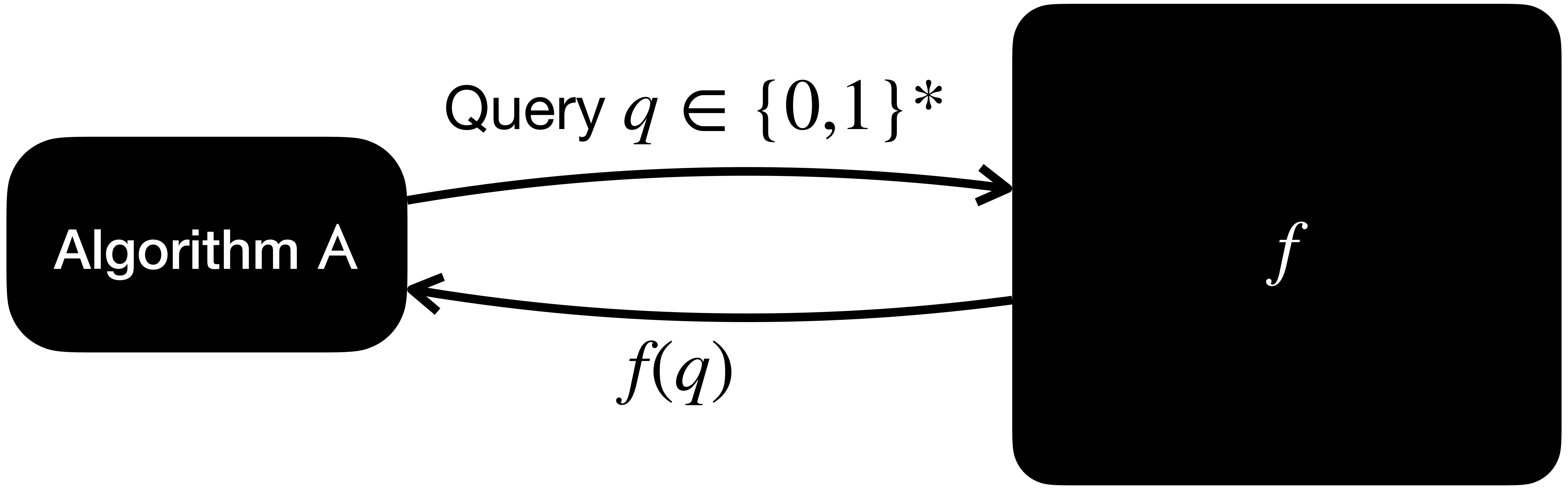
Breaking Verifiable Delay Functions in the Random Oracle Model

Ziyi Guan, Artur Riazanov, Weiqiang Yuan

Random oracle model

Random oracle $\mathcal{O} := \{\mathcal{O}_\ell\}_{\ell \in \mathbb{N}}$

For every $\ell \in \mathbb{N}$, $\mathcal{O}_\ell$ is the uniform distribution over all functions $f\colon \{0,1\}^* \to \{0,1\}^\ell$.



Verifiable Delay Function (VDF)

T queries — t queries — t ≪ T

$$\mathsf{Eval}^f(x) \xrightarrow{(y,\pi)} \mathsf{Verify}^f(x,y,\pi)$$

Completeness. For every security parameter λ and input x,

$$\Pr\left[\mathsf{Verify}^f(x,y,\pi)=1 \;\middle|\; \begin{array}{r} f \leftarrow \mathcal{O}(\lambda) \\ (y,\pi) \leftarrow \mathsf{Eval}^f(x) \end{array}\right] = 1.$$

Sequentiality. For every security parameter λ, input x, and poly(t)-round poly(T)-query adversary Adv,

$$\Pr\left[y = \mathsf{Eval}^f(x) \;\middle|\; \begin{array}{r} f \leftarrow \mathcal{O}(\lambda) \\ (y,\pi) \leftarrow \mathsf{Adv}^f(x) \end{array}\right] \leq \mathsf{negl}(\lambda).$$

Computational Uniqueness. For every security parameter λ, input x, and poly(T)-query adversary Adv,

$$\Pr\left[\begin{array}{l} y \neq \mathsf{Eval}^f(x) \\ \wedge\, \mathsf{Verify}^f(x,y,\pi)=1 \end{array} \;\middle|\; \begin{array}{r} f \leftarrow \mathcal{O}(\lambda) \\ (y,\pi) \leftarrow \mathsf{Adv}^f(x) \end{array}\right] \leq \mathsf{negl}(\lambda).$$

Why study VDF?

Randomness beacon

An ideal service that regularly publish randomness
that no one can predict/manipulate

Previous approach:
- Apply a randomness extractor to stock prices;
- Issue: stock prices can be manipulated to bias the output randomness.

Using VDF: because of the delay (sequentiality), adversaries cannot quickly compute output randomness to decide how to manipulate the sources (stock prices).

Blockchain: leader election

Select the participant that determines the next block

- Unpredictability (sequentiality): adversaries do not know the next leader until shortly before the announcement.
- Uniqueness: exactly one leader is chosen each time.

Our result

Verifiable delay functions do not exist in the random oracle model!
No black-box constructions of VDFs from OWF, OWP, CRHF, etc.
Cryptography is necessary for VDF constructions!

Main Theorem. Consider $\mathsf{VDF} = (\mathsf{Eval}, \mathsf{Verify})$ in the random oracle model (ROM).
There exists a $O(\mathrm{t})$-round $O(\mathrm{t} \cdot \mathrm{T})$-query adversary Adv that breaks the sequentiality of VDF.

No adversary can find alternative solutions.

No query-bounded adversary can find alternative solutions with non-negl. probability.

	Perfect Completeness	**Imperfect Completeness**
Perfect Uniqueness	[MSW20]: ❌ ROM **Main Theorem:** ❌ ROM	**Main Theorem:** ❌ ROM
Computational Uniqueness	[DGMV20]: ❌ tight VDFs in ROM [RSS20]: ❌ cyclic groups of known orders [EFKP20]: ✅ ROM + repeated squaring **Main Theorem:** ❌ ROM	**Main Theorem:** ❌ ROM

Warm-up: perfect uniqueness

Recap: perfectly unique VDFs in the ROM

T queries | t queries | q $\ll$ Q

$\mathsf{Eval}^f(x) \xrightarrow{(y,\pi)} \mathsf{Verify}^f(x,y,\pi)$

Completeness. For every security parameter λ and input x,

$$\Pr\left[\mathsf{Verify}^f(x,y,\pi)=1 \;\middle|\; \begin{array}{r} f \leftarrow \mathcal{O}(\lambda) \\ (y,\pi) \leftarrow \mathsf{Eval}^f(x) \end{array}\right] = 1.$$

Sequentiality. For every security parameter λ, input x, and poly(t)-round poly(T)-query adversary Adv,

$$\Pr\left[y = \mathsf{Eval}^f(x) \;\middle|\; \begin{array}{r} f \leftarrow \mathcal{O}(\lambda) \\ (y,\pi) \leftarrow \mathsf{Adv}^f(x) \end{array}\right] \leq \mathsf{negl}(\lambda).$$

Perfect Uniqueness. For every security parameter λ, input x, and unbounded adversary Adv,

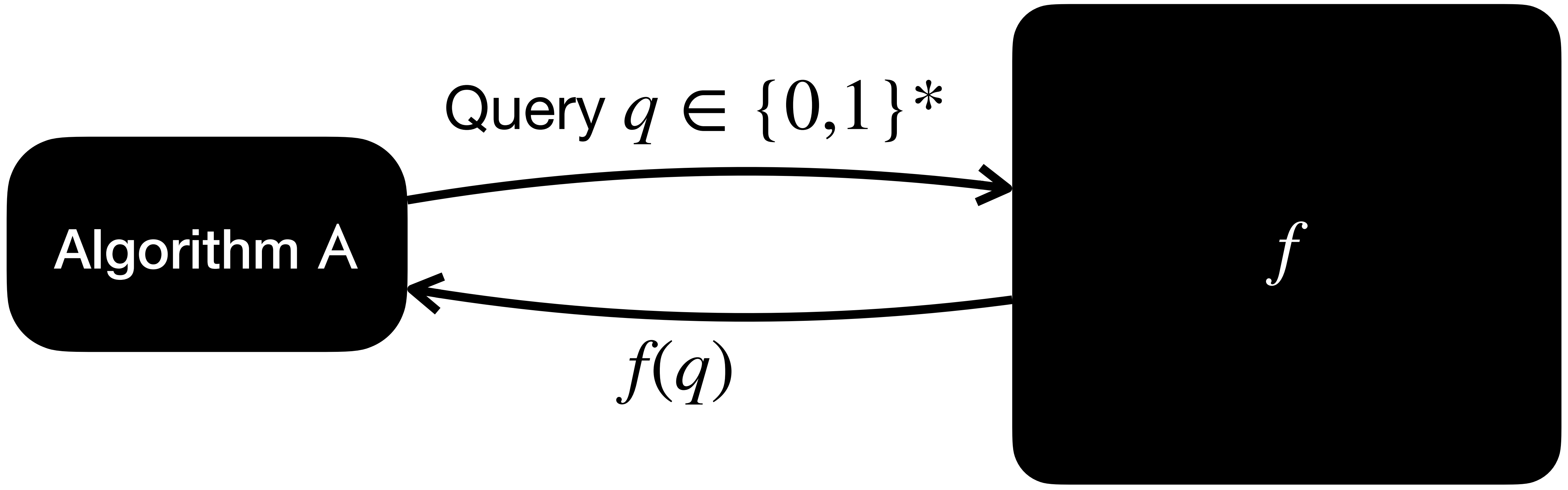
$$\Pr\left[\begin{array}{l} y \neq \mathsf{Eval}^f(x) \\ \wedge\, \mathsf{Verify}^f(x,y,\pi)=1 \end{array} \;\middle|\; \begin{array}{r} f \leftarrow \mathcal{O}(\lambda) \\ (y,\pi) \leftarrow \mathsf{Adv}^f(x) \end{array}\right] = 0.$$

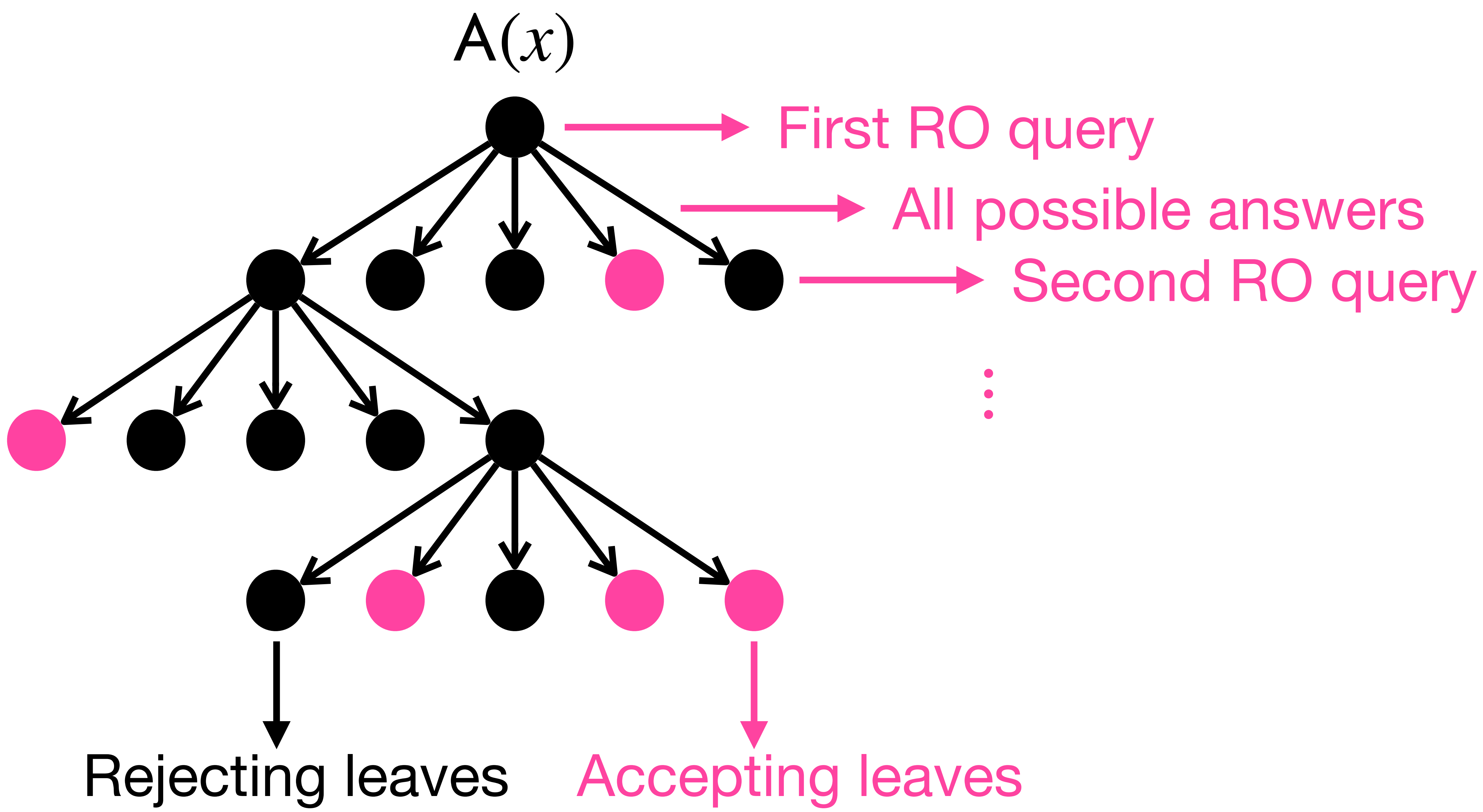
i.e. For every f and x, $\mathsf{Verify}^f(x)$ accepts one and only one output y.

Brief detour: decision tree algorithms

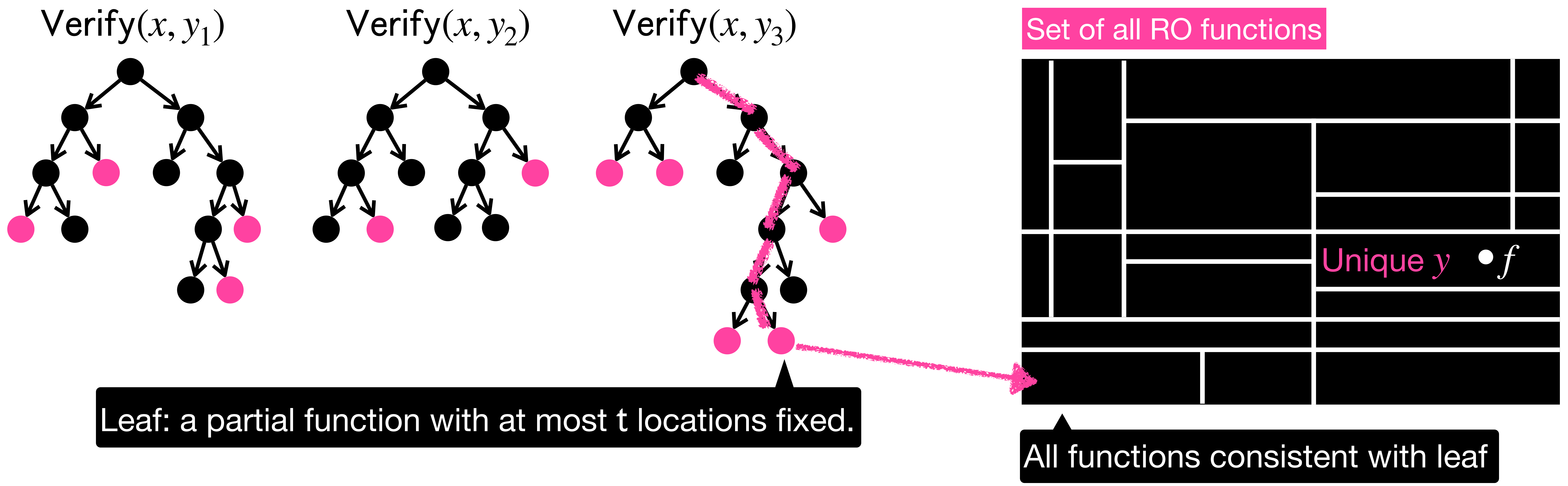
Random oracle $\mathcal{O} := \{\mathcal{O}_\ell\}_{\ell \in \mathbb{N}}$

For every $\ell \in \mathbb{N}$, $\mathcal{O}_\ell$ is the uniform distribution over all functions $f\colon \{0,1\}^* \to \{0,1\}^\ell$.





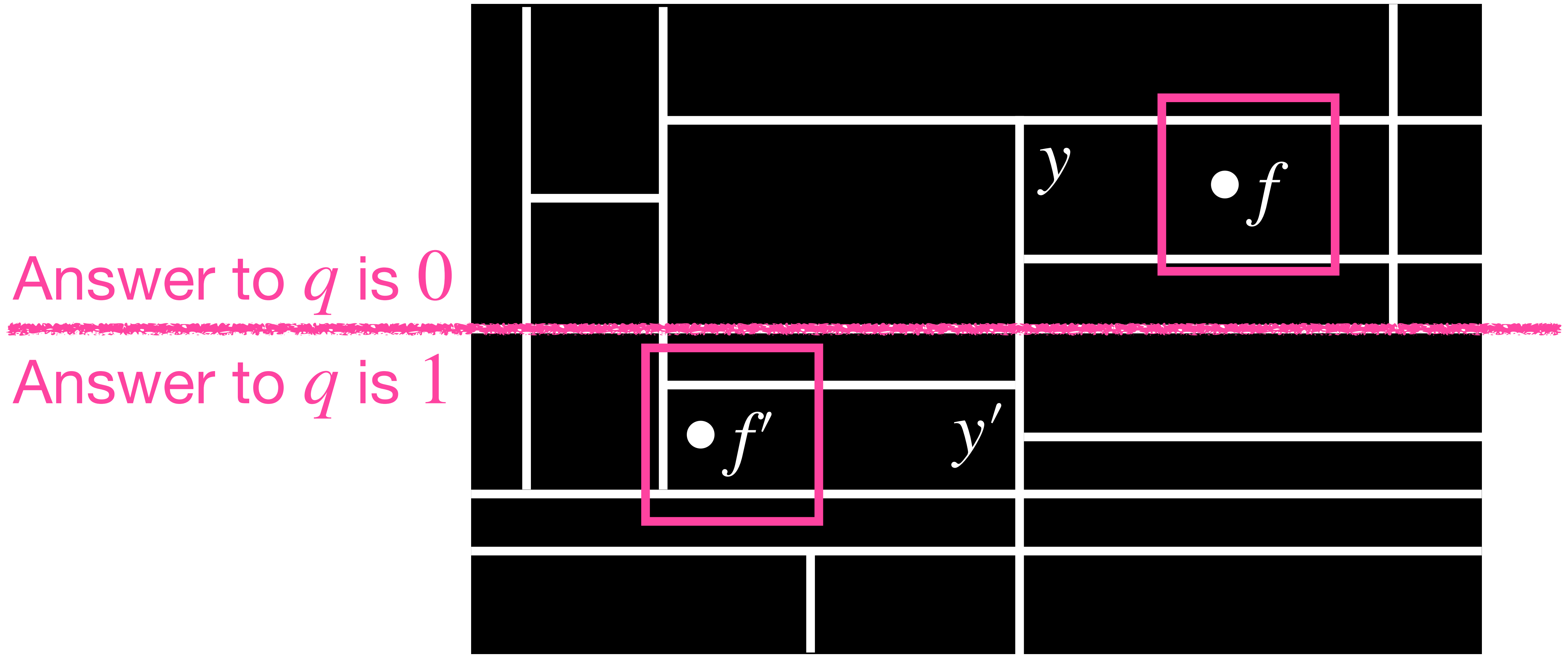
Verify$(x, \cdot\,)$ partitions random oracles



Every RO function f in this rectangle satisfies Verify$^f(x, y) = 1$.

Adversary that breaks sequentiality

$\{0,1\}^*$: set of all RO functions partitioned using Verify



$2t+1$ rounds of queries
At most T queries each round

$\mathsf{Adv}^f(x)$:

1. For $i \in [2t+1]$:
   a. Pick $f' \in \{0,1\}^*$ consistent with current view of f.
   b. Compute $y' := \mathsf{Eval}^{f'}(x)$.
   c. Let $Q_{\mathsf{Eval}}(f', x)$ be the query set of $\mathsf{Eval}^{f'}(x)$.
   d. Query f with $Q_{\mathsf{Eval}}(f', x)$ in one round.
2. Output $\mathsf{Majority}(y_1, \ldots, y_{2t+1})$.

View of f at the beginning of iter. i

- $y \neq y' \implies$ Adv queries at least one new position $q \in Q_{\mathsf{Verify}}(f, x, y) \backslash Q_i$.
  - Otherwise, $\mathsf{Verify}^f(x, y') = \mathsf{Verify}^{f'}(x, y') = 1$, contradicting perfect uniqueness.
- $|Q_{\mathsf{Verify}}(f, x, y)| \leq t \implies$ At most t iterations have $y' \neq y$.

Computational uniqueness

Recap: computationally unique VDFs in the ROM

T queries t queries $t \ll T$

$$\mathsf{Eval}^f(x) \xrightarrow{(y, \pi)} \mathsf{Verify}^f(x, y, \pi)$$

Completeness. For every security parameter λ and input x,

$$\Pr\left[\mathsf{Verify}^f(x,y,\pi) = 1 \;\middle|\; \begin{array}{r} f \leftarrow \mathcal{O}(\lambda) \\ (y,\pi) \leftarrow \mathsf{Eval}^f(x) \end{array}\right] = 1.$$

Sequentiality. For every security parameter λ, input x, and poly(t)-round poly(T)-query adversary Adv,

$$\Pr\left[y = \mathsf{Eval}^f(x) \;\middle|\; \begin{array}{r} f \leftarrow \mathcal{O}(\lambda) \\ (y,\pi) \leftarrow \mathsf{Adv}^f(x) \end{array}\right] \leq \mathsf{negl}(\lambda).$$

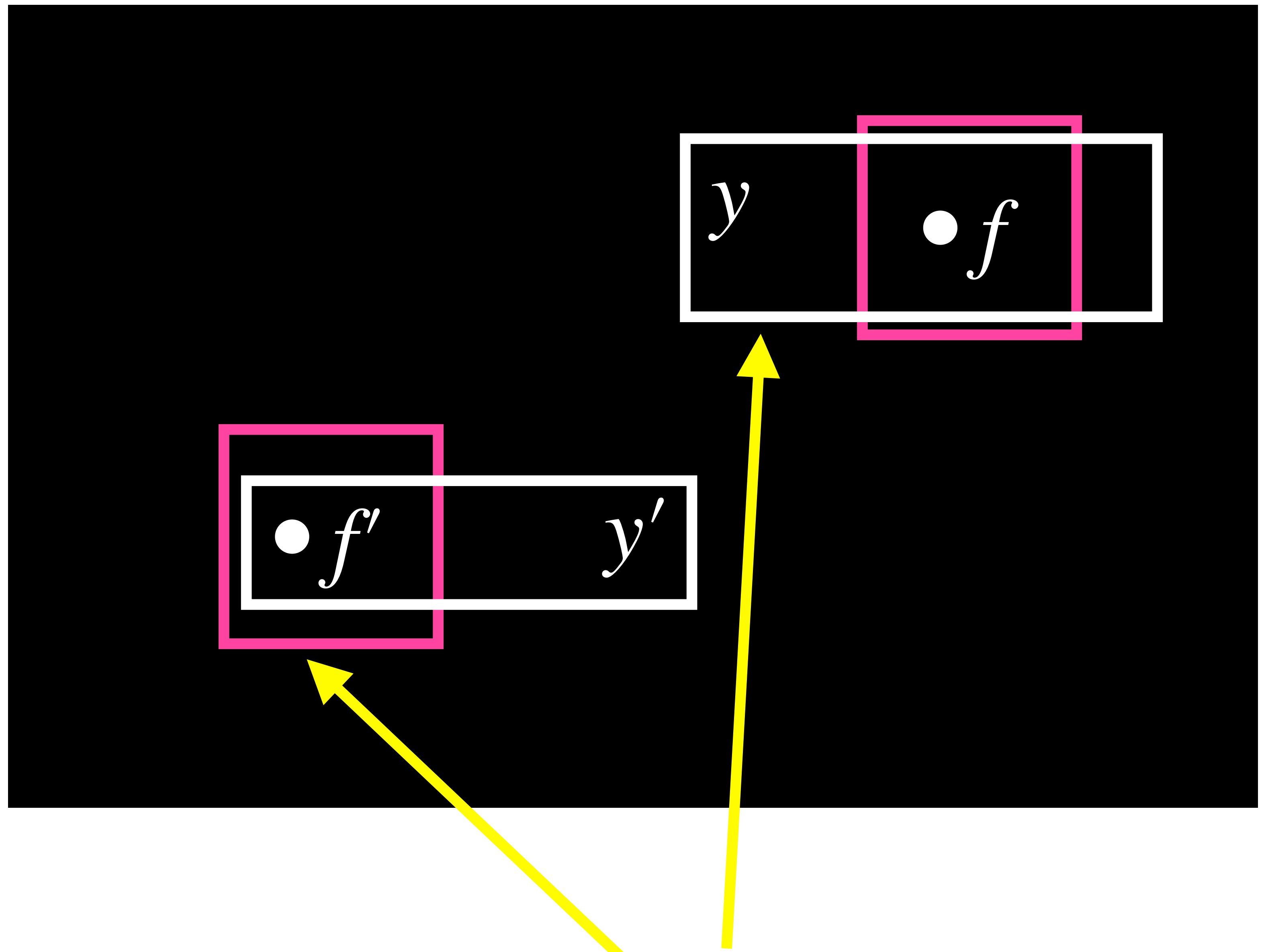
Computational Uniqueness. For every security parameter λ, input x, and poly(T)-query adversary Adv,

$$\Pr\left[\begin{array}{l} y \neq \mathsf{Eval}^f(x) \\ \wedge \mathsf{Verify}^f(x,y,\pi) = 1 \end{array} \;\middle|\; \begin{array}{r} f \leftarrow \mathcal{O}(\lambda) \\ (y,\pi) \leftarrow \mathsf{Adv}^f(x) \end{array}\right] \leq \mathsf{negl}(\lambda).$$

i.e. For every x and poly(T)-query Adv, there are at most $\mathsf{negl}(\lambda)$-fraction of f where Adv can find $y' \neq \mathsf{Eval}^f(x)$ and $\mathsf{Verify}^f(x, y') = 1$.

How does previous adversary fail?

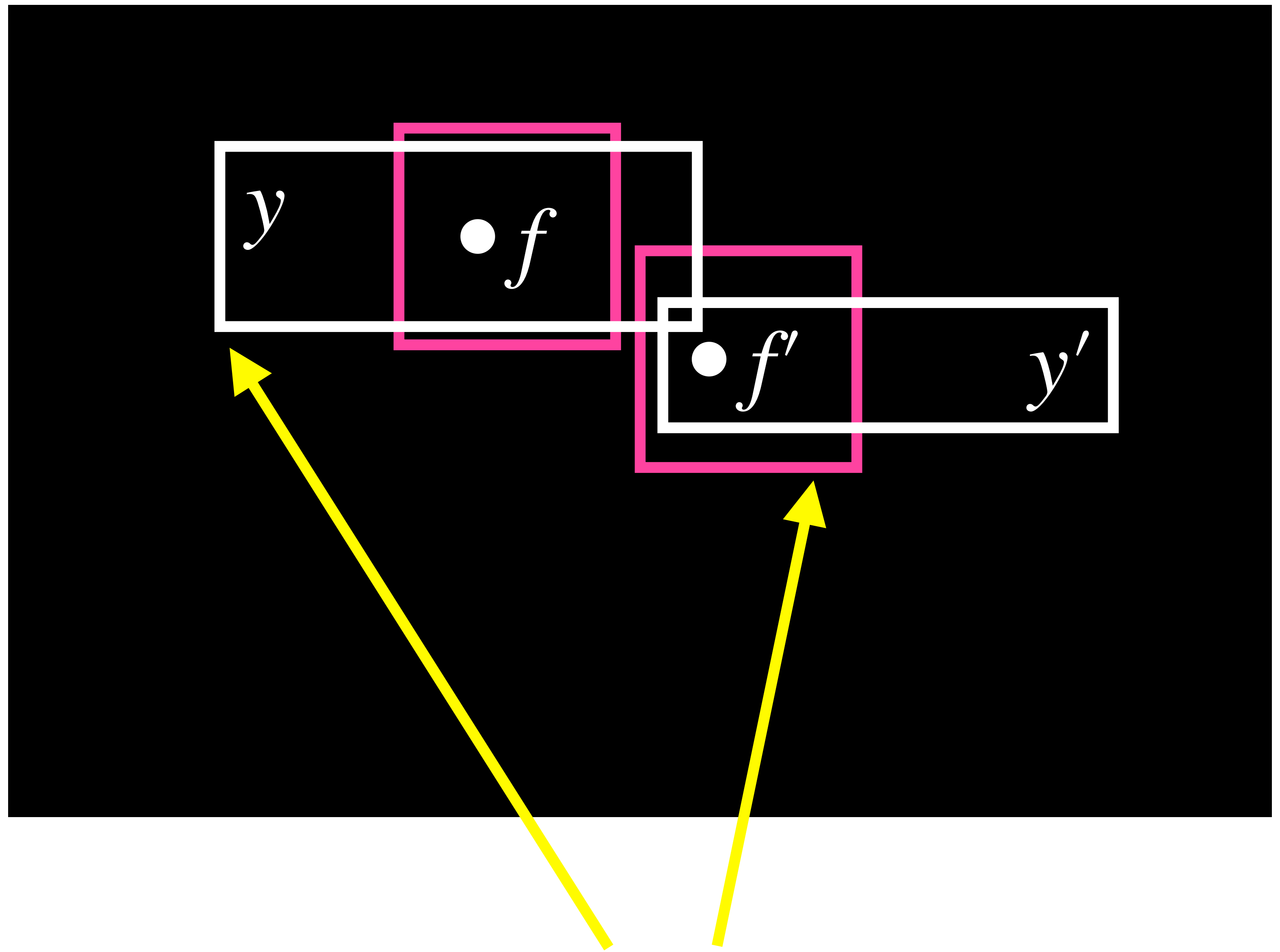
Perfect uniqueness



Disjoint: can learn new location

$y \neq y'$
$\Longrightarrow q \in \mathsf{Q}_{\mathsf{Eval}}(f', x) \cap \mathsf{Q}_{\mathsf{Verify}}(f, x, y) \backslash \mathsf{Q}_i$.
Otherwise, $\mathsf{Verify}^f(x, y') = \mathsf{Verify}^{f'}(x, y') = 1$, contradicting perfect uniqueness.

Computationaly uniqueness



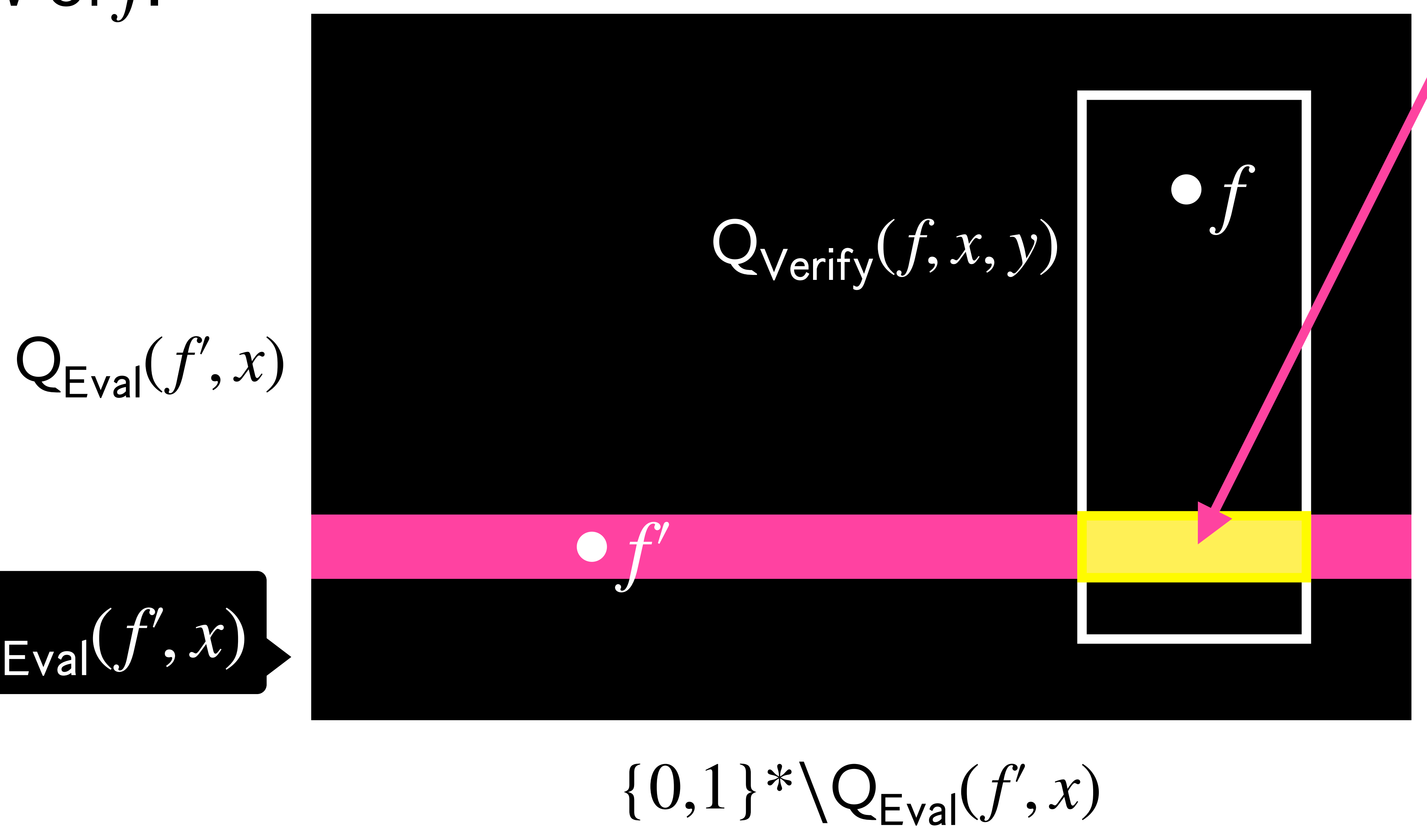
Might intersect: cannot learn new location

$y \neq y'$
$\not\Longrightarrow q \in \mathsf{Q}_{\mathsf{Eval}}(f', x) \cap \mathsf{Q}_{\mathsf{Verify}}(f, x, y) \backslash \mathsf{Q}_i$.
Since $\mathsf{Verify}^f(x, y') = \mathsf{Verify}^{f'}(x, y') = 1$ doesn't contradict computational uniqueness.

Coupling with a uniqueness breaker [1/2]

$\mathsf{Adv}^f(x)$:

1. For $i \in [2t+1]$:
    a. Pick $f' \in \{0,1\}^*$ consistent with current view of f.
    b. Compute $y' := \mathsf{Eval}^{f'}(x)$.
    c. Let $\mathsf{Q}_{\mathsf{Eval}}(f', x)$ be the query set of $\mathsf{Eval}^{f'}(x)$.
    d. Query f with $\mathsf{Q}_{\mathsf{Eval}}(f', x)$ in one round.
2. Output $\mathsf{Majority}(y_1, \ldots, y_{2t+1})$.

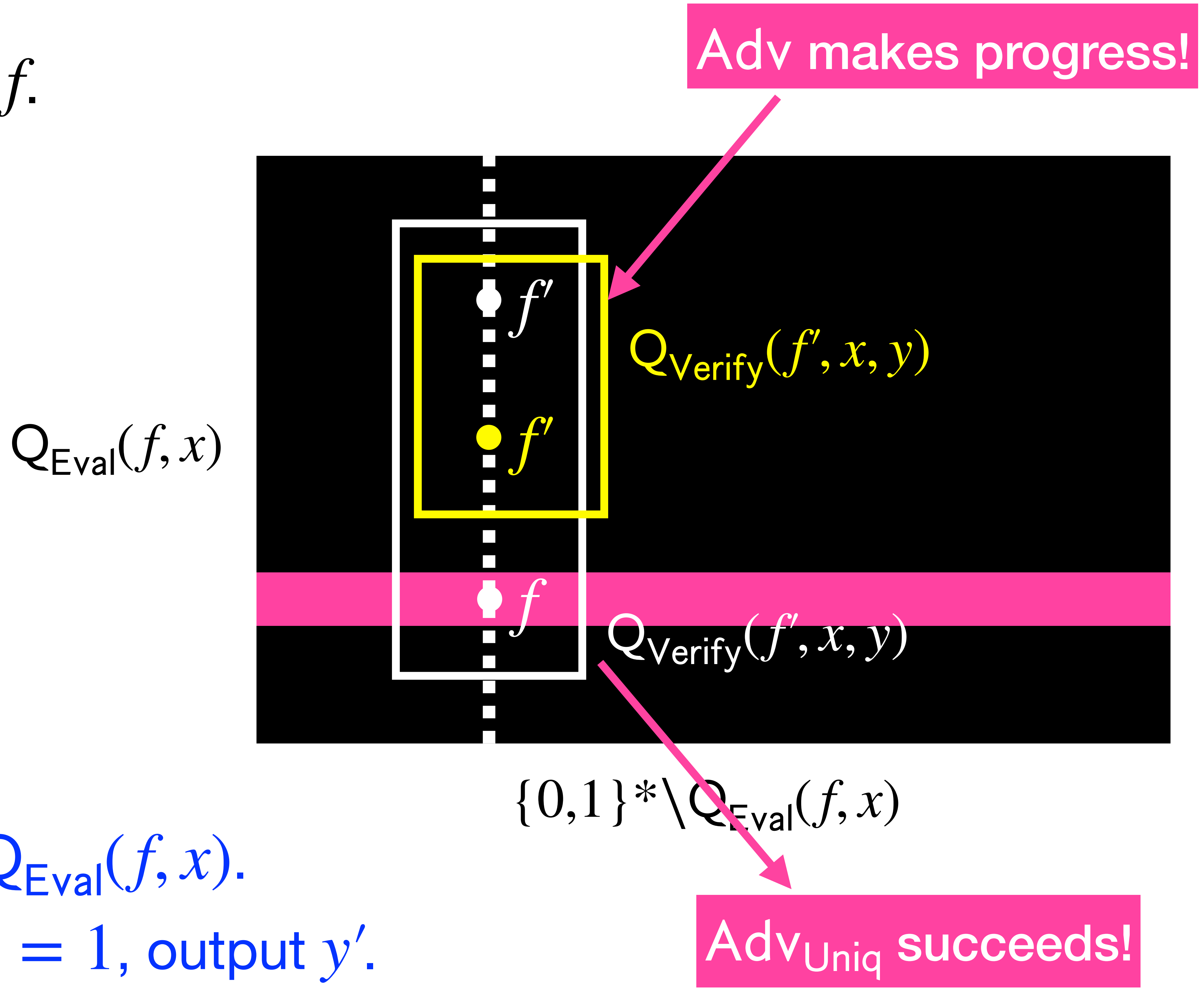


$\{0,1\}^*$: set of all RO functions shuffled by positions in $\mathsf{Q}_{\mathsf{Eval}}(f', x)$

Coupling with a uniqueness breaker [2/2]

$\mathsf{Adv}^f(x)$:

1. For $i \in [3t]$:
   a. Sample $f' \leftarrow \{0,1\}^*$ consistent with current view of f.
   b. Compute $y' := \mathsf{Eval}^{f'}(x)$.
   c. Let $\mathsf{Q}_{\mathsf{Eval}}(f', x)$ be the query set of $\mathsf{Eval}^{f'}(x)$.
   d. Query f with $\mathsf{Q}_{\mathsf{Eval}}(f', x)$ in one round.
2. Output $\mathsf{Majority}(y_1, \ldots, y_{3t})$.



$\mathsf{Adv}^f_{\mathsf{Uniq}}(x)$:

1. Compute $y := \mathsf{Eval}^f(x)$.
2. For $i \in [3t]$:
   a. Sample $f' \leftarrow \{0,1\}^*$ that agrees with f on $\{0,1\}^* \backslash \mathsf{Q}_{\mathsf{Eval}}(f, x)$.
   b. Compute $y' := \mathsf{Eval}^{f'}(x)$. If $y' \neq y$ and $\mathsf{Verify}^f(x, y') = 1$, output y'.
   c. Sample $f'' \leftarrow \{0,1\}^*$ consistent with current view of f and query f with $\mathsf{Q}_{\mathsf{Eval}}(f'', x)$ in one round.

Improved lower bounds for perfect uniqueness and statistical uniqueness

For every f and x,
$\mathsf{Verify}^f(x)$ accepts one and only one output y.

For every x,
there are at most $\mathrm{negl}(\lambda)$-fraction of f where
$\mathsf{Verify}^f(x)$ accepts more than one output y.

Proof of work function (PoWF)

T queries | t queries | $t \ll T$

$$\boxed{\mathsf{Solve}^f(x)} \xrightarrow{(y,\pi)} \boxed{\mathsf{Verify}^f(x,y,\pi)}$$

Completeness. For every security parameter λ and input x,

$$\Pr\left[\mathsf{Verify}^f(x,y,\pi)=1 \;\middle|\; \begin{array}{r} f \leftarrow \mathcal{O}(\lambda) \\ (y,\pi) \leftarrow \mathsf{Eval}^f(x) \end{array}\right] = 1.$$

Adv can be parallel/sequential, we only count total query complexity.

Soundness. For every security parameter λ, input x, and T′-query (T′ < T) adversary Adv,

$$\Pr\left[\exists \pi, \mathsf{Verify}^f(x,y,\pi)=1 \;\middle|\; \begin{array}{r} f \leftarrow \mathcal{O}(\lambda) \\ y \leftarrow \mathsf{Adv}^f(x) \end{array}\right] \le \mathsf{negl}(\lambda).$$

Computational Uniqueness. For every security parameter λ, input x, and poly(T)-query adversary Adv,

$$\Pr\left[\begin{array}{l} y \neq \mathsf{Eval}^f(x) \\ \wedge\, \mathsf{Verify}^f(x,y,\pi)=1 \end{array} \;\middle|\; \begin{array}{r} f \leftarrow \mathcal{O}(\lambda) \\ (y,\pi) \leftarrow \mathsf{Adv}^f(x) \end{array}\right] \le \mathsf{negl}(\lambda).$$

Statistically unique PoWF do not exist in the ROM

Theorem. Consider statistically unique $\mathsf{PoWF} = (\mathsf{Solve}, \mathsf{Verify})$ in the random oracle model (ROM). There exists a $O(\mathrm{t}^2)$-query adversary Adv that breaks the soundness of PoWF.

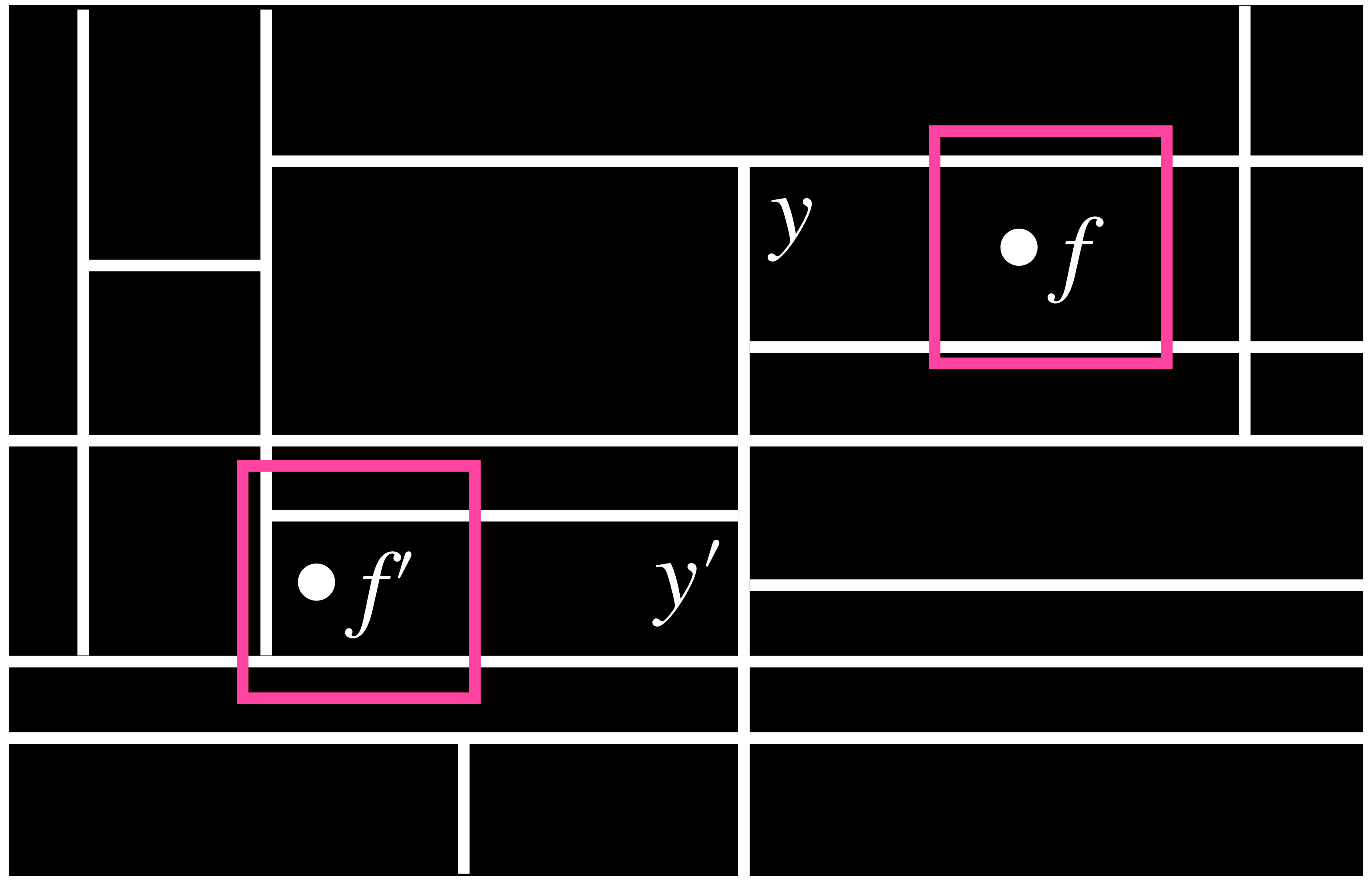
Corollary. Consider statistically unique $\mathsf{VDF} = (\mathsf{Eval}, \mathsf{Verify})$ in the random oracle model (ROM). There exists a $O(\mathrm{t}^2)$-query adversary Adv that breaks the sequentiality of VDF.

In the random oracle model

	With Sequentiality	Without Sequentiality
Perfect Uniqueness	Main Theorem: ❌	Theorem: ❌
Statistical Uniqueness	Main Theorem: ❌	Theorem: ❌
Computational Uniqueness	Main Theorem: ❌	Open Problem
No Uniqueness	Proof of sequential work [DLM19]: ✅	Proof of work [GKL15]: ✅

Proof sketch

$\{0,1\}^*$: set of all RO functions partitioned using Verify



Adv learns at least one query in $\mathsf{Q}_{\mathsf{Verify}}(f, x, y)$ each iteration

The rectangles are disjoint [BI87, AB09]:

- For every $f \neq f'$, $\exists q$ s.t. $f(q) \neq f'(q)$.
- Otherwise, $\mathsf{Verify}^{f}(x, y') = \mathsf{Verify}^{f'}(x, y') = 1$.

$\mathsf{Adv}^{f}(x)$:

t rounds of queries
At most t queries each round

1. For $i \in [\mathsf{t}]$:
   a. Pick $f' \in \{0,1\}^*$ consistent with current view of f.
   b. Compute $y' := \mathsf{Solve}^{f'}(x)$.
   c. Query f with $\mathsf{Q}_{\mathsf{Verify}}(f', x, y)$ in one round.
2. Output $y := \mathsf{Solve}^{f^\star}(x)$, where $f^\star$ is the current view of f.

Generalization to statistical uniqueness: approximate version of [BI87, AB09].
We extend [KSS11] to suit the context of VDF/PoWF.

Thank you!

https://eprint.iacr.org/2024/766