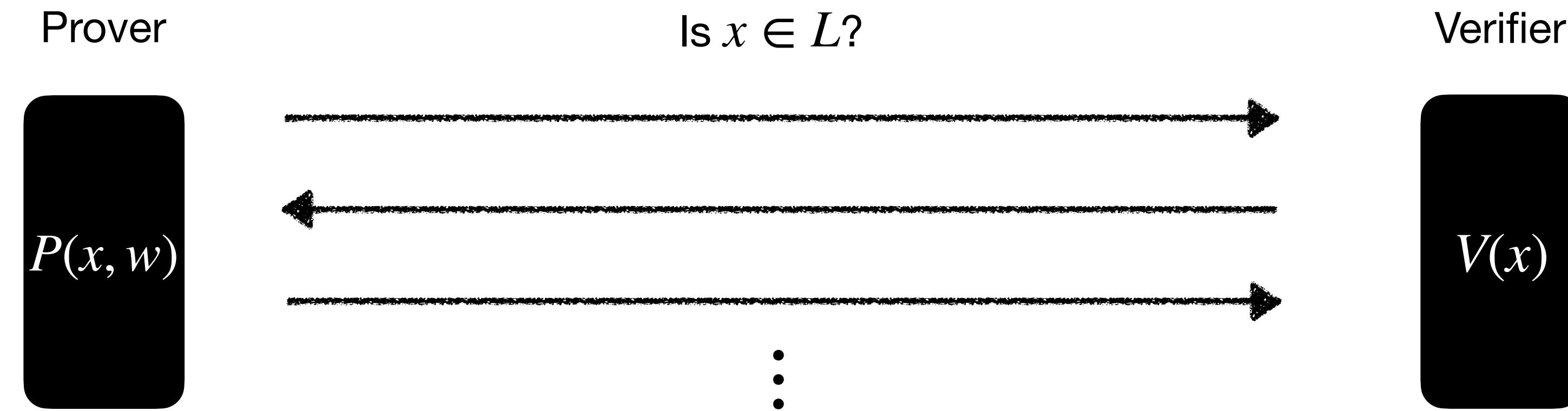


On the Security of Succinct Arguments from Vector Commitments

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

Interactive proofs



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

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

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

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

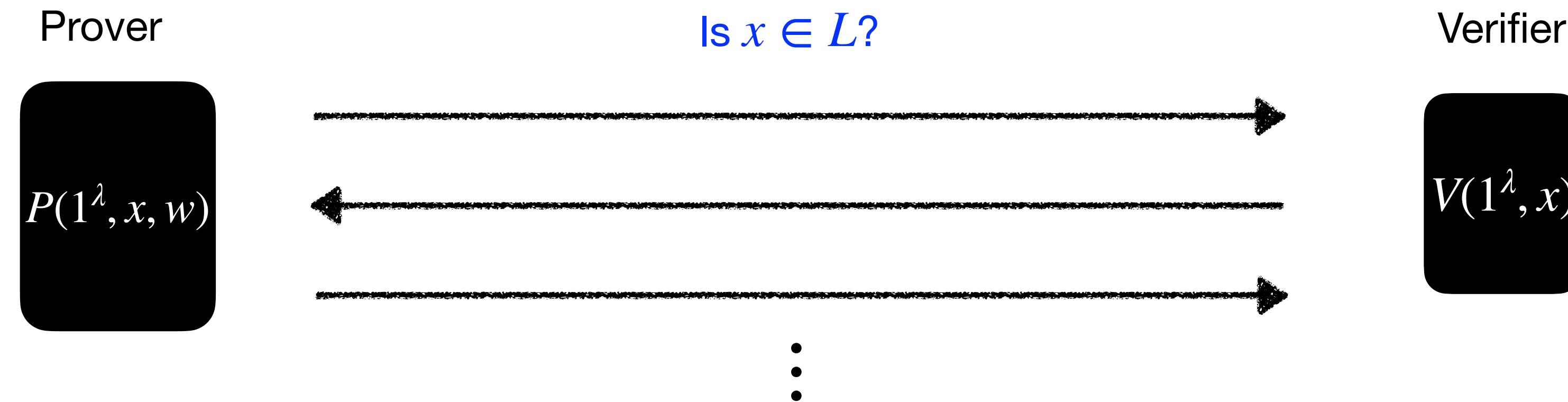
Basic efficiency metric: **COMMUNICATION COMPLEXITY** (number of bits exchanged during the interaction).

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, $\text{IP}[cc] \subseteq \text{BPTIME}[2^{cc}]$.)

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} ,

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

← 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 ($\text{time}(V) \ll |w|$) 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

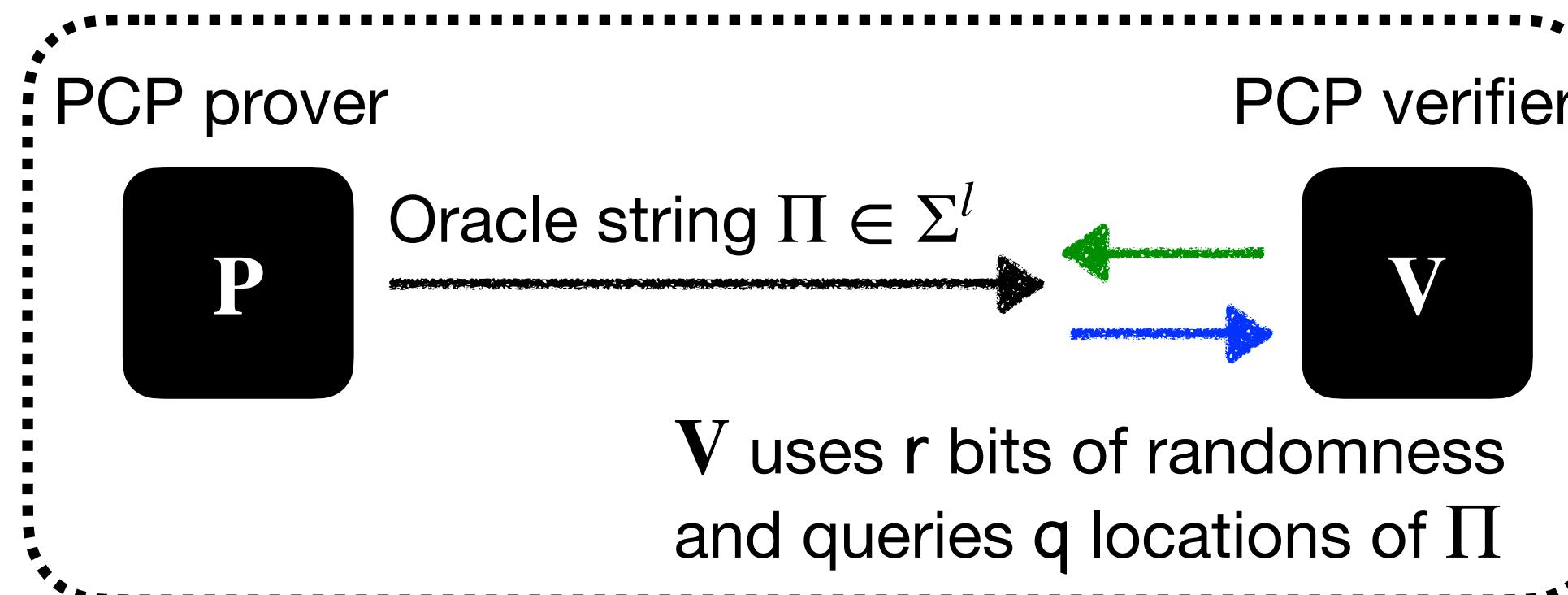
The starting point of this talk is:

Kilian's protocol, the first and simplest succinct argument

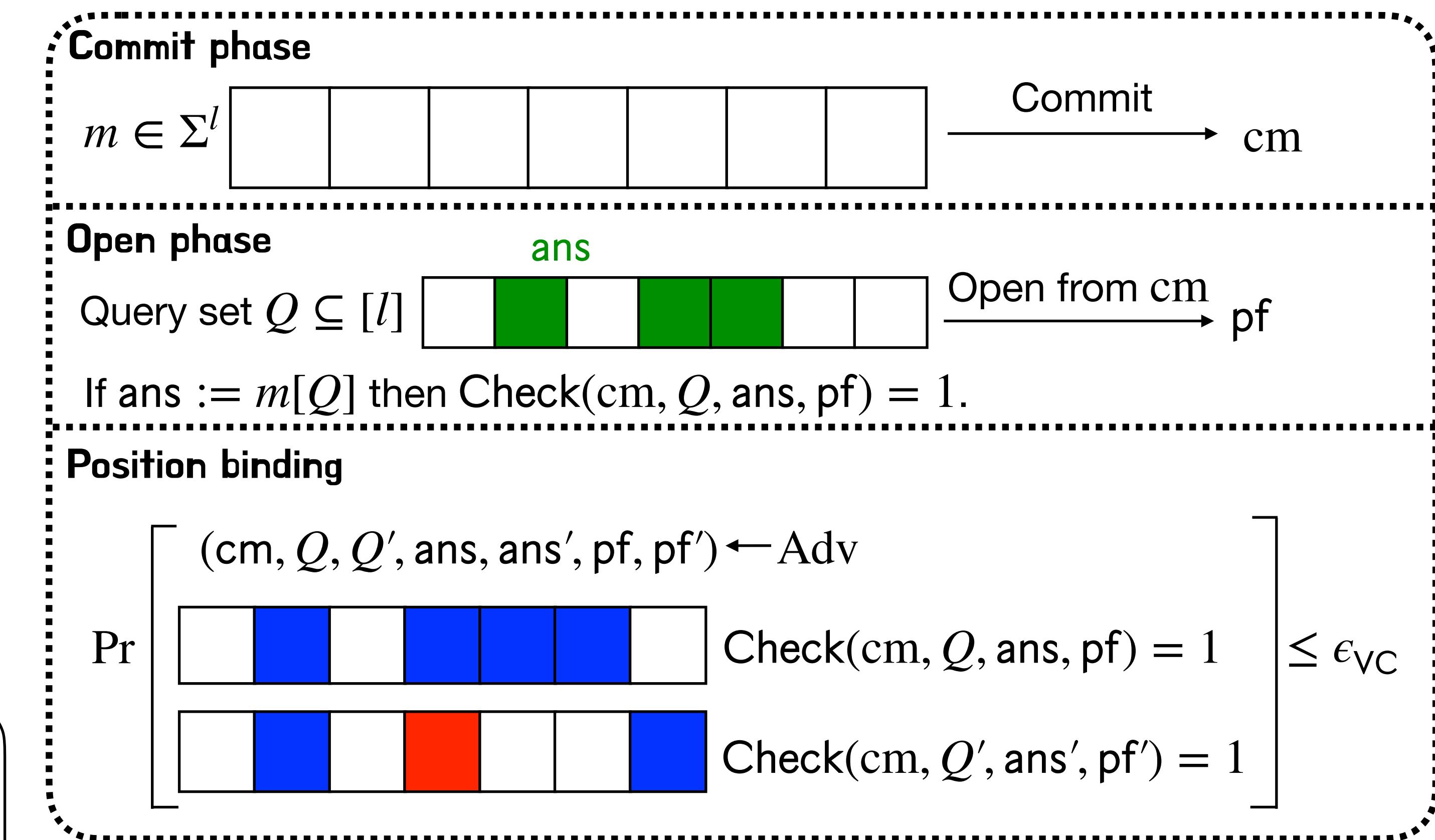
Kilian's protocol

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

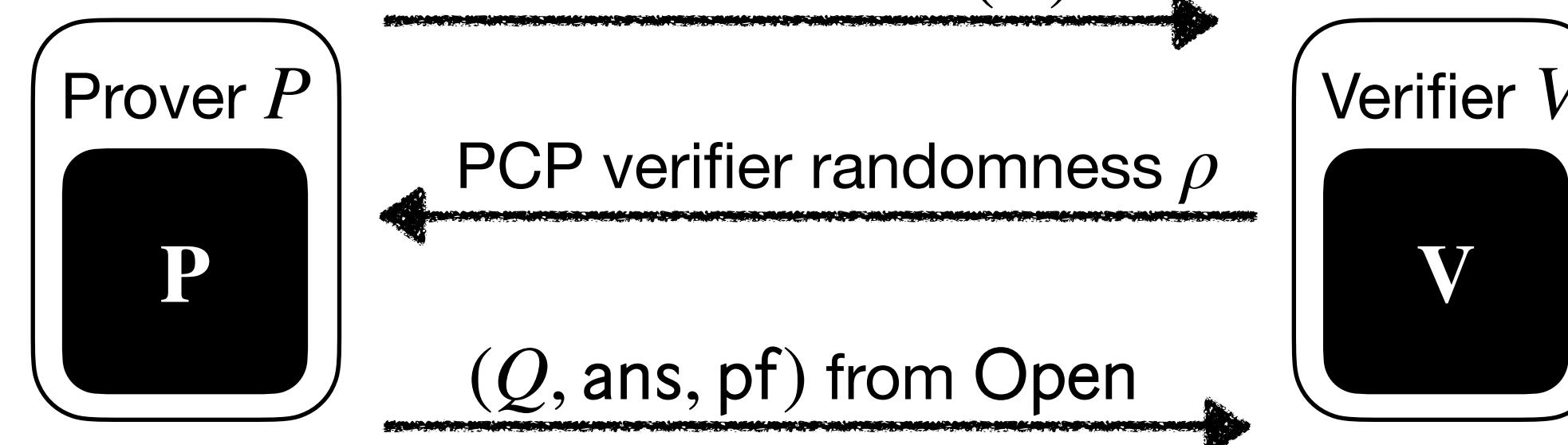
Building block #1: probabilistically checkable proof (PCP)



Building block #2: vector commitment scheme (VC)

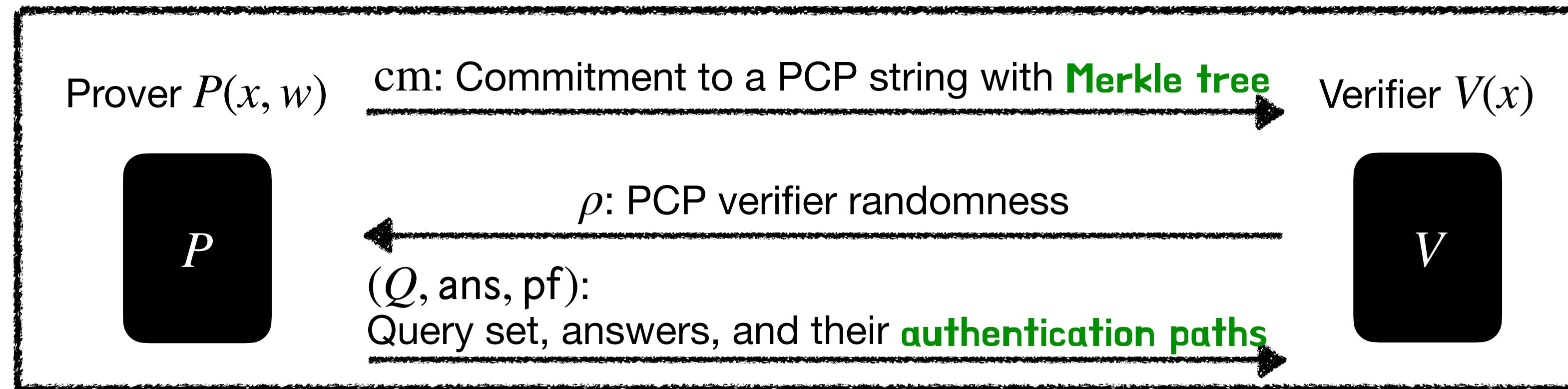


The protocol:



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

What is the security of Kilian's protocol?



Previously:

- [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 $\epsilon_{\text{ARG}} \leq 5 \cdot \epsilon_{\text{PCP}} + \text{negl}$ (**multiplicative constant overhead**)
- Kilian's protocol is widely used across cryptography but lacks a security proof in the general case

Our question: Given any PCP and any 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 $\epsilon > 0$,

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

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

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

On the price of rewinding

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

Standard model

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

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

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

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

Random oracle model

For every $x \notin L$,

$$\epsilon_{\text{ARG}}(\lambda, x, t_{\text{ARG}}) \leq \epsilon_{\text{PCP}}(x) + \frac{t_{\text{ARG}}^2}{2^\lambda}.$$

[CY24]

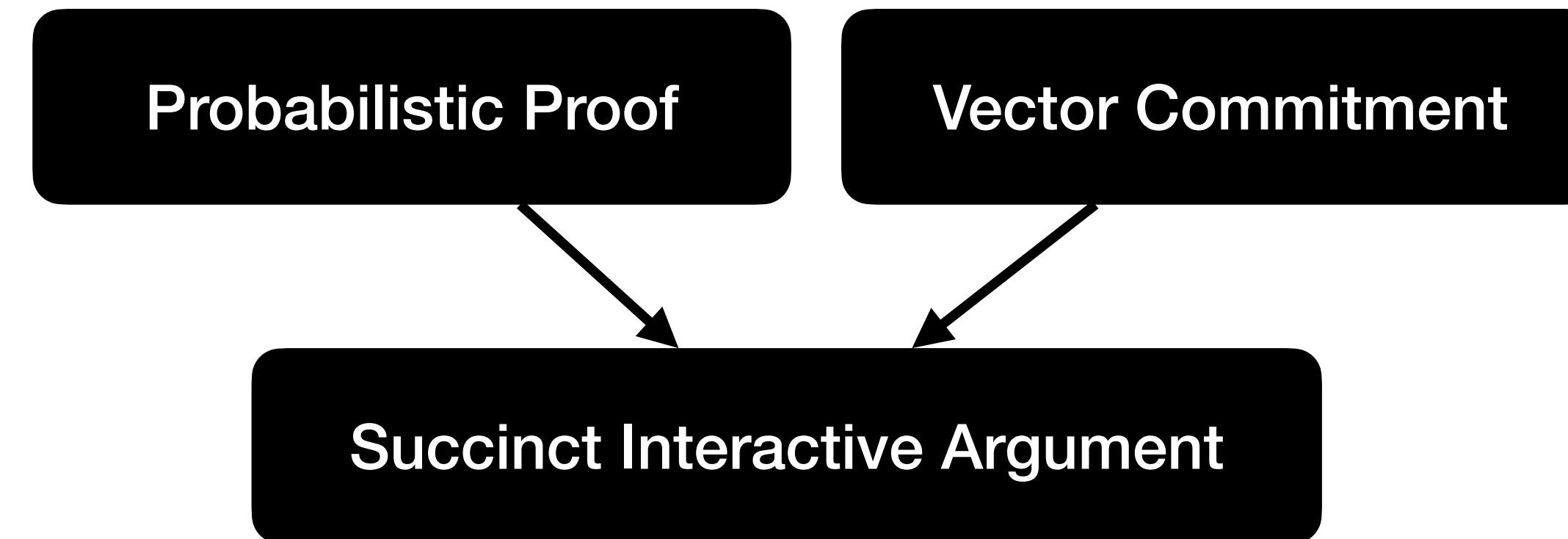
- Suppose $\epsilon_{\text{PCP}} = 2^{-42}$
- $\epsilon_{\text{VC}} \leq \frac{t_{\text{ARG}}^2}{2^\lambda} = 2^{120-\lambda}$
- Set $\lambda = 162$ to achieve the desired bound.

- 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**



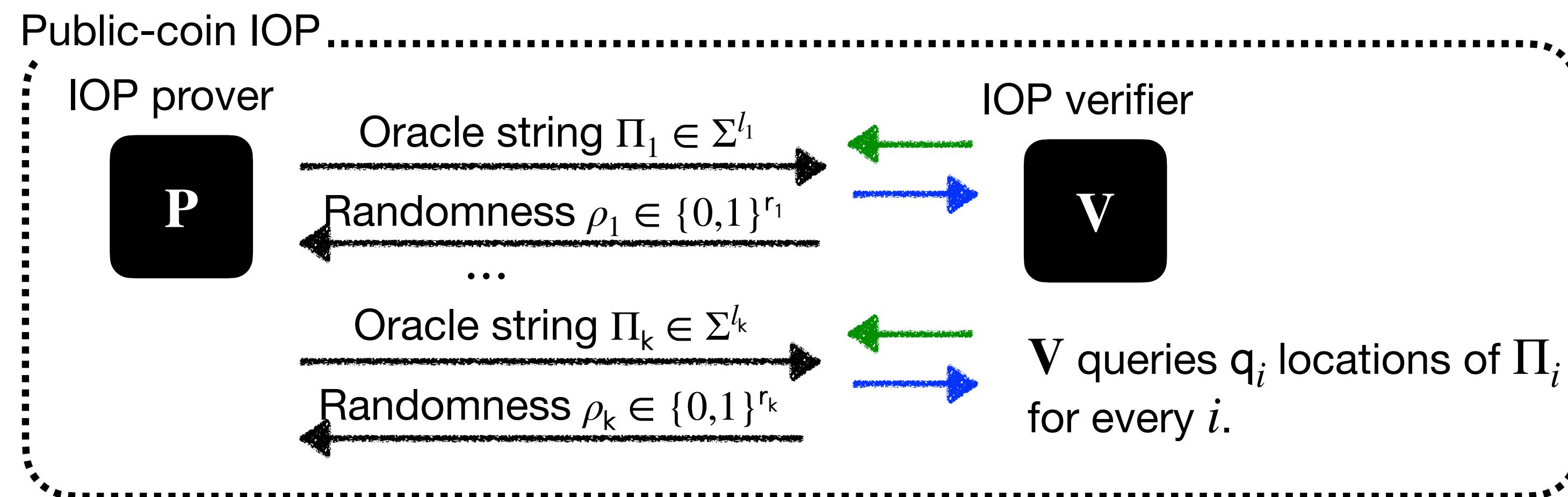
BASIC QUESTIONS:
How general is this paradigm?
When can we prove its security?

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.

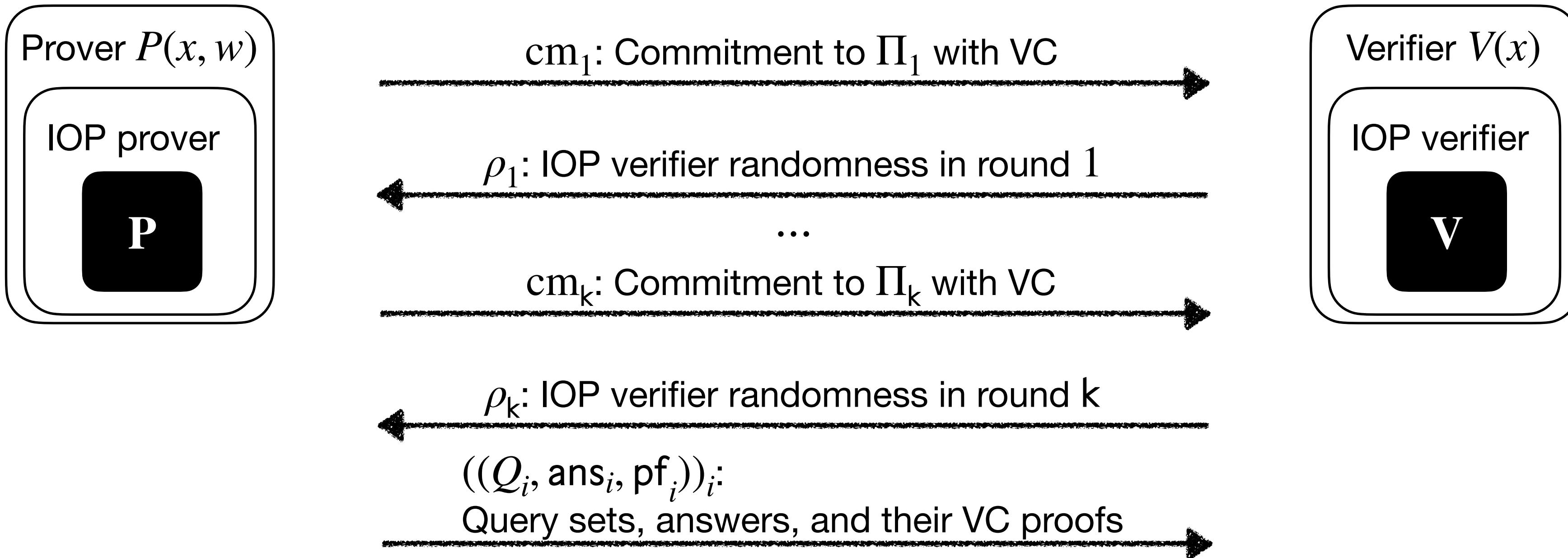
The case of public-coin IOPs

[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].

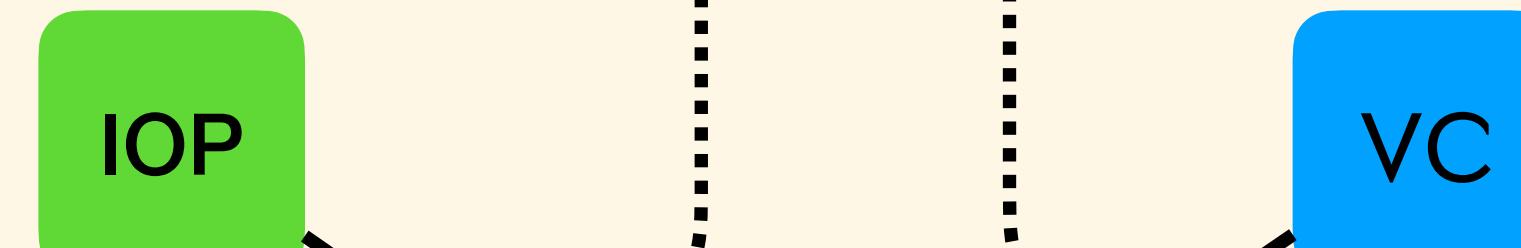
PROBLEM: there is **no security analysis** of the IBCS protocol. 😅

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



Vector commitment scheme with
position binding error ϵ_{VC}

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

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

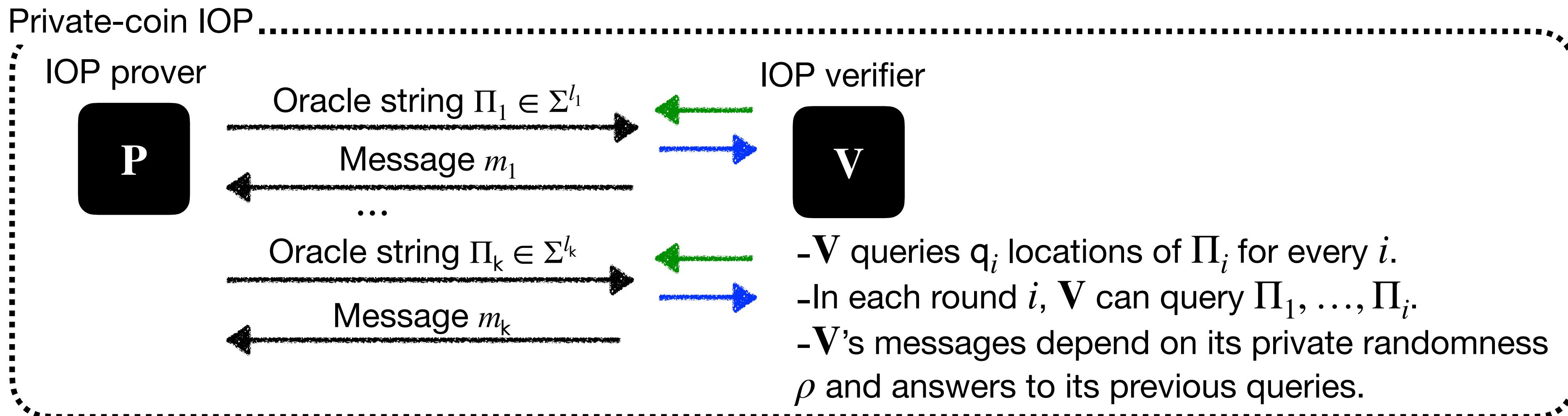
can improve to l_{\max} and q_{\max}

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

Beyond public-coin IOPs?

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

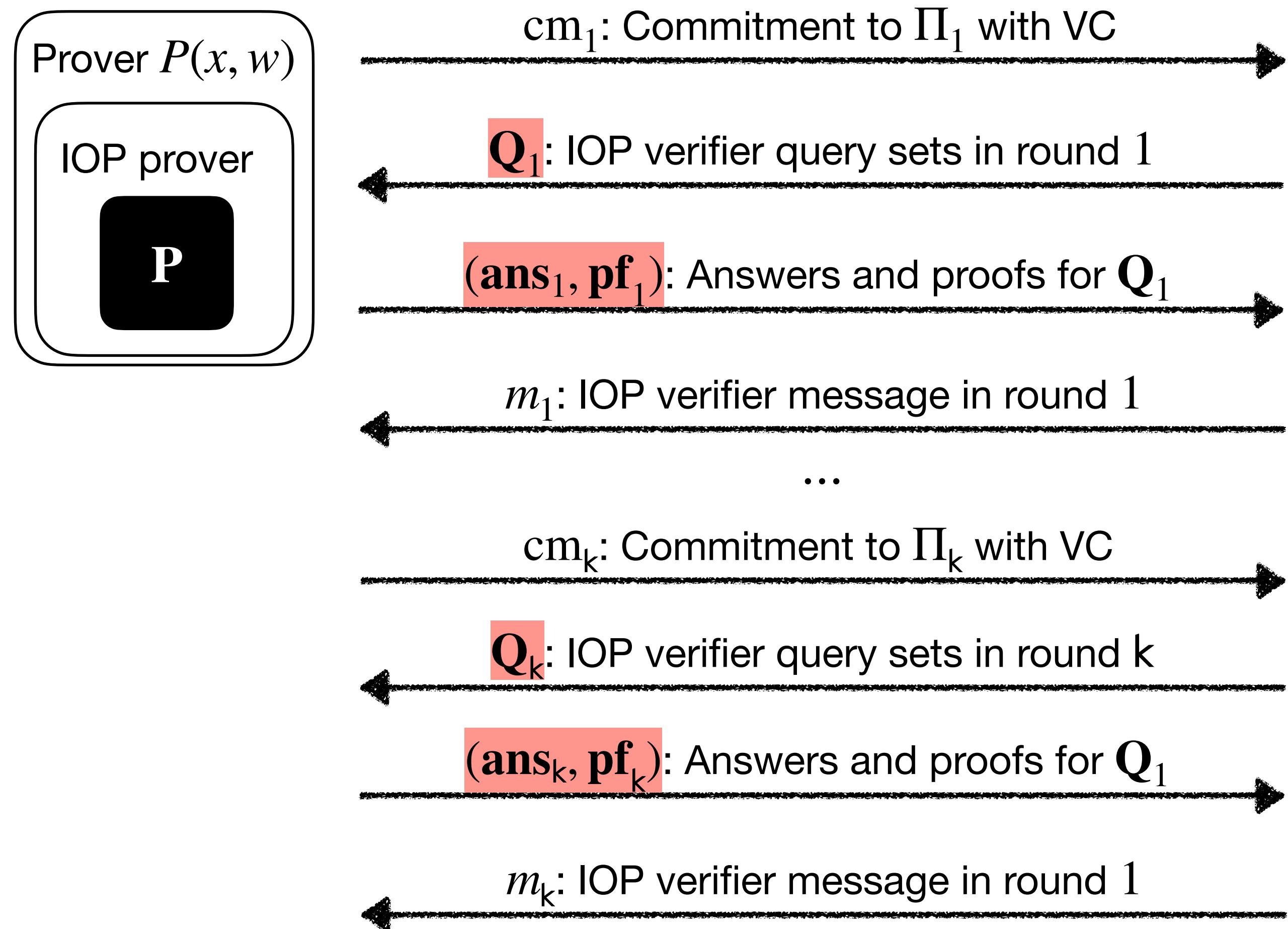
In general, a private-coin IOP looks like this:



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



Boldface because in each round i , Q_i contains verifier's queries to Π_1, \dots, Π_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?

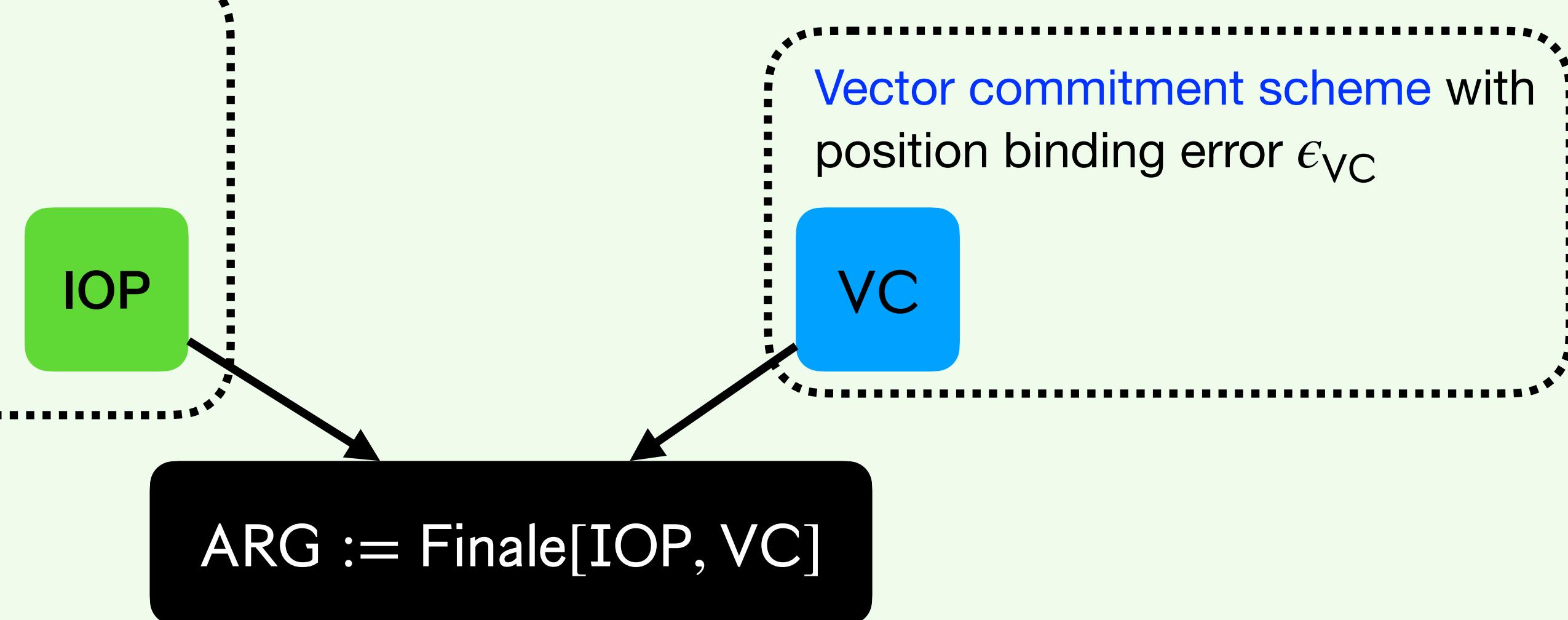
Our result on Finale protocol

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**

Random Continuation Sampler
(will define later)



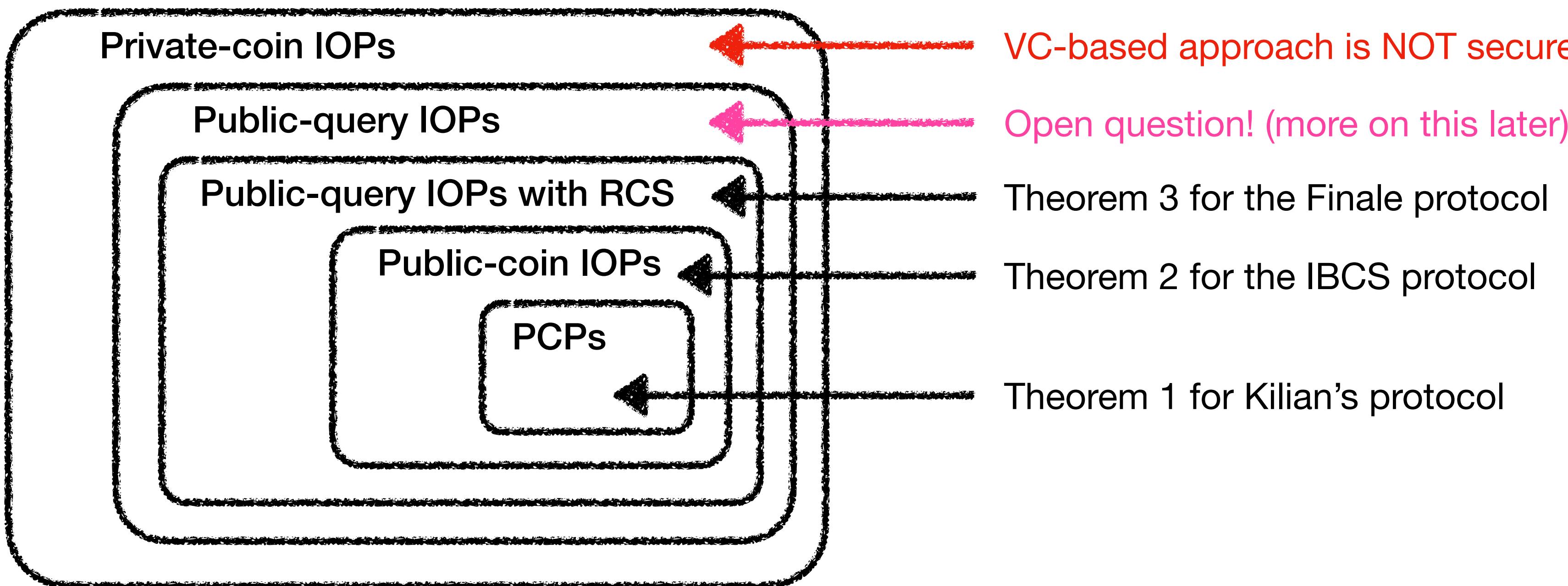
For every $x \notin L$ and $\epsilon > 0$,

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

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

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

Summary of results

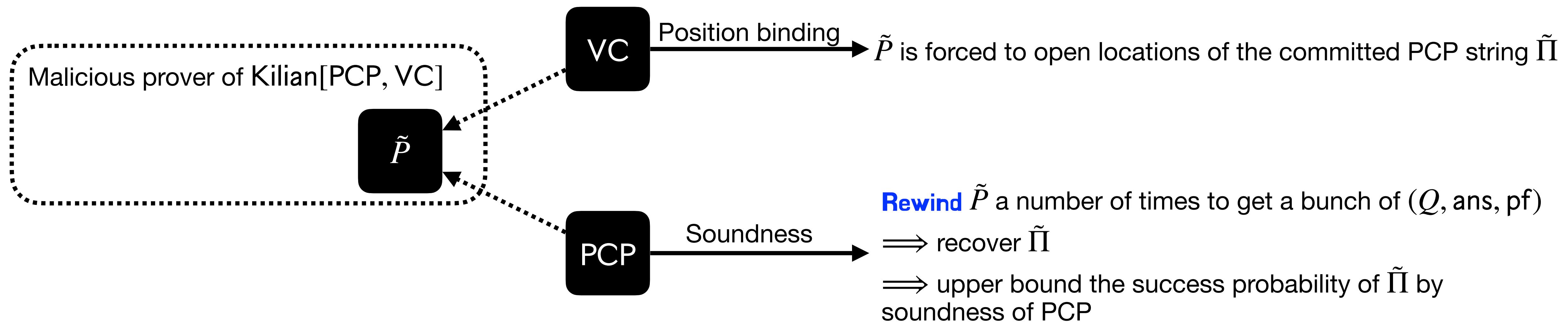


Kilian's protocol

Security from rewinding [1/2]

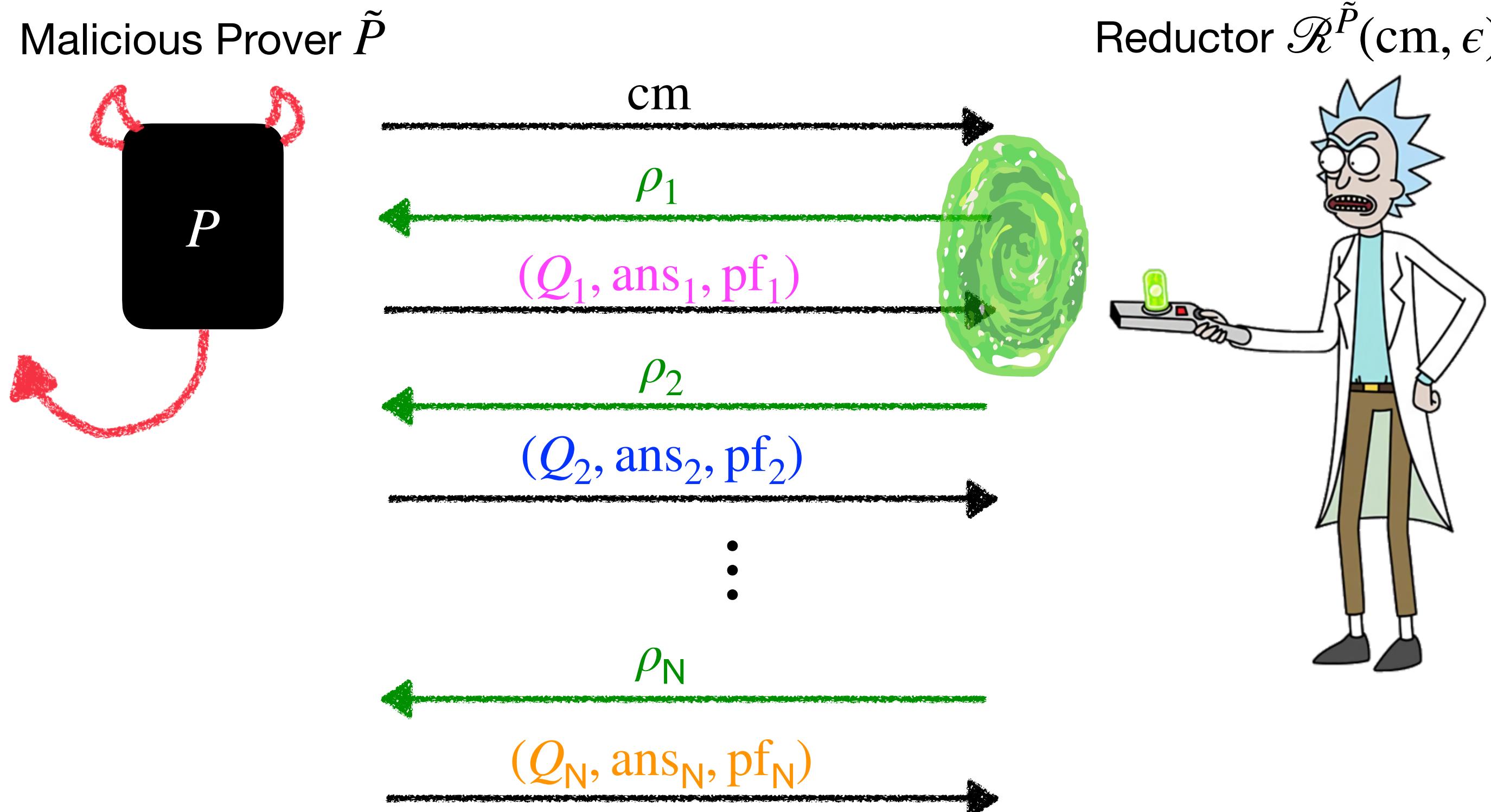
Goal: relate the soundness error of Kilian[PCP, VC]

to the soundness error of PCP and the position binding error of VC.

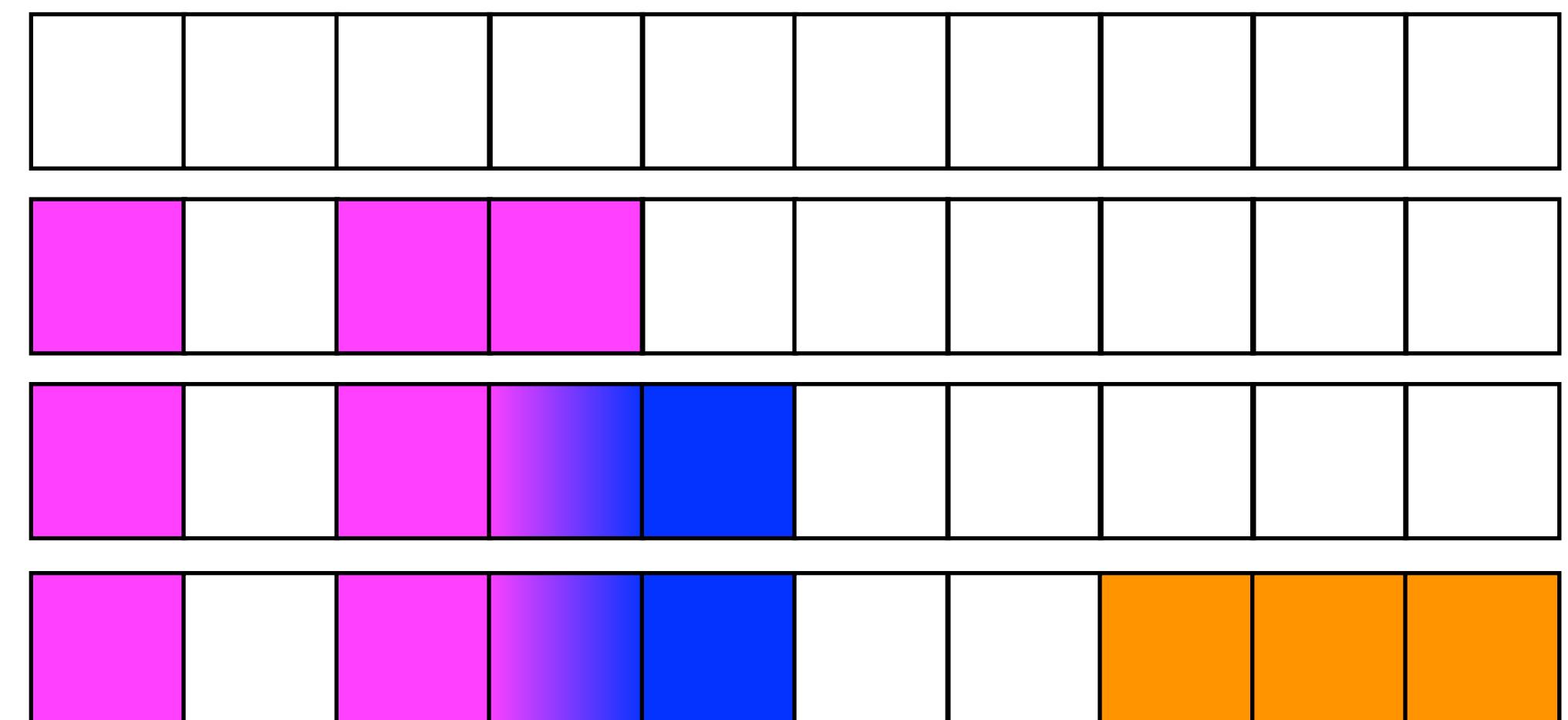


Security from rewinding [2/2]

How to rewind?



Recover $\tilde{\Pi}$



Soundness of Kilian's protocol

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

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

Soundness of PCP ✓
 $\implies \leq \epsilon_{\text{PCP}}(x)$

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

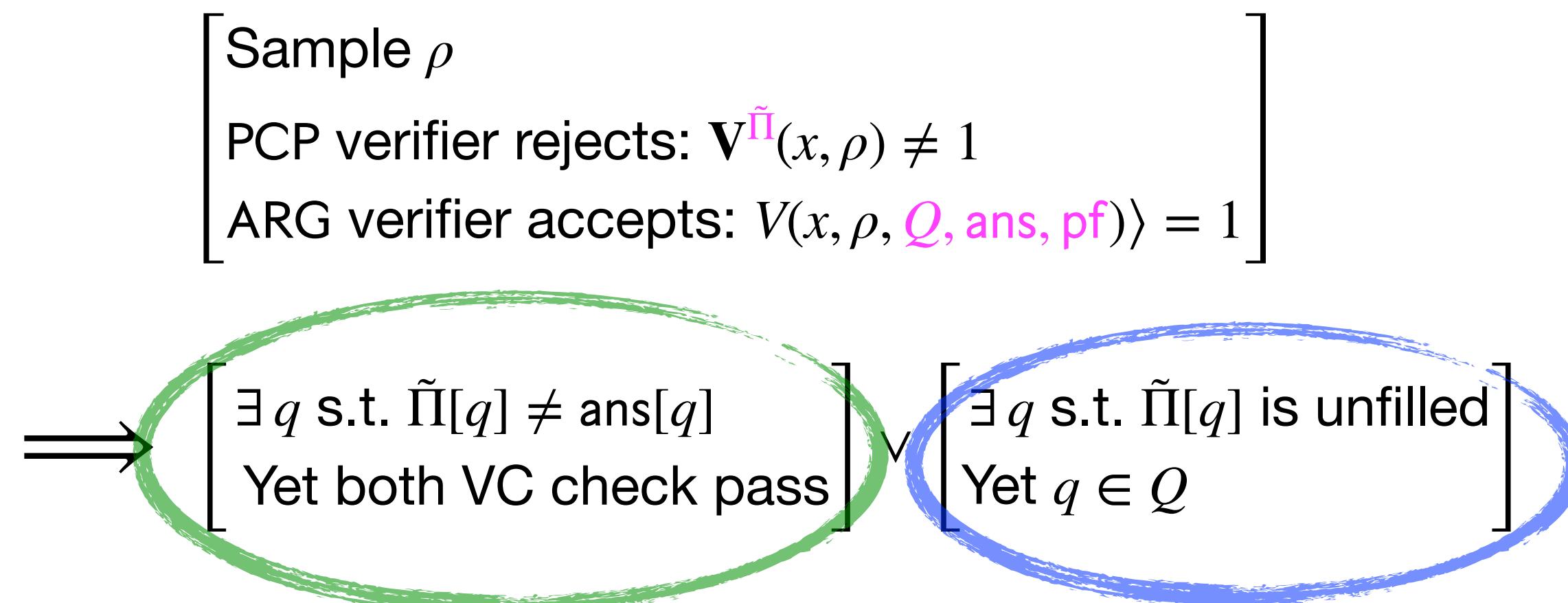
$$\Pr \left[\begin{array}{l} \text{Sample } \rho \\ \text{PCP verifier rejects: } V^{\tilde{\Pi}}(x, \rho) \neq 1 \\ \text{ARG verifier accepts: } V(x, \rho, Q, \text{ans}, \text{pf}) = 1 \end{array} \right]$$

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

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

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

Proof of the Security reduction lemma



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

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

Missing queries

- For each q , the probability that q is not queried by the reductor \mathcal{R} but is queried by the ARG verifier V is $1/N$:
 - Not hitting q for N times but hit it for the $(N + 1)$ -th time
- Probability that there exists such a $q \leq l/N$
- Setting $N := l/\epsilon \implies \leq \epsilon$
- t_{VC} also depends on N : VC adversary runs the reductor \mathcal{R}

Recap: Security of Kilian's protocol

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

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

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

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?

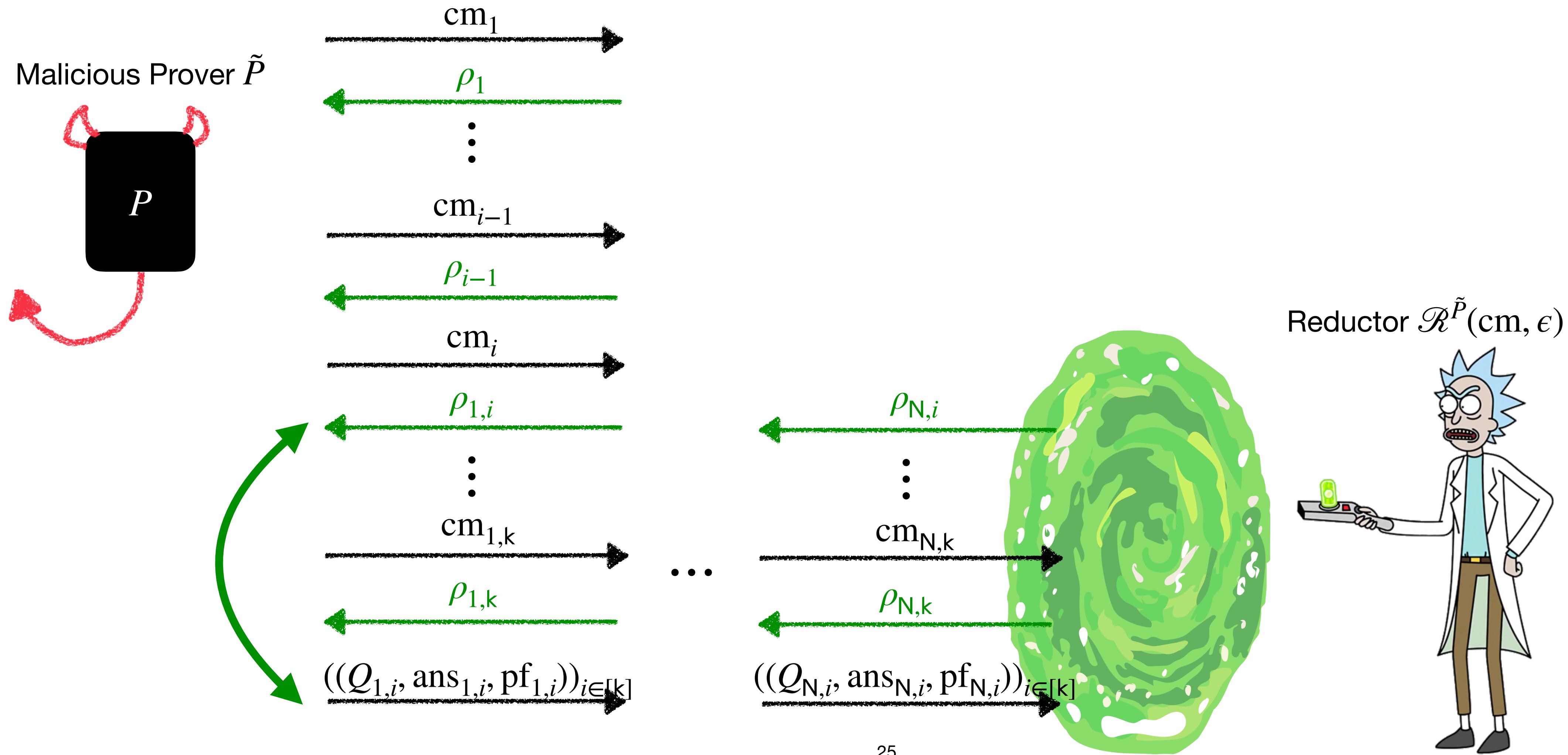
Why 30 years for a security proof of Kilian's protocol?

- 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

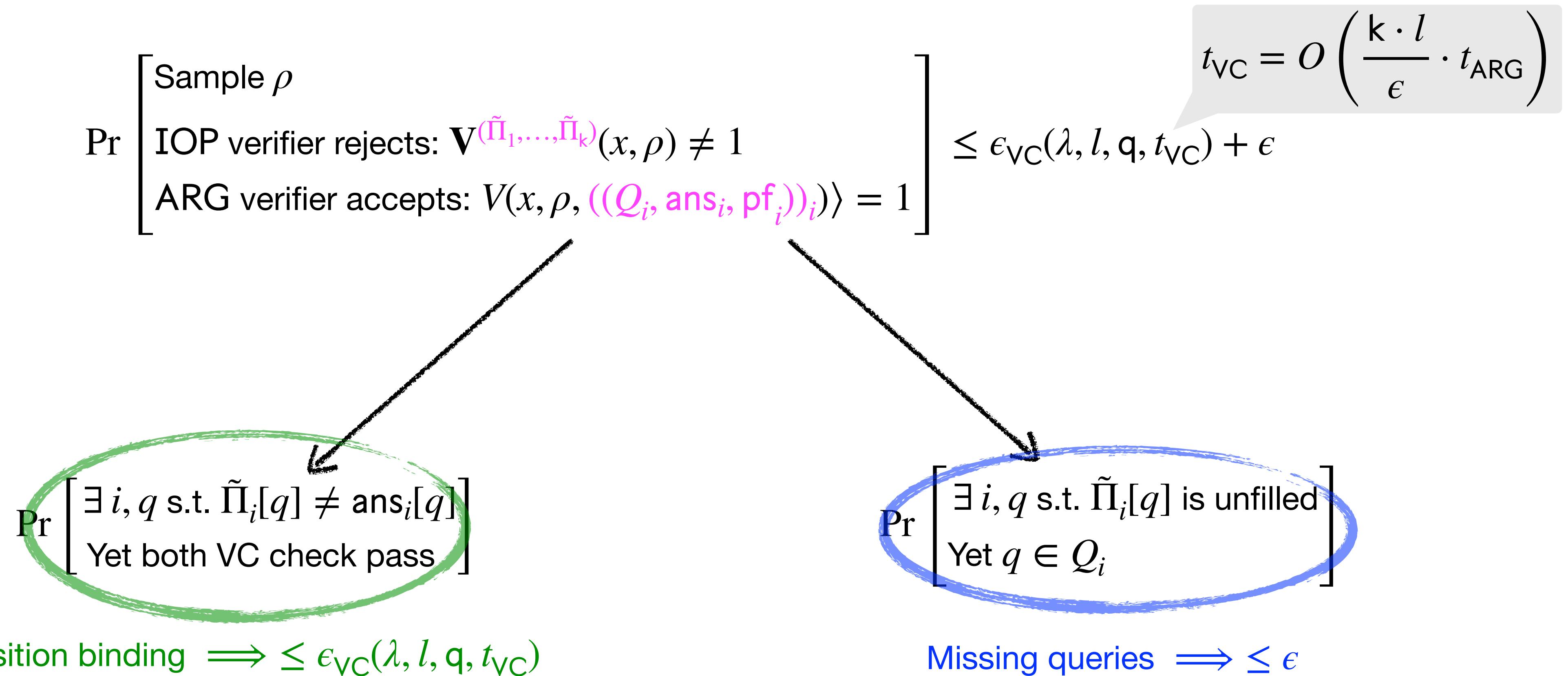
IBCS protocol

Security from rewinding

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



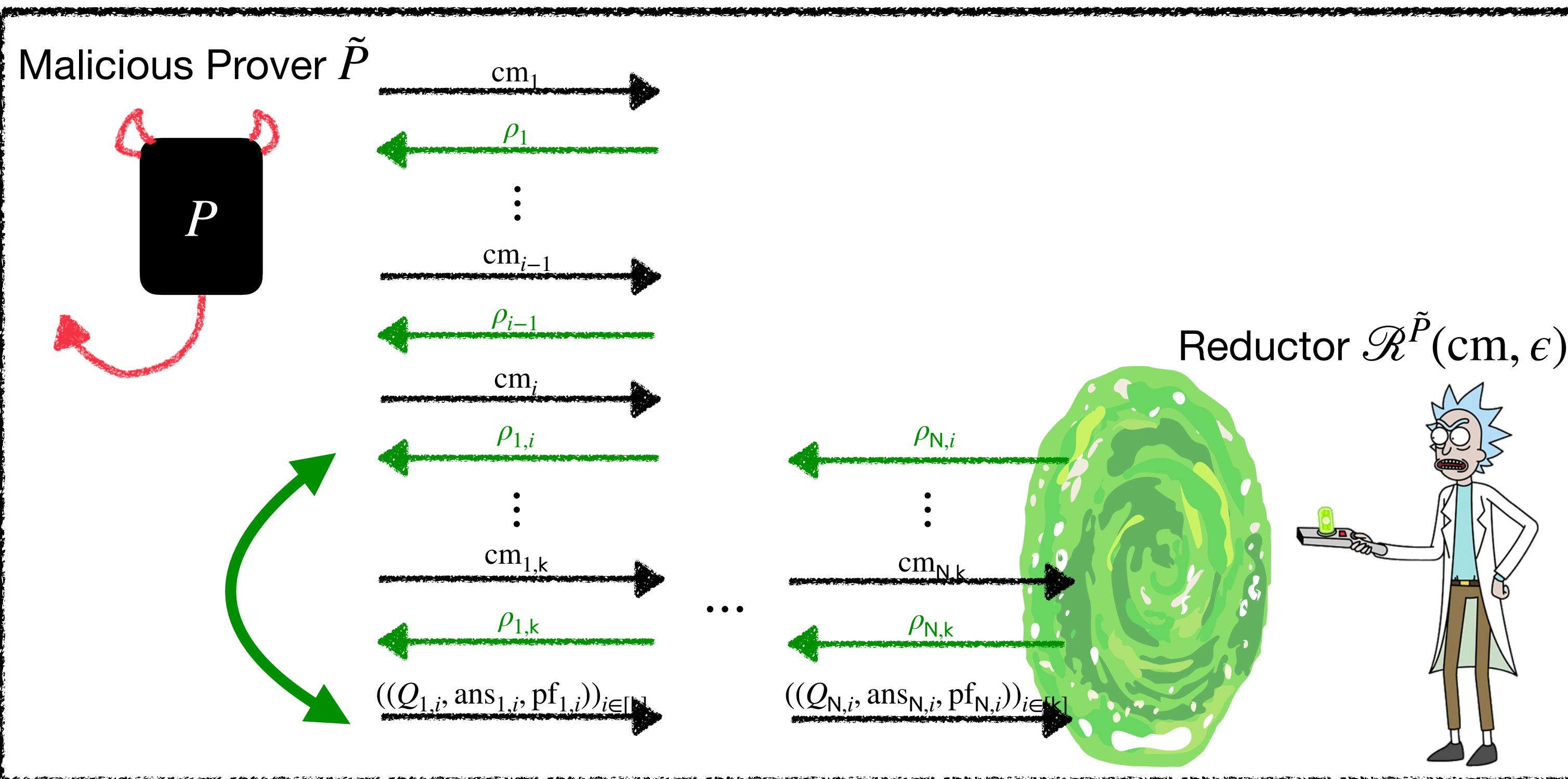
Security reduction lemma



How about private-coin IOPs?

Security from rewinding [1/2]

How to rewind?



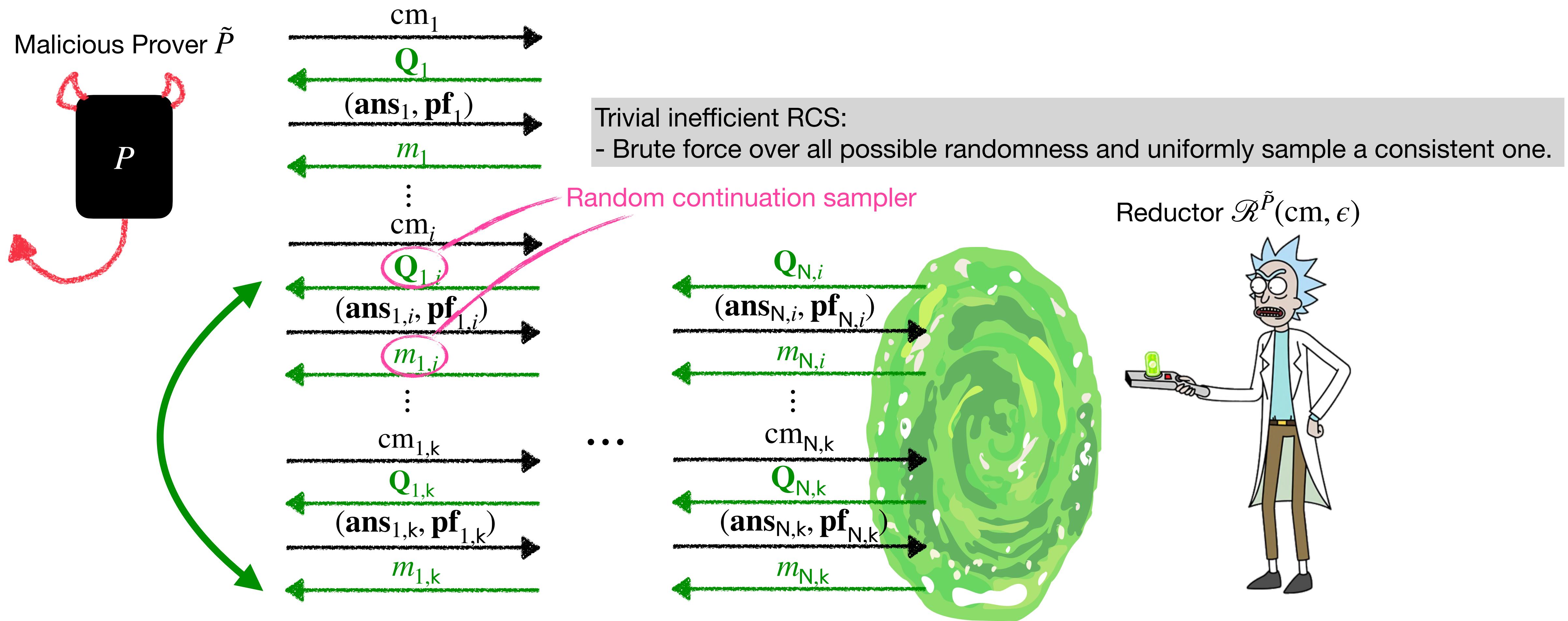
Key: the reductor \mathcal{R} must sample **consistent random continuations** of the argument interaction.

- 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 \mathcal{R} must finish the interaction consistently (with respect to the unknown private verifier randomness)

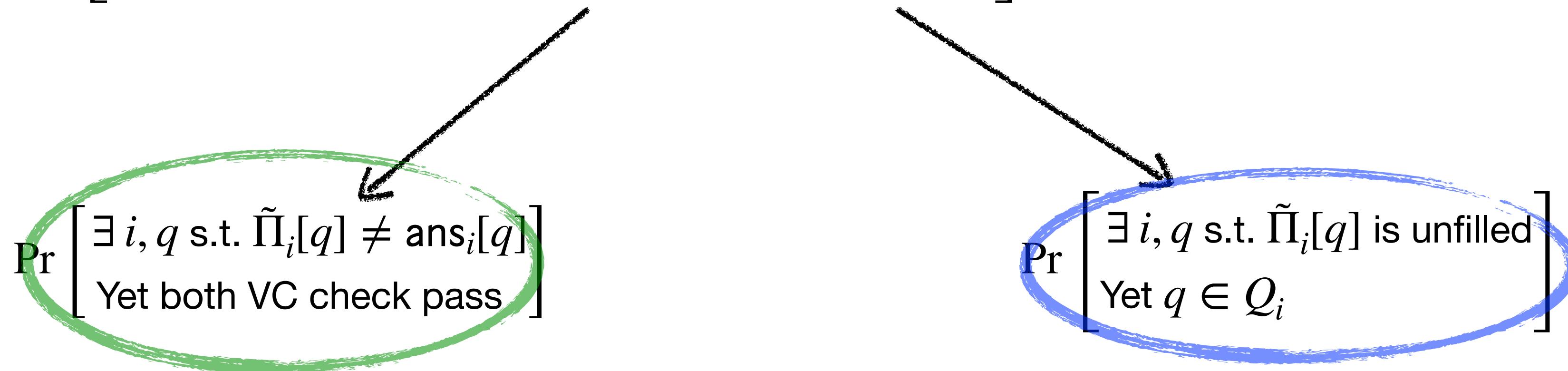
⇒ **Random continuation sampler (RCS)**



Security reduction lemma

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

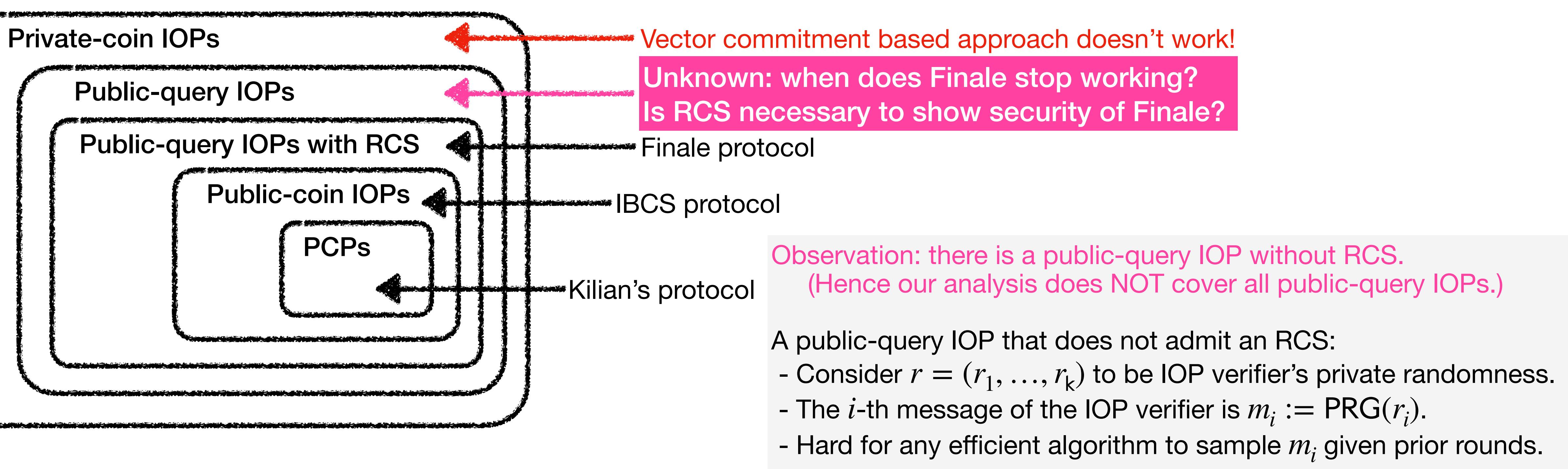
$$t_{\text{VC}} = O \left(\frac{k \cdot l}{\epsilon} \cdot (t_{\text{ARG}} + t_S) \right)$$



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

Missing queries $\implies \leq \epsilon$

Open question



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

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

A partial result: Finale[IOP, VC] has RCS iff IOP has RCS.

Thank you!

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