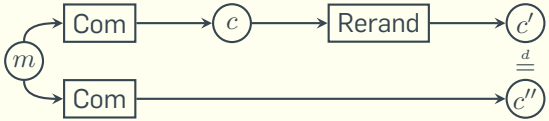
On Publicly-Accountable Zero-Knowledge and Small Shuffle Arguments

Nils Fleischhacker and Mark Simkin

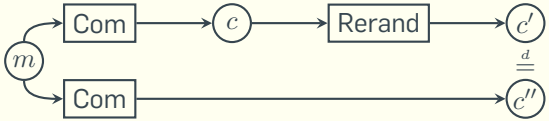
May 13, 2021

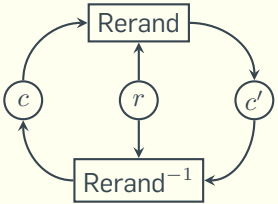
Shuffling Rerandomizable Commitments

Shuffling Rerandomizable Commitments

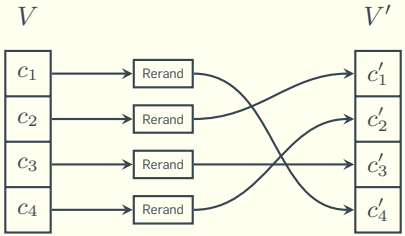


Shuffling Rerandomizable Commitments



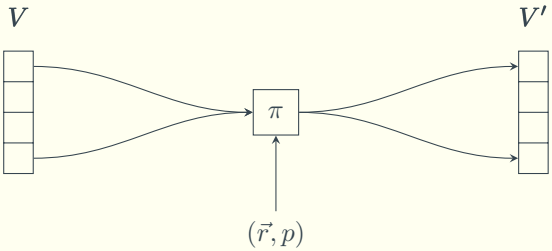


Shuffling Rerandomizable Commitments

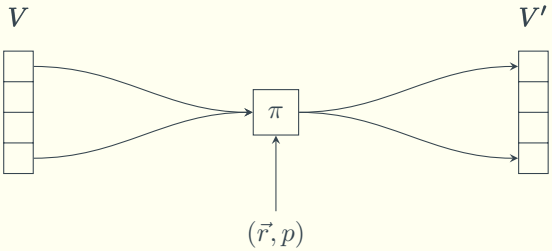


- V' is a shuffle of V iff there exist r_1, r_2, r_3, r_4 and a permutation p, such that $c'_{p(i)} = \mathsf{Rerand}(c_i, r_i)$.
- We also write this as $V' := \pi(V, \vec{r}, p)$.

A Simple Shuffle Argument



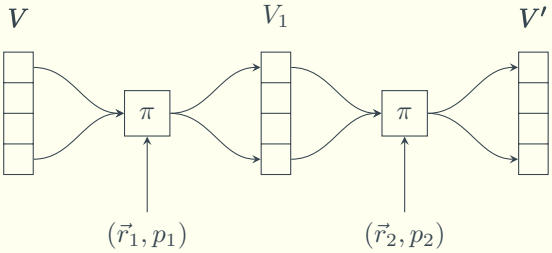
A Simple Shuffle Argument



Idea: Split $(\vec{r}, p)$ into $(\vec{r_1}, p_1)$ and $(\vec{r_2}, p_2)$ such that

$$V' = \pi(\pi(\vec{r_1}, p_1), \vec{r_2}, p_2).$$

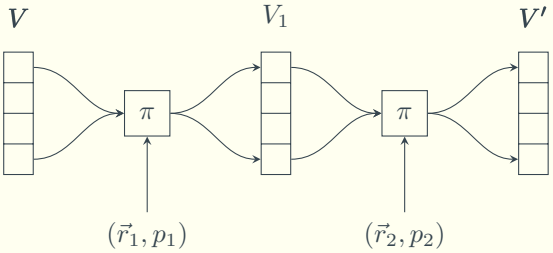
A Simple Shuffle Argument

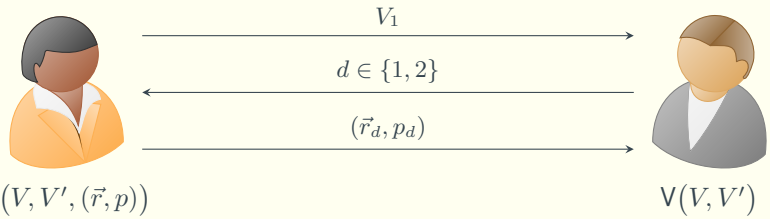


Idea: Split $(\vec{r}, p)$ into $(\vec{r_1}, p_1)$ and $(\vec{r_2}, p_2)$ such that

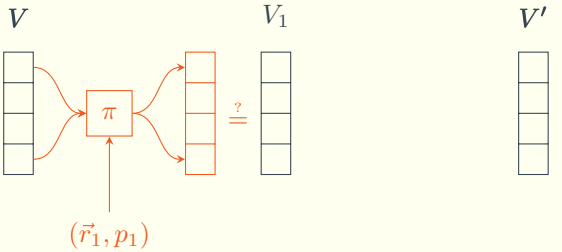
$$V' = \pi(\pi(\vec{r_1}, p_1), \vec{r_2}, p_2).$$

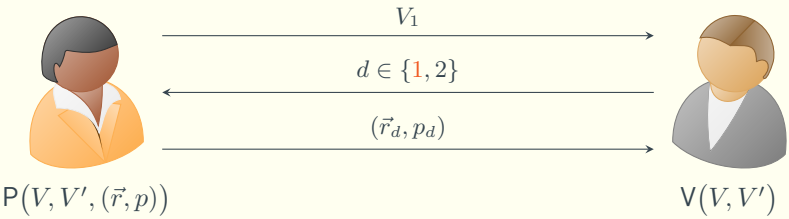
A Simple Shuffle Argument



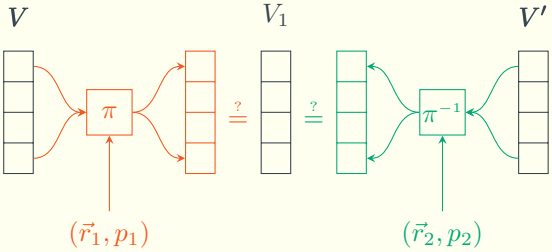


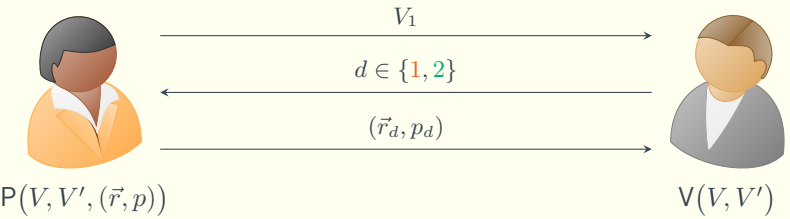
A Simple Shuffle Argument



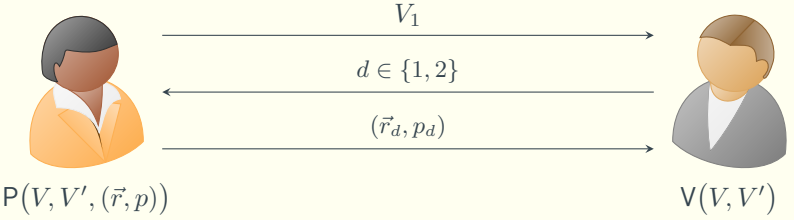


A Simple Shuffle Argument

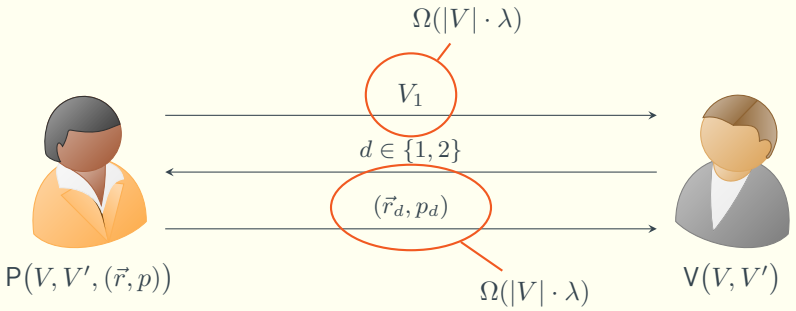




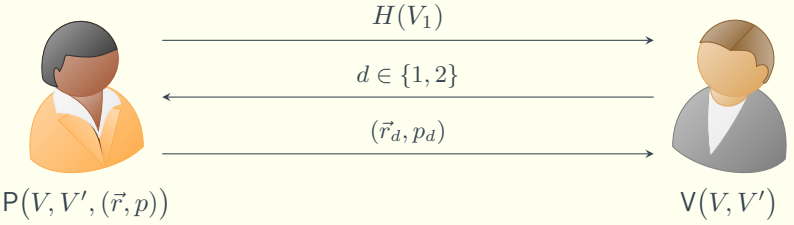
A Simple Shuffle Argument



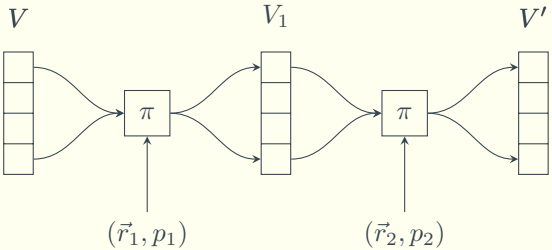
A Simple Shuffle Argument

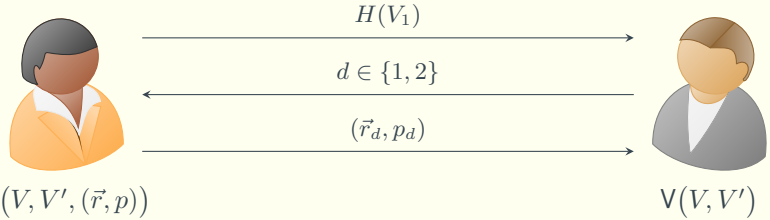


A Simple Shuffle Argument

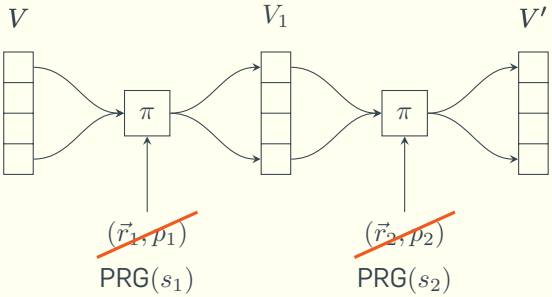


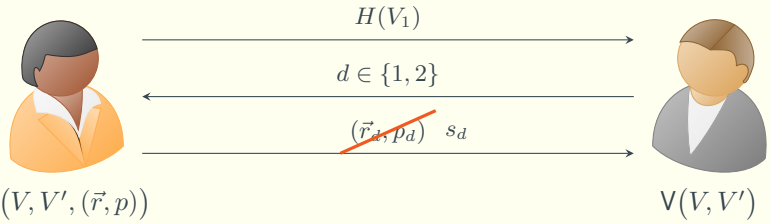
Minimizing Communication Cost





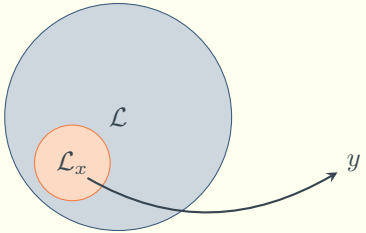
Minimizing Communication Cost



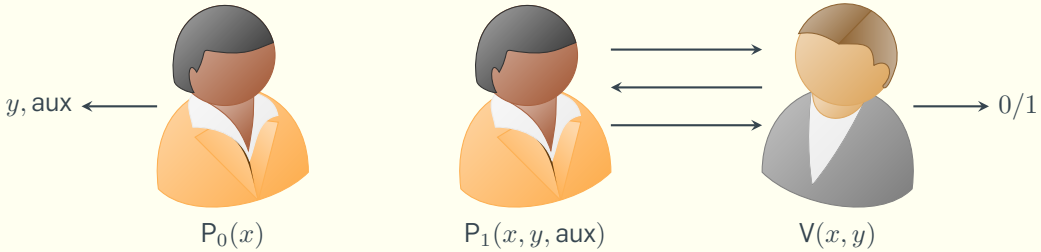


Partially Fixable Languages

- Let $\mathcal{L} \subseteq X \times Y$ be an NP language.
- For any $x \in X$, let $\mathcal{L}_x = \{(x, y) \mid y \in Y \wedge (x, y) \in \mathcal{L}\}$.
- $\mathcal{L}$ is partially fixable if there exists an efficient uniform sampler for all $\mathcal{L}_x \neq \emptyset$.



Arguments for Partially Fixed Statements



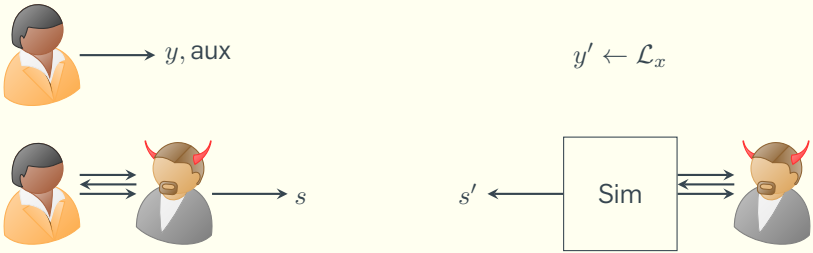
Completeness: For any $x \in X$ with $\mathcal{L}_x \neq \emptyset$ it holds that

$$\Pr[(y, \mathsf{aux}) \leftarrow \mathsf{P}_0(x); b \leftarrow \langle \mathsf{P}_1(x, y, \mathsf{aux}), \mathsf{V}(x, y) \rangle : (x, y) \in \mathcal{L} \wedge b = 1] = 1.$$

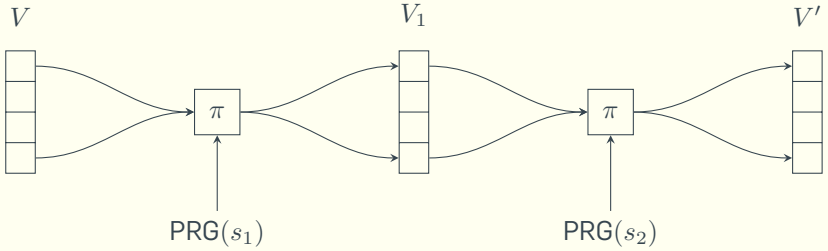
Soundness: For any PPT prover P^* and any $(x, y) \notin \mathcal{L}$ it holds that

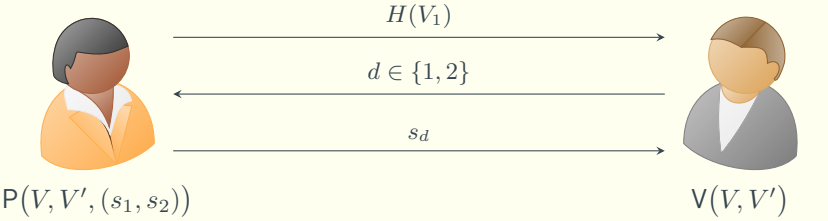
$$\Pr[b \leftarrow \langle \mathsf{P}^*(1^n), \mathsf{V}(x, y) \rangle : b = 1] \leq \epsilon + \mathsf{negl}(n).$$

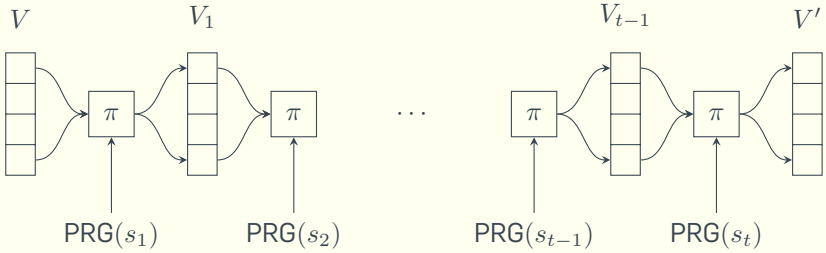
Zero-Knowledge Arguments for Partially Fixed Statements

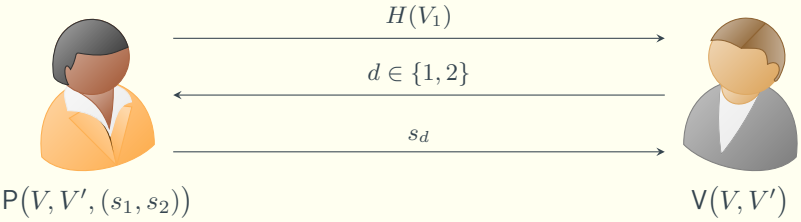


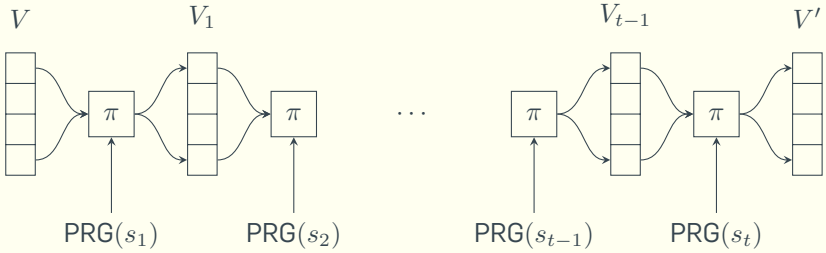
$$(x, y, s) \approx_c (x, y', s')$$

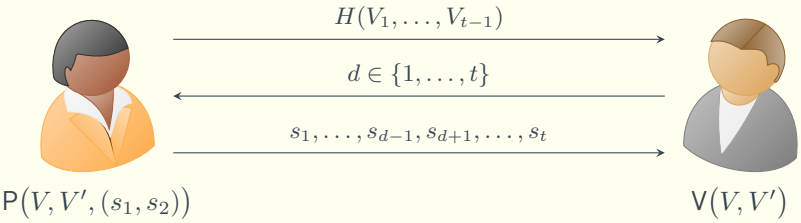

V
V_1
V'
π
π
$\mathsf{PRG}(s_1)$
$\mathsf{PRG}(s_2)$

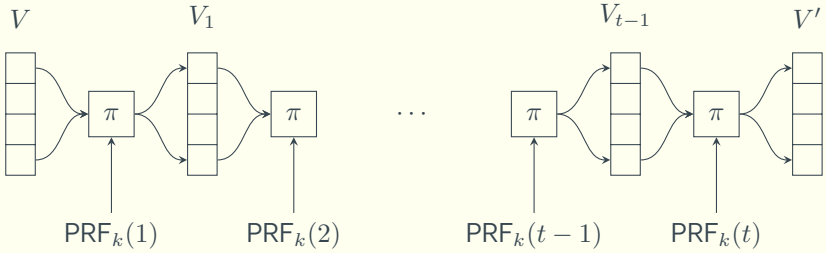

$H(V_1)$
$d \in \{1,2\}$
s_d
$\mathsf{P}(V, V', (s_1, s_2))$
$\mathsf{V}(V, V')$

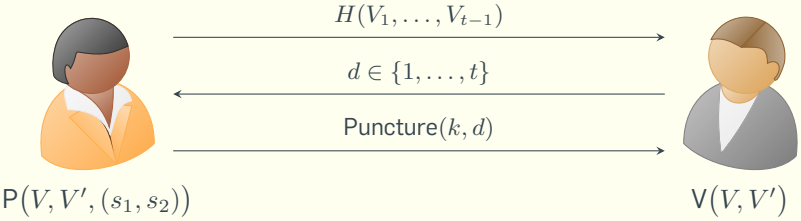

V
V_1
V_{t-1}
V'
π
π
$\cdots$
π
π
$\mathsf{PRG}(s_1)$
$\mathsf{PRG}(s_2)$
$\mathsf{PRG}(s_{t-1})$
$\mathsf{PRG}(s_t)$


$H(V_1)$
$d \in \{1,2\}$
s_d
$\mathsf{P}(V, V', (s_1, s_2))$
$\mathsf{V}(V, V')$


V
V_1
V_{t-1}
V'
π
π
$\cdots$
π
π
$\mathsf{PRG}(s_1)$
$\mathsf{PRG}(s_2)$
$\mathsf{PRG}(s_{t-1})$
$\mathsf{PRG}(s_t)$


$H(V_1, \ldots, V_{t-1})$
$d \in \{1, \ldots, t\}$
$s_1, \ldots, s_{d-1}, s_{d+1}, \ldots, s_t$
$\mathsf{P}(V, V', (s_1, s_2))$
$\mathsf{V}(V, V')$


V
V_1
V_{t-1}
V'
π
π
$\cdots$
π
π
$\mathsf{PRF}_k(1)$
$\mathsf{PRF}_k(2)$
$\mathsf{PRF}_k(t-1)$
$\mathsf{PRF}_k(t)$


$H(V_1, \ldots, V_{t-1})$
$d \in \{1, \ldots, t\}$
$\mathsf{Puncture}(k, d)$
$\mathsf{P}(V, V', (s_1, s_2))$
$\mathsf{V}(V, V')$

The Issues

Only HVZK

Constant soundness error

Interactive

The Issues

Only HVZK

Constant soundness error

Interactive

Dealing with HVZK

Theorem

Let (P, V) *be a three-move public-coin honest verifier ZK argument for language* $\mathcal{L}$ *and let* $\mathcal{C}$ *be the associated challenge space. If* $|\mathcal{C}| \leq \mathsf{poly}(n)$ *then* (P, V) *is also ZK against malicious verifiers.*

Dealing with HVZK

Theorem

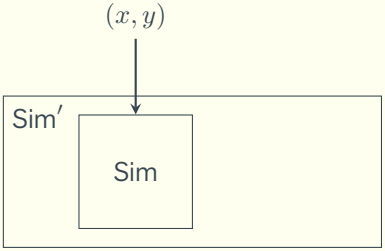
Let (P, V) *be a three-move public-coin honest verifier ZK argument for language* $\mathcal{L}$ *and let* $\mathcal{C}$ *be the associated challenge space. If* $|\mathcal{C}| \leq \mathsf{poly}(n)$ *then* (P, V) *is also ZK against malicious verifiers.*

Sim

Dealing with HVZK

Theorem

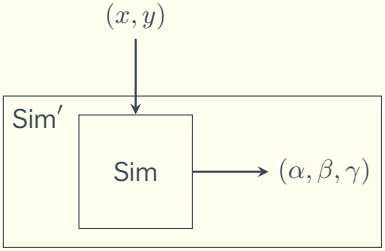
Let (P, V) *be a three-move public-coin honest verifier ZK argument for language* $\mathcal{L}$ *and let* $\mathcal{C}$ *be the associated challenge space. If* $|\mathcal{C}| \leq \mathsf{poly}(n)$ *then* (P, V) *is also ZK against malicious verifiers.*



Dealing with HVZK

Theorem

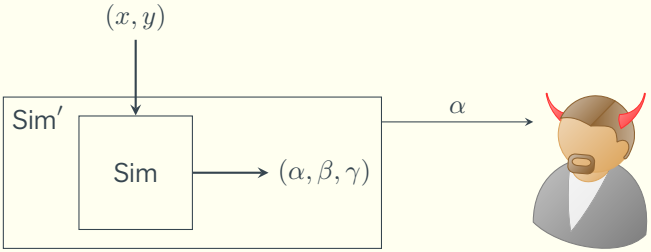
Let (P, V) *be a three-move public-coin honest verifier ZK argument for language* $\mathcal{L}$ *and let* $\mathcal{C}$ *be the associated challenge space. If* $|\mathcal{C}| \leq \mathsf{poly}(n)$ *then* (P, V) *is also ZK against malicious verifiers.*



Dealing with HVZK

Theorem

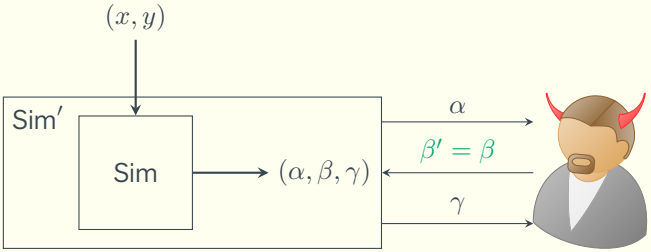
Let (P, V) *be a three-move public-coin honest verifier ZK argument for language* $\mathcal{L}$ *and let* $\mathcal{C}$ *be the associated challenge space. If* $|\mathcal{C}| \leq \mathsf{poly}(n)$ *then* (P, V) *is also ZK against malicious verifiers.*



Dealing with HVZK

Theorem

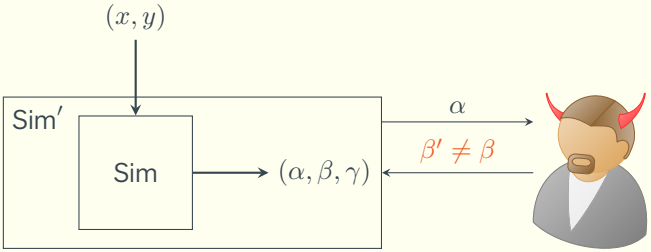
Let (P, V) *be a three-move public-coin honest verifier ZK argument for language* $\mathcal{L}$ *and let* $\mathcal{C}$ *be the associated challenge space. If* $|\mathcal{C}| \leq \mathsf{poly}(n)$ *then* (P, V) *is also ZK against malicious verifiers.*



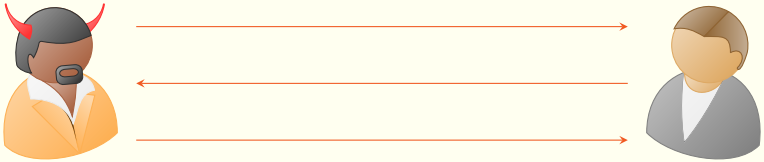
Dealing with HVZK

Theorem

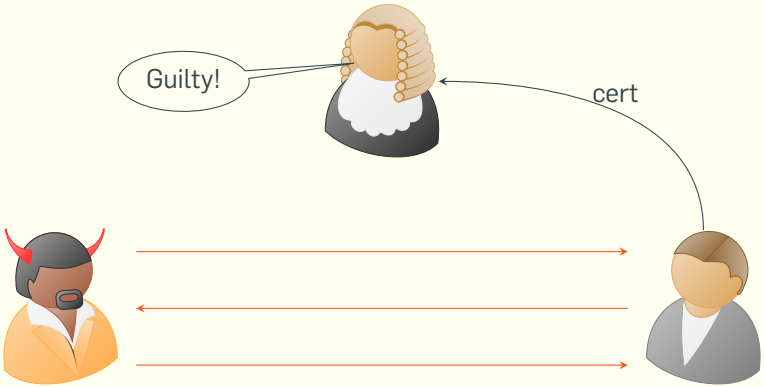
Let (P, V) *be a three-move public-coin honest verifier ZK argument for language* $\mathcal{L}$ *and let* $\mathcal{C}$ *be the associated challenge space. If* $|\mathcal{C}| \leq \mathsf{poly}(n)$ *then* (P, V) *is also ZK against malicious verifiers.*



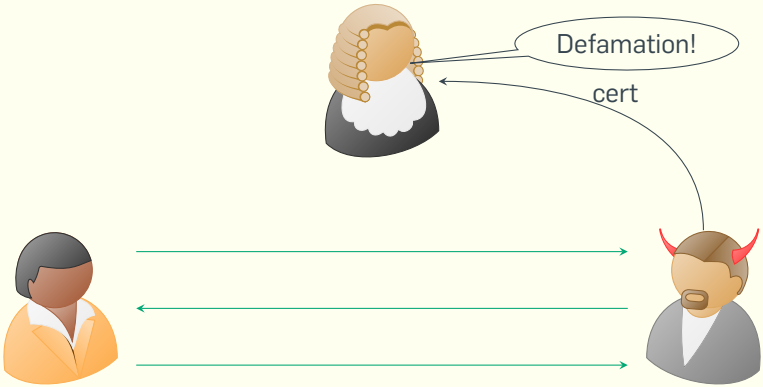
Mitigating the Problems of Constant Soundness Errors



Mitigating the Problems of Constant Soundness Errors



Mitigating the Problems of Constant Soundness Errors



Publicly-Accountable Zero-Knowledge Arguments

Accountability: For any $(x, y) \notin \mathcal{L}$ and any PPT P^* such that

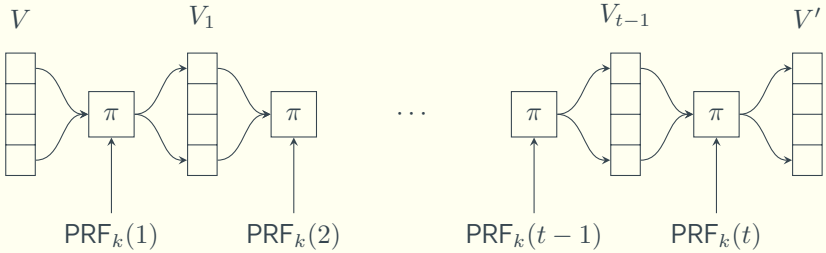
$$\Pr[\mathsf{crs} \leftarrow \mathsf{Setup}(1^n); b \leftarrow \langle \mathsf{P}^*(\mathsf{crs}), \mathsf{V}(\mathsf{crs}, x, y)\rangle : b = 1] \geq \delta\epsilon,$$

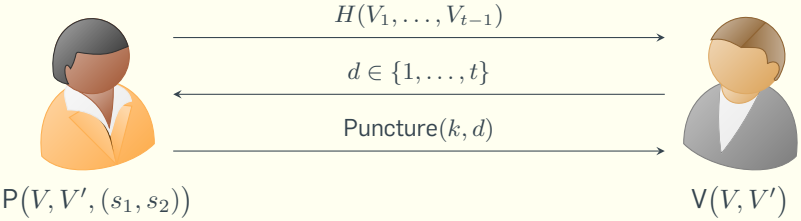
it holds that

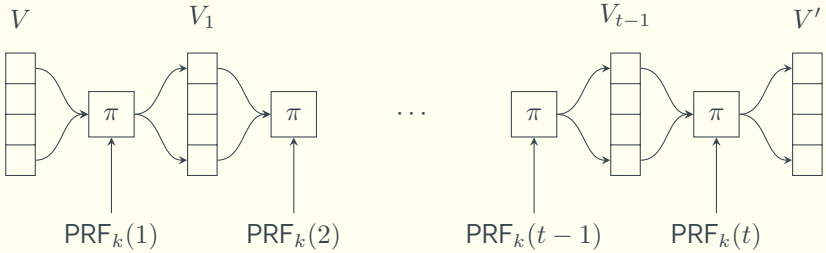
$$\begin{aligned}&\Pr[\mathsf{crs} \leftarrow \mathsf{Setup}(1^n); \mathsf{cert} \leftarrow \langle \mathsf{P}^*(\mathsf{crs}), \mathsf{V}(\mathsf{crs}, x, y)\rangle : \mathsf{J}(\mathsf{crs}, \mathsf{pk}, \mathsf{cert}) = 1] \\ \geq&\delta(1-\epsilon) - \mathsf{negl}(n).\end{aligned}$$

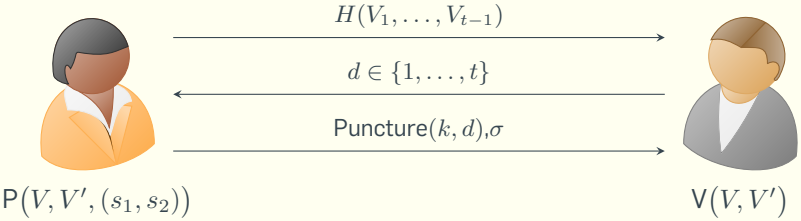
Defamation-Freeness: For any $x \in X$ and any malicious PPT verifier V^*, it holds that

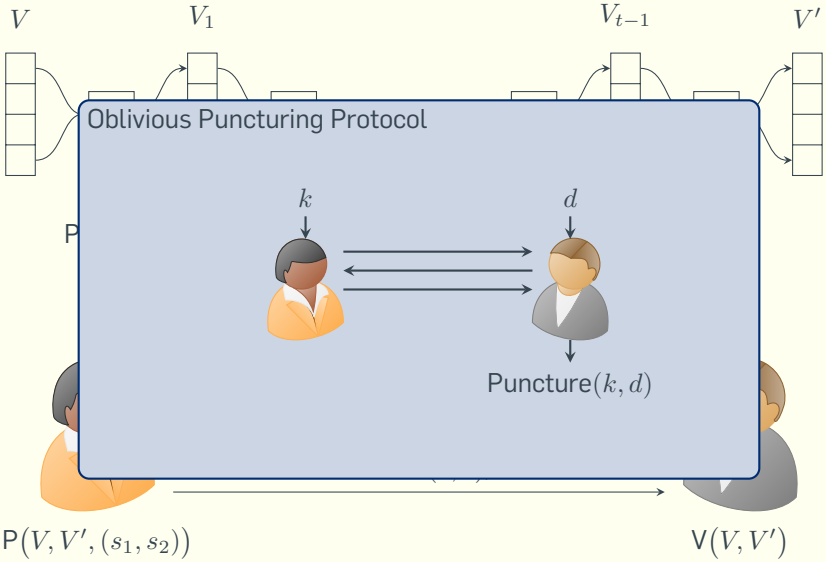
$$\Pr\left[\begin{array}{c}\mathsf{crs} \leftarrow \mathsf{Setup}(1^n); \\ (y, \mathsf{aux}) \leftarrow \mathsf{P}_0(\mathsf{crs}, x); \\ \mathsf{cert} \leftarrow \langle \mathsf{P}_1(x, y, \mathsf{aux}), \mathsf{V}^*(\mathsf{crs}, x, y)\rangle\end{array} : \mathsf{J}(\mathsf{crs}, \mathsf{pk}, \mathsf{cert}) = 1\right] \leq \mathsf{negl}(n).$$

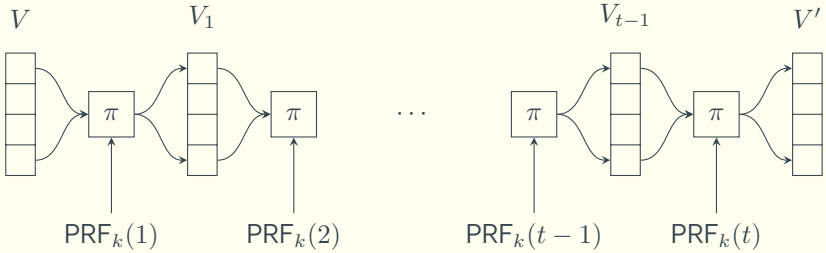

V
V_1
V_{t-1}
V'
π
π
$\cdots$
π
π
$\mathsf{PRF}_k(1)$
$\mathsf{PRF}_k(2)$
$\mathsf{PRF}_k(t-1)$
$\mathsf{PRF}_k(t)$

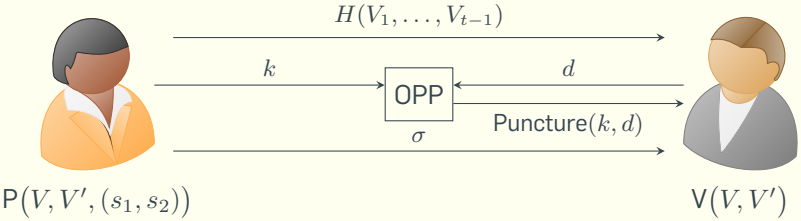

$H(V_1, \ldots, V_{t-1})$
$d \in \{1, \ldots, t\}$
$\mathsf{Puncture}(k, d)$
$\mathsf{P}\big(V, V', (s_1, s_2)\big)$
$\mathsf{V}\big(V, V'\big)$


V
V_1
V_{t-1}
V'
π
π
$\cdots$
π
π
$\mathsf{PRF}_k(1)$
$\mathsf{PRF}_k(2)$
$\mathsf{PRF}_k(t-1)$
$\mathsf{PRF}_k(t)$


$H(V_1, \ldots, V_{t-1})$
$d \in \{1, \ldots, t\}$
$\mathsf{Puncture}(k, d)$,σ
$\mathsf{P}\big(V, V', (s_1, s_2)\big)$
$\mathsf{V}\big(V, V'\big)$


V
V_1
V_{t-1}
V'
Oblivious Puncturing Protocol
k
d
Puncture(k, d)
P$(V, V', (s_1, s_2))$
V(V, V')


V
V_1
V_{t-1}
V'
π
π
$\cdots$
π
π
$\mathsf{PRF}_k(1)$
$\mathsf{PRF}_k(2)$
$\mathsf{PRF}_k(t-1)$
$\mathsf{PRF}_k(t)$


$H(V_1, \ldots, V_{t-1})$
k
OPP
d
σ
$\mathsf{Puncture}(k, d)$
$\mathsf{P}(V, V', (s_1, s_2))$
$\mathsf{V}(V, V')$

Comparison of shuffling 100,000 commitments.

Scheme	Assumptions	Trusted Setup	Soundness	Communication Cost (byte)
This Work	CRHF,PRG	None	2^{-2} 2^{-5} $2^{-\gamma}$	81 153 $32 + \lceil \gamma \cdot 24\frac{1}{8} \rceil$
This Work (with accountability)	CRHF,PRG,DDH	CRS	2^{-2} 2^{-5} $2^{-\gamma}$	416 992 $32 + \gamma \cdot 192$
Bayer-Groth [BG12]	Discrete Logarithm	CRS	negl(n)	$700,000$
Bulletproofs [BBBPWM18]	Discrete Logarithm	CRS	negl(n)	$1,600$
SNARKs [Groth16]	Generic Bilinear Group	SRS	negl(n)	144

Thanks!