Karlsruhe Institute of Technology

Card-based Cryptographic Protocols Using a Minimal Number of Cards

ASIACRYPT 2015 | Alexander Koch, Stefan Walzer, Kevin Härtel

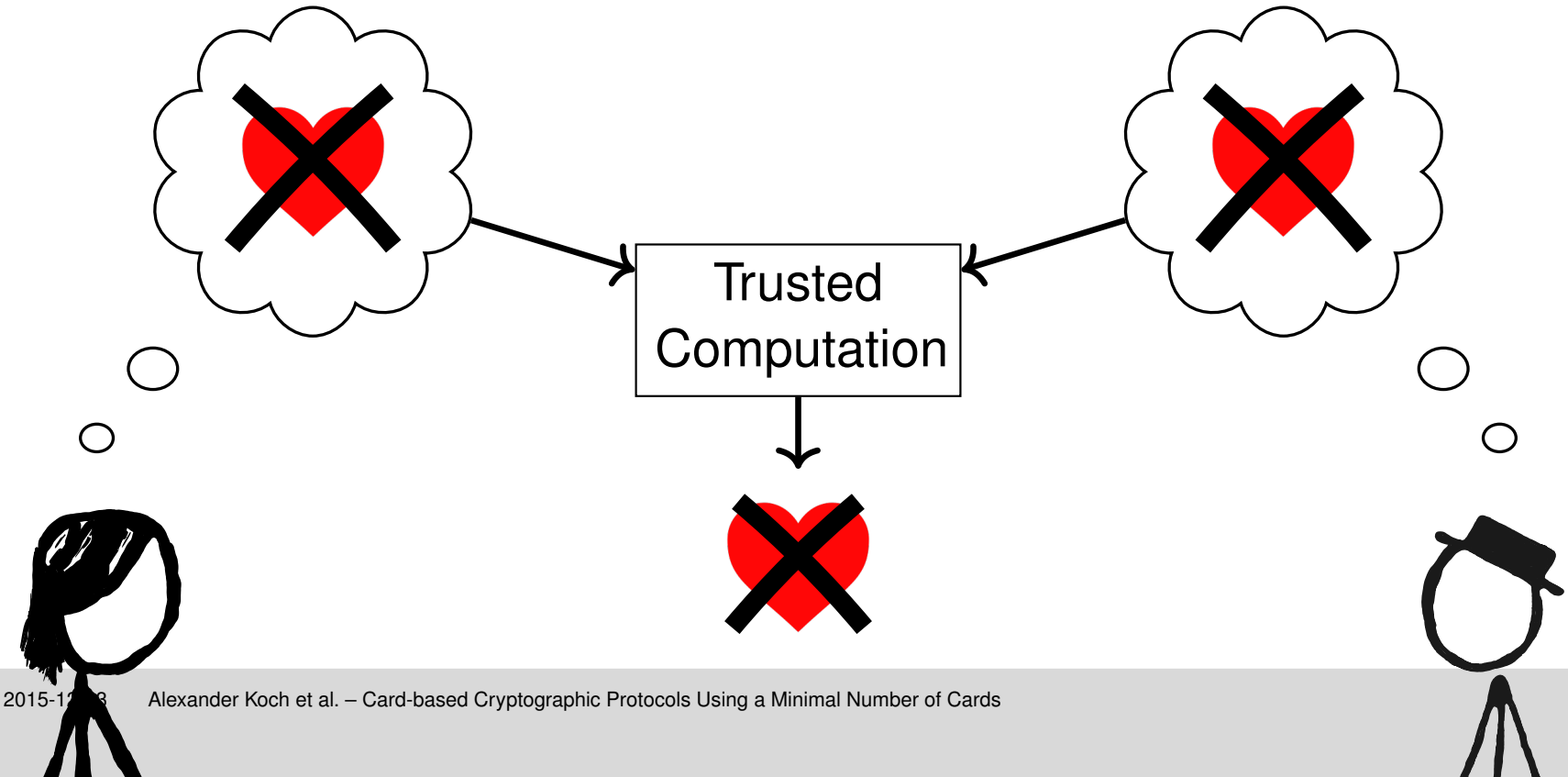
DEPARTMENT OF INFORMATICS, INSTITUTE OF THEORETICAL INFORMATICS

Motivating Scenario I

Calculating Mutual Interest with Playing Cards

- Secrets: Do I fancy him/her?
- To compute: Is there mutual interest?

⇝ Secure 2-party AND without computers

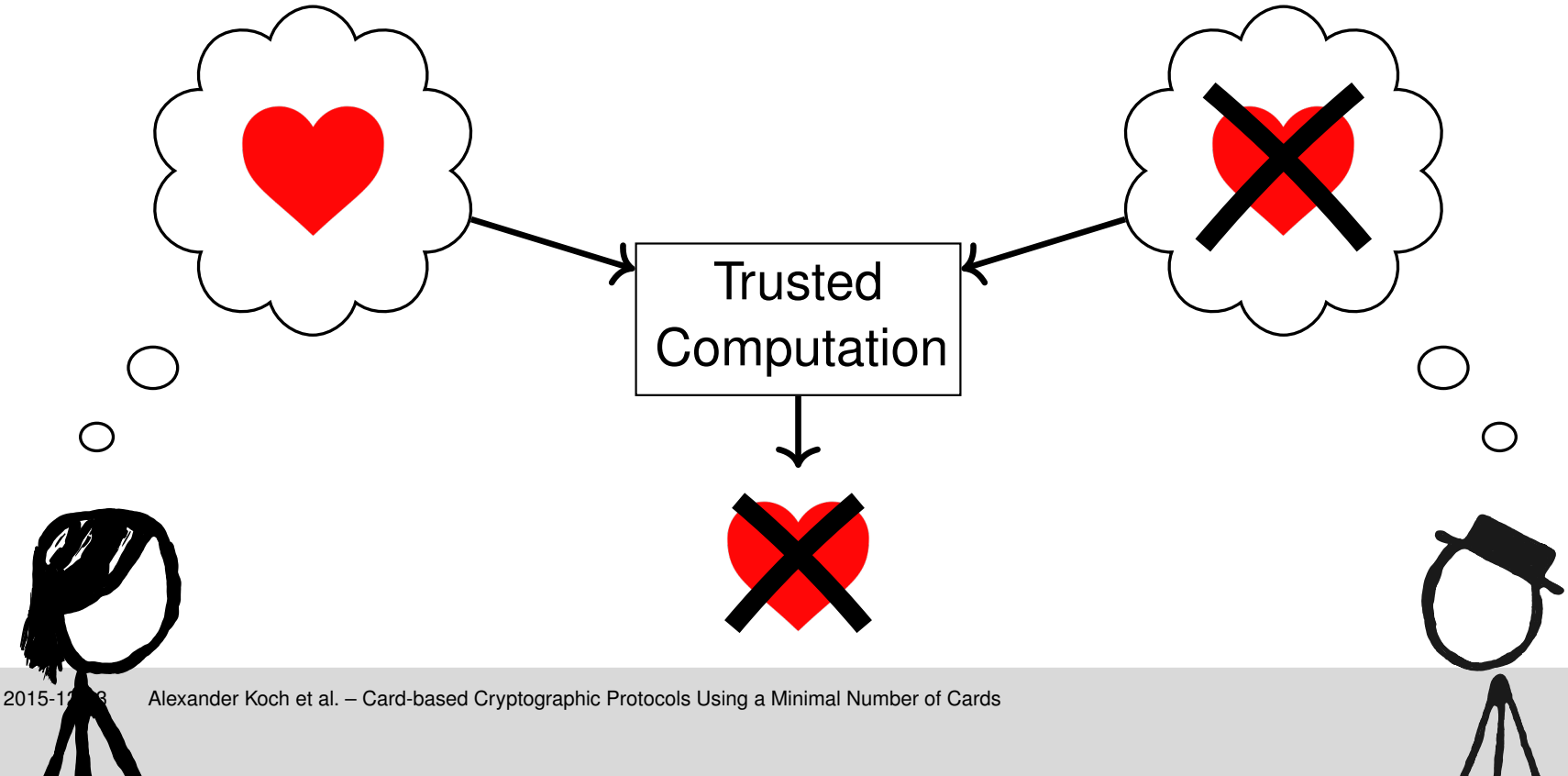


Motivating Scenario I

Calculating Mutual Interest with Playing Cards

- Secrets: Do I fancy him/her?
- To compute: Is there mutual interest?

⇝ Secure 2-party AND without computers

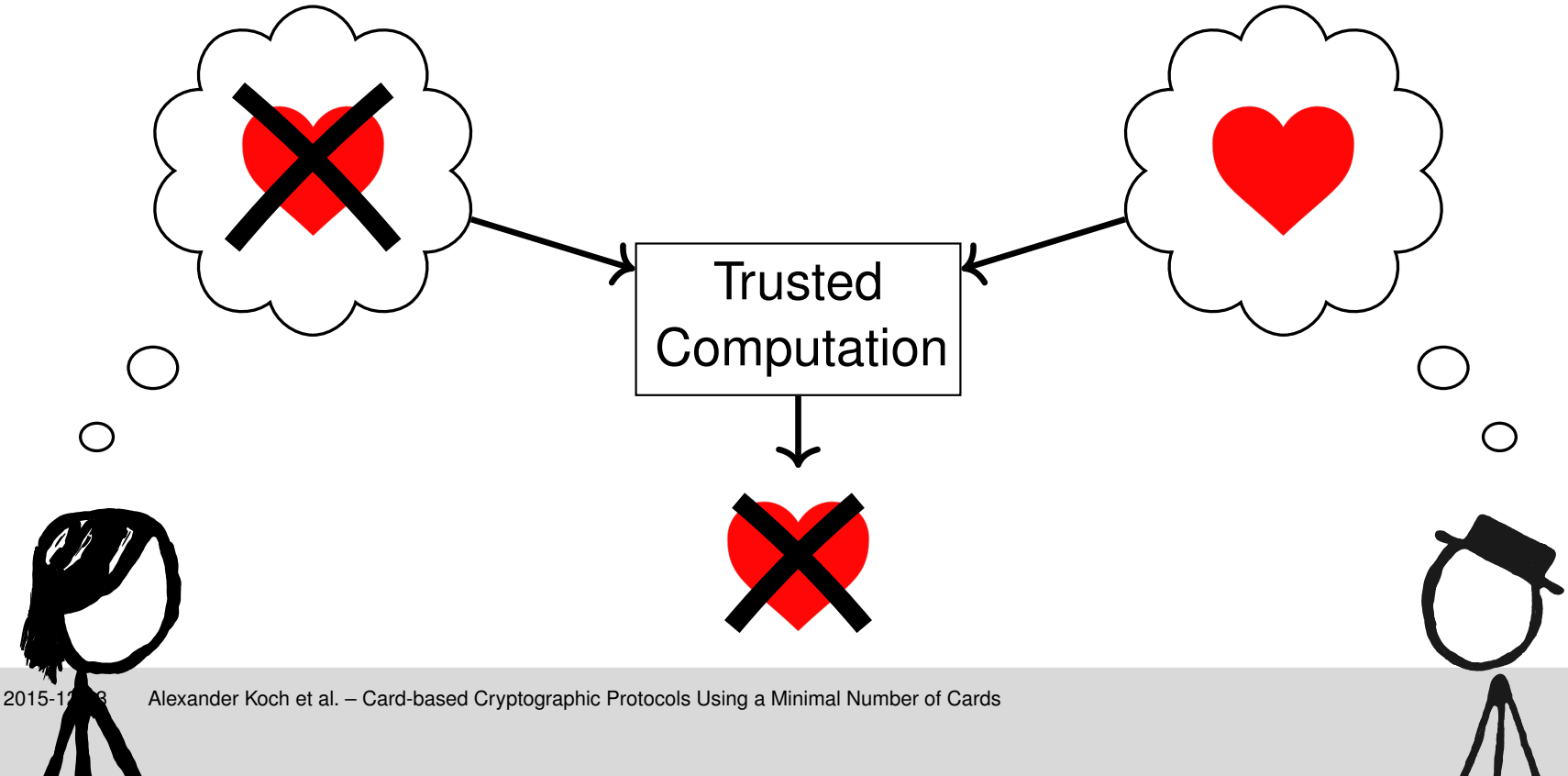


Motivating Scenario I

Calculating Mutual Interest with Playing Cards

- Secrets: Do I fancy him/her?
- To compute: Is there mutual interest?

⇝ Secure 2-party AND without computers

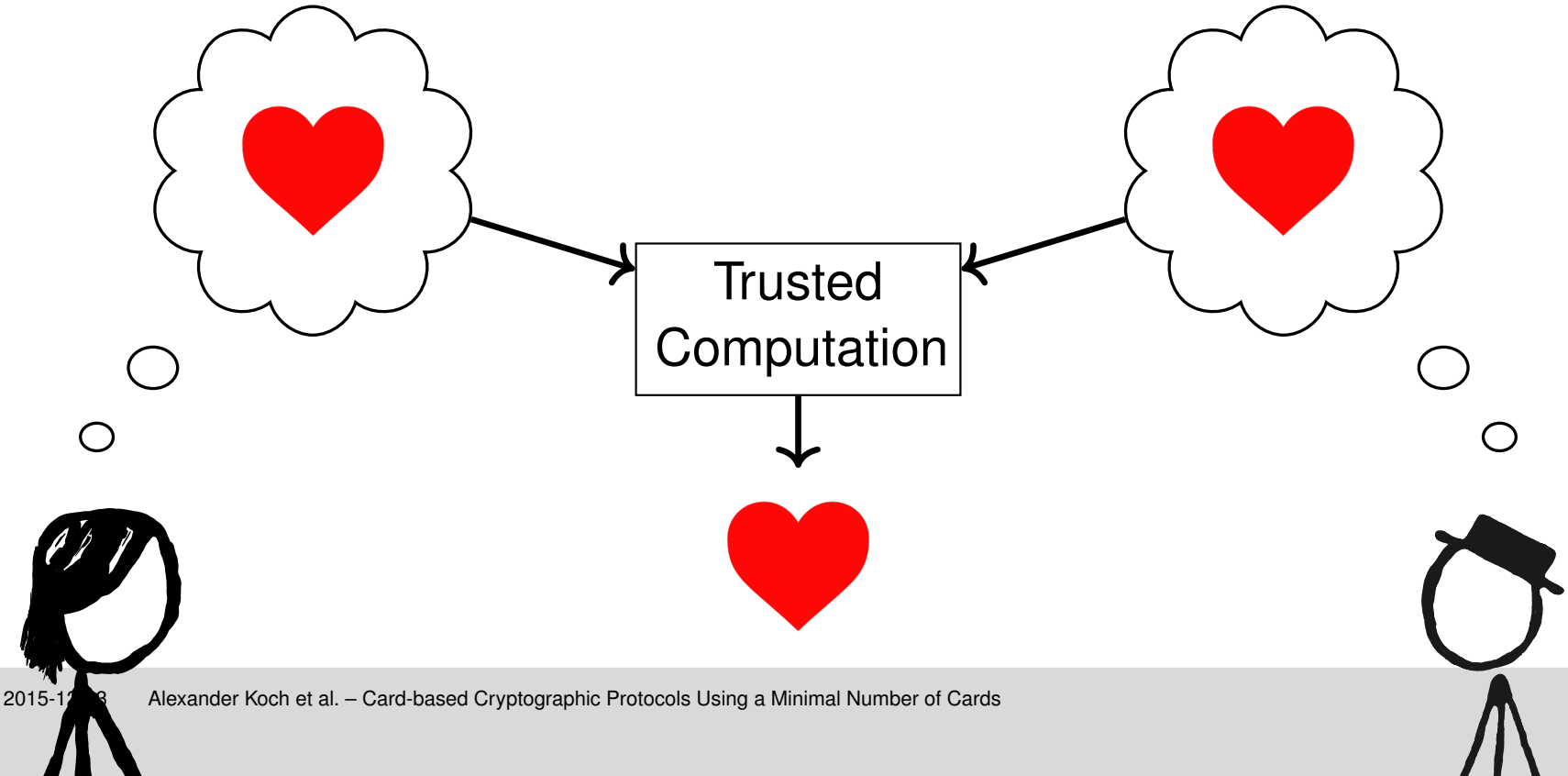


Motivating Scenario I

Calculating Mutual Interest with Playing Cards

- Secrets: Do I fancy him/her?
- To compute: Is there mutual interest?

⇝ Secure 2-party AND without computers



Motivating Scenario II

Explaining MPC to Non-Experts/Students

- You meet s.o. at a bar and want to explain MPC as an example from your work life.

Motivating Scenario II

Explaining MPC to Non-Experts/Students

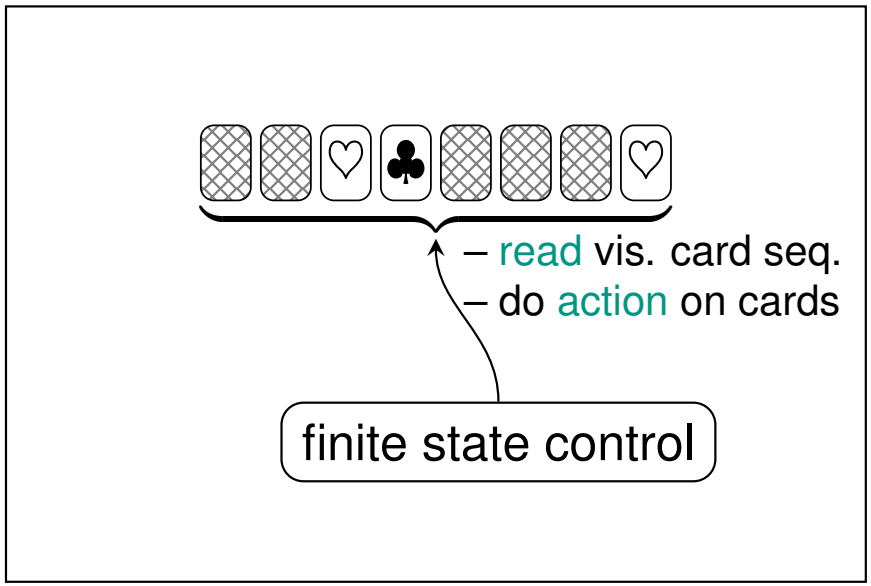
- You meet s.o. at a bar and want to explain MPC as an example from your work life. Or to students in class

Motivating Scenario III

You are a theoretician

- What is possible with unconventional computational models?
- MPC from indistinguishability of cards & correct shuffling

⇝ cf. to physical assumptions like tamper-proofness of hardware



Setting and Goal

Two types of indistinguishable cards:

Heart ♡ and club ♣ with backside ▨.

Encode bits as

- ♣ ♡ $\hat{=}$ 0
- ♡ ♣ $\hat{=}$ 1

Our goal (“committed format”)

- Take face-down input ▨ ▨ ▨ ▨ (bits a, b)
- Compute face-down output ▨ ▨ ($a \wedge b$)
- Learn nothing about the input or output during protocol run.

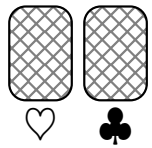
Setting and Goal

Two types of indistinguishable cards:

Heart ♡ and club ♣ with backside ▨.

Encode bits as

- ♣ ♡ $\hat{=}$ 0
- ♡ ♣ $\hat{=}$ 1



Curiosity:
is perfectly hiding & binding

Our goal (“committed format”)

- Take face-down input ▨ ▨ (bits a, b)
- Compute face-down output ▨ (bits $a \wedge b$)
- Learn nothing about the input or output during protocol run.

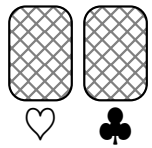
Setting and Goal

Two types of indistinguishable cards:

Heart ♡ and club ♣ with backside .

Encode bits as

- ♣ ♡ $\hat{=}$ 0
- ♡ ♣ $\hat{=}$ 1



Curiosity:
is perfectly hiding & binding

Our goal (“committed format”)

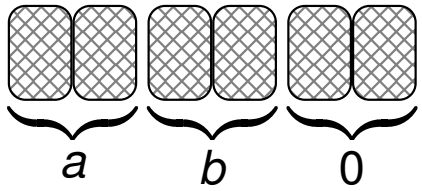
- Take face-down input (bits a, b)
- Compute face-down output ($a \wedge b$)
- Learn nothing about the input or output during protocol run.

A Simple Six-Card AND Protocol

Mizuki and Sone [MS09]

Observation: $(a \wedge b) \equiv (\text{if} \quad a \text{ then } b \text{ else } 0)$

The Protocol:



- Turn first two cards

$\rightsquigarrow$ result is cards 3, 4

$\rightsquigarrow$ result is cards 5, 6

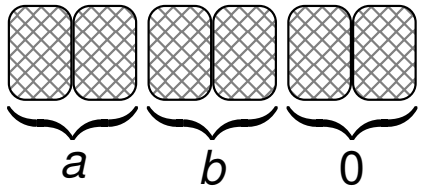
A Simple Six-Card AND Protocol

Mizuki and Sone [MS09]

Observation:

$$(a \wedge b) \equiv (\text{if } \quad a \text{ then } b \text{ else } 0)$$
$$\equiv (\text{if } \neg a \text{ then } 0 \text{ else } b)$$

The Protocol:



- Turn first two cards


$\hat{=} 1$ $\rightsquigarrow$ result is cards 3, 4


$\hat{=} 0$ $\rightsquigarrow$ result is cards 5, 6

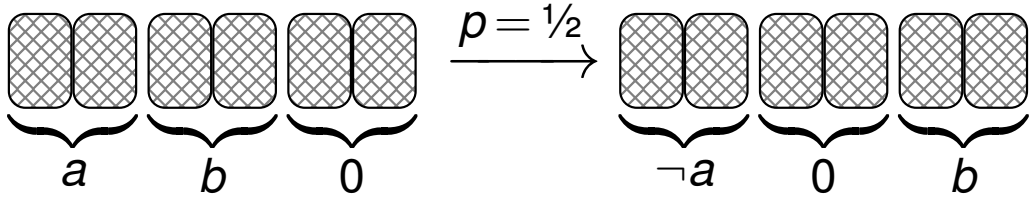
A Simple Six-Card AND Protocol

Mizuki and Sone [MS09]

Observation:

$$(a \wedge b) \equiv (\text{if } \quad a \text{ then } b \text{ else } 0)$$
$$\equiv (\text{if } \neg a \text{ then } 0 \text{ else } b)$$

The Protocol:



- With probability ½: Apply permutation $(1\ 2)(3\ 5)(4\ 6)$.
- Turn first two cards

$\rightsquigarrow$ result is cards 3, 4

$\rightsquigarrow$ result is cards 5, 6

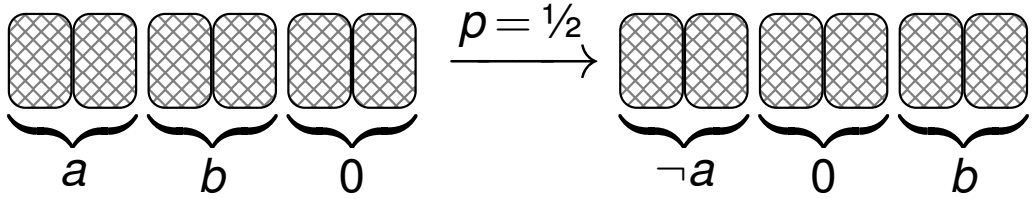
A Simple Six-Card AND Protocol

Mizuki and Sone [MS09]

Observation:

$$(a \wedge b) \equiv (\text{if } \quad a \text{ then } b \text{ else } 0)$$
$$\equiv (\text{if } \neg a \text{ then } 0 \text{ else } b)$$

The Protocol:



- With probability ½: Apply permutation $(1\ 2)(3\ 5)(4\ 6)$.
- For privacy: each player once, without the other looking.
- Turn first two cards


$\hat{=} 1$ $\rightsquigarrow$ result is cards 3, 4

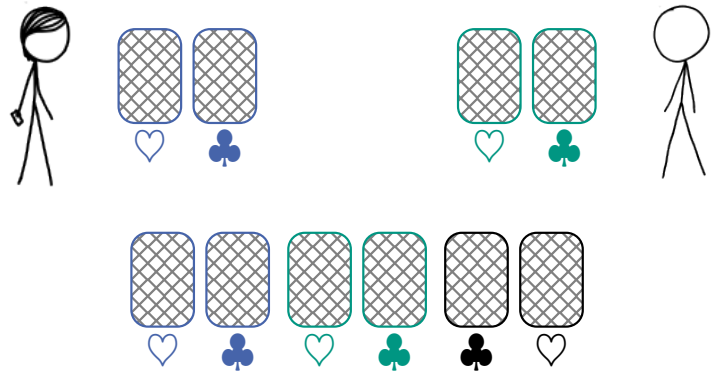

$\hat{=} 0$ $\rightsquigarrow$ result is cards 5, 6

Demonstration of the Six-Card-Protocol

Mizuki and Sone [MS09]

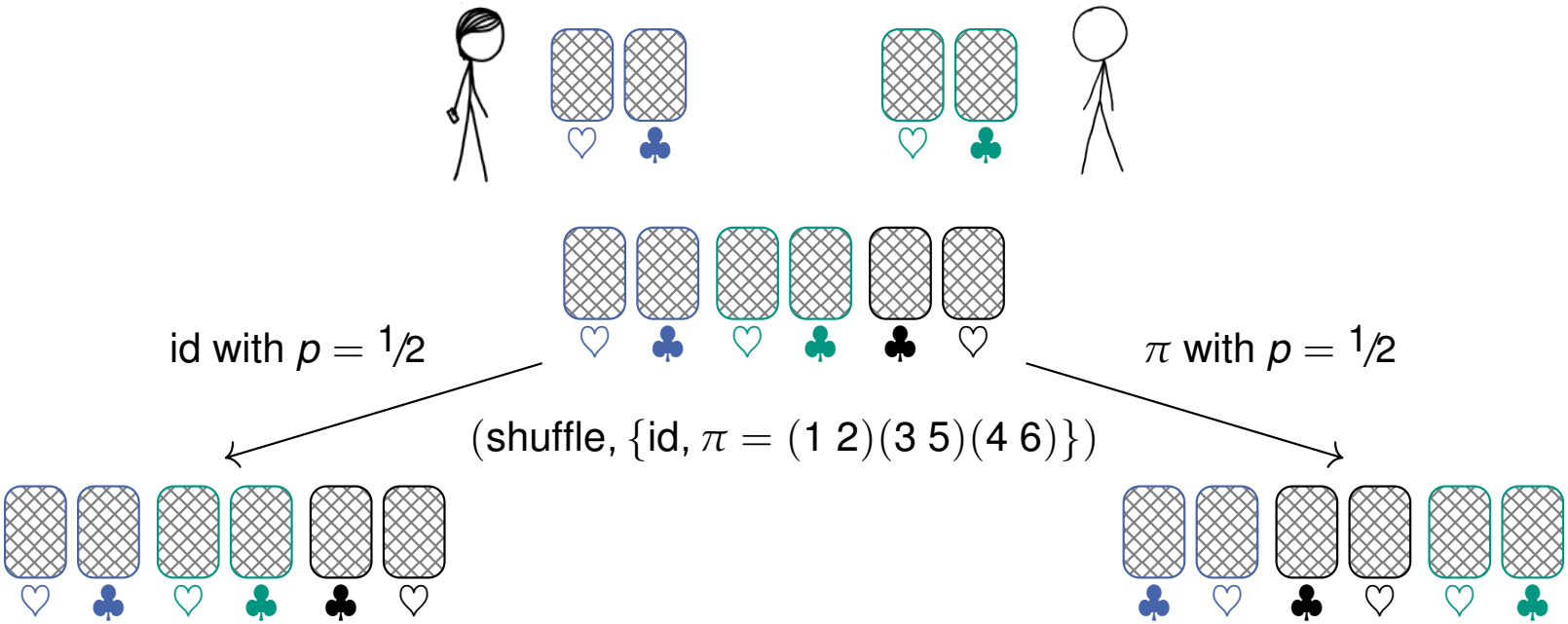
Demonstration of the Six-Card-Protocol

Mizuki and Sone [MS09]



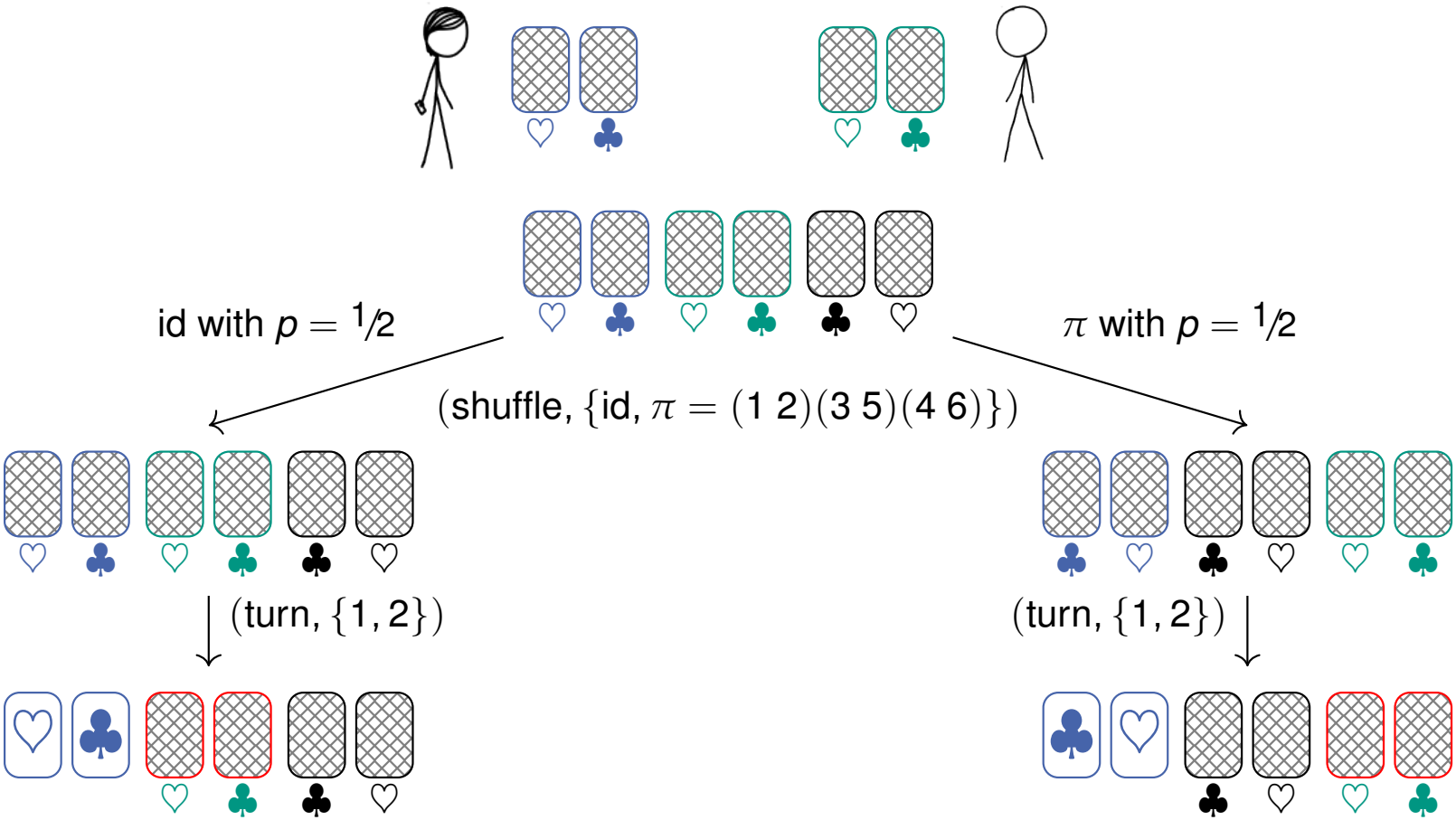
Demonstration of the Six-Card-Protocol

Mizuki and Sone [MS09]



Demonstration of the Six-Card-Protocol

Mizuki and Sone [MS09]



Can we do better than six cards?

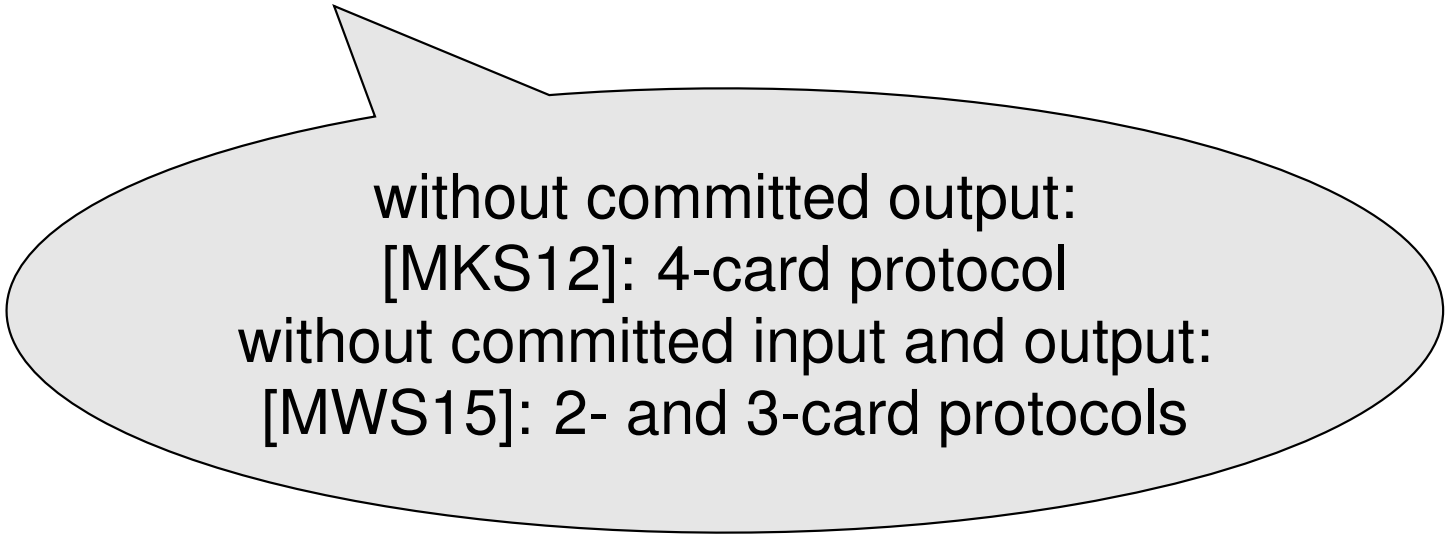
Open problem from [MS09; MS14; MKS12]

Main Question: Can $a \wedge b$ be computed with 4 cards?
in committed format (in the model of Mizuki and Shizuya [MS14])

Can we do better than six cards?

Open problem from [MS09; MS14; MKS12]

Main Question: Can $a \wedge b$ be computed with 4 cards?
in committed format (in the model of Mizuki and Shizuya [MS14])



Can we do better than six cards?

Open problem from [MS09; MS14; MKS12]

Main Question: Can $a \wedge b$ be computed with 4 cards?
in committed format (in the model of Mizuki and Shizuya [MS14])

Our Results

1. Yes, 4 cards suffice...
2. But 4-card protocols are necessarily Las Vegas (LV)
   - no a priori bound on runtime
   - method: analyze "states" of protocols
3. Yes, 5 cards suffice for finite-runtime protocols
4. LV protocol for k-ary functions using $2k$ cards
5. Note: "Complex" Shuffles needed.

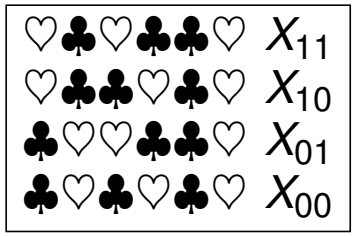
State Transitions: The Six-Card Protocol

Karlsruhe Institute of Technology

Protocol State:

Annotate currently possible sequences with probability in terms of symbolic input prob.

$X_{ij} = \Pr[a = i, b = j]$



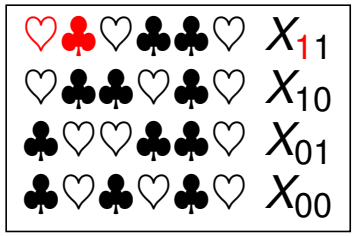
State Transitions: The Six-Card Protocol

Karlsruhe Institute of Technology

Protocol State:

Annotate currently possible sequences with probability in terms of symbolic input prob.

$X_{ij} = \Pr[a = i, b = j]$



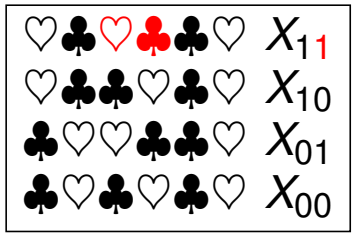
State Transitions: The Six-Card Protocol

Karlsruhe Institute of Technology

Protocol State:

Annotate currently possible sequences with probability in terms of symbolic input prob.

$X_{ij} = \Pr[a = i, b = j]$

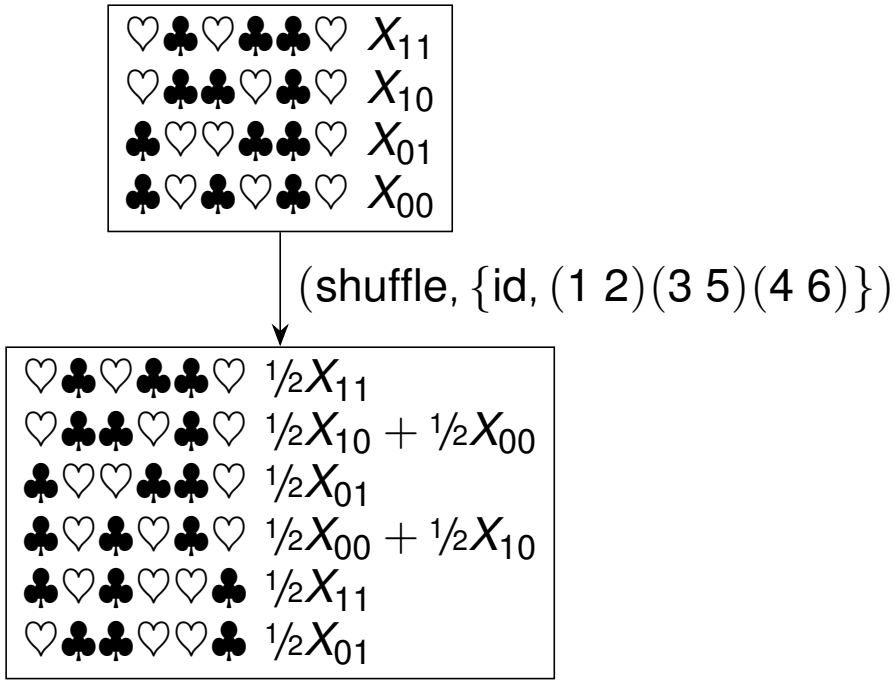


State Transitions: The Six-Card Protocol

Karlsruhe Institute of Technology

Protocol State:

Annotate currently possible sequences with probability in terms of symbolic input prob.
$X_{ij} = \Pr[a = i, b = j]$



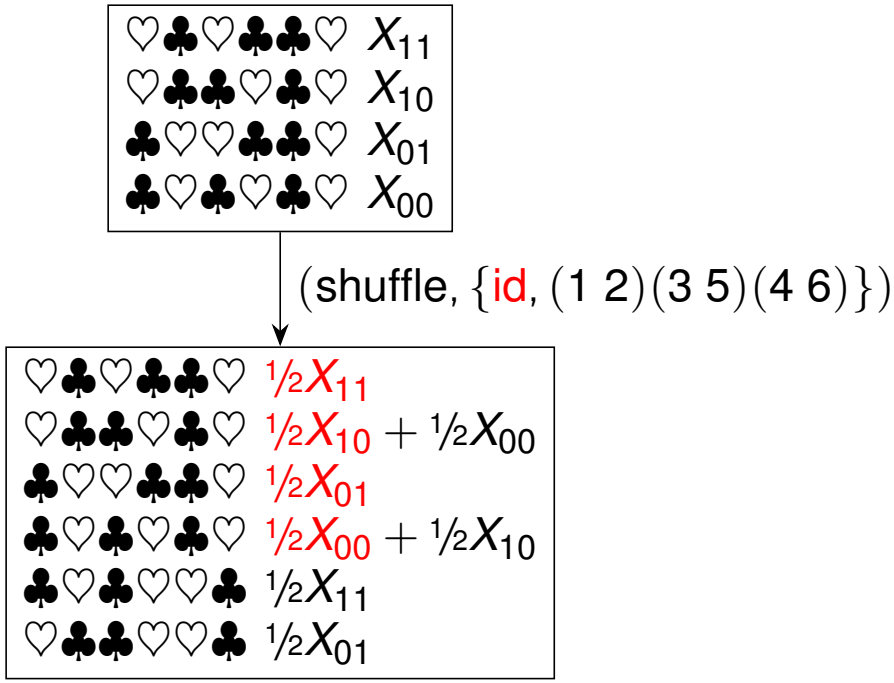
State Transitions: The Six-Card Protocol

Karlsruhe Institute of Technology

Protocol State:

Annotate currently possible sequences with probability in terms of symbolic input prob.

$X_{ij} = \Pr[a = i, b = j]$



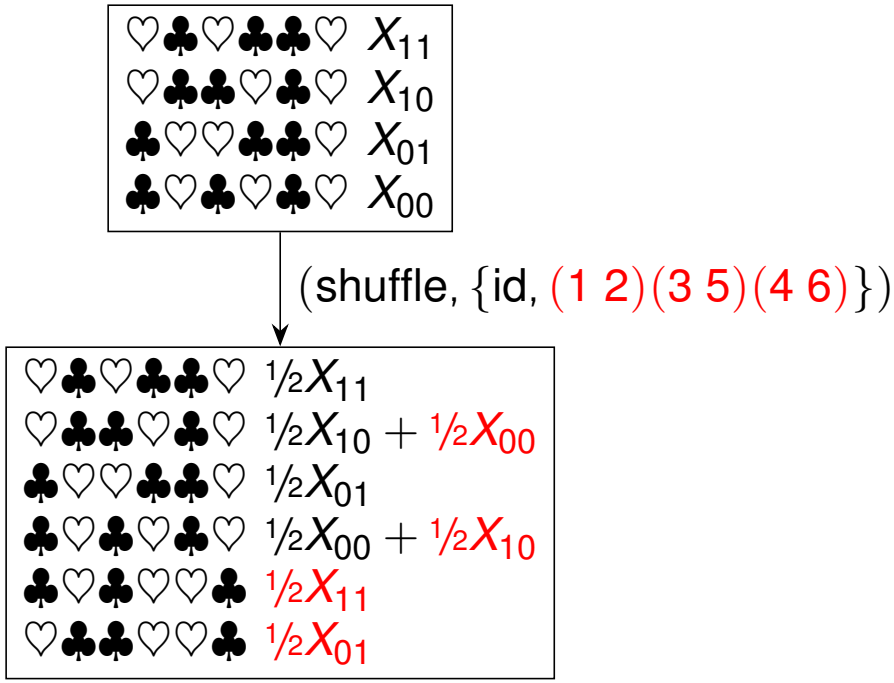
State Transitions: The Six-Card Protocol

Karlsruhe Institute of Technology

Protocol State:

Annotate currently possible sequences with probability in terms of symbolic input prob.

$X_{ij} = \Pr[a = i, b = j]$



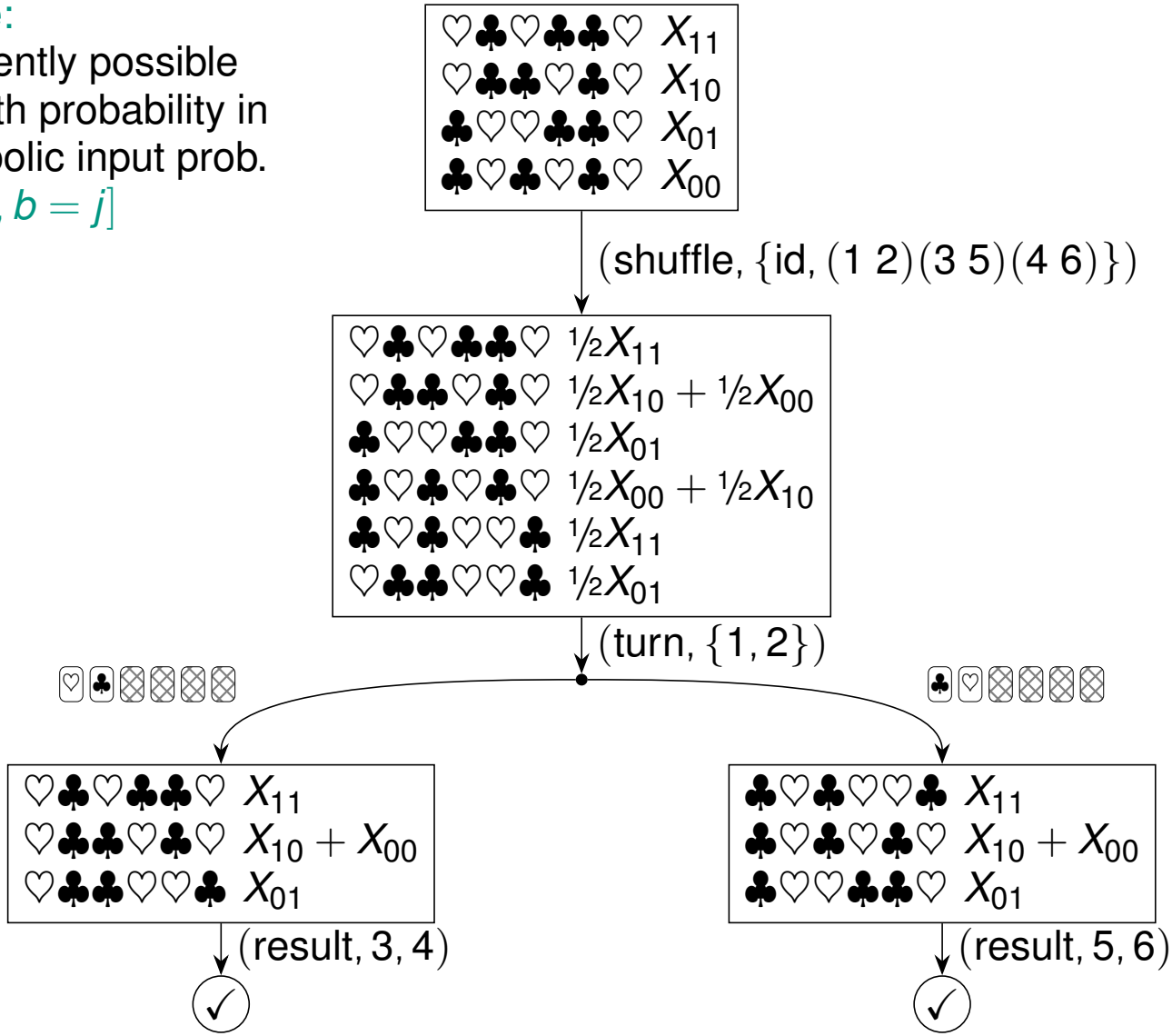
State Transitions: The Six-Card Protocol

Karlsruhe Institute of Technology

Protocol State:

Annotate currently possible sequences with probability in terms of symbolic input prob.

$X_{ij} = \Pr[a = i, b = j]$



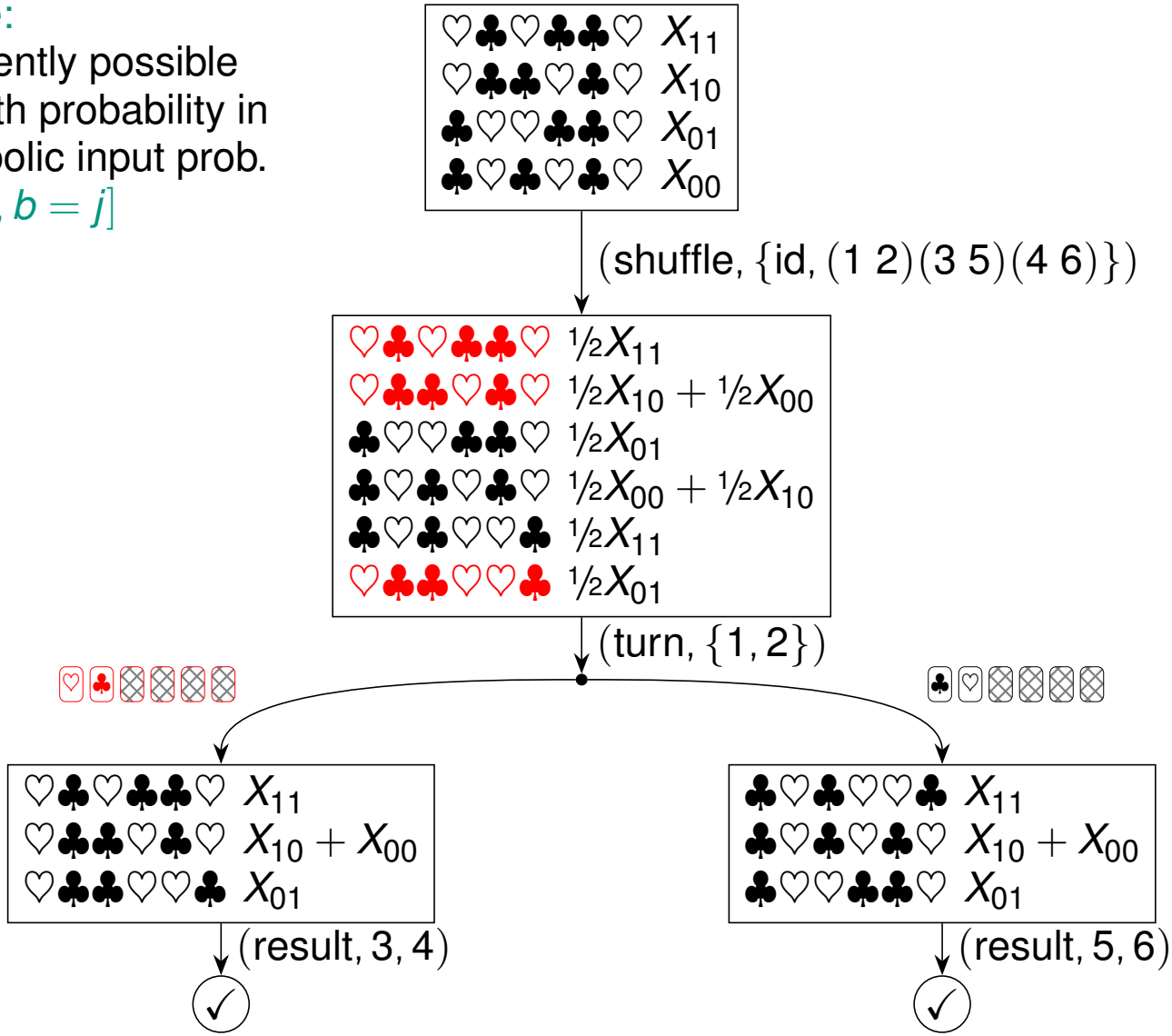
State Transitions: The Six-Card Protocol

Karlsruhe Institute of Technology

Protocol State:

Annotate currently possible sequences with probability in terms of symbolic input prob.

$X_{ij} = \Pr[a = i, b = j]$



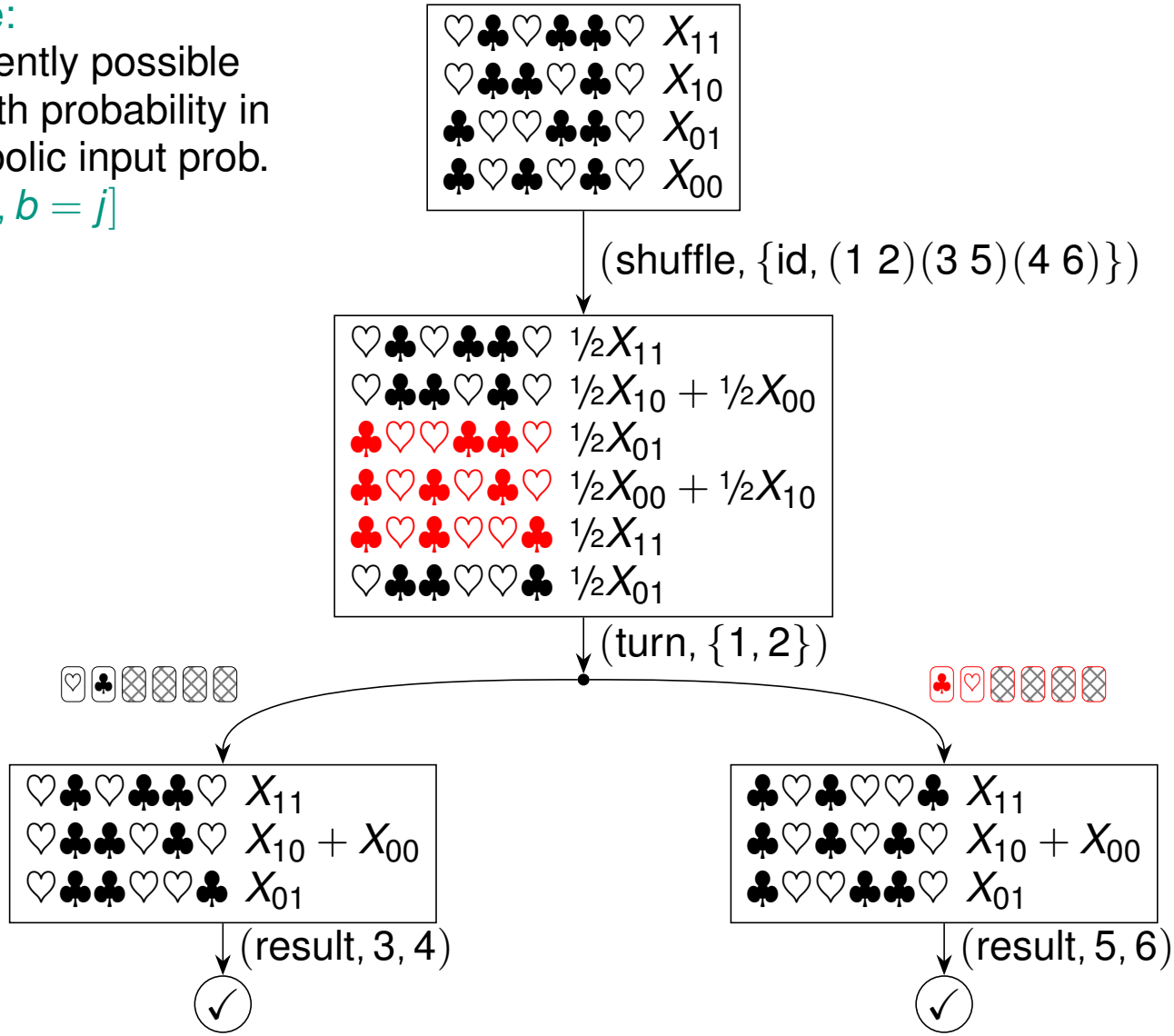
State Transitions: The Six-Card Protocol

Karlsruhe Institute of Technology

Protocol State:

Annotate currently possible sequences with probability in terms of symbolic input prob.

$X_{ij} = \Pr[a = i, b = j]$



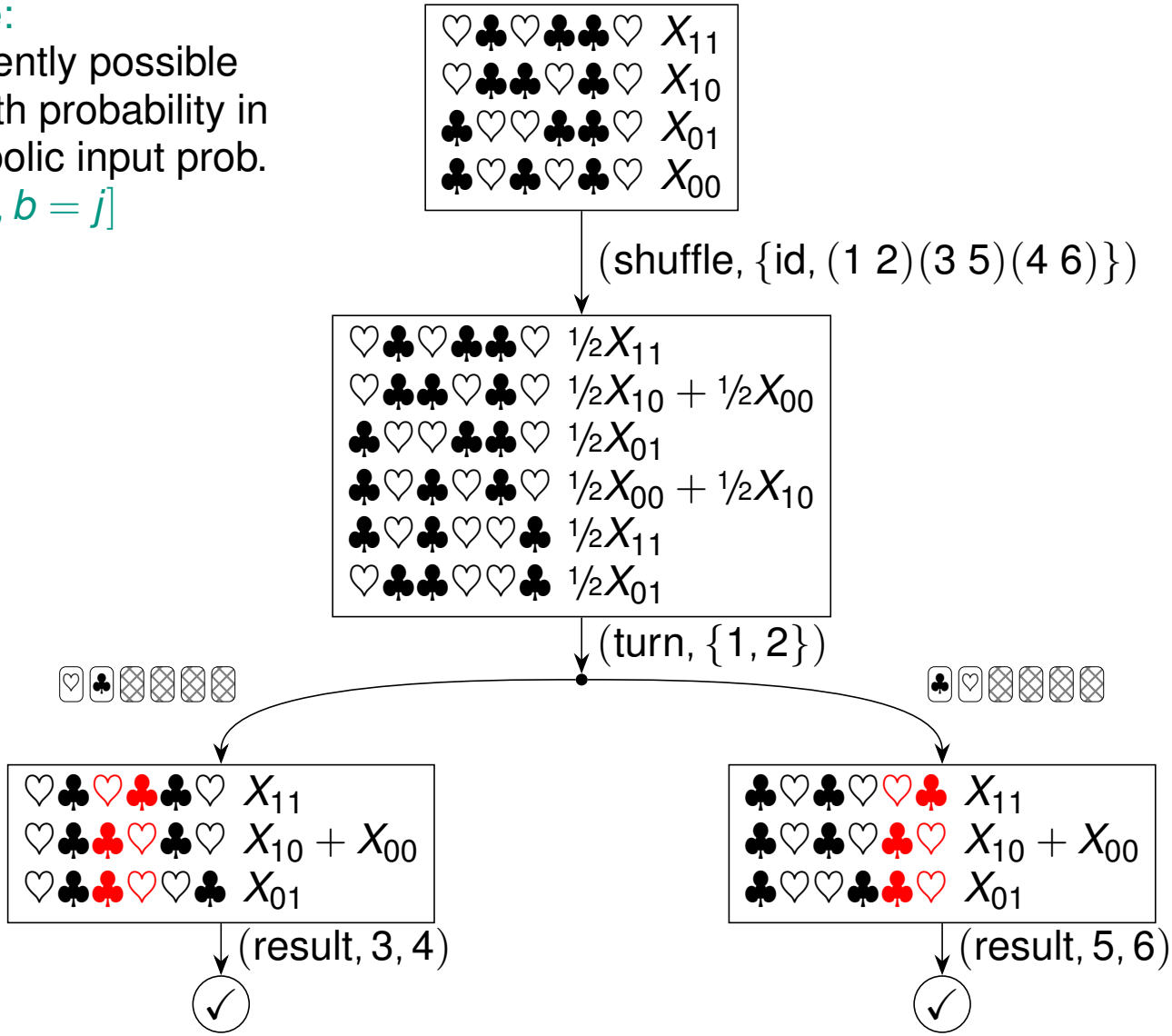
State Transitions: The Six-Card Protocol

Karlsruhe Institute of Technology

Protocol State:

Annotate currently possible sequences with probability in terms of symbolic input prob.

$X_{ij} = \Pr[a = i, b = j]$



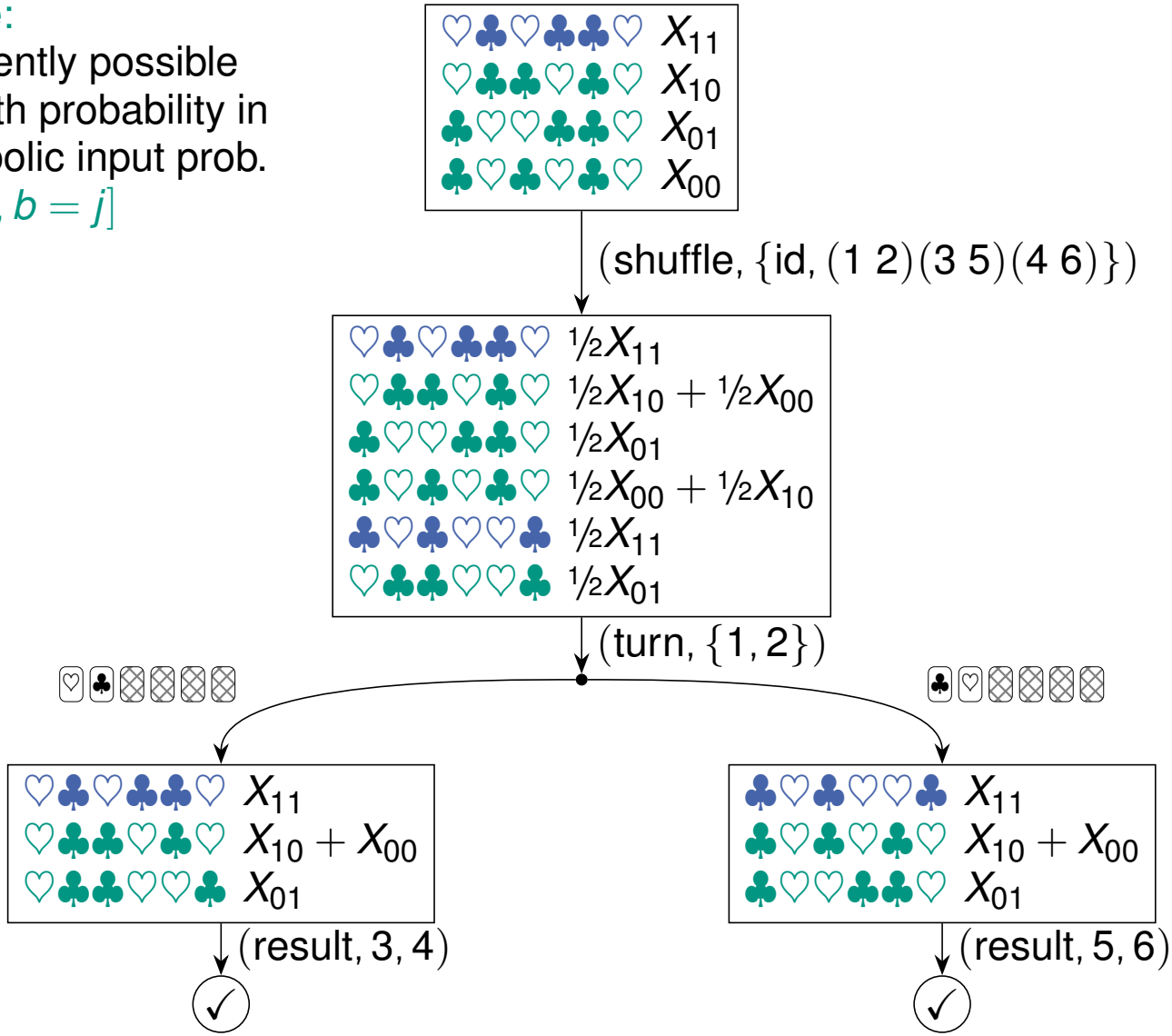
State Transitions: The Six-Card Protocol

Karlsruhe Institute of Technology

Protocol State:

Annotate currently possible sequences with probability in terms of symbolic input prob.

$X_{ij} = \Pr[a = i, b = j]$



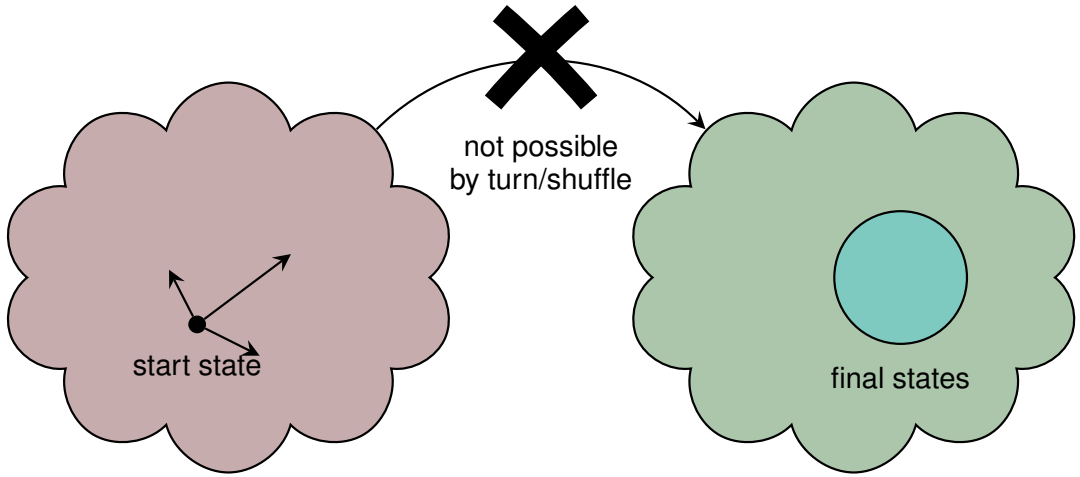
Impossibility Result

Theorem

There is no secure finite-runtime four-card AND protocol

Proof Idea

- Each sequence belongs either to output 0 or to 1.
- An $i|j$-state has i 0-sequences and j 1-sequences.
- Define non-reachable “good” states:

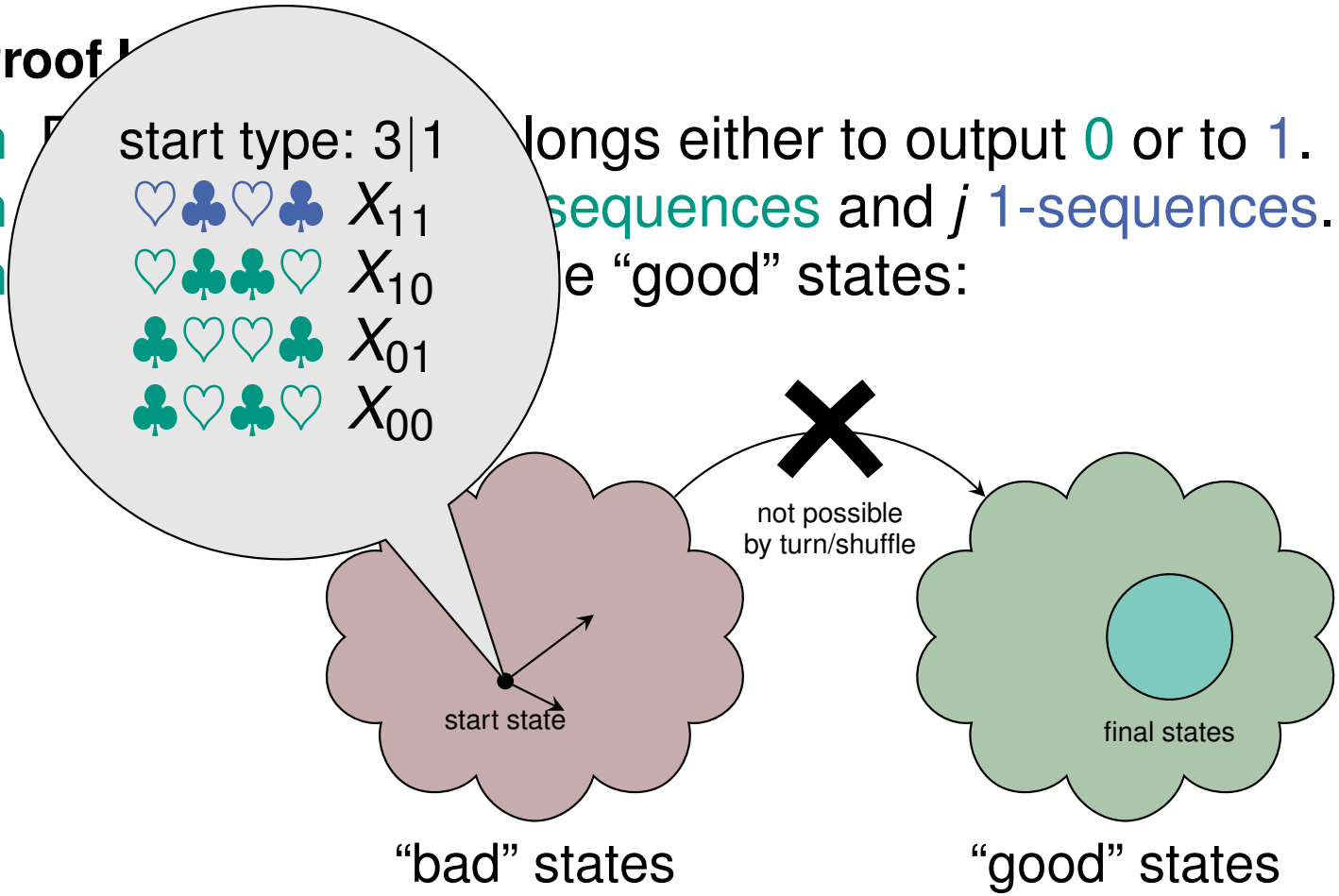


Impossibility Result

Theorem
There is no secure finite-runtime four-card AND protocol

Proof I

- P… longs either to output 0 or to 1.
- …sequences and *j* 1-sequences.
- …e "good" states:

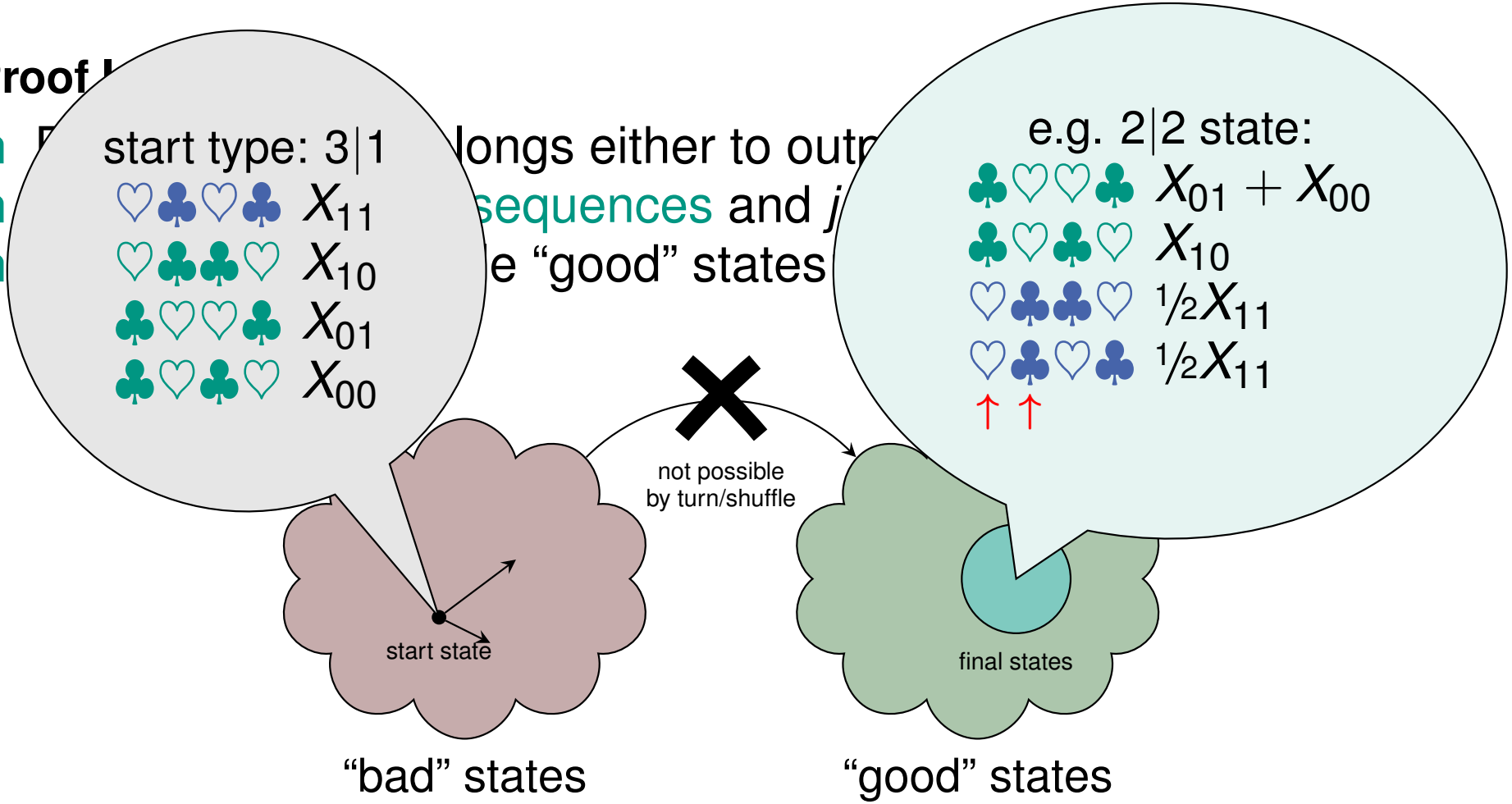


Impossibility Result

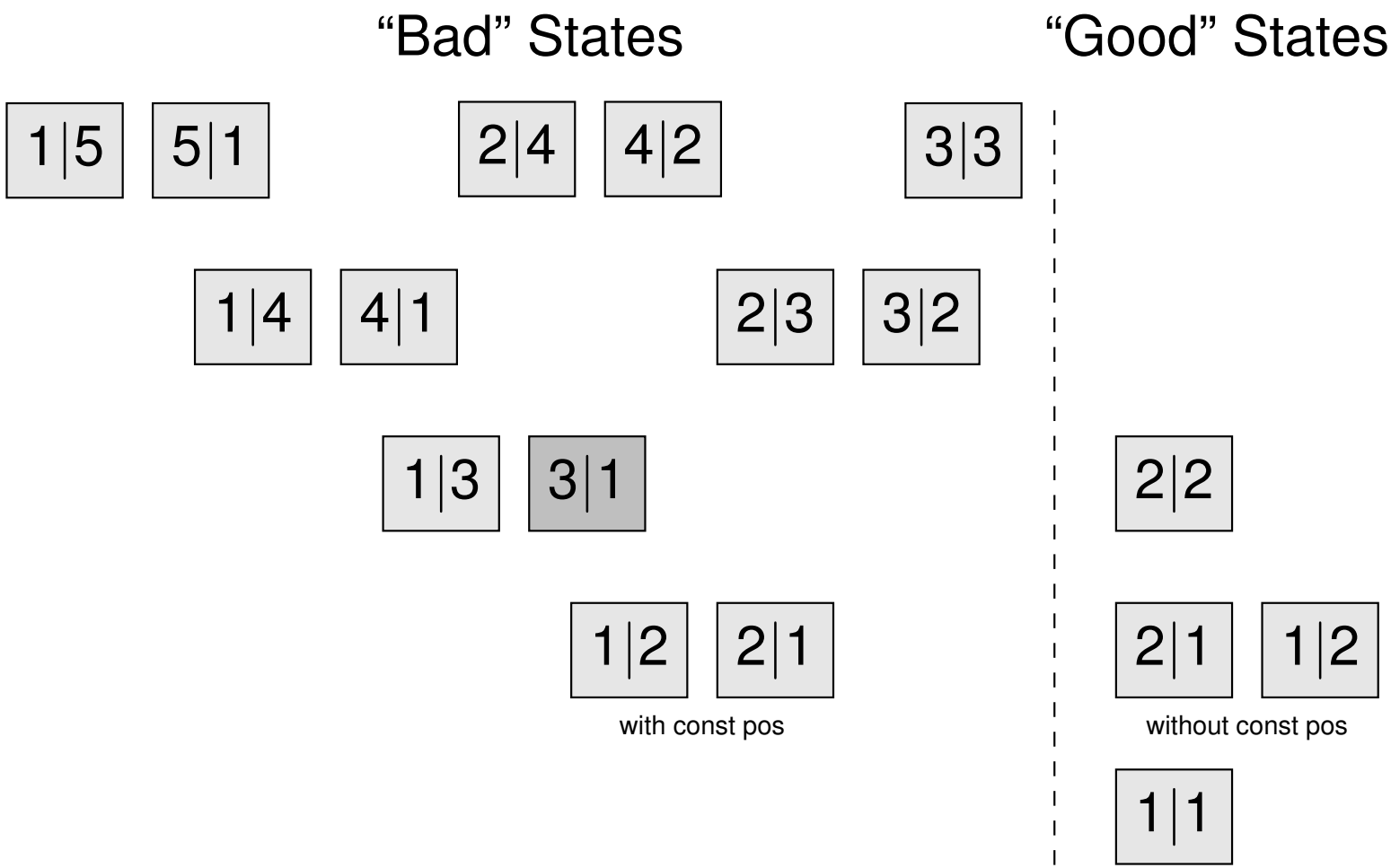
Theorem
There is no secure finite-runtime four-card AND protocol

Proof I

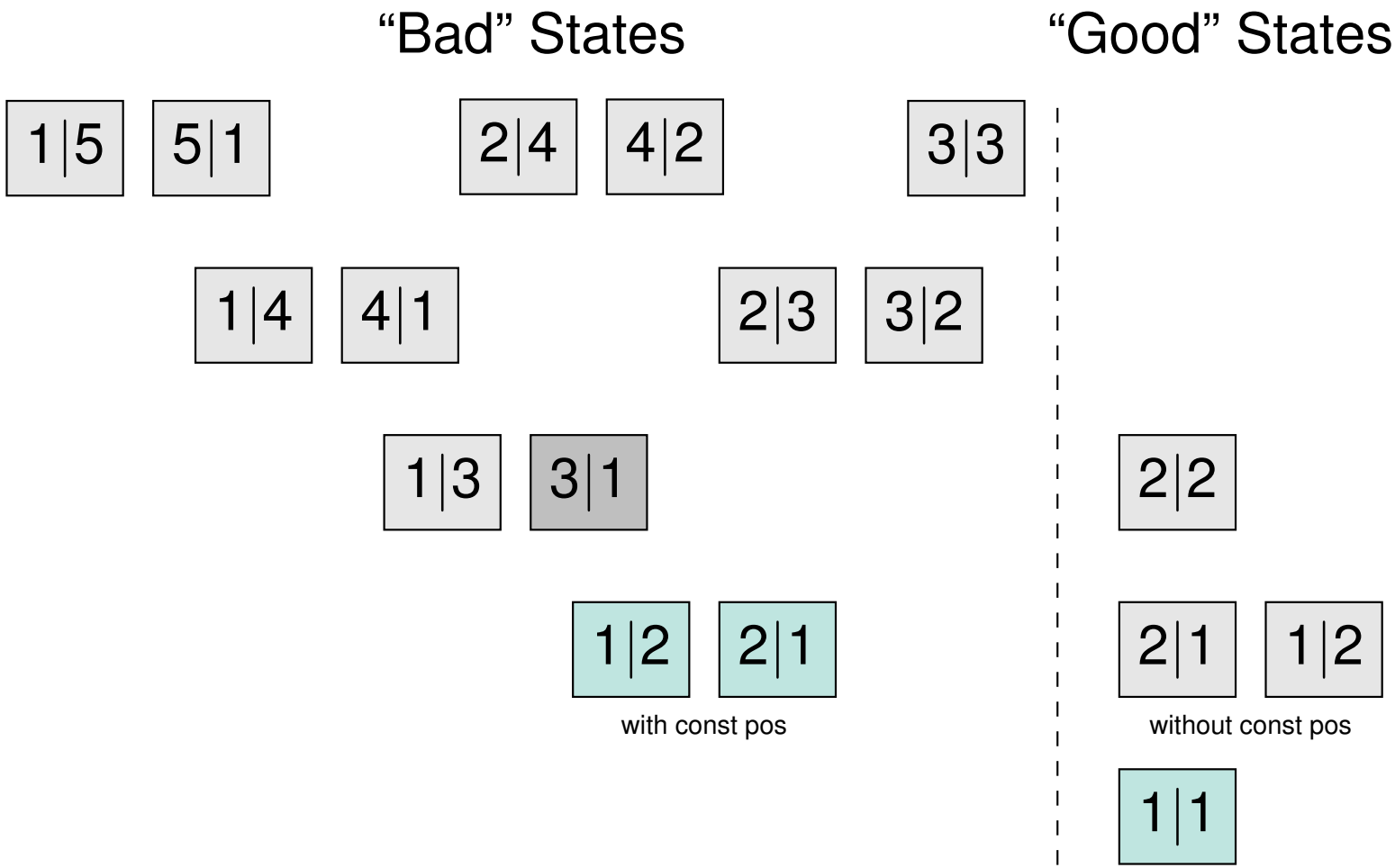
- P… longs either to outp…
- …sequences and *j*…
- …e "good" states



Proof Idea – Single Card Turns



Proof Idea – Single Card Turns



Observation 1. After turn: with const pos. and ≤ 3 sequences.

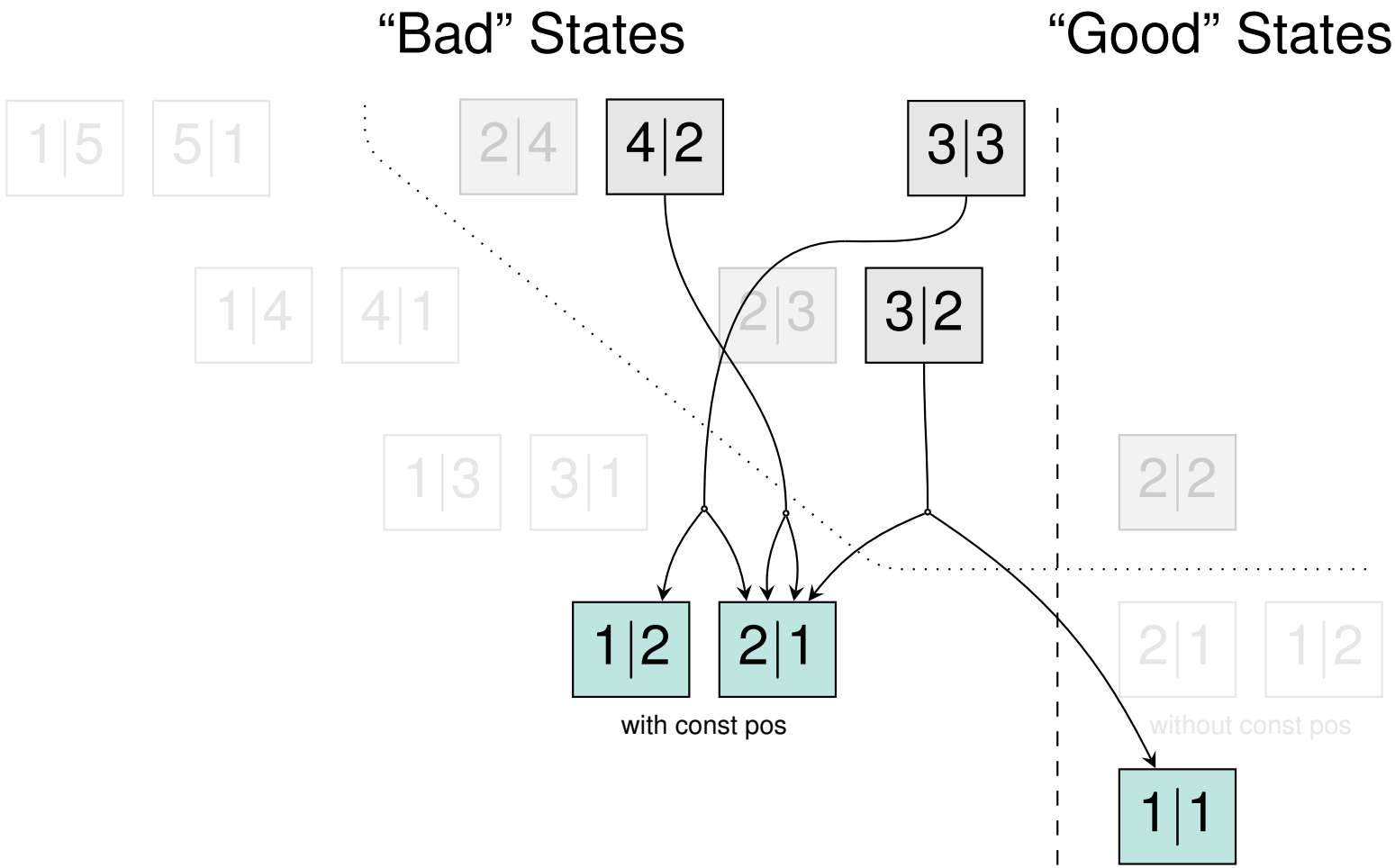
Proof Idea – Single Card Turns

Observation 2. Turnable states are $i|j$ with $i, j \geq 2$.

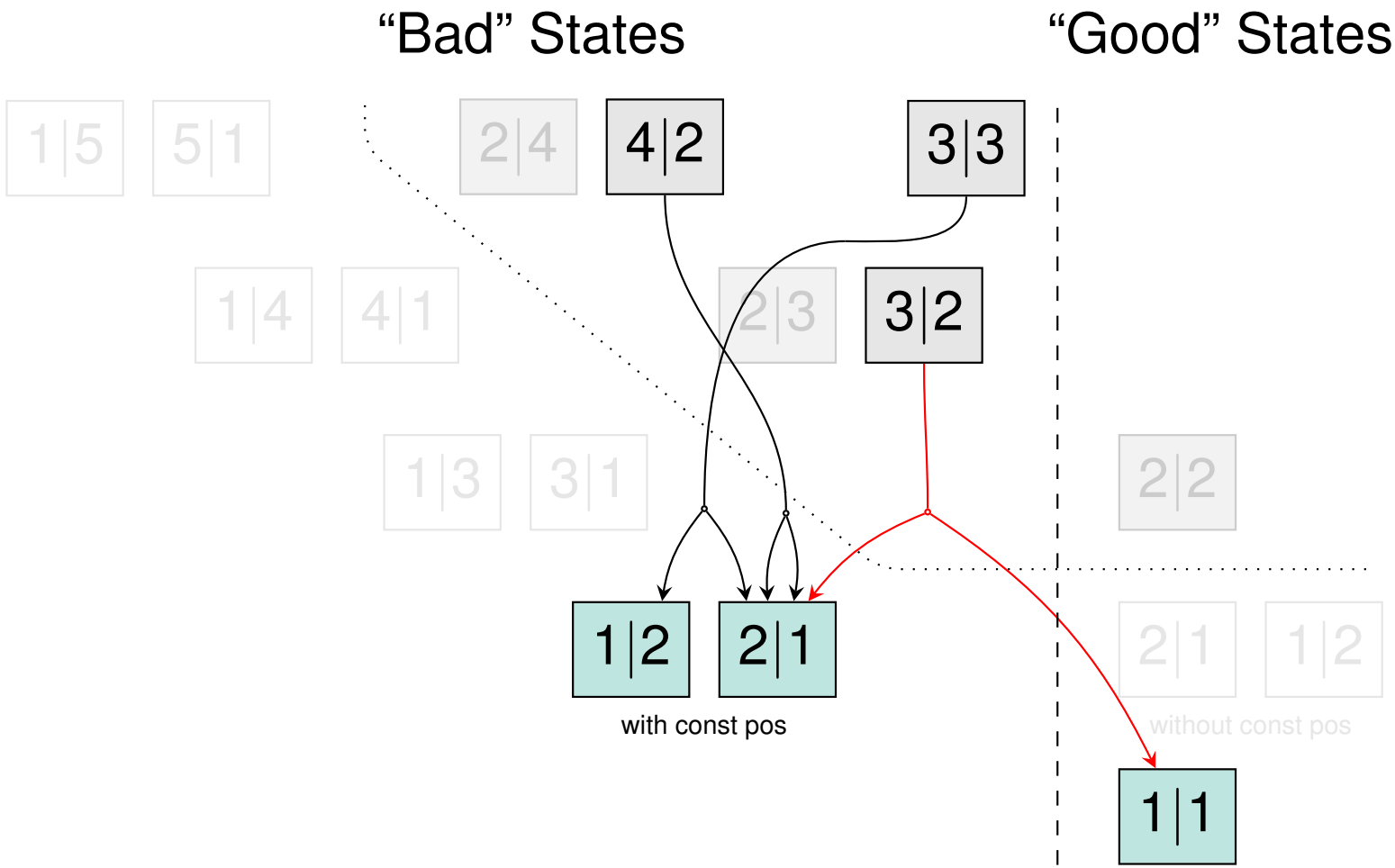
Proof Idea – Single Card Turns

Observation 3. W.l.o.g. we need to consider half of the states.

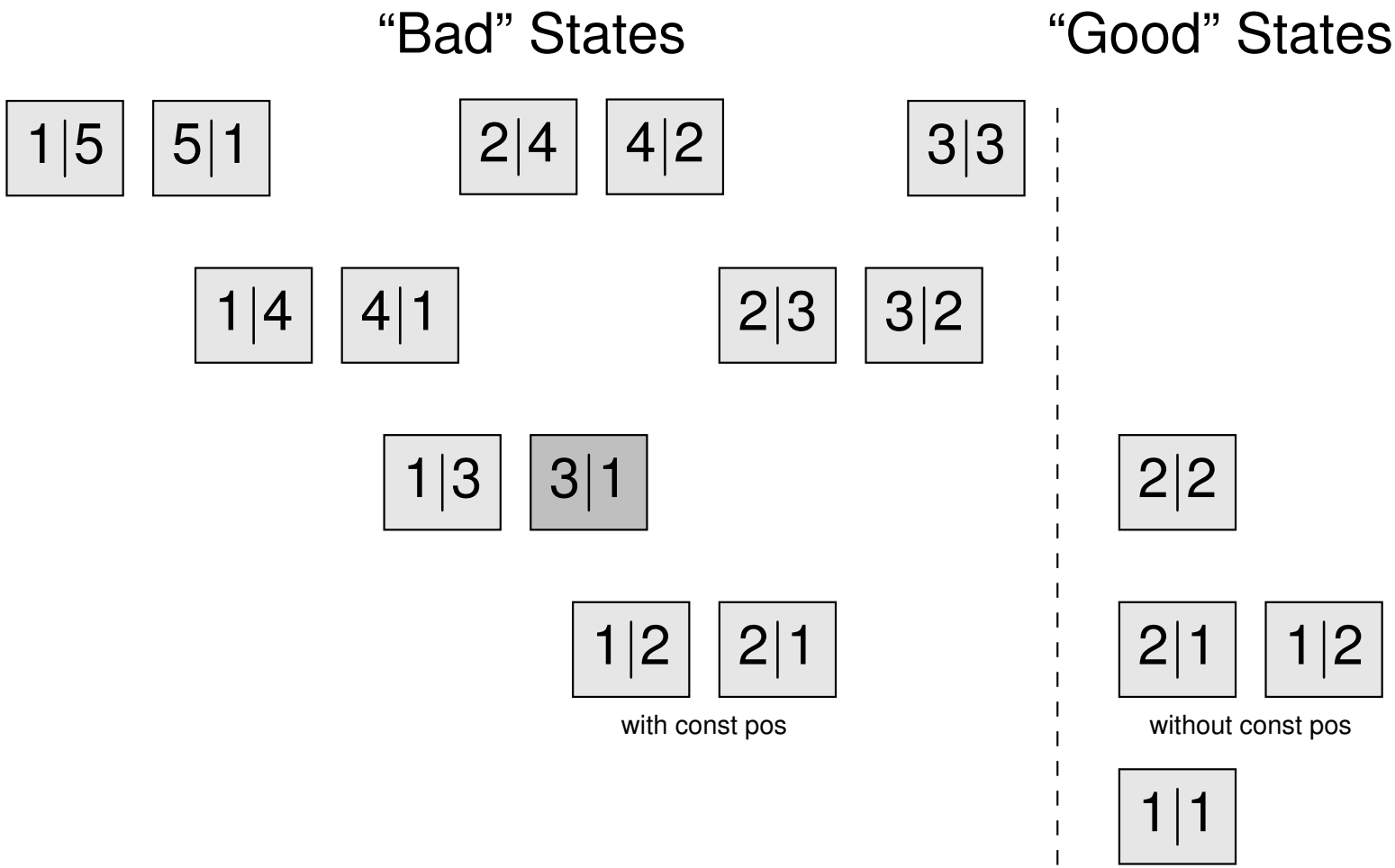
Proof Idea – Single Card Turns



Proof Idea – Single Card Turns



Proof Idea – Shuffles

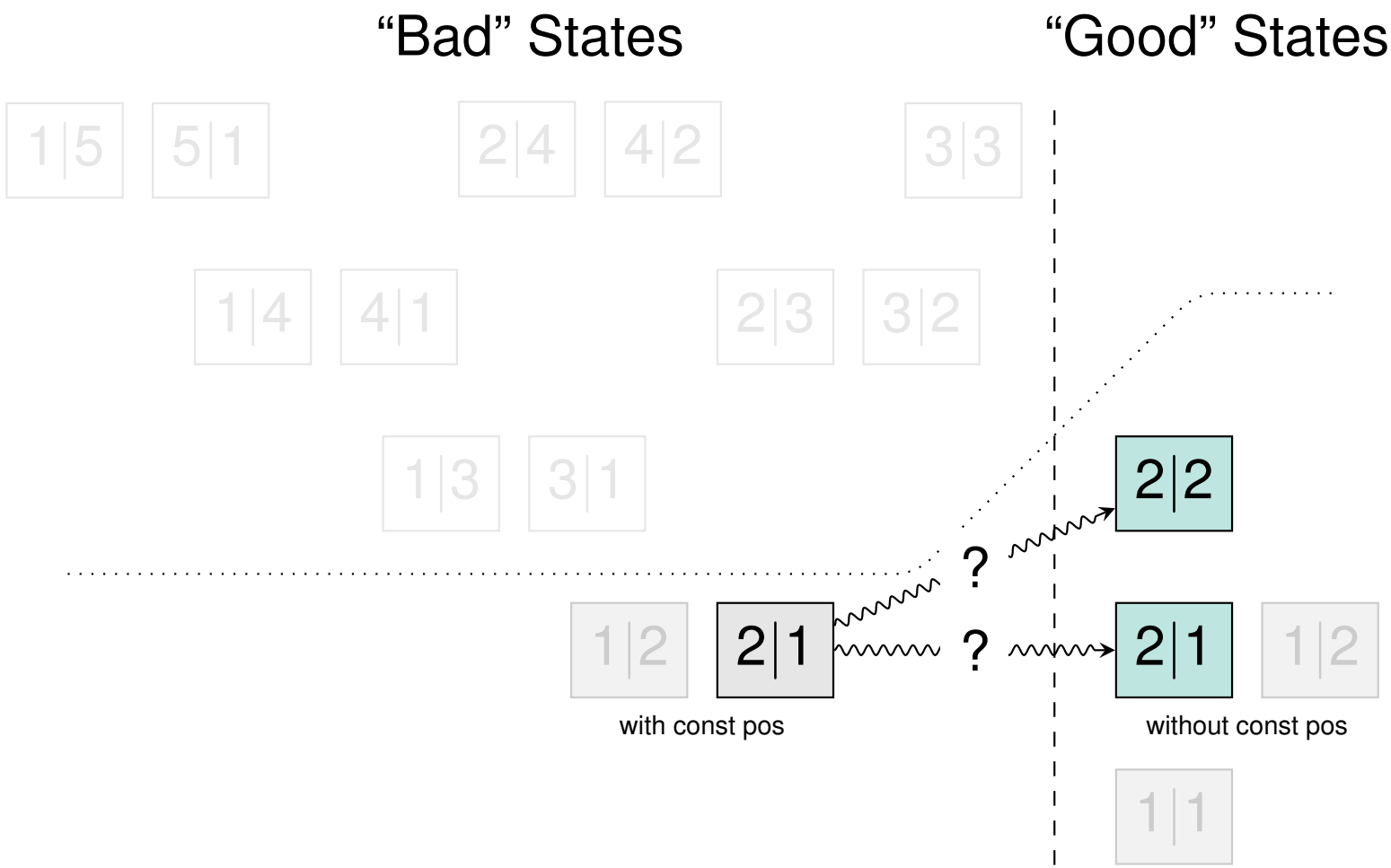


Observation 1. Shuffles increase #sequences per type

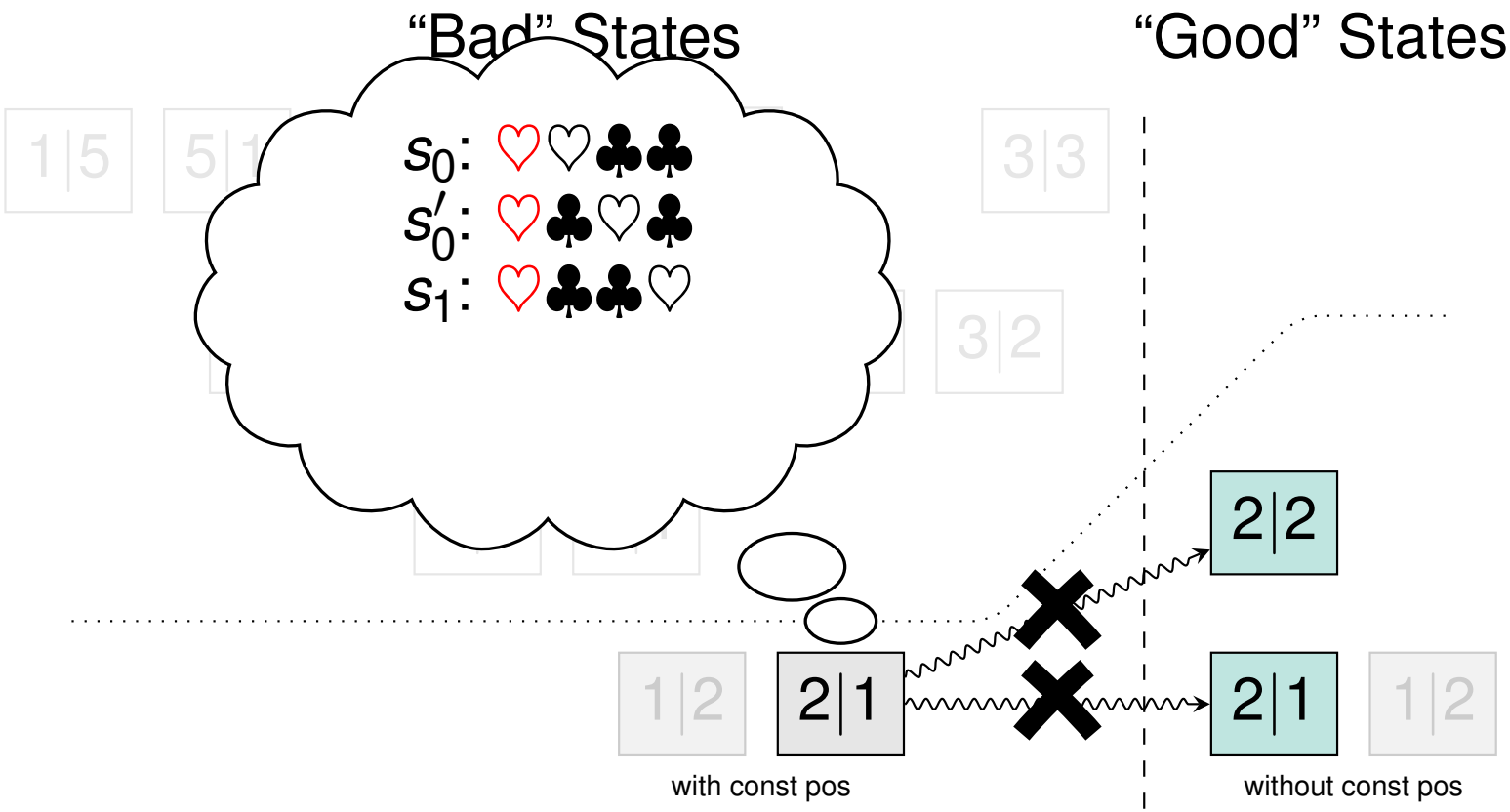
Proof Idea – Shuffles

Observation 1. Shuffles increase #sequences per type

Proof Idea – Shuffles

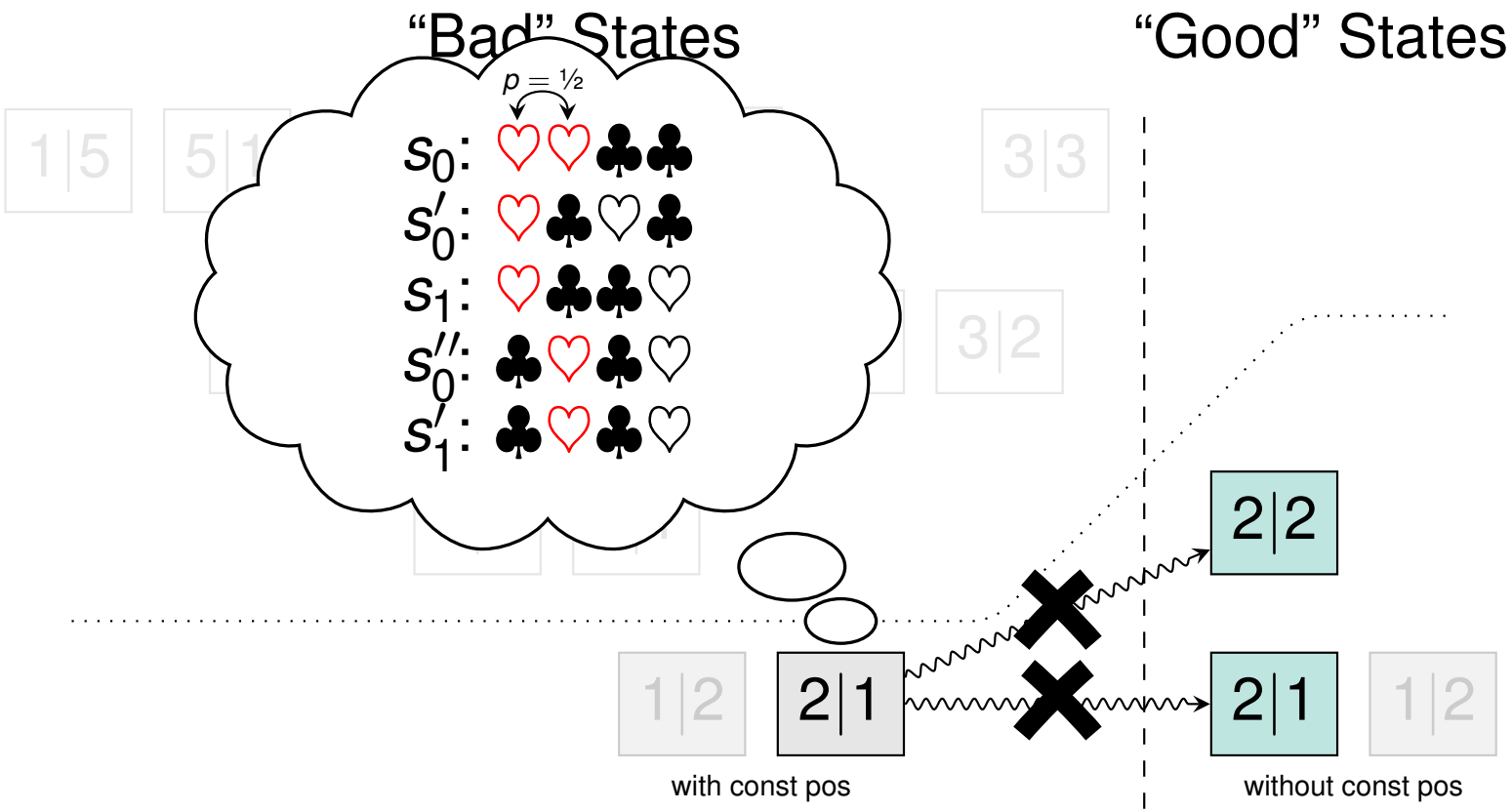


Proof Idea – Shuffles



Apply (shuffle, Π, $\mathcal{F}$) to this state.
Case 1: All $\pi \in \Pi$ put constant column to same position.
$\Longrightarrow$ the resulting state still has a constant column.

Proof Idea – Shuffles

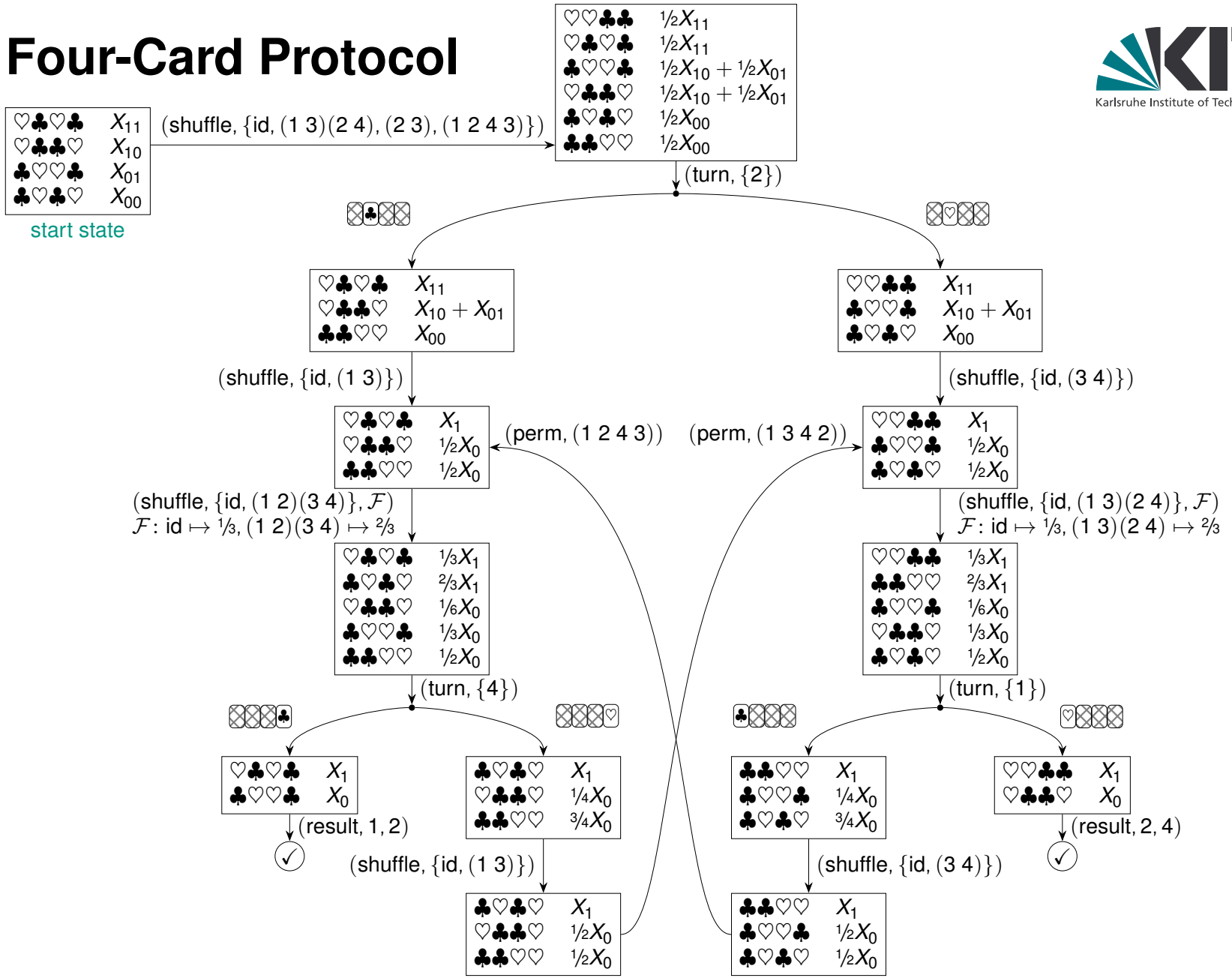


Apply (shuffle, Π, $\mathcal{F}$) to this state.

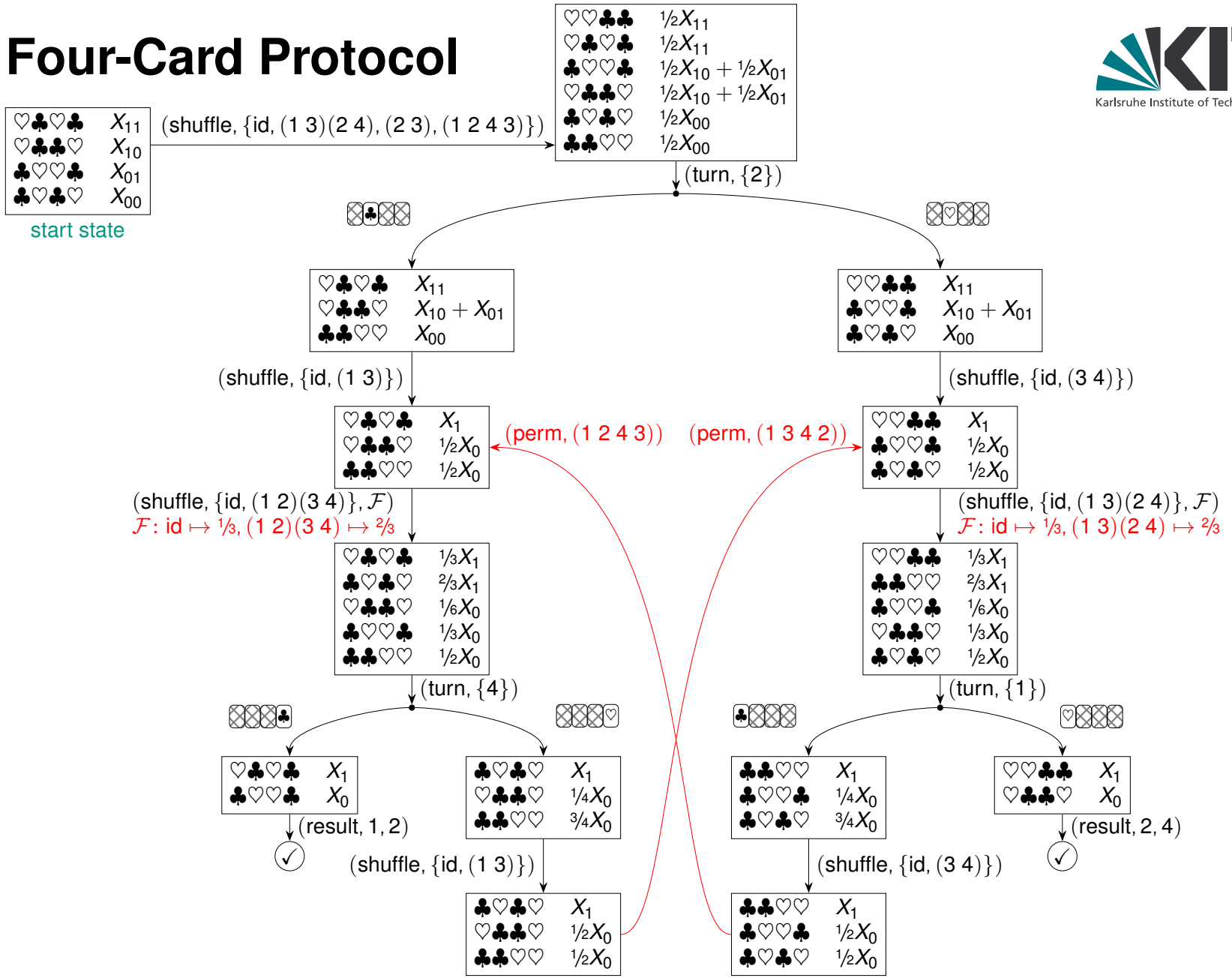
Case 2: There are $\pi_1, \pi_2 \in \Pi$ putting the const. col. in different pos.

$\Longrightarrow$ the resulting state has at least 5 sequences. □

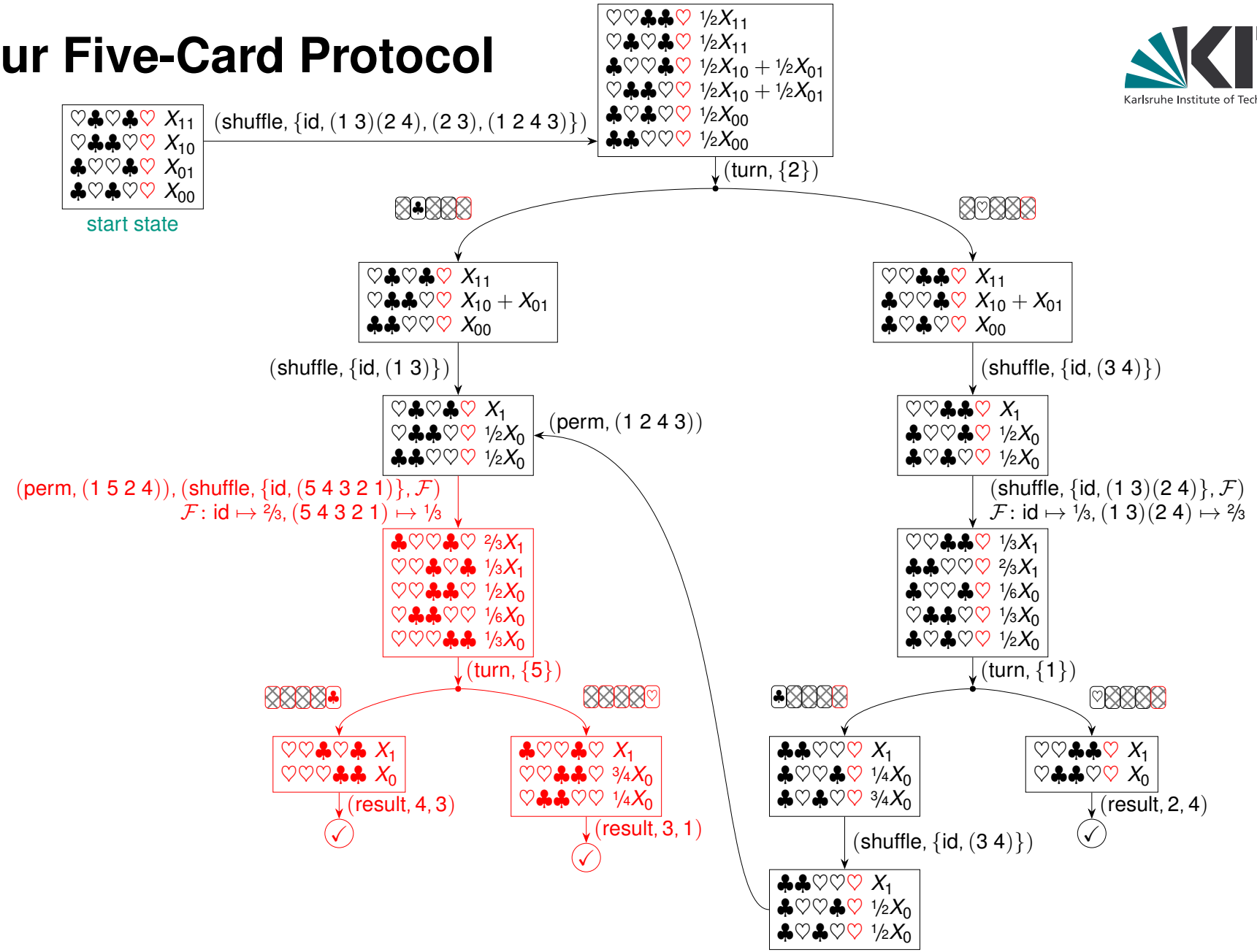
Our Four-Card Protocol



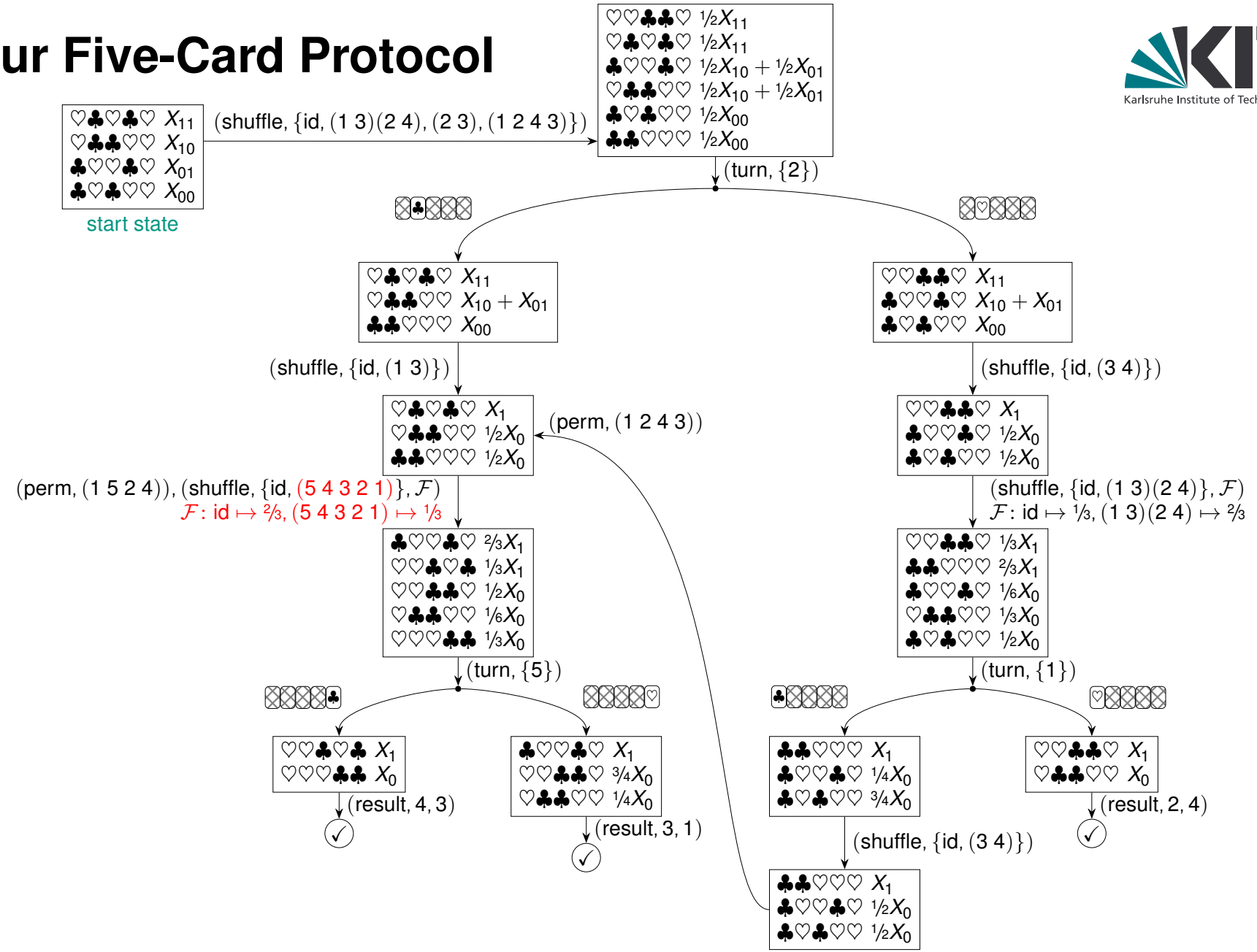
Our Four-Card Protocol



Our Five-Card Protocol



Our Five-Card Protocol



Summary

Runtime	Shuffles	#Cards	Reference
exp. finite	non-uniform closed	4	[KWH15]
exp. finite	uniform non-closed	4	[KWH15]
finite	non-uniform non-closed	5	[KWH15]
finite	uniform closed	≤ 6	[MS09]

Summary

Runtime	Shuffles	#Cards	Reference
exp. finite	non-uniform closed	4	[KWH15]
exp. finite	uniform non-closed	4	[KWH15]
finite	non-uniform non-closed	5	[KWH15]
finite	uniform closed	≤ 6	[MS09]

Open Question: What if we restrict the computational model?

Summary

Runtime	Shuffles	#Cards	Reference
exp. finite	non-uniform closed	4	[KWH15]
exp. finite	uniform non-closed	4	[KWH15]
finite	non-uniform non-closed	5	[KWH15]
finite	uniform closed	≤ 6	[MS09]

Open Question: What if we restrict the computational model?

Thank you for your attention!

References: I

A. Koch, S. Walzer, and K. Härtel. “Card-based Cryptographic Protocols Using a Minimal Number of Cards”. In: ASIACRYPT 2015. Ed. by T. Iwata and J. Cheon. Vol. 9452. LNCS. Springer, 2015, pp. 783–807.

T. Mizuki, M. Kumamoto, and H. Sone. “The Five-Card Trick Can Be Done with Four Cards”. In: ASIACRYPT 2012. Ed. by X. Wang and K. Sako. Vol. 7658. LNCS. Springer, 2012, pp. 598–606.

T. Mizuki and H. Sone. “Six-Card Secure AND and Four-Card Secure XOR”. In: FAW 2009. Ed. by X. Deng, J. E. Hopcroft, and J. Xue. Vol. 5598. LNCS. Springer, 2009, pp. 358–369.

References: II

T. Mizuki and H. Shizuya. “A formalization of card-based cryptographic protocols via abstract machine”. In: Int. J. Inf. Secur. 13.1 (2014), pp. 15–23.

A. Marcedone, Z. Wen, and E. Shi. Secure Dating with Four or Fewer Cards. Cryptology ePrint Archive, Report 2015/1031. https://eprint.iacr.org/. 2015.

References: III

Various Artists. Title image from http://pdpics.com/photo/6619-ten-cards-of-all-suits/, public domain. Image of Bar from https://pixabay.com/en/bar-pub-restaurant-rustic-barrels-406884/, public domain. Image of lecture hall from brett jordan, https://www.flickr.com/photos/x1brett/1472187414, CC-BY-2.0. XKCD comic figures by Randall Munroe from https://xkcd.com/, CC-BY-NC-2.5.