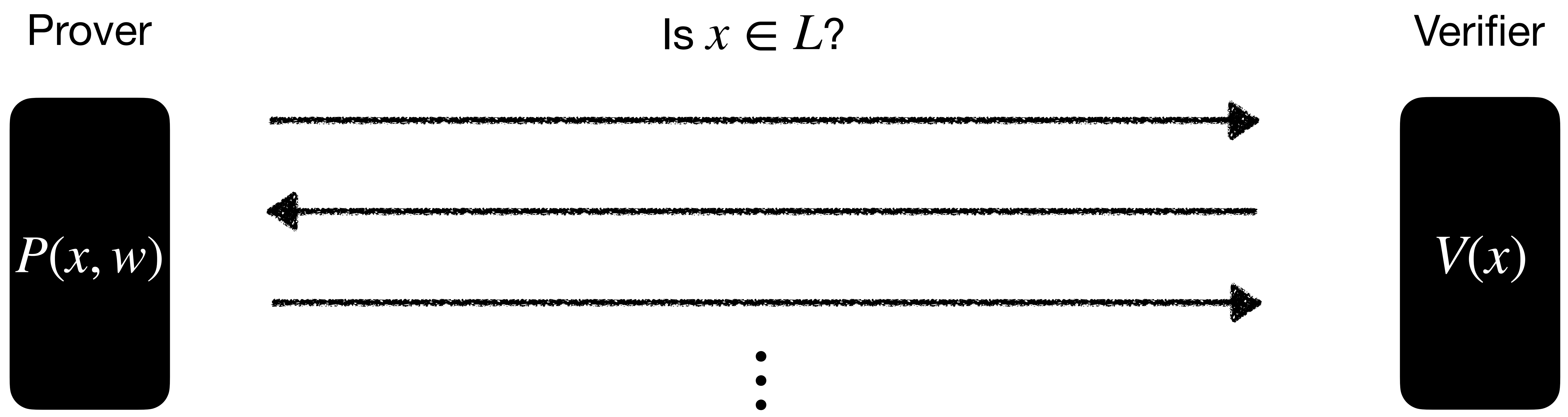
On the Security of Succinct Arguments from Vector Commitments

Alessandro Chiesa, Marcel Dall'Agnol, Ziyi Guan, Nick Spooner

Interactive proofs



Perfect completeness: For every instance $x \in L$,

$$\Pr\left[\langle P(x,w), V(x)\rangle = 1\right] = 1.$$

Soundness: For every instance $x \notin L$ and adversary $\tilde{P}$,

$$\Pr\left[\langle \tilde{P}, V(x)\rangle = 1\right] \leq \epsilon(x).$$

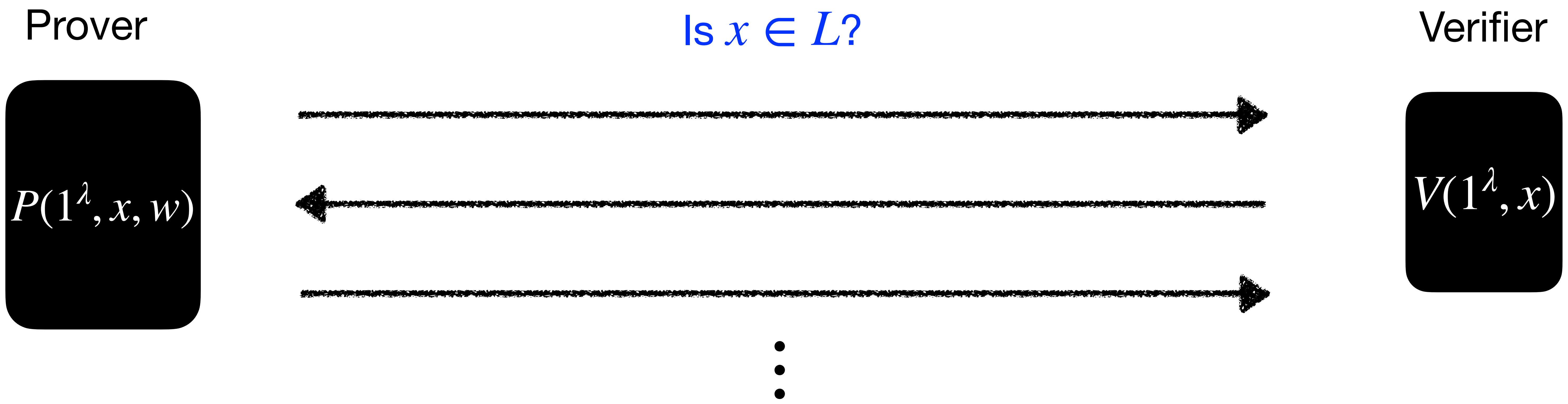
Basic efficiency metric: **COMMUNICATION COMPLEXITY** (number of bits exchanged during the interaction).

Limitation: NP-complete languages do not have IPs with cc $\ll |w|$ (or else the language would be easy).

(Indeed, [GH97] proved that, in general, $\mathrm{IP}[\mathrm{cc}] \subseteq \mathrm{BPTIME}[2^{\mathrm{cc}}]$.)

Interactive arguments

Interactive proofs with computational soundness



Computational soundness: For every $x \notin L$, security parameter $\lambda \in \mathbb{N}$, and t_{ARG}-bounded adversary $\tilde{P}$,

$$\Pr\left[\langle \tilde{P}, V(1^\lambda, x)\rangle = 1\right] \leq \epsilon_{\text{ARG}}(\lambda, x, t_{\text{ARG}}).$$

relaxes the soundness guarantee of interactive proofs

Limitations on the communication complexity of interactive proofs no longer hold,

AMAZING: there exist interactive arguments for NP with cc $\ll |w|$ (given basic cryptography)

These are known as **Succinct Interactive Arguments**.

Why study succinct interactive arguments?

A **fundamental primitive** known to exist assuming only simple cryptography (e.g. collision-resistant hash functions).

The savings in communication (cc $\ll |w|$) or even verification (time$(V) \ll |w|$) are remarkably useful.

Succinct arguments play a key role in notable applications (e.g., zero-knowledge with non-black-box simulation, malicious MPC, ...).

They also serve as a stepping stone towards succinct **non-interactive** arguments (SNARGs).

Recall: SNARGs for NP cannot be realized via a black-box reduction to a falsifiable assumption [GW11].

Often (though not always): SNARG = succinct interactive argument + non-falsifiable assumption / idealized model

The starting point of this talk is:

Kilian's protocol, the first and simplest succinct argument

Kilian's protocol

abstraction for a succinct commitment
with local openings (e.g. Merkle tree)

Building block #1: probabilistically checkable proof (PCP)

PCP prover

PCP verifier

P

Oracle string $\Pi \in \Sigma^l$

V

$\mathbf{V}$ uses r bits of randomness
and queries q locations of Π

The protocol:

Prover P

P

$\mathrm{cm} := \mathrm{Commit}(\Pi)$

PCP verifier randomness ρ

$(Q, \mathrm{ans}, \mathrm{pf})$ from Open

Verifier V

V

Building block #2: vector commitment scheme (VC)

Commit phase

$m \in \Sigma^l$ $\xrightarrow{\text{Commit}}$ cm

Open phase

ans

Query set $Q \subseteq [l]$ $\xrightarrow{\text{Open from cm}}$ pf

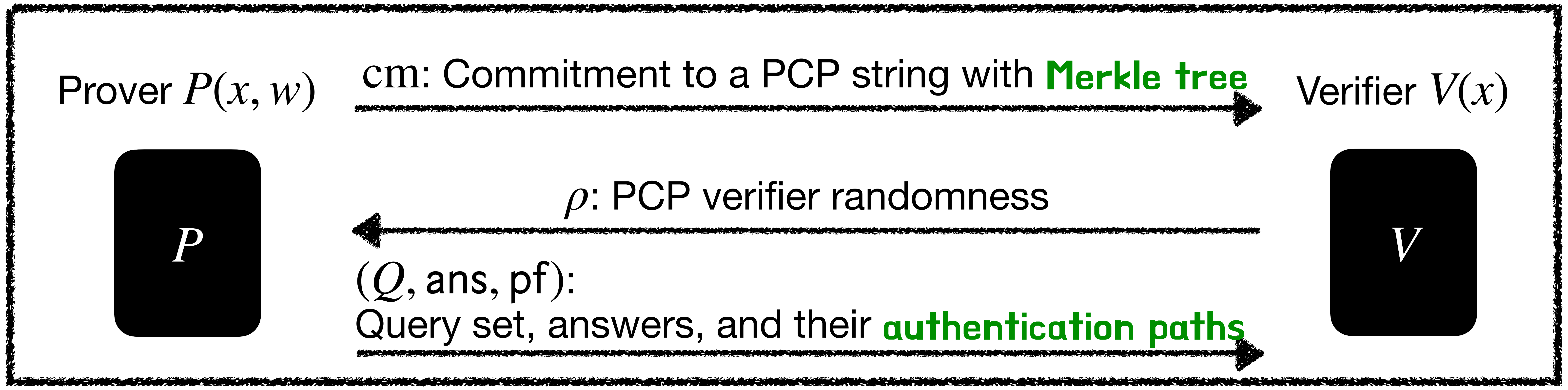
If ans $:= m[Q]$ then $\mathrm{Check}(\mathrm{cm}, Q, \mathrm{ans}, \mathrm{pf}) = 1$.

Position binding

$$\Pr\left[\begin{array}{l} (\mathrm{cm}, Q, Q', \mathrm{ans}, \mathrm{ans}', \mathrm{pf}, \mathrm{pf}') \leftarrow \mathrm{Adv} \\ \mathrm{Check}(\mathrm{cm}, Q, \mathrm{ans}, \mathrm{pf}) = 1 \\ \mathrm{Check}(\mathrm{cm}, Q', \mathrm{ans}', \mathrm{pf}') = 1 \end{array}\right] \leq \epsilon_{\mathrm{VC}}$$

Fundamental question:
What is the security of Kilian's protocol?

What is the security of Kilian's protocol?



Previously:

non-trivial restrictions on the PCP.

- [Kilian92] gives an informal analysis.
- [BG08] proves security of Kilian's protocol **assuming** the underlying PCP is non-adaptive and reverse-samplable.
  Their analysis is NOT tight: roughly $\epsilon_{\mathsf{ARG}} \leq 8 \cdot \epsilon_{\mathsf{PCP}} + \sqrt[3]{\epsilon_{\mathsf{VC}}}$ (multiplicative constant overhead)
- Kilian's protocol is widely used across cryptography but lacks a security proof in the general case

Our question: Given <u>any</u> PCP and <u>any</u> vector commitment scheme (VC), what is the security of Kilian's protocol wrt the security of the PCP and the VC?

Our result on Kilian's protocol

Theorem 1.

PCP for language L with
- proof length l
- query complexity q
- soundness error ϵ_{PCP}

PCP

Vector commitment scheme with position binding error ϵ_{VC}

VC

ARG := Kilian[PCP, VC]

For every $x \notin L$ and $\epsilon > 0$,

$$\epsilon_{\mathsf{ARG}}(\lambda, x, t_{\mathsf{ARG}}) \leq \epsilon_{\mathsf{PCP}}(x) + \epsilon_{\mathsf{VC}}(\lambda, l(x), \mathsf{q}(x), t_{\mathsf{VC}}) + \epsilon.$$

$$t_{\mathsf{VC}} = O\left(\frac{l}{\epsilon} \cdot t_{\mathsf{ARG}}\right)$$

Open: Is the $\frac{l}{\epsilon}$ overhead tight?

On the price of rewinding

Goal: achieve $\epsilon_{\mathsf{ARG}} = 2^{-40}$ against adversaries of size 2^{60} for Kilian's protocol.

Standard model

$t_{\mathsf{VC}} = O\left(\frac{l}{\epsilon} \cdot t_{\mathsf{ARG}}\right)$

For every $x \notin L$ and $\epsilon > 0$,
$\epsilon_{\mathsf{ARG}}(\lambda, x, t_{\mathsf{ARG}}) \leq \epsilon_{\mathsf{PCP}}(x) + \epsilon_{\mathsf{VC}}(\lambda, l(x), \mathsf{q}(x), t_{\mathsf{VC}}) + \epsilon.$

- Suppose $\epsilon_{\mathsf{PCP}} = 2^{-42}$ with $l = 2^{30}$.
- Suppose $\epsilon_{\mathsf{VC}} = (\lambda, l, \mathsf{q}, t_{\mathsf{VC}}) \leq \frac{t_{\mathsf{VC}}^2}{2^\lambda}$ (achieved by ideal Merkle trees).
- Setting $\epsilon := 2^{-42}$:
  - $t_{\mathsf{VC}} \leq 4 \cdot \frac{2^{30}}{2^{-42}} \cdot t_{\mathsf{ARG}} < 2^{80} \cdot t_{\mathsf{ARG}}$
  - $\epsilon_{\mathsf{VC}} \leq \frac{(2^{80} \cdot t_{\mathsf{ARG}})^2}{2^\lambda} = 2^{160-\lambda} \cdot t_{\mathsf{ARG}}^2 = 2^{280-\lambda}$
- Set $\lambda = 322$ to achieve the desired bound.

Random oracle model

For every $x \notin L$, [CY24]
$\epsilon_{\mathsf{ARG}}(\lambda, x, t_{\mathsf{ARG}}) \leq \epsilon_{\mathsf{PCP}}(x) + \frac{t_{\mathsf{ARG}}^2}{2^\lambda}.$

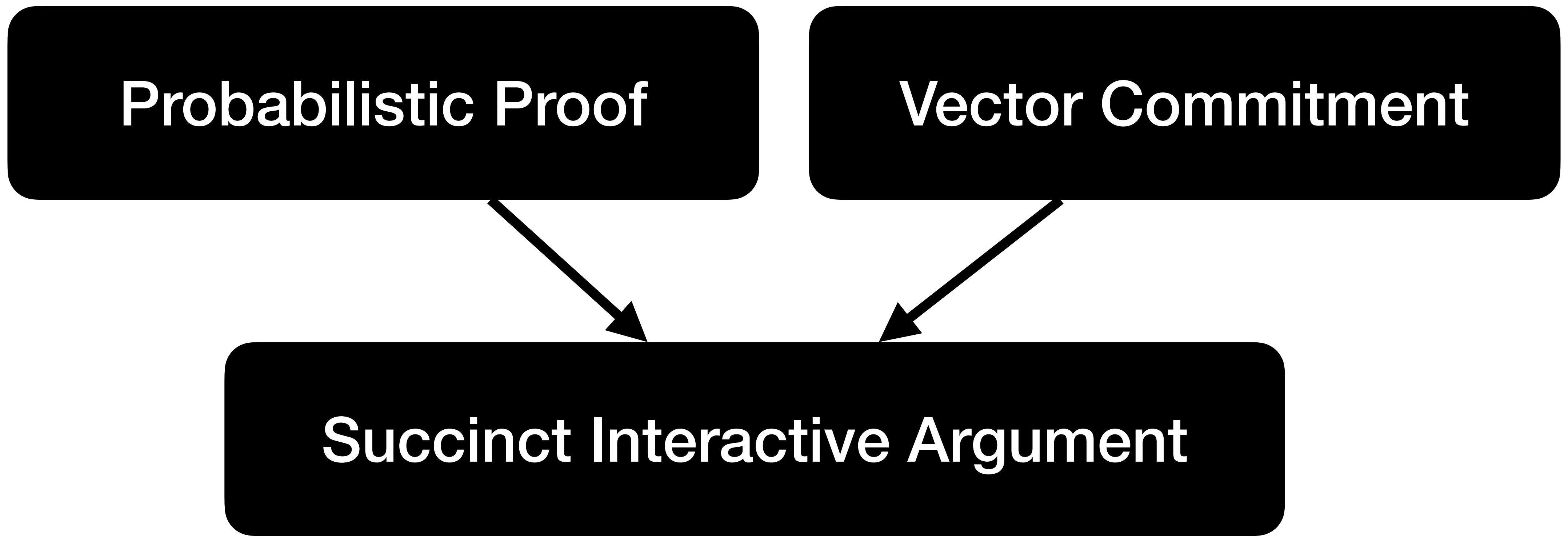
- Suppose $\epsilon_{\mathsf{PCP}} = 2^{-42}$
- $\epsilon_{\mathsf{VC}} \leq \frac{t_{\mathsf{ARG}}^2}{2^\lambda} = 2^{120-\lambda}$
- Set $\lambda = 162$ to achieve the desired bound.

- If the hash function is assumed ideal then extraction is straightline.
- If the hash function is merely collision-resistant then extraction is rewinding.
These computations illustrate the **PRICE OF REWINDING**.

Beyond Kilian: the VC-Based Approach

We understand Kilian's protocol ✅

Kilian's protocol is an example of a more general paradigm: the **VC-Based Approach**

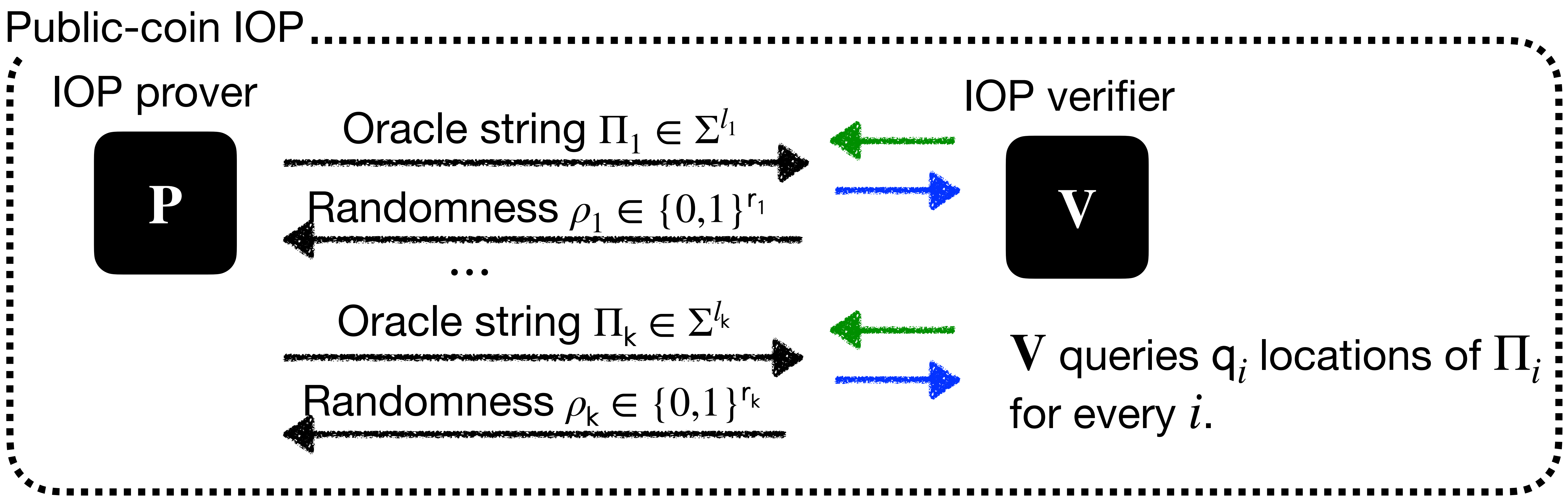


BASIC QUESTIONS:
How general is this paradigm?
When can we prove its security?

The case of public-coin IOPs [1/2]

Interactive oracle proofs (IOPs) are a multi-round generalization of PCPs [BCS16,RRR16].

An exciting line of works achieve public-coin IOPs with excellent efficiency. (In contrast, known PCPs have poor efficiency.)



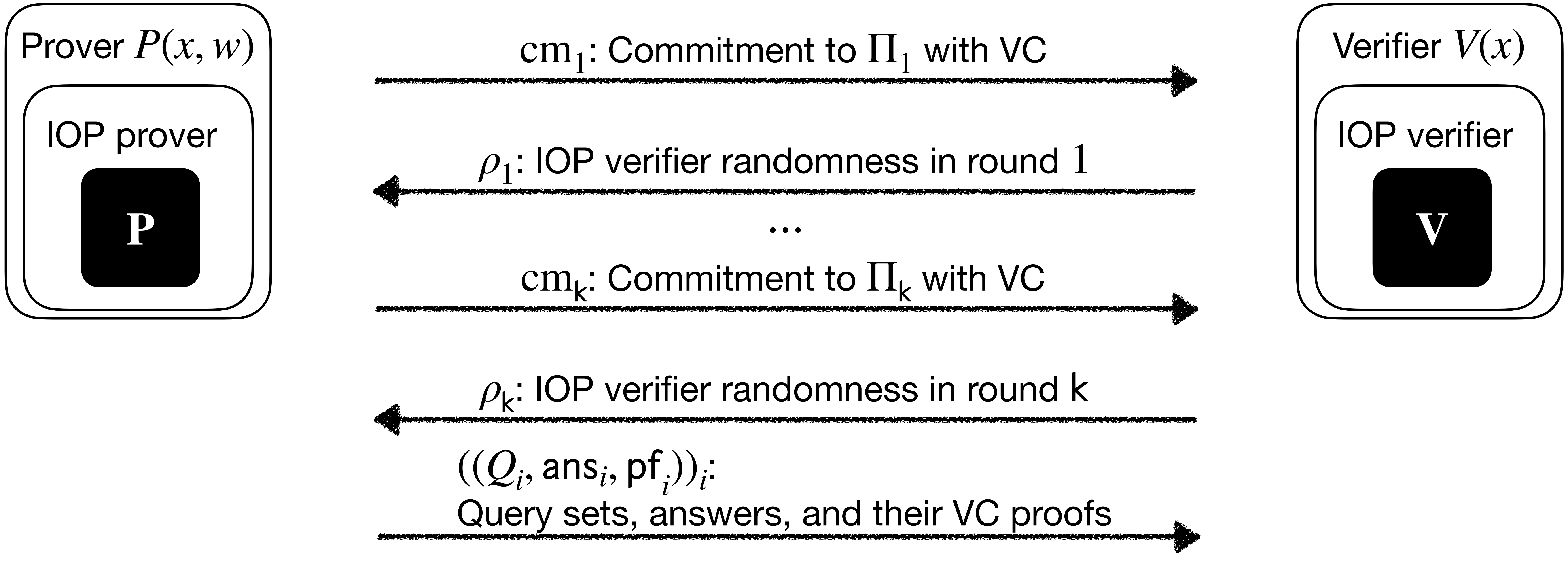
Public-coin IOPs play a key role in the construction of **efficient** succinct (interactive & non-interactive) arguments.

The case of public-coin IOPs [2/2]

The VC-based approach naturally extends to public-coin IOPs.

interactive variant of the BCS protocol [BCS16]
(public-coin IOP + random oracle = SNARG)

IBCS protocol



The IBCS protocol is a key ingredient in a line of work on linear-time succinct arguments [BCG20; RR22; HR22].

PROBLEM: there is no security analysis of the IBCS protocol. 😅

Our result on the IBCS protocol

Theorem 2.

Public-coin IOP for language L with
- total proof length l
- total query complexity q
- soundness error ϵ_{IOP}
- round complexity k

IOP

Vector commitment scheme with position binding error ϵ_{VC}

VC

ARG := IBCS[IOP, VC]

For every $x \notin L$ and $\epsilon > 0$,

$$\epsilon_{\mathrm{ARG}}(\lambda, x, t_{\mathrm{ARG}}) \leq \epsilon_{\mathrm{IOP}}(x) + \epsilon_{\mathrm{VC}}(\lambda, l(x), \mathsf{q}(x), t_{\mathrm{VC}}) + \epsilon.$$

can improve to $l_{\max}$ and $\mathsf{q}_{\max}$

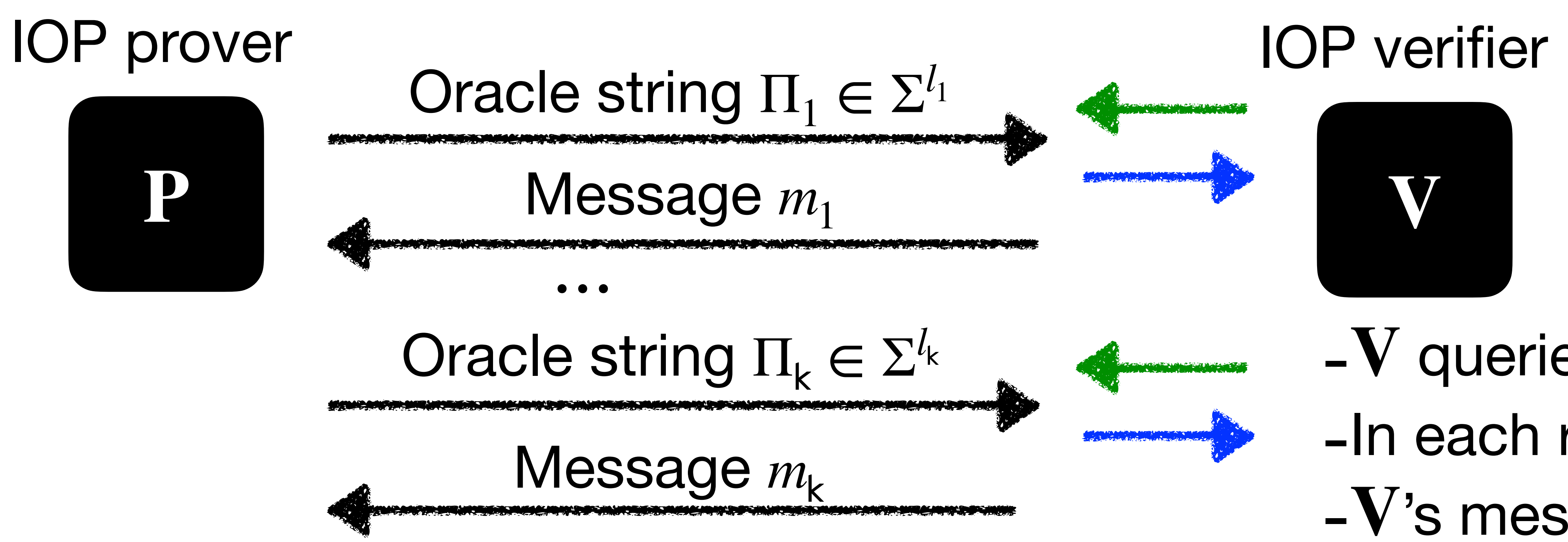
$$t_{\mathrm{VC}} = O\left(\frac{\mathsf{k} \cdot l}{\epsilon} \cdot t_{\mathrm{ARG}}\right)$$

Beyond public-coin IOPs?

Why should the VC-based approach "care" if the underlying IOP is public-coin?

In general, a private-coin IOP looks like this:

Private-coin IOP



-$\mathbf{V}$ queries q_i locations of Π_i for every i.
-In each round i, $\mathbf{V}$ can query $\Pi_1, \ldots, \Pi_i$.
-$\mathbf{V}$'s messages depend on its private randomness ρ and answers to its previous queries.

Applying the VC-based approach to a private-coin IOP directly leads to this protocol...

Finale protocol

The VC-based approach for private-coin IOPs

Prover $P(x, w)$

IOP prover

P

cm_1: Commitment to Π_1 with VC →

← $\mathbf{Q}_1$: IOP verifier query sets in round 1

$(\mathbf{ans}_1, \mathbf{pf}_1)$: Answers and proofs for $\mathbf{Q}_1$ →

← m_1: IOP verifier message in round 1

…

cm_k: Commitment to Π_k with VC →

← $\mathbf{Q}_\mathrm{k}$: IOP verifier query sets in round k

$(\mathbf{ans}_\mathrm{k}, \mathbf{pf}_\mathrm{k})$: Answers and proofs for $\mathbf{Q}_1$ →

← m_k: IOP verifier message in round 1

Verifier $V(x)$

IOP verifier

V

Boldface because in each round i, $\mathbf{Q}_i$ contains verifier's queries to $\Pi_1, \ldots, \Pi_i$.

Is the Finale protocol secure?
No. If the security of the IOP relies on queries being secret, then the Finale protocol is NOT secure.
(e.g. IOP verifier accepts if IOP prover guesses all its queries)

Def: An IOP is **public-query** if queries can be learned by the prover (in "real-time") without breaking security.

Clearly, the Finale protocol is secure whenever the underlying IOP is public-query... right?

Our result on Finale protocol

Theorem 3.

Public-query IOP for language L with
- total proof length l
- total query complexity q
- soundness error ϵ_{IOP}
- round complexity k
- **RCS with running time t_S**

IOP

Random Continuation Sampler (will define later)

Vector commitment scheme with position binding error ϵ_{VC}

VC

ARG := Finale[IOP, VC]

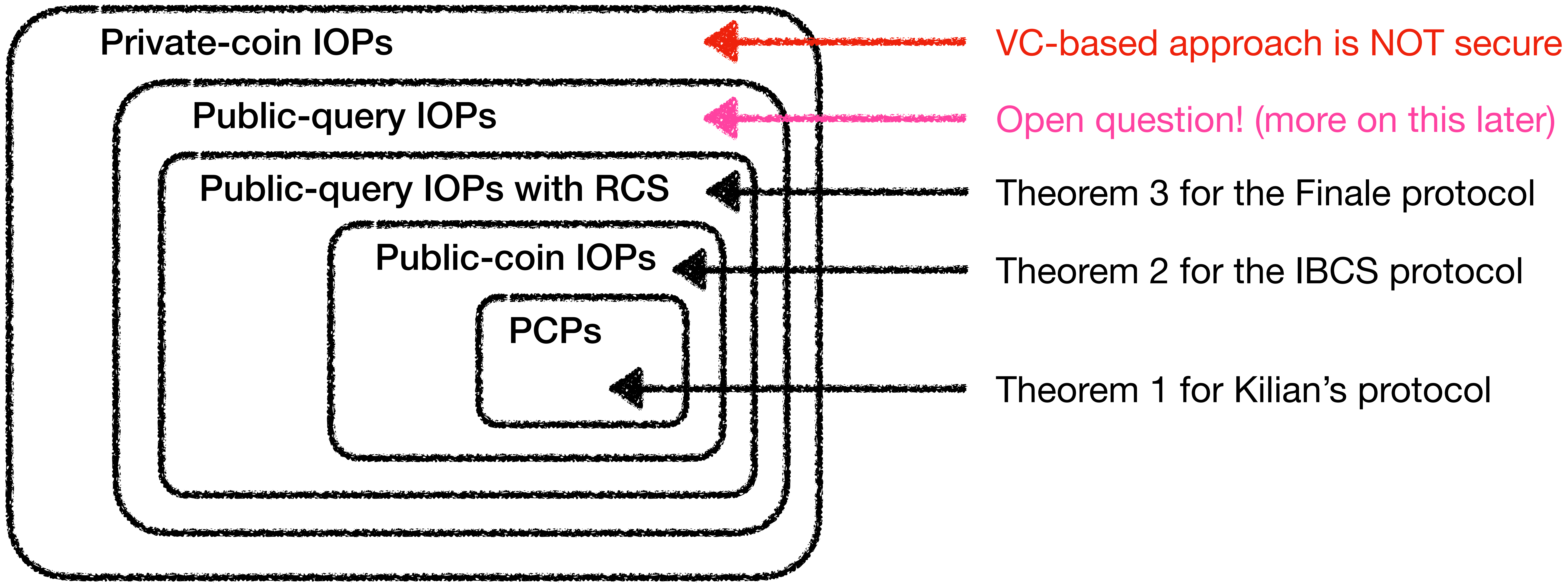
For every $x \notin L$ and $\epsilon > 0$,

$$\epsilon_{\text{ARG}}(\lambda, x, t_{\text{ARG}}) \leq \epsilon_{\text{IOP}}(x) + \epsilon_{\text{VC}}(\lambda, l(x), \mathsf{q}(x), t_{\text{VC}}) + \epsilon.$$

can improve to l_{max} and q_{max}

$$t_{\text{VC}} = O\left(\frac{\mathsf{k} \cdot l}{\epsilon} \cdot (t_{\text{ARG}} + t_S)\right)$$

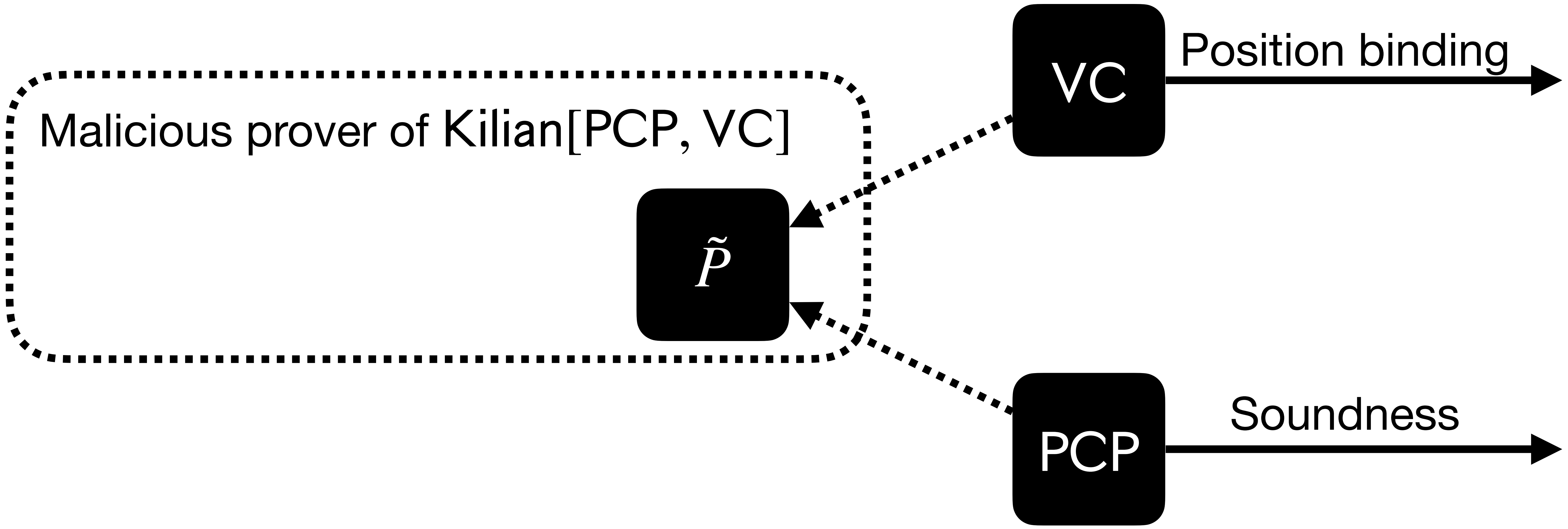
Summary of results



Kilian’s protocol

Security from rewinding [1/2]

Goal: relate the soundness error of $\mathsf{Kilian}[\mathsf{PCP}, \mathsf{VC}]$
to the soundness error of PCP and the position binding error of VC.



$\tilde{P}$ is forced to open locations of the committed PCP string $\tilde{\Pi}$

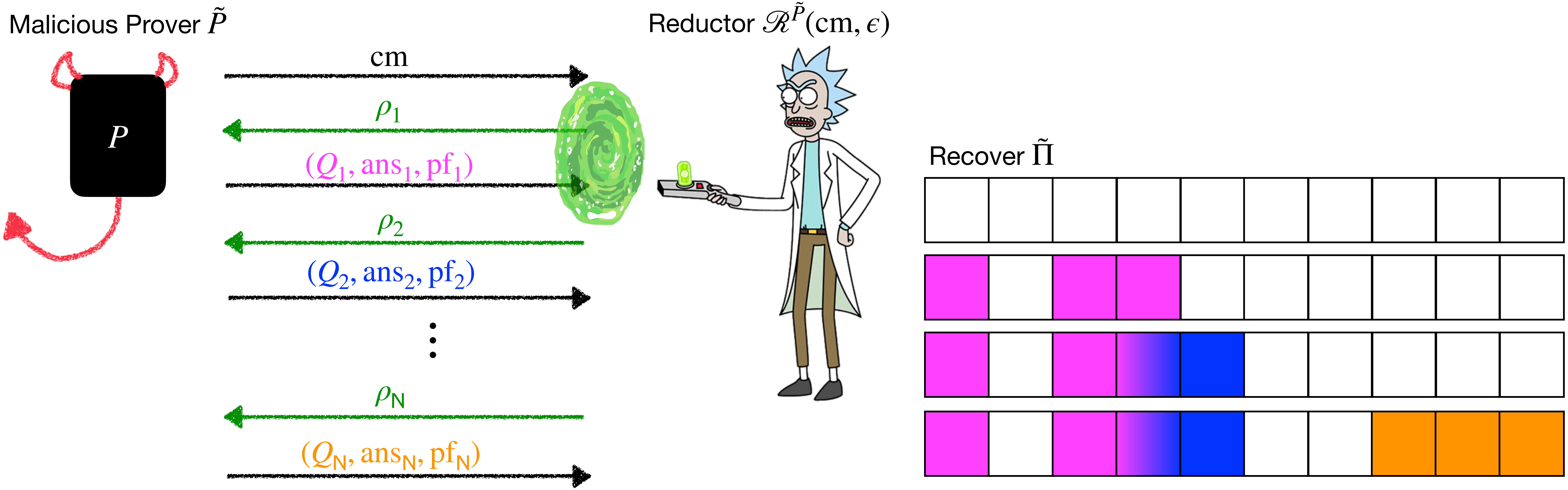
Rewind $\tilde{P}$ a number of times to get a bunch of $(Q, \mathsf{ans}, \mathsf{pf})$

$\Longrightarrow$ recover $\tilde{\Pi}$

$\Longrightarrow$ upper bound the success probability of $\tilde{\Pi}$ by soundness of PCP

Security from rewinding [2/2]

How to rewind?



Soundness of Kilian's protocol

Goal: $\Pr\left[\langle \tilde{P}, V(x)\rangle = 1\right] \leq \epsilon_{\mathsf{PCP}}(x) + \epsilon_{\mathsf{VC}}(\lambda, l, \mathsf{q}, t_{\mathsf{VC}}) + \epsilon$

$$\Pr\left[\begin{array}{l}\text{Sample } \rho \\ \text{PCP verifier accepts: } \mathbf{V}^{\tilde{\Pi}}(x,\rho) = 1 \\ \text{ARG verifier accepts: } V(x, \rho, Q, \mathsf{ans}, \mathsf{pf})\rangle = 1\end{array}\right]$$

Produced by the reductor $\mathscr{R}^{\tilde{P}}$

Produced by a t_{ARG}-time adversary $\tilde{P}$ given ρ

Soundness of PCP ✅

$\implies \leq \epsilon_{\mathsf{PCP}}(x)$

$$\Pr\left[\begin{array}{l}\text{Sample } \rho \\ \text{PCP verifier rejects: } \mathbf{V}^{\tilde{\Pi}}(x,\rho) \neq 1 \\ \text{ARG verifier accepts: } V(x, \rho, Q, \mathsf{ans}, \mathsf{pf})\rangle = 1\end{array}\right]$$

Security reduction lemma $\implies \leq \epsilon_{\mathsf{VC}}(\lambda, l, \mathsf{q}, t_{\mathsf{VC}}) + \epsilon$

$$t_{\mathsf{VC}} = O\left(\frac{l}{\epsilon} \cdot t_{\mathsf{ARG}}\right)$$

Proof of the Security reduction lemma

$$\begin{bmatrix} \text{Sample } \rho \\ \text{PCP verifier rejects: } \mathbf{V}^{\tilde{\Pi}}(x,\rho) \neq 1 \\ \text{ARG verifier accepts: } V(x,\rho,Q,\mathsf{ans},\mathsf{pf})\rangle = 1 \end{bmatrix}$$

$$\Longrightarrow \begin{bmatrix} \exists\, q \text{ s.t. } \tilde{\Pi}[q] \neq \mathsf{ans}[q] \\ \text{Yet both VC check pass} \end{bmatrix} \vee \begin{bmatrix} \exists\, q \text{ s.t. } \tilde{\Pi}[q] \text{ is unfilled} \\ \text{Yet } q \in Q \end{bmatrix}$$

VC position binding $\Longrightarrow \leq \epsilon_{\mathsf{VC}}(\lambda, l, \mathsf{q}, t_{\mathsf{VC}})$

$$t_{\mathsf{VC}} = O\left(\frac{l}{\epsilon} \cdot t_{\mathsf{ARG}}\right)$$

Missing queries

-For each q, the probability that q is not queried by the reductor $\mathscr{R}$ but is queried by the ARG verifier V is $1/\mathsf{N}$:

- Not hitting q for N times but hit it for the $(\mathsf{N}+1)$-th time

-Probability that there exists such a $q \leq l/\mathsf{N}$

-Setting $\mathsf{N} := l/\epsilon \implies \leq \epsilon$

-t_{VC} also depends on N: VC adversary runs the reductor $\mathscr{R}$

Recap: Security of Kilian’s protocol

$$t_{\mathsf{VC}} = O\left(\frac{l}{\epsilon} \cdot t_{\mathsf{ARG}}\right)$$

For every $x \notin L$ and $\epsilon > 0$,

$$\epsilon_{\mathsf{ARG}}(\lambda, x, t_{\mathsf{ARG}}) \leq \epsilon_{\mathsf{PCP}}(x) + \epsilon_{\mathsf{VC}}(\lambda, l(x), \mathsf{q}(x), t_{\mathsf{VC}}) + \epsilon.$$

On the $\frac{l}{\epsilon}$ overhead:

- Rewinding l times is necessary (maybe all PCP queries but 1 are fixed)
- Some rewinds may yield garbage so need $1/\epsilon$ more times as buffer
  - The query answers were found in previous rewinds
  - VC check does not accept the query answers

Wonderful open question: is the overhead tight or not?

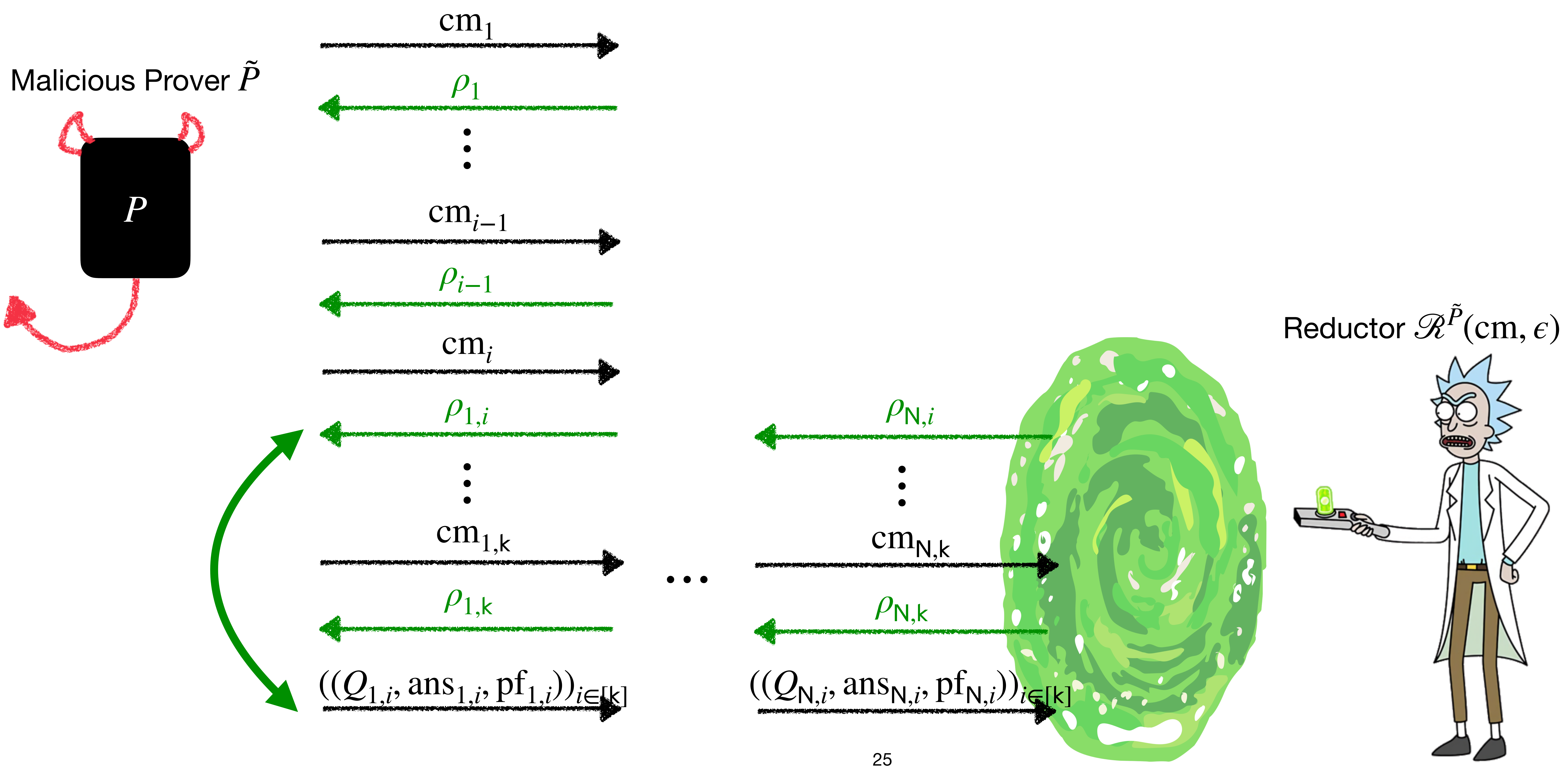
Why 30 years for a security proof of Kilian's protocol?

- The focus of the security analysis of [BG08] is specific for "universal arguments"
  - Do not have a polynomial bound on the size of the hash tree used by $\tilde{P}$.
  - PCP must be (efficiently) reverse-samplable.
- The intuition for the security of Kilian's protocol is clear but achieving a general security analysis of it has (bizarrely) not been done until this work

IBCS protocol

Security from rewinding

How to rewind to recover $\tilde{\Pi}_i$?



Security reduction lemma

$$t_{\mathsf{VC}} = O\left(\frac{\mathsf{k} \cdot l}{\epsilon} \cdot t_{\mathsf{ARG}}\right)$$

$$\Pr\left[\begin{array}{l}\text{Sample } \rho \\ \text{IOP verifier rejects: } \mathbf{V}^{(\tilde{\Pi}_1,\ldots,\tilde{\Pi}_k)}(x,\rho) \neq 1 \\ \text{ARG verifier accepts: } V(x,\rho,((Q_i,\mathsf{ans}_i,\mathsf{pf}_i))_i)\rangle = 1\end{array}\right] \leq \epsilon_{\mathsf{VC}}(\lambda, l, \mathsf{q}, t_{\mathsf{VC}}) + \epsilon$$

$$\Pr\left[\begin{array}{l}\exists\, i, q \text{ s.t. } \tilde{\Pi}_i[q] \neq \mathsf{ans}_i[q] \\ \text{Yet both VC check pass}\end{array}\right]$$

VC position binding $\Longrightarrow \leq \epsilon_{\mathsf{VC}}(\lambda, l, \mathsf{q}, t_{\mathsf{VC}})$

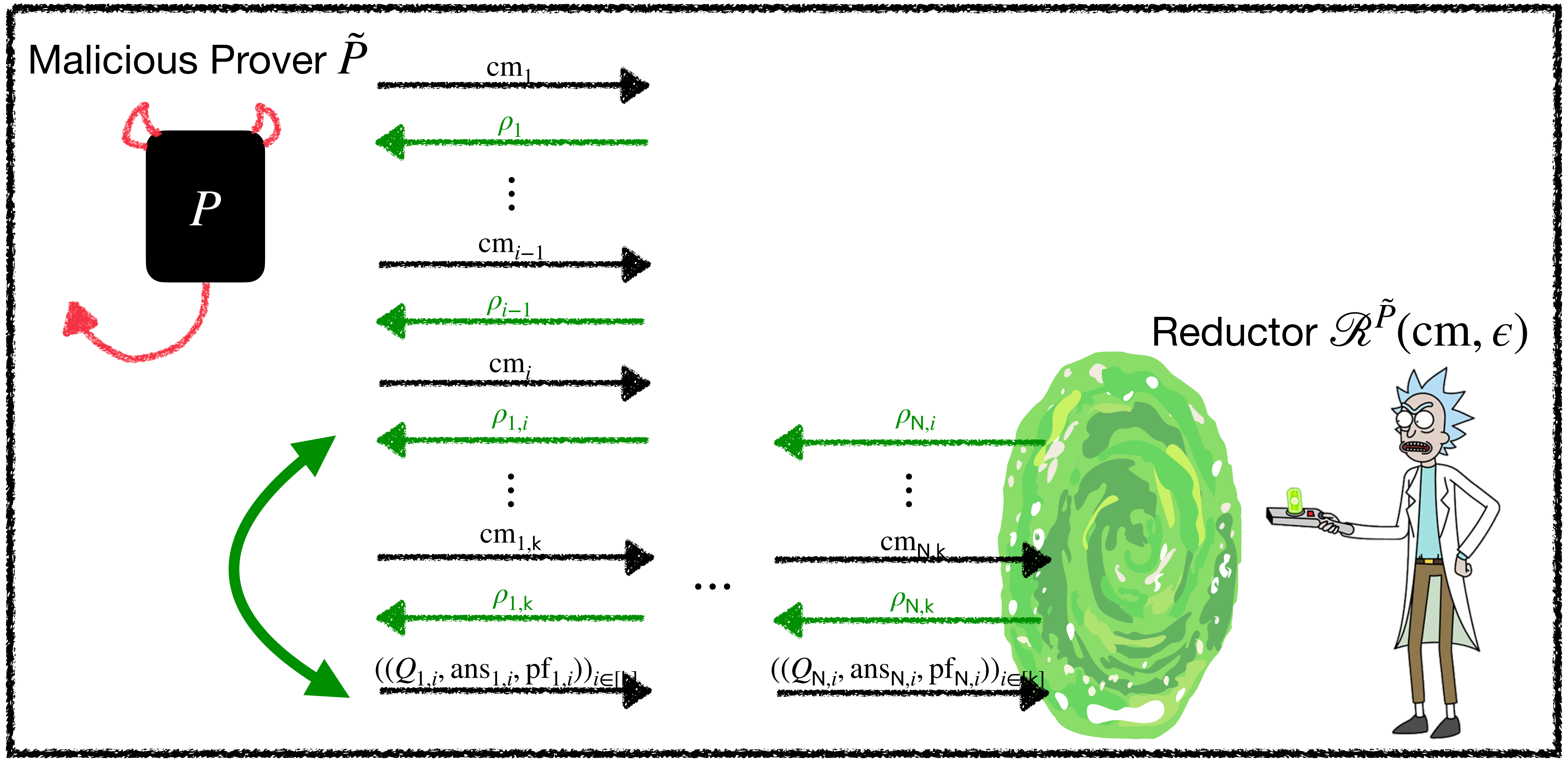
$$\Pr\left[\begin{array}{l}\exists\, i, q \text{ s.t. } \tilde{\Pi}_i[q] \text{ is unfilled} \\ \text{Yet } q \in Q_i\end{array}\right]$$

Missing queries $\Longrightarrow \leq \epsilon$

How about private-coin IOPs?

Security from rewinding [1/2]

How to rewind?



Key: the reductor $\mathscr{R}$ must sample **consistent random continuations** of the argument interaction.

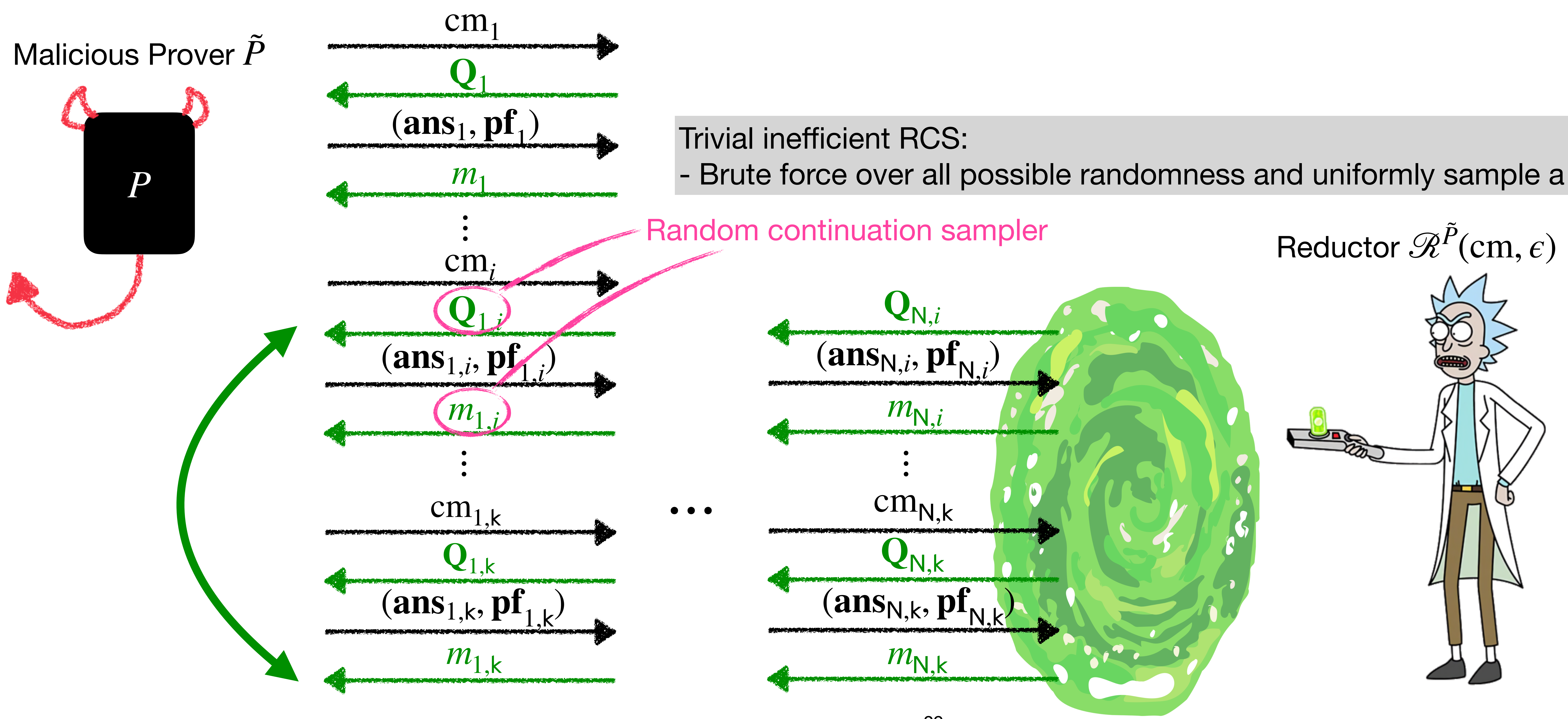
- Kilian reductor: sample uniform randomness of the PCP verifier
- IBCS reductor: sample uniform randomness of the IOP verifier starting from round i

Security from rewinding [2/2]

Key: given partial interaction transcript, the reductor $\mathscr{R}$ must finish the interaction consistently (with respect to the unknown private verifier randomness)

$\Longrightarrow$ **Random continuation sampler (RCS)**

Trivial inefficient RCS:
- Brute force over all possible randomness and uniformly sample a consistent one.



Security reduction lemma

$$t_{\mathsf{VC}} = O\left(\frac{\mathsf{k} \cdot l}{\epsilon} \cdot (t_{\mathsf{ARG}} + t_S)\right)$$

$$\Pr\left[\begin{array}{l} \text{Fix } \rho \\ \text{IOP verifier rejects: } \mathbf{V}^{(\tilde{\Pi}_1,\ldots,\tilde{\Pi}_k)}(x,\rho) \neq 1 \\ \text{ARG verifier accepts: } V(x,\rho,((Q_i,\mathsf{ans}_i,\mathsf{pf}_i))_i)\rangle = 1 \end{array}\right] \leq \epsilon_{\mathsf{VC}}(\lambda, l, \mathsf{q}, t_{\mathsf{VC}}) + \epsilon$$

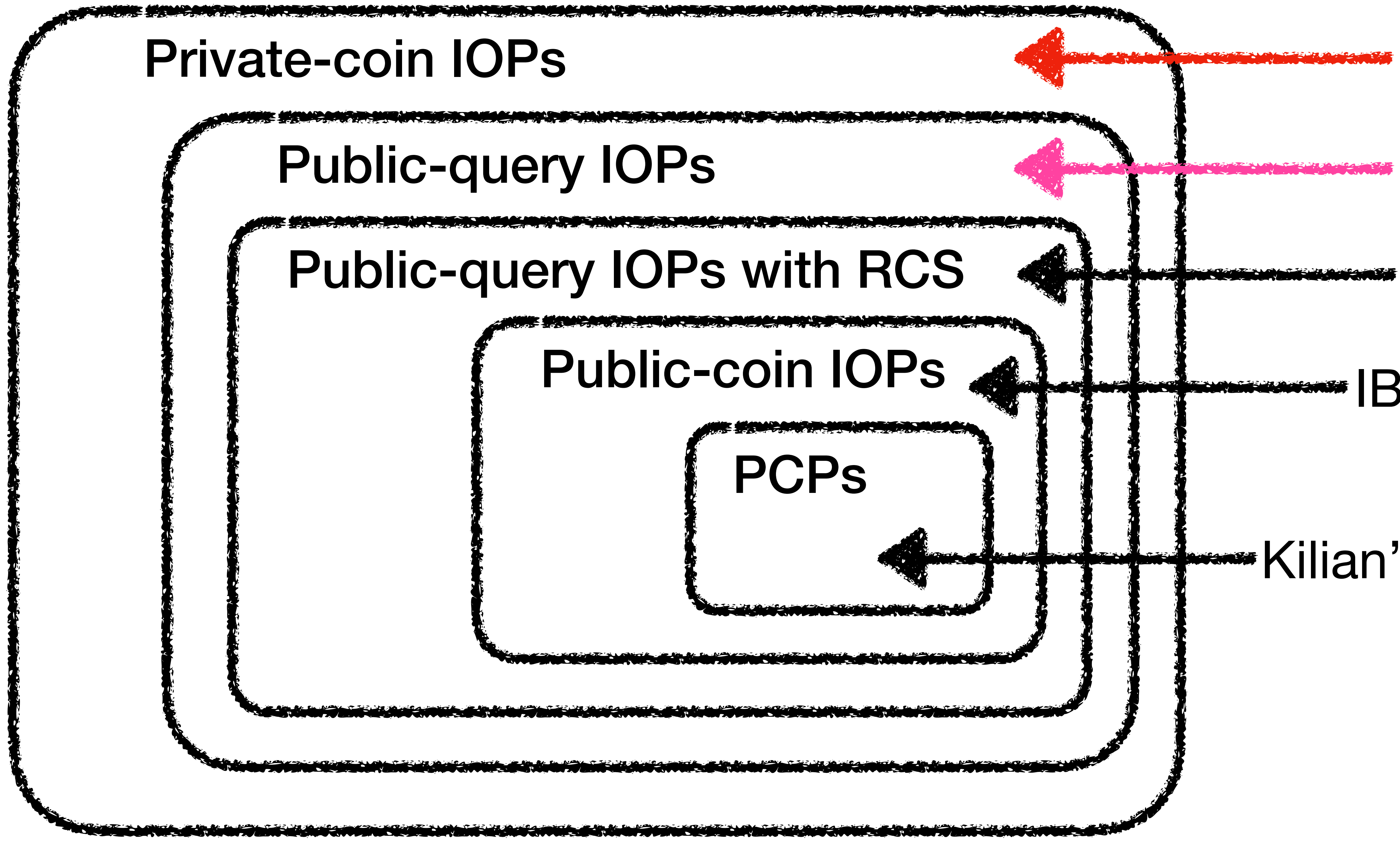
$$\Pr\left[\begin{array}{l} \exists\, i, q \text{ s.t. } \tilde{\Pi}_i[q] \neq \mathsf{ans}_i[q] \\ \text{Yet both VC check pass} \end{array}\right]$$

VC position binding $\Longrightarrow \leq \epsilon_{\mathsf{VC}}(\lambda, l, \mathsf{q}, t_{\mathsf{VC}})$

$$\Pr\left[\begin{array}{l} \exists\, i, q \text{ s.t. } \tilde{\Pi}_i[q] \text{ is unfilled} \\ \text{Yet } q \in Q_i \end{array}\right]$$

Missing queries $\Longrightarrow \leq \epsilon$

Open question



Vector commitment based approach doesn't work!

Unknown: when does Finale stop working?
Is RCS necessary to show security of Finale?

Finale protocol

IBCS protocol

Kilian's protocol

Observation: there is a public-query IOP without RCS.
(Hence our analysis does NOT cover all public-query IOPs.)

A public-query IOP that does not admit an RCS:
- Consider $r = (r_1, \ldots, r_k)$ to be IOP verifier's private randomness.
- The i-th message of the IOP verifier is $m_i := \mathrm{PRG}(r_i)$.
- Hard for any efficient algorithm to sample m_i given prior rounds.

Question: Is there a different analysis that could cover them all?

A conjecture: No. (black-box reduction $\Longrightarrow$ rewinding $\Longrightarrow$ RCS)

A partial result: $\mathrm{Finale}[\mathrm{IOP}, \mathrm{VC}]$ has RCS iff IOP has RCS.

Thank you!

https://eprint.iacr.org/2023/1737