Karlsruhe Institute of Technology

KASTEL

An Automated Approach to Generating Card-Based Cryptographic Protocols

Final Bachelor's Thesis Presentation

Anne Hoff | March 30, 2023

Multi-Party Computation with Physical Objects

Multi-Party Computation (MPC):

Players want to correctly and securely compute a public function over their private inputs.

Physical Objects:

E.g.

(a) Borscht with Carrots and Onions (Miyahara et al., 2021)

(b) Coins (Komano and Mizuki, 2018)

(c) Playing Cards (Boer, 1990)

Advantages of MPC with Physical Objects

- Computers are not available, fail and/or are not trusted (Niemi and Renvall, 1998)
- Computations are performed by hand, so principles can be easily understood (Miyahara et al., 2021)
- Use in classrooms and lectures to illustrate MPC to nonexperts (Koch et al., 2015)

Motivation and Research Question

Current State of Research:
Protocols are mostly found "by hand"

↓

Koch et al. (2021) introduced an automated method
using formal verification to find card-based protocols

↓

Problems of the Automated Method:

- Just finds protocols for the *AND* operator
- Only works for small problems

↓

Research Question:
Can we expand upon the method of Koch et al. (2021) to make it more **universally applicable** and more **efficient**?

What are Card-Based Protocols?

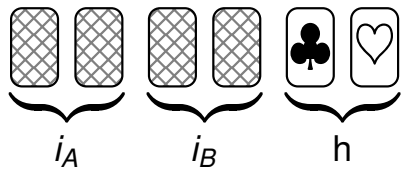
- Two-color deck: $\{\clubsuit, \heartsuit\}^n$
- Indistinguishable backsides: 🂠
- Commitment (encoding of a value):
  - $\clubsuit\ \heartsuit = 0 \mathrel{\widehat{=}}$ "No"
  - $\heartsuit\ \clubsuit = 1 \mathrel{\widehat{=}}$ "Yes"

input: Bob: $i_B =$ 🂠 🂠

Alice: $i_A =$ 🂠 🂠

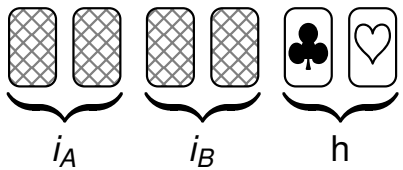
A Simple Card-Based Protocol

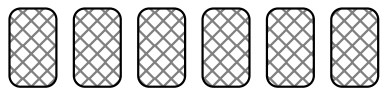
AND protocol by Mizuki and Sone (2009)

1. 

A Simple Card-Based Protocol

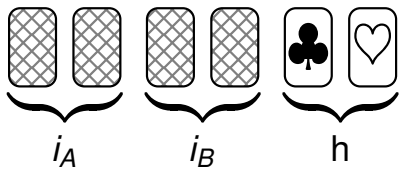
AND protocol by Mizuki and Sone (2009)

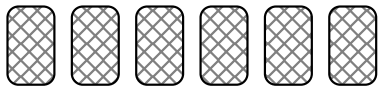
1. 

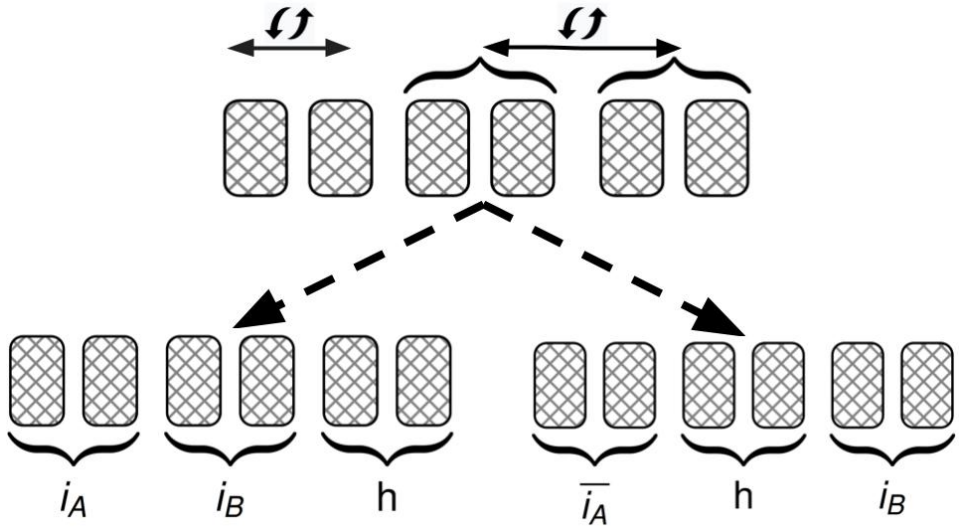
2. **Turn** all cards face-down: 

A Simple Card-Based Protocol

AND protocol by Mizuki and Sone (2009)

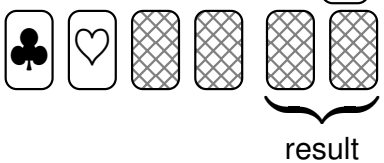
1. 

2. **Turn** all cards face-down: 
3. **Shuffle** the cards

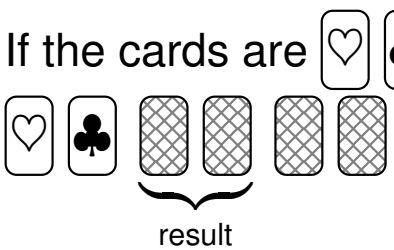


A Simple Card-Based Protocol

4. **Turn** around the first two cards
   - If the cards are ♣ ♡:

   - If the cards are ♡ ♣:

result: ♣ ♡ → "No"

♡ ♣ → "Yes"

Definitions for Card-based Protocols

Security

visible cards and output of the protocol do not reveal anything about the input

Input-possibilistic Security:
every input can produce any state of the protocol

Output-possibilistic Security:
at any state of the protocol, every output can still be possible

Shuffle Properties

Uniform Shuffle:
every permutation has the same probability

Closed Shuffle:
set of possible permutations is invariant under repetition

Finding Protocols with Software Bounded Model Checking

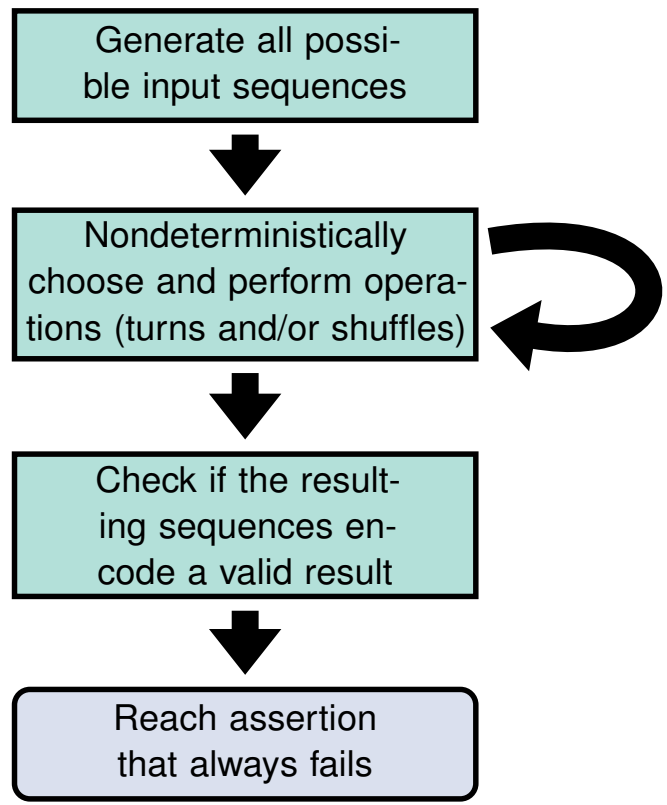
Karlsruhe Institute of Technology

- **Software Bounded Model Checking (SBMC)**:
  - Finds violations of assertions in programs (within given bound)
  - Performs static analysis without executing the programs on specific values
  - `assume()`: condition that provides more information about the program
  - `assert()`: property to check whether the program satisfies certain safety/correctness properties

Finding Protocols with Software Bounded Model Checking

- **Software Bounded Model Checking (SBMC)**:
  - Finds violations of assertions in programs (within given bound)
  - Performs static analysis without executing the programs on specific values
  - `assume()`: condition that provides more information about the program
  - `assert()`: property to check whether the program satisfies certain safety/correctness properties

- **The symbolic program by Koch et al. (2021)**:



Goal

Research Question

Can we expand upon the method of Koch et al. (2021) to increase its **universality** and **efficiency**?

Goal

Research Question

Can we expand upon the method of Koch et al. (2021) to increase its **universality** and **efficiency**?

Tasks:

1. Adapt symbolic program to find protocols for any function (Universal Application)

Goal

Research Question

Can we expand upon the method of Koch et al. (2021) to increase its **universality** and **efficiency**?

Tasks:

1. Adapt symbolic program to find protocols for any function (Universal Application)
2. Introduce nested approach that uses protocols as operations (Universal Application & Efficiency)

Goal

Research Question

Can we expand upon the method of Koch et al. (2021) to increase its **universality** and **efficiency**?

Tasks:

1. Adapt symbolic program to find protocols for any function (Universal Application)
2. Introduce nested approach that uses protocols as operations (Universal Application & Efficiency)
3. Explore different SAT solvers (Efficiency)

Goal

Research Question

Can we expand upon the method of Koch et al. (2021) to increase its **universality** and **efficiency**?

Tasks:

1. Adapt symbolic program to find protocols for any function (Universal Application)
2. Introduce nested approach that uses protocols as operations (Universal Application & Efficiency)
3. Explore different SAT solvers (Efficiency)
4. Evaluate bit-level data structure for efficient operations (Efficiency)

1. Generalizing the Symbolic Program

Karlsruhe Institute of Technology

Question

Can we adapt the symbolic program to find protocols for any function?

1. Generalizing the Symbolic Program

Karlsruhe Institute of Technology

Question

Can we adapt the symbolic program to find protocols for any function?

Approach

- Function components: **domain**, **codomain** and **function behaviour**
- adapt symbolic program when components change
- E.g. *COPY* instead of *AND* $\rightarrow$ domain: one input commitment $\rightarrow$ alter generation of possible input sequences

Protocols Found with the Generalized Symbolic Program

- Secure protocols found (excerpt):

Boolean Function	Nr. Cards	Nr Steps (Best Case)	Type Shuffle
OR	4	4	uniform, not closed
	4	6	not uniform, closed
COPY	5	2	uniform, not closed

- Function without any protocols found: half adder, instead:
  - timeout or an "out-of-memory" error for five or more cards

2. Introducing a Nested Structure

Karlsruhe Institute of Technology

Question

Can we implement a nested approach that uses protocols as operations?

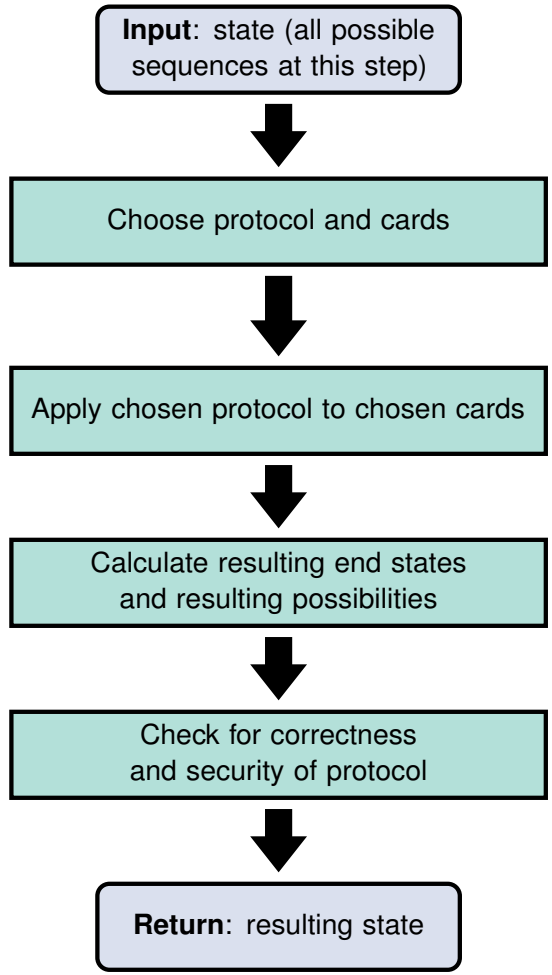
2. Introducing a Nested Structure

Question

Can we implement a nested approach that uses protocols as operations?

Why use protocols as operations?

- Reduce complexity of search space
- use known protocols from the literature to compose correct and secure larger protocols



Evaluation of the Nested Structure

Implementation:

- General implementation of the protocol action
- Can be integrated into the symbolic program for any function

Evaluation of the Nested Structure

Implementation:

- General implementation of the protocol action
- Can be integrated into the symbolic program for any function

Experiment Setup:

- Implemented *AND*, *OR*, *XOR* and *COPY* protocols as operations
- Searched for *COPY* and half adder protocols

Evaluation of the Nested Structure

Implementation:

- General implementation of the protocol action
- Can be integrated into the symbolic program for any function

Experiment Setup:

- Implemented *AND*, *OR*, *XOR* and *COPY* protocols as operations
- Searched for *COPY* and half adder protocols

Results:

- A *COPY* protocol using the *AND* protocol by Mizuki and Sone (2009)
- No protocols found for the half adder

Evaluation of the Nested Structure

Implementation:

- General implementation of the protocol action
- Can be integrated into the symbolic program for any function

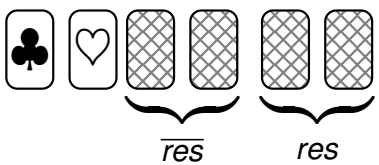
Experiment Setup:

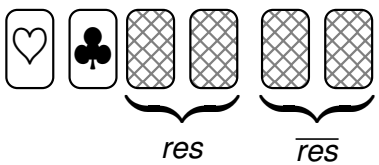
- Implemented *AND*, *OR*, *XOR* and *COPY* protocols as operations
- Searched for *COPY* and half adder protocols

Results:

- A *COPY* protocol using the *AND* protocol by Mizuki and Sone (2009)
- No protocols found for the half adder

The new *COPY* protocol:

- Card we want to copy: i_C
- Apply *AND* protocol with $i_A = i_C$ and i_B = "0"
- After turning the first two cards:
  - 

  - 

3. Experimenting with SAT Solvers

Karlsruhe Institute of Technology

Question

Can we make our method more efficient by using a different SAT solver?

[1] https://www.labri.fr/perso/lsimon/research/glucose/
[2] http://fmv.jku.at/cadical/

3. Experimenting with SAT Solvers

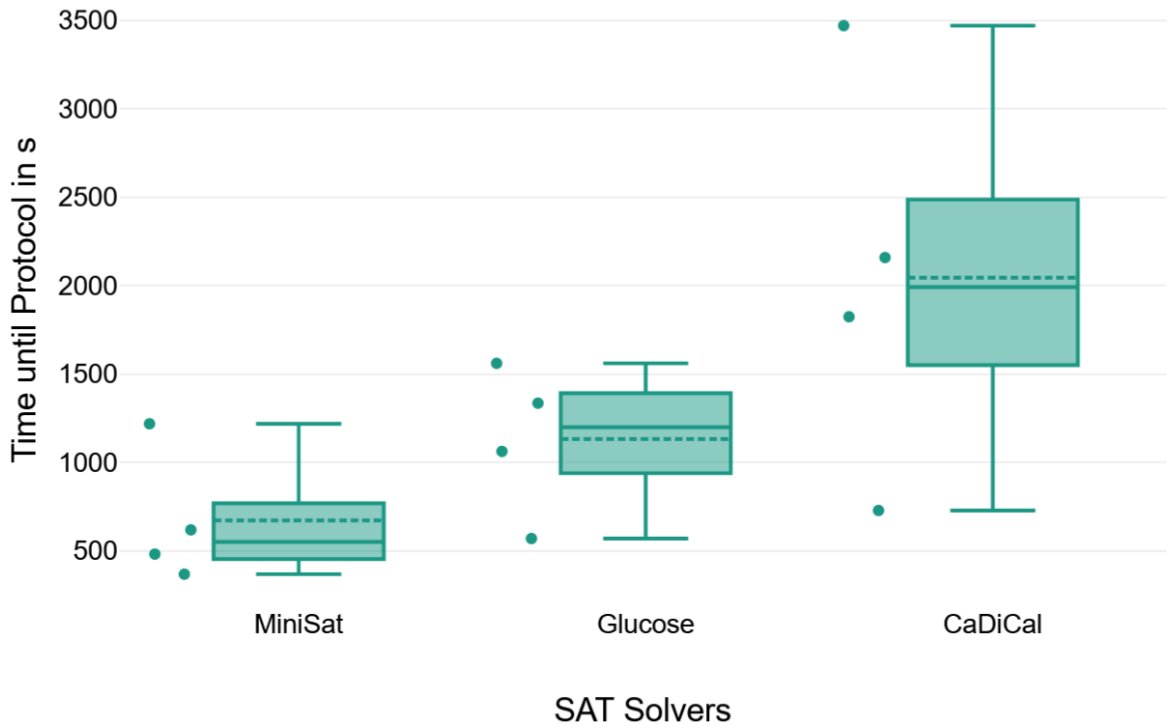
Question

Can we make our method more efficient by using a different SAT solver?

- Built-in SAT solver: MiniSat
- Alternative SAT solvers:
  - CBMC interface for various SAT and SMT solvers
  - Chosen solvers: Glucose [1] and Cadical [2]
    - Fast and efficient, award-winning SAT-Solvers

[1] https://www.labri.fr/perso/lsimon/research/glucose/
[2] http://fmv.jku.at/cadical/

Results of SAT Solver Experiments



- 4 experiments
  - 2x with *XOR* function that has a protocol and different security definitions
  - 2x with *OR* function where no protocol exists and different security definitions

4. Evaluating an Alternative Data Structure

Question

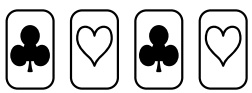
Is there a data structure that is more efficient than the one by Koch et al. (2021)?

4. Evaluating an Alternative Data Structure

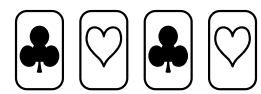
Question

Is there a data structure that is more efficient than the one by Koch et al. (2021)?

Koch et al. (2021):

- Arrays
- E.g.  → `[1, 2, 1, 2]`

Alternative data structure:

- Cards are bits in a single variable
- E.g.  → `0101`

4. Evaluating an Alternative Data Structure

Question

Is there a data structure that is more efficient than the one by Koch et al. (2021)?

Koch et al. (2021):

- Arrays
- E.g. ♣ ♡ ♣ ♡ → `[1, 2, 1, 2]`

Alternative data structure:

- Cards are bits in a single variable
- E.g. ♣ ♡ ♣ ♡ → `0101`

Advantage of the Alternative Data Structure

- **Hypothesis**: Bitwise operations are **faster** than array accesses

Turn Operation

Using Array Representation

```
turnedCardNumber
    = seq.val[turnPosition];
```

Using Bit Representation

```
turnedCardNumber =
    (sequence.val &
    (1 << turnPosition));
```

Shuffle Operation

Applying permutation j to sequence i:

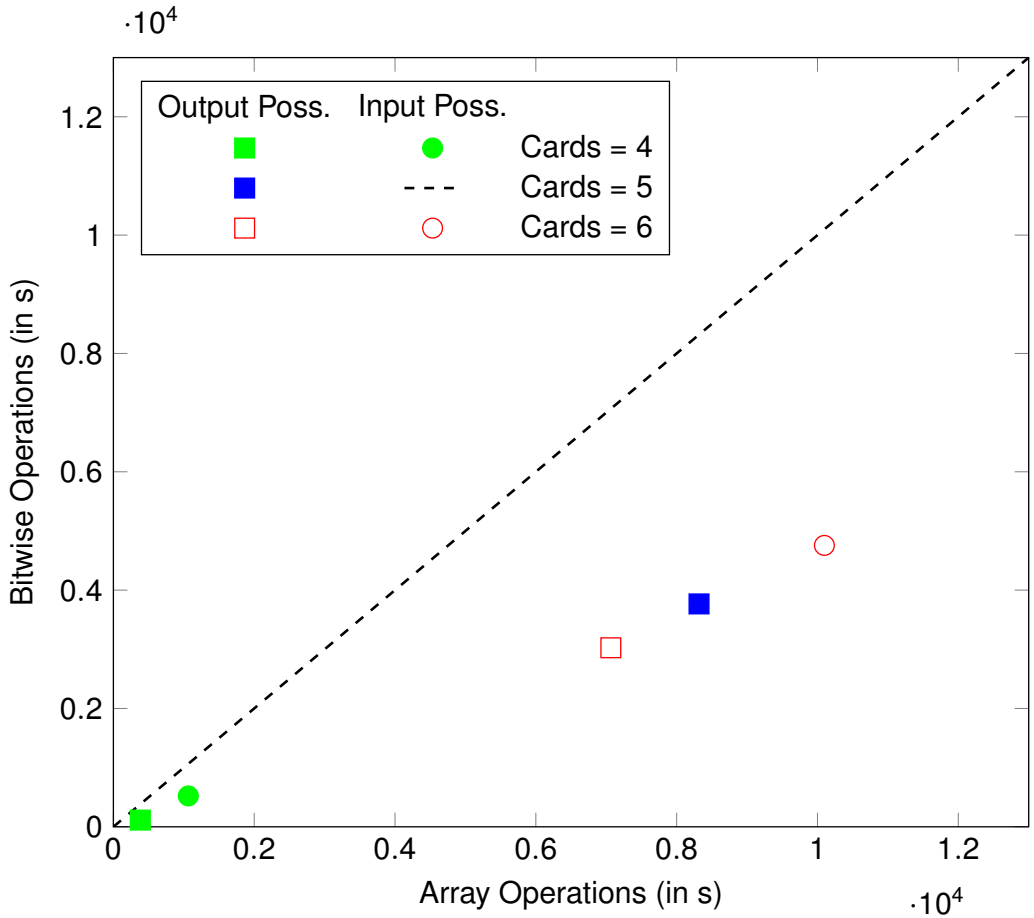
Using Array Representation

```
for (k = 0; k < N; k++) {
    resultingSeq.arr[
        permutationSet[j][k]]
    = s.seq[i].val[k];
}
```

Using Bit Representation

```
resultingSeq = 0;
for (k = 0; k < N; k++) {
    temp = seq.val & (1 << k);
    shift =
        permutationSet[j][k]
        - k;
    resultingSeq =
        resultingSeq |
        temp << (shift);
}
```

Experiments with the Alternative Data Structure



Conclusion

Results

- Generalized the symbolic program by Koch et al. (2021)
  - Discovered new protocols
- Introduced and implemented technique to integrate arbitrary protocols as actions
  - Discovered a *COPY* protocol using boolean operators
- Evaluated the efficiency of using different SAT solvers
- Introduced a bitwise data structure for sequence representation
  - Improved the runtime of the bounded model checker in an experiment setting

Conclusion

Results

- Generalized the symbolic program by Koch et al. (2021)
  - Discovered new protocols
- Introduced and implemented technique to integrate arbitrary protocols as actions
  - Discovered a *COPY* protocol using boolean operators
- Evaluated the efficiency of using different SAT solvers
- Introduced a bitwise data structure for sequence representation
  - Improved the runtime of the bounded model checker in an experiment setting

Future Work

- Standardized program using the bitwise data structure
- Test with further SAT or SMT solvers

Bibliography I


Boer, Bert den (1990). “More Efficient Match-Making and Satisfiability The Five Card Trick”. en. In: *Advances in Cryptology — EUROCRYPT ’89*. Ed. by Jean-Jacques Quisquater and Joos Vandewalle. Lecture Notes in Computer Science. Berlin, Heidelberg: Springer, pp. 208–217. ISBN: 978-3-540-46885-1. DOI: 10.1007/3-540-46885-4_23.


Koch, Alexander, Michael Schrempp, and Michael Kirsten (2021). “Card-Based Cryptography Meets Formal Verification”. In: *New Gener. Comput.* 39.1, pp. 115–158. DOI: 10.1007/s00354-020-00120-0.

Bibliography II


Koch, Alexander, Stefan Walzer, and Kevin Härtel (2015). “Card-Based Cryptographic Protocols Using a Minimal Number of Cards”. In: *Advances in Cryptology - ASIACRYPT 2015 - 21st International Conference on the Theory and Application of Cryptology and Information Security, Auckland, New Zealand, November 29 - December 3, 2015, Proceedings, Part I*. Ed. by Tetsu Iwata and Jung Hee Cheon. Vol. 9452. Lecture Notes in Computer Science. Springer, pp. 783–807. DOI: 10.1007/978-3-662-48797-6_32.


Komano, Yuichi and Takaaki Mizuki (2018). “Multi-party Computation Based on Physical Coins”. In: *Theory and Practice of Natural Computing - 7th International Conference, TPNC 2018, Dublin, Ireland, December 12-14, 2018, Proceedings*. Ed. by David Fagan et al. Vol. 11324. Lecture Notes in Computer Science. Springer, pp. 87–98. DOI: 10.1007/978-3-030-04070-3_7. URL: https://doi.org/10.1007/978-3-030-04070-3_7.

Bibliography III


Miyahara, Daiki et al. (2021). “Cooking Cryptographers: Secure Multiparty Computation Based on Balls and Bags”. In: *34th IEEE Computer Security Foundations Symposium, CSF 2021, Dubrovnik, Croatia, June 21-25, 2021*. IEEE, pp. 1–16. DOI: 10.1109/CSF51468.2021.00034. URL: https://doi.org/10.1109/CSF51468.2021.00034.


Mizuki, Takaaki and Hideaki Sone (2009). “Six-Card Secure AND and Four-Card Secure XOR”. In: *Frontiers in Algorithmics, Third International Workshop, FAW 2009, Hefei, China, June 20-23, 2009. Proceedings*. Ed. by Xiaotie Deng, John E. Hopcroft, and Jinyun Xue. Vol. 5598. Lecture Notes in Computer Science. Springer, pp. 358–369. DOI: 10.1007/978-3-642-02270-8_36.

Bibliography IV

 Niemi, Valtteri and Ari Renvall (Jan. 1998). “Secure multiparty computations without computers”. en. In: *Theoretical Computer Science* 191.1, pp. 173–183. ISSN: 0304-3975. DOI: 10.1016/S0304-3975(97)00107-2. URL: https://www.sciencedirect.com/science/article/pii/S0304397597001072.

Picture Sources

- Soup: https://images.emojiterra.com/google/noto-emoji/v2.034/512px/1f372.png
- Cards: http://clipart-library.com/clipart/kTMbeg59c.htm
- Coin: https://images.emojiterra.com/google/android-11/512px/1fa99.png
- Coffee Date: https://c8.alamy.com/comp/2DA676P/single-continuous-line-drawing-of-young-happy-male-and-female-couple-doing-romantic-date-and-dinner-together-at-coffee-shop-marriage-life-concept-2DA676P.jpg
- Title Image: https://encrypted-tbn1.gstatic.com/images?q=tbn:ANd9GcSV7IX0PlyXBbN8PXqQH-xWOwZhK5PP9BZM0RO_TkUFXZzH_SbO

Karlsruhe Institute of Technology

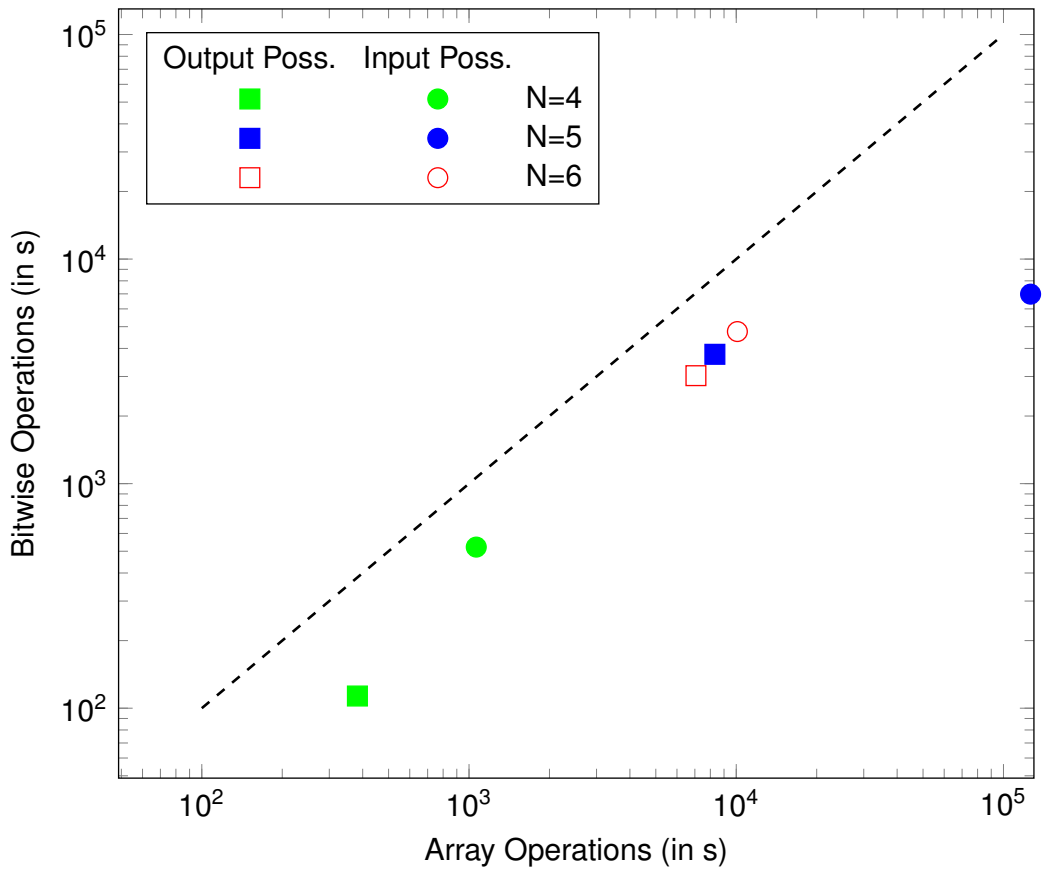
KASTEL

An Automated Approach to Generating Card-Based Cryptographic Protocols

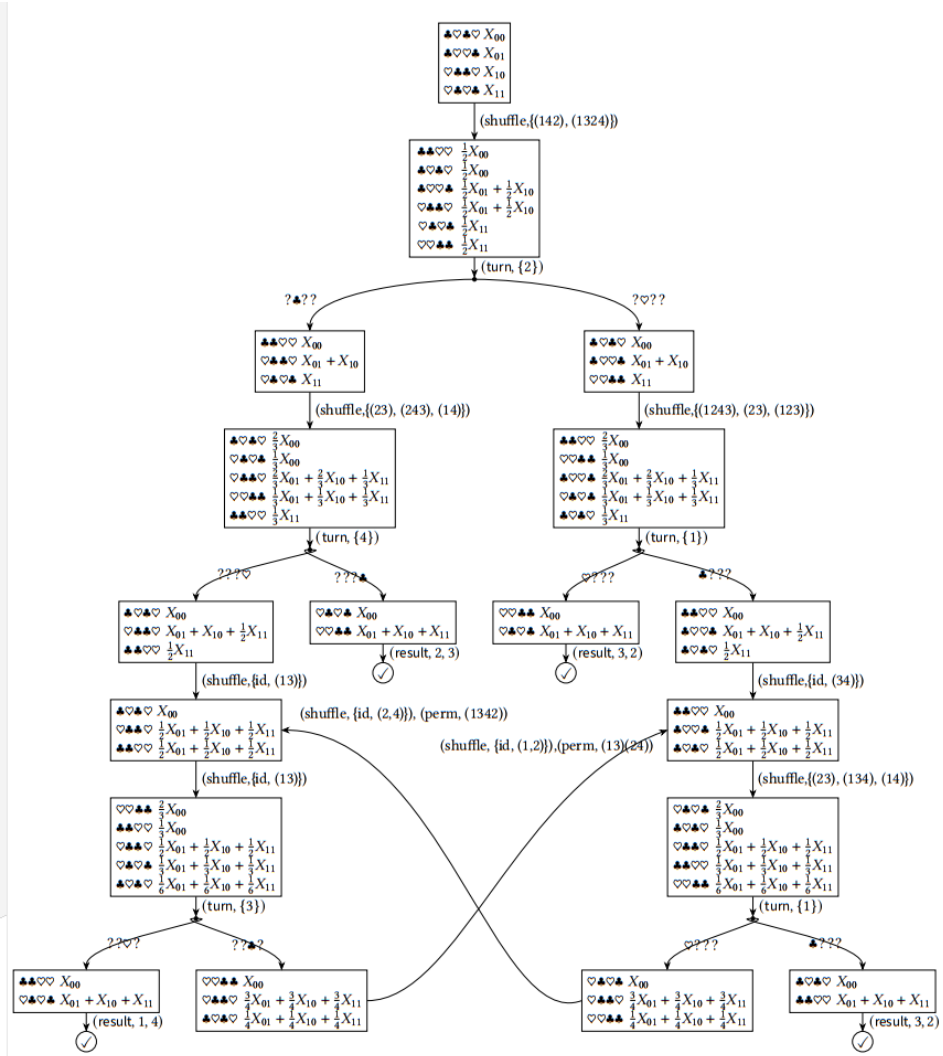
Final Bachelor's Thesis Presentation

Anne Hoff | March 30, 2023

Experiments with the Alternative Data Structure - Logarithmic Scale



OR Protocol with Uniform, Non-closed Shuffles



OR Protocol with Non-Uniform, closed Shuffles