Constructions of Hidden-Bits Generator for NIZKs

Shen Dong

SJTU

July 3, 2024

Outline

Contents

Hidden-bits Generator

Definition

A Hidden-Bits Generator (HBG) is given by a set of PPT algorithms (Setup, GenBits, Verify) satisfying ***statistical binding and computationally hiding****:*

- $\mathsf{Setup}\left(1^{\lambda}, 1^{k}\right)$ *: Outputs a common reference string* crs.
- $\mathsf{GenBits}(\mathsf{crs})$*: Outputs a triple* $\left(\mathsf{com}, r, \{\pi_i\}_{i\in[k]}\right)$, *where* $r \in \{0,1\}^k$.
- $\mathsf{Verify}(\mathsf{crs}, \mathsf{com}, i, r_i, \pi_i)$: *Outputs accept or reject, where* $i \in [k]$.

Correctness: $\forall k = poly(\lambda)$ and $\forall i \in [k]$:

$$\Pr\left[\mathsf{Verify}(\mathsf{crs}, \mathsf{com}, i, r_i, \pi_i) = \mathsf{accept} \ : \ \begin{array}{cl} \mathsf{crs} & \leftarrow \mathsf{Setup}(1^{\lambda}, 1^{k}) \\ (\mathsf{com}, r, \pi_{[k]}) & \leftarrow \mathsf{GenBits}(\mathsf{crs}) \end{array}\right] = 1.$$

Properties

Succinct Commitment:
$\mathcal{COM}(\lambda)$: ~~set of all valid commitments,~~ contains all possible commitments from $\mathsf{GenBits}$ (and possibly more):

$$\forall \mathsf{com} \notin \mathcal{COM}(\lambda),\ \ \mathsf{Verify}(\mathsf{crs}, \mathsf{com}, \cdot, \cdot) = \text{reject}$$

$\exists\ \delta < 1$ s.t. $|\mathcal{COM}(\lambda)| \leq 2^{k^{\delta}\ \text{poly}(\lambda)}$

as a part of proof in CRS model: for bounding the soundness error in CRS-model NIZK, intuitively this limits a prover's chances for cheating.

Properties

Statistical Binding: $\exists$ (**inefficient**) deterministic algorithm Open outputs r such that for every (potentially unbounded) cheating prover $\widetilde{\mathcal{P}}$:

$$\Pr\left[\begin{array}{ll} & r_i^* \neq r_i \\ \wedge & \mathsf{Verify}(\mathsf{crs}, \mathsf{com}, i, r_i^*, \pi_i) = \mathsf{accept} \end{array} : \begin{array}{rl} \mathsf{crs} & \leftarrow \mathsf{Setup}(1^\lambda, 1^k) \\ (\mathsf{com}, i, r_i^*, \pi_i) & \leftarrow \widetilde{\mathcal{P}}(\mathsf{crs}) \\ r & \leftarrow \mathsf{Open}(1^k, \mathsf{crs}, \mathsf{com}) \end{array}\right] \leq \mathsf{negl}(\lambda).$$

Computationally Hiding: $\forall I \subseteq [k]$, the two following distributions are computationally indistinguishable:

$$(\mathsf{crs}, \mathsf{com}, I, r_I, \pi_I, r_{\bar{I}})$$
$$\overset{c}{\approx}$$
$$(\mathsf{crs}, \mathsf{com}, I, r_I, \pi_I, r'_{\bar{I}}), \ r' \xleftarrow{\$} \{0,1\}^k$$

Designated-Verifier Hidden-bits Generator

Definition

- $\mathsf{Setup}\left(1^{\lambda}, 1^{k}\right)$ *: outputs* $(\mathsf{crs}, \mathsf{td})$, td *trapdoor associated to* crs*;*
- Verify $(\mathsf{crs}, \mathsf{td}, \mathsf{com}, i, r_i, \pi_i\ )$ *takes the trapdoor td as an additional input, and outputs accept or reject;*

Statistical Binding: the cheating prover $\widetilde{\mathcal{P}}$ can make a polynomial number of oracle queries to Verify $(\mathsf{crs}, \mathsf{td}, \cdots)$. $\forall \widetilde{P}$:

$$\Pr\left[\begin{array}{ll} & r_i^* \neq r_i \\ \wedge & \mathsf{Verify}(\mathsf{crs}, \mathsf{td}, \mathsf{com}, i, r_i^*, \pi_i) = \mathsf{accept} \end{array} : \begin{array}{cl} (\mathsf{crs}, \mathsf{td}) & \leftarrow \mathsf{Setup}(1^{\lambda}, 1^{k}) \\ (\mathsf{com}, i, r_i^*, \pi_i) & \leftarrow \widetilde{\mathcal{P}}^{\mathsf{Verify}(\mathsf{crs}, \mathsf{td}, \cdots)}(\mathsf{crs}) \\ r & \leftarrow \mathsf{Open}(1^{k}, \mathsf{crs}, \mathsf{com}) \end{array}\right] \leq \mathsf{negl}(\lambda)$$

Computational Hiding: we require indistinguishability given associated td:

$$(\mathsf{crs}, \mathsf{td}, \mathsf{com}, I, r_I, \pi_I, r_{\bar{I}}) \overset{\mathsf{c}}{\approx} (\mathsf{crs}, \mathsf{td}, \mathsf{com}, I, r_I, \pi_I, r'_{\bar{I}})$$

Construction

Consider the following candidate NIZK $(Setup^{ZK}, \mathcal{P}, \mathcal{V})$ in the CRS model:

1. $Setup^{\mathrm{ZK}}\left(1^\lambda, 1^n\right)$:
$$crs^{\mathrm{BG}} \leftarrow Setup^{\mathrm{BG}}\left(1^\lambda, 1^k\right)$$
$$s \xleftarrow{\$} \{0,1\}^k$$
$$\text{output: } crs = \left(crs^{\mathrm{BG}}, s\right)$$

2. $\mathcal{P}(\mathrm{crs}, x, w)$:
$$\left(com, r^{\mathrm{BG}}, \pi_{[k]}\right) \leftarrow GenBits\left(crs^{\mathrm{BG}}\right)$$
$$r_i = r_i^{\mathrm{BG}} \oplus s_i \forall i \in [k]$$
$$\text{invoke } \left(I \subseteq [k], \pi^{\mathrm{HB}}\right) \leftarrow \mathcal{P}^{\mathrm{HB}}(r, x, w)$$
$$\text{output: } \Pi = \left(I, \pi^{\mathrm{HB}}, com, r_I, \pi_I\right)$$

3. $\mathcal{V}\left(crs, x, \Pi = \left(\left(I, \pi^{\mathrm{HB}}, com, r_I, \pi_I\right)\right)\right)$: $r_i^{\mathrm{BG}} = r_i \oplus s_i,\ \forall i \in [k]$. Accept if $\forall i \in I, Verify\left(crs^{\mathrm{BG}}, com, i, r_i^{\mathrm{BG}}, \pi_i\right)$ accepts, and if $\mathcal{V}^{\mathrm{HB}}\left(I, r_I, x, \pi^{\mathrm{HB}}\right)$ also accepts.

Contents

CDH DV-HBG Construction: Parameters and Setup

Parameters:

1. $(\mathbb{G}, p, g) \leftarrow \mathsf{GroupGen}(1^\lambda)$, p is prime order, g generator.
2. hc is the corresponding Goldreich-Levin hard-core bit.

$\mathsf{Setup}\left(1^\lambda, 1^k\right)$:

1. $(\mathbb{G}, p, g) \leftarrow \mathsf{GroupGen}\left(1^\lambda\right)$.
2. $\forall i \in [k]$, $a_i, b_i \stackrel{\mathbb{S}}{\leftarrow} \mathbb{Z}_p$, $h_i \stackrel{\$}{\leftarrow} \mathbb{G}$ and compute:$f_i = h_i^{a_i} \cdot g^{b_i}$.
3. Random coins γ matching the randomness used by hc $(\cdot)$.

Output:$\left(\mathsf{crs} = \left(\mathbb{G}, \{(h_i, f_i)\}_{i \in [k]}, \gamma\right), \mathsf{td} = \{(a_i, b_i)\}_{i \in [k]}\right)$.

CDH DV-HBG Construction: GenBits and Verify

$\mathsf{GenBits}(\mathsf{crs})$:

1. $y \xleftarrow{\$} \mathbb{Z}_p$, $\forall i \in [k] : t_i = h_i^y$ and $u_i = f_i^y$.
2. Output:

$$\begin{aligned} \mathsf{com} &= s = g^y, \\ \{r_i &= \mathrm{hc}\,(t_i;\gamma)\}_{i\in[k]}\,, \\ \{\pi_i &= (u_i, t_i)\}_{i\in[k]}\,. \end{aligned}$$

$\mathsf{Verify}(\mathsf{crs}, \mathsf{td} = \{(a_i, b_i)\}, \text{com } = s, i, r_i, \pi_i = (u_i, t_i))$:

1. Compute: $\rho_i = t_i^{a_i} \cdot s^{b_i}$.
2. Accept if and only if $\rho_i = u_i$, and $r_i = \mathrm{hc}\,(t_i;\gamma)$.

Validation Proof

1. Completeness:
   - $\rho_i = t_i^{a_i} \cdot s^{b^i} = h_i^{a_i y_i} g^y = (h_i^{a_i} \cdot g^{b_i})^y = f_i^y = u_i$ (certificate is right, from randomness y)
   - and $r_i = hc(t_i; \gamma)$ (hardcore bit is right).
2. Succinctness: $|\mathcal{COM}| = |\mathbb{G}| = p = 2^{\mathrm{poly}(\lambda)}$, independent of k.
3. Computational Hiding:
   - $\{r_i = hc(h_i^y; \gamma)\}_{i \notin I}$ look pseudorandom. Hardcore bit of $h_i^y = g^{x_i y}$ is still computationally unpredictable given $h_i = g^{x_i}$ and $s = g^y$.

Validation Proof

Lemma

$\mathbb{G}$ is a group of prime order p, $h \in \mathbb{G}$, $\forall (s := g^y, t \neq h^y) \in \mathbb{G}^2$, we have that for all $a, b \xleftarrow{\$} \mathbb{Z}_p$: $\big((t^a \cdot s^b), (h^a \cdot g^b)\big) \equiv \mathcal{U}(\mathbb{G} \times \mathbb{G})$ over (a, b).

$h = g^x :\ \big(g^{az+by}, g^{ax+b}\big)$, the exponents are linearly independent, equivalent to guessing an independent random pair.

Statistical Binding:

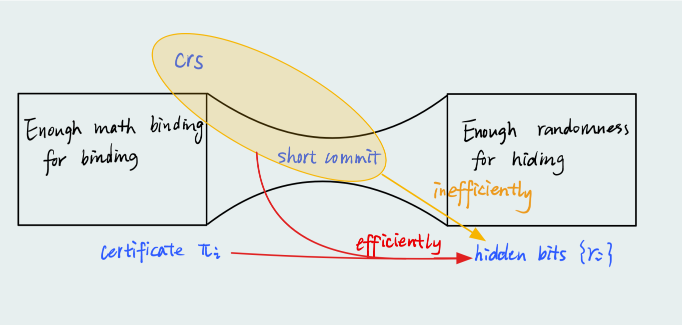
1. $\mathsf{Open}(1^k, \mathsf{crs} = \{\mathbb{G}, \{(h_i, f_i)\}_{i\in[k]}, \gamma\}, \mathsf{com} = g^y)$: traverse the group to find y, computes $\{r_i = hc(h_i^y; \gamma)\}_{i\in[k]}$.
2. Prob that $\tilde{\mathcal{P}}$ outputs $t \neq h_i^y$ but Verify accepts (in j-th attempts among polynomial queries) is

$$\Pr[s^{b_i} \cdot t^{a_i} = (f^i)^y | t_i \neq h_i^y, t^{a_i} \cdot s^{b_i} \text{ uniform over } \mathbb{G} \backslash U] = \frac{1}{|\mathbb{G} \backslash U|} = \text{negl}$$

Union bound also negl.

Contents

What is Non-trivial?



1. Succinctness: commitment cannot be too long, even shorter than k, so the hidden bits must be **pseudorandom**.
2. Open with only crs and com: our short commitment must contain all information in hidden bits, com **functions as (part of) seed for pseudo-randomness in hidden bits**, and perhaps auxiliary info in crs.
3. Two paths: $\{\pi_i\}$ functions like **trapdoor** (or with sk in DV-NIZK).

Tweak Exact-LPN String Commitment

1. Parameters: security parameter λ, $k = \text{poly}(\lambda)$ and $N = O(k^\delta)$ where $0 < \delta < 1$; $n = \Theta(N)$, an idealized $\frac{1}{\sigma}$-stretch PRG.
2. $\mathsf{Setup}(1^\lambda, 1^k)$:$\forall i \in [k], A_i \xleftarrow{\$} \{0,1\}^{N \times n}$, exact weight constant w, PRG,$\mathsf{crs} = (\{A_i\}, w, \text{PRG})$.
3. $\mathsf{GenBits}(\mathsf{crs})$:
   - $seed \xleftarrow{\$} \{0,1\}^{k^\delta}$, $\{r_i\}_{i\in[k]} \leftarrow \text{PRG}(seed)$.
   - $s \xleftarrow{\$} \{0,1\}^{n-1}, e_i \xleftarrow{\$} X_w^n$ s.t. $|e_i| = w$ with exact weight.
   - Output $\left(\mathsf{com} = s, \{\pi_i = A_i \cdot (s \| r_i) + e_i\}_{i\in[k]}, \{r_i\}_{i\in[k]}\right)$.
4. $\mathsf{Verify}(\mathsf{crs}, \mathsf{com}, \mathsf{r_i}, \pi_\mathsf{i})$: accept iff PRG checks valid and

$$|\pi_i \oplus A_i \cdot (\mathsf{com} \| r_i)| = w$$

.

Validation

Lemma

Parameterize $w = 2N\mu$ *s.t.* $\{(A, Ax \oplus e) \mid e \xleftarrow{\$} B_\mu^N\}$ *and* $\{(A, Ax \oplus f) \mid f \xleftarrow{\$} B_\mu^N, |f| \leq w\}$ *are statistically IND, the statistical difference of the two random ensembles is bounded above by* $2e^{-2\mu^2 N}$*.*

Statistical binding:

- Open: guess *seed* inefficiently from com.
- Perfect binding from minimum distance of exact-LPN: Assume exist different $\boldsymbol{m}_i, \boldsymbol{r}_i, i = 1, 2$ for commitment $\boldsymbol{c}$ i.e.

$$\boldsymbol{e}_i = \boldsymbol{c} \oplus \mathbf{A}.\left(\boldsymbol{r}_i \| \boldsymbol{m}_i\right), |e_i| = w, \forall i = 1, 2$$

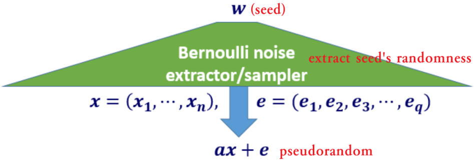
$\therefore \boldsymbol{e}_1 \oplus \boldsymbol{e}_2 = \mathbf{A} \cdot (\boldsymbol{r}_1 \| \boldsymbol{m}_1 \oplus \boldsymbol{r}_2 \| \boldsymbol{m}_2)$ is a codeword of length

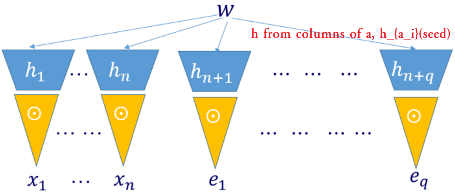
$$\|\boldsymbol{e}_1 \oplus \boldsymbol{e}_2\|_1 \leq \|\boldsymbol{e}_1\|_1 + \|\boldsymbol{e}_2\|_1 \leq 2w$$

contradicting the distance of the code generated by $\mathbf{A}$.

Computationally Hiding: From the PRG

Need a Good PRG for Binding





How to be verifiable for each bit without disclosing others? To hide other bits, the seed must be hidden!

Variant Definition: Short crs

TODO:

1. Why this definition is equivalent with short commitment?
2. Fit LPN construction into this definition?