Bulletproofs+ with double-blinded commitments (DRAFT)

sowle[1]

[1]Zano project, val@zano.org

February 2022

1 Introduction

The original Bulletproofs+[1] uses the following group-based range relation:

$$\{(\boldsymbol{g}, \boldsymbol{h} \in \mathbb{G}^n,\ g, h, V \in \mathbb{G},\ v, \gamma \in \mathbb{Z}_p) : V = g^v h^\gamma \ \wedge\ v \in [0, 2^n - 1]\}$$

which is only compatible with single-blinded commitments in form $V = g^v h^\gamma$. Below we extend Bulletproofs+ protocol for double-blinded commitments using group-based range relation as follows:

$$\{(\boldsymbol{g}, \boldsymbol{h} \in \mathbb{G}^n,\ g, h_1, h_2, V \in \mathbb{G},\ v, \gamma_1, \gamma_2 \in \mathbb{Z}_p) : V = g^v h_1^{\gamma_1} h_2^{\gamma_2} \ \wedge\ v \in [0, 2^n - 1]\}$$

Instead of a single blinding mask generator h we introduce two generators $h_1, h_2 \in \mathbb{G}$ and corresponding blinding masks $\gamma_1, \gamma_2 \in \mathbb{Z}_p$. It is assumed that discrete logarithm relation is unknown for all used generators.

2 Zero knowledge argument for weighted inner product relation

WIP agrument protocol with double-blinded commitments support is shown on Fig. 2.1. We provide the security statement for the proposed updated zk-WIP protocol in Theorem 1. The proof of Theorem 1 mostly corresponds to the original proof in Bulletproofs+ paper[1]. For reader's convenience all changes are highlighted.

Theorem 1. *Let y be a constant in $\mathbb{Z}_p^*$. The zero-knowledge argument for WIP presented in Fig.2.1 has perfect completeness, perfect honest verifier zero-knowledge and computational witness-extended emulation.*

Original Bulletproofs+	Modified Bulletproofs+
$\boxed{\text{zk-WIP} \underset{y^n}{\longrightarrow} (\boldsymbol{g}, \boldsymbol{h}, g, h, P; \boldsymbol{a}, \boldsymbol{b}, \alpha)}$	$\boxed{\text{zk-WIP} \underset{y^n}{\longrightarrow} (\boldsymbol{g}, \boldsymbol{h}, g, h_1, h_2, P; \boldsymbol{a}, \boldsymbol{b}, \alpha_1, \alpha_2)}$
Relation:	
$\{(\boldsymbol{g}, \boldsymbol{h} \in \mathbb{G}^n, g, h, P \in \mathbb{G}; \boldsymbol{a}, \boldsymbol{b} \in \mathbb{Z}_p^n, \alpha \in \mathbb{Z}_p) : P = \boldsymbol{g}^{\boldsymbol{a}} \boldsymbol{h}^{\boldsymbol{b}} g^{\boldsymbol{a} \odot_y \boldsymbol{b}} h^{\alpha}\}$	$\{(\boldsymbol{g}, \boldsymbol{h} \in \mathbb{G}^n, g, h_1, h_2, P \in \mathbb{G}; \boldsymbol{a}, \boldsymbol{b} \in \mathbb{Z}_p^n, \alpha_1, \alpha_2 \in \mathbb{Z}_p) : P = \boldsymbol{g}^{\boldsymbol{a}} \boldsymbol{h}^{\boldsymbol{b}} g^{\boldsymbol{a} \odot_y \boldsymbol{b}} h_1^{\alpha_2} h_2^{\alpha_2}\}$
$\mathcal{P}$'s input:	
$(\boldsymbol{g}, \boldsymbol{h}, g, h, P; \boldsymbol{a}, \boldsymbol{b}, \alpha)$	$(\boldsymbol{g}, \boldsymbol{h}, g, h_1, h_2, P; \boldsymbol{a}, \boldsymbol{b}, \alpha_1, \alpha_2)$
$\mathcal{V}$'s input:	
$(\boldsymbol{g}, \boldsymbol{h}, g, h, P)$	$(\boldsymbol{g}, \boldsymbol{h}, g, h_1, h_2, P)$
$\mathcal{P}$'s output: none	
$\mathcal{V}$'s output: Accept or Reject	
$\boxed{\text{If } n = 1:}$	
$\boxed{\mathcal{P}}$: $r, s, \delta, \eta \xleftarrow{\$} \mathbb{Z}_p$ and computes: $A = \boldsymbol{g}^r \boldsymbol{h}^s g^{r \odot_y \boldsymbol{b} + s \odot_y \boldsymbol{a}} h^{\delta} \in \mathbb{G}$ $B = g^{r \odot_y s} h^{\eta} \in \mathbb{G}$	$\boxed{\mathcal{P}}$: $r, s, \delta_1, \delta_2, \eta_1, \eta_2 \xleftarrow{\$} \mathbb{Z}_p$ and computes: $A = \boldsymbol{g}^r \boldsymbol{h}^s g^{r \odot_y \boldsymbol{b} + s \odot_y \boldsymbol{a}} h_1^{\delta_1} h_2^{\delta_2} \in \mathbb{G}$ $B = g^{r \odot_y s} h_1^{\eta_1} h_2^{\eta_2} \in \mathbb{G}$
$\boxed{\mathcal{P} \to \mathcal{V}}$: A, B $\boxed{\mathcal{V}}$: $e \xleftarrow{\$} \mathbb{Z}_p^*$ $\boxed{\mathcal{P} \leftarrow \mathcal{V}}$: e $\boxed{\mathcal{P}}$: computes: $r' = r + \boldsymbol{a} \cdot e \in \mathbb{Z}_p$ $s' = s + \boldsymbol{b} \cdot e \in \mathbb{Z}_p$	
$\delta' = \eta + \delta \cdot e + \alpha \cdot e^2 \in \mathbb{Z}_p$	$\delta_1' = \eta_1 + \delta_1 \cdot e + \alpha_1 \cdot e^2 \in \mathbb{Z}_p$ $\delta_2' = \eta_2 + \delta_2 \cdot e + \alpha_2 \cdot e^2 \in \mathbb{Z}_p$
$\boxed{\mathcal{P} \to \mathcal{V}}$: r', s', δ'	$\boxed{\mathcal{P} \to \mathcal{V}}$: $r', s', \delta_1', \delta_2'$
$\boxed{\mathcal{V}}$: outputs *Accept* iff the following holds: $P^{e^2} A^e B = \boldsymbol{g}^{r' \cdot e} \boldsymbol{h}^{s' \cdot e} g^{r' \odot_y s'} h^{\delta'} \in \mathbb{G}$	$\boxed{\mathcal{V}}$: outputs *Accept* iff the following holds: $P^{e^2} A^e B = \boldsymbol{g}^{r' \cdot e} \boldsymbol{h}^{s' \cdot e} g^{r' \odot_y s'} h_1^{\delta_1'} h_2^{\delta_2'} \in \mathbb{G}$
$\boxed{\text{else } (n > 1):}$ Let $\widehat{n} = \frac{n}{2},\ \boldsymbol{a} = (\boldsymbol{a}_1, \boldsymbol{a}_2),\ \boldsymbol{b} = (\boldsymbol{b}_1, \boldsymbol{b}_2),\ \boldsymbol{g} = (\boldsymbol{g}_1, \boldsymbol{g}_2),\ \boldsymbol{h} = (\boldsymbol{h}_1, \boldsymbol{h}_2)$ (where $\boldsymbol{a}_i, \boldsymbol{b}_i, \boldsymbol{g}_i$ and $\boldsymbol{h}_i$ are of the same length $\widehat{n}$)	
$\boxed{\mathcal{P}}$: $d_L, d_R \xleftarrow{\$} \mathbb{Z}_p$ and computes:	$\boxed{\mathcal{P}}$: $d_L', d_R', d_L'', d_R'' \xleftarrow{\$} \mathbb{Z}_p$ and computes:
$c_L = \boldsymbol{a_1} \odot_y \boldsymbol{b_2} \in \mathbb{Z}_p$ $c_R = (y^{\widehat{n}} \cdot \boldsymbol{a_2}) \odot_y \boldsymbol{b_1} \in \mathbb{Z}_p$	
$L = \boldsymbol{g}_2^{(y^{-\widehat{n}} \cdot \boldsymbol{a}_1)} \boldsymbol{h}_1^{\boldsymbol{b}_2} g^{c_L} h^{d_L} \in \mathbb{G}$ $R = \boldsymbol{g}_1^{(y^{\widehat{n}} \cdot \boldsymbol{a}_2)} \boldsymbol{h}_2^{\boldsymbol{b}_1} g^{c_R} h^{d_R} \in \mathbb{G}$	$L = \boldsymbol{g}_2^{(y^{-\widehat{n}} \cdot \boldsymbol{a}_1)} \boldsymbol{h}_1^{\boldsymbol{b}_2} g^{c_L} h_1^{d_L'} h_2^{d_L''} \in \mathbb{G}$ $R = \boldsymbol{g}_1^{(y^{\widehat{n}} \cdot \boldsymbol{a}_2)} \boldsymbol{h}_2^{\boldsymbol{b}_1} g^{c_R} h_1^{d_R'} h_2^{d_R''} \in \mathbb{G}$
$\boxed{\mathcal{P} \to \mathcal{V}}$: L, R $\boxed{\mathcal{V}}$: $e \xleftarrow{\$} \mathbb{Z}_p^*$ $\boxed{\mathcal{P} \leftarrow \mathcal{V}}$: e $\boxed{\mathcal{P} \text{ and } \mathcal{V}}$: compute: $\widehat{\boldsymbol{g}} = \boldsymbol{g}_1^{e^{-1}} \circ \boldsymbol{g}_2^{e \cdot y^{-\widehat{n}}} \in \mathbb{G}^{\widehat{n}}$ $\widehat{\boldsymbol{h}} = \boldsymbol{h}_1^{e} \circ \boldsymbol{h}_2^{-e} \in \mathbb{G}^{\widehat{n}}$ $\widehat{P} = L^{e^2} P R^{e^{-2}} \in \mathbb{G}$ $\boxed{\mathcal{P}}$: computes: $\widehat{\boldsymbol{a}} = \boldsymbol{a}_1 \cdot e + (\boldsymbol{a}_2 \cdot y^{\widehat{n}}) \cdot e^{-1} \in \mathbb{Z}_p^{\widehat{n}}$ $\widehat{\boldsymbol{b}} = \boldsymbol{b}_1 \cdot e^{-1} + \boldsymbol{b}_2 \cdot e \in \mathbb{Z}_p^{\widehat{n}}$	
$\widehat{\alpha} = d_L \cdot e^2 + \alpha + d_R \cdot e^{-2} \in \mathbb{Z}_p$	$\widehat{\alpha_1} = d_L' \cdot e^2 + \alpha_1 + d_R' \cdot e^{-2} \in \mathbb{Z}_p$ $\widehat{\alpha_2} = d_L'' \cdot e^2 + \alpha_2 + d_R'' \cdot e^{-2} \in \mathbb{Z}_p$
$\boxed{\mathcal{P} \text{ and } \mathcal{V}}$: run: $\text{zk-WIP} \underset{y^n}{\longrightarrow} (\widehat{\boldsymbol{g}}, \widehat{\boldsymbol{h}}, g, h, \widehat{P}; \widehat{\boldsymbol{a}}, \widehat{\boldsymbol{b}}, \widehat{\alpha})$	$\boxed{\mathcal{P} \text{ and } \mathcal{V}}$: run: $\text{zk-WIP} \underset{y^n}{\longrightarrow} (\widehat{\boldsymbol{g}}, \widehat{\boldsymbol{h}}, g, h_1, h_2, \widehat{P}; \widehat{\boldsymbol{a}}, \widehat{\boldsymbol{b}}, \widehat{\alpha_1}, \widehat{\alpha_2})$

Fig. 2.1. Zero Knowledge Argument for WIP relation

Proof. (perfect completeness) We show that the WIP argument has perfect completeness. First, we assume that $P = \boldsymbol{g}^{\boldsymbol{a}}\boldsymbol{h}^{\boldsymbol{b}}g^{\boldsymbol{a}\odot_y\boldsymbol{b}}h_1^{\alpha_1}h_2^{\alpha_2}$ and show that the case $n = 1$ satisfies the perfect completeness. That is, we show that the verification equation holds. It is sufficient to show that the corresponding five equations with bases $\boldsymbol{g}, \boldsymbol{h}, g, h_1, h_2$, respectively, hold.

$$\begin{aligned}
\boldsymbol{a}e^2 + re = (\boldsymbol{a}e + r)e = r'e &\quad \in \mathbb{Z}_p \\
\boldsymbol{b}e^2 + se = (\boldsymbol{b}e + s)e = s'e &\quad \in \mathbb{Z}_p \\
\boldsymbol{a}y\boldsymbol{b}e^2 + (ry\boldsymbol{b} + sy\boldsymbol{a})e + rys = (\boldsymbol{b}e + s)(\boldsymbol{a}ye + ry) = r' \odot_y s' &\quad \in \mathbb{Z}_p \\
\alpha_1 e^2 + \delta_1 e + \eta_1 = \delta_1' &\quad \in \mathbb{Z}_p \\
\alpha_2 e^2 + \delta_2 e + \eta_2 = \delta_2' &\quad \in \mathbb{Z}_p
\end{aligned}$$

From the above five equalities, the perfect completeness for the case $n = 1$ is proven.

Next, we move to the case $n > 1$. For every end of recursive step, if the parameters $(\widehat{\boldsymbol{g}}, \widehat{\boldsymbol{h}}, g, h_1, h_2\widehat{P}; \widehat{\boldsymbol{a}}, \widehat{\boldsymbol{b}}, \widehat{\alpha}_1, \widehat{\alpha}_2)$ that will be used for the next call satisfy the relation $\widehat{P} = \widehat{\boldsymbol{g}}^{\widehat{\boldsymbol{a}}}\widehat{\boldsymbol{h}}^{\widehat{\boldsymbol{b}}}g^{\widehat{\boldsymbol{a}}\odot_y\widehat{\boldsymbol{b}}}h_1^{\widehat{\alpha}_1}h_2^{\widehat{\alpha}_2}$ when $P = \boldsymbol{g}^{\boldsymbol{a}}\boldsymbol{h}^{\boldsymbol{b}}g^{\boldsymbol{a}\odot_y\boldsymbol{b}}h_1^{\alpha_1}h_2^{\alpha_2}$ then we can be sure that the protocol will drive to end up with a correct input for the last step of $n = 1$. Therefore we show that if the input P is of the form $\boldsymbol{g}^{\boldsymbol{a}}\boldsymbol{h}^{\boldsymbol{b}}g^{\boldsymbol{a}\odot_y\boldsymbol{b}}h_1^{\alpha_1}h_2^{\alpha_2}$ and $\widehat{P}, \widehat{\boldsymbol{g}}, \widehat{\boldsymbol{h}}, \widehat{\boldsymbol{a}}, \widehat{\boldsymbol{b}}$ are computed as the protocol, then $\widehat{P}$ has the desired form $\widehat{\boldsymbol{g}}^{\widehat{\boldsymbol{a}}}\widehat{\boldsymbol{h}}^{\widehat{\boldsymbol{b}}}g^{\widehat{\boldsymbol{a}}\odot_y\widehat{\boldsymbol{b}}}h_1^{\widehat{\alpha}_1}h_2^{\widehat{\alpha}_2}$.

Let $P, \widehat{P}, \widehat{\boldsymbol{g}}, \widehat{\boldsymbol{h}}, \widehat{\boldsymbol{a}}, \widehat{\boldsymbol{b}}$ be the form in the protocol description for the case $n > 1$. If L and R are computed as the description of the protocol, then $\widehat{P}$ is computed by $\widehat{P} = L^{e^2}PR^{e^{-2}}$ and we can write $\widehat{P}$ according to the corresponding bases.

$$\begin{aligned}
\boldsymbol{a}_1 + y^{\widehat{n}}\boldsymbol{a}_2e^{-2} = (\boldsymbol{a}_1e + y^{\widehat{n}}\boldsymbol{a}_2e^{-1})e^{-1} = \widehat{\boldsymbol{a}}e^{-1} &\quad \in \mathbb{Z}_p \\
y^{-\widehat{n}}\boldsymbol{a}_1e^2 + \boldsymbol{a}_2 = (\boldsymbol{a}_1e + y^{\widehat{n}}\boldsymbol{a}_2e^{-1})ey^{-\widehat{n}} = \widehat{\boldsymbol{a}}ey^{-\widehat{n}} &\quad \in \mathbb{Z}_p \\
\boldsymbol{b}_2e^2 + \boldsymbol{b}_1 = (\boldsymbol{b}_2e + \boldsymbol{b}_1e^{-1})e = \widehat{\boldsymbol{b}}e &\quad \in \mathbb{Z}_p \\
\boldsymbol{b}_2 + \boldsymbol{b}_1e^{-2} = (\boldsymbol{b}_2e + \boldsymbol{b}_1e^{-1})e^{-1} = \widehat{\boldsymbol{b}}e^{-1} &\quad \in \mathbb{Z}_p \\
c_Le^2 + \boldsymbol{a}\odot_y\boldsymbol{b} + c_Re^{-2} = \boldsymbol{a}_1\odot_y\boldsymbol{b}_2e^2 + \boldsymbol{a}\odot_y\boldsymbol{b} + y^{\widehat{n}}\boldsymbol{a}_2\odot_y\boldsymbol{b}_1e^{-2} &\quad \in \mathbb{Z}_p \\
d_L'e^2 + \alpha_1 + d_R'e^{-2} = \widehat{\alpha}_1 &\quad \in \mathbb{Z}_p \\
d_L''e^2 + \alpha_2 + d_R''e^{-2} = \widehat{\alpha}_1 &\quad \in \mathbb{Z}_p
\end{aligned}$$

Furthermore, from the definition of $\widehat{\boldsymbol{a}}$ and $\widehat{\boldsymbol{b}}$, we see that

$$\begin{aligned}
&\widehat{\boldsymbol{a}}\odot_y\widehat{\boldsymbol{b}} \\
&= (\boldsymbol{a}_1e + (\boldsymbol{a}_2y^{\widehat{n}})e^{-1})\odot_y(\boldsymbol{b}_1e^{-1} + \boldsymbol{b}_2e) \\
&= \boldsymbol{a}_1\odot_y\boldsymbol{b}_1 + \boldsymbol{a}_1\odot_y\boldsymbol{b}_2e^2 + (y^{\widehat{n}}\boldsymbol{a}_2)\odot_y\boldsymbol{b}_1e^{-2} + (y^{\widehat{n}}\boldsymbol{a}_2)\odot_y\boldsymbol{b}_2 \\
&= \boldsymbol{a}_1\odot_y\boldsymbol{b}_2e^2 + \boldsymbol{a}\odot_y\boldsymbol{b} + (y^{\widehat{n}}\boldsymbol{a}_2)\odot_y\boldsymbol{b}_1e^{-2} \quad \in \mathbb{Z}_p,
\end{aligned}$$

which is equal to the g-base exponent of $\widehat{P}$. Using the above observation, we can easily check that the following holds.

$$\begin{aligned}
\widehat{P} &= \boldsymbol{g}_1^{\widehat{\boldsymbol{a}}e^{-1}}\boldsymbol{g}_2^{\widehat{\boldsymbol{a}}ey^{-\widehat{n}}}\boldsymbol{h}_1^{\widehat{\boldsymbol{b}}e}\boldsymbol{h}_2^{\widehat{\boldsymbol{b}}e^{-1}}g^{\widehat{\boldsymbol{a}}\odot_y\widehat{\boldsymbol{b}}}h_1^{\widehat{\alpha}_1}h_2^{\widehat{\alpha}_2} \\
&= (\boldsymbol{g}_1^{e^{-1}}\boldsymbol{g}_2^{ey^{-\widehat{n}}})^{\widehat{\boldsymbol{a}}}(\boldsymbol{h}_1^e\boldsymbol{h}_2^{e^{-1}})^{\widehat{\boldsymbol{b}}}g^{\widehat{\boldsymbol{a}}\odot_y\widehat{\boldsymbol{b}}}h_1^{\widehat{\alpha}_1}h_2^{\widehat{\alpha}_2} \\
&= \widehat{\boldsymbol{g}}^{\widehat{\boldsymbol{a}}}\widehat{\boldsymbol{h}}^{\widehat{\boldsymbol{b}}}g^{\widehat{\boldsymbol{a}}\odot_y\widehat{\boldsymbol{b}}}h_1^{\widehat{\alpha}_1}h_2^{\widehat{\alpha}_2} \quad \in \mathbb{G}
\end{aligned}$$

This completes the proof of the perfect completeness.

(perfect SHVZK) To prove the argument system is perfect special honest verifier zero-knowledge, we construct a simulator, given only the public input, it outputs a simulated transcript that is identical to the valid transcript produced by the prover and verifier in the real interaction.

We first describe our simulator construction, and then analyze it. The simulator begins with taking the statement and the randomness ρ of the verifier as input. Using ρ, the simulator can generates all challenges whose distribution is identical to that of the real argument. We describe how the simulator generates a non-challenge part. For each $n > 1$, the simulator chooses two random group elements and set those L_n, R_n. For the case of $n = 1$, the simulator chooses $A_s \xleftarrow{\$} \mathbb{G}$ and $r'_s, s'_s, \delta'_{1,s}, \delta'_{2,s} \xleftarrow{\$} \mathbb{Z}_p$ at random and computes

$$B_s = (P^{e^2} A^e \boldsymbol{g}^{-r'_s \cdot e} \boldsymbol{h}^{-s'_s \cdot e} g^{-r'_s \odot_y s'_s} h_1^{-\delta'_{1,s}} h_2^{-\delta'_{2,s}})^{-1} \quad \in \mathbb{G}.$$

Next, we analyze the distribution of the simulated transcript for the non-challenge part $(\{(L_i, R_i)\}_i, A_s, B_s, r'_s, s'_s, \delta'_{1,s}, \delta'_{2,s})$. In the protocol description, $\forall i, (L_i, R_i)$ distributes uniformly and independently due to blinding factors $d'_{L_i}, d'_{R_i}, d''_{L_i}$, and d''_{R_i} and all (L_i, R_i)'s contribute to generate P used in the case $n = 1$. The simulator generates uniformly (L_i, R_i)'s at random, so that its' distribution is identical to that of the real argument. From now, we analyze the distribution of $(A_s, B_s, r'_s, s'_s, \delta'_{1,s}, \delta'_{2,s})$ for given P in the case $n = 1$.

Before analyzing the simulated transcript $(A_s, B_s, r'_s, s'_s, \delta'_{1,s}, \delta'_{2,s})$, we first analyze the real transcript $(A, B, r', s', \delta'_1, \delta'_2)$ and then show two distributions are identical.

Here, we focus on $(A, r', s', \delta'_1, \delta'_2)$ and claim that it is uniformly distributed in $\mathbb{G} \times \mathbb{Z}_p^4$ when $(r, s, \delta_1, \delta_2, \eta_1, \eta_2)$ in uniformly distributed in $\mathbb{Z}_p^6$. To this end, it is sufficient to prove the following claim.

Claim: *There exists a one-to-one correspondence between* $(r, s, \delta_1, \delta_2, \eta_1, \eta_2)$ *and* $(A, r', s', \delta'_1, \delta'_2)$.

Proof: First, consider the following function mapping from $(r, s, \delta_1, \delta_2, \eta_1, \eta_2)$ to $(A, r', s', \delta'_1, \delta'_2)$.

$$\begin{pmatrix} A \\ r' \\ s' \\ \delta'_1 \\ \delta'_2 \end{pmatrix} = \begin{pmatrix} \boldsymbol{g}^r \boldsymbol{h}^s g^{r \odot_y \boldsymbol{b} + s \odot_y \boldsymbol{a}} h_1^{\delta_1} h_2^{\delta_2} \\ r + \boldsymbol{a} \cdot e \\ s + \boldsymbol{b} \cdot e \\ \eta_1 + \delta_1 \cdot e + \alpha_1 \cdot e^2 \\ \eta_2 + \delta_2 \cdot e + \alpha_2 \cdot e^2 \end{pmatrix} \quad \in \mathbb{G} \times \mathbb{Z}_P^4$$

Assume that there is another tuple $(\tilde{r}, \tilde{s}, \tilde{\delta_1}, \tilde{\delta_2}, \tilde{\eta_1}, \tilde{\eta_2}) \neq (r, s, \delta_1, \delta_2, \eta_1, \eta_2)$ which image via the above function is also $(A, r', s', \delta'_1, \delta'_2)$. Then, comparing two function values we obtain

$$\begin{pmatrix} \boldsymbol{g}^r \boldsymbol{h}^s g^{r \odot_y \boldsymbol{b} + s \odot_y \boldsymbol{a}} h_1^{\delta_1} h_2^{\delta_2} \\ r + \boldsymbol{a} \cdot e \\ s + \boldsymbol{b} \cdot e \\ \eta_1 + \delta_1 \cdot e + \alpha_1 \cdot e^2 \\ \eta_2 + \delta_2 \cdot e + \alpha_2 \cdot e^2 \end{pmatrix} = \begin{pmatrix} \boldsymbol{g}^{\tilde{r}} \boldsymbol{h}^{\tilde{s}} g^{\tilde{r} \odot_y \boldsymbol{b} + \tilde{s} \odot_y \boldsymbol{a}} h_1^{\tilde{\delta_1}} h_2^{\tilde{\delta_2}} \\ \tilde{r} + \boldsymbol{a} \cdot e \\ \tilde{s} + \boldsymbol{b} \cdot e \\ \tilde{\eta_1} + \tilde{\delta_1} \cdot e + \alpha_1 \cdot e^2 \\ \tilde{\eta_2} + \tilde{\delta_2} \cdot e + \alpha_2 \cdot e^2 \end{pmatrix}$$

From the second and third rows we get $r = \tilde{r}$ and $s = \tilde{s}$. Using it, from the first one we obtain:

$$\begin{aligned} \boldsymbol{g}^r \boldsymbol{h}^s g^{r \odot_y \boldsymbol{b} + s \odot_y \boldsymbol{a}} h_1^{\delta_1} h_2^{\delta_2} \cdot (\boldsymbol{g}^{\tilde{r}} \boldsymbol{h}^{\tilde{s}} g^{\tilde{r} \odot_y \boldsymbol{b} + \tilde{s} \odot_y \boldsymbol{a}} h_1^{\tilde{\delta_1}} h_2^{\tilde{\delta_2}})^{-1} = 1_{\mathbb{G}} \Rightarrow \\ h_1^{\delta_1 - \tilde{\delta_1}} h_2^{\delta_2 - \tilde{\delta_2}} = 1_{\mathbb{G}} \end{aligned}$$

Using the discrete logarithm relation assumption for h_1 and h_2, we get

$$\delta_1 = \tilde{\delta_1}$$
$$\delta_2 = \tilde{\delta_2}$$

Using that, from the last two equations for η_1 and η_2 we finally obtain $\eta_1 = \tilde{\eta_1}$ and $\eta_2 = \tilde{\eta_2}$. Thus:

$$(\tilde{r}, \tilde{s}, \tilde{\delta_1}, \tilde{\delta_2}, \tilde{\eta_1}, \tilde{\eta_2}) = (r, s, \delta_1, \delta_2, \eta_1, \eta_2)$$

This contradiction concludes the proof of the claim. □

In the generation of the real transcript $(A, B, r', s', \delta'_1, \delta'_2)$, only six random integer r, s, $\delta_1, \delta_2, \eta_1$ and η_2 are used. Therefore, the above result implies that the distribution of $(A, B, r', s', \delta'_1, \delta'_2)$ is identical to the distribution that $(A, r', s', \delta'_1, \delta'_2)$ is uniformly distributed and B is uniquely defined by the others and the verification equation. In fact, the latter process is exactly same as the simulated transcript. Therefore, the simulated transcript is identical to that of the real transcript for given P in the case $n = 1$. Overall, we complete the proof of the perfect special honest verifier zero-knowledge.

(witness-extended emulation) For witness extended emulation, we construct an expected polynomial time extractor χ that extracts a witness using a $poly(\lambda)$-bounded tree of accepting transcripts, so that to meet the requirements of the general forking lemma. Consider the case $n = 1$. At the first move, the prover sends A and B to verifier. By rewinding the oracle $\langle \mathcal{P}^*, \mathcal{V} \rangle$ four times with five distinct challenges e_1, e_2, e_3, e_4, and e_5 while using the same A and B, the extractor obtains five tuples $(r'_i, s'_i, \delta'_{1,i}, \delta'_{2,i})$ satisfying the following verification equation.

$$P^{e_i^2} A^{e_i} B = \boldsymbol{g}^{r'_i \cdot e_i} \boldsymbol{h}^{s'_i \cdot e_i} g^{r'_i \odot_y s'_i} h_1^{\delta'_{1,i}} h_2^{\delta'_{2,i}} \quad \text{for } i = 1, \ldots, 5 \tag{1}$$

Using the first three challenges and the corresponding valid responses, we can interpret the exponents as a product of a 3×3 Vandermonde matrix (which is invertible in $\mathbb{Z}_p^{3\times 3}$ since all e_i's are distinct):

$$\begin{pmatrix} e_1^2 & e_1 & 1 \\ e_2^2 & e_2 & 1 \\ e_3^2 & e_3 & 1 \end{pmatrix} \begin{pmatrix} a_P & b_P & c_P & d_P & f_P \\ a_A & b_A & c_A & d_A & f_A \\ a_B & b_B & c_B & d_B & f_B \end{pmatrix} = \begin{pmatrix} r'_1 e & s'_1 e & r'_1 \odot_y s'_1 & \delta'_{1,1} & \delta'_{2,1} \\ r'_2 e & s'_2 e & r'_2 \odot_y s'_2 & \delta'_{1,2} & \delta'_{2,2} \\ r'_2 e & s'_2 e & r'_2 \odot_y s'_2 & \delta'_{1,2} & \delta'_{2,2} \end{pmatrix}$$

The other exponents in the right hand side of Eq. (1) are public as well. Thus, from those three challenges and responses, we can obtain the exponents $a_P, b_P, c_P, d_P, f_P, a_A, b_A, c_A, d_A, f_A, a_B, b_B, c_B, d_B, f_B$ such that

$$P = \boldsymbol{g}^{a_P} \boldsymbol{h}^{b_P} g^{c_P} h_1^{d_P} h_2^{f_P}$$
$$A = \boldsymbol{g}^{a_A} \boldsymbol{h}^{b_A} g^{c_A} h_1^{d_A} h_2^{f_A}$$
$$B = \boldsymbol{g}^{a_B} \boldsymbol{h}^{b_B} g^{c_B} h_1^{d_B} h_2^{f_B}$$

Using the above three equations and the verification equation, we obtain for each $e_i \in \{e_1, e_2, e_3, e_4, e_5\}$,

$$\boldsymbol{g}^{r'_i e_i - a_P e_i^2 - a_A e_i - a_B} \boldsymbol{h}^{s'_i e_i - b_P e_i^2 - b_A e_i - b_B}$$
$$\cdot\, g^{r'_i \odot_y s'_i - c_P e_i^2 - c_A e_i - c_B} h_1^{\delta'_{1,i} - d_P e_i^2 - d_A e_i - d_B} h_2^{\delta'_{2,i} - f_P e_i^2 - f_A e_i - f_B} = 1_{\mathbb{G}}.$$

Thus, under the discrete logarithm relation assumption, we have five equations of exponents accord-

ing to the bases $\boldsymbol{g}, \boldsymbol{h}, g, h_1, h_2$,

$$\begin{aligned} r'_i e_i - a_P e_i^2 - a_A e_i - a_B &= 0 \\ s'_i e_i - b_P e_i^2 - b_A e_i - b_B &= 0 \\ r'_i \odot_y s'_i - c_P e_i^2 - c_A e_i - c_B &= 0 \\ \delta'_{1,i} - d_P e_i^2 - d_A e_i - d_B &= 0 \\ \delta'_{2,i} - f_P e_i^2 - f_A e_i - f_B &= 0 \end{aligned}$$

and, equivalently,

$$r'_i = a_P e_i + a_A + a_B e_i^{-1} \tag{2}$$
$$s'_i = b_P e_i + b_A + b_B e_i^{-1} \tag{3}$$
$$r'_i \odot_y s'_i = c_P e_i^2 + c_A e_i + c_B \tag{4}$$
$$\delta'_{1,i} = d_P e_i^2 + d_A e_i + d_B$$
$$\delta'_{2,i} = f_P e_i^2 + f_A e_i + f_B$$

By elimination r'_i and s'_i from Eq. (2), Eq. (3), and Eq. (4), we have for $i \in \{1, \ldots, 5\}$

$$\begin{aligned} & a_P \odot_y b_P \cdot e_i^2 + (a_P \odot_y b_A + b_P \odot_y a_A) \cdot e_i \\ & + (a_P \odot_y b_B + b_P \odot_y a_B + a_A \odot_y b_A) + (a_A \odot_y b_B + b_A \odot_y a_B) \cdot e_i^{-1} \\ & + a_B \odot_y b_B \cdot e_i^{-2} \\ & = c_P e_i^2 + c_A e_i + c_B \quad \in \mathbb{Z}_p \end{aligned} \tag{5}$$

This equation can be considered as an inner-product with $(e_i^2, e_i, 1, e_i^{-1}, e_i^{-2})$ and constants vector. Since Eq. (5) holds for all five distinct challenges $e_i \in \{e_1, \ldots, e_5\}$ and so $(e_i^2, e_i, 1, e_i^{-1}, e_i^{-2})$'s are linearly independent, each coefficient in the left hand side of Eq. (5) must be equal to the corresponding coefficient in the right hand side of Eq. (5). As we intended, the extractor either extracts a witness (a_P, b_P) satisfying $a_P \odot_y b_P = c_P$, or a discrete logarithm relation between the generators.

Next, we move to the case $n > 1$. We prove the case $n > 1$ recursively. That is, we construct an extractor χ_{2k} for the case $n = 2k$ using an extractor χ_k and let χ_1 be the extractor χ we constructed for the case $n = 1$. We start with input $(\boldsymbol{g}, \boldsymbol{h}, g, h_2, h_2, P)$ for the case $n = 2k$. Assume that we have the extractor χ_k for the case $n = k$. The extractor χ_{2k} runs the prover to get L and R. At this point, the extractor χ_{2k} rewinds the oracle four times, uses four distinct challenges e_i for $i = 1, \ldots, 4$, and sets

$$\widehat{\boldsymbol{g}}_i = \boldsymbol{g}_1^{e_i^{-1}} \circ \boldsymbol{g}_2^{e_i \cdot y^{-k}}, \quad \widehat{\boldsymbol{h}}_i = \boldsymbol{h}_1^{e_i} \circ \boldsymbol{h}_2^{e^{-1}}, \quad \widehat{P}_i = L^{e_i^2} P R^{e_i^{-2}} \quad \in \mathbb{G} \text{ for } i = 1, \ldots, 4$$

Then, for each i, it feeds $(\widehat{\boldsymbol{g}}_i, \widehat{\boldsymbol{h}}_i, g, h_1, h_2, \widehat{P}_i)$ to χ_k and obtain the corresponding witness $\widehat{\boldsymbol{a}}_i, \widehat{\boldsymbol{b}}_i$, $\widehat{\alpha}_{1,i}$, and $\widehat{\alpha}_{2,i}$ that satisfy

$$L^{e_i^2} P R^{e_i^{-2}} = (\boldsymbol{g}^{e_i^{-1}} \circ \boldsymbol{g}_2^{e_i \cdot y^{-k}})^{\widehat{\boldsymbol{a}}_i} (\boldsymbol{h}_1^{e_i} \circ \boldsymbol{h}_2^{e_i^{-1}})^{\widehat{\boldsymbol{b}}_i} g^{\widehat{\boldsymbol{a}}_i \odot_y \widehat{\boldsymbol{b}}_i} h_1^{\widehat{\alpha}_{1,i}} h_2^{\widehat{\alpha}_{2,i}}, \quad i \in [1, 4] \tag{6}$$

For the first three challenges e_1, e_2, e_3, $(e_i^2, 1, e_i^{-2})$'s are linearly independent and so compose of a 3×3 invertible matrix in $\mathbb{Z}_p^{3 \times 3}$. We can see that all exponents are constants known to the extractor. Thus, by applying the elementary linear algebra in the public exponent of the first three equations of

Eq. (6), we can find the exponents $\boldsymbol{a}_P, \boldsymbol{b}_P, c_P, d_P, f_P, \boldsymbol{a}_L, \boldsymbol{b}_L, c_L, d_L, f_L, \boldsymbol{a}_R, \boldsymbol{b}_R, c_R, d_R, f_R$ satisfying

$$\begin{aligned} P &= \boldsymbol{g}^{\boldsymbol{a}_P}\boldsymbol{h}^{\boldsymbol{b}_P} g^{c_P} h_1^{d_P} h_2^{f_P} && \in \mathbb{G}\,, \\ L &= \boldsymbol{g}^{\boldsymbol{a}_L}\boldsymbol{h}^{\boldsymbol{b}_L} g^{c_L} h_1^{d_L} h_2^{f_L} && \in \mathbb{G}\,, \\ R &= \boldsymbol{g}^{\boldsymbol{a}_R}\boldsymbol{h}^{\boldsymbol{b}_R} g^{c_R} h_1^{d_R} h_2^{f_R} && \in \mathbb{G}\,. \end{aligned}$$

From now, we prove that those exponents satisfy the desired relation $c_P = \boldsymbol{a}_P \odot_y \boldsymbol{b}_P$. Putting the above representations of P, L, R into Eq. (6) for each i, we have the following equations with bases $\boldsymbol{g}, \boldsymbol{h}, g, h_1, h_2$ under the discrete logarithm relation assumption.

$$\boldsymbol{g}^{\boldsymbol{a}_L e_i^2}\boldsymbol{g}^{\boldsymbol{a}_P}\boldsymbol{g}^{\boldsymbol{a}_R e_i^{-2}} = (\boldsymbol{g}_1^{e_i^{-1}} \circ \boldsymbol{g}_2^{e_i \cdot y^{-k}})^{\widehat{\boldsymbol{a}}_i} \tag{7}$$

$$\boldsymbol{h}^{\boldsymbol{b}_L e_i^2}\boldsymbol{h}^{\boldsymbol{b}_P}\boldsymbol{h}^{\boldsymbol{b}_R e_i^{-2}} = (\boldsymbol{h}_1^{e_i} \circ \boldsymbol{h}_2^{e_i^{-1}})^{\widehat{\boldsymbol{b}}_i} \tag{8}$$

$$g^{c_L e_i^2} g^{c_P} g^{c_R e_i^{-2}} = g^{\widehat{\boldsymbol{a}}_i \odot_y \widehat{\boldsymbol{b}}_i} \tag{9}$$

$$h_1^{d_L e_i^2} h_1^{d_P} h_1^{d_R e_i^{-2}} = h_1^{\widehat{\alpha}_{1,i}}$$

$$h_2^{f_L e_i^2} h_2^{f_P} h_2^{f_R e_i^{-2}} = h_2^{\widehat{\alpha}_{2,i}}$$

That is, Eq. (6) is separated into the above five equations according to the bases $\boldsymbol{g}, \boldsymbol{h}, g, h_1, h_2$. If we find exponents satisfying Eq. (6) but not the above five equations, it directly implies a non-trivial relation between the generators and so break the discrete logarithm assumption. We use the above five equations to prove $\boldsymbol{a}_P \odot_y \boldsymbol{b}_P = c_P$. To this end, we first find a relation between $\widehat{\boldsymbol{a}}_i$ and $\boldsymbol{a}_P$ from Eq. (7), second find another relation between $\widehat{\boldsymbol{b}}_i$ and $\boldsymbol{b}_P$ from Eq. (8), and then finally use Eq. (9) containing $c_P, \widehat{\boldsymbol{a}}_i, \widehat{\boldsymbol{b}}_i$ variables in order to show the desired relation between $c_P, \boldsymbol{a}_P$, and $\boldsymbol{b}_P$.

First, we show that relation between $\widehat{\boldsymbol{a}}_i$ and $\boldsymbol{a}_P$ from Eq. (7). By the discrete logarithm assumption, it is infeasible to find relation between $\boldsymbol{g}_1$ and $\boldsymbol{g}_2$, so that Eq. (7) induces two equations with the base $\boldsymbol{g}_1$ and $\boldsymbol{g}_2$, which are equivalent to the following equations.

$$\begin{aligned} \boldsymbol{a}_{L,1} e_i^2 + \boldsymbol{a}_{P,1} + \boldsymbol{a}_{R,1} e_i^{-2} &= e_i^{-1} \widehat{\boldsymbol{a}}_i \\ \boldsymbol{a}_{L,2} e_i^2 + \boldsymbol{a}_{P,2} + \boldsymbol{a}_{R,2} e_i^{-2} &= y^{-k} e_i \widehat{\boldsymbol{a}}_i\,, \end{aligned}$$

where $\boldsymbol{a}_P = (\boldsymbol{a}_{P,1}, \boldsymbol{a}_{P,2})$, $\boldsymbol{a}_L = (\boldsymbol{a}_{L,1}, \boldsymbol{a}_{L,2})$, $\boldsymbol{a}_R = (\boldsymbol{a}_{R,1}, \boldsymbol{a}_{R,2}) \in \mathbb{Z}_p^k \times \mathbb{Z}_p^k$. By eliminating $\widehat{\boldsymbol{a}}_i$ from the above two equations, we obtain

$$\boldsymbol{a}_{L,1} e_i^3 + \boldsymbol{a}_{P,1} e_i + \boldsymbol{a}_{R,1} e_i^{-1} = \boldsymbol{a}_{L,2} y^k e_i + \boldsymbol{a}_{P,2} y^k e_i^{-1} + \boldsymbol{a}_{R,2} y^k e_i^{-3} \tag{10}$$

Eq. (10) holds for all four challenges $e_1, \ldots, e_4$ and there are four variable terms $e_i, e_i^3, e_i^{-1}, e_i^{-3}$. This implies that the following must holds.

$$\begin{aligned} \boldsymbol{a}_{L,1} &= 0 && \in \mathbb{Z}_p^k \\ \boldsymbol{a}_{P,1} &= \boldsymbol{a}_{L,2} y^k && \in \mathbb{Z}_p^k \\ \boldsymbol{a}_{R,1} &= \boldsymbol{a}_{P,2} y^k && \in \mathbb{Z}_p^k \\ \boldsymbol{a}_{R,2} &= 0 && \in \mathbb{Z}_p^k \end{aligned}$$

Using the above result with Eq. (7), we obtain that the exponent of the base $\boldsymbol{g}_1$ in Eq. (7) is

$$\boldsymbol{a}_{P,1} + \boldsymbol{a}_{P,2} y^k e_i^{-2} = e_i^{-1} \widehat{\boldsymbol{a}}_i \,,$$

so that we have a relation between $\widehat{\boldsymbol{a}}_i$ and $\boldsymbol{a}_P$,

$$\widehat{\boldsymbol{a}}_i = \boldsymbol{a}_{P,1} e_i + \boldsymbol{a}_{P,2} y^k e_i^{-1} \tag{11}$$

Second, we show that relation between $\widehat{\boldsymbol{b}}_i$ and $\boldsymbol{b}_P$ from Eq. (8). Under the discrete logarithm assumption, we extract the exponent of the base $\boldsymbol{h}_1$ and $\boldsymbol{h}_2$ from Eq. (8).

$$\begin{aligned}
\boldsymbol{b}_{L,1} e_i^2 + \boldsymbol{b}_{P,1} + \boldsymbol{b}_{R,1} e_i^{-2} &= e_i \widehat{\boldsymbol{b}}_i \\
\boldsymbol{b}_{L,2} e_i^2 + \boldsymbol{b}_{P,2} + \boldsymbol{b}_{R,2} e_i^{-2} &= e_i^{-1} \widehat{\boldsymbol{b}}_i
\end{aligned}$$

where $\boldsymbol{b}_P = (\boldsymbol{b}_{P,1}, \boldsymbol{b}_{P,2}), \boldsymbol{b}_L = (\boldsymbol{b}_{L,1}, \boldsymbol{b}_{L,2}), \boldsymbol{b}_R = (\boldsymbol{b}_{R,1}, \boldsymbol{b}_{R,2}) \in \mathbb{Z}_p^k \times \mathbb{Z}_p^k$. By eliminating $\widehat{\boldsymbol{b}}_i$ from the above two equations, we obtain

$$\boldsymbol{b}_{L,1} \cdot e_i + \boldsymbol{b}_{P,1} \cdot e_i^{-1} + \boldsymbol{b}_{R,1} \cdot e_i^{-3} = \boldsymbol{b}_{L,2} y^k \cdot e_i^3 + \boldsymbol{b}_{P,2} \cdot e_i + \boldsymbol{b}_{R,2} \cdot e_i^{-1} \tag{12}$$

Eq. (12) holds for all four challenges $e_1, \ldots e_4$ and there are four variable terms $e_i, e_i^3, e_i^{-1}, e_i^{-3}$. This implies that the following must holds.

$$\begin{aligned}
\boldsymbol{b}_{L,1} &= \boldsymbol{b}_{P,2} && \in \mathbb{Z}_p^k \\
\boldsymbol{b}_{P,1} &= \boldsymbol{b}_{R,2} && \in \mathbb{Z}_p^k \\
\boldsymbol{b}_{R,1} &= 0 && \in \mathbb{Z}_p^k \\
\boldsymbol{b}_{L,2} &= 0 && \in \mathbb{Z}_p^k
\end{aligned}$$

Using the above result with Eq. (8), we obtain that the exponent of the base $\boldsymbol{h}_1$ in Eq. (8) is

$$\boldsymbol{b}_{P,2} e_i^2 + \boldsymbol{b}_{P,1} = e_i \widehat{\boldsymbol{b}}_i \,,$$

so that we have a relation between $\widehat{\boldsymbol{b}}_i$ and $\boldsymbol{b}_P$,

$$\widehat{\boldsymbol{b}}_i = \boldsymbol{b}_{P,2} e_i + \boldsymbol{b}_{P,1} e_i^{-1} \tag{13}$$

Finally, we use Eq. (9) in order to show relation between $c_P, \boldsymbol{a}_P$, and $\boldsymbol{b}_P$. Taking the WIP $\odot_y$ on Eq. (11) and Eq. (13), we have

$$\begin{aligned}
&\widehat{\boldsymbol{a}}_i \odot_y \widehat{\boldsymbol{b}}_i \\
&= (\boldsymbol{a}_{P,1} \odot_y \boldsymbol{b}_{P,2}) e_i^2 + (\boldsymbol{a}_{P,1} \odot_y \boldsymbol{b}_{P,1} + \boldsymbol{a}_{P,2} \odot_y \boldsymbol{b}_{P,2} \cdot y^k) \\
&\quad + (\boldsymbol{a}_{P,2} \odot_y \boldsymbol{b}_{P,1} \cdot y^k) e_i^{-2}
\end{aligned}$$

Combining this result with Eq. (9), we have

$$\left(\boldsymbol{a}_{P,1} \odot_y \boldsymbol{b}_{P,2} - c_L,\ \boldsymbol{a}_{P,1} \odot_y \boldsymbol{b}_{P,1} + \boldsymbol{a}_{P,2} \odot_y \boldsymbol{b}_{P,2} y^k - c_P,\ \boldsymbol{a}_{P,2} \odot_y \boldsymbol{b}_{P,1} y^k - c_R\right) \cdot \begin{pmatrix} e_i^2 \\ 1 \\ e_i^{-2} \end{pmatrix} = 0$$

The above equation holds for three distinct challenges $e_1, \ldots, e_3$. Since the vectors $(e_i^2, 1, e_i^{-2})$'s are linear independent, this implies that the following must holds.

$$\boldsymbol{a}_P \odot_y \boldsymbol{b}_P = \boldsymbol{a}_{P,1} \odot_y \boldsymbol{b}_{P,1} + \boldsymbol{a}_{P,2} \odot_y \boldsymbol{b}_{P,2} \cdot y^k = c_P$$

For each recursive step, the extractor χ_{2k} uses 4 transcripts and χ_1 uses 5 transcripts, so that the final extractor χ_n uses $5 \cdot 4^{\log_2(n)}$ transcripts in total and this runs in expected polynomial time in λ since n is polynomial in λ. Then, by the general forking lemma, we conclude that the proposed WIP argument system has computational witness extended emulation. □

References

[1] Heewon Chung et al. *Bulletproofs+: Shorter Proofs for Privacy-Enhanced Distributed Ledger*. https://eprint.iacr.org/2020/735. 2020.