Parallel Repetition of $(k_1, \ldots, k_\mu)$-Special-Sound Multi-Round Interactive Proofs

Thomas Attema and Serge Fehr

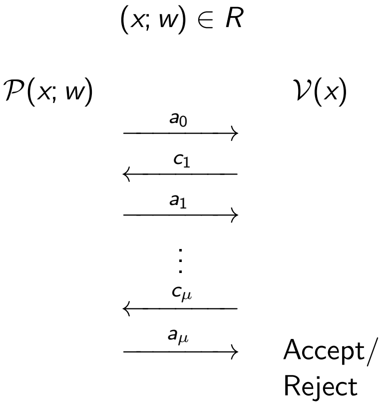
August 15, 2022

ePrint 2021/1259

Presentation Outline

A (binary) relation is a set $R = \{(x; w)\}$ of statement-witness pairs.

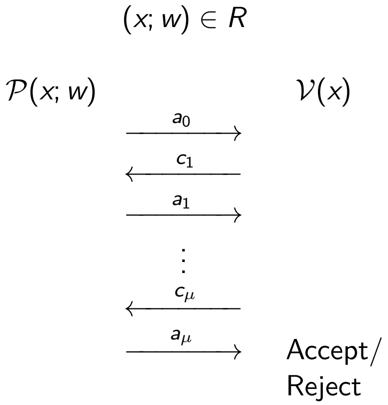
A (binary) relation is a set $R = \{(x; w)\}$ of statement-witness pairs.



A (binary) relation is a set $R = \{(x; w)\}$ of statement-witness pairs.

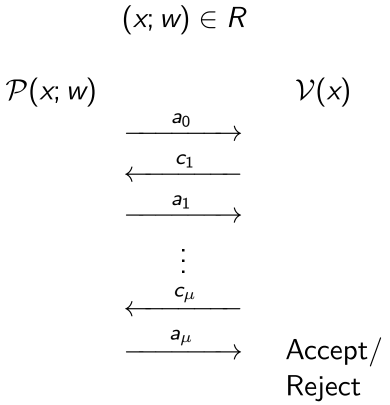
Goal of an Interactive Proof (of Knowledge):

- *Prove knowledge of a witness w for a public statement x.*



Preliminaries - Interactive Proofs

A (binary) relation is a set $R = \{(x; w)\}$ of statement-witness pairs.



Goal of an Interactive Proof (of Knowledge):

- *Prove knowledge of a witness w for a public statement x.*

We only consider public-coin protocols, i.e., the verifier's messages c_i are challenges sampled uniformly at random.

Preliminaries - Security Properties

Desirable Security Properties:

- Completeness: *Honest provers always succeed in convincing a verifier.*
- **Knowledge Soundness:** ***Dishonest provers (almost) never succeed.***
- Zero-Knowledge: *No information about the witness is revealed.*

Preliminaries - Knowledge Soundness

Knowledge soundness $\iff$ existence of a *knowledge extractor*.

Knowledge extractor

- Input: Statement x and oracle access to a prover $\mathcal{P}^*$ attacking the protocol.
- Goal: Compute a witness w for statement x.

Knowledge Soundness

- $\epsilon(x, \mathcal{P}^*)$: success probability of $\mathcal{P}^*$ on public input x.
- $\kappa(|x|)$: knowledge error of the protocol.

Knowledge Soundness

- $\epsilon(x, \mathcal{P}^*)$: success probability of $\mathcal{P}^*$ on public input x.
- $\kappa(|x|)$: knowledge error of the protocol.

Definition (Standard Definition - Knowledge Soundness)

If $\epsilon(x, \mathcal{P}^*) > \kappa(|x|)$, knowledge extractor extracts in expected runtime

$$\frac{\mathrm{poly}(|x|)}{\epsilon(x, \mathcal{P}^*) - \kappa(|x|)}.$$

Knowledge Soundness - Deterministic Provers

Lemma (Informal)

It is sufficient to consider deterministic provers $\mathcal{P}^*$.

Hence, $\mathcal{P}^*$ always starts with the same message.

Reducing the Knowledge Error

t-Fold Sequential Repetition:

- Reduces knowledge error from κ down to κ^t;
- Increases the number of rounds.

Reducing the Knowledge Error

t-Fold Sequential Repetition:

- Reduces knowledge error from κ down to κ^t;
- Increases the number of rounds.

t-Fold Parallel Repetition:

- Increases challenge set size from N to N^t;
- Hope is *strong* knowledge error reduction from κ down to κ^t.

Reducing the Knowledge Error

t-Fold Sequential Repetition:

- Reduces knowledge error from κ down to κ^t;
- Increases the number of rounds.

t-Fold Parallel Repetition:

- Increases challenge set size from N to N^t;
- Hope is *strong* knowledge error reduction from κ down to κ^t.

Generic (*weak*) result for any public-coin interactive proof:

- Reduces knowledge error from κ down to $\kappa^t + \nu$ for any non-negligible ν [ACK21].

Reducing the Knowledge Error

t-Fold Sequential Repetition:

- Reduces knowledge error from κ down to κ^t;
- Increases the number of rounds.

t-Fold Parallel Repetition:

- Increases challenge set size from N to N^t;
- Hope is *strong* knowledge error reduction from κ down to κ^t.

Generic (*weak*) result for any public-coin interactive proof:

- Reduces knowledge error from κ down to $\kappa^t + \nu$ for any non-negligible ν [ACK21].

This work: Strong parallel repetition result for a rich subclass of protocols: *special-sound* protocols.

Another Notion - Special-Soundness

- Easier to prove special-soundness than knowledge soundness.

Another Notion - Special-Soundness

- Easier to prove special-soundness than knowledge soundness.

Definition

2-out-of-N special-soundness: Efficient algorithm to extract a witness w from 2 'colliding' protocol transcripts (a, c, z) and (a, c', z').

Another Notion - Special-Soundness

- Easier to prove special-soundness than knowledge soundness.

Definition

2-out-of-N special-soundness: Efficient algorithm to extract a witness w from 2 'colliding' protocol transcripts (a, c, z) and (a, c', z').

2-out-of-N special-soundness implies knowledge soundness with knowledge error $1/N$.

Another Notion - Special-Soundness

- Easier to prove special-soundness than knowledge soundness.

Definition

2-out-of-N special-soundness: Efficient algorithm to extract a witness w from 2 'colliding' protocol transcripts (a, c, z) and (a, c', z').

2-out-of-N special-soundness implies knowledge soundness with knowledge error $1/N$.

Natural generalizations:

- k-out-of-N special-soundness $\implies$ knowledge error $(k-1)/N$.
- multi-round protocols:
  - Also here special-soundness tightly implies knowledge soundness (CRYPTO'21 [ACK21]).

Presentation Outline

1. Preliminaries
2. Prior Knowledge Extractor - Single Invocation
3. Parallel Repetition - Naive Extractor
4. Our Solution - Parallel Repetition of 3-Round Interactive Proofs
5. Our Solution - Parallel Repetition of Multi-Round Interactive Proofs
6. Summary

Knowledge Extractor - k-out-of-N Special-Sound Protocols (1/2)

Let Π be k-out-of-N special-sound,

- and $\mathcal{P}^*$ a *deterministic* prover attacking Π on input x.

$$\mathcal{P}^* \colon \mathcal{C} \to \{0,1\}^*, \quad c \mapsto z\,.$$

- $\mathcal{P}^*$'s first message a is fixed;
- $\mathcal{P}^*$ is successful if (a, c, z) is an accepting transcript.

Knowledge Extractor - k-out-of-N Special-Sound Protocols (1/2)

Let Π be k-out-of-N special-sound,

- and $\mathcal{P}^*$ a *deterministic* prover attacking Π on input x.

$$\mathcal{P}^* : \mathcal{C} \to \{0,1\}^*, \quad c \mapsto z\,.$$

- $\mathcal{P}^*$'s first message a is fixed;
- $\mathcal{P}^*$ is successful if (a, c, z) is an accepting transcript.

$\mathcal{P}^*$'s behavior can be summarized by a binary vector $H(\mathcal{P}^*)$ indexed by the challenges c_i.

- 1-entry corresponds to $\mathcal{P}^*$ succeeding;
- 0-entry corresponds to $\mathcal{P}^*$ failing.
- $\epsilon(x, \mathcal{P}^*)$ equals fraction of 1-entries.

$$\begin{array}{cccccccc} & & c_1 & c_2 & c_3 & \cdots & c_{N-1} & c_N \\ H(\mathcal{P}^*) = & (& 0 & 1 & 0 & \cdots & 0 & 1\ ) \end{array}$$

Knowledge Extractor - k-out-of-N Special-Sound Protocols (2/2)

$$\begin{array}{c c} & \begin{array}{cccccc} c_1 & c_2 & c_3 & \cdots & c_{N-1} & c_N \end{array} \\ H(\mathcal{P}^*) = & \begin{pmatrix} 0 & 1 & 0 & \cdots & 0 & 1 \end{pmatrix} \end{array}$$

Simple extraction algorithm:

(1) Sample entries until a 1-entry is found $\implies$ Expected time $1/\epsilon(x, \mathcal{P}^*)$.

Knowledge Extractor - k-out-of-N Special-Sound Protocols (2/2)

$$\begin{array}{rcccccc} & c_1 & c_2 & c_3 & \cdots & c_{N-1} & c_N \\ H(\mathcal{P}^*) = & (\,0 & 1 & 0 & \cdots & 0 & 1\,) \end{array}$$

Simple extraction algorithm:

(1) Sample entries until a 1-entry is found $\implies$ Expected time $1/\epsilon(x, \mathcal{P}^*)$.

(2) Sample entries until second 1-entry is found $\implies$ Expected time $\leq \dfrac{1}{\epsilon(x, \mathcal{P}^*) - 1/N}$.

Knowledge Extractor - k-out-of-N Special-Sound Protocols (2/2)

$$\begin{array}{cccccccc} & & c_1 & c_2 & c_3 & \cdots & c_{N-1} & c_N \\ H(\mathcal{P}^*) = & (& 0 & 1 & 0 & \cdots & 0 & 1 &) \end{array}$$

Simple extraction algorithm:

(1) Sample entries until a 1-entry is found $\implies$ Expected time $1/\epsilon(x, \mathcal{P}^*)$.

(2) Sample entries until second 1-entry is found $\implies$ Expected time $\leq \dfrac{1}{\epsilon(x, \mathcal{P}^*) - 1/N}$.

$\vdots$

(k) Sample entries until k-th 1-entry is found $\implies$ Expected time $\leq \dfrac{1}{\epsilon(x, \mathcal{P}^*) - (k-1)/N}$.

Knowledge Extractor - k-out-of-N Special-Sound Protocols (2/2)

$$\begin{array}{cccccccc} & & c_1 & c_2 & c_3 & \cdots & c_{N-1} & c_N \\ H(\mathcal{P}^*) = & (& 0 & 1 & 0 & \cdots & 0 & 1 &) \end{array}$$

Simple extraction algorithm:

(1) Sample entries until a 1-entry is found $\implies$ Expected time $1/\epsilon(x, \mathcal{P}^*)$.

(2) Sample entries until second 1-entry is found $\implies$ Expected time $\leq \dfrac{1}{\epsilon(x, \mathcal{P}^*) - 1/N}$.

$\vdots$

(k) Sample entries until k-th 1-entry is found $\implies$ Expected time $\leq \dfrac{1}{\epsilon(x, \mathcal{P}^*) - (k-1)/N}$.

Expected runtime $\leq \dfrac{k}{\epsilon(x, \mathcal{P}^*) - (k-1)/N}$.

Presentation Outline

2-Fold Parallel Repetition - Naive Extractor (1/4)

Consider $\mathcal{P}^*$ attacking the $t = 2$-fold parallel repetition Π^t.

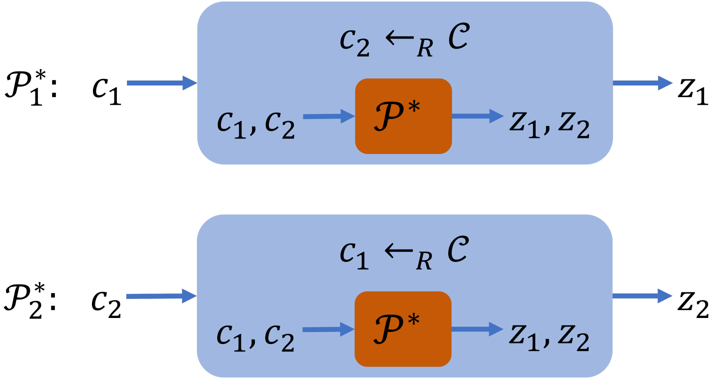
2-Fold Parallel Repetition - Naive Extractor (1/4)

Consider $\mathcal{P}^*$ attacking the $t = 2$-fold parallel repetition Π^t.

$\mathcal{P}^*$ is a (deterministic) function:

$$\mathcal{P}^* \colon \mathcal{C} \times \mathcal{C} \to \{0,1\}^*, \quad (c_1, c_2) \mapsto (z_1, z_2)\,.$$

$\mathcal{P}^*$ defines two provers attacking a single invocation of Π:



2-Fold Parallel Repetition - Naive Extractor (3/3)

Knowledge extractor:

- Run the "simple" knowledge extractor for both $\mathcal{P}_1^*$ and $\mathcal{P}_2^*$.

2-Fold Parallel Repetition - Naive Extractor (3/3)

Knowledge extractor:

- Run the "simple" knowledge extractor for both $\mathcal{P}_1^*$ and $\mathcal{P}_2^*$.
- The same analysis holds, even though $\mathcal{P}_1^*$ and $\mathcal{P}_2^*$ are not deterministic.

2-Fold Parallel Repetition - Naive Extractor (3/3)

Knowledge extractor:

- Run the "simple" knowledge extractor for both $\mathcal{P}_1^*$ and $\mathcal{P}_2^*$.
- The same analysis holds, even though $\mathcal{P}_1^*$ and $\mathcal{P}_2^*$ are not deterministic.

This does not work:

- Gives the same knowledge error $(k-1)/N$;
- Goal is to reduce knowledge error down to $(k-1)^2/N^2$.

Presentation Outline

Our Solution - Parallel Repetition of 3-Round Interactive Proofs

Technical Overview:

1. Introduce more fine-grained quality measure $\delta_k(x, \mathcal{P}^*)$ (instead of $\epsilon(x, \mathcal{P}^*)$).

Our Solution - Parallel Repetition of 3-Round Interactive Proofs

Technical Overview:

1. Introduce more fine-grained quality measure $\delta_k(x, \mathcal{P}^*)$ (instead of $\epsilon(x, \mathcal{P}^*)$).

2. Extractor for single invocations actually runs in time

$$\leq \frac{k}{\delta_k(x, \mathcal{P}^*)}.$$

Our Solution - Parallel Repetition of 3-Round Interactive Proofs

Technical Overview:

1. Introduce more fine-grained quality measure $\delta_k(x, \mathcal{P}^*)$ (instead of $\epsilon(x, \mathcal{P}^*)$).
2. Extractor for single invocations actually runs in time

$$\leq \frac{k}{\delta_k(x, \mathcal{P}^*)} .$$

3. **Parallel repetition**: At least one of the δ's is large enough, i.e., $\delta_k(x, \mathcal{P}_1^*)$ or $\delta_k(x, \mathcal{P}_2^*)$.

More Fine-Grained Analysis

Currently, the figure of merit is $\epsilon(x, \mathcal{P}^*)$, i.e.,

- the quality of the extractor is expressed in terms of $\epsilon(x, \mathcal{P}^*)$.

More Fine-Grained Analysis

Currently, the figure of merit is $\epsilon(x, \mathcal{P}^*)$, i.e.,

- the quality of the extractor is expressed in terms of $\epsilon(x, \mathcal{P}^*)$.

We define a 'punctured' success probability:

$$\delta_\ell(x, \mathcal{P}^*) = \min_{S \subset \mathcal{C} : |S| < \ell} \Pr\left(\mathcal{P}^*(C) \text{ succeeds} \mid C \notin S\right).$$

More Fine-Grained Analysis

Currently, the figure of merit is $\epsilon(x, \mathcal{P}^*)$, i.e.,

- the quality of the extractor is expressed in terms of $\epsilon(x, \mathcal{P}^*)$.

We define a 'punctured' success probability:

$$\delta_\ell(x, \mathcal{P}^*) = \min_{S \subset \mathcal{C} : |S| < \ell} \Pr\big(\mathcal{P}^*(C) \text{ succeeds} \mid C \notin S\big) \,.$$

$\delta_\ell(x, \mathcal{P}^*)$ lower bounds the success probability of $\mathcal{P}^*$ when "removing" $\ell - 1$ challenges.

Knowledge Extractor - Single Invocation and Probabilistic $\mathcal{P}^*$

Probabilistic $\mathcal{P}^*$ attacking a single invocation of a k-out-of-N special-sound protocol Π.

Simple extraction algorithm $\mathcal{E}^{\mathcal{P}^*}$:

(1) Sample entries until a 1-entry is found $\implies$ Expected time $1/\epsilon(x,\mathcal{P}^*) = 1/\delta_1(x,\mathcal{P}^*)$.

(2) Sample entries until second 1-entry is found $\implies$ Expected time $\leq 1/\delta_2(x,\mathcal{P}^*)$.

$\vdots$

(k) Sample entries until k-th 1-entry is found $\implies$ Expected time $\leq 1/\delta_k(x,\mathcal{P}^*)$.

Expected runtime $\leq \dfrac{k}{\delta_k(x,\mathcal{P}^*)}$.

Why does this refinement help?

$\mathcal{P}^*$ attacking the $t = 2$-fold parallel repetition Π^t.

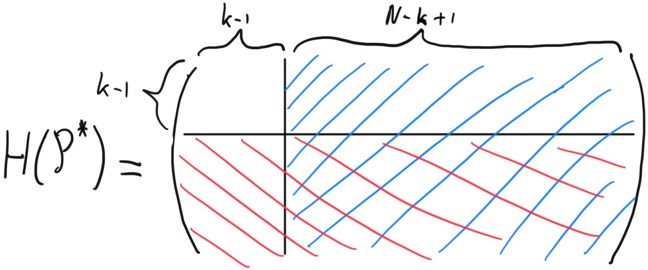
Why does this refinement help?

$\mathcal{P}^*$ attacking the $t = 2$-fold parallel repetition Π^t.

$$\begin{array}{c c} & \begin{array}{ccccc} c_1 & c_2 & \cdots & c_{N-1} & c_N \end{array} \\ H(\mathcal{P}^*) = & \begin{pmatrix} 0 & 1 & \cdots & 0 & 1 \\ 1 & 0 & \cdots & 1 & 1 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 1 & 1 & \cdots & 0 & 1 \end{pmatrix} \begin{array}{c} c_1 \\ c_2 \\ \vdots \\ c_N \end{array} \end{array}$$

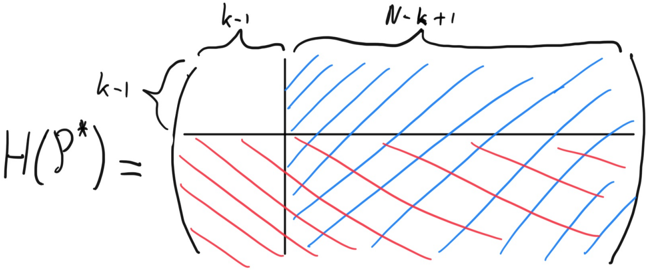
Why does this refinement help?

W.l.o.g. assume $H(\mathcal{P}^*)$'s rows and columns are sorted based on fraction of 1-entries.



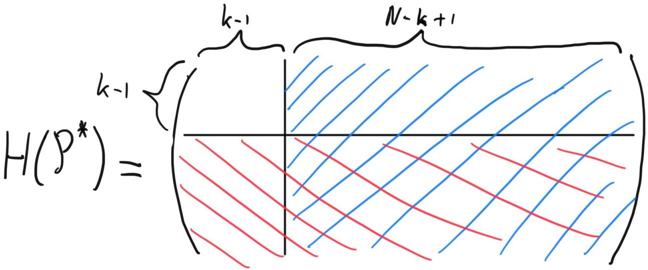
Why does this refinement help?

W.l.o.g. assume $H(\mathcal{P}^*)$'s rows and columns are sorted based on fraction of 1-entries.

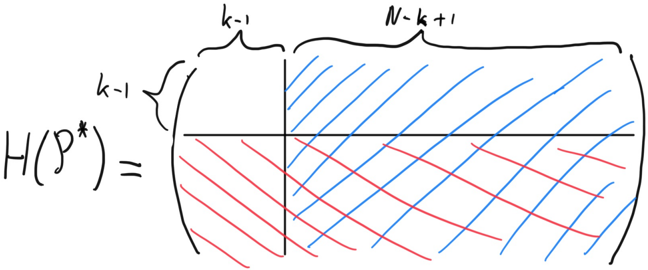


- $\delta_k(x, \mathcal{P}_1^*)$ = fraction of 1-entries in blue region.
- $\delta_k(x, \mathcal{P}_2^*)$ = fraction of 1-entries in red region.

Why does this refinement help?



Why does this refinement help?



$$\delta_k(x, \mathcal{P}_1^*) + \delta_k(x, \mathcal{P}_2^*) \geq \epsilon(x, \mathcal{P}^*) - \frac{(k-1)^2}{N^2}$$

$$\implies \max\left(\delta_k(x, \mathcal{P}_1^*), \delta_k(x, \mathcal{P}_2^*)\right) \geq \left(\epsilon(x, \mathcal{P}^*) - \frac{(k-1)^2}{N^2}\right) / 2$$

Strong Parallel Repetition Results for k-out-of-N Special-Sound Protocols

Theorem (3-Round Protocols)

The t-fold parallel repetition of a k-out-of-N special-sound interactive proof is knowledge sound with knowledge error

$$\frac{(k-1)^t}{N^t} \,.$$

Presentation Outline

Our Solution - Parallel Repetition of Multi-Round Interactive Proofs

- Natural recursive strategy from 3-round to $2\mu + 1$-round extraction [ACK21].

Our Solution - Parallel Repetition of Multi-Round Interactive Proofs

- Natural recursive strategy from 3-round to $2\mu + 1$-round extraction [ACK21].
- However, for the above extractor this gives runtime exponential in the number of rounds.

Our Solution - Parallel Repetition of Multi-Round Interactive Proofs

- Natural recursive strategy from 3-round to $2\mu + 1$-round extraction [ACK21].
- However, for the above extractor this gives runtime exponential in the number of rounds.
- **Solution:** New extractor for 3-round protocols properties making it amenable for this recursive strategy (see paper).

Summary

- New figure of merit δ capturing "how well we can extract".

$\Longrightarrow$ strong parallel repetition result for 3-round special-sound protocols.

- Novel 3-round extractor to handle multi-round protocols.

$\Longrightarrow$ strong parallel repetition for multi-round special-sound protocols.

- Also works for **threshold** parallel repetition.
    - Allowing to decrease completeness and knowledge error simultaneously.

Thanks!

Bibliography I

Thomas Attema, Ronald Cramer, and Lisa Kohl.
A compressed Σ-protocol theory for lattices.
In Tal Malkin and Chris Peikert, editors, *CRYPTO 2021, Part II*, volume 12826 of *LNCS*, pages 549–579, Virtual Event, August 2021. Springer, Heidelberg.