Lower Bound on SNARGs in the Random Oracle Model

Daniel Nukrai

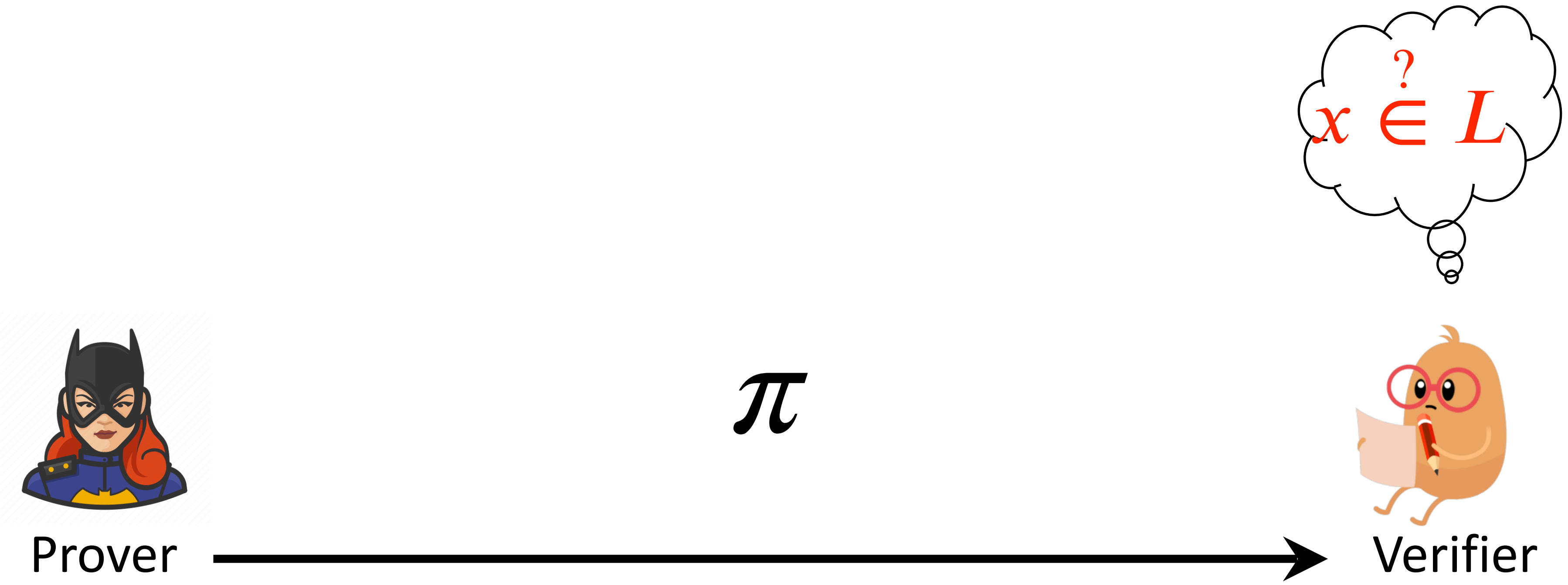
TEL AVIV UNIVERSITY

Joint work with

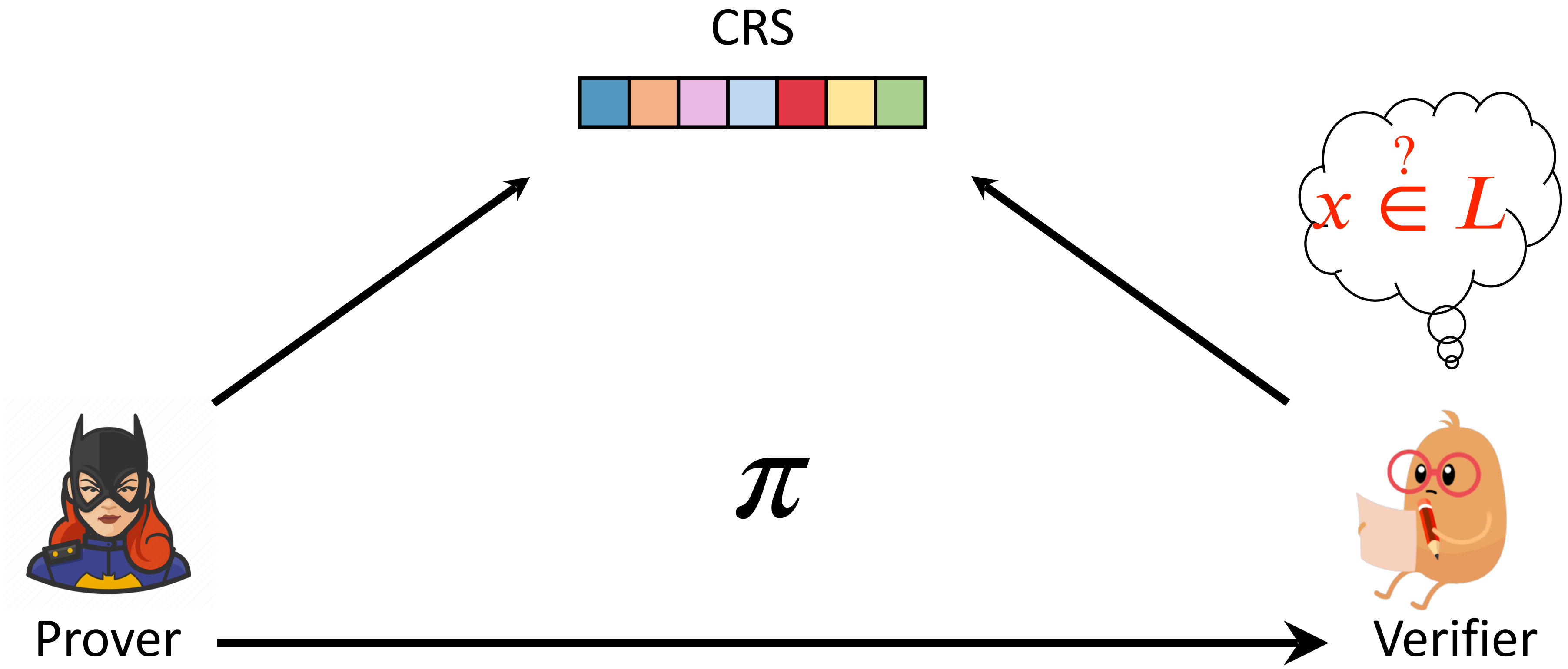
Iftach Haitner & Eylon Yogev

Succinct **N**on-Interactive **Arg**uments
SNARGS

SNARGs



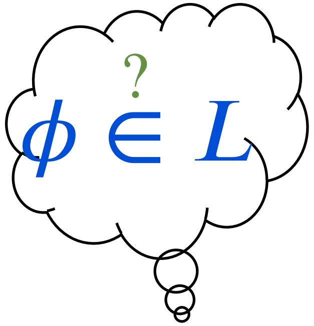
SNARGs



SNARGs in the ROM

SNARG: **S**uccinct **N**on-interactive **A**rgument

ROM: **R**andom **O**racle **M**odel



Prover

π

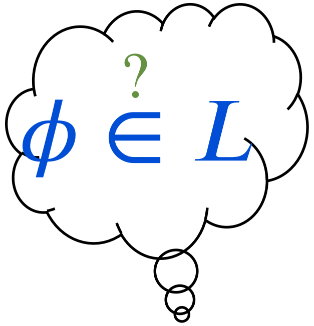
Verifier

SNARGs in the ROM

SNARG: **S**uccinct **N**on-interactive **A**rgument

ROM: **R**andom **O**racle **M**odel

$\zeta: \{0,1\}^* \to \{0,1\}^\lambda$



Prover

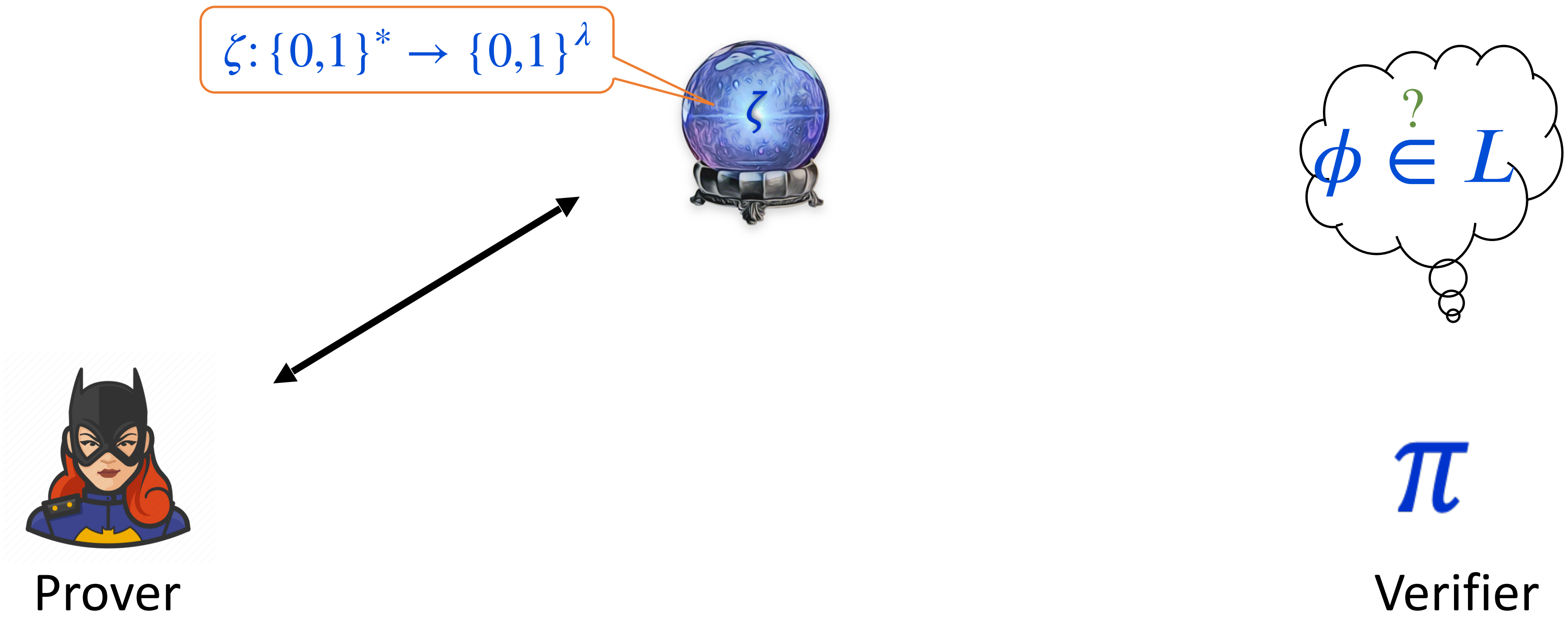
π

Verifier

SNARGs in the ROM

SNARG: **S**uccinct **N**on-interactive **A**rgument

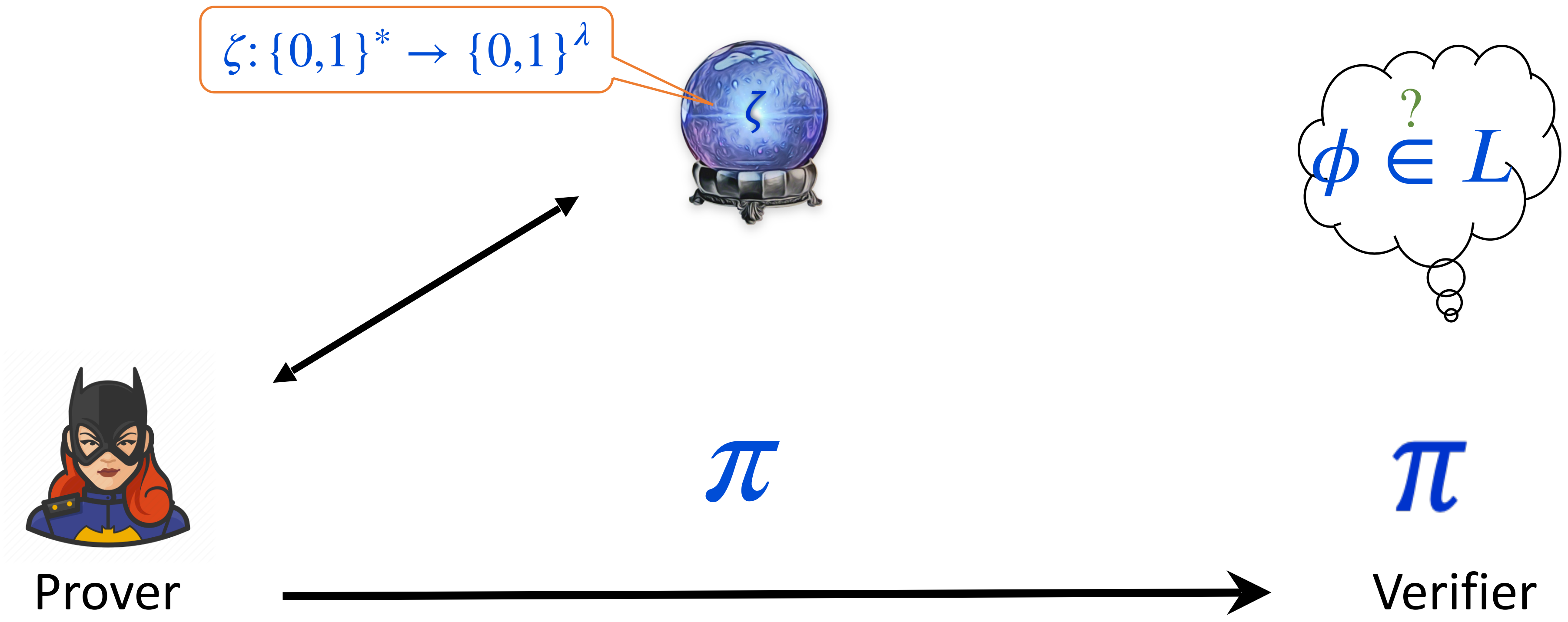
ROM: **R**andom **O**racle **M**odel



SNARGs in the ROM

SNARG: **S**uccinct **N**on-interactive **A**rgument

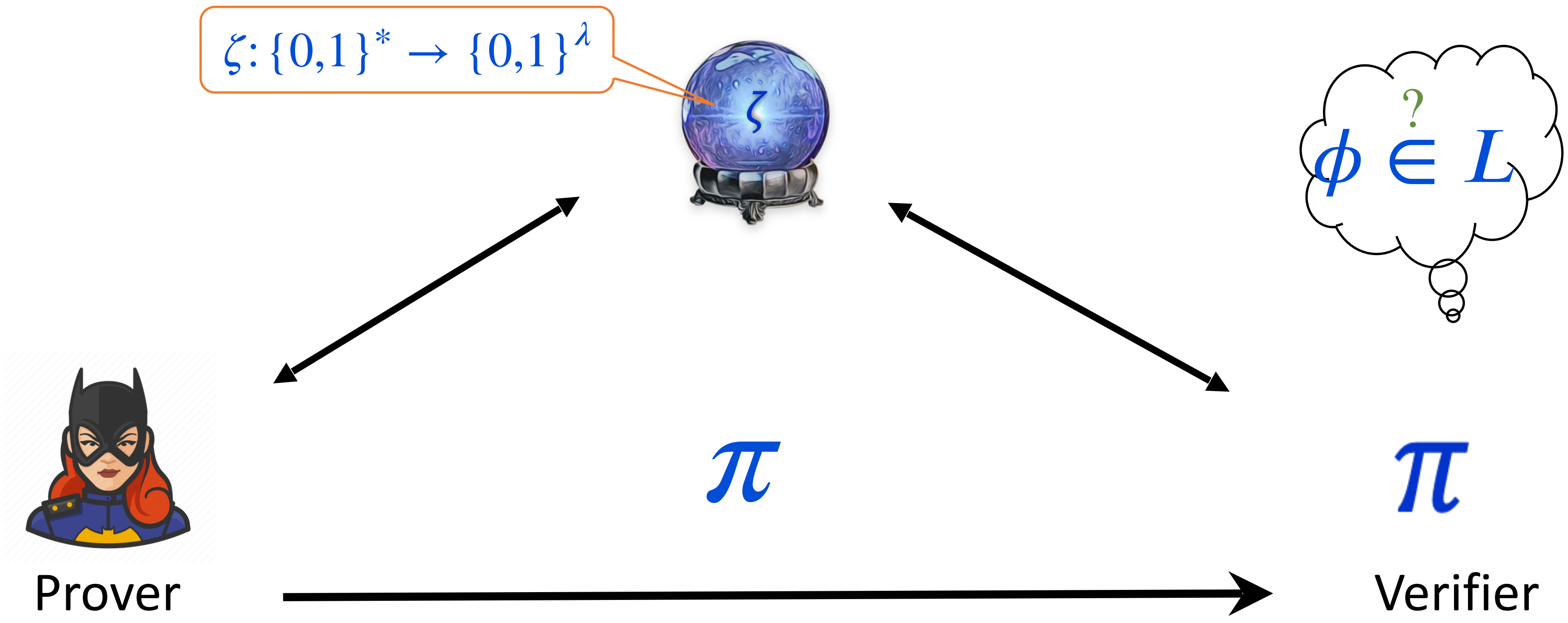
ROM: **R**andom **O**racle **M**odel



SNARGs in the ROM

SNARG: **S**uccinct **N**on-interactive **A**rgument

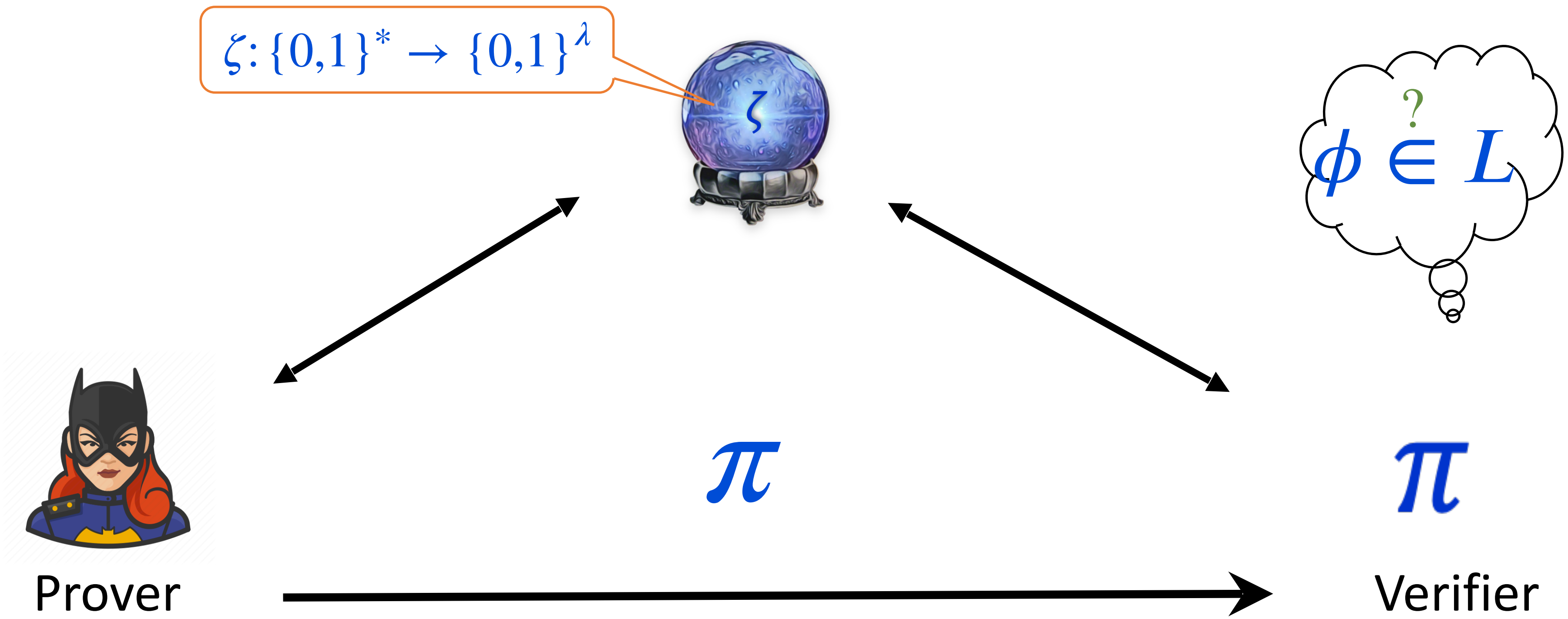
ROM: **R**andom **O**racle **M**odel



SNARGs in the ROM

SNARG: **S**uccinct **N**on-interactive **A**rgument

ROM: **R**andom **O**racle **M**odel

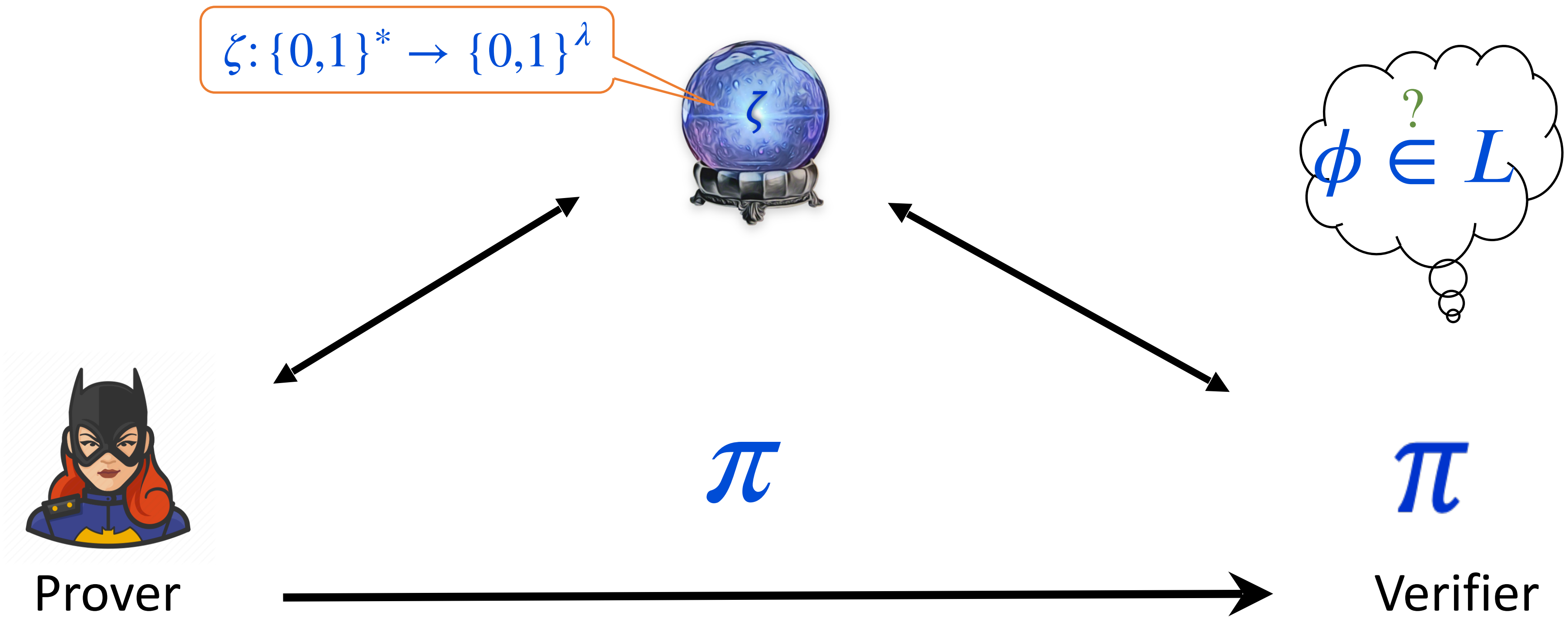


- Soundness against (computationally unbounded) query bounded provers

SNARGs in the ROM

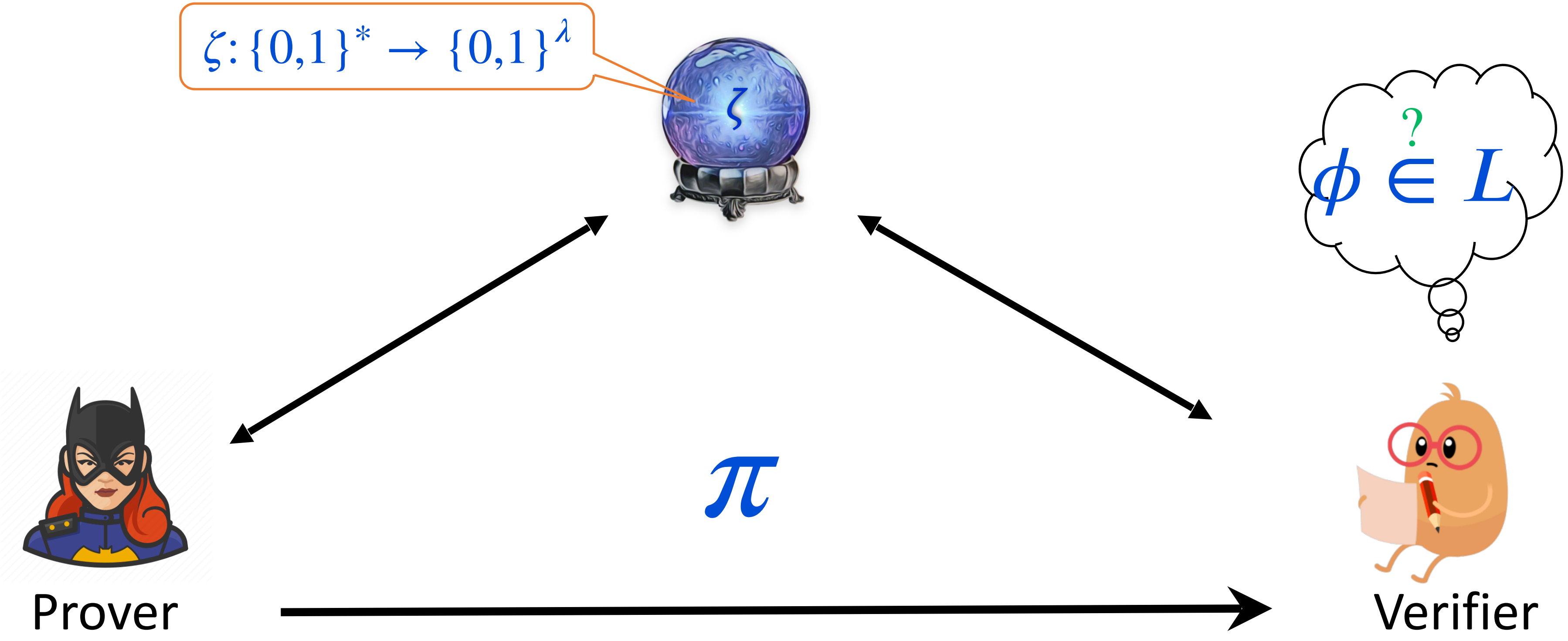
SNARG: **S**uccinct **N**on-interactive **A**rgument

ROM: **R**andom **O**racle **M**odel

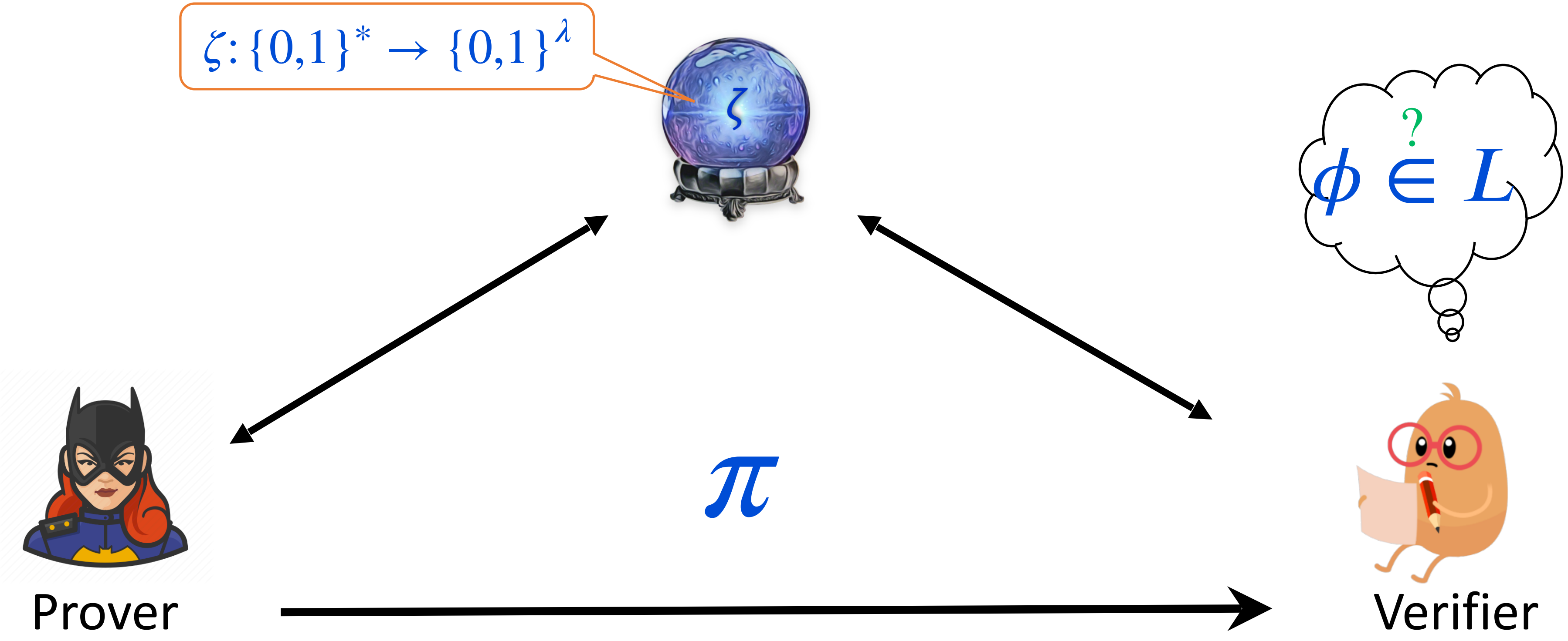


- Soundness against (computationally unbounded) query bounded provers
- $2^\lambda \gg$ instance size (n) and cheating prover running time (t)

Completeness



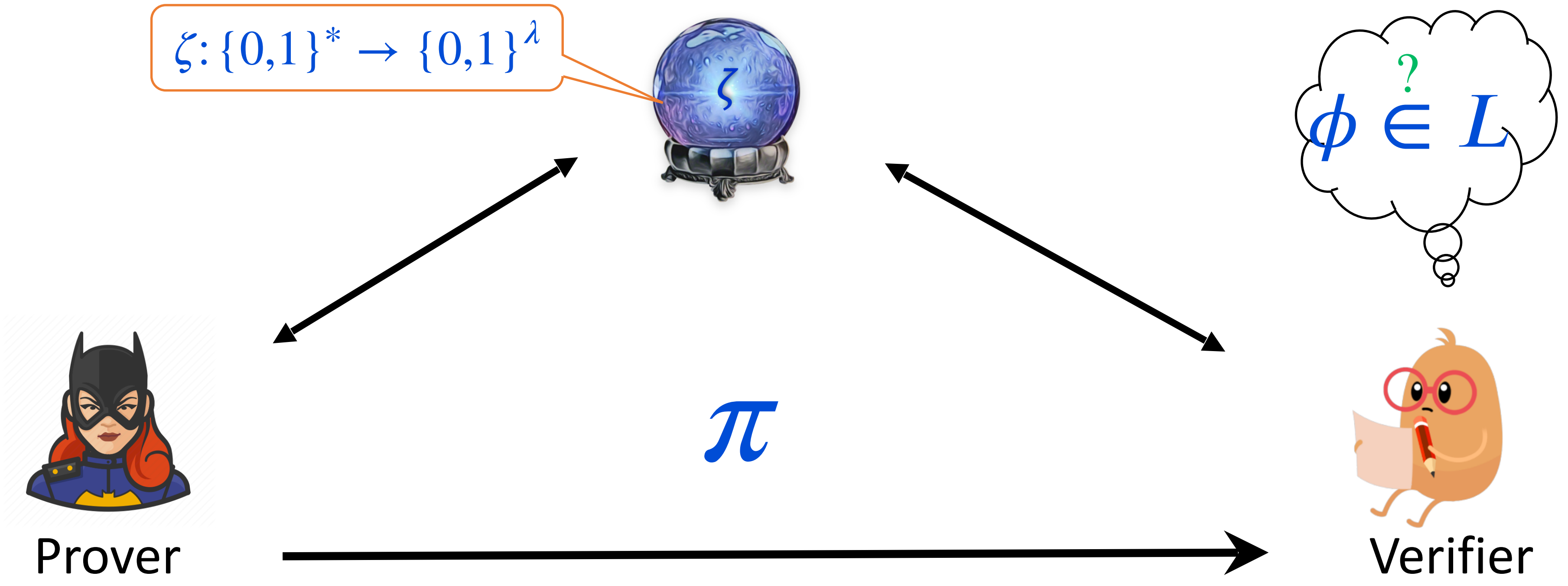
Completeness



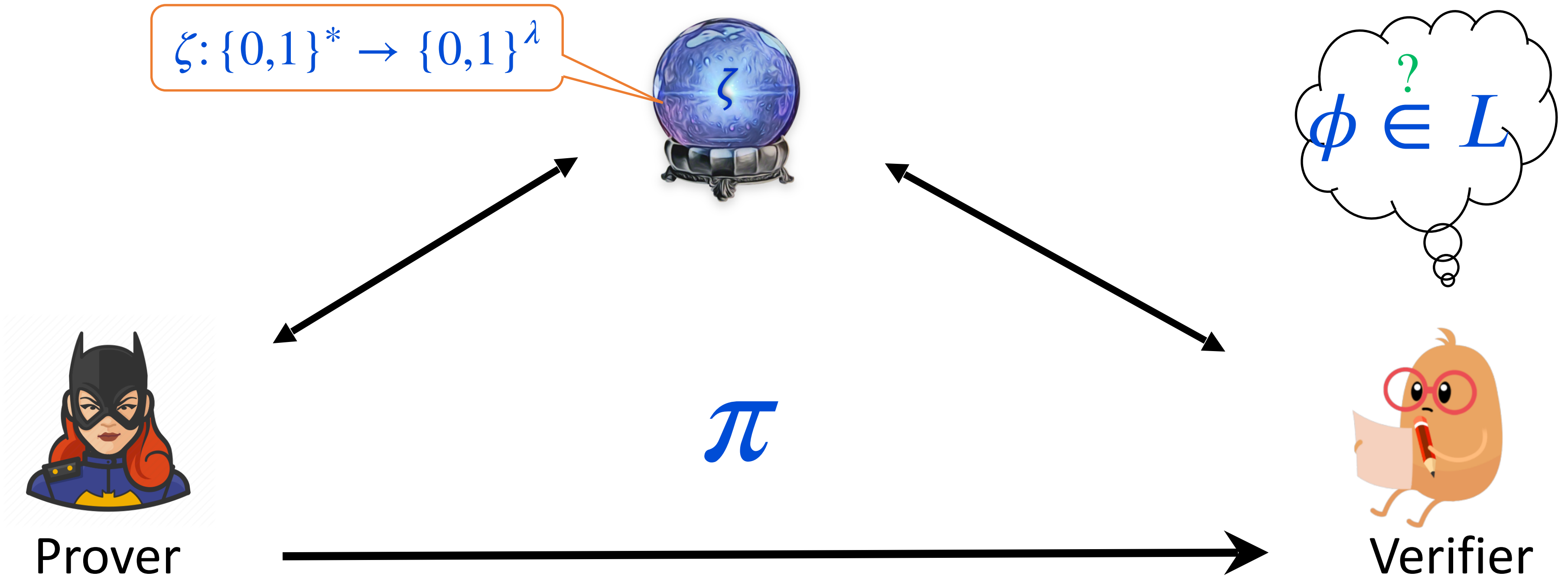
α-completeness: for every $\phi \in L$:

$$\Pr_{\zeta}\left[V^{\zeta}(\phi, \pi) = 1 : \pi \leftarrow P^{\zeta}\right] \geq \alpha$$

(t, ϵ)-soundness



(t, ϵ)-soundness



(t, ϵ)**-soundness:** for any $\phi \notin L$ and t-query (comp. unbounded) $\tilde{P}$:

$$\Pr_{\zeta}\left[V^{\zeta}(\phi, \pi) = 1 : \pi \leftarrow \tilde{P}^{\zeta}\right] \leq \epsilon$$

Importance of the ROM

Importance of the ROM

- **Simple** information-theoretic model

Importance of the ROM

- **Simple** information-theoretic model
- Supports many well-known constructions

Importance of the ROM

- **Simple** information-theoretic model
- Supports many well-known constructions
- Supports many well-known lower bounds

Importance of the ROM

- **Simple** information-theoretic model
- Supports many well-known constructions
- Supports many well-known lower bounds
- ROM huristic: ROM is instantiated via **lightweight** crypto (e.g. SHA-256)

Importance of the ROM

- **Simple** information-theoretic model
- Supports many well-known constructions
- Supports many well-known lower bounds
- ROM huristic: ROM is instantiated via **lightweight** crypto (e.g. SHA-256)

Constructions in ROM huristic are:

Importance of the ROM

- **Simple** information-theoretic model
- Supports many well-known constructions
- Supports many well-known lower bounds
- ROM huristic: ROM is instantiated via **lightweight** crypto (e.g. SHA-256)

Constructions in ROM huristic are:

- Fast to compute

Importance of the ROM

- **Simple** information-theoretic model
- Supports many well-known constructions
- Supports many well-known lower bounds
- ROM huristic: ROM is instantiated via **lightweight** crypto (e.g. SHA-256)

Constructions in ROM huristic are:

- Fast to compute
- No trusted setup

Importance of the ROM

- **Simple** information-theoretic model
- Supports many well-known constructions
- Supports many well-known lower bounds
- ROM huristic: ROM is instantiated via **lightweight** crypto (e.g. SHA-256)

Constructions in ROM huristic are:

- Fast to compute
- No trusted setup
- Potentially post-quantum …

Importance of the ROM

- **Simple** information-theoretic model
- Supports many well-known constructions
- Supports many well-known lower bounds
- ROM huristic: ROM is instantiated via **lightweight** crypto (e.g. SHA-256)

Constructions in ROM huristic are:

- Fast to compute
- No trusted setup
- Potentially post-quantum **…**
- Widely used in practice

Known ROM-SNARGS constructions

Known ROM-SNARGS constructions

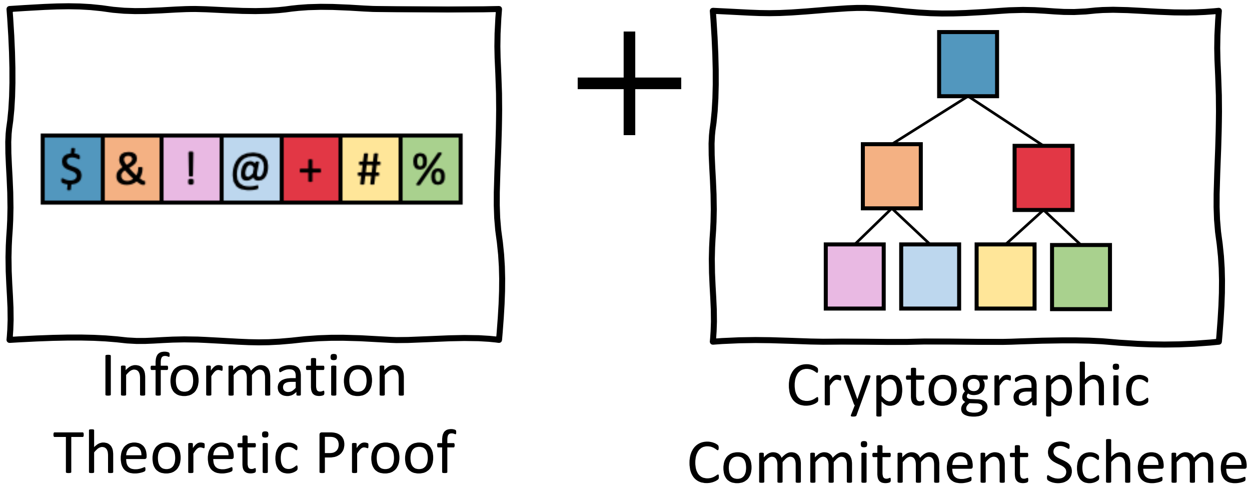
- Micali'94, BCS'16:
  - Proof length: $O\left(\left(\log\frac{t}{\epsilon}\right)^2 \cdot \log n\right)$
  - # verifier queries: $\Theta\left(\log\frac{t}{\epsilon}\right)$

Known ROM-SNARGS constructions

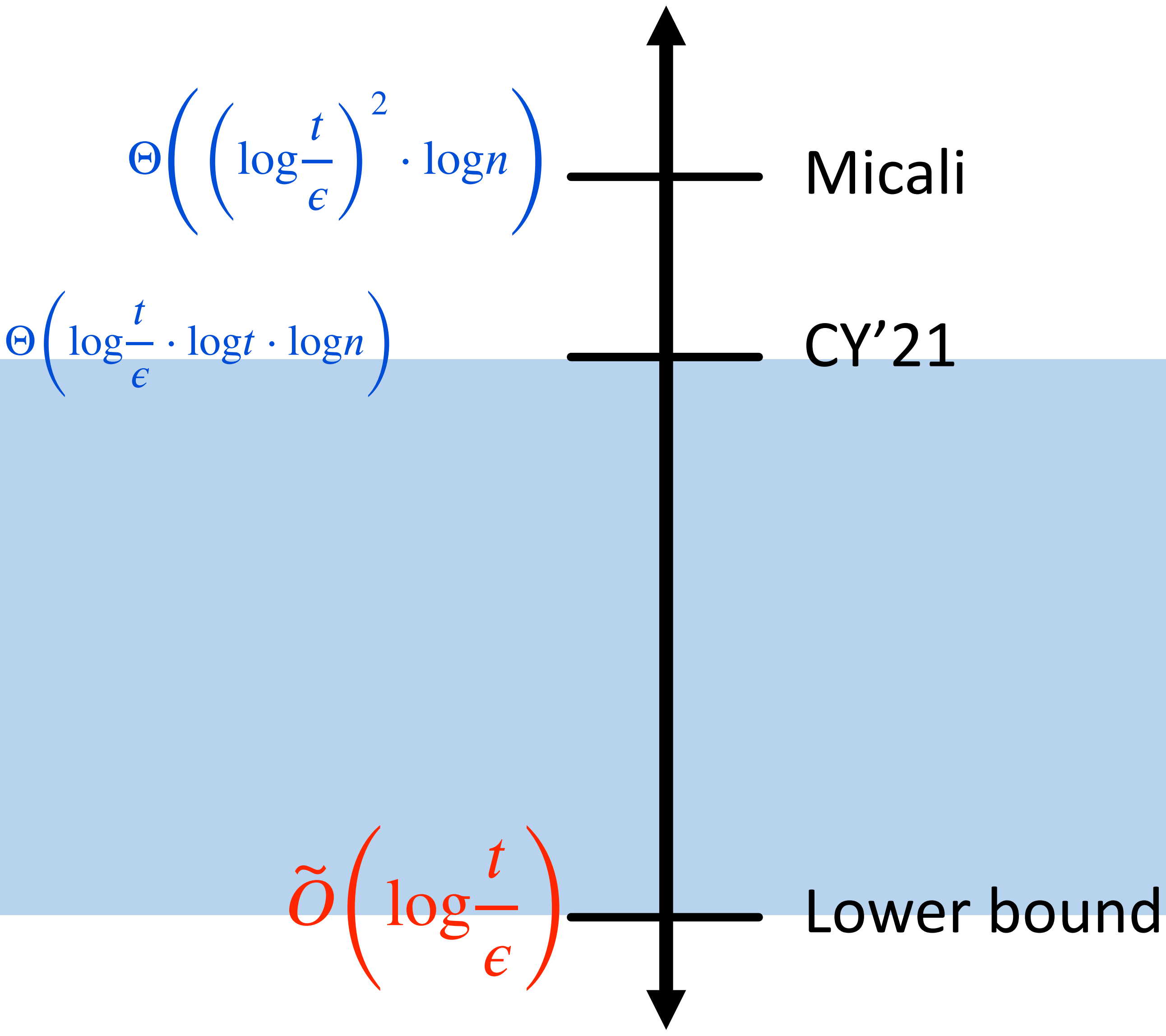
- Micali'94, BCS'16:
  - Proof length: $O\left(\left(\log\frac{t}{\epsilon}\right)^2 \cdot \log n\right)$
  - # verifier queries: $\Theta\left(\log\frac{t}{\epsilon}\right)$
- CY'21:
  - Proof length: $O\left(\log\frac{t}{\epsilon} \cdot \log t \cdot \log n\right)$
  - # verifier queries: $\Theta\left(\log\frac{t}{\epsilon}\right)$

Known ROM-SNARGS constructions

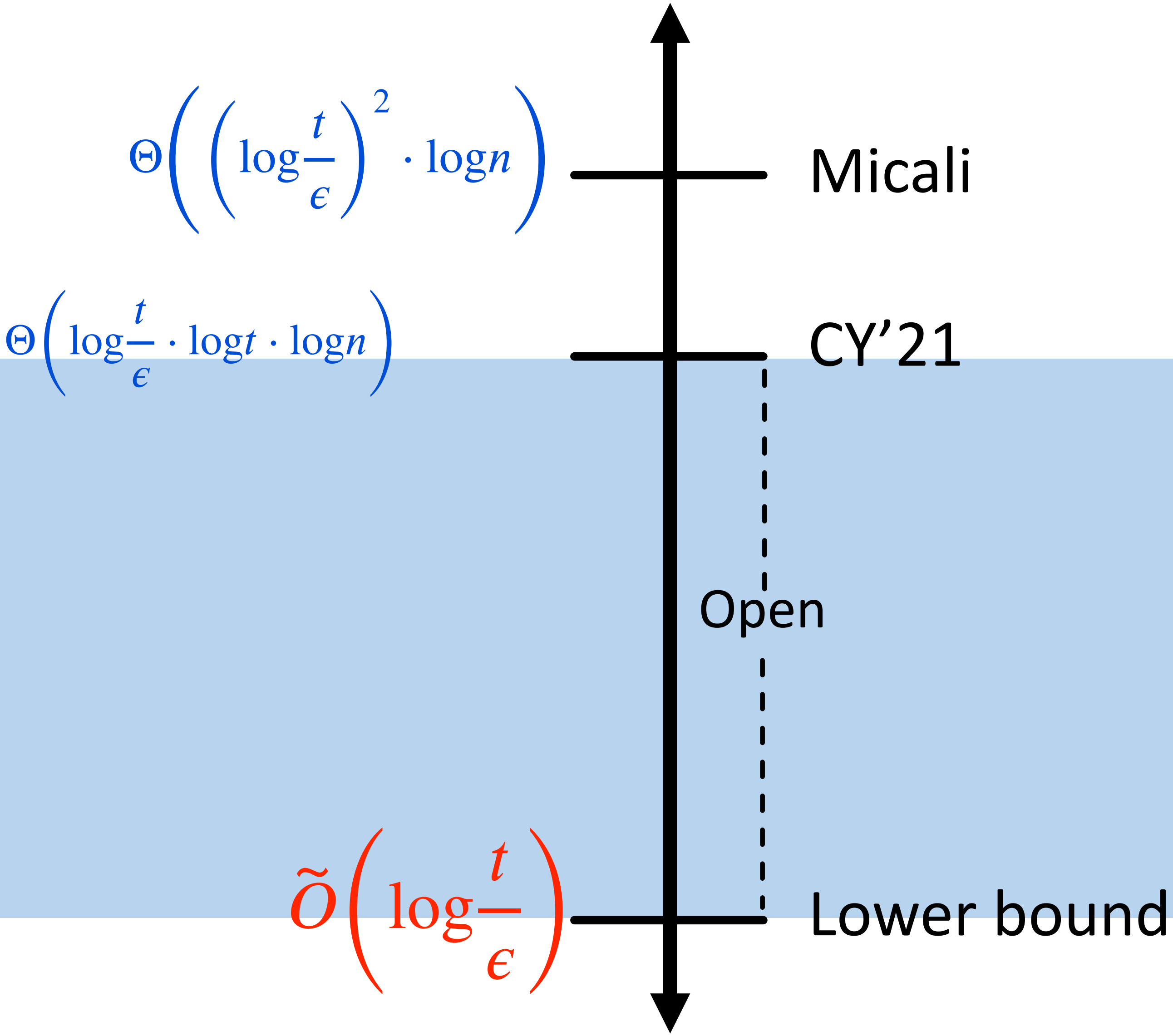
- Micali'94, BCS'16:
  - Proof length: $O\left(\left(\log\frac{t}{\epsilon}\right)^2 \cdot \log n\right)$
  - # verifier queries: $\Theta\left(\log\frac{t}{\epsilon}\right)$
- CY'21:
  - Proof length: $O\left(\log\frac{t}{\epsilon} \cdot \log t \cdot \log n\right)$
  - # verifier queries: $\Theta\left(\log\frac{t}{\epsilon}\right)$



Proof length



Proof length



Our lower bound

Our lower bound

Thm: Assuming rnd ETH, any "natural" ROM-SNARG (P, V) of (t, ϵ)-soundness has proof size $\Omega\left(\log\frac{t}{\epsilon} \cdot \log t / \log q_P\right)$

Tight up to $\log \mathrm{n} \cdot \log \mathrm{q}_P$ term ([CY'21] proof size is $\Theta\left(\log\frac{t}{\epsilon} \cdot \log t \cdot \log n\right)$)

Our lower bound

Thm: Assuming rnd ETH, any "natural" ROM-SNARG (P, V) of (t, ϵ)-soundness has proof size $\Omega\left(\log\frac{t}{\epsilon} \cdot \log t / \log q_P\right)$

Tight up to $\log n \cdot \log q_P$ term ([CY'21] proof size is $\Theta\left(\log\frac{t}{\epsilon} \cdot \log t \cdot \log n\right)$)

Natural constructions:

Our lower bound

Thm: Assuming rnd ETH, any "natural" ROM-SNARG (P, V) of (t, ϵ)-soundness has proof size $\Omega\left(\log\frac{t}{\epsilon} \cdot \log t / \log q_P\right)$

Tight up to $\log n \cdot \log q_P$ term ([CY'21] proof size is $\Theta\left(\log\frac{t}{\epsilon} \cdot \log t \cdot \log n\right)$)

Natural constructions:

1. Non-adaptive deterministic verifier

Our lower bound

Thm: Assuming rnd ETH, any "natural" ROM-SNARG (P, V) of (t, ϵ)-soundness has proof size $\Omega\left(\log\frac{t}{\epsilon} \cdot \log t / \log q_P\right)$

Tight up to $\log \mathrm{n} \cdot \log \mathrm{q}_P$ term ([CY'21] proof size is $\Theta\left(\log\frac{t}{\epsilon} \cdot \log t \cdot \log n\right)$)

Natural constructions:

1. Non-adaptive deterministic verifier
2. Salted soundness

Our lower bound

Thm: Assuming rnd ETH, any "natural" ROM-SNARG (P, V) of (t, ϵ)-soundness has proof size $\Omega\left(\log\frac{t}{\epsilon} \cdot \log t / \log q_P\right)$

Tight up to $\log \mathrm{n} \cdot \log \mathrm{q}_P$ term ([CY'21] proof size is $\Theta\left(\log\frac{t}{\epsilon} \cdot \log t \cdot \log n\right)$)

Natural constructions:

1. Non-adaptive deterministic verifier
2. Salted soundness
3. Reasonable q_P and q_V (P/V query complexity) as functions of n

Our lower bound

Thm: Assuming rnd ETH, any "natural" ROM-SNARG (P, V) of (t, ϵ)-soundness has proof size $\Omega\left(\log\frac{t}{\epsilon} \cdot \log t / \log q_P\right)$

Tight up to $\log \mathrm{n} \cdot \log \mathrm{q}_P$ term ([CY'21] proof size is $\Theta\left(\log\frac{t}{\epsilon} \cdot \log t \cdot \log n\right)$)

Natural constructions:

1. Non-adaptive deterministic verifier
2. Salted soundness
3. Reasonable q_P and q_V (P/V query complexity) as functions of n

All known (non-contrived) constructions are natural

Proof size for natural constructions

- $\Theta\left(\left(\log\frac{t}{\epsilon}\right)^2 \cdot \log n\right)$ — Micali
- $\Theta\left(\log\frac{t}{\epsilon} \cdot \log t \cdot \boldsymbol{\log n}\right)$ — CY’21
- $\Omega\left(\log\frac{t}{\epsilon} \cdot \log t / \boldsymbol{\log q_P}\right)$ — This work
- $\Omega\left(\log\frac{t}{\epsilon}\right)$ — Lower bound

Lower bound on ROM SubVector Commitment

Subvector commitment (SVC) – **non-interactive** cmt with local opening.

Lower bound on ROM SubVector Commitment

Subvector commitment (SVC) – **non-interactive** cmt with local opening.

- (t, ϵ)-binding in ROM

Lower bound on ROM SubVector Commitment

Subvector commitment (SVC) – **non-interactive** cmt with local opening.

- (t, ϵ)-binding in ROM
- α – commitment length

Lower bound on ROM SubVector Commitment

Subvector commitment (SVC) – **non-interactive** cmt with local opening.

- (t, ϵ)-binding in ROM
- α – commitment length
- $\beta(m)$ – length of opening m elements.

Lower bound on ROM SubVector Commitment

Subvector commitment (SVC) – **non-interactive** cmt with local opening.

- (t, ϵ)-binding in ROM
- α – commitment length
- $\beta(m)$ – length of opening m elements.

Thm: Assuming rnd ETH, any "natural" ROM-SVC (S, R) of (t, ϵ)-binding has $\alpha + \beta\left(\log\frac{t}{\epsilon}\right) \in \Omega\left(\log\frac{t}{\epsilon} \cdot \log t/\log q_S\right)$

Lower bound on ROM SubVector Commitment

Subvector commitment (SVC) – **non-interactive** cmt with local opening.

- (t, ϵ)-binding in ROM
- α – commitment length
- $\beta(m)$ – length of opening m elements.

Thm: Assuming rnd ETH, any "natural" ROM-SVC (S, R) of

(t, ϵ)-binding has $\alpha + \beta\left(\log\frac{t}{\epsilon}\right) \in \Omega\left(\log\frac{t}{\epsilon} \cdot \log t / \log q_S\right)$

- Tight bound upto $\log \mathrm{n} \cdot \log \mathrm{q}_S$ term (n is commited string length)

Lower bound on ROM SubVector Commitment

Subvector commitment (SVC) – **non-interactive** cmt with local opening.

- (t, ϵ)-binding in ROM
- α – commitment length
- $\beta(m)$ – length of opening m elements.

Thm: Assuming rnd ETH, any "natural" ROM-SVC (S, R) of (t, ϵ)-binding has $\alpha + \beta\left(\log\frac{t}{\epsilon}\right) \in \Omega\left(\log\frac{t}{\epsilon} \cdot \log t / \log q_S\right)$

- Tight bound upto $\log \mathrm{n} \cdot \log \mathrm{q}_S$ term (n is commited string length)

- **How to prove:** SVC + PCP $\rightarrow$ SNARG

Lower bound on ROM SubVector Commitment

Subvector commitment (SVC) – **non-interactive** cmt with local opening.

- (t, ϵ)-binding in ROM
- α – commitment length
- $\beta(m)$ – length of opening m elements.

Thm: Assuming rnd ETH, any "natural" ROM-SVC (S, R) of (t, ϵ)-binding has $\alpha + \beta\left(\log\frac{t}{\epsilon}\right) \in \Omega\left(\log\frac{t}{\epsilon} \cdot \log t / \log q_S\right)$

- Tight bound upto $\log \mathrm{n} \cdot \log \mathrm{q}_S$ term (n is commited string length)

- **How to prove:** SVC + PCP $\rightarrow$ SNARG

Salted Soundness

Malicious prover can resample queries, and choose the answers he likes

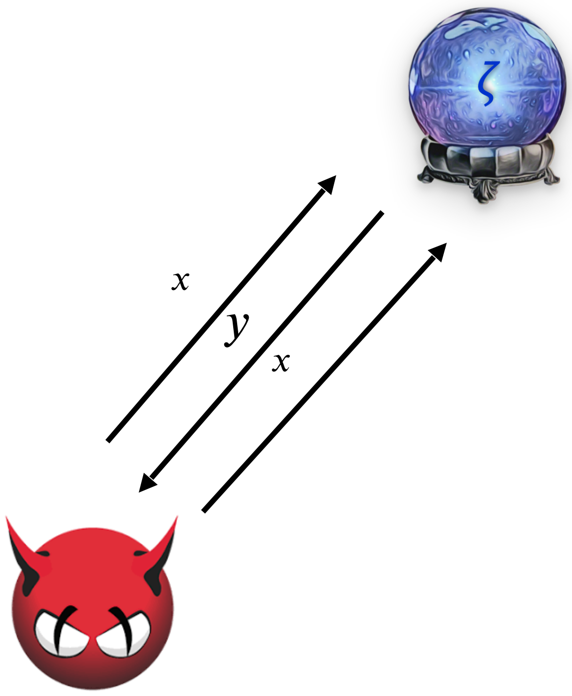
Salted Soundness

Malicious prover can resample queries, and choose the answers he likes

Salted Soundness

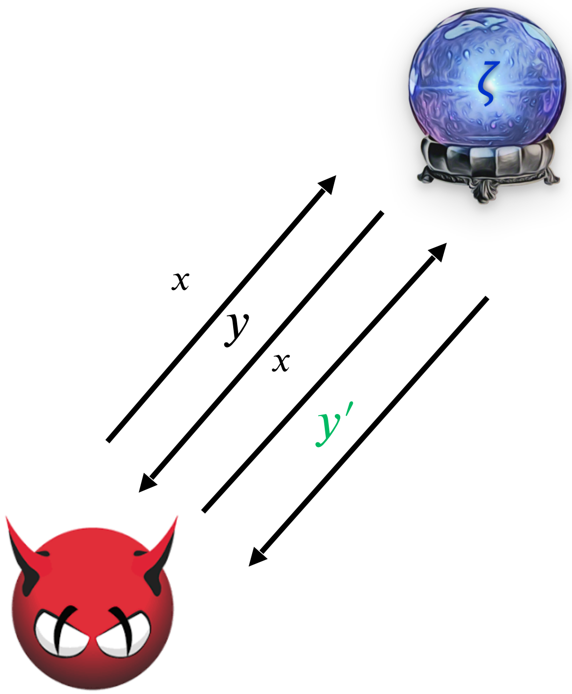
Malicious prover can resample queries, and choose the answers he likes

Salted Soundness



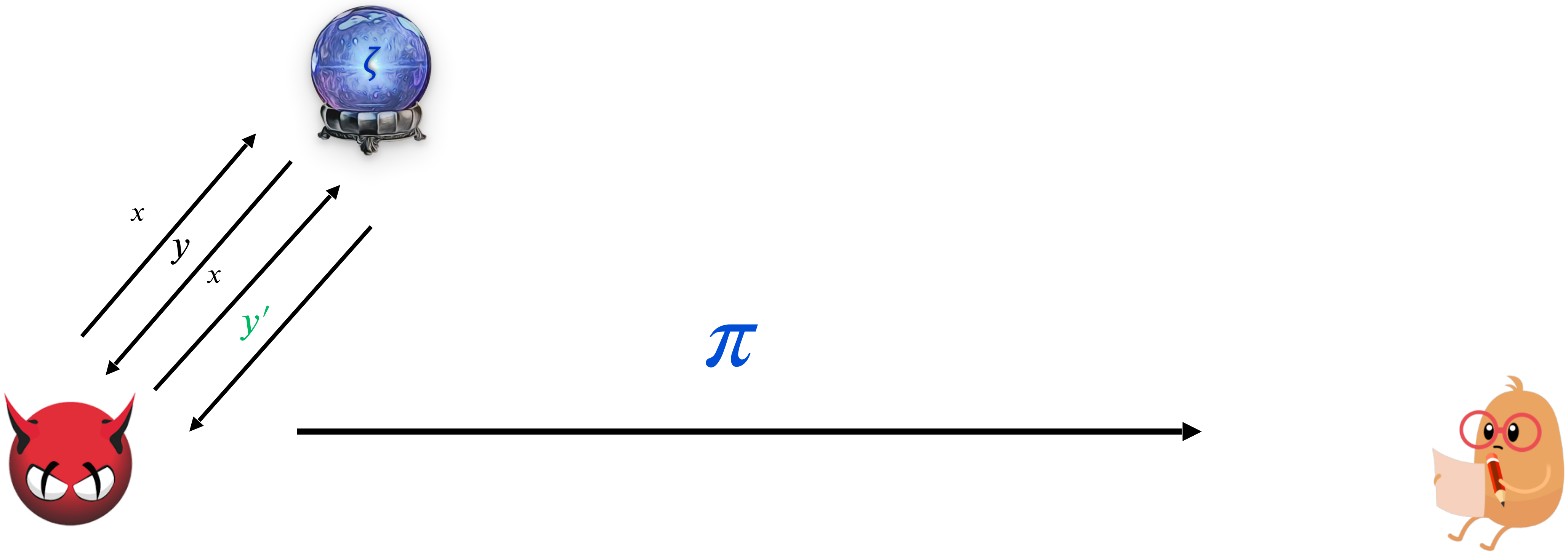
Malicious prover can resample queries, and choose the answers he likes

Salted Soundness



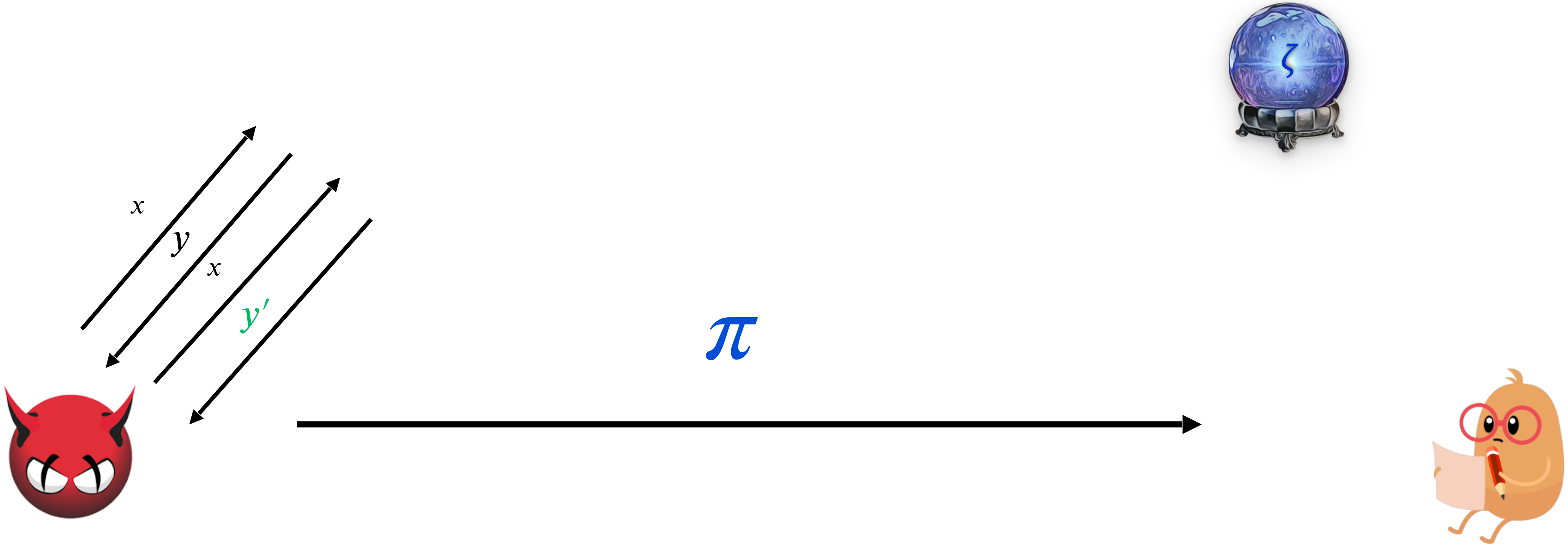
Malicious prover can resample queries, and choose the answers he likes

Salted Soundness



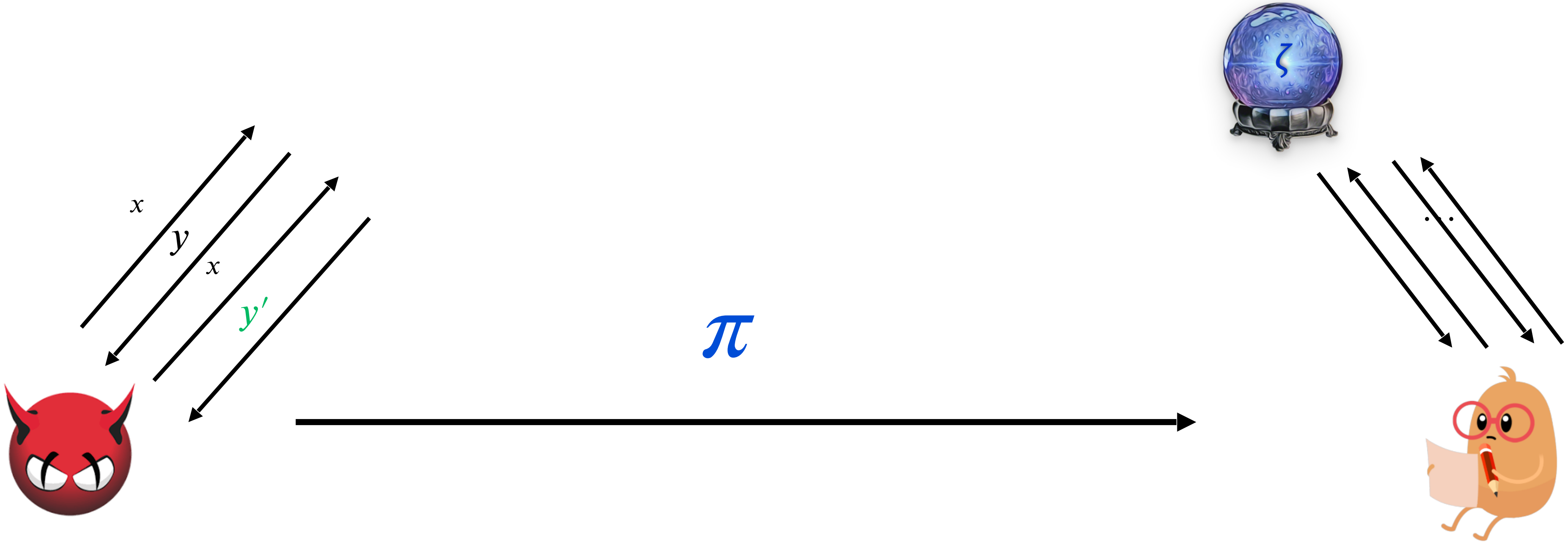
Malicious prover can resample queries, and choose the answers he likes

Salted Soundness



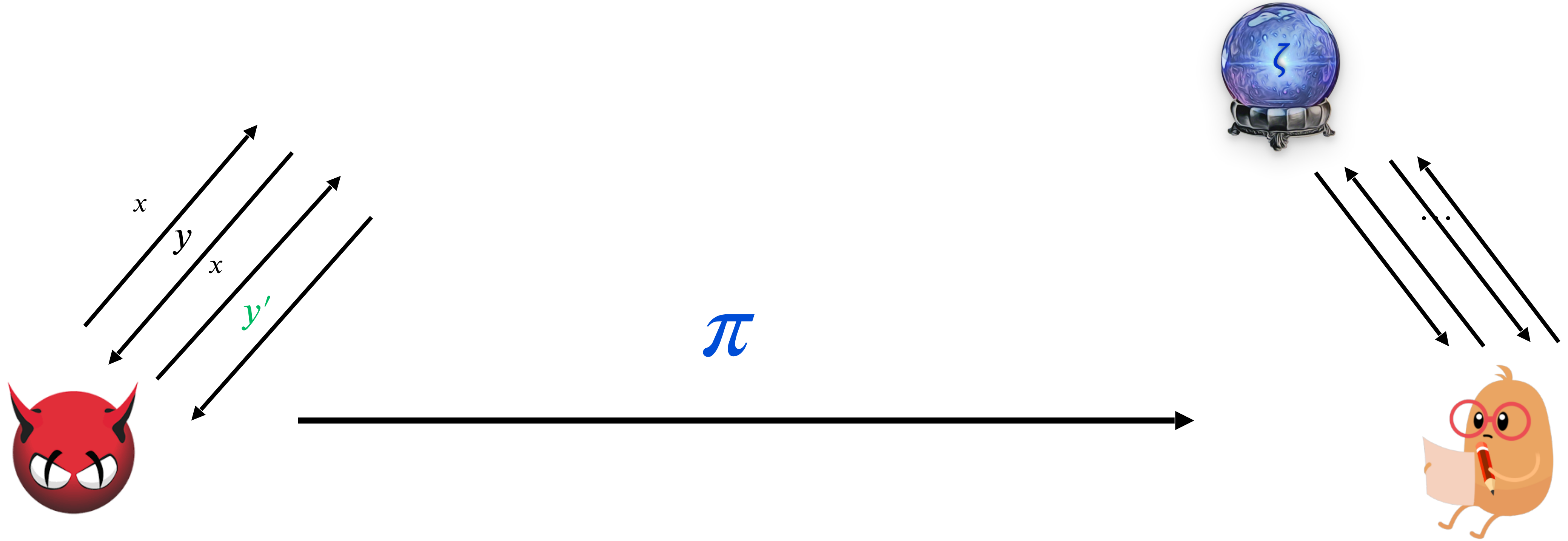
Malicious prover can resample queries, and choose the answers he likes

Salted Soundness



Malicious prover can resample queries, and choose the answers he likes

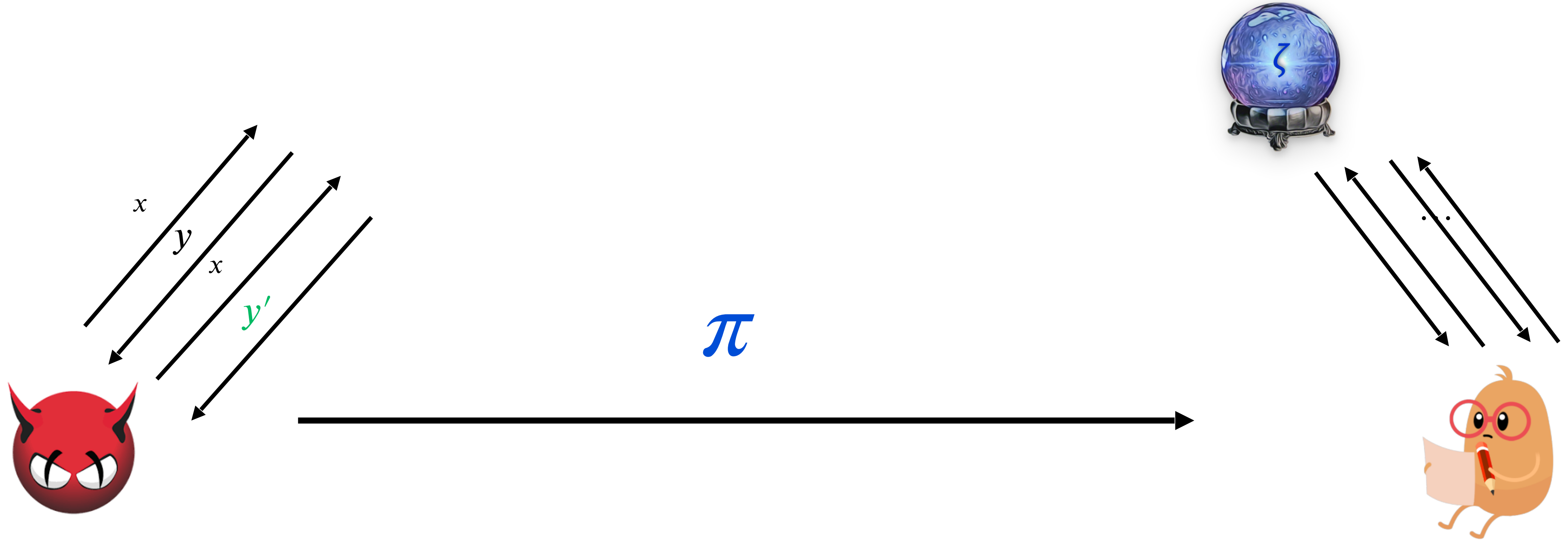
Salted Soundness



Malicious prover can resample queries, and choose the answers he likes

- All known constructions have salted soundness

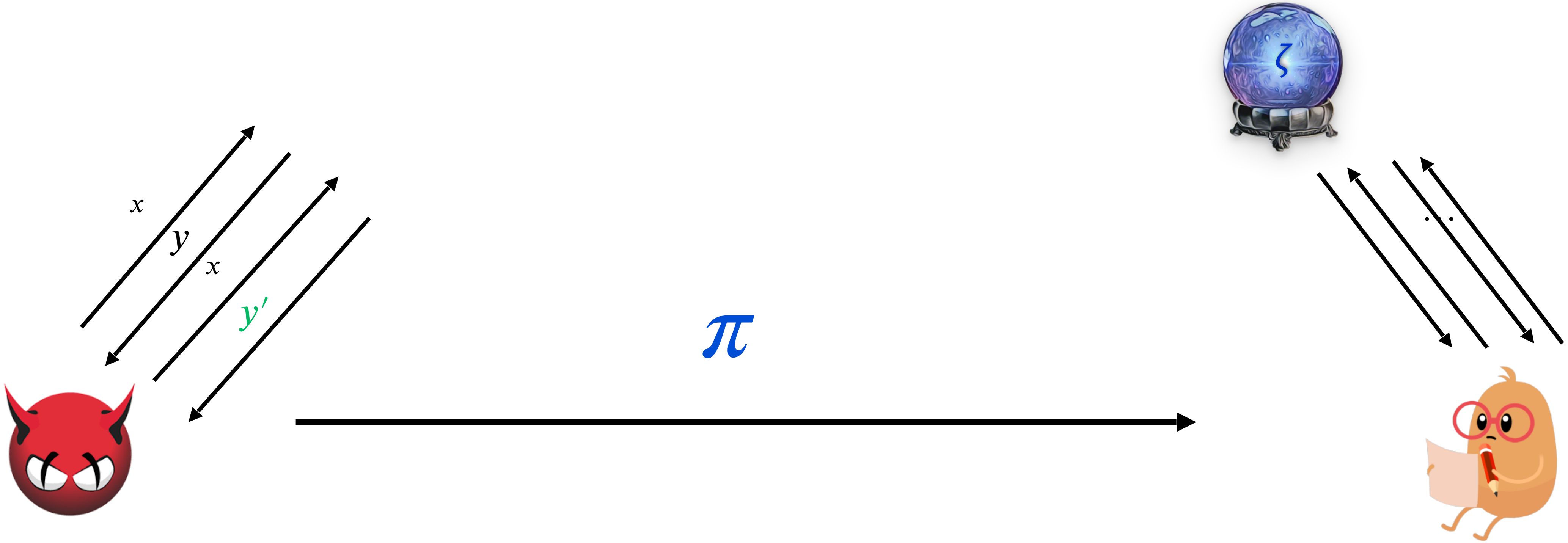
Salted Soundness



Malicious prover can resample queries, and choose the answers he likes

- All known constructions have salted soundness
- Easy to construct a SNARG that has no salted soundness

Salted Soundness



Malicious prover can resample queries, and choose the answers he likes

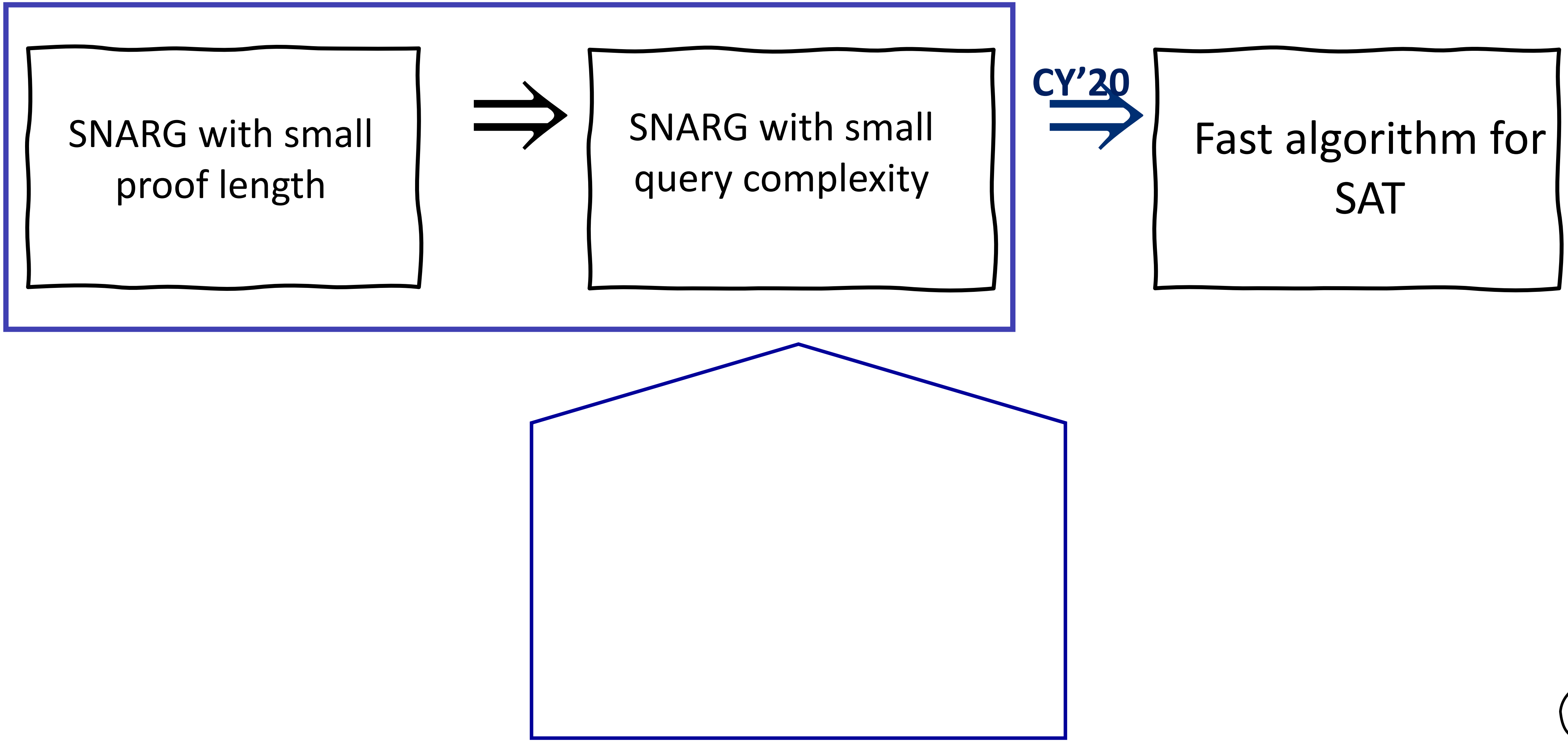
- All known constructions have salted soundness
- Easy to construct a SNARG that has no salted soundness
- Seems hard to get rid of w/o making the verifier adaptive

Short SNARGS to Fast Algorithms

SNARG with small proof length ⇒ SNARG with small query complexity

CY'20

Short SNARGS to Fast Algorithms



Short SNARGS to Fast Algorithms

SNARG with small proof length

⇒

SNARG with small query complexity

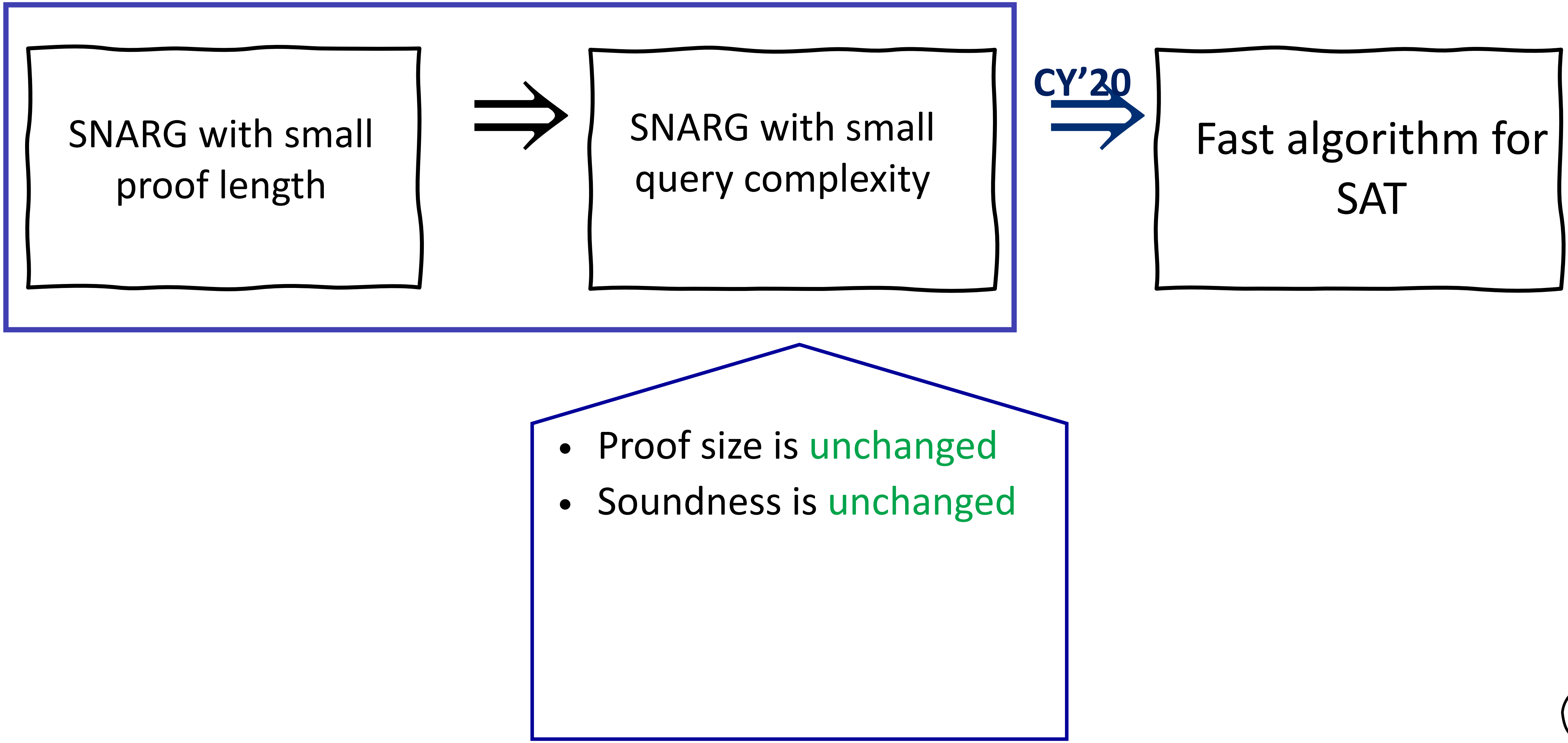
CY'20

⇒

Fast algorithm for SAT

- Proof size is unchanged

Short SNARGS to Fast Algorithms



Short SNARGS to Fast Algorithms

SNARG with small proof length ⇒ SNARG with small query complexity

CY'20 ⇒ Fast algorithm for SAT

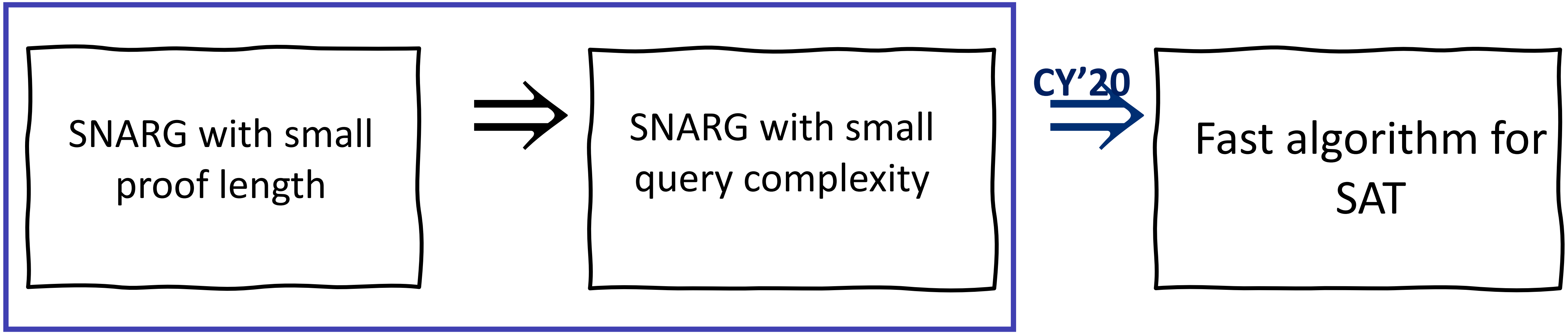
- Proof size is unchanged
- Soundness is unchanged
- Nontrivial completeness

Short SNARGS to Fast Algorithms

SNARG with small proof length ⇒ SNARG with small query complexity ⇒ (CY'20) Fast algorithm for SAT

- Proof size is unchanged
- Soundness is unchanged
- Nontrivial completeness
- Verifier running time: $t^{1/C}$

Short SNARGS to Fast Algorithms



- Proof size is unchanged
- Soundness is unchanged
- Nontrivial completeness
- Verifier running time: $t^{1/C}$
- Query complexity $\log \frac{t}{\epsilon}$

Short SNARGs to Low-query SNARGs

$\tilde{V}$

Short SNARGs to Low-query SNARGs

Given SNARG (P, V), we modify to $\tilde{V}$ as follows (P is unchanged):

$\tilde{V}$

Short SNARGs to Low-query SNARGs

Given SNARG (P, V), we modify to $\tilde{V}$ as follows (P is unchanged):

Oracle: $\zeta : \{0,1\}^* \rightarrow \{0,1\}^\lambda$

$\tilde{V}$

Short SNARGs to Low-query SNARGs

Given SNARG (P, V), we modify to $\tilde{V}$ as follows (P is unchanged):

$\tilde{V}$

Oracle: $\zeta: \{0,1\}^* \rightarrow \{0,1\}^\lambda$

Input: (ϕ, π)

Short SNARGs to Low-query SNARGs

Given SNARG (P, V), we modify to $\tilde{V}$ as follows (P is unchanged):

Oracle: $\zeta: \{0,1\}^* \rightarrow \{0,1\}^\lambda$

Input: (ϕ, π)

$\tilde{V}$

1. Let $u_1, \ldots, u_m$ denote the queries of $V(\phi, \pi)$ (Recall, V is non-adaptive)

Short SNARGs to Low-query SNARGs

Given SNARG $(P,\ V)$, we modify to $\tilde{V}$ as follows (P is unchanged):

$\tilde{V}$

Oracle: $\zeta : \{0,1\}^* \to \{0,1\}^\lambda$

Input: (ϕ, π)

1. Let $u_1, \ldots, u_m$ denote the queries of $V(\phi, \pi)$ (Recall, V is non-adaptive)
2. Sample unifrom k-size subset $J \subseteq [m]$

Short SNARGs to Low-query SNARGs

Given SNARG $(P,\ V)$, we modify to $\tilde{V}$ as follows (P is unchanged):

Oracle: $\zeta : \{0,1\}^* \to \{0,1\}^\lambda$

Input: (ϕ, π)

$\tilde{V}$

1. Let $u_1, \ldots, u_m$ denote the queries of $V(\phi, \pi)$ (Recall, V is non-adaptive)
2. Sample unifrom k-size subset $J \subseteq [m]$
3. For each $j \in J$: let $a_j = \zeta(u_j)$

Short SNARGs to Low-query SNARGs

Given SNARG (P, V), we modify to $\tilde{V}$ as follows (P is unchanged):

$\tilde{V}$

Oracle: $\zeta : \{0,1\}^* \to \{0,1\}^\lambda$

Input: (ϕ, π)

1. Let $u_1, \ldots, u_m$ denote the queries of $V(\phi, \pi)$ (Recall, V is non-adaptive)
2. Sample unifrom k-size subset $J \subseteq [m]$
3. For each $j \in J$: let $a_j = \zeta(u_j)$
4. For each $j \notin J$: uniformly sample 2^γ candidate answers $\{a_{j,t}\}$

Short SNARGs to Low-query SNARGs

Given SNARG $(P,\ V)$, we modify to $\tilde{V}$ as follows (P is unchanged):

$\tilde{V}$

Oracle: $\zeta: \{0,1\}^* \to \{0,1\}^\lambda$

Input: (ϕ, π)

1. Let $u_1, \ldots, u_m$ denote the queries of $V(\phi, \pi)$ (Recall, V is non-adaptive)
2. Sample unifrom k-size subset $J \subseteq [m]$
3. For each $j \in J$: let $a_j = \zeta(u_j)$
4. For each $j \notin J$: uniformly sample 2^γ candidate answers $\{a_{j,t}\}$
5. Accept if any combination of these answers makes $V(\phi, \pi)$ accept

Short SNARGs to Low-query SNARGs

Given SNARG $(P,\ V)$, we modify to $\widetilde{V}$ as follows (P is unchanged):

$\widetilde{V}$

Oracle: $\zeta: \{0,1\}^* \to \{0,1\}^\lambda$

Input: (ϕ, π)

1. Let $u_1, \ldots, u_m$ denote the queries of $V(\phi, \pi)$ (Recall, V is non-adaptive)
2. Sample unifrom k-size subset $J \subseteq [m]$
3. For each $j \in J$: let $a_j = \zeta(u_j)$
4. For each $j \notin J$: uniformly sample 2^γ candidate answers $\{a_{j,t}\}$
5. Accept if any combination of these answers makes $V(\phi, \pi)$ accept

- $\gamma \approx \log \mathrm{t}$ and $k \approx |\pi|/\gamma$ (hence, $|\pi| < \log(t/\epsilon) \cdot \log t \to q_{\widetilde{V}} < \log(t/\epsilon)$)

Short SNARGs to Low-query SNARGs

Given SNARG (P, V), we modify to $\tilde{V}$ as follows (P is unchanged):

$\tilde{V}$

Oracle: $\zeta: \{0,1\}^* \to \{0,1\}^\lambda$

Input: (ϕ, π)

1. Let $u_1, \ldots, u_m$ denote the queries of $V(\phi, \pi)$ (Recall, V is non-adaptive)
2. Sample unifrom k-size subset $J \subseteq [m]$
3. For each $j \in J$: let $a_j = \zeta(u_j)$
4. For each $j \notin J$: uniformly sample 2^γ candidate answers $\{a_{j,t}\}$
5. Accept if any combination of these answers makes $V(\phi, \pi)$ accept

- $\gamma \approx \log t$ and $k \approx |\pi|/\gamma$ (hence, $|\pi| < \log(t/\epsilon) \cdot \log t \to q_{\tilde{V}} < \log(t/\epsilon)$)
- (P, V) has (t, ϵ)-salted-soundness $\to$ $(P, \tilde{V})$ has (t, ϵ)-soundness

Short SNARGs to Low-query SNARGs

Given SNARG $(P,\ V)$, we modify to $\tilde{V}$ as follows (P is unchanged):

$\tilde{V}$

Oracle: $\zeta : \{0,1\}^* \to \{0,1\}^\lambda$

Input: (ϕ, π)

1. Let $u_1, \ldots, u_m$ denote the queries of $V(\phi, \pi)$ (Recall, V is non-adaptive)
2. Sample unifrom k-size subset $J \subseteq [m]$
3. For each $j \in J$: let $a_j = \zeta(u_j)$
4. For each $j \notin J$: uniformly sample 2^γ candidate answers $\{a_{j,t}\}$
5. Accept if any combination of these answers makes $V(\phi, \pi)$ accept

- $\gamma \approx \log \mathrm{t}$ and $k \approx |\pi|/\gamma$ (hence, $|\pi| < \log(t/\epsilon) \cdot \log t \to q_{\tilde{V}} < \log(t/\epsilon)$)
- $(P,\ V)$ has $(\mathrm{t},\ \epsilon)$-salted-soundness $\to$ $(P, \tilde{V})$ has $(\mathrm{t},\ \epsilon)$-soundness
- $(P, \tilde{V})$ has completeness $\left(\gamma \cdot q_V \cdot \binom{q_V}{k}\right)^{-1}$

Motivating examples

Motivating examples

Consider the two following argument systems with $\zeta\colon \{0,1\}^* \to \{0,1\}^\lambda$:

Motivating examples

Consider the two following argument systems with $\zeta : \{0,1\}^* \to \{0,1\}^\lambda$:

$$y = \zeta(q_1)$$

π

Motivating examples

Consider the two following argument systems with $\zeta: \{0,1\}^* \to \{0,1\}^\lambda$:

$$y = \zeta(q_1)$$

π

Local information: $\pi = \zeta(0)$ and V verifies that

$\tilde{V}$ accepts if $j \in J$ (j is the index of 0)

Short π cannot contain many such oracle answers

Motivating examples

Consider the two following argument systems with $\zeta: \{0,1\}^* \rightarrow \{0,1\}^\lambda$:

$y = \zeta(q_1)$

π

Local information: $\pi = \zeta(0)$ and V verifies that

$\tilde{V}$ accepts if $j \in J$ (j is the index of 0)

Short π cannot contain many such oracle answers

$y = \zeta(q_1) \oplus \ldots \oplus \zeta(q_k)$

π

Motivating examples

Consider the two following argument systems with $\zeta: \{0,1\}^* \to \{0,1\}^\lambda$:

$$y = \zeta(q_1)$$

π

Local information: $\pi = \zeta(0)$ and V verifies that

$\tilde{V}$ accepts if $j \in J$ (j is the index of 0)

Short π cannot contain many such oracle answers

$$y = \zeta(q_1) \oplus \ldots \oplus \zeta(q_k)$$

π

Global information: $\pi = \zeta(0) \oplus \zeta(1) \ldots \oplus \zeta(k)$

$\tilde{V}$ samples $\approx 2^\gamma$ options for each $\zeta(q_i)$

If $\gamma > \lambda/k$, then $\tilde{V}$ accepts whp.

Motivating examples

Consider the two following argument systems with $\zeta : \{0,1\}^* \to \{0,1\}^\lambda$:

$y = \zeta(q_1)$

π

Local information: $\pi = \zeta(0)$ and V verifies that

$\tilde{V}$ accepts if $j \in J$ (j is the index of 0)

Short π cannot contain many such oracle answers

$y = \zeta(q_1) \oplus \ldots \oplus \zeta(q_k)$

π

Global information: $\pi = \zeta(0) \oplus \zeta(1) \ldots \oplus \zeta(k)$

$\tilde{V}$ samples $\approx 2^\gamma$ options for each $\zeta(q_i)$

If $\gamma > \lambda / k$, then $\tilde{V}$ accepts whp.

Completeness

Completeness

- The lemma shows V must make $\approx |\pi|/\gamma$ queries, and the rest can be completed by uniform sampling with some probability

Completeness

- The lemma shows V must make $\approx |\pi|/\gamma$ queries, and the rest can be completed by uniform sampling with some probability
- The probability V guesses correctly the important queries is small, yet nontrivial as $|\pi|$ is small

Completeness

- The lemma shows V must make $\approx |\pi|/\gamma$ queries, and the rest can be completed by uniform sampling with some probability
- The probability V guesses correctly the important queries is small, yet nontrivial as $|\pi|$ is small
- This yields completeness slightly larger than the soundness error, ϵ

Hitting High-Entropy Events Lemma

Hitting High-Entropy Events Lemma

Lemma [Hitting High Entropy Events, Informal]**:**

Let $X = X_1, \ldots, X_n$ be variables over $\left(\{0,1\}^{\lambda}\right)^n$, with $H(X) \geq \lambda \cdot n - \ell$

Hitting High-Entropy Events Lemma

Lemma [Hitting High Entropy Events, Informal]**:**

Let $X = X_1, \ldots, X_n$ be variables over $\left(\{0,1\}^{\lambda}\right)^n$, with $H(X) \geq \lambda \cdot n - \ell$

Then, $\mathrm{x} \leftarrow X$ consist of $O(\ell/\gamma)$ binding coordinates, when the rest can be completed using unifom sampling of size 2^{γ}

Hitting High-Entropy Events Lemma

Lemma [Hitting High Entropy Events, Informal]**:**

Let $X = X_1, \ldots, X_n$ be variables over $\left(\{0,1\}^{\lambda}\right)^n$, with $H(X) \geq \lambda \cdot n - \ell$

Then, $\mathrm{x} \leftarrow X$ consist of $O(\ell/\gamma)$ binding coordinates, when the rest can be completed using unifom sampling of size 2^{γ}

- We first show that for $\mathrm{x} \leftarrow X$, exists $B \subseteq [n]$ such that for

Hitting High-Entropy Events Lemma

Lemma [Hitting High Entropy Events, Informal]**:**
Let $X = X_1, \ldots, X_n$ be variables over $\left(\{0,1\}^{\lambda}\right)^n$, with $H(X) \geq \lambda \cdot n - \ell$
Then, $\mathrm{x} \leftarrow X$ consist of $O(\ell/\gamma)$ binding coordinates, when the rest can be completed using unifom sampling of size 2^{γ}

- We first show that for $\mathrm{x} \leftarrow X$, exists $B \subseteq [n]$ such that for

$$\mathrm{X}' = (X_{[n]\setminus B} \mid X_B = x_B) \text{ and all } I \subseteq \left[n - |B|\right],\ \ \mathrm{H}\left(\mathrm{X}'_{\mathrm{I}}\right) \geq (\lambda - \gamma) \cdot |I|$$

Hitting High-Entropy Events Lemma

Lemma [Hitting High Entropy Events, Informal]**:**
Let $X = X_1, \ldots, X_n$ be variables over $\left(\{0,1\}^{\lambda}\right)^n$, with $H(X) \geq \lambda \cdot n - \ell$
Then, $\mathrm{x} \leftarrow X$ consist of $O(\ell/\gamma)$ binding coordinates, when the rest can be completed using unifom sampling of size 2^{γ}

- We first show that for $\mathrm{x} \leftarrow X$, exists $B \subseteq [n]$ such that for

$\mathrm{X}' = (X_{[n]\setminus B} \mid X_B = x_B)$ and all $I \subseteq \left[n - |B|\right]$, $\mathrm{H}\left(\mathrm{X}'_{\mathrm{I}}\right) \geq (\lambda - \gamma) \cdot |I|$

- Then we show that for such B, sampling $S \leftarrow \left(\{0,1\}^{\gamma}\right)^n$ intersects the support of X' with high probability

Hitting High-Entropy Events Lemma

Lemma [Hitting High Entropy Events, Informal]**:**

Let $X = X_1, \ldots, X_n$ be variables over $\left(\{0,1\}^{\lambda}\right)^n$, with $H(X) \geq \lambda \cdot n - \ell$

Then, $\mathrm{x} \leftarrow X$ consist of $O(\ell/\gamma)$ binding coordinates, when the rest can be completed using unifom sampling of size 2^{γ}

- We first show that for $\mathrm{x} \leftarrow X$, exists $B \subseteq [n]$ such that for

$$\mathrm{X}' = (X_{[n]\setminus B} \mid X_B = x_B) \text{ and all } I \subseteq \left[n - |B|\right],\ \ \mathrm{H}\left(\mathrm{X}'_{\mathrm{I}}\right) \geq (\lambda - \gamma) \cdot |I|$$

- Then we show that for such B, sampling $S \leftarrow \left(\{0,1\}^{\gamma}\right)^n$ intersects the support of X' with high probability
- We conclude by showing that the expected size of B is $O(\ell/\gamma)$

Soundness

Soundness

Given malicous P' that fools $\widetilde{V}$, we construct P that wins the salted soundness game:

Soundness

Given malicous P' that fools $\tilde{V}$, we construct P that wins the salted soundness game:

1. P simulates P' to obtain a proof π

Soundness

Given malicous P' that fools $\widetilde{V}$, we construct P that wins the salted soundness game:

1. P simulates P' to obtain a proof π
2. Then, P emulates $\widetilde{V}(\pi)$

Soundness

Given malicous P' that fools $\widetilde{V}$, we construct P that wins the salted soundness game:

1. P simulates P' to obtain a proof π
2. Then, P emulates $\widetilde{V}(\pi)$
   - $\widetilde{V}$'s queries are emulated by queires in the game

Soundness

Given malicous P' that fools $\widetilde{V}$, we construct P that wins the salted soundness game:

1. P simulates P' to obtain a proof π
2. Then, P emulates $\widetilde{V}(\pi)$
   - $\widetilde{V}$'s queries are emulated by queires in the game
3. P chooses the answers that made $\widetilde{V}$ accept

Soundness

Given malicous P' that fools $\widetilde{V}$, we construct P that wins the salted soundness game:

1. P simulates P' to obtain a proof π
2. Then, P emulates $\widetilde{V}(\pi)$
   - $\widetilde{V}$'s queries are emulated by queires in the game
3. P chooses the answers that made $\widetilde{V}$ accept

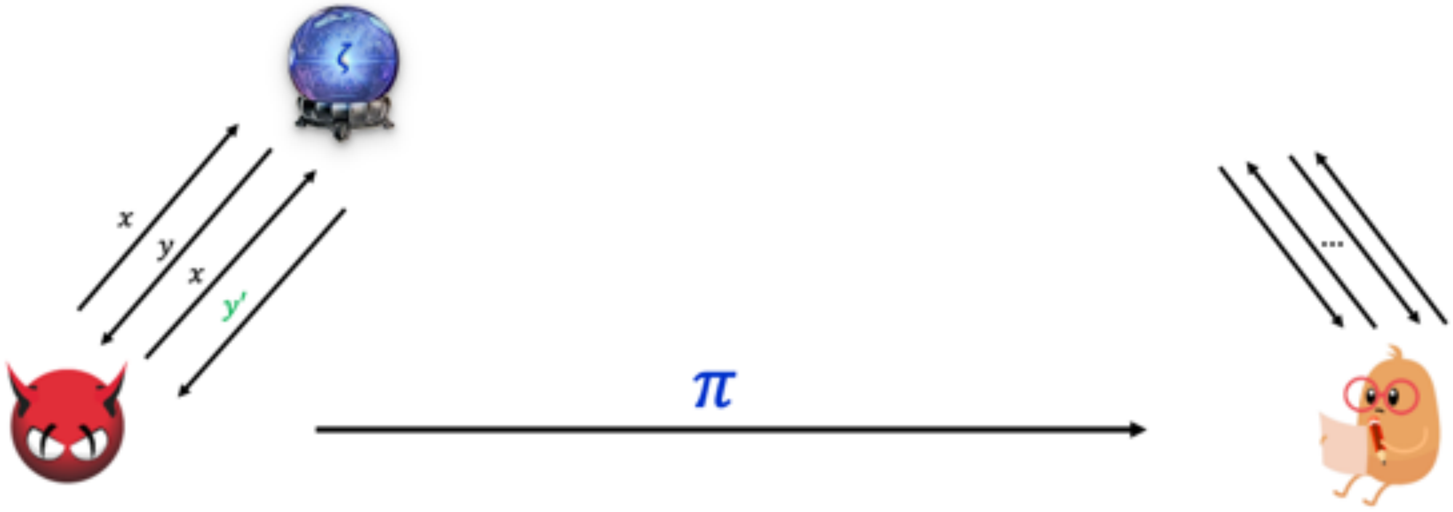
- Notice the similarity of the salted soundness game to the definition of $\widetilde{V}$

Soundness

Given malicous P' that fools $\tilde{V}$, we construct P that wins the salted soundness game:

1. P simulates P' to obtain a proof π
2. Then, P emulates $\tilde{V}(\pi)$
   - $\tilde{V}$'s queries are emulated by queires in the game
3. P chooses the answers that made $\tilde{V}$ accept

- Notice the similarity of the salted soundness game to the definition of $\tilde{V}$



Conclusions and open problems

Conclusions and open problems

SNARGs in the ROM:

Conclusions and open problems

SNARGs in the ROM:

- Have optimal size between $O\left(\log\frac{t}{\epsilon} \cdot \log t \cdot \log n\right)$ and $O\left(\log\frac{t}{\epsilon}\right)$

Conclusions and open problems

SNARGs in the ROM:

- Have optimal size between $O\left(\log\frac{t}{\epsilon}\cdot\log t\cdot\log n\right)$ and $O\left(\log\frac{t}{\epsilon}\right)$
- Have size $\Omega\left(\log\frac{t}{\epsilon}\cdot\log t/\log q_P\right)$ for "natural" constructions

Conclusions and open problems

SNARGs in the ROM:

- Have optimal size between $O\left(\log\frac{t}{\epsilon}\cdot\log t\cdot\log n\right)$ and $O\left(\log\frac{t}{\epsilon}\right)$
- Have size $\Omega\left(\log\frac{t}{\epsilon}\cdot\log t/\log q_P\right)$ for "natural" constructions

Conclusions and open problems

SNARGs in the ROM:

- Have optimal size between $O\left(\log\frac{t}{\epsilon}\cdot\log t\cdot\log n\right)$ and $O\left(\log\frac{t}{\epsilon}\right)$
- Have size $\Omega\left(\log\frac{t}{\epsilon}\cdot\log t/\log q_P\right)$ for “natural” constructions

Open questions:

Conclusions and open problems

SNARGs in the ROM:

- Have optimal size between $O\left(\log\frac{t}{\epsilon}\cdot\log t\cdot\log n\right)$ and $O\left(\log\frac{t}{\epsilon}\right)$
- Have size $\Omega\left(\log\frac{t}{\epsilon}\cdot\log t/\log q_P\right)$ for “natural” constructions

Open questions:

- General lower bound (for adaptive verifier or without salted soundness)

Conclusions and open problems

SNARGs in the ROM:

- Have optimal size between $O\left(\log\frac{t}{\epsilon}\cdot\log t\cdot\log n\right)$ and $O\left(\log\frac{t}{\epsilon}\right)$
- Have size $\Omega\left(\log\frac{t}{\epsilon}\cdot\log t/\log q_P\right)$ for "natural" constructions

Open questions:

- General lower bound (for adaptive verifier or without salted soundness)
- Build an improved SNARG without salted soundness

Conclusions and open problems

SNARGs in the ROM:

- Have optimal size between $O\left(\log\frac{t}{\epsilon}\cdot\log t\cdot\log n\right)$ and $O\left(\log\frac{t}{\epsilon}\right)$
- Have size $\Omega\left(\log\frac{t}{\epsilon}\cdot\log t/\log q_P\right)$ for "natural" constructions

Open questions:

- General lower bound (for adaptive verifier or without salted soundness)
- Build an improved SNARG without salted soundness

Thank You!