in \mathbb{K}^N$

$\mathbf{u} \in \mathbb{F}^N, \mathbf{w} \in \mathbb{K}^N$

$||\mathbf{u}|| = 1, \mathbf{u}[\alpha] = \beta \in \mathbb{F}$



- Setup: $\mathcal{F}_{\text{sVOLE}}$ with Δ global key
- Prepare $[s_0], [s_1], ..., [s_n] \in \mathbb{F}$

P_0 $\xleftarrow{(d_0, d_1, ..., d_n) := (s_0, ..., s_n) - (1, r_1, ..., r_n) \cdot \beta}$ P_1

$\mathsf{K}[\beta] := \mathsf{K}[s_0] + d_0 \cdot \Delta \qquad \mathsf{M}[\beta] = \mathsf{M}[s_0]$

- Setup cGGM using $k \leftarrow \mathbb{K}$ and $k - \mathsf{K}[\beta]$

$$\mathsf{M}[r_i] = \mathsf{K}[r_i] + r_i \cdot \mathsf{K}[\beta]$$
$$\mathsf{M}[r_i] = \mathsf{K}[r_i] + r_i \cdot (\mathsf{M}[\beta] - \beta \cdot \Delta)$$
$$r_i \cdot \mathsf{M}[\beta] - \mathsf{M}[r_i] = -\mathsf{K}[r_i] + r_i\beta \cdot \Delta$$

$\mathsf{K}[r_i] := -\mathsf{K}[s_i] - d_i \cdot \Delta \qquad \mathsf{M}[r_i] := r_i \cdot \mathsf{M}[\beta] - \mathsf{M}[s_i]$

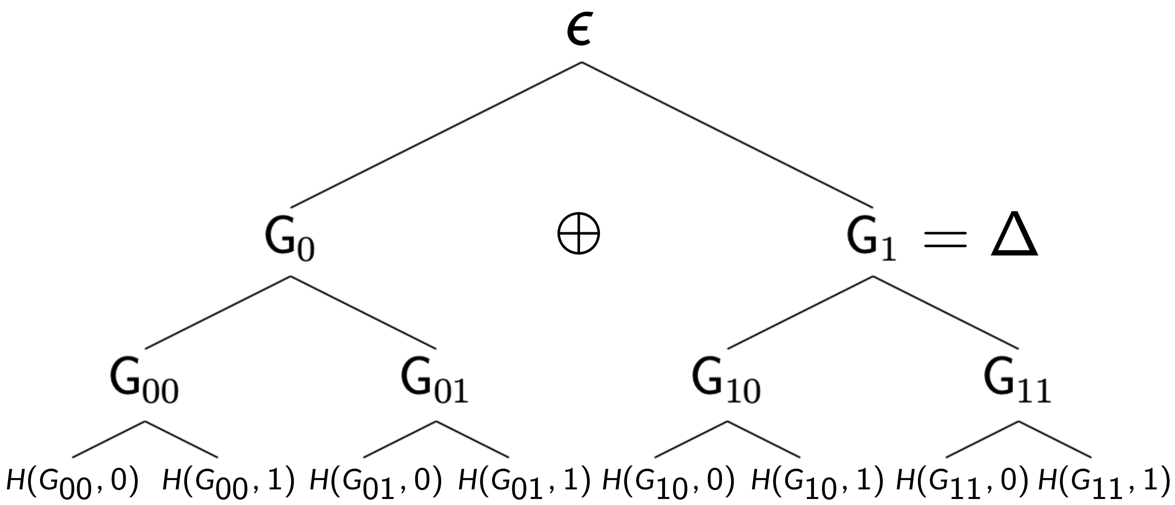
$\xrightarrow{c_i := K_0^i + \mathsf{K}[r_i]}$

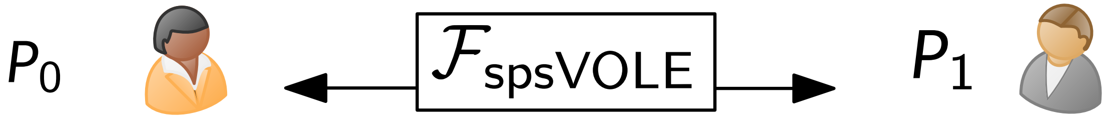
For $x \in [1, N]$

$\mathbf{v}[x] = G_x$

$$\mathbf{w}[x] = \begin{cases} G_x & x \neq \alpha \\ -\sum_{x \neq \alpha} G_x + \mathsf{M}[\beta] & x = \alpha \end{cases}$$

Example 3: Single Point sVOLE from Pseudorandom cGGM



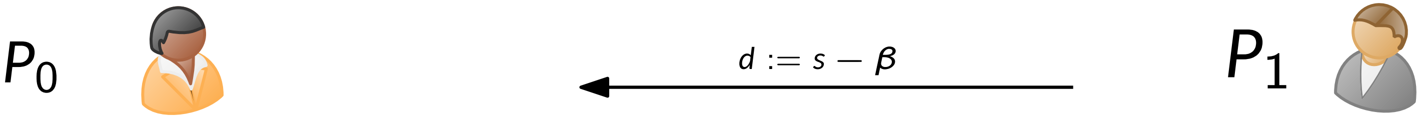


$\mathbf{u} \in \mathbb{F}^N, \mathbf{w} \in \mathbb{K}^N$

$\Delta \in \mathbb{K}, \mathbf{v} \in \mathbb{K}^N$

$||\mathbf{u}|| = 1, \mathbf{u}[\alpha] = \beta \in \mathbb{F}$

- Setup: $\mathcal{F}_{\mathsf{sVOLE}}$ with Γ global key, $\mathcal{F}_{\mathsf{COT}}$ with Δ global key
- Prepare $[r_1], \ldots, [r_n] \in \mathbb{F}_2$, $[s] \in \mathbb{F}$

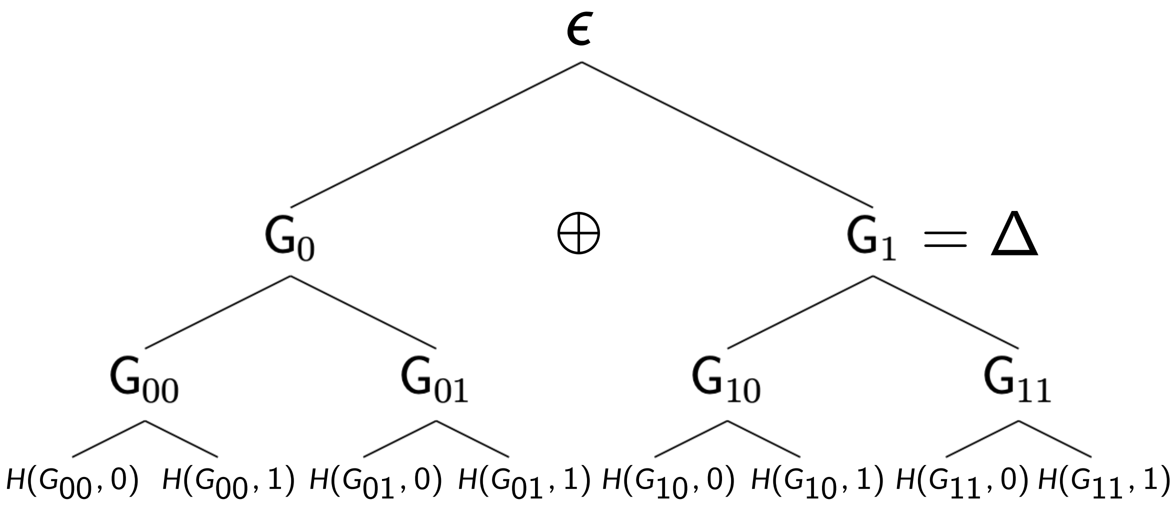


Get $\mathsf{K}[\beta]$

Run pcGGM using $\mathsf{K}[\beta]$

Get $\mathsf{M}[\beta]$

Example 3: Single Point sVOLE from Pseudorandom cGGM



ϵ

$G_0 \quad \oplus \quad G_1 = \Delta$

$G_{00} \quad G_{01} \quad G_{10} \quad G_{11}$

$H(G_{00},0)\ H(G_{00},1)\ H(G_{01},0)\ H(G_{01},1)\ H(G_{10},0)\ H(G_{10},1)\ H(G_{11},0)\ H(G_{11},1)$

$P_0 \xleftarrow{\quad} \boxed{\mathcal{F}_{\text{spsVOLE}}} \xrightarrow{\quad} P_1$

$\Delta \in \mathbb{K}, \mathbf{v} \in \mathbb{K}^N$

$\mathbf{u} \in \mathbb{F}^N, \mathbf{w} \in \mathbb{K}^N$

$||\mathbf{u}|| = 1, \mathbf{u}[\alpha] = \beta \in \mathbb{F}$

- Setup: $\mathcal{F}_{\text{sVOLE}}$ with Γ global key, $\mathcal{F}_{\text{COT}}$ with Δ global key
- Prepare $[r_1], ..., [r_n] \in \mathbb{F}_2, [s] \in \mathbb{F}$

P_0

Get $\mathsf{K}[\beta]$

Run pcGGM using $\mathsf{K}[\beta]$

P_1

Get $\mathsf{M}[\beta]$

$P_0 \xleftarrow{d := s - \beta} P_1$

$P_0 \xrightarrow{c_i := K_0^i + \mathsf{K}[r_i] \text{ for } i \in [1, n-1]} P_1$

$c_n^0 := H(\mathsf{K}[r_n]) + K_0^n, c_n^1 := H(\mathsf{K}[r_n] \oplus \Delta) + K_1^n$

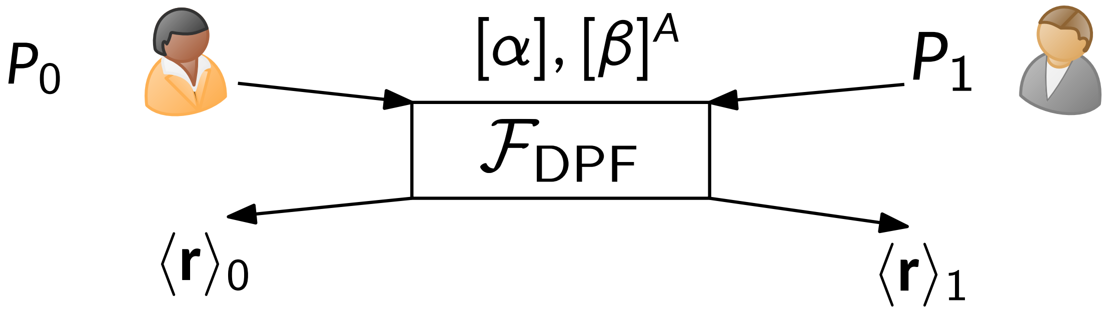
$\phi := K_0^n + K_1^n - \mathsf{K}[\beta]$

For $x \in [1, N]$

$\mathbf{v}[x] = G_x$

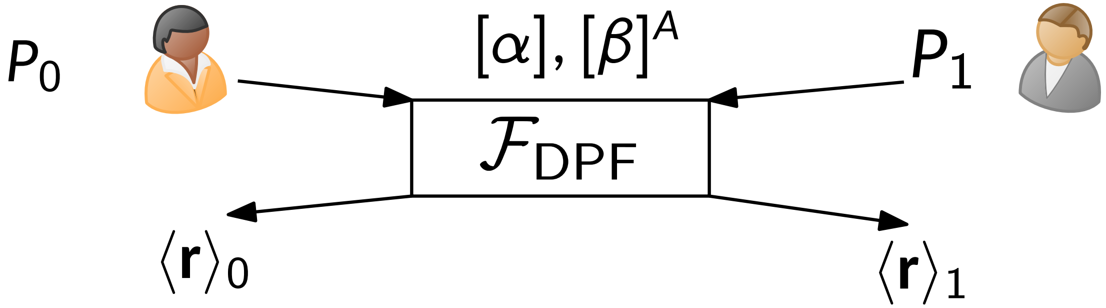
$$\mathbf{w}[x] = \begin{cases} G_x & x \neq \alpha \\ -\sum_{x \neq \alpha} G_x + \phi + \mathsf{M}[\beta] & x = \alpha \end{cases}$$

Example 4: Distributed DPF from pcGGM

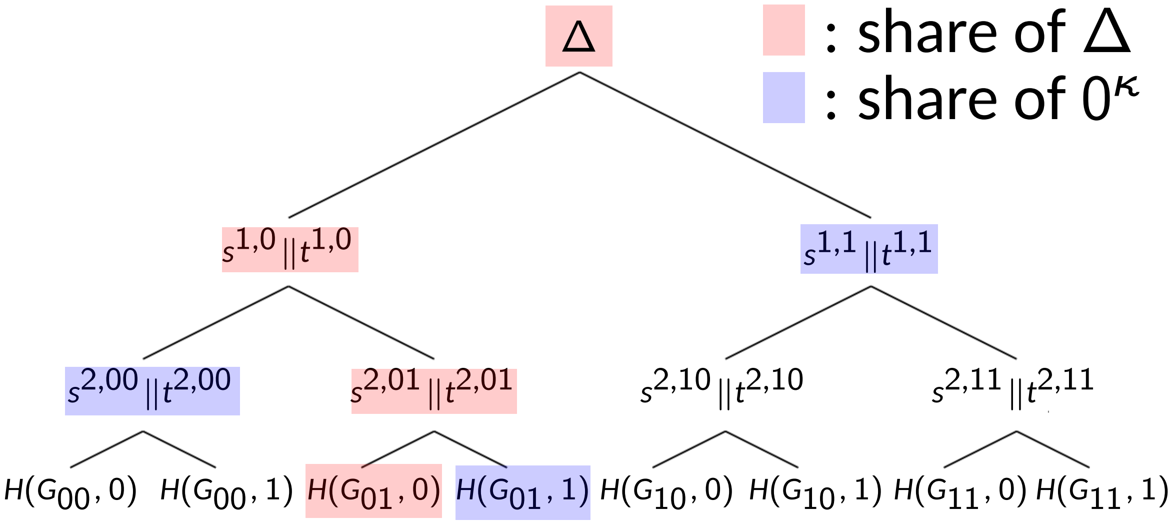

P_0
$[\alpha], [\beta]^A$
P_1
$\mathcal{F}_{\mathrm{DPF}}$
$\langle \mathbf{r} \rangle_0$
$\langle \mathbf{r} \rangle_1$

$$\langle \mathbf{r} \rangle_0[x] + \langle \mathbf{r} \rangle_1[x] = \begin{cases} 0 & x \neq \alpha \\ \beta & x = \alpha \end{cases}$$

Example 4: Distributed DPF from pcGGM

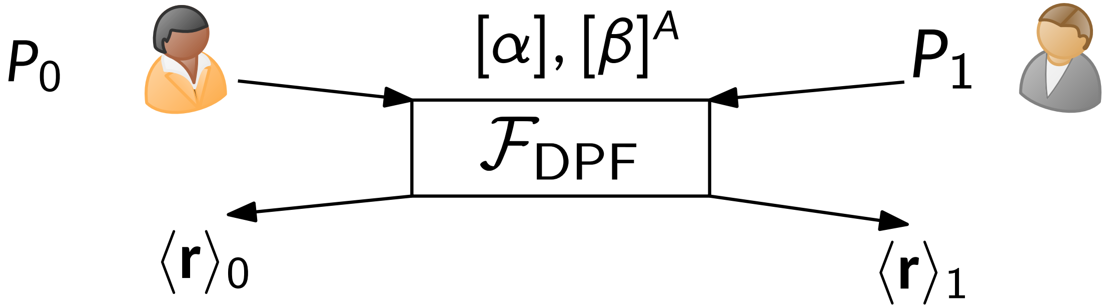

P_0
$[\alpha], [\beta]^A$
P_1
$\mathcal{F}_{\mathrm{DPF}}$
$\langle \mathbf{r} \rangle_0$
$\langle \mathbf{r} \rangle_1$

$$\langle \mathbf{r} \rangle_0[x] + \langle \mathbf{r} \rangle_1[x] = \begin{cases} 0 & x \neq \alpha \\ \beta & x = \alpha \end{cases}$$

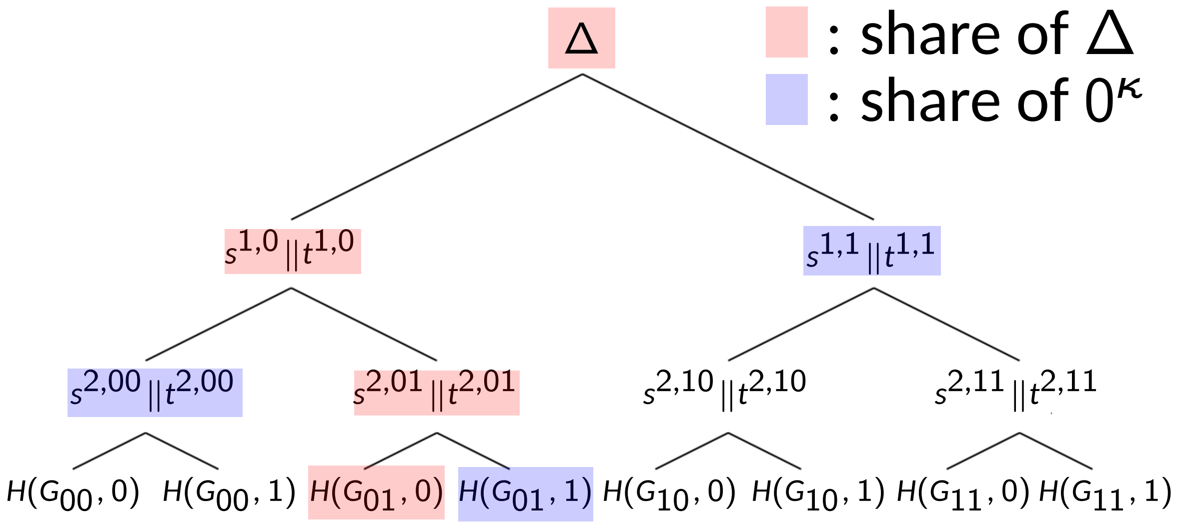


- Sample $\Delta = \langle \Delta \rangle_0 + \langle \Delta \rangle_1$ s.t. $\mathrm{lsb}(\Delta) = 1$
- Authenticate $\langle \alpha \rangle_0, \langle \alpha \rangle_1$
- Run $n + 2$ rounds to compute $CW_1, \ldots, CW_{n+1}$

Example 4: Distributed DPF from pcGGM


P_0
$[\alpha], [\beta]^A$
P_1
$\mathcal{F}_{\mathrm{DPF}}$
$\langle \mathbf{r} \rangle_0$
$\langle \mathbf{r} \rangle_1$

$$\langle \mathbf{r} \rangle_0[x] + \langle \mathbf{r} \rangle_1[x] = \begin{cases} 0 & x \neq \alpha \\ \beta & x = \alpha \end{cases}$$



- Sample $\Delta = \langle \Delta \rangle_0 + \langle \Delta \rangle_1$ s.t. $\mathrm{lsb}(\Delta) = 1$
- Authenticate $\langle \alpha \rangle_0, \langle \alpha \rangle_1$
- Run $n + 2$ rounds to compute $CW_1, \ldots, CW_{n+1}$

$$CW_i := \mathrm{H}(\langle K_0^i \rangle_0) \oplus \mathrm{H}(\langle K_0^i \rangle_1) \oplus \bar{\alpha}_i \cdot \Delta$$

$$CW_n := \mathrm{H}(\langle K_{\bar{\alpha}_n}^n \rangle_0) \oplus \mathrm{H}(\langle K_{\bar{\alpha}_n}^n \rangle_1)$$ NB: need 2-bit to correct LSB

$$CW_{n+1} := \left(\sum_{i=1}^N t_0^i - \sum_{i=1}^N t_1^i\right) \cdot \left(\sum_{i=1}^N s_1^i - \sum_{i=1}^N s_0^i + \beta\right)$$

$$\langle \mathbf{r} \rangle_0[x] := (-1)^0 \cdot (s_0^x + t_0^x \cdot CW_{n+1})$$

$$\langle \mathbf{r} \rangle_1[x] := (-1)^1 \cdot (s_1^x + t_1^x \cdot CW_{n+1})$$

$\boxed{\mathcal{F}_{\mathrm{OLE}}}$ is required for non-$\mathbb{F}_{2^k}$ fields

$$\langle \mathbf{r} \rangle_0[x] + \langle \mathbf{r} \rangle_1[x] = \begin{cases} 0 & x \neq \alpha \\ s_0^\alpha - s_1^\alpha + (t_0^\alpha - t_1^\alpha)^2 \cdot CW_{n+1} & x = \alpha \end{cases}$$

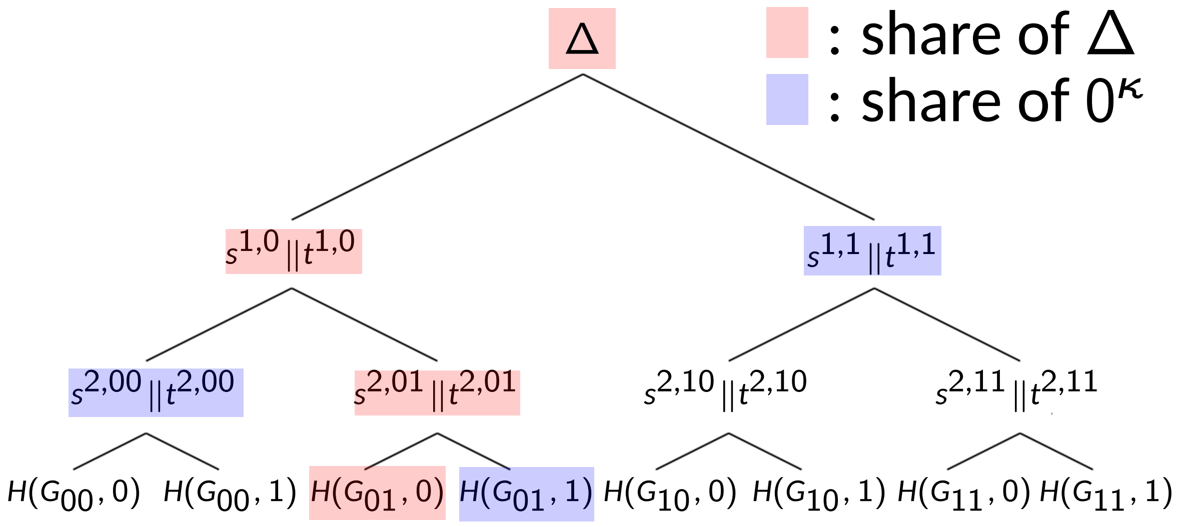
Example 5: Distributed DCF from pcGGM

$$\langle \mathbf{r} \rangle_0[x] + \langle \mathbf{r} \rangle_1[x] = \begin{cases} 0 & x \leq \alpha \\ \beta & x > \alpha \end{cases}$$

- First run DPF
- Then compute $VCW^{i+1} :=$ $(t_0^i - t_1^i) \cdot (v_1^{\alpha[1,i]} - v_0^{\alpha[1,i]} + (\alpha_{i+1} - \alpha_i) \cdot \beta)$

$$\langle \mathbf{r} \rangle_0[x] := \mathrm{DPF}_0(x) + \textstyle\sum_{i=0}^{n-1} (v_0^{x[1,i]} + t_0^i \cdot VCW_{i+1})$$

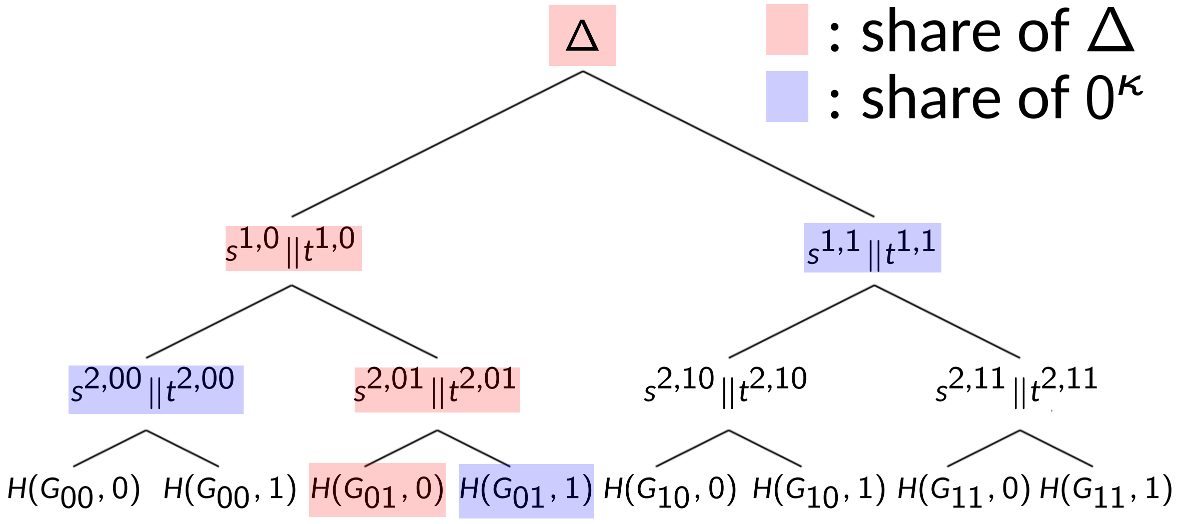
$$\langle \mathbf{r} \rangle_1[x] := \mathrm{DPF}_1(x) - \textstyle\sum_{i=0}^{n-1} (v_1^{x[1,i]} + t_1^i \cdot VCW_{i+1})$$



Example 5: Distributed DCF from pcGGM

$$\langle \mathbf{r}\rangle_0[x] + \langle \mathbf{r}\rangle_1[x] = \begin{cases} 0 & x \le \alpha \\ \beta & x > \alpha \end{cases}$$

- First run DPF
- Then compute $VCW^{i+1} :=$ $(t_0^i - t_1^i) \cdot (v_1^{\alpha[1,i]} - v_0^{\alpha[1,i]} + (\alpha_{i+1} - \alpha_i) \cdot \beta)$



$$\langle \mathbf{r}\rangle_0[x] := \mathrm{DPF}_0(x) + \textstyle\sum_{i=0}^{n-1}(v_0^{x[1,i]} + t_0^i \cdot VCW_{i+1})$$

$$\langle \mathbf{r}\rangle_1[x] := \mathrm{DPF}_1(x) - \textstyle\sum_{i=0}^{n-1}(v_1^{x[1,i]} + t_1^i \cdot VCW_{i+1})$$

- If $x \neq \alpha$, let $x[1,j] = \alpha[1,j]$,

$$\langle \mathbf{r}\rangle_0[x] + \langle \mathbf{r}\rangle_1[x] = \mathrm{DPF}_0(x) + \mathrm{DPF}_1(x) + \textstyle\sum_{i=0}^{n-1}(v_0^i - v_1^i + (t_0^i - t_1^i) \cdot VCW_{i+1})$$

$$= \textstyle\sum_{i=0}^{j}(v_0^i - v_1^i + (t_0^i - t_1^i) \cdot VCW_{i+1}) = \textstyle\sum_{i=0}^{j}(\alpha_{i+1} - \alpha_i) \cdot \beta = \alpha_{j+1} \cdot \beta$$

- If $x = \alpha$, we let $\mathrm{DPF}(\alpha) = -\alpha_n \cdot \beta$

$$\langle \mathbf{r}\rangle_0[x] + \langle \mathbf{r}\rangle_1[x] = -\alpha_n \cdot \beta + \textstyle\sum_{i=0}^{n-1}(v_0^i - v_1^i + (t_0^i - t_1^i) \cdot VCW_{i+1}) = -\alpha_n \cdot \beta + \textstyle\sum_{i=0}^{n-1}(\alpha_{i+1} - \alpha_i) \cdot \beta = 0$$