An Introduction to Commitment-Based Succinct Arguments

Alessandro Chiesa

EPFL
StarkWare

2025.07.14

Today's Protagonist:

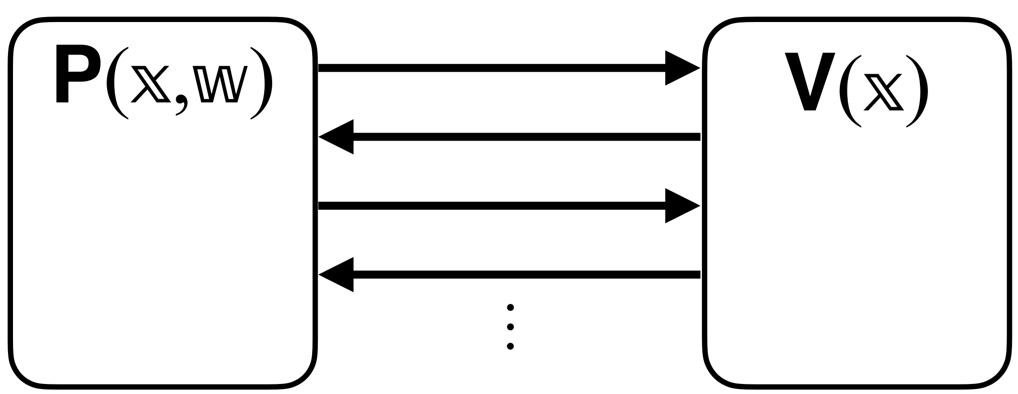
Succinct Arguments

*Cryptographic proofs for computation integrity that are **super short** and **super fast to verify**.*

Cryptographic Proofs aka **Arguments**

Fix a relation $R \coloneqq \{(\mathbb{x},\mathbb{w}) \mid \ldots\}$ and its language $L(R)$.

Interactive **Proof**



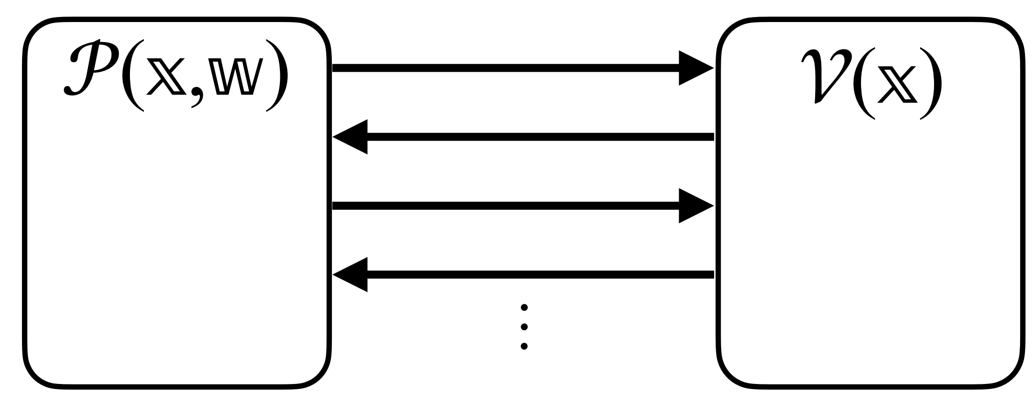
Completeness: $\forall\ (\mathbb{x},\mathbb{w}) \in R$

$$\Pr[\ \langle \mathbf{P}(\mathbb{x},\mathbb{w}), \mathbf{V}(\mathbb{x}) \rangle = 1\ ] = 1$$

Soundness: $\forall\ \mathbb{x} \notin L(R)$ $\forall\ \mathbf{A}$

$$\Pr[\ \langle \mathbf{A}, \mathbf{V}(\mathbb{x}) \rangle = 1\ ] \leq \epsilon$$

Interactive **Argument**



Completeness: $\forall\ (\mathbb{x},\mathbb{w}) \in R$

$$\Pr[\ \langle \mathcal{P}(\mathbb{x},\mathbb{w}), \mathcal{V}(\mathbb{x}) \rangle = 1\ ] = 1$$

Soundness: $\forall\ \mathbb{x} \notin L(R)$ $\forall$ efficient $\mathcal{A}$

$$\Pr[\ \langle \mathcal{A}, \mathcal{V}(\mathbb{x}) \rangle = 1\ ] \leq \epsilon$$

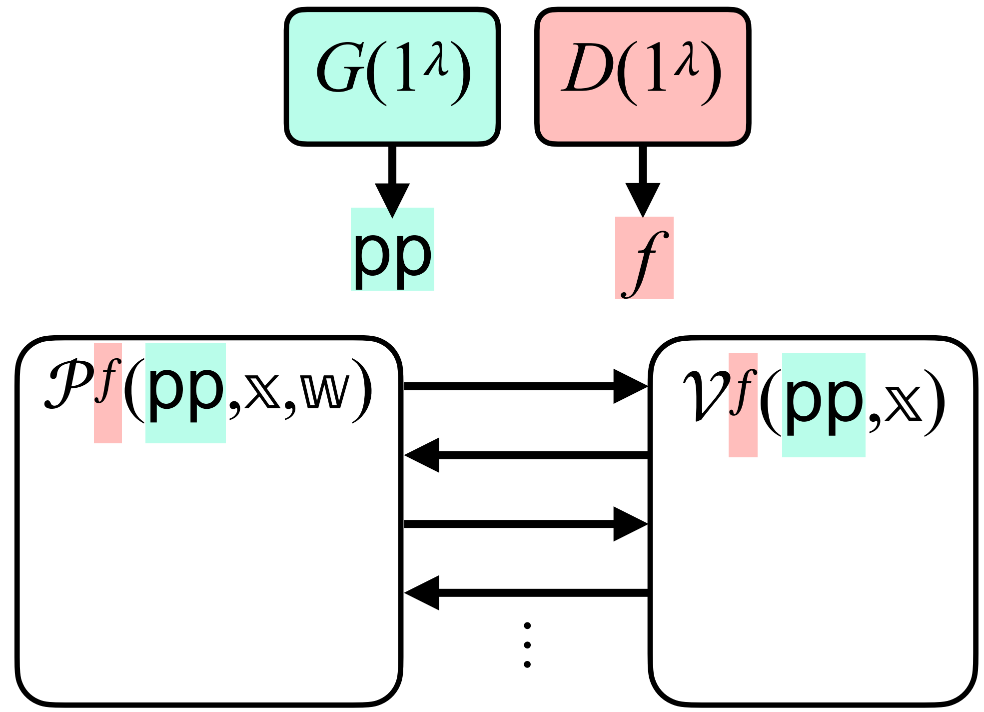
Cryptographic Proofs aka **Arguments**

A system setup is useful: $G(1^\lambda)$ samples public parameters pp.

Ditto for oracle(s) for idealized analyses: $D(1^\lambda)$ samples oracle(s) f.

Completeness: $\forall\ (\mathbb{x},\mathbb{w})\in R$

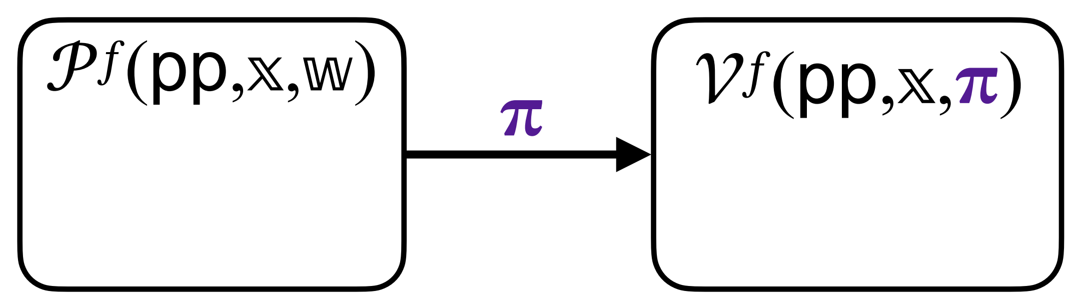
$$\Pr\left[\langle \mathcal{P}^f(\mathsf{pp},\mathbb{x},\mathbb{w}),\mathcal{V}^f(\mathsf{pp},\mathbb{x})\rangle = 1 \;\middle|\; \begin{array}{l} f \leftarrow D(1^\lambda) \\ \mathsf{pp} \leftarrow G(1^\lambda) \end{array}\right] = 1$$



Soundness: $\forall\ \mathbb{x}\notin L(R)\ \forall\ q$-query t-time $\mathcal{A}$

$$\Pr\left[\langle \mathcal{A}^f(\mathsf{pp}),\mathcal{V}^f(\mathsf{pp},\mathbb{x})\rangle = 1 \;\middle|\; \begin{array}{l} f \leftarrow D(1^\lambda) \\ \mathsf{pp} \leftarrow G(1^\lambda) \end{array}\right] \le \epsilon(\lambda, q, t)$$

Notable special case:
non-interactive arguments

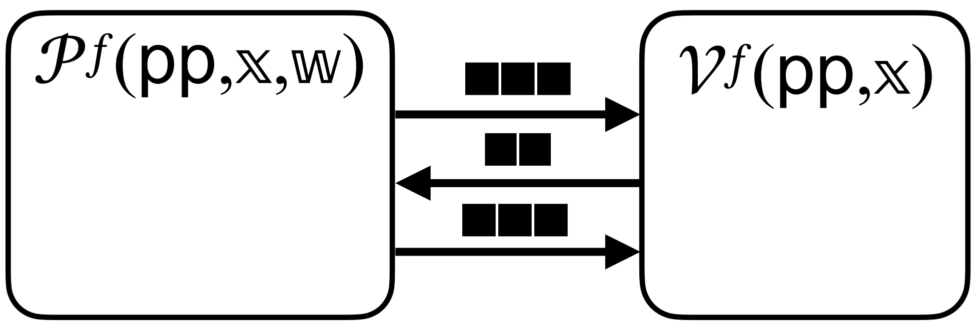


Not today: preprocessing, reductions, ...

Succinctness

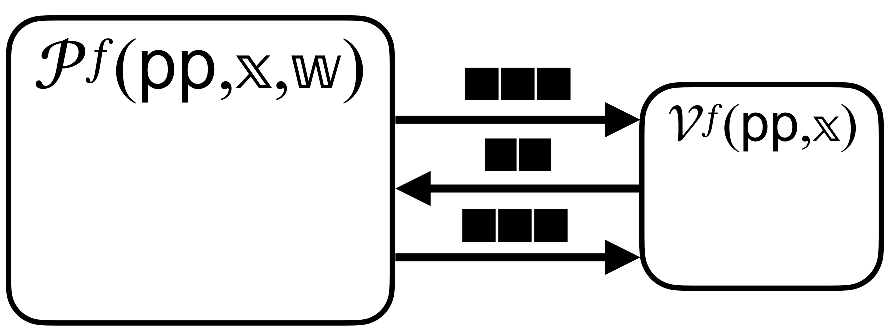
Range of definitions. Informally:

- succinctness in *communication*: $|\mathrm{cc}| \ll |\mathbb{w}|$



compression

- succinctness in *verification*: $\mathrm{time}(\mathcal{V}) \ll \mathrm{time}_R(\mathbb{x},\mathbb{w})$



speedup

Interactive proofs **CANNOT** be succinct (under standard complexity assumptions).

Example: $\mathrm{IP}[\mathrm{public}, \mathrm{cc}] \subseteq \mathrm{BPTIME}[2^{\mathrm{cc}}]$.

Not today: other useful properties like zero knowledge, knowledge soundness, ...

Results Landscape

Fundamental Succinctness Theorems (aka Hashing Suffices):

- Assuming CRH, ∃ **4**-message succinct arguments.
- In ROM, ∃ **1**-message succinct arguments.

Two main lines of research in last 15 years (to a first order).

1) Minimizing assumptions to achieve succinctness.

- succinct *interactive* arguments ← multi-CRH (rather than CRH)

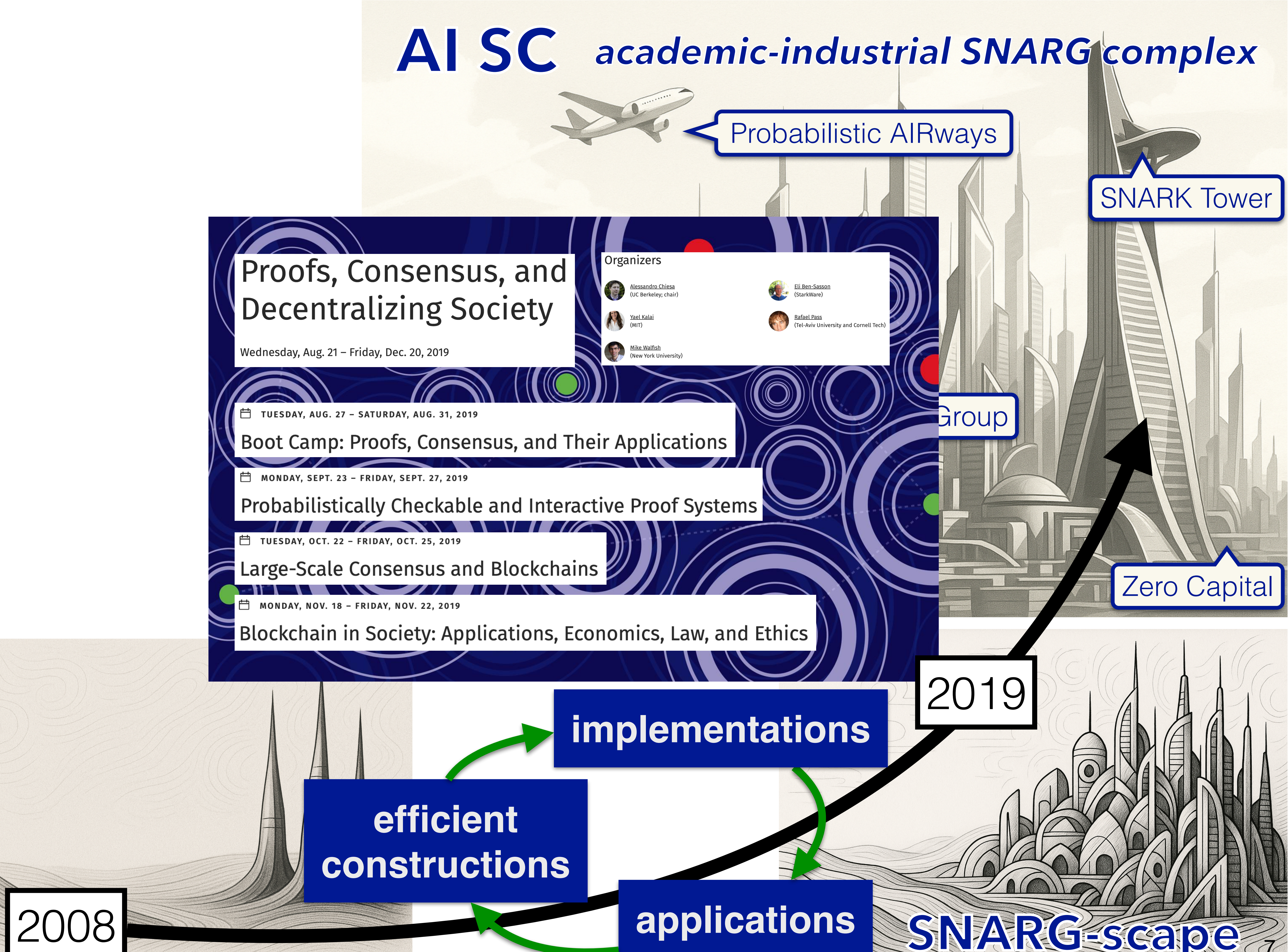
- succinct *non-interactive* arguments ← falsifiable assumptions (no ROM)

Open: adaptive SNARGs from LWE (requires non-black-box reductions)

2) Minimizing (asymptotic&concrete) cost of succinct (non-interactive) arguments, given oracle models or non-falsifiable assumptions (or both).

This talk

Highly-efficient constructions+implementations that aim for the best communication complexity and prover/verifier time/space.


AI SC
academic-industrial SNARG complex
Probabilistic AIRways
SNARK Tower
Group
Zero Capital
Proofs, Consensus, and Decentralizing Society
Wednesday, Aug. 21 – Friday, Dec. 20, 2019
Organizers
Alessandro Chiesa (UC Berkeley; chair)
Eli Ben-Sasson (StarkWare)
Yael Kalai (MIT)
Rafael Pass (Tel-Aviv University and Cornell Tech)
Mike Walfish (New York University)
TUESDAY, AUG. 27 – SATURDAY, AUG. 31, 2019
Boot Camp: Proofs, Consensus, and Their Applications
MONDAY, SEPT. 23 – FRIDAY, SEPT. 27, 2019
Probabilistically Checkable and Interactive Proof Systems
TUESDAY, OCT. 22 – FRIDAY, OCT. 25, 2019
Large-Scale Consensus and Blockchains
MONDAY, NOV. 18 – FRIDAY, NOV. 22, 2019
Blockchain in Society: Applications, Economics, Law, and Ethics
2019
implementations
efficient constructions
applications
2008
SNARG-scape

Basic Anatomy

Two Complementary Tools

Computational Complexity • **Cryptography**

(some form of)
Probabilistic Proof

- security vs. inefficient adversaries
- succinct verification in a query model

(some form of)
Commitment Scheme

- security vs. efficient adversaries
- succinct commitment/opening for query model

Determines
the type of computation
(eg. machine vs circuit)

Commit-Then-Open
Transformation(s)

Determines
- cryptographic costs
- setup (public vs private)
- pre- vs post-quantum

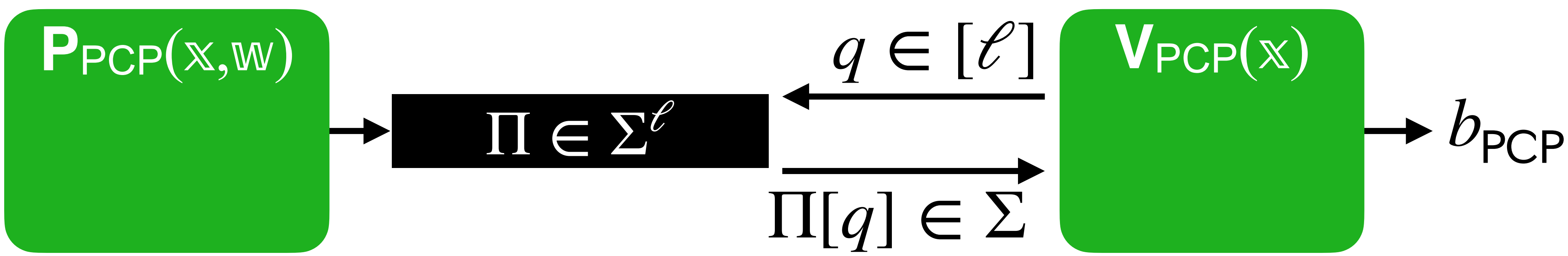
Succinct Argument

Not in this talk:
- linear-only encodings
- sumcheck arguments
- ...

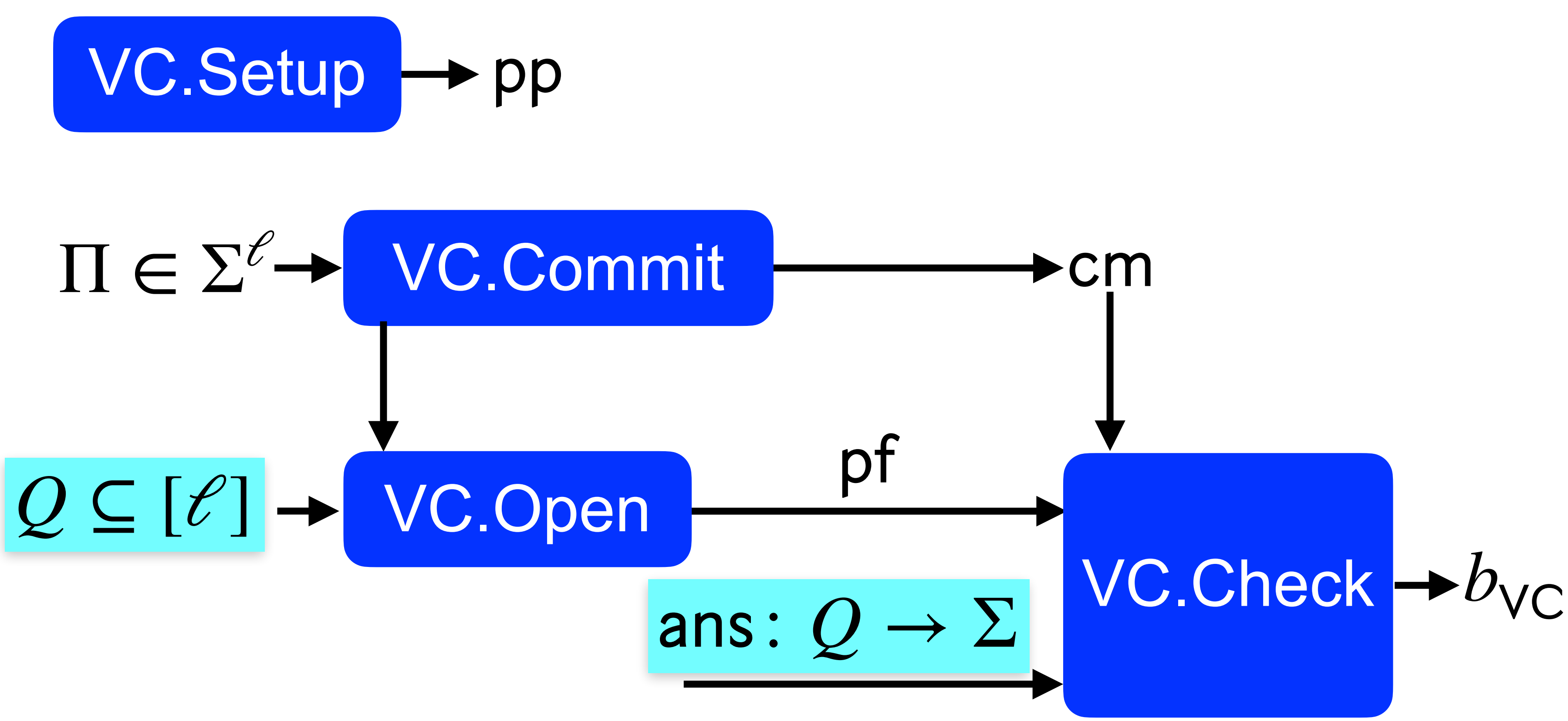
Part I:
The Interactive Case

Kilian Protocol: Tools

Tool #1: probabilistically checkable proof (PCP)

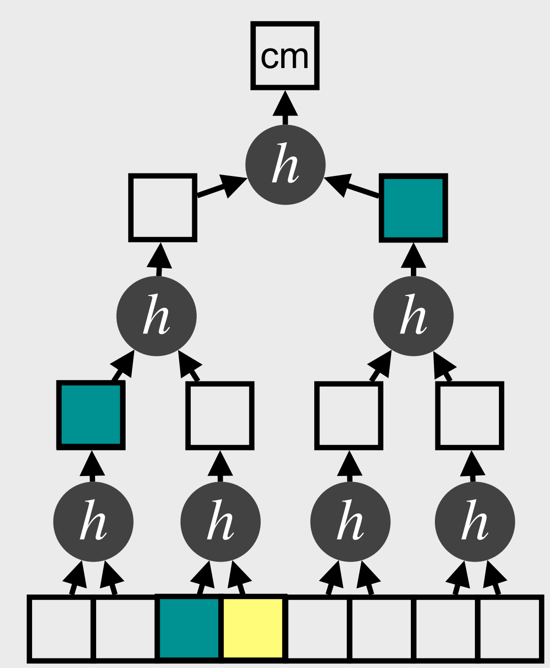


Tool #2: vector commitment scheme (VC)



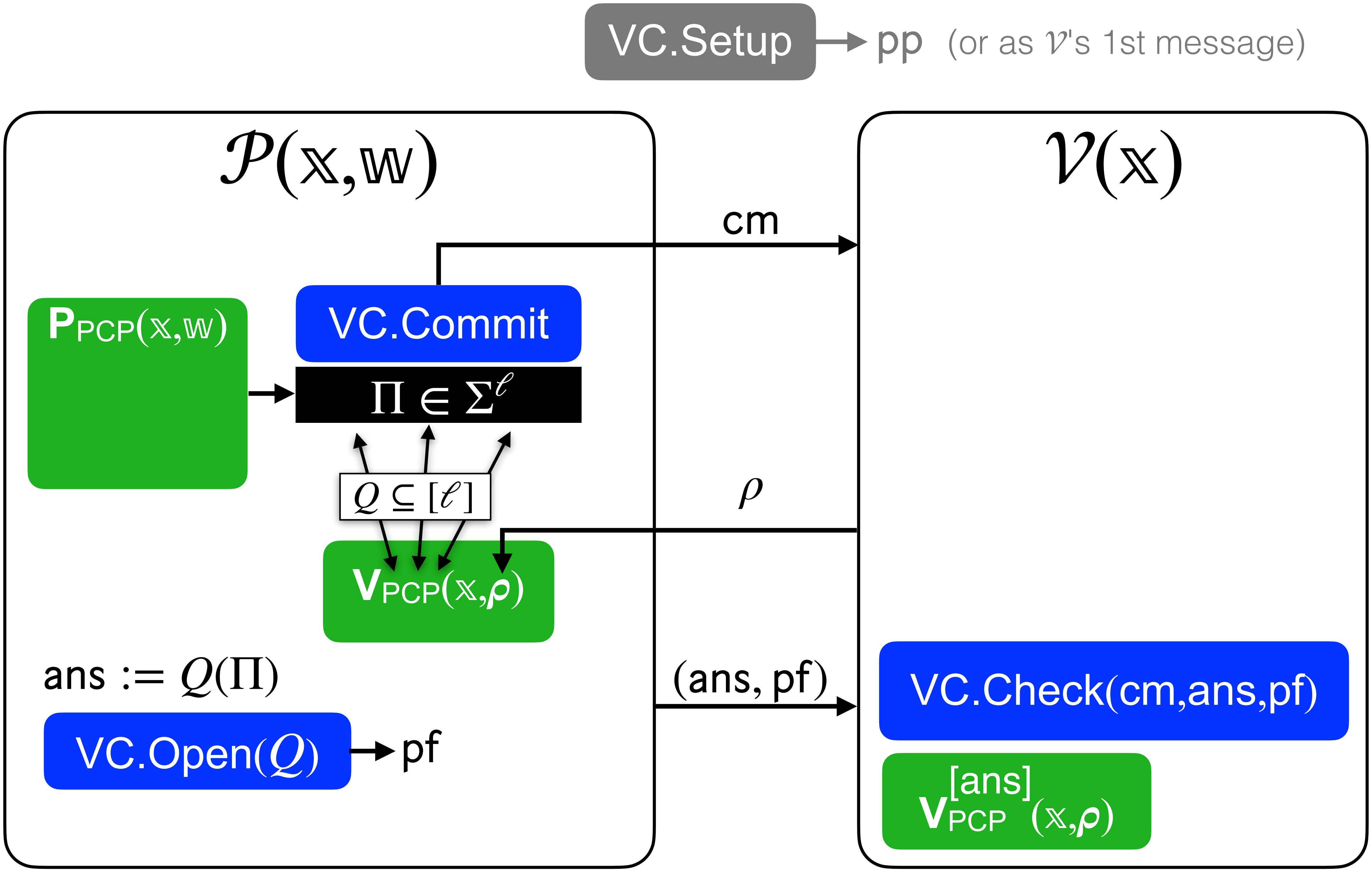
Example:
Merkle Commitment

pp $= h\colon \{0,1\}^{2\lambda} \to \{0,1\}^{\lambda}$



Kilian Protocol

VC-commit to the PCP string.
Then VC-open the queried locations of the PCP string.



Kilian Protocol: Security

Note: extraction
↓
binding

Two incomparable security analyses.

- **Black-box prover rewinding (no oracles):**

$$\forall\, \epsilon > 0 \quad \epsilon_{\mathsf{ARG}}(t_{\mathsf{ARG}}) \le \epsilon_{\mathsf{PCP}} + \epsilon_{\mathsf{VC}}\left(\frac{\ell \cdot t_{\mathsf{ARG}}}{\epsilon}\right) + \epsilon$$

PCP soundness error

VC position-binding error

Tradeoff seems inherent.

$$\Pr\left[\begin{array}{l} \forall\, i \;\; \mathsf{VC.Check}_{\mathsf{pp}}(\mathsf{cm},\mathsf{ans}_i,\mathsf{pf}_i)=1 \\ \nexists\, \Pi \in \Sigma^{\ell}\;\; \forall\, i \;\; \Pi[Q_i] = \mathsf{ans}_i \end{array} \middle| \begin{array}{c} \mathsf{pp} \leftarrow \mathsf{VC.Setup}(1^{\lambda}) \\ \mathsf{cm} \\ ((\mathsf{ans}_i,\mathsf{pf}_i))_i \leftarrow A(\mathsf{pp}) \end{array} \right] \le \epsilon_{\mathsf{VC}}(t_{\mathsf{VC}})$$

Or in expected time: $\epsilon^{\mathbb{E}}_{\mathsf{ARG}}(t^{\mathbb{E}}_{\mathsf{ARG}}) \le \epsilon_{\mathsf{PCP}} + \epsilon^{\mathbb{E}}_{\mathsf{VC}}\left(\frac{\ell \cdot t^{\mathbb{E}}_{\mathsf{ARG}}}{1 - \epsilon_{\mathsf{PCP}}}\right)$

- **Straightline extraction in ideal model (with oracles):**

$$\epsilon_{\mathsf{ARG}}(q_{\mathsf{ARG}}) \le \epsilon_{\mathsf{PCP}} + \kappa_{\mathsf{VC}}(q_{\mathsf{ARG}})$$

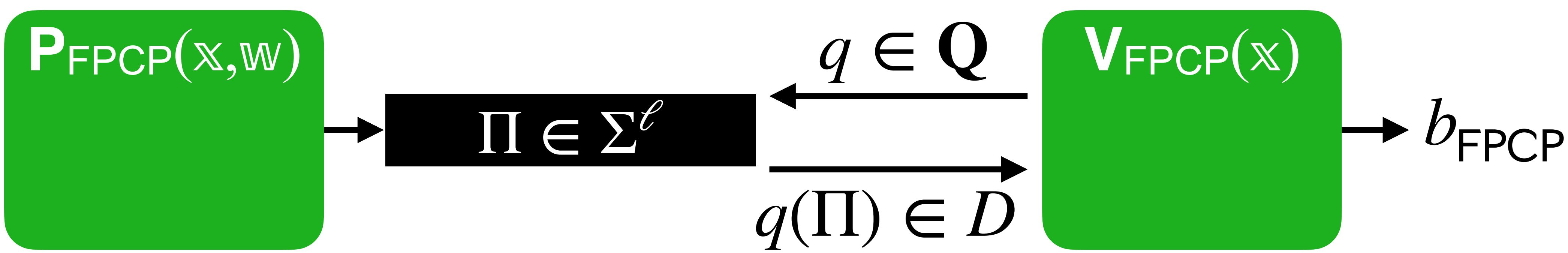
VC straightline-extraction error

$$\Pr\left[\begin{array}{l} \mathsf{VC.Check}^{f}(\mathsf{cm},\mathsf{ans},\mathsf{pf}) = 1 \\ \Pi[Q] \neq \mathsf{ans} \end{array} \middle| \begin{array}{l} (\mathsf{cm},\mathsf{aux}) \xleftarrow{\mathsf{tr}} A^{f}(\mathsf{pp}) \\ \Pi \leftarrow \mathsf{VC.Extract}(\mathsf{pp},\mathsf{cm},\mathsf{tr}) \\ (\mathsf{ans},\mathsf{pf}) \leftarrow A^{f}(\mathsf{aux}) \end{array} \right] \le \kappa_{\mathsf{VC}}(q_{\mathsf{VC}})$$

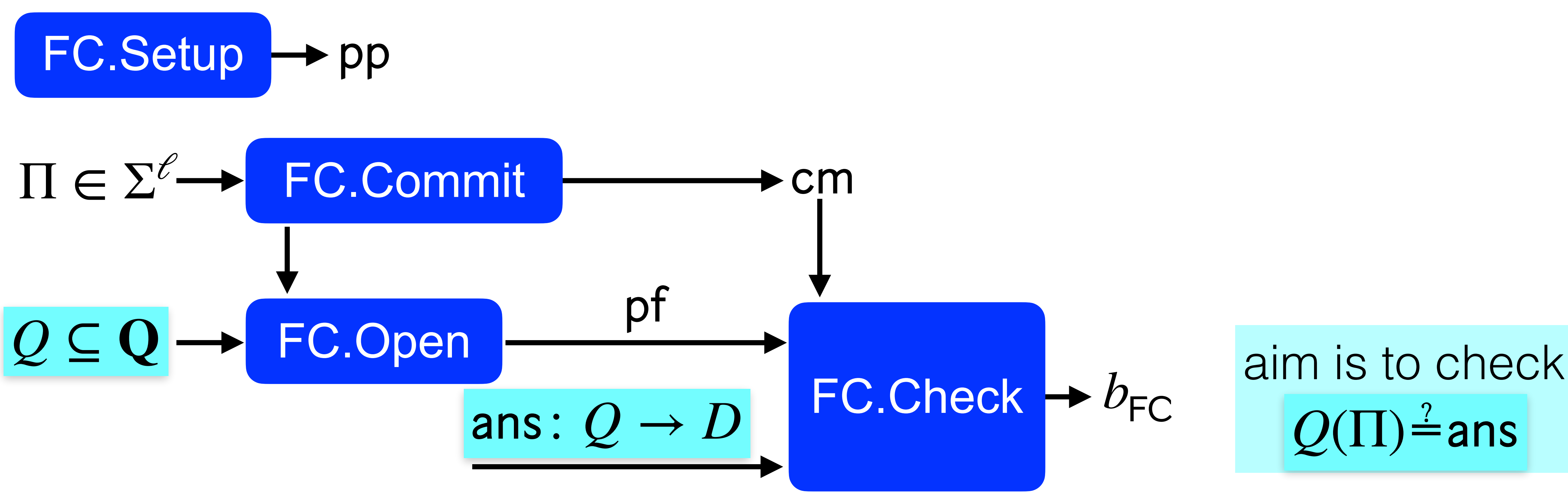
From Point Queries to Function Queries

A **query class** $\mathbf{Q}$ for proof strings in Σ^{ℓ} is a set of functions $q \colon \Sigma^{\ell} \to D$.

Tool #1: $\mathbf{Q}$-functional probabilistically checkable proof (FPCP)

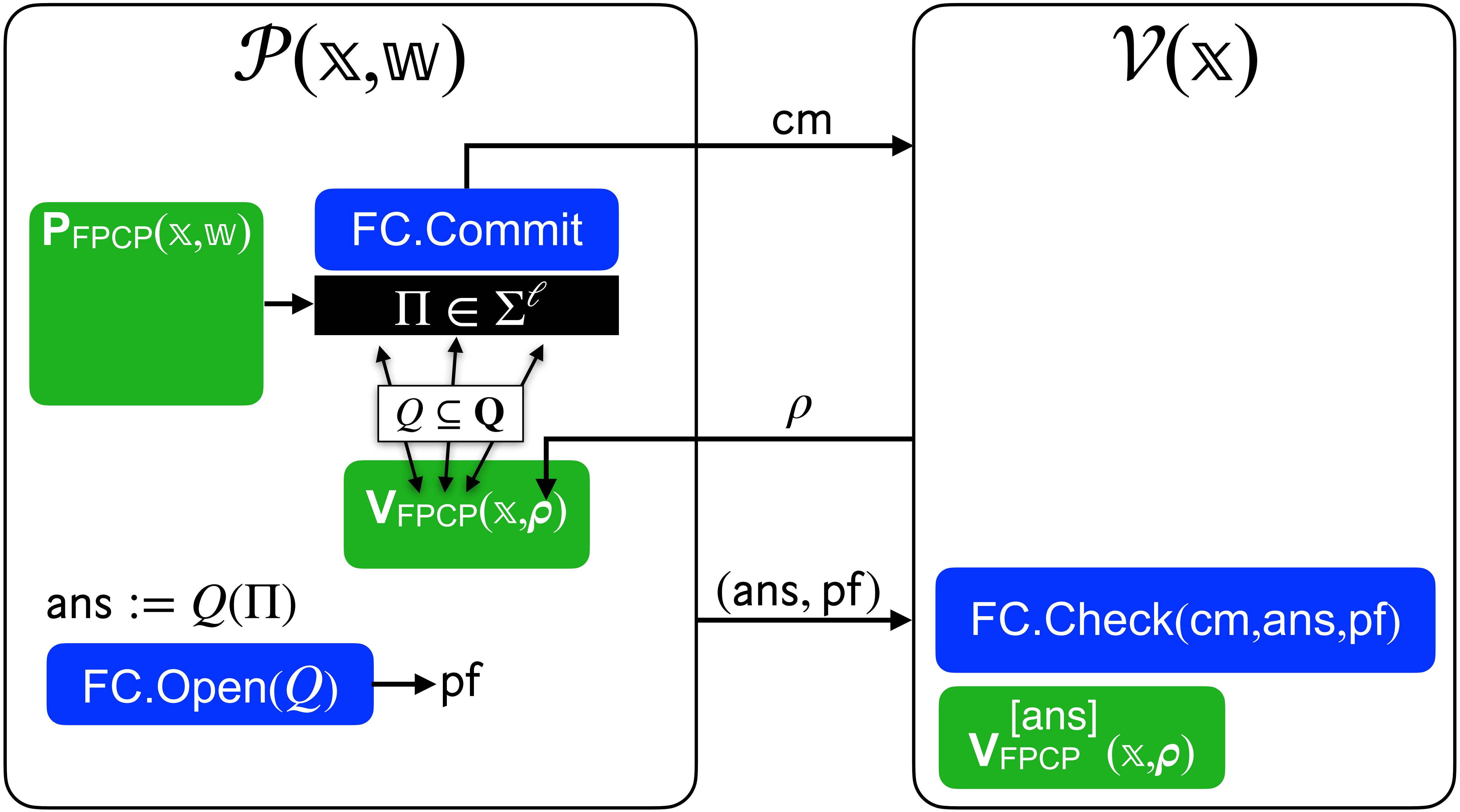


Tool #2: $\mathbf{Q}$-functional commitment scheme (FC)



Functional Extension of the Kilian Protocol

The commit-then-open strategy **extends** to any query class $\mathbf{Q}$.



Examples of Query Classes

A **query class** $\mathbf{Q}$ for proof strings in Σ^{ℓ} is a set of functions $q\colon \Sigma^{\ell} \to D$.

$\mathbf{Q}_{\text{point}}$	$\Pi \in \Sigma^{\ell}$	$q(\Pi) = \Pi[i] \in \Sigma$ for $i \in [\ell]$
$\mathbf{Q}_{\text{linear}}$	$\Pi \in \mathbb{F}^{n}$	$q(\Pi) = \sum_{i \in [n]} \Pi[i]\alpha[i] \in \mathbb{F}$ for $\alpha \in \mathbb{F}^{n}$
$\mathbf{Q}_{\text{upoly}}$ univariate polynomial	$\Pi \in \mathbb{F}^{d}$	$q(\Pi) = \sum_{i \in [d]} \Pi[i]\alpha^{i-1} \in \mathbb{F}$ for $\alpha \in \mathbb{F}$
$\mathbf{Q}_{\text{mlpoly}}$ multilinear polynomial	$\Pi \in \mathbb{F}^{2^n}$	$q(\Pi) = \sum_{b \in \{0,1\}^n} \Pi[b]\alpha^{b} \in \mathbb{F}$ for $\alpha \in \mathbb{F}^{n}$
$\mathbf{Q}_{\text{spoly}}$ "structured" polynomial	$\Pi \in (\mathbb{F}^{d})^{m+n}$ $= (f_1, \ldots, f_m, g_1, \ldots, g_n)$	$q(\Pi) = \sum_{k \in [n]} h_k(f_1(\alpha), \cdots, f_m(\alpha))g_k(\alpha) \in \mathbb{F}$ for $h_1, \ldots, h_k \in \mathbb{F}^{\leq d_h}[X_1, \ldots, X_m]$ and $\alpha \in \mathbb{F}$

These make the Commit-Then-Open Transformation extremely **flexible**.

Security For Any Query Class

Note:
extraction
↓
binding

Here too there are two incomparable security analyses.

- **Black-box prover rewinding (no oracles):**

solver time for $\mathbf{Q}$

$$\forall N \quad \epsilon_{\mathsf{ARG}}(t_{\mathsf{ARG}}) \leq \epsilon_{\mathsf{FPCP}} + \epsilon_{\mathsf{FC}}(N \cdot t_{\mathsf{ARG}} + t_{\mathbf{Q}}) + \epsilon_{\mathbf{Q}}(N)$$

FPCP soundness error

FC function-binding error

$$\Pr\left[\begin{array}{l} \forall\, i \;\; \mathsf{FC.Check}_{\mathsf{pp}}(\mathsf{cm},\mathsf{ans}_i,\mathsf{pf}_i)=1 \\ \nexists\, \Pi \in \Sigma^{\ell}\;\; \forall\, i\;\; Q_i(\Pi) = \mathsf{ans}_i \end{array} \;\middle|\; \begin{array}{c} \mathsf{pp} \leftarrow \mathsf{FC.Setup}(1^{\lambda}) \\ \mathsf{cm} \\ ((\mathsf{ans}_i,\mathsf{pf}_i))_i \leftarrow A(\mathsf{pp}) \end{array}\right] \leq \epsilon_{\mathsf{FC}}(t_{\mathsf{FC}})$$

tail error for $\mathbf{Q}$
max probability that after
N+1 query-answer samples
(i) there is consistent $\Pi \in \Sigma^{\ell}$
(ii) last sample "adds new info"

- **Straightline extraction in ideal model (with oracles):**

$$\epsilon_{\mathsf{ARG}}(q_{\mathsf{ARG}}) \leq \epsilon_{\mathsf{FPCP}} + \kappa_{\mathsf{FC}}(q_{\mathsf{ARG}})$$

FC straightline-extraction error

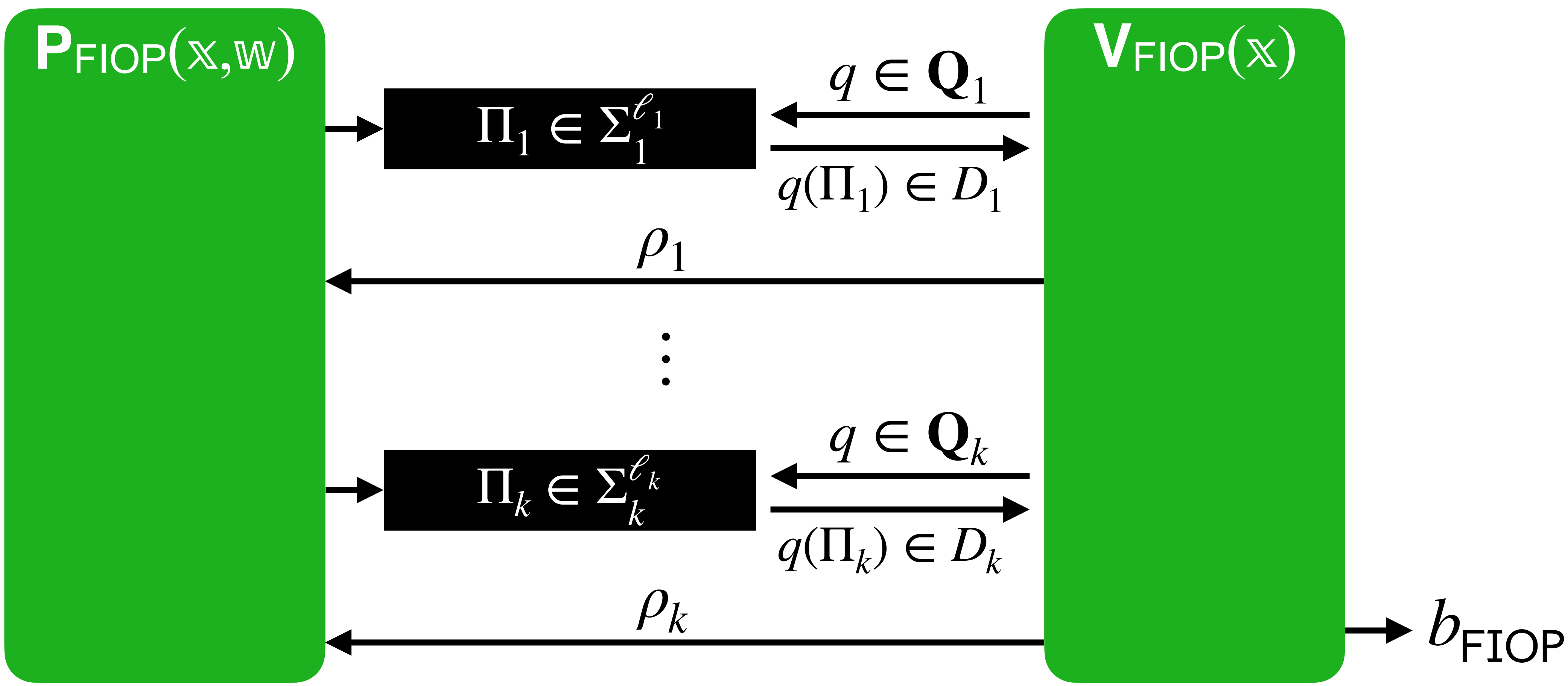
$$\Pr\left[\begin{array}{l} \mathsf{FC.Check}^{f}(\mathsf{cm},\mathsf{ans},\mathsf{pf}) = 1 \\ Q(\Pi) \neq \mathsf{ans} \end{array} \;\middle|\; \begin{array}{l} (\mathsf{cm},\mathsf{aux}) \xleftarrow{\mathsf{tr}} A^{f}(\mathsf{pp}) \\ \Pi \leftarrow \mathsf{FC.Extract}(\mathsf{pp},\mathsf{cm},\mathsf{tr}) \\ (\mathsf{ans},\mathsf{pf}) \leftarrow A^{f}(\mathsf{aux}) \end{array}\right] \leq \kappa_{\mathsf{FC}}(q_{\mathsf{FC}})$$

More Rounds → More Efficiency

Probabilistic proofs with **more rounds** are **VASTLY** more efficient.

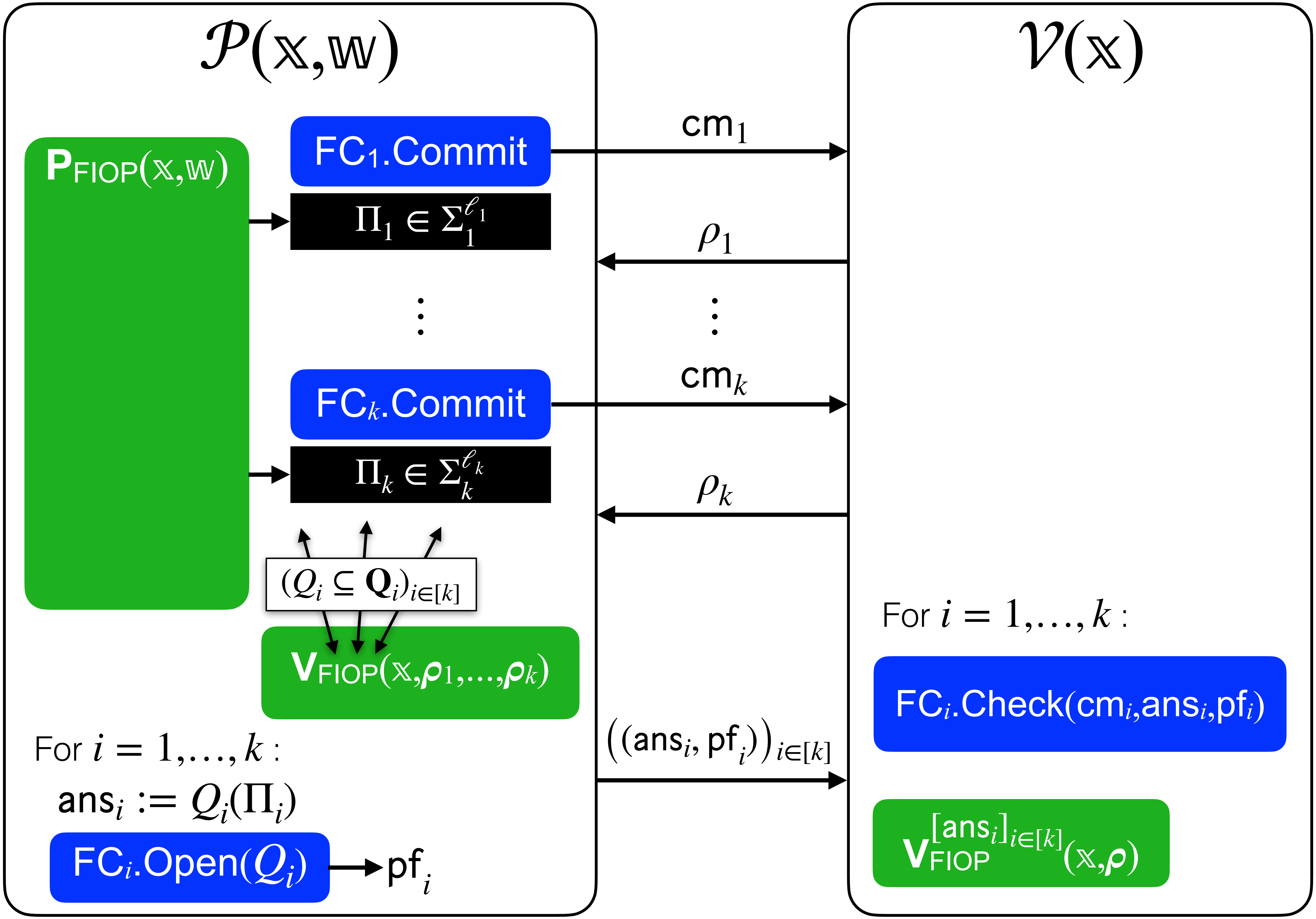
Interaction allows: sumcheck protocol, divide-and-combine, out-of-domain-sampling, permutation tests, ...

Consinder a $\mathbf{Q}$-functional interactive oracle proof (FIOP):



The Commit-Then-Open Transformation is adapted (following BCS).

Commit-Then-Open For Each Round



Security of (Generalized) Interactive BCS

Security reductions extend (with more work...) to **any number of rounds**.

- **Black-box prover rewinding (no oracles):**

solver time for $\mathbf{Q}_i$

$$\forall N \quad \epsilon_{\mathsf{ARG}}(t_{\mathsf{ARG}}) \leq \epsilon_{\mathsf{FIOP}} + \sum_{i\in[k]} \epsilon_{\mathsf{FC}_i}(N \cdot t_{\mathsf{ARG}} + t_{\mathbf{Q}_i}) + \sum_{i\in[k]} \epsilon_{\mathbf{Q}_i}(N)$$

FIOP soundness error

FC$_i$ function-binding error

tail error for $\mathbf{Q}_i$

- **Straightline extraction in ideal model (with oracles):**

$$\epsilon_{\mathsf{ARG}}(q_{\mathsf{ARG}}) \leq \epsilon_{\mathsf{FIOP}} + \sum_{i\in[k]} \kappa_{\mathsf{FC}_i}(q_{\mathsf{FC}_i}) \qquad \sum_{i\in[k]} q_{\mathsf{FC}_i} = q_{\mathsf{ARG}}$$

FC$_i$ straightline-extraction error

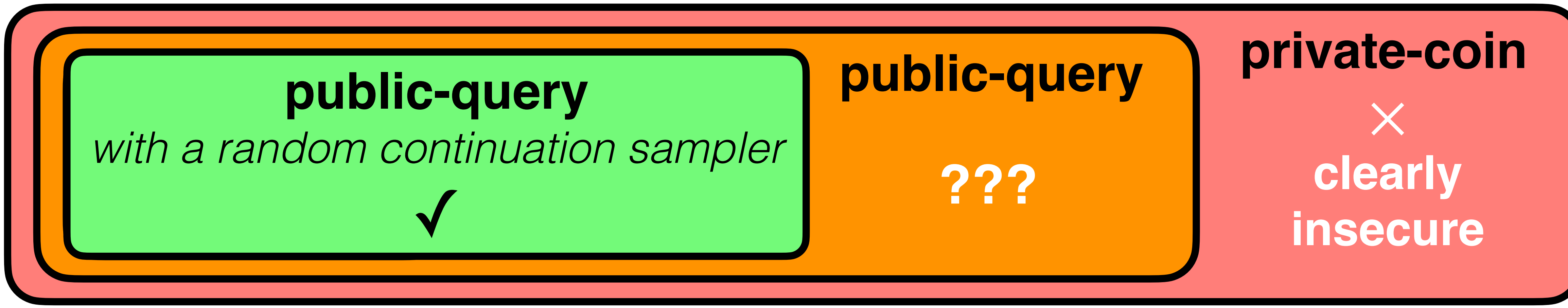
Can further extend construction+analyses to **batch and interactive FC**:

$\mathsf{FC}_{[k]}.\mathsf{Open}(\mathsf{aux}_{[k]}, Q_{[k]})$ $\rightleftarrows$ $\mathsf{FC}_{[k]}.\mathsf{Check}(\mathsf{cm}_{[k]}, \mathsf{ans}_{[k]})$

$\vdots$

Remarks

Beyond public-coin (F)IOPs?



Tightness of rewinding

Rewinding (eg for Kilian) yields $\epsilon_{\mathsf{ARG}}(t_{\mathsf{ARG}}) \leq \epsilon_{\mathsf{PCP}} + \epsilon_{\mathsf{VC}}\left(\frac{\ell \cdot t_{\mathsf{ARG}}}{\epsilon}\right) + \epsilon$.

The ϵ-tradeoff is **expensive**: for ideal VC, $\epsilon_{\mathsf{ARG}}(t_{\mathsf{ARG}}) \leq \epsilon_{\mathsf{PCP}} + \sqrt[3]{\epsilon_{\mathsf{VC}}(\ell \cdot t_{\mathsf{ARG}})}$.

Proving $\epsilon_{\mathsf{ARG}}(t_{\mathsf{ARG}}) \leq \epsilon_{\mathsf{PCP}} + \epsilon_{\mathsf{VC}}(\ell \cdot t_{\mathsf{ARG}})$ implies a breakthrough in Schnorr.

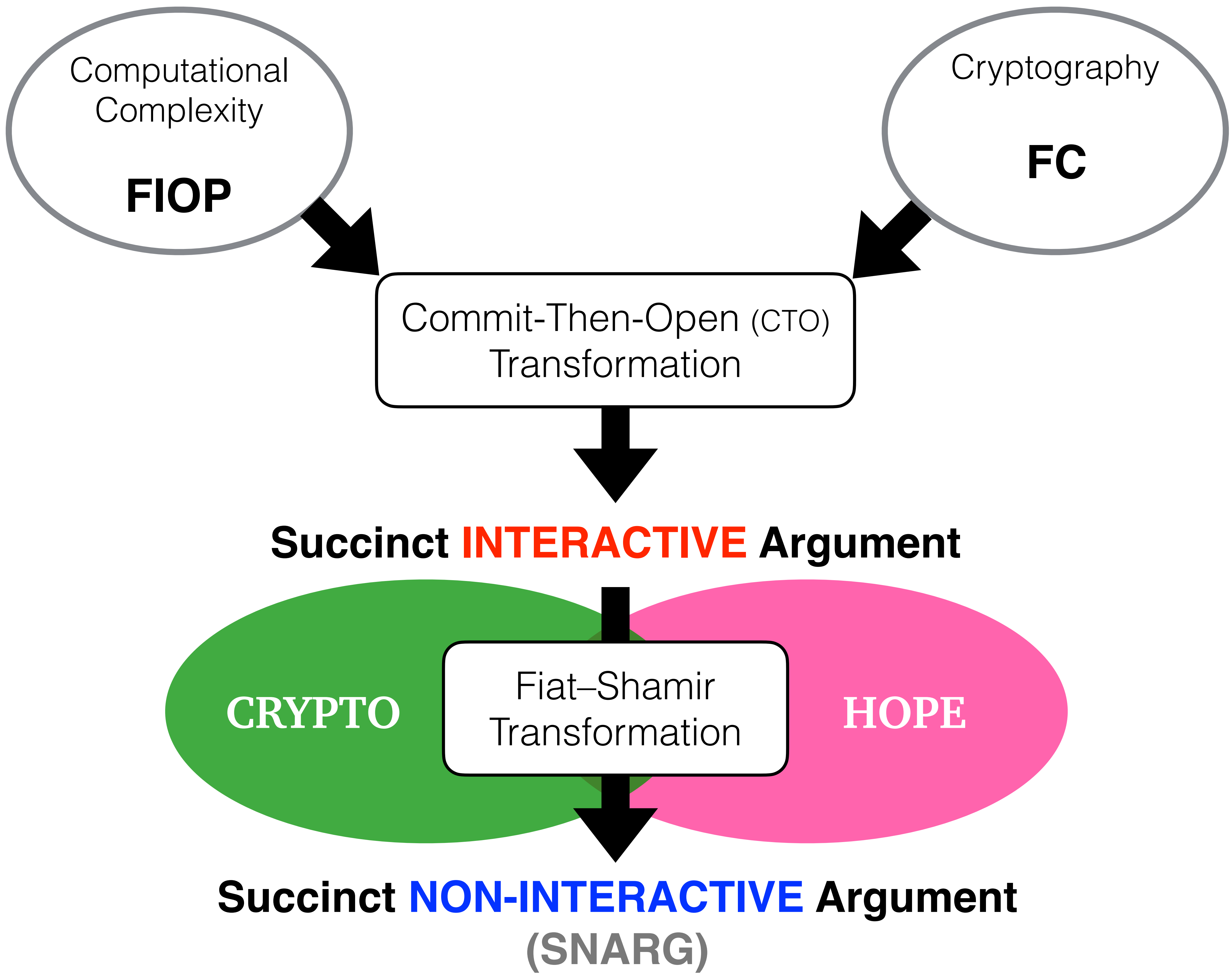
Post-quantum security

Involved. Known for VC, though should make sense for FC too.

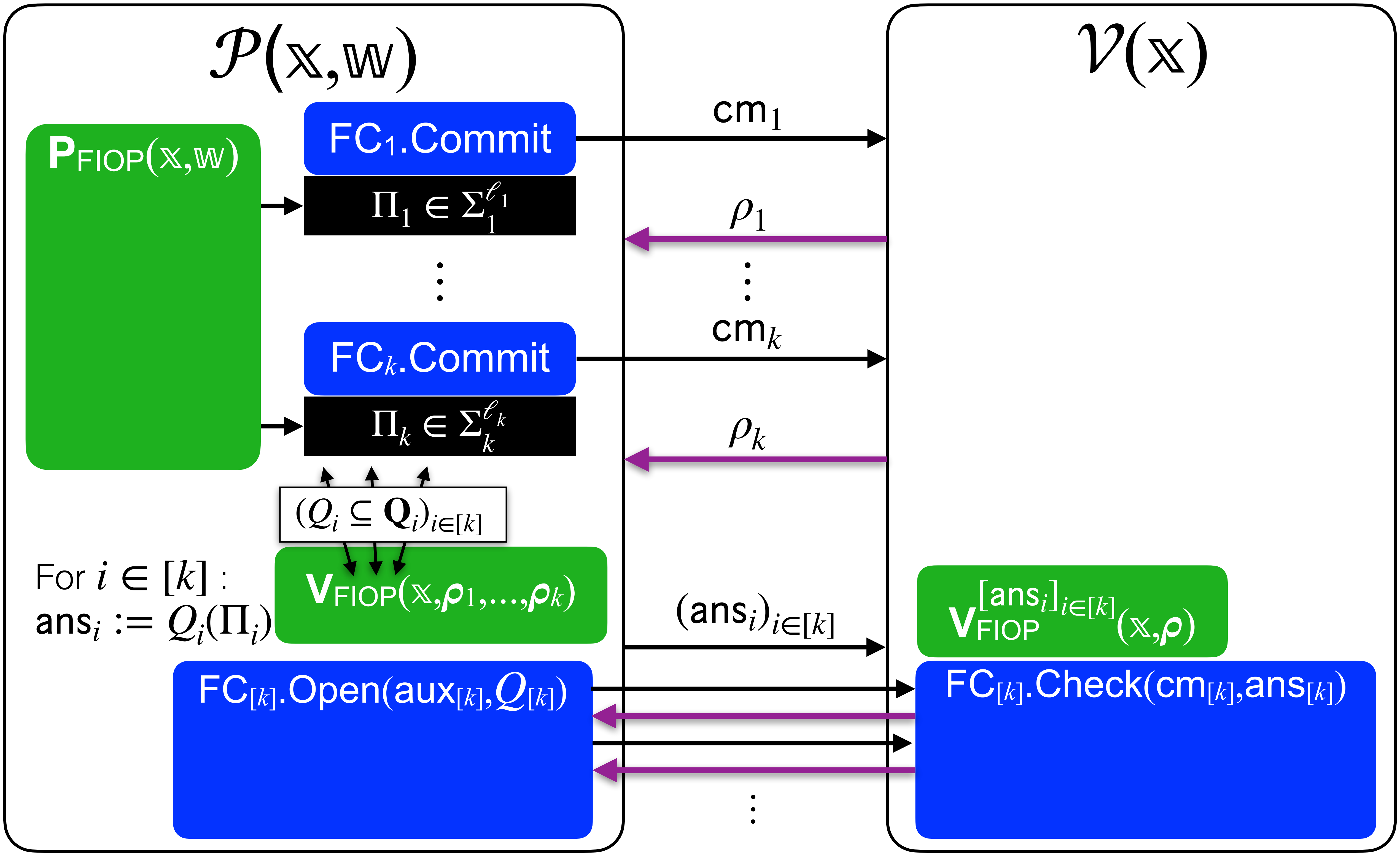
- black-box quantum rewinding (VC must be *collapse position binding*)
- straightline quantum extraction (in the QROM)

Part 2:
The Non-Interactive Case

(Almost) All Roads Lead to Fiat–Shamir



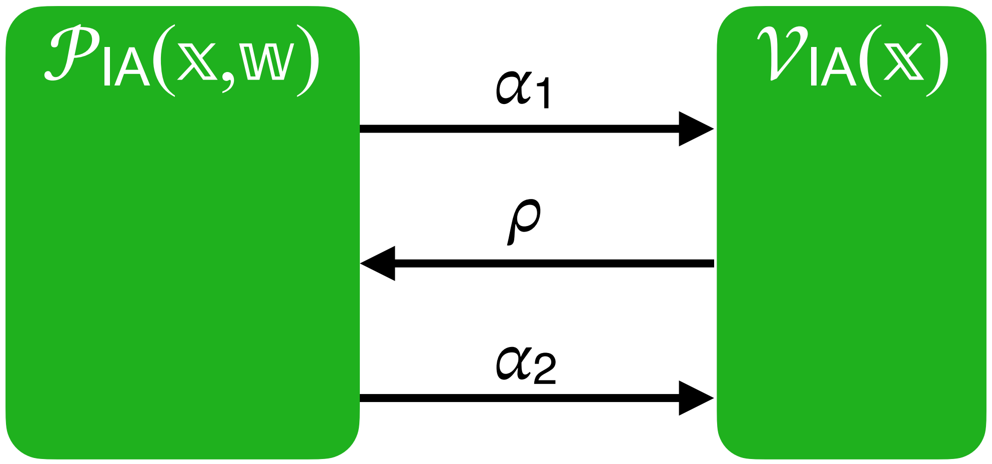
Commit-Then-Open Preserves Public Coins



Interactive argument is **public-coin** if **FIOP** and **$\mathrm{FC}_{[k]}$** are both public-coin.

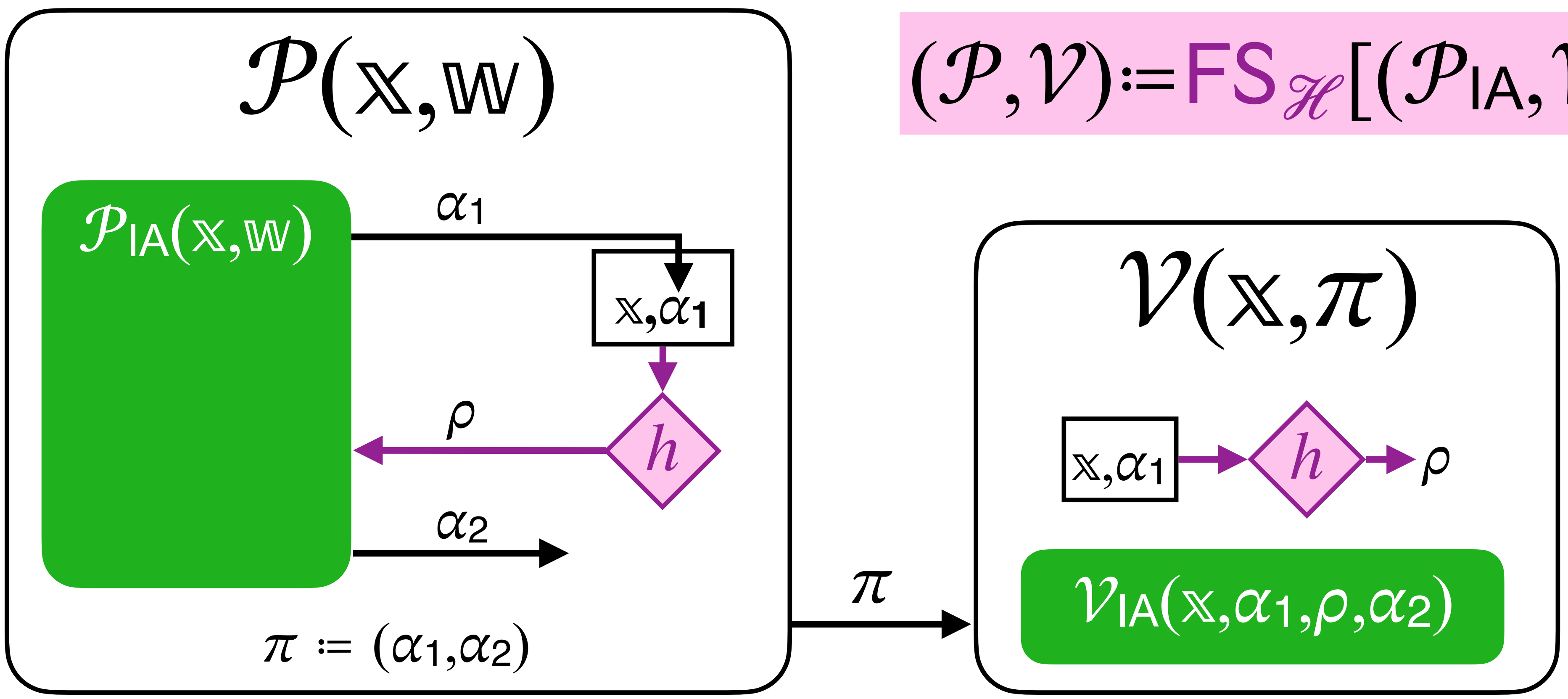
Fiat–Shamir Transformation

Consider a 3-message interactive argument:

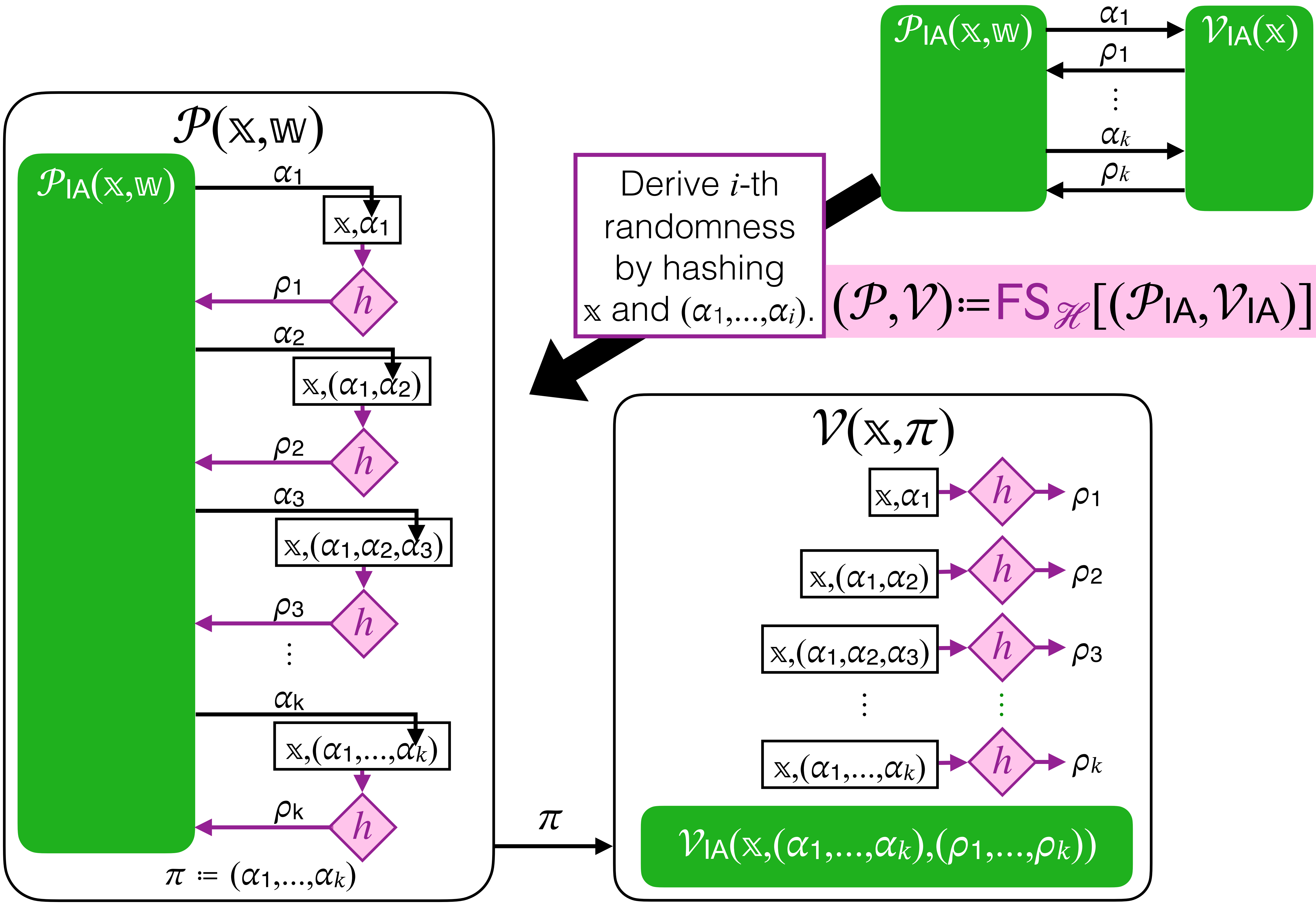


Derive the verifier's randomness via a "good" hash function $h \leftarrow \mathscr{H}$:

$(\mathcal{P},\mathcal{V}) \coloneqq \mathsf{FS}_{\mathscr{H}}[(\mathcal{P}_{\mathsf{IA}},\mathcal{V}_{\mathsf{IA}})]$



The Case of Multiple Rounds



Security of the Fiat–Shamir Transformation

WANT: standard-model security of $\mathsf{FS}_{\mathscr{H}}$ for **good sleep**
+ ideal-model security of $\mathsf{FS}_{\mathscr{H}}$ for **good parameters**

HAVE: not what we want (and research is ongoing to improve this)

- **Idealized model:** $h \leftarrow \mathscr{H}$ is a random oracle.

  This talk

  Security of $\mathsf{FS}_{\mathscr{H}}[(\mathcal{P}_{\mathsf{IA}},\mathcal{V}_{\mathsf{IA}})]$ is well understood.

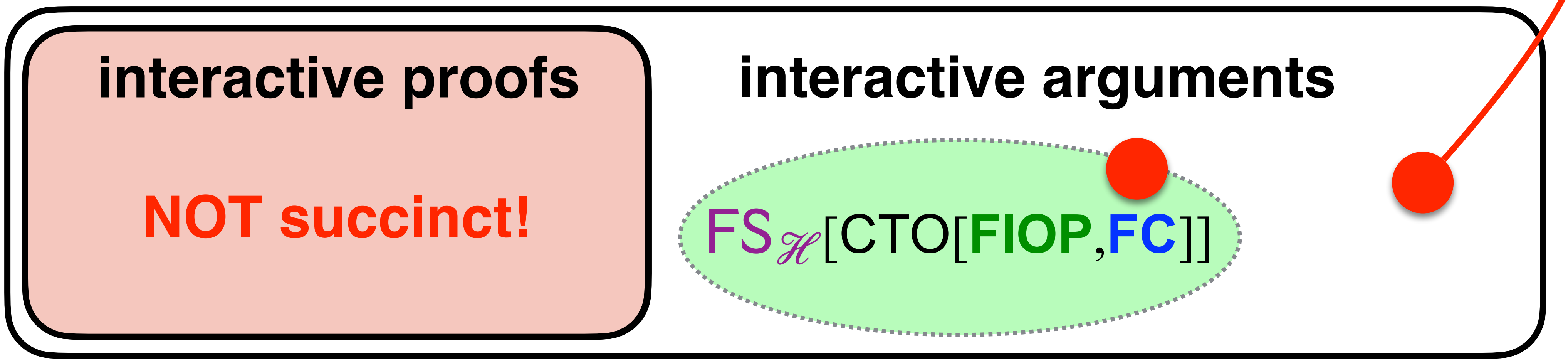
- **Standard model:** $h \leftarrow \mathscr{H}$ is an efficient "good" hash.

  Next talk

Problem: $\exists$ 3-message $(\mathcal{P}_{\mathsf{IA}},\mathcal{V}_{\mathsf{IA}})$ $\forall$ efficient $\mathscr{H}$
$\mathsf{FS}_{\mathscr{H}}[(\mathcal{P}_{\mathsf{IA}},\mathcal{V}_{\mathsf{IA}})]$ is **NOT** secure (!)

→ Security of $\mathsf{FS}_{\mathscr{H}}[(\mathcal{P}_{\mathsf{IA}},\mathcal{V}_{\mathsf{IA}})]$ must rely on **special properties** of $(\mathcal{P}_{\mathsf{IA}},\mathcal{V}_{\mathsf{IA}})$.

Alternatively, find a "better" $\mathsf{AltFS}_{\mathscr{H}}$.



Soundness Does NOT Suffice

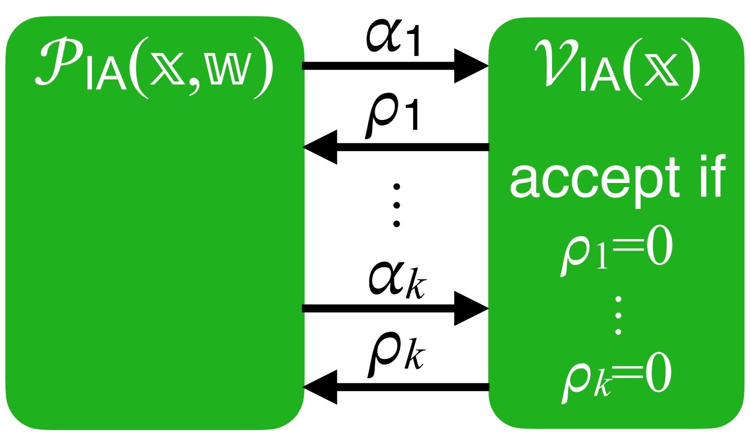
$(\mathcal{P},\mathcal{V}) := \mathsf{FS}_{\mathscr{H}}[(\mathcal{P}_{\mathsf{IA}},\mathcal{V}_{\mathsf{IA}})]$

Consider the case where $h \leftarrow \mathscr{H}$ is a random oracle.

Easy: for 3-message $(\mathcal{P}_{\mathsf{IA}},\mathcal{V}_{\mathsf{IA}})$, $\epsilon_{\mathsf{NARG}}(q_{\mathsf{RO}}) \leq (q_{\mathsf{RO}}+1)\cdot\epsilon_{\mathsf{IA}}$.

Generally: for k-round $(\mathcal{P}_{\mathsf{IA}},\mathcal{V}_{\mathsf{IA}})$, $\epsilon_{\mathsf{NARG}}(q_{\mathsf{RO}}) \leq \binom{q_{\mathsf{RO}}+k}{k}\cdot\epsilon_{\mathsf{IA}}$.

This (huge) soundness loss **CAN** happen:

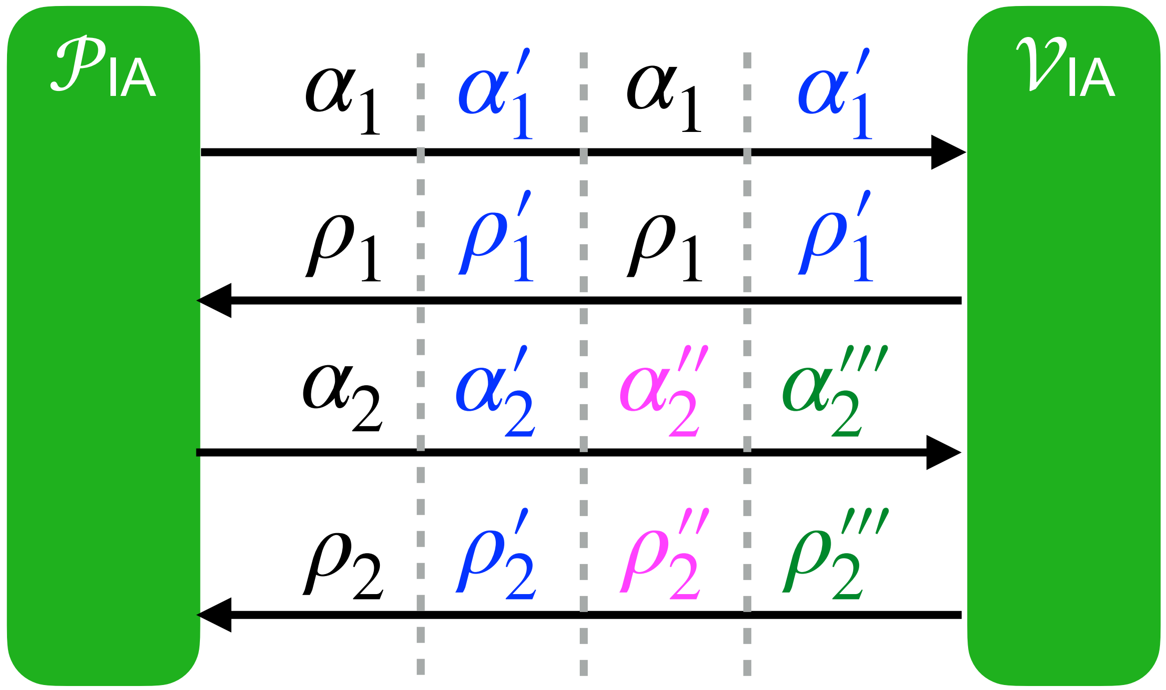


- $(\mathcal{P}_{\mathsf{IA}},\mathcal{V}_{\mathsf{IA}})$ has soundness error 2^{-k} (unconditional)
- $(\mathcal{P},\mathcal{V})$ has soundness error $\epsilon_{\mathsf{NARG}}(q_{\mathsf{RO}}) = \Omega\left(\left(\frac{q_{\mathsf{RO}}}{k}\right)^k\right)$

good soundness of $(\mathcal{P}_{\mathsf{IA}},\mathcal{V}_{\mathsf{IA}})$ $\not\rightarrow$ good soundness of $(\mathcal{P},\mathcal{V})$

State-Restoration Attacks

$\mathsf{FS}_{\mathscr{H}}[(\mathcal{P}_{\mathsf{IA}},\mathcal{V}_{\mathsf{IA}})]$ allows attacking $\mathcal{V}_{\mathsf{IA}}$ across **multiple interactions**, by trying different prover messages to obtain different transcripts.



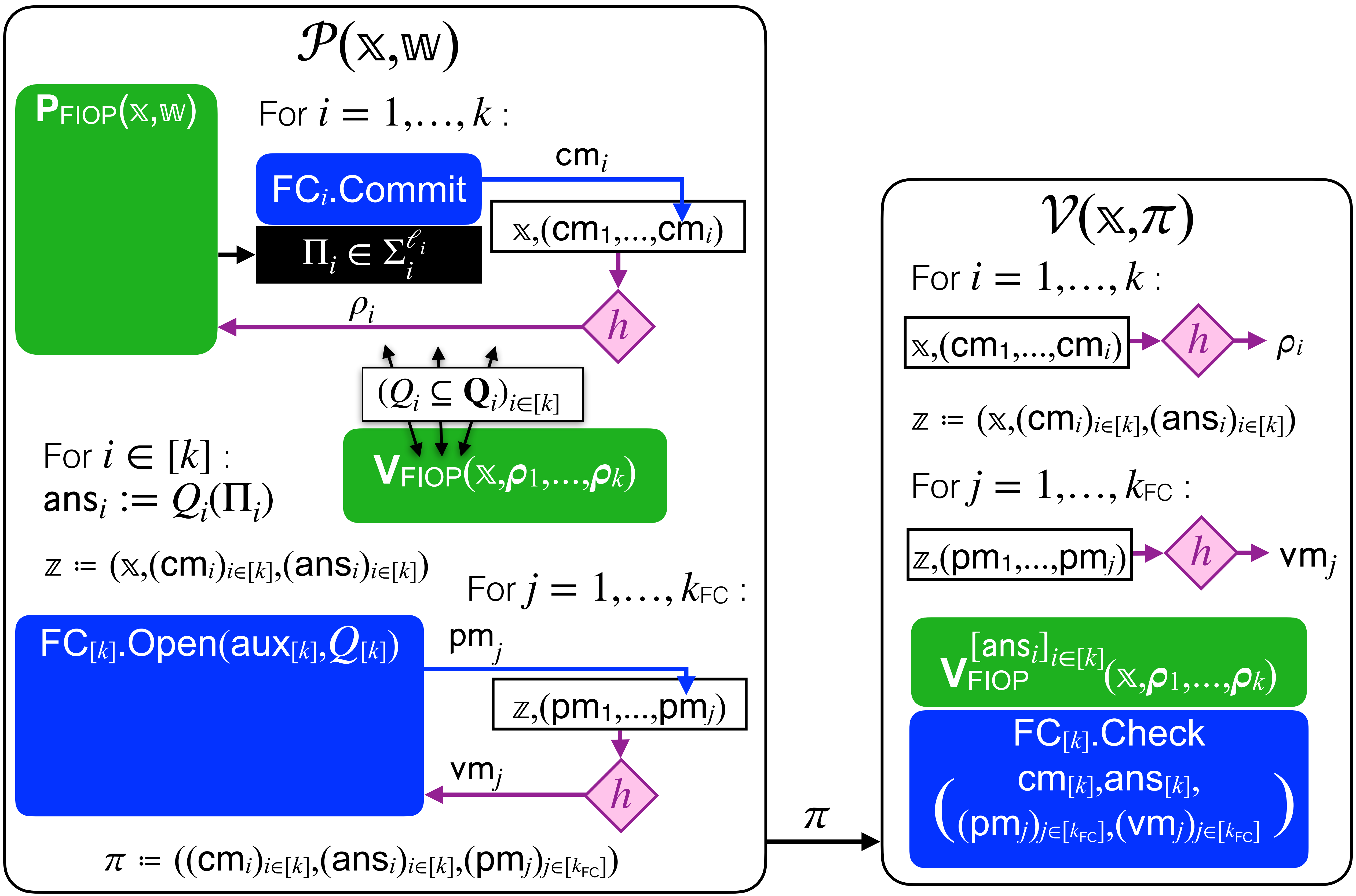
The attacker wins by finding **any** accepting transcript.

state-restoration attack

Define a **state-restoration game** that models this.

Lemma: $(\mathcal{P}_{\mathsf{IA}},\mathcal{V}_{\mathsf{IA}})$ has **state-restoration soundness error** $\epsilon_{\mathsf{IA}}^{\mathsf{SR}}(q_{\mathsf{SR}}, t_{\mathsf{IA}})$
→ $\mathsf{FS}_{\mathscr{H}}[(\mathcal{P}_{\mathsf{IA}},\mathcal{V}_{\mathsf{IA}})]$ has soundness error $\epsilon_{\mathsf{NARG}}(q_{\mathsf{RO}}, t_{\mathsf{NARG}}) \leq \epsilon_{\mathsf{IA}}^{\mathsf{SR}}(q_{\mathsf{RO}}, t_{\mathsf{NARG}})$.

(Functional Extension of) The BCS Protocol



SR Soundness of Commit-Then-Open

We can further extend security reductions to handle state-restoration:

FIOP and **FC** satisfy state-restoration soundness
→ $(\mathcal{P}_{\mathsf{IA}}, \mathcal{V}_{\mathsf{IA}}) := \mathsf{CommitThenOpen}[\textbf{FIOP}, \textbf{FC}]$
satisfies state-restoration soundness 🥳

- **Black-box prover rewinding (no oracles):**

$$\forall N \quad \epsilon_{\mathsf{IA}}^{\mathsf{SR}}(q_{\mathsf{SR}}, t_{\mathsf{IA}}) \leq \epsilon_{\mathsf{FIOP}}^{\mathsf{SR}}(q_{\mathsf{SR}}) + \sum_{i\in[k]} \epsilon_{\mathsf{FC}_i}^{\mathsf{SR}}(N q_{\mathsf{SR}}, N t_{\mathsf{IA}} + t_{\mathbf{Q}_i}) + \sum_{i\in[k]} \epsilon_{\mathbf{Q}_i}(N)$$

FIOP **SR** soundness error

FC_i **SR** function-binding error

solver time for $\mathbf{Q}_i$

tail error for $\mathbf{Q}_i$

- **Straightline extraction in ideal model (with oracles):**

$$\epsilon_{\mathsf{IA}}^{\mathsf{SR}}(q_{\mathsf{SR}}, q_{\mathsf{IA}}) \leq \epsilon_{\mathsf{FIOP}}^{\mathsf{SR}}(q_{\mathsf{SR}}) + \sum_{i\in[k]} \kappa_{\mathsf{FC}_i}^{\mathsf{SR}}(q_{\mathsf{SR}}, q_{\mathsf{FC}_i}) \qquad \sum_{i\in[k]} q_{\mathsf{FC}_i} = q_{\mathsf{IA}}$$

FC_i **SR** straightline-extraction error

Achieving SR Soundness

Proving that a protocol satisfies (good) SR soundness can be **laborious**.

Easier: prove the protocol satisfies a stronger soundness notion!

(many protocols of interest do)

(1) RBR (round-by-round) soundness

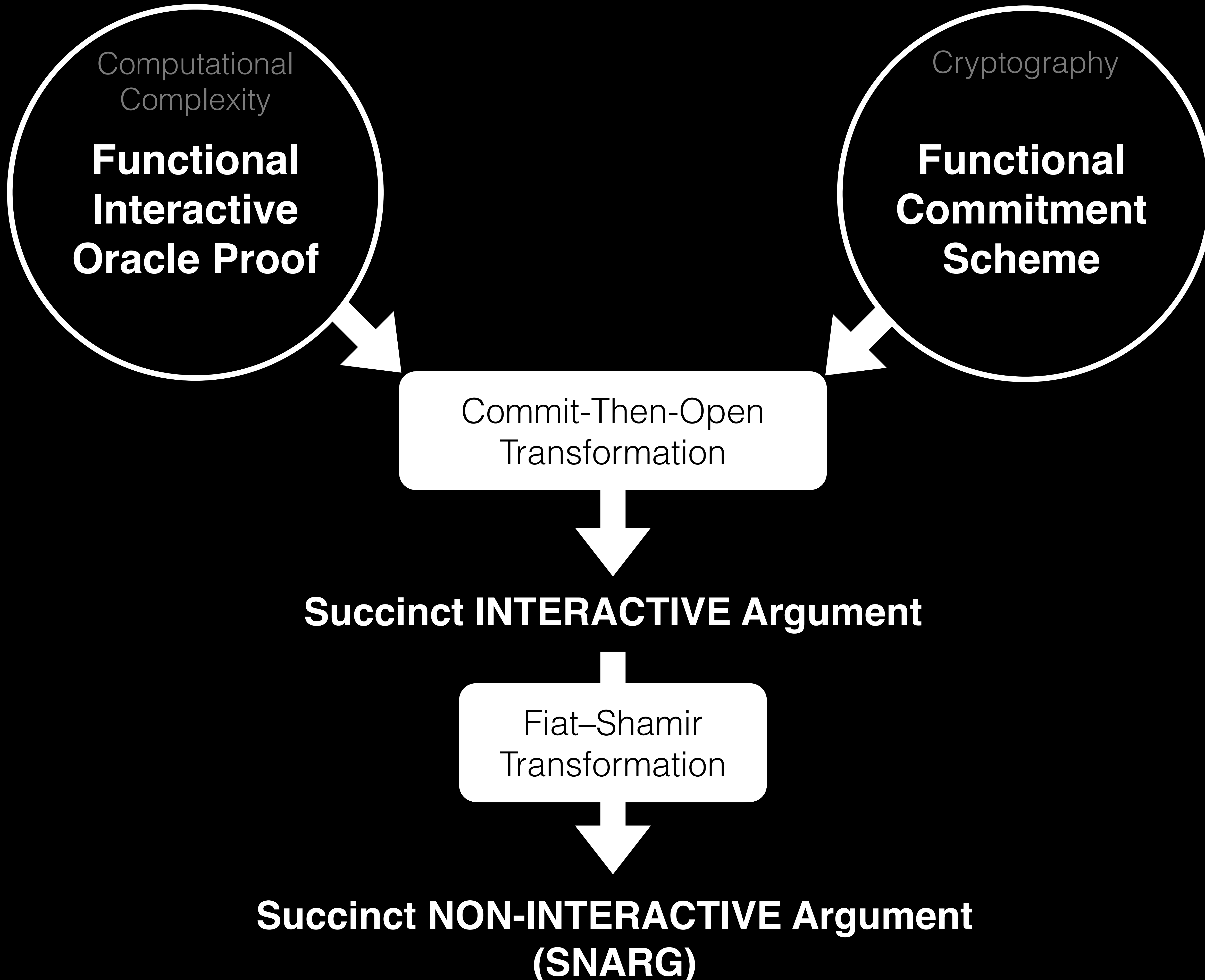
k-round **FIOP** has RBR soundness error $\epsilon^{\text{RBR}}_{\text{FIOP}}$
→ **FIOP** has **SR** soundness error $\epsilon^{\text{SR}}_{\text{FIOP}}(q_{\text{SR}}) \leq (q_{\text{SR}} + k) \cdot \epsilon^{\text{RBR}}_{\text{FIOP}}$

(2) special soundness

k-round **FIOP** has $((a_i, N_i))_{i \in [k]}$-special soundness
→ **FIOP** has **SR** soundness error $\epsilon^{\text{SR}}_{\text{FIOP}}(q_{\text{SR}}) \leq (q_{\text{SR}} + 1) \cdot \sum_{i \in [k]} \frac{a_i - 1}{N_i}$

(Similar implications hold for knowledge soundness.)

Conclusion


Computational Complexity
Functional Interactive Oracle Proof
Cryptography
Functional Commitment Scheme
Commit-Then-Open Transformation
Succinct INTERACTIVE Argument
Fiat–Shamir Transformation
Succinct NON-INTERACTIVE Argument (SNARG)

Building Cryptographic Proofs from Hash Functions

Alessandro Chiesa & **Eylon Yogev**

Comprehensive and rigorous treatment of SNARGs in the ROM.

Part I
What are Cryptographic Proofs?

Part II
NARGs Based on SPs
- FS protocol for SPs

Part III
NARGs Based on IPs
- FS protocol for IPs
- "fast" FS protocol for IPs

Part IV
Commitment Schemes
- basic commitment
- Merkle commitment

Part V
SNARGs Based on PCPs
- Kilian protocol
- Micali protocol

Part VI
SNARGs Based on IOPs
- iBCS protocol
- BCS protocol

Part VII
Practical Considerations
- setting parameters
- Merkle optimizations
- RBR soundness
- special soundness

snargsbook.org

Thanks!