Non-Interactive Zero-Knowledge Proofs with Fine-Grained Security

Yuyu Wang[1] and Jiaxin pan[2]

1. University of Electronic Science and Technology of China
2. NTNU - Norwegian University of Science and Technology

Standard cryptography

Honest party

polynomial-time

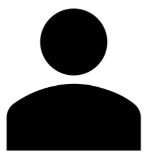
Adversary

polynomial-time

Assumption:
- Basic ones (e.g., one-way function)
- More advanced ones (e.g., factoring, discrete logarithm, DDH, LWE)
- Exotic ones (e.g, generic groups, algebraic groups)

Standard cryptography

Honest party



polynomial-time

Adversary

polynomial-time

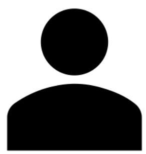
Unproven

Assumption:

- Basic ones (e.g., one-way function)
- More advanced ones (e.g., factoring, discrete logarithm, DDH, LWE)
- Exotic ones (e.g, generic groups, algebraic groups)

Fine-grained cryptography

Honest party



Adversary

An honest party uses less resources than the adversary

Fine-grained cryptography

Honest party

An honest party uses less resources than the adversary

Adversary

The resources of an adversary can be a-prior bounded

Fine-grained cryptography

Honest party

An honest party uses less resources than the adversary

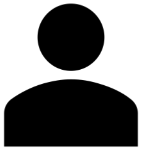
Adversary

The resources of an adversary can be a-prior bounded

- Based only on mild assumption

Fine-grained cryptography

Honest party



An honest party uses less resources than the adversary

Adversary

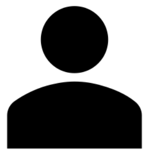
The resources of an adversary can be a-prior

Existing fine-grained primitives:
NIKE [Mer78], OWF [BC20], PKE [DVV16], verifiable computation [CG18], trapdoor one-way function [EWT21], ABE [WPC21]

- Based only on mild assumption

Fine-grained cryptography

Honest party



An honest party uses less resources than the adversary

Adversary

The resources of an adversary can be a-prior

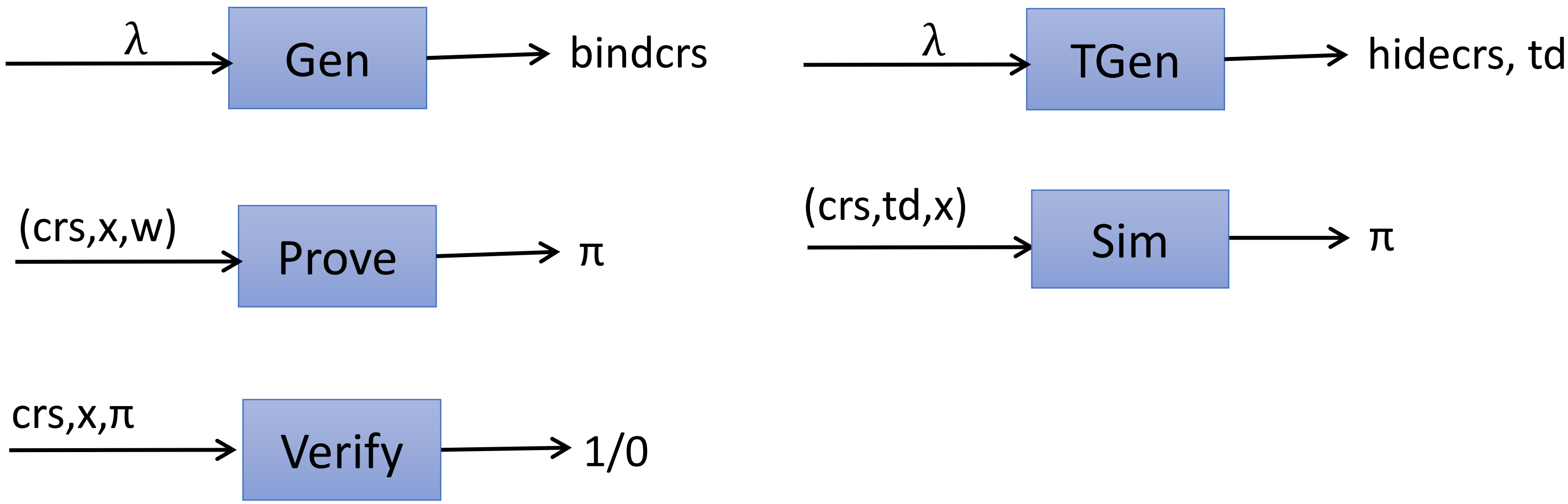
Existing fine-grained primitives:

NIZK?

- Based only on mild assumption

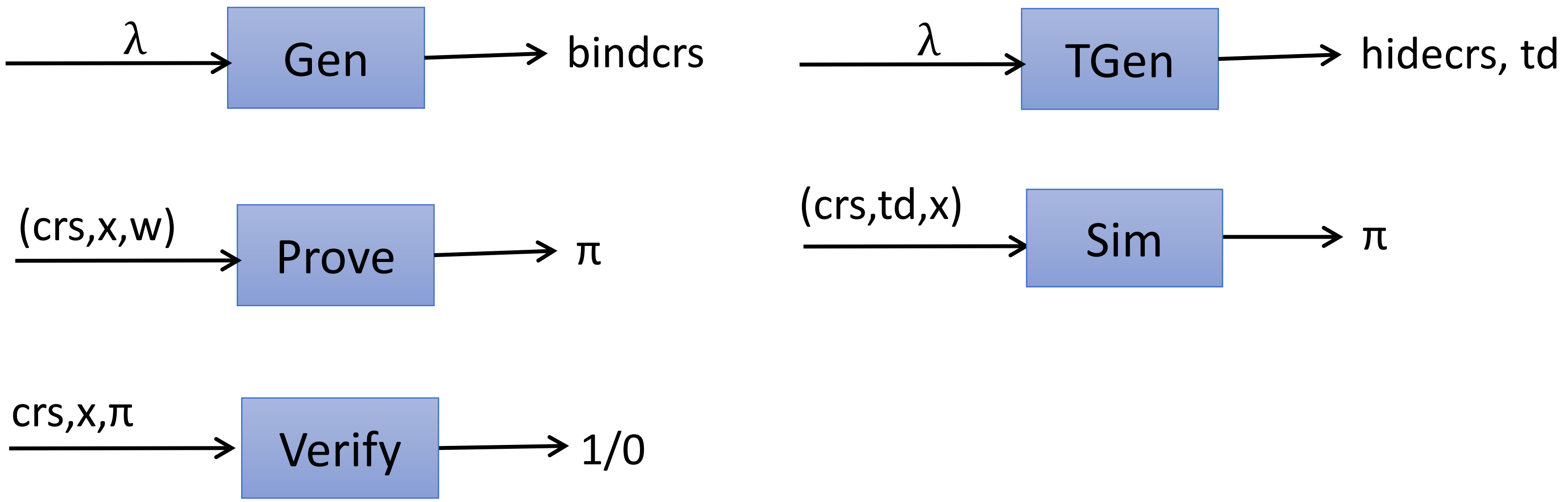
Definition of NIZK

$$x \in L \text{ iff } \exists w \text{ s.t. } R(x, w) = 1$$



Definition of NIZK

$$x \in L \text{ iff } \exists w \text{ s.t. } R(x, w) = 1$$



Completeness: honest proofs must pass the verification.

Perfect soundness: when crs is binding, there exists no valid proof for x if $x \notin L$.

(Composable) zero knowledge: bindcrs and hidecrs are indistinguishable, and when crs is hiding, Sim perfectly simultates honest proofs.

Existing Fine-Grained Proof Systems

- Hash proof system [EWT19]
- QA-NIZK [WPC21]
- NIZK with inefficient prover [BDK20]

Existing Fine-Grained Proof Systems

- Hash proof system [EWT19]
- QA-NIZK [WPC21]
- NIZK with inefficient prover [

Secure against adversaries in NC^1 under the assumption: $\mathsf{NC}^1 \neq \oplus \mathrm{L/poly}$.

Existing Fine-Grained Proof Systems

- Hash proof system [EWT19]
- QA-NIZK [WPC21]
- NIZK with inefficient prover [E

Circuits with logarithmic depth

Secure against adversaries in NC^1 under the assumption: $\mathsf{NC}^1 \neq \oplus \mathsf{L/poly}$.

Existing Fine-Grained Proof Systems

- Hash proof system [EWT19]
- QA-NIZK [WPC21]
- NIZK with inefficient prover [E

Secure against adversaries in NC^1 under the assumption:
$\mathrm{NC}^1 \neq \oplus \mathrm{L/poly}$.

The class of languages with polynomial-sized branching programs.

Existing Fine-Grained Proof Systems

- Hash proof system [EWT19]
- QA-NIZK [WPC21]
- NIZK with inefficient prover [

Secure against adversaries in NC^1 under the assumption: $\mathrm{NC}^1 \neq \oplus \mathrm{L/poly}$.

This assumption is widely believed to hold.

Existing Fine-Grained Proof Systems

- Hash proof system [EWT19]
  - Verifier needs a secret key
- QA-NIZK [WPC21]
- NIZK with inefficient prover [BDK20]

Existing Fine-Grained Proof Systems

- Hash proof system [EWT19]
  - Verifier needs a secret key
- QA-NIZK [WPC21]
  - Only supports linear languages
  - CRSs are dependent on language parameter
- NIZK with inefficient prover [BDK20]

Existing Fine-Grained Proof Systems

- Hash proof system [EWT19]
  - Verifier needs a secret key
- QA-NIZK [WPC21]
  - Only supports linear languages
  - CRSs are dependent on language parameter
- NIZK with inefficient prover [BDK20]
  - Not in the fully fine-grained setting: the prover needs more computation resources than NC^1

Our results

A fully fine-grained NIZK for NC^1-circuit satisfiability (SAT)

- the CRS generator, prover, verifier, simulator run in NC^1
- secure against adversaries in NC^1
- assumption: $\mathrm{NC}^1 \neq \oplus\, \mathrm{L/poly}$.

Our results

A fully fine-grained NIZK for NC^1-circuit satisfiability (SAT)

- the CRS generator, pr ver, verifier, simulator run in NC^1
- secure against adve ies in NC^1
- assumption: NC^1 /poly.

All statements verifiable in NC1

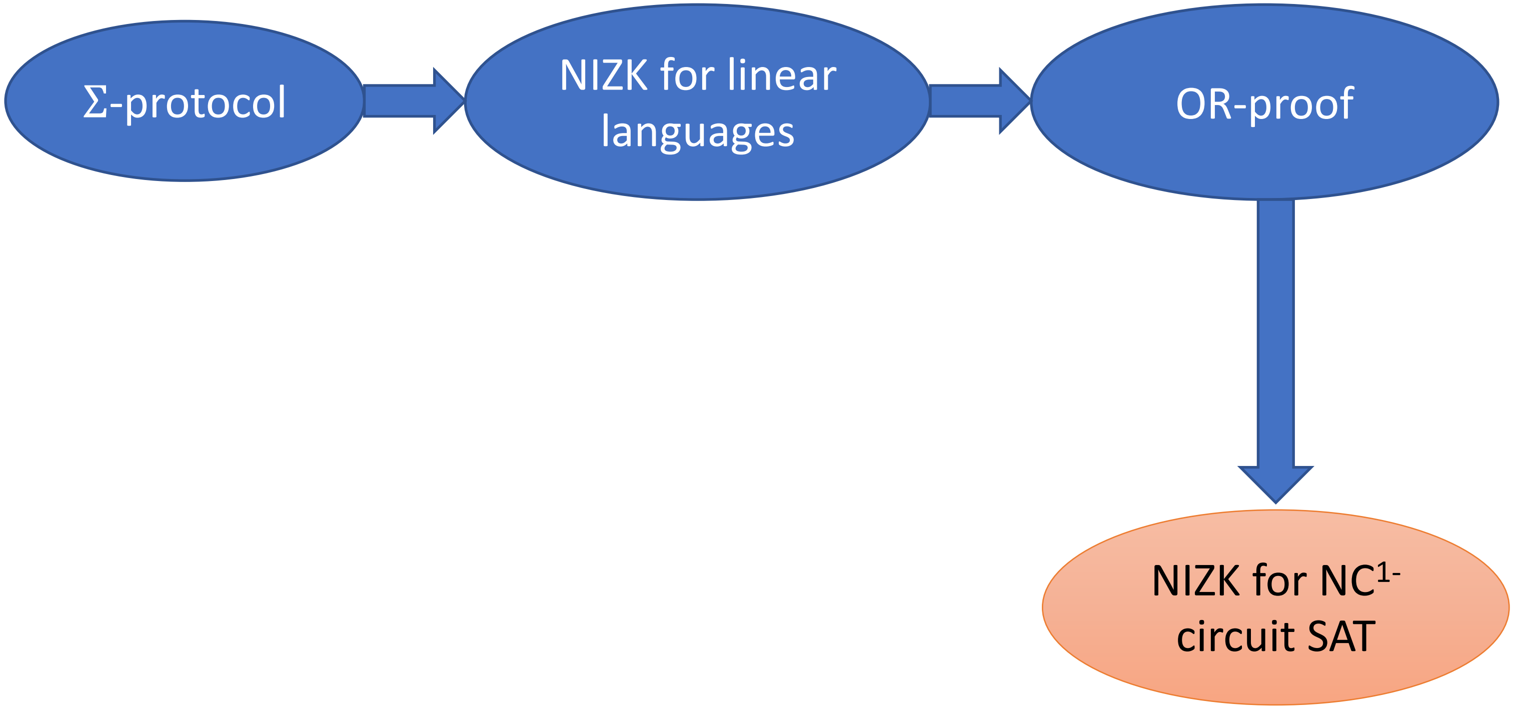
Our results

A fully fine-grained NIZK for NC^1-circuit satisfiability (SAT)

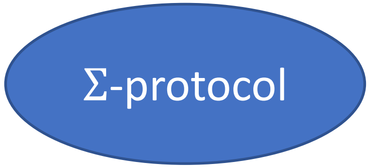
- ❖the CRS generator, prov r, verifier, simulator run in NC^1
- ❖secure against advers in NC^1
- ❖assumption: $NC^1 \neq$ ly.

A statement circuit cannot go beyond NC^1. Otherwise, even the honest prover in NC^1 cannot decide with the witness whether the statement is true or not.

Fine-Grained NIZK



Fine-Grained NIZK

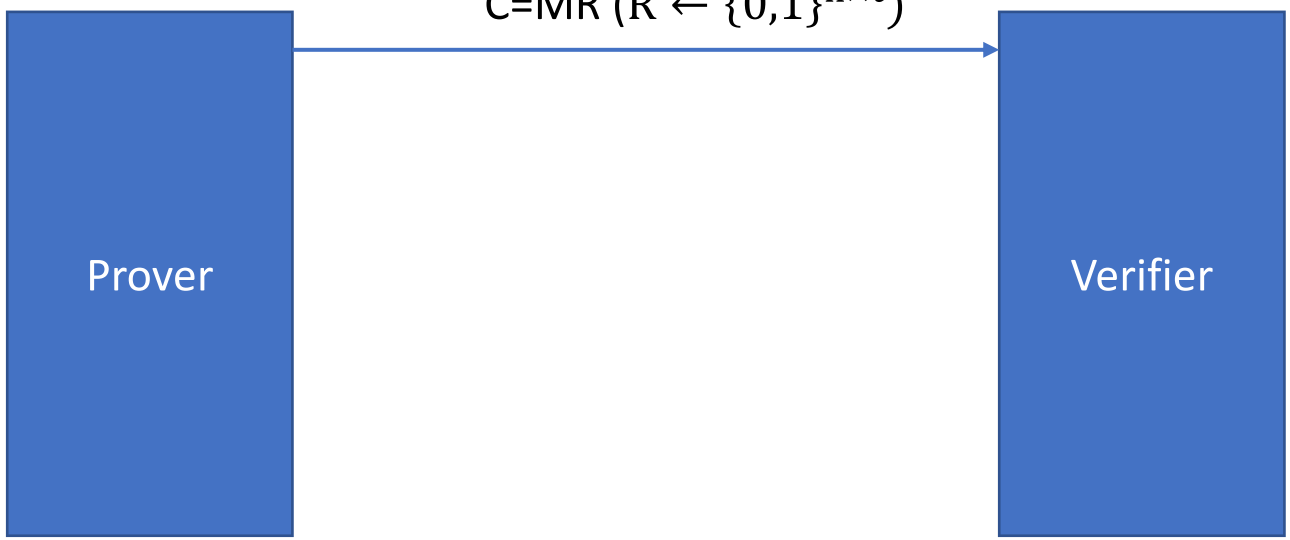


Σ-protocol

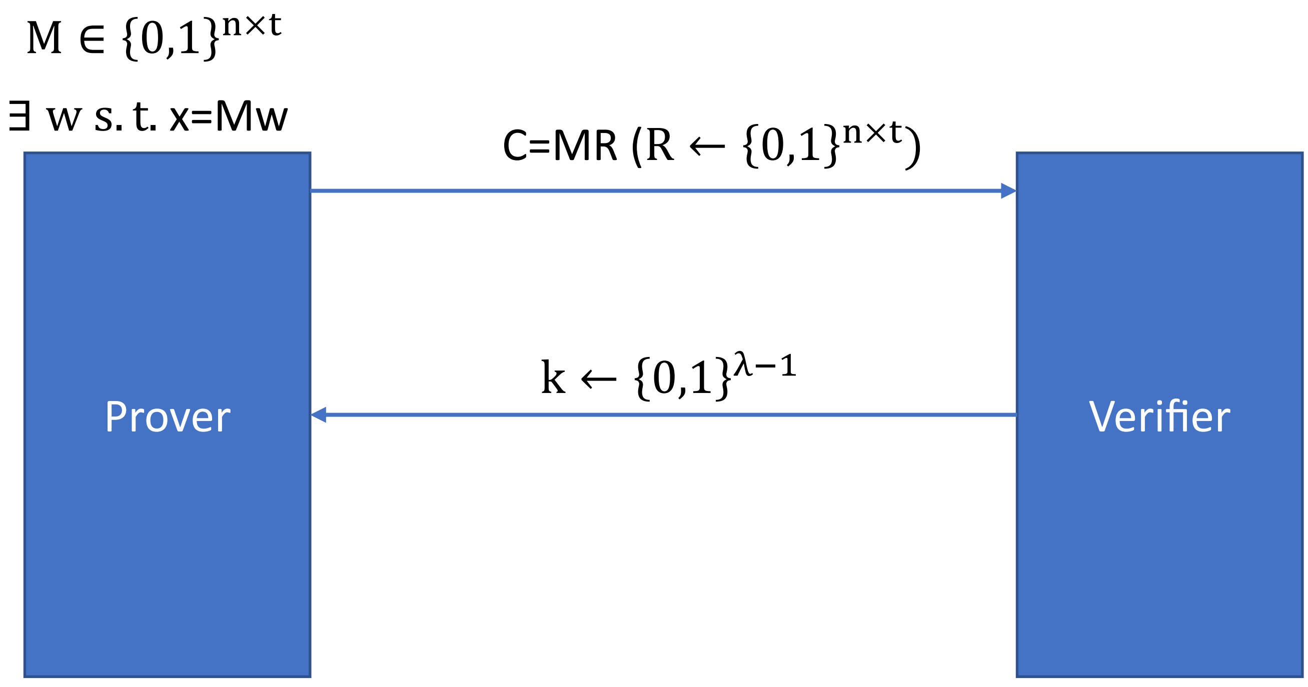
$M \in \{0,1\}^{n \times t}$

$\exists$ w s. t. x=Mw

C=MR $(R \leftarrow \{0,1\}^{n \times t})$

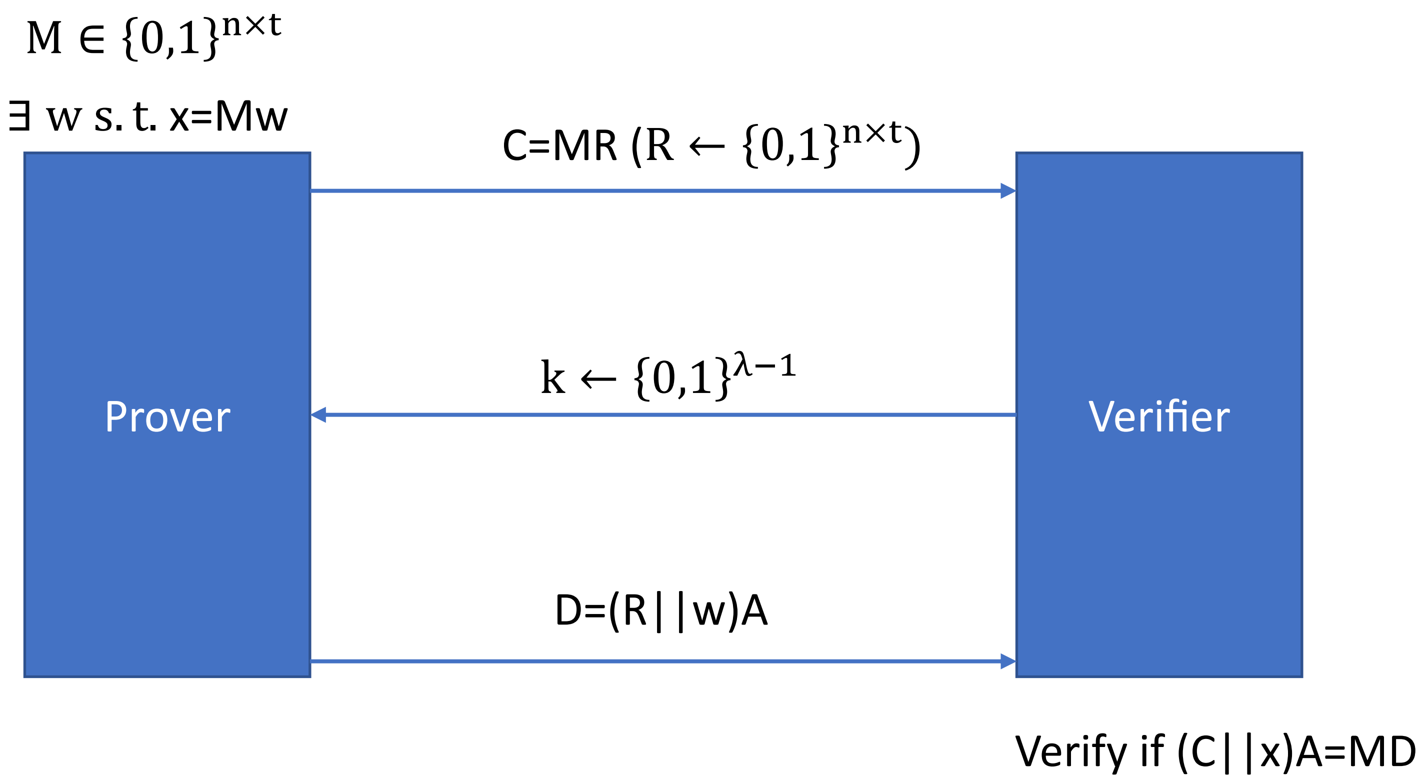


Σ-protocol



$M \in \{0,1\}^{n\times t}$
$\exists$ w s. t. x=Mw
C=MR ($R \leftarrow \{0,1\}^{n\times t}$)
$k \leftarrow \{0,1\}^{\lambda-1}$
Prover
Verifier

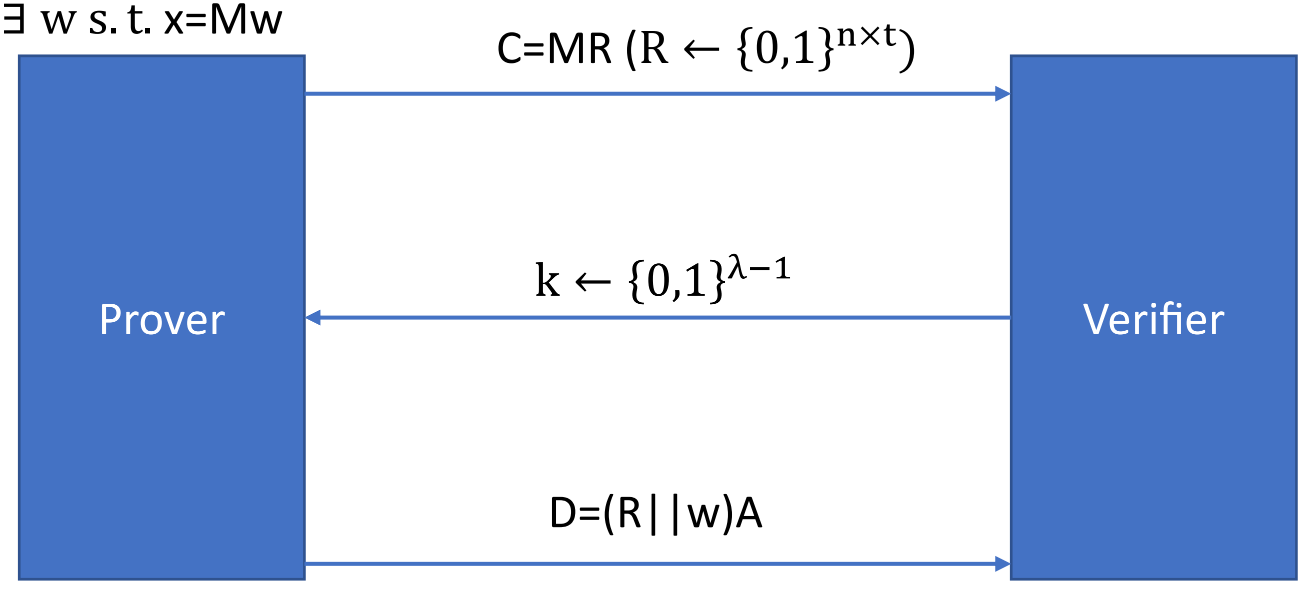
Σ-protocol


$M \in \{0,1\}^{n\times t}$
$\exists$ w s. t. x=Mw
C=MR ($R \leftarrow \{0,1\}^{n\times t}$)
$k \leftarrow \{0,1\}^{\lambda-1}$
D=(R||w)A
Prover
Verifier
Verify if (C||x)A=MD

Σ-protocol

$M \in \{0,1\}^{n \times t}$

$\exists\, w$ s.t. $x = Mw$



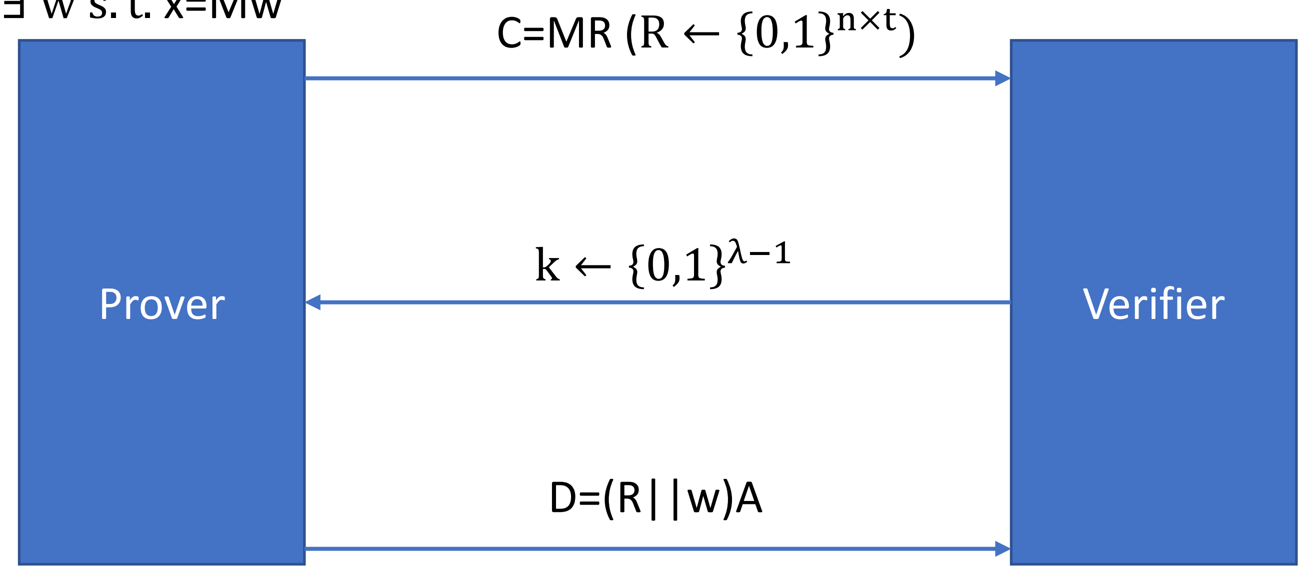
Verify if $(C||x)A = MD$

$A = (S||Sk)^{\top}$

Σ-protocol

$M \in \{0,1\}^{n \times t}$

$\exists\ w\ \text{s.t.}\ x=Mw$



Verify if $(C||x)A=MD$

$S = (0||I)^{\top}$, $A=(S||Sk)^{\top}$

Σ-protocol

$M \in \{0,1\}^{n \times t}$

$\exists$ w s. t. x=Mw

C=MR ($R \leftarrow \{0,1\}^{n \times t}$)

Completeness ✓

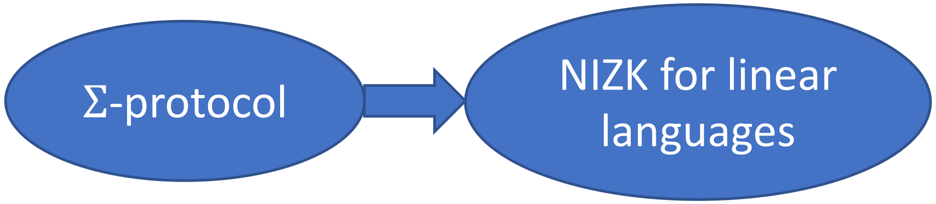
Special soundness ✓

Special honest-verifier zero-knowledge ✓

verify if (C||x)A=MD

$S = (0||I)^{\top}$, A=(S||Sk)$^{\top}$

NIZK for linear languages



NIZK for linear languages

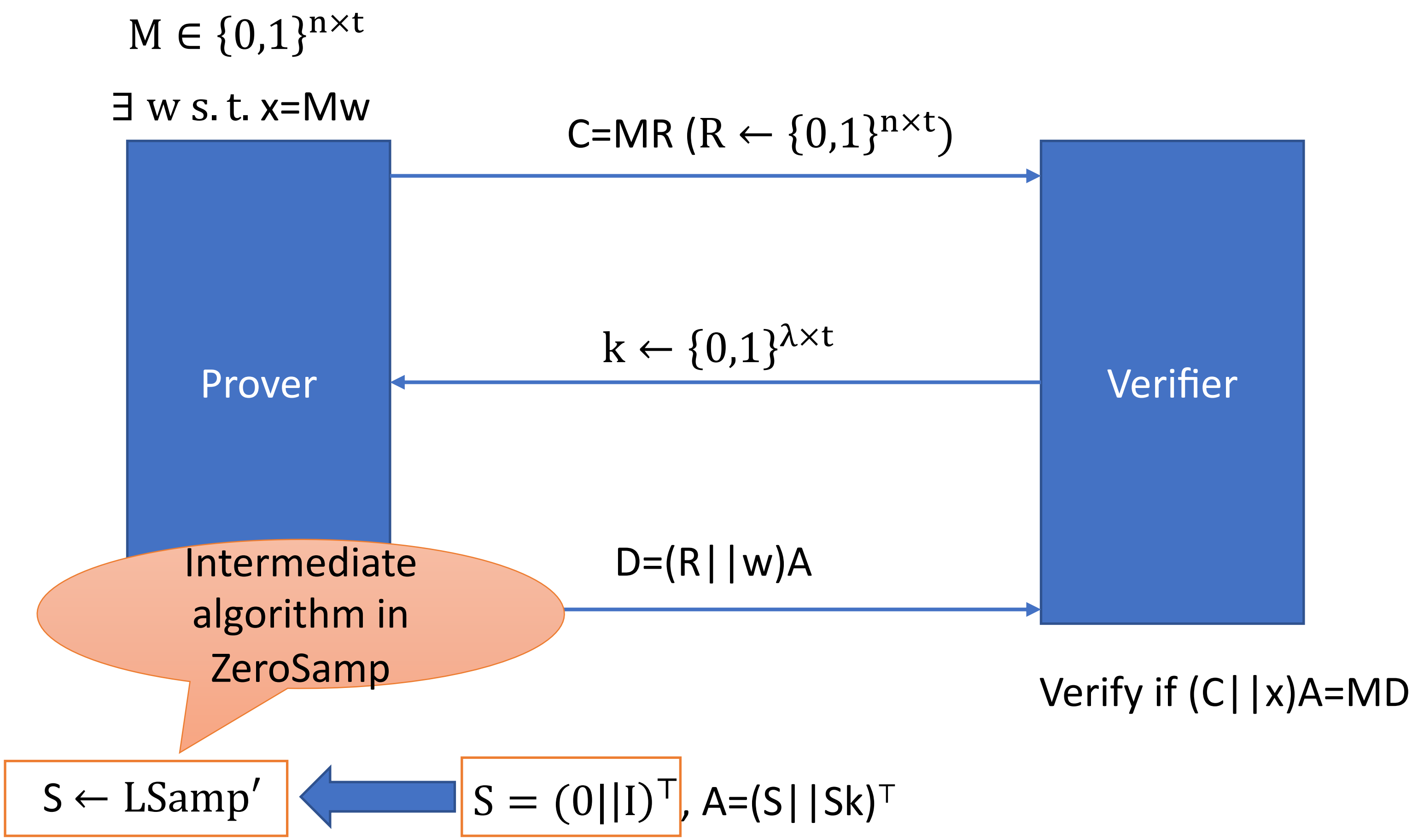
$\xrightarrow{\lambda}$ ZeroSamp $\longrightarrow$ (M_0,s) $\lambda\times\lambda$ matrix (Rank λ-1), s is in the kernel of M_0

$\approx$

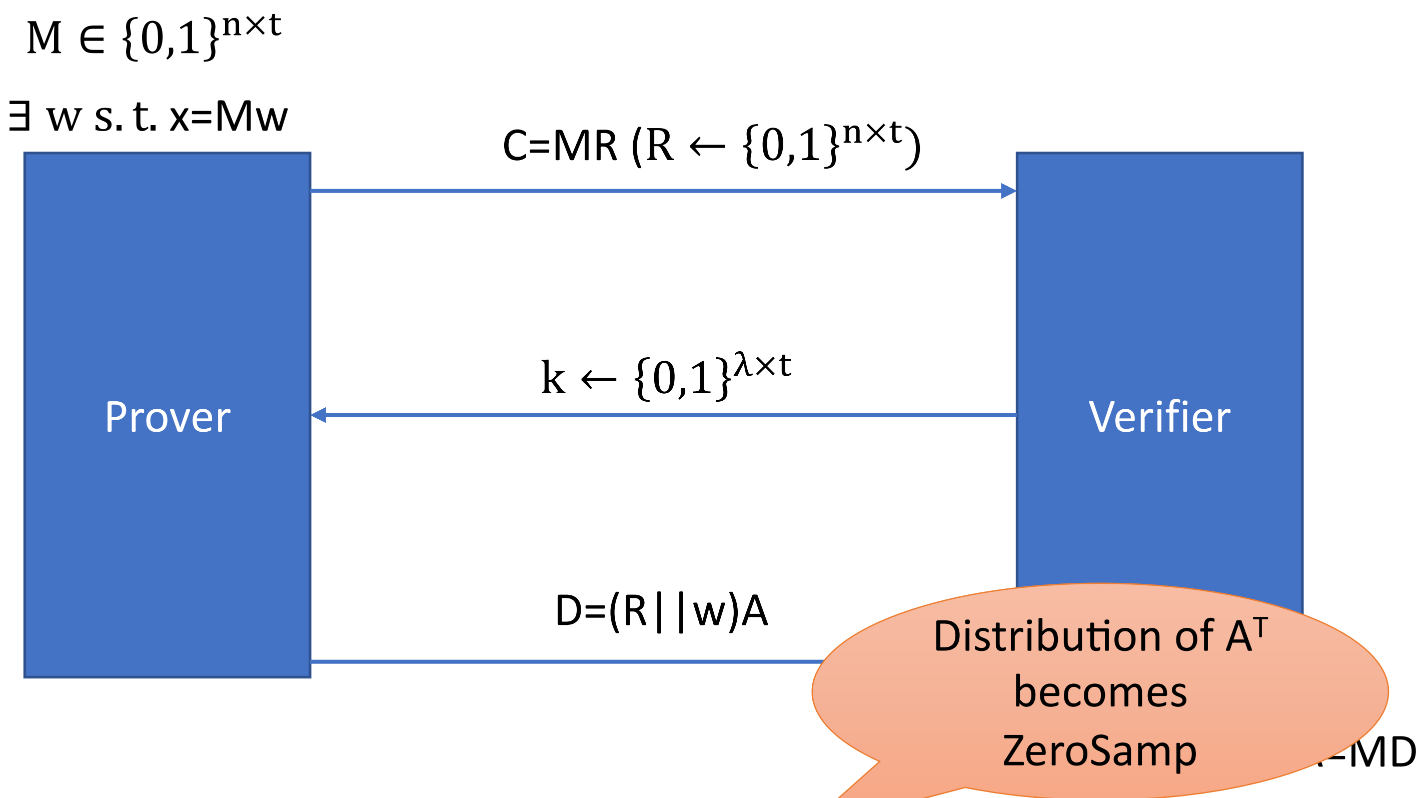
$\xrightarrow{\lambda}$ OneSamp $\longrightarrow$ M_1 $\lambda\times\lambda$ matrix (Rank λ)

Indistinguishable against NC1
adversaries if $\mathrm{NC}^1 \neq \oplus \mathrm{L/poly}$ [DVV16]

Σ-protocol


$M \in \{0,1\}^{n \times t}$
$\exists$ w s. t. x=Mw
C=MR ($R \leftarrow \{0,1\}^{n \times t}$)
$k \leftarrow \{0,1\}^{\lambda \times t}$
Prover
Verifier
D=(R||w)A
Intermediate algorithm in ZeroSamp
Verify if (C||x)A=MD
$S \leftarrow \mathrm{LSamp}'$
$S = (0||I)^{\top}$, A=(S||Sk)$^{\top}$

Σ-protocol



$S \leftarrow \mathrm{LSamp}'$ $\Leftarrow$ $S = (0||I)^T$, $\boxed{A=(S||Sk)^T}$

NIZK for linear languages

td is k

$A = (S||Sk)^T$ $(S \leftarrow \mathrm{LSamp}')$

hideCRS

$M \in \{0,1\}^{n \times t}$

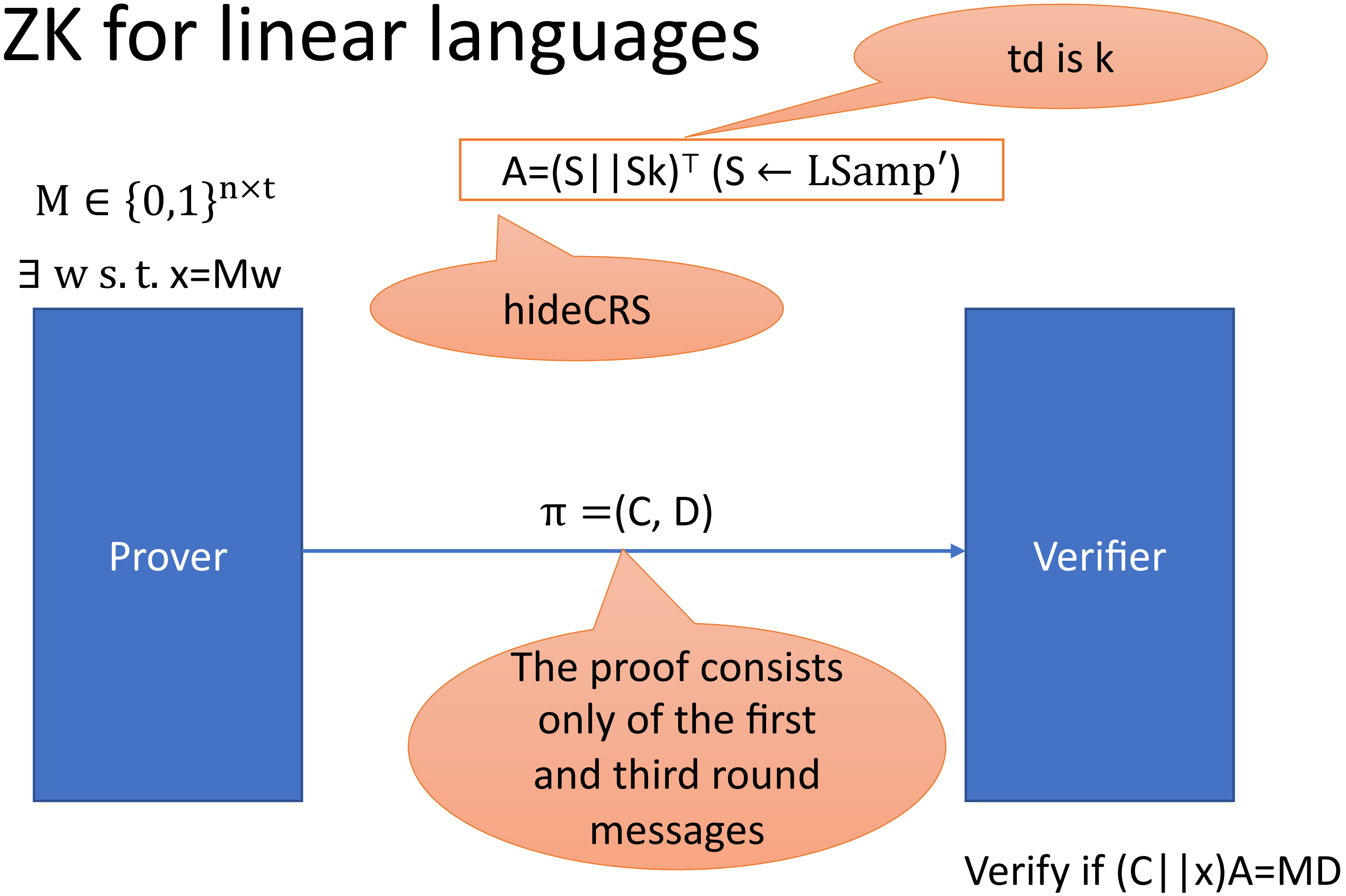
$\exists$ w s. t. x=Mw

Prover

Verifier

Verify if (C||x)A=MD

NIZK for linear languages

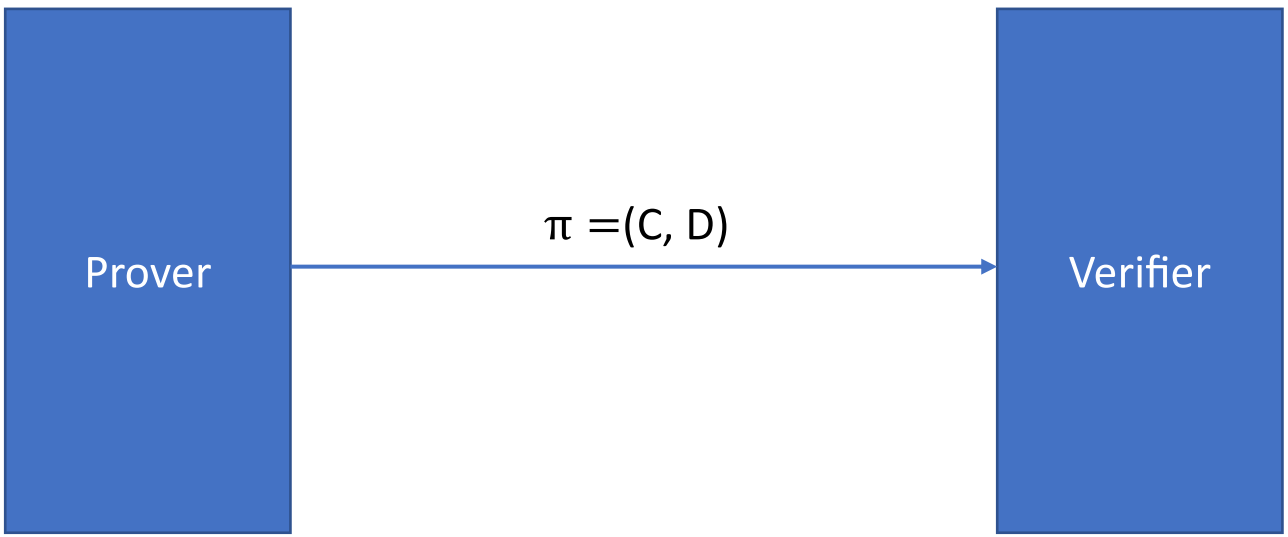


NIZK for linear languages

$A=(S||Sk)^{T}$ ($S \leftarrow \mathrm{LSamp}'$)

$M \in \{0,1\}^{n\times t}$

$\exists$ w s.t. x=Mw



Verify if (C||x)A=MD

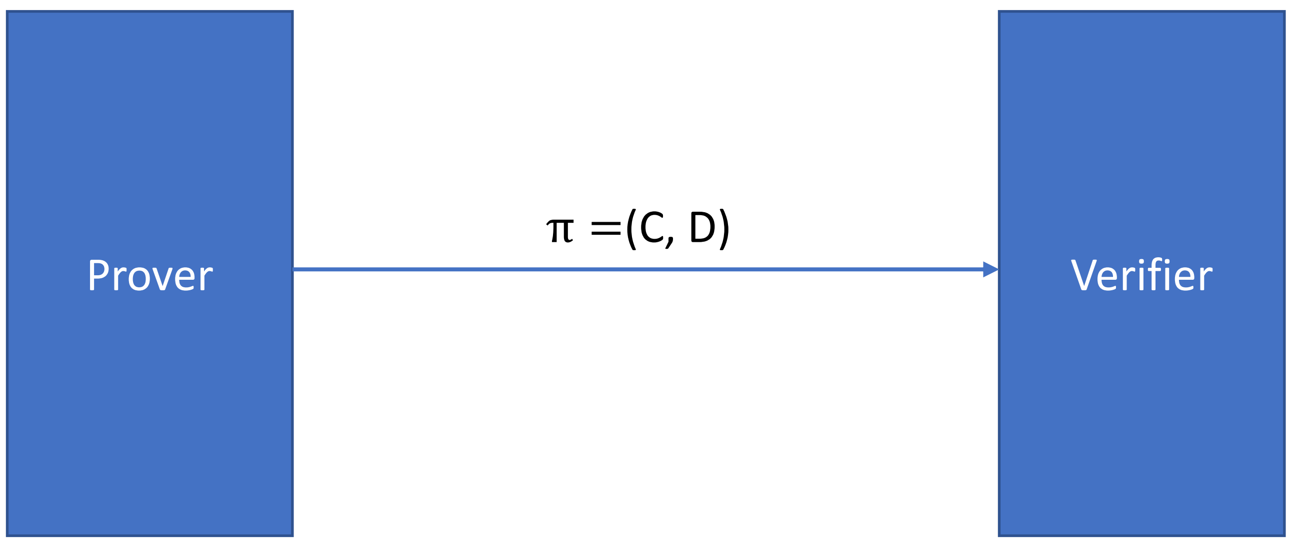
Now the Σ-protocol becomes a NIZK

NIZK for linear languages

$$\boxed{A = (S||Sk)^{\top}\ (S \leftarrow \mathrm{LSamp}')}$$

$M \in \{0,1\}^{n \times t}$

$\exists\ w$ s.t. $x = Mw$



Verify if $(C||x)A = MD$

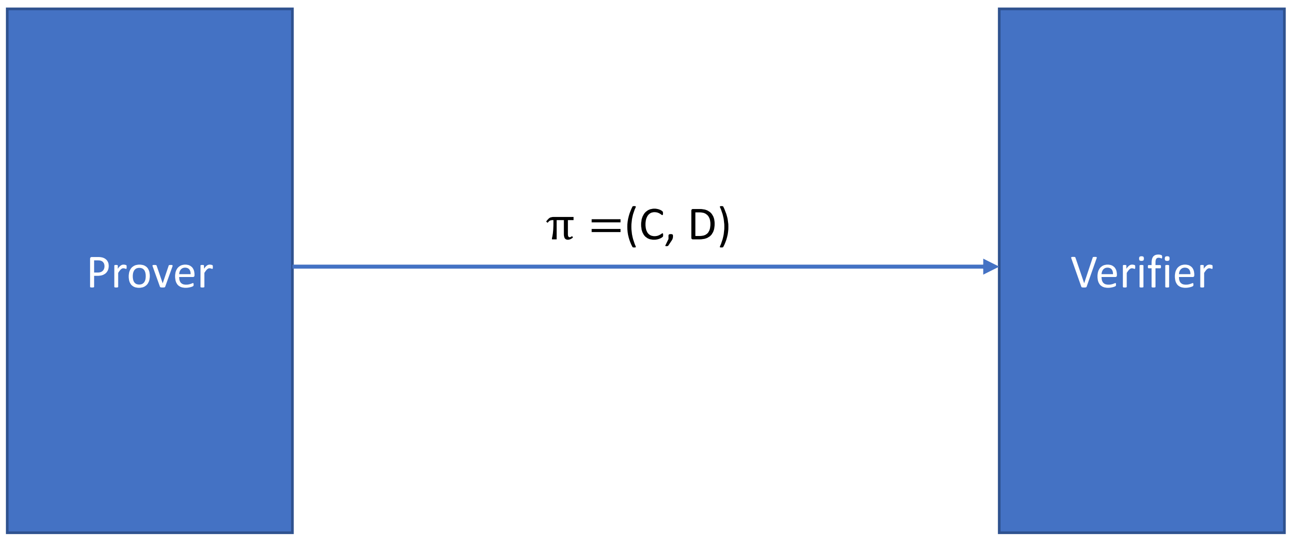
Completeness of NIZK ← Completeness of Σ-protocol
Zero-knowledge of NIZK ← SHVZK of Σ-protocol

NIZK for linear languages

$A=(S||Sk)^{T}$ $(S \leftarrow \mathrm{LSamp}')$

$M \in \{0,1\}^{n \times t}$

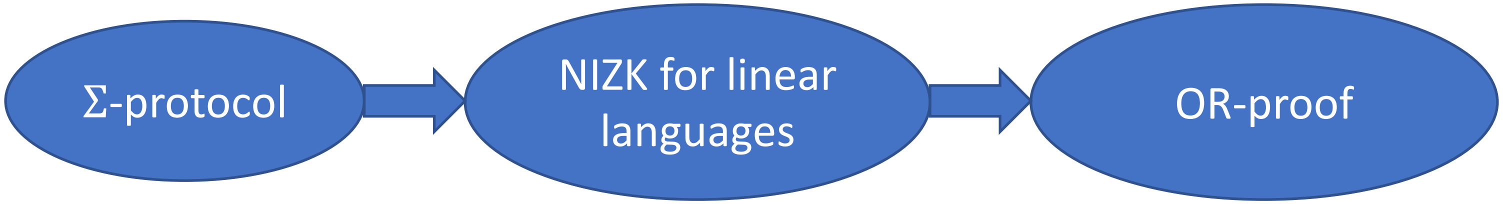
$\exists$ w s. t. x=Mw



Verify if (C||x)A=MD

Soundness of NIZK ← when switching the distribution of A^{T} to OneSamp, the kernel of A becomes empty and no invalid x can pass the verification.

OR-proof



NIZK for OR-languages

$M_0 \in \{0,1\}^{n\times t}$

$M_1 \in \{0,1\}^{n\times t}$

$x_j = M_j w$ for some $j \in \{0,1\}$

$A=(S||t)^T$

Prover

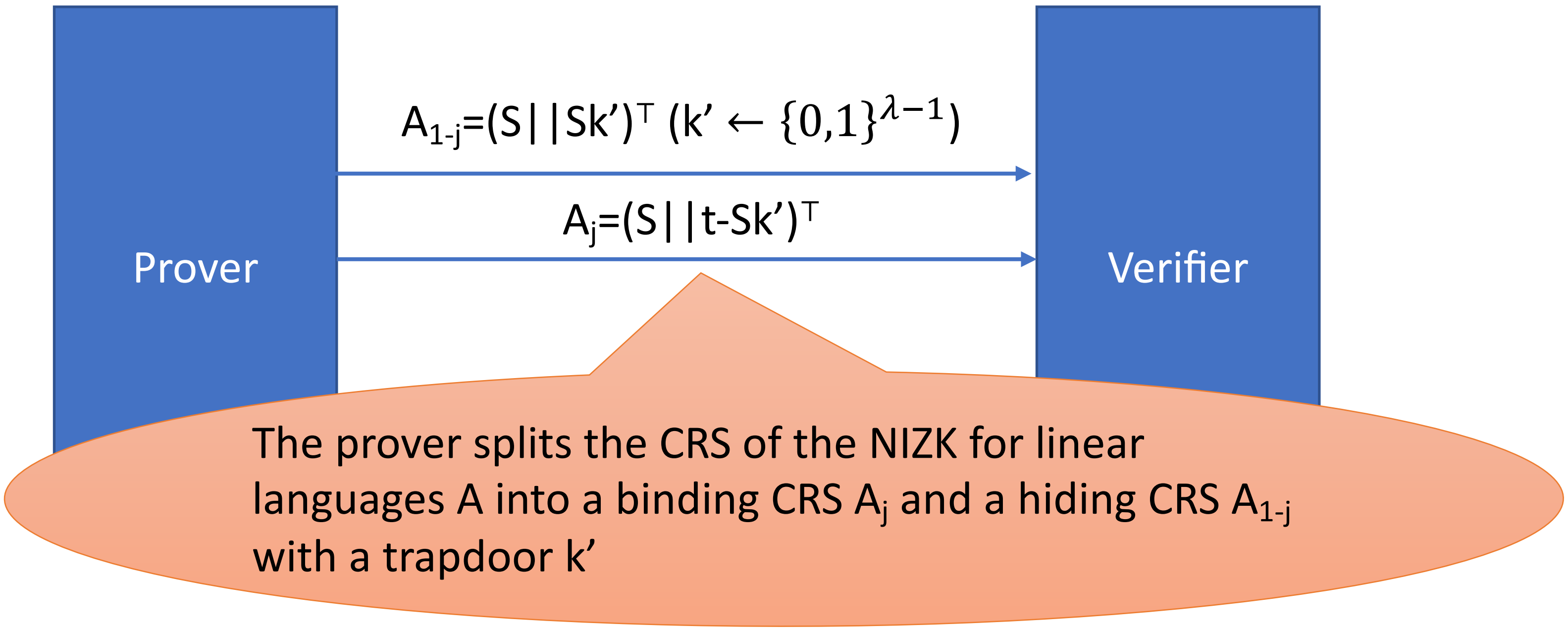
Verifier

NIZK for OR-languages

$M_0 \in \{0,1\}^{n \times t}$
$M_1 \in \{0,1\}^{n \times t}$

$x_j = M_j w$ for some $j \in \{0,1\}$

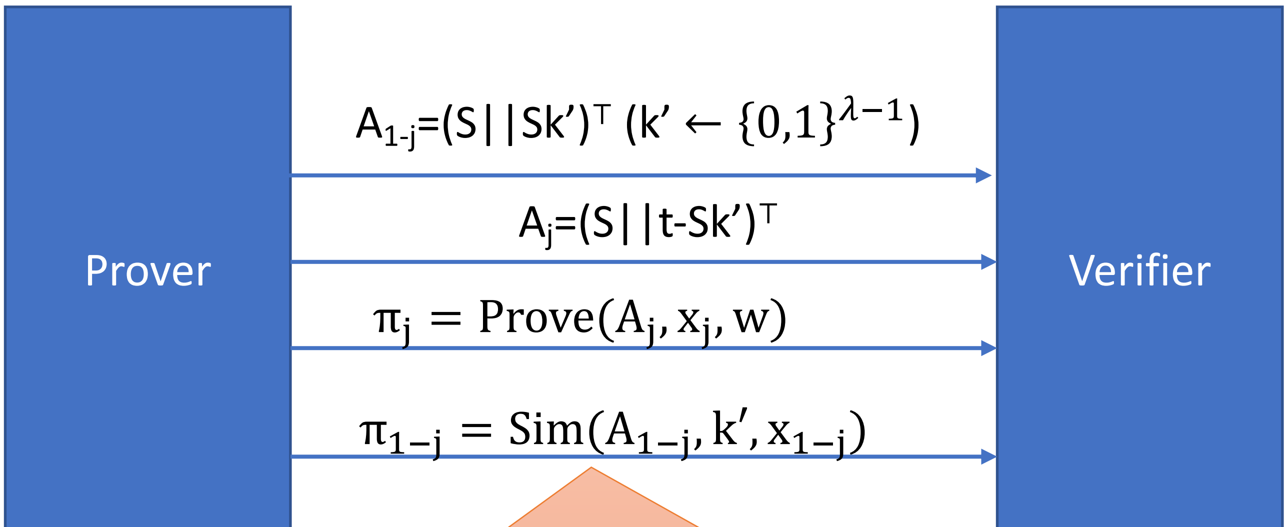
$A = (S||t)^\top$



NIZK for OR-languages

$M_0 \in \{0,1\}^{n\times t}$
$M_1 \in \{0,1\}^{n\times t}$
$x_j = M_j w$ for some $j \in \{0,1\}$

$A = (S||t)^T$



Then it generates proofs for x_j and x_{1-j} with w and k' respectively by making use of the prover and simulator of our NIZK for linear languages.

and π_1 are valid

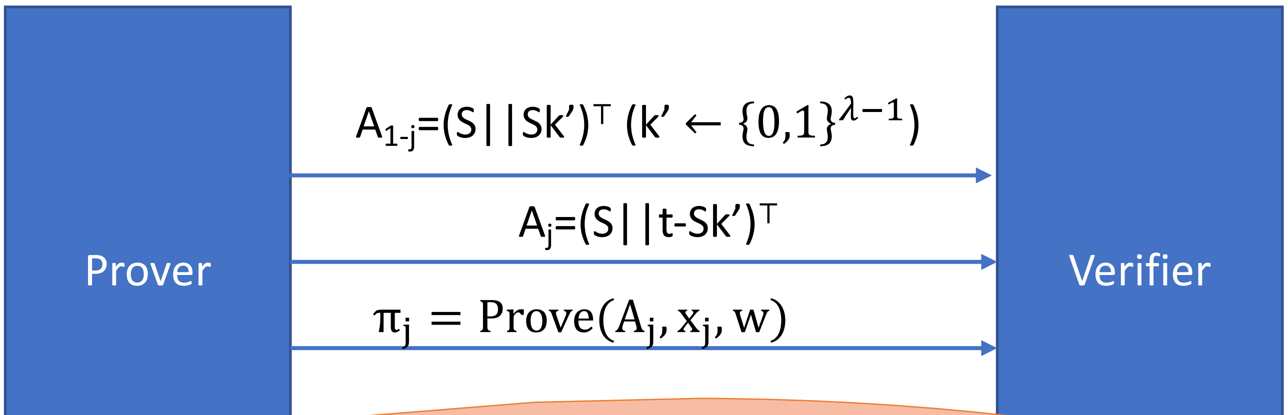
nguages

NIZK for OR-languages

$M_0 \in \{0,1\}^{n\times t}$
$M_1 \in \{0,1\}^{n\times t}$

$x_j = M_j w$ for some $j \in \{0,1\}$

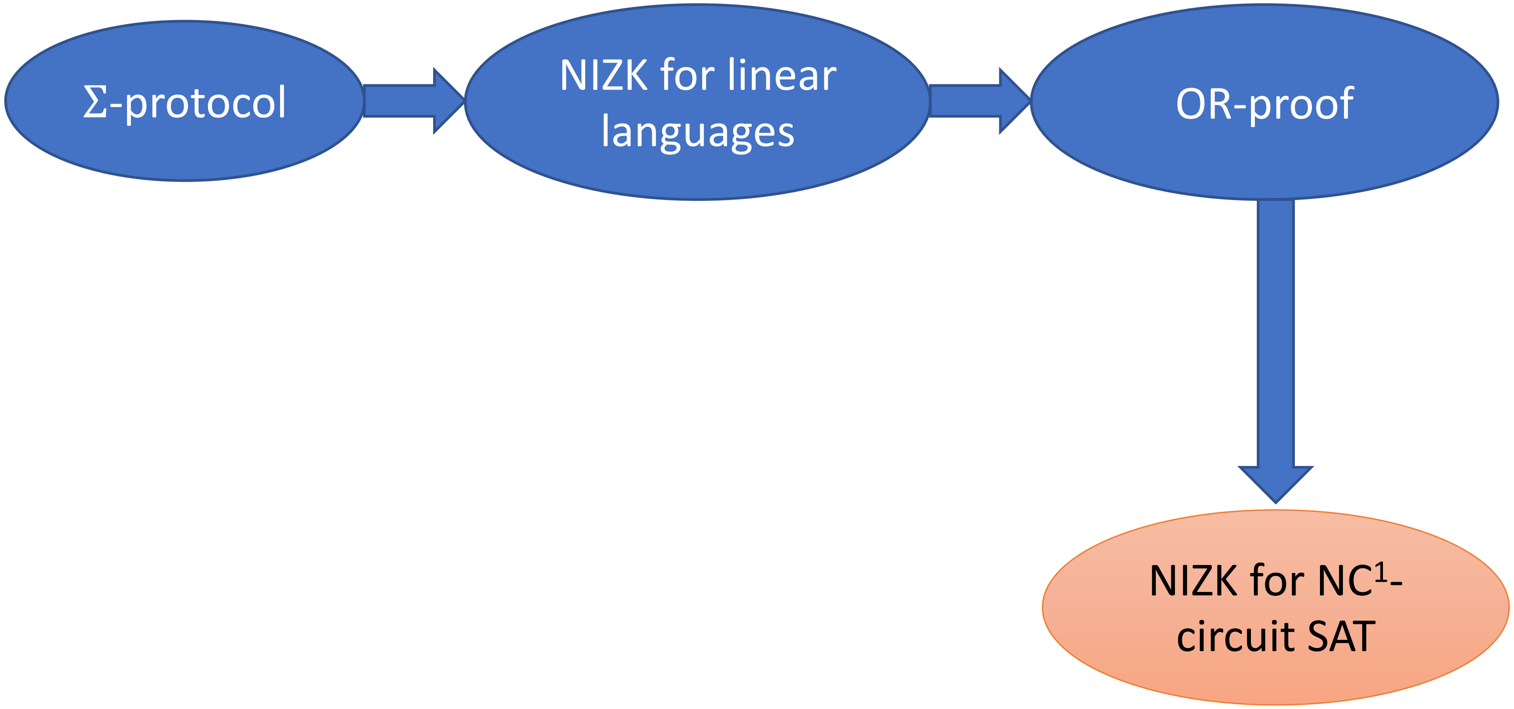
$\boxed{A=(S||t)^T}$



Soundness: when $A^T \leftarrow \mathrm{OneSamp}(\lambda)$,
either A_0 or A_1 must be binding

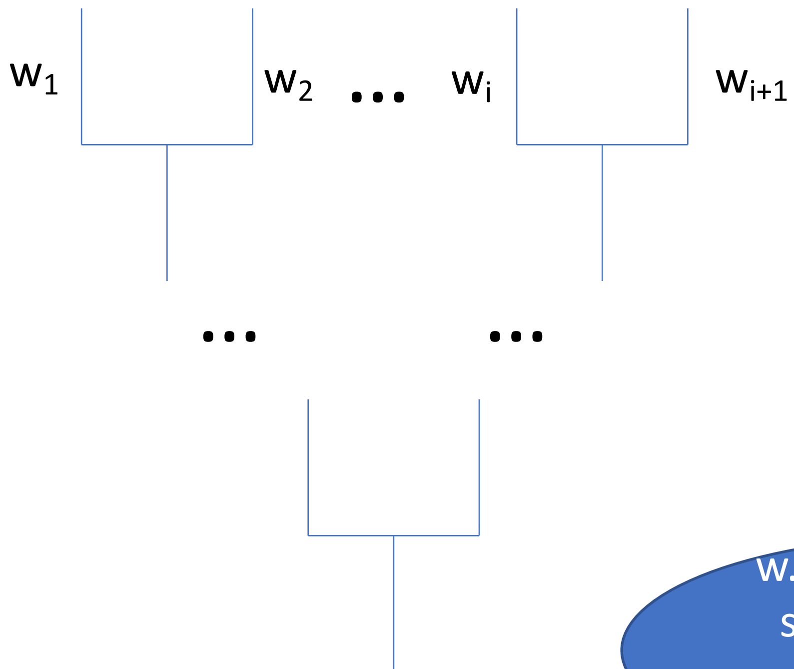
Zero-knowledge: when $(A^T, s) \leftarrow \mathrm{ZeroSamp}(\lambda)$,
both are hiding

NIZK for NC^1 circuits



NIZK for NC^1 circuits

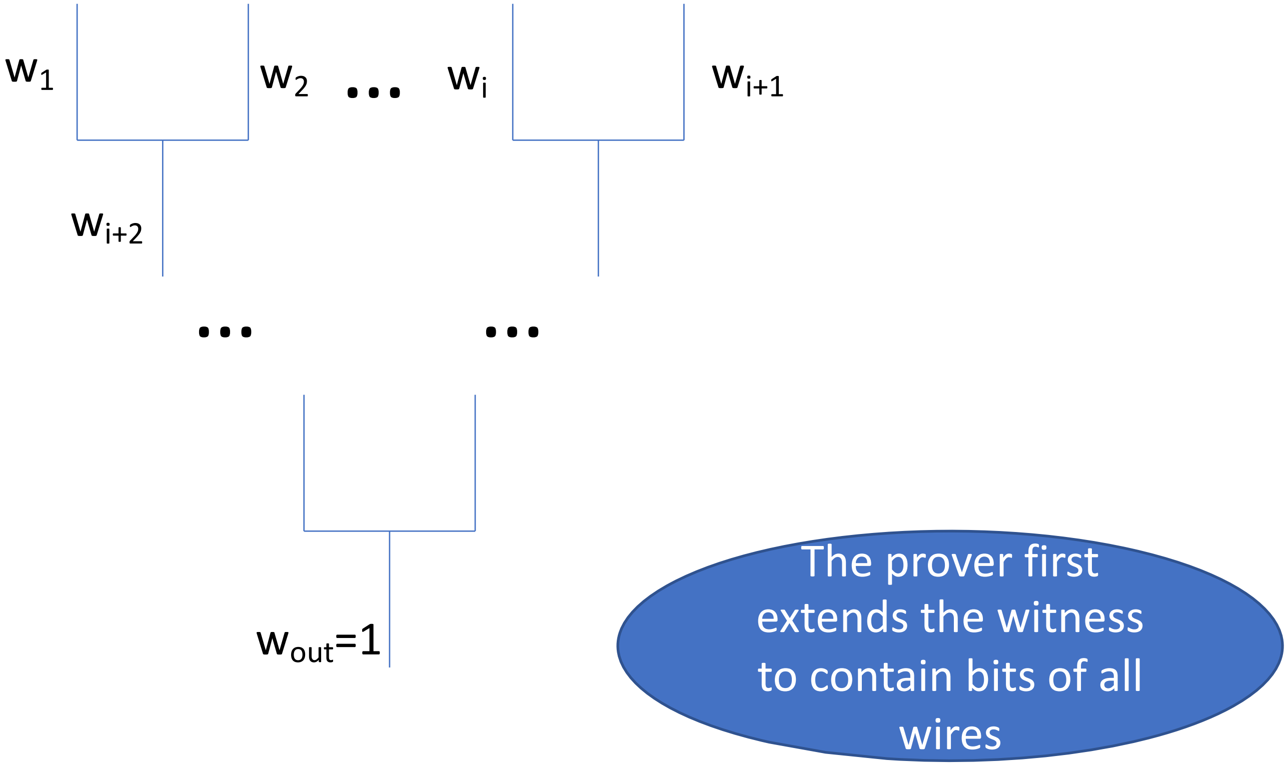
Prover:



w.l.o.g., we consider statement circuits consisting of only NAND gates

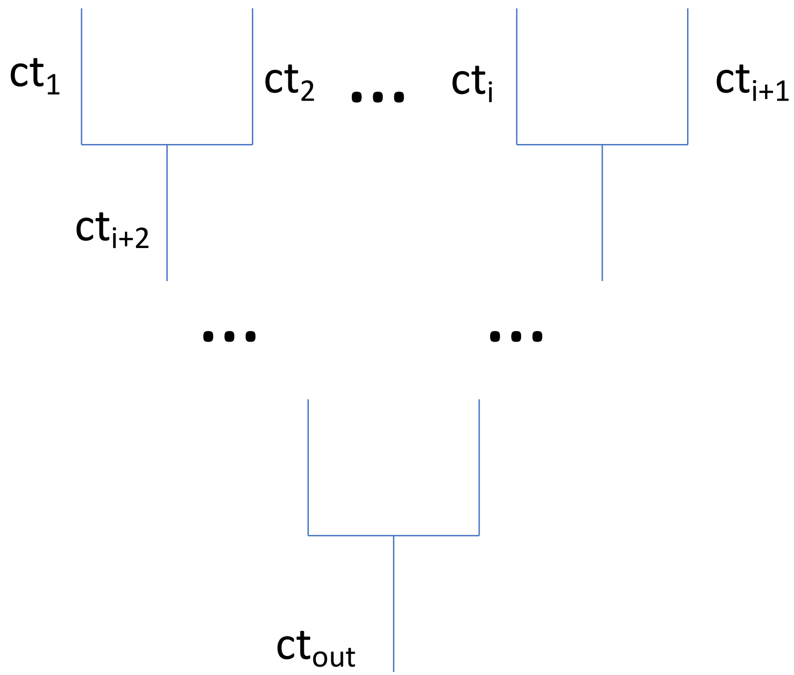
NIZK for NC^1 circuits

Prover:



NIZK for NC^1 circuits

Prover:

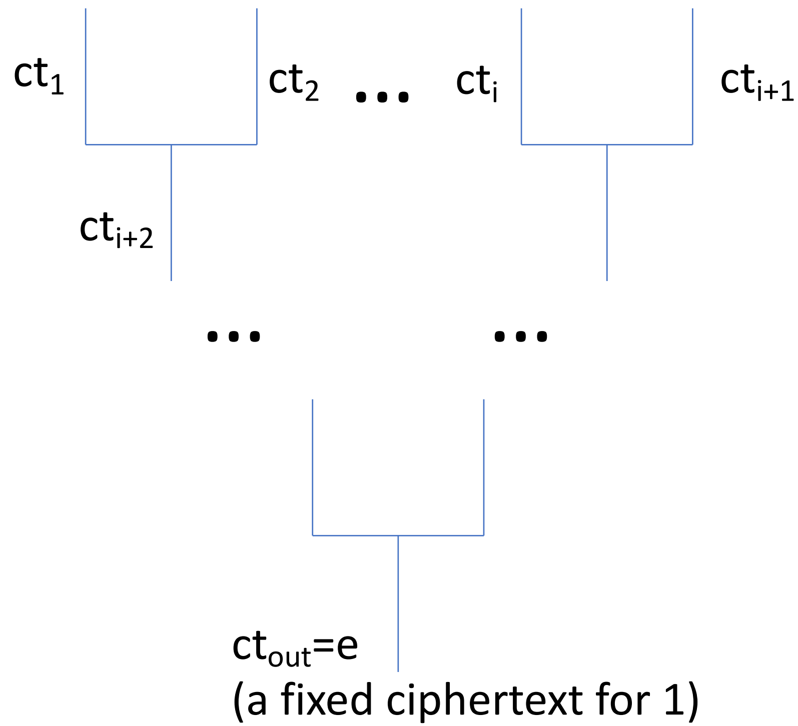


$(A^\top, s) \leftarrow \mathrm{ZeroSamp}(\lambda)$
ct_i=Enc(A,w_i)
w_i=Dec(s,ct_i)

DVV NC_1-fine-grained PKE

NIZK for NC^1 circuits

Prover:



$(A^{\top}, s) \leftarrow \mathrm{ZeroSamp}(\lambda)$

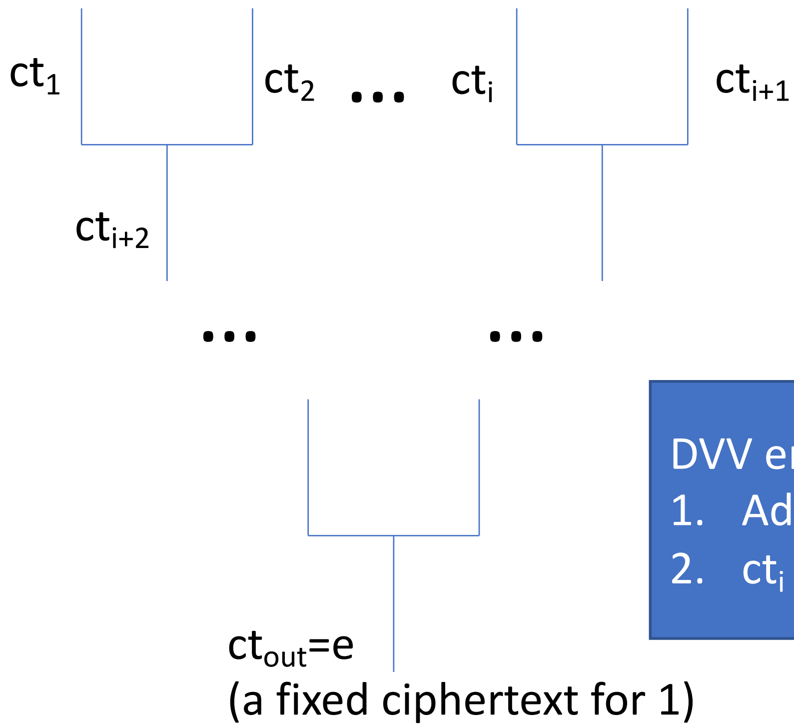
ct_i=Enc(A,w_i)

w_i=Dec(s,ct_i)

DVV NC_1-fine-grained PKE

NIZK for NC^1 circuits

Prover:

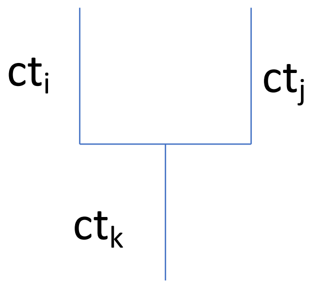


$(A^\top, s) \leftarrow \mathrm{ZeroSamp}(\lambda)$
$ct_i = Enc(A, w_i)$
$w_i = Dec(s, ct_i)$

DVV encryption has two properties:
1. Additive homomorphism
2. $ct_i \in \mathrm{Span}(A)$ iff $w_i = 0$

NIZK for NC^1 circuits

Prover:



NAND gate

The prover proves that the input/output ciphertexts satisfies a relation supported by our OR-proof.

$e + ct_i + ct_k \in \mathrm{Span}(A)$
And
$e + ct_j \in \mathrm{Span}(A)$

or

$e + ct_k \in \mathrm{Span}(A)$
And
$ct_j \in \mathrm{Span}(A)$

NIZK for NC^1 circuits

Soundness:

$e + ct_i + ct_k \in \mathrm{Span}(A)$
And
$e + ct_j \in \mathrm{Span}(A)$

or

$e + ct_k$ …
… $\mathrm{pan}(A)$

$1 + w_i + w_k = 0$
And
$1 + w_j = 0$

or

$1 + w_k = 0$
And
$w_j = 0$

If the ciphertexts satisfy the relation, the corresponding plaintexts must be a valid input/output tuple of the NAND gate

NIZK for NC^1 circuits

Soundness:

$\mathrm{e} + \mathrm{ct}_\mathrm{i} + \mathrm{ct}_\mathrm{k} \in \mathrm{Span}(\mathrm{A})$
And
$\mathrm{e} + \mathrm{ct}_\mathrm{j} \in \mathrm{Span}(\mathrm{A})$

or

$\mathrm{e} + \mathrm{ct}_\mathrm{k} \in \mathrm{Span}(\mathrm{A})$
And
$\mathrm{ct}_\mathrm{j} \in \mathrm{Span}(\mathrm{A})$

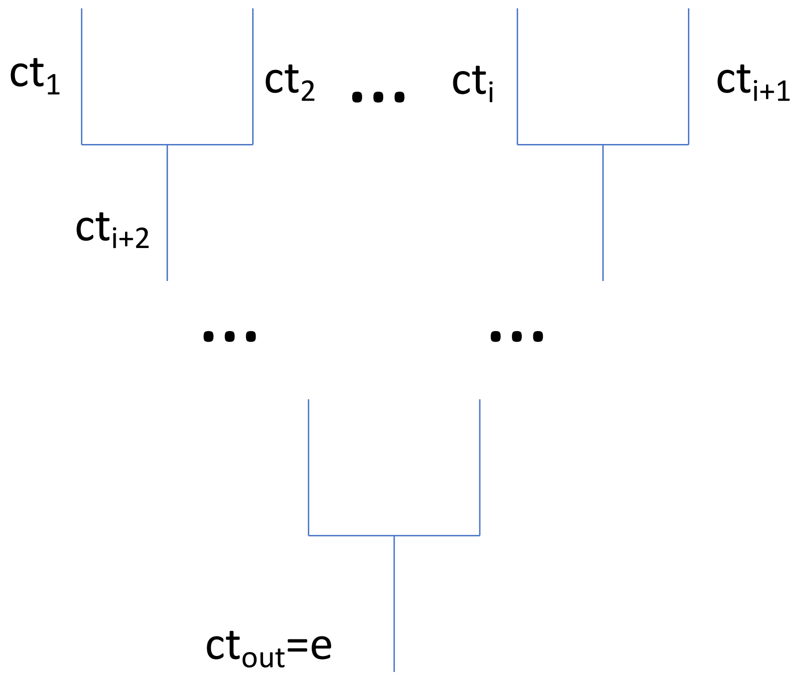
$\Downarrow$

$1 + \mathrm{w}_\mathrm{i} + \mathrm{w}_\mathrm{k} = 0$
And
$1 + \mathrm{w}_\mathrm{j} = 0$

or

Then soundness: we can extract a valid witness from a valid proof by decrypting the ciphertexts.

NIZK for NC^1 circuits

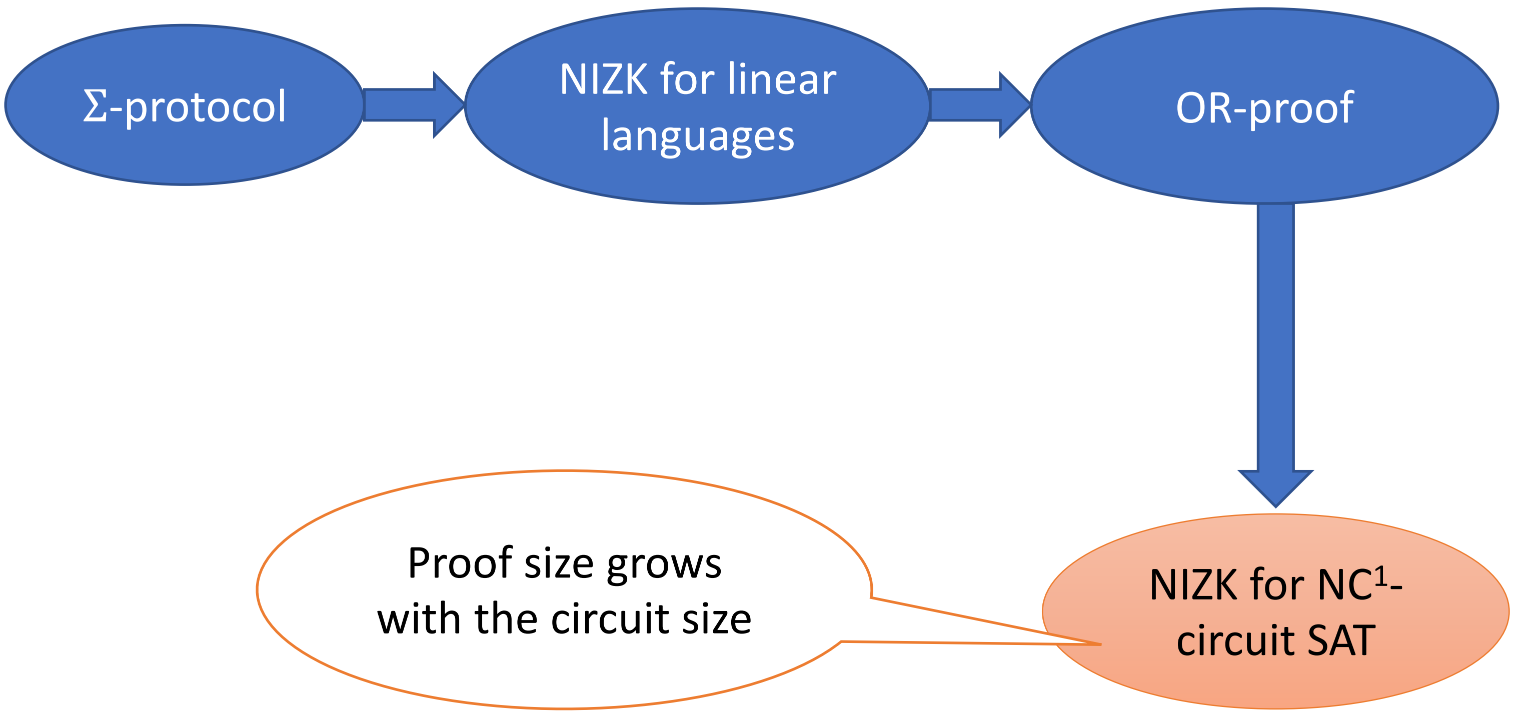


$(A^\top, s) \leftarrow \text{ZeroSamp}(\lambda)$
ct_i=Enc(A,w_i)
w_i=Dec(s,ct_i)

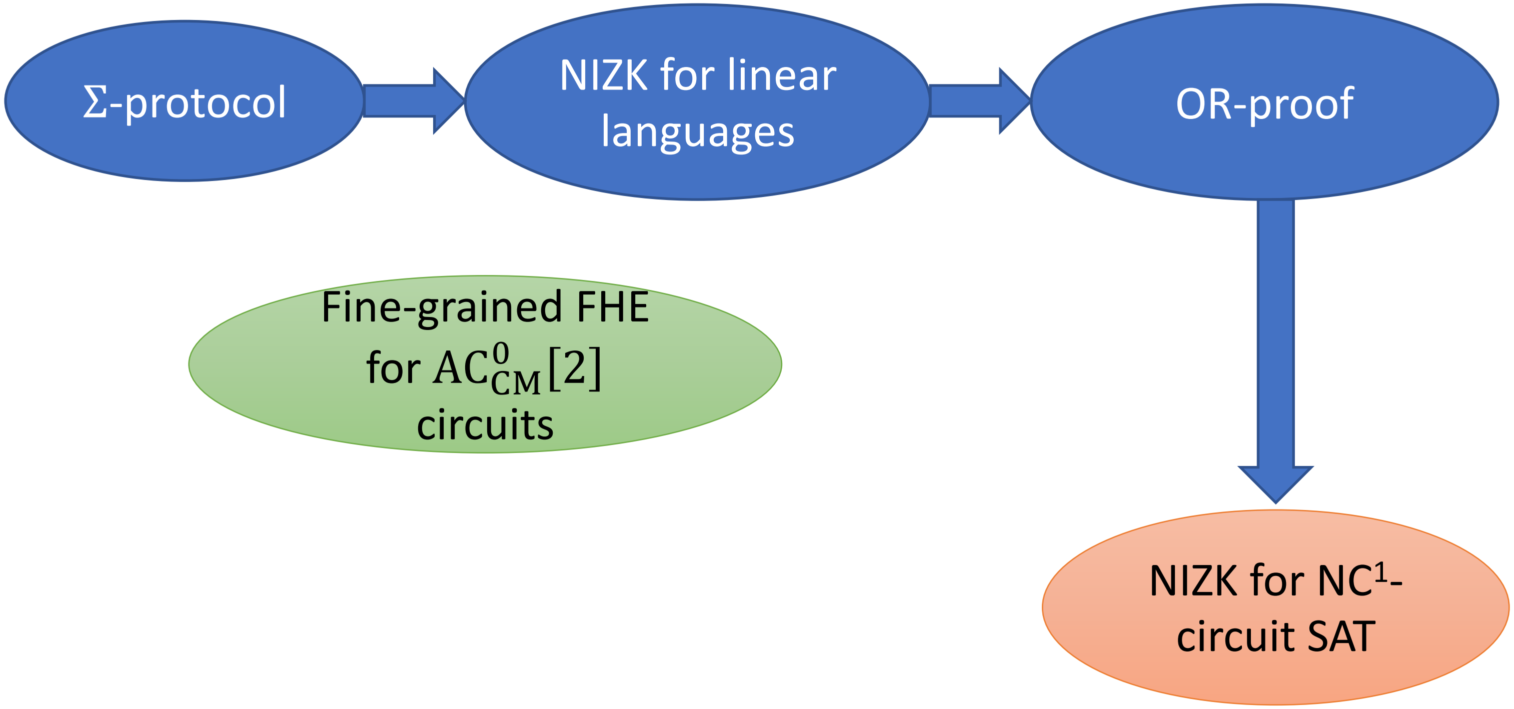
Zero-knowledge:

1. Ciphertexts become random matrices (when switching the distribution of A to OneSamp)
2. OR-proofs reveals no useful information due to its ZK

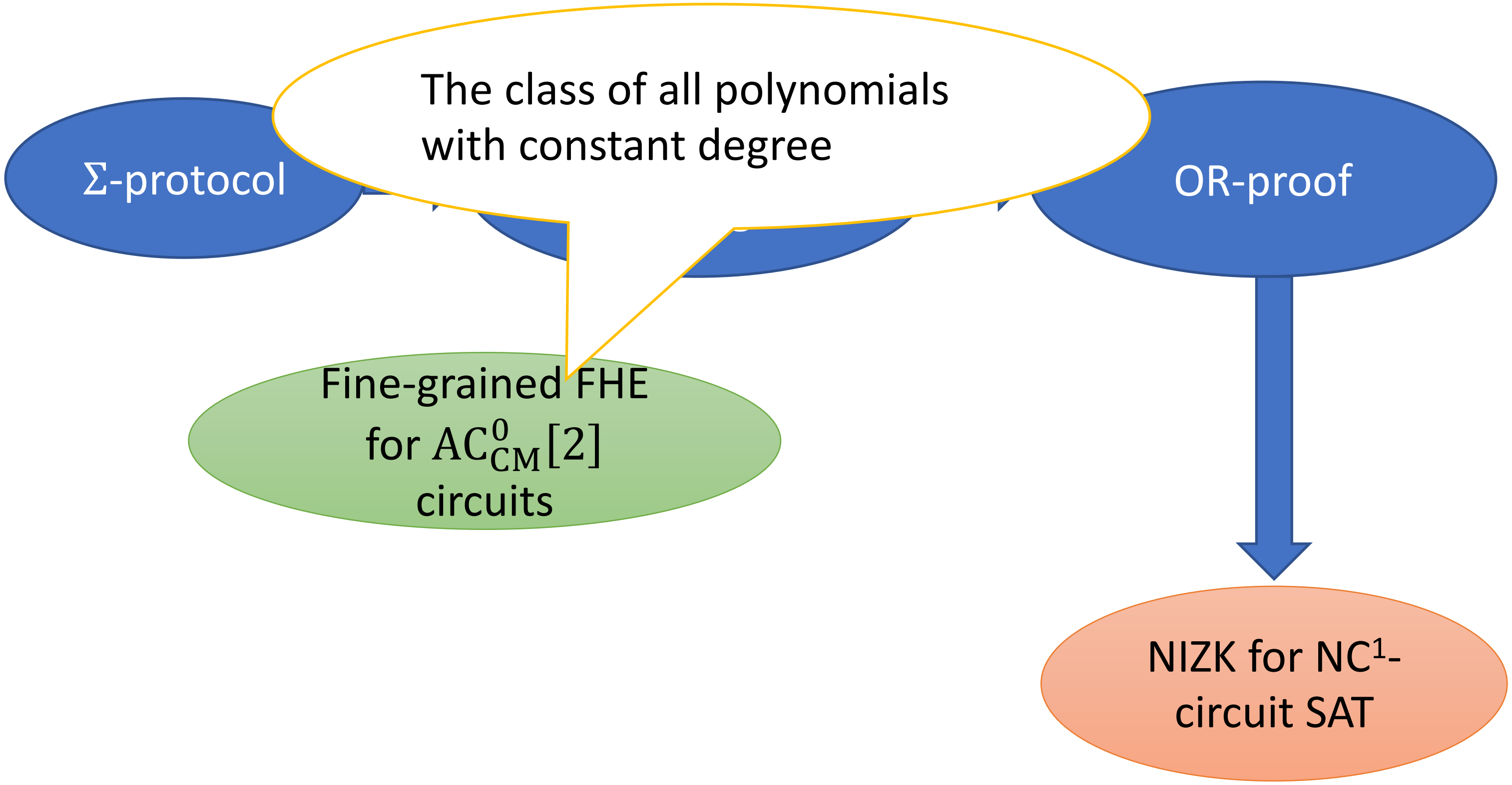
NIZK for NC^1 circuits



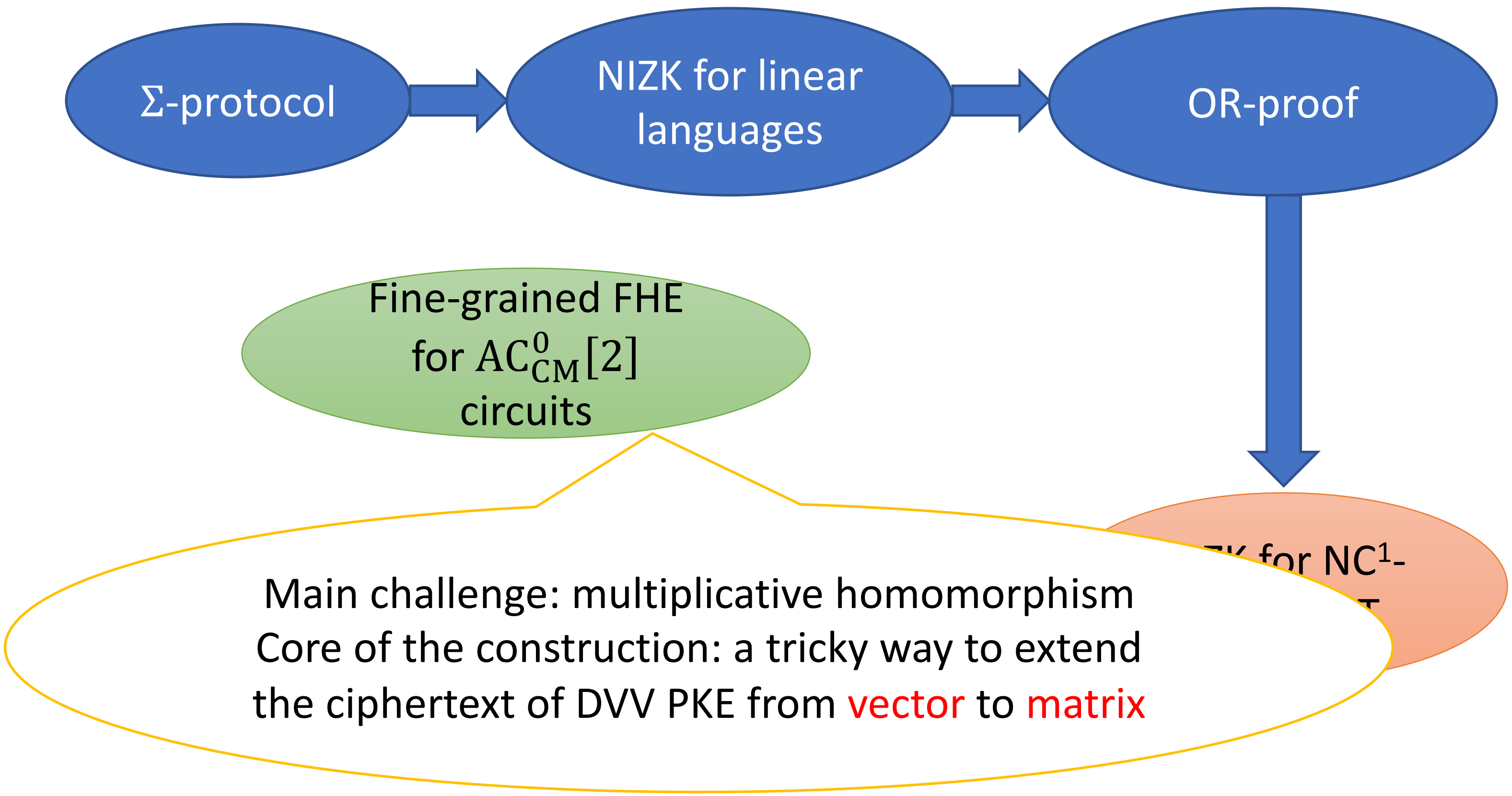
Fine-grained FHE



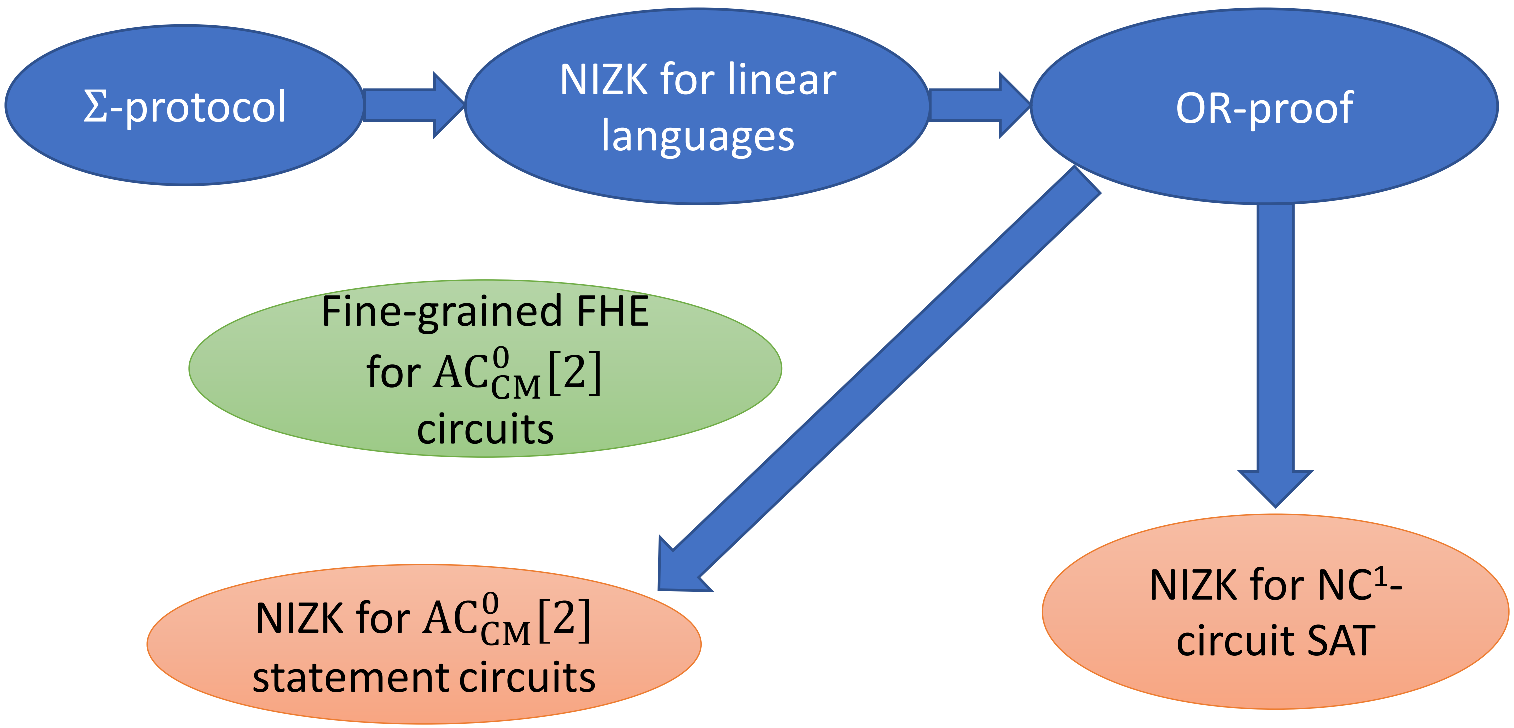
Fine-grained FHE



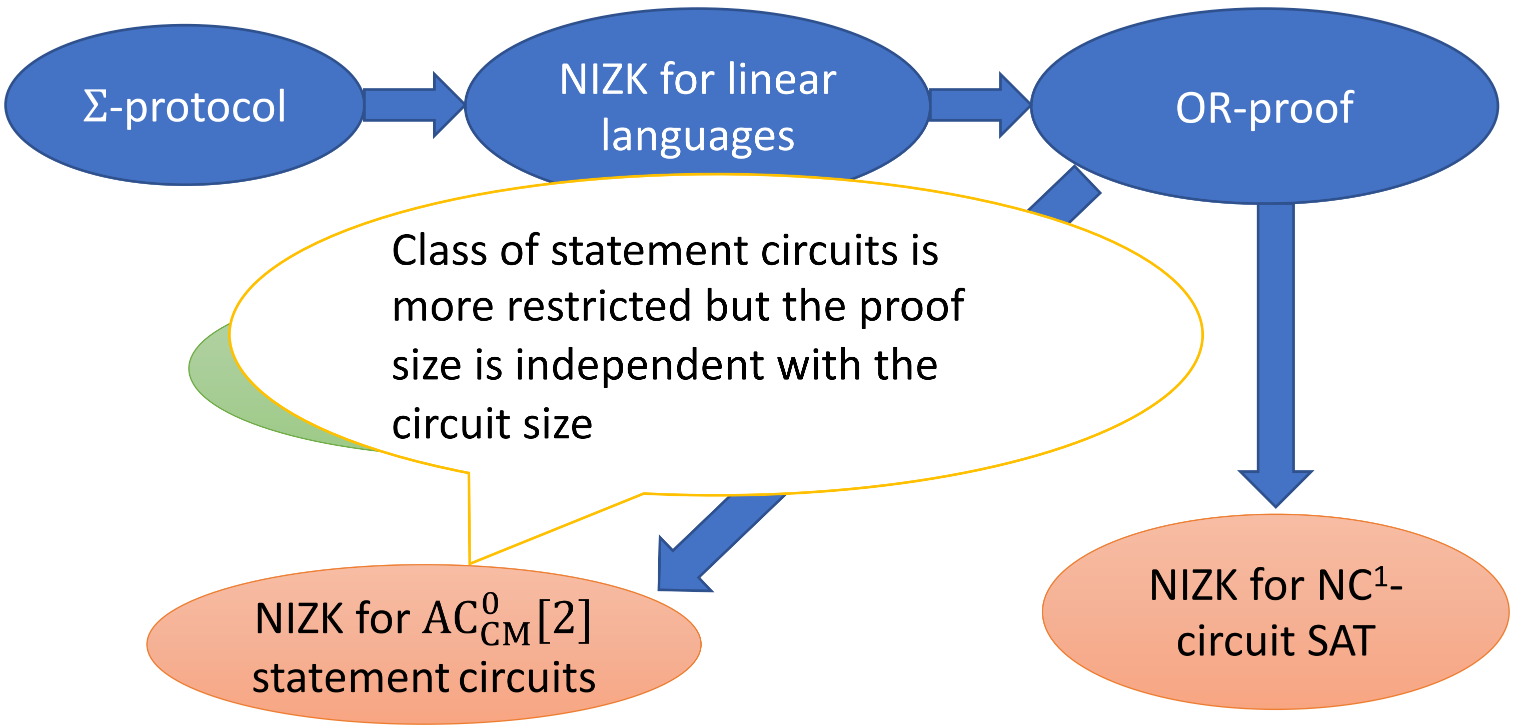
Fine-grained FHE



NIZK for $\mathrm{AC}^0_{\mathrm{CM}}[2]$ circuits with short proofs



NIZK for $\mathrm{AC}^0_{\mathrm{CM}}[2]$ circuits with short proofs



Extensions

- ❖ Conversion to non-interactive zaps (NIWI in the plain model)

Extensions

- Conversion to non-interactive zaps (NIWI in the plain model)

Extensions

❖ Conversion to non-interactive zaps (NIWI in the plain model)

❖ Conversion to NIZKs in the URS model

Extensions

- Conversion to non-interactive zaps (NIWI in the plain model)

Proof: our NIZKs have verifiable correlated key generation

GOS conversion technique [GOS12]

Non-interactive zaps

- Conversion to NIZKs in the URS model

Proof: a random matrix (with some particular form) is a binding/hiding CRS with half-half probabiltiy

Running NIZK with the random CRS in parallel

NIZK in the URS model

Extensions

- Conversion to non-interactive zaps (NIWI in the plain model)

Proof: our NIZKs have verifiable correlated key generation

GOS conversion technique [GOS12]

…tive zaps

ZK: ZK of NIZK in the CRS model
Statistical soundness: for multiple random strings, at least one should be binding with overwhelming probability.

- Conversion to NIZKs in the

Running NIZK with the random CRS in parallel

Proof: a random matrix (with some particular form) is a binding/hiding CRS with half-half probabiltiy

NIZK in the URS model

Conclusion

Proof systems secure against NC^1 adversaries under $\mathrm{NC}^1 \neq \bigoplus$L/poly

1. NIZK for NC_1-circuit SAT

2. NIZK for $\mathrm{AC}^0_{\mathrm{CM}}[2]$ circuits with short proofs
   - Fully homomorphic encryption for $\mathrm{AC}^0_{\mathrm{CM}}[2]$

3. Non-interactive zaps

4. NIZKs in the URS model