Secure bit commitment from relativistic constraints

Jed Kaniewski

Centre for Quantum Technologies
National University of Singapore

joint work with Marco Tomamichel, Esther Hänggi and Stephanie Wehner

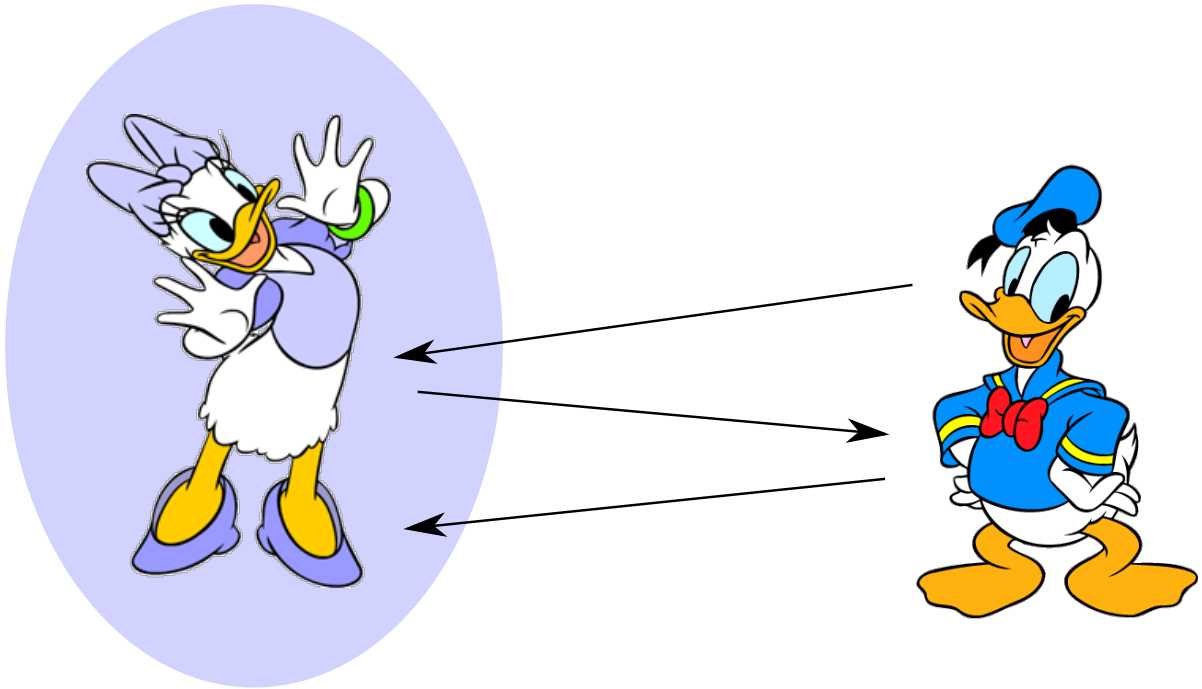
[arXiv: 1206.1740]

QCrypt 2012, Singapore

Outline

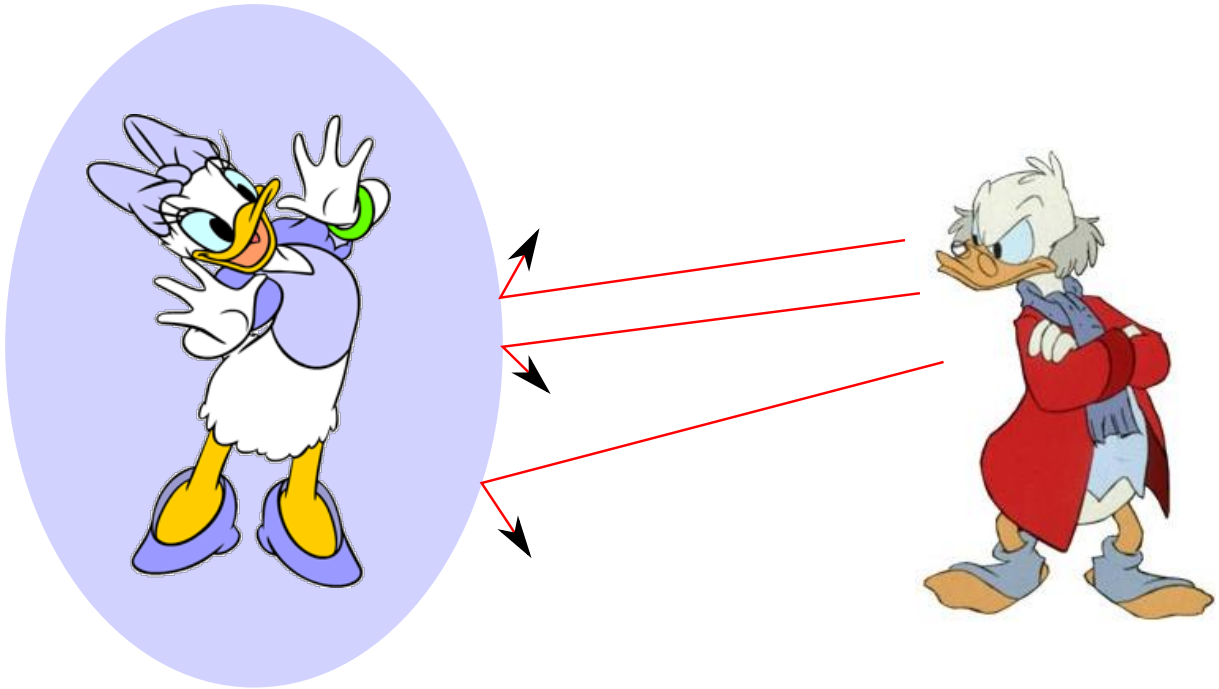
- Two-party cryptographic primitives
- Classical and quantum bit commitment
- Relativistic setting
- Relativistic bit commitment protocol [Kent'11]
- Security proof
- Summary and open questions

Two-party crypto – concept



both honest $\Longrightarrow$ protocol goes through and result is as expected

Two-party crypto – concept



Alice is honest $\Longrightarrow$ she is protected against dishonest Bob (e.g. she catches him cheating, aborts the protocol or he remains ignorant about her input) and *vice versa*

Two-party crypto – examples

secure function evaluation

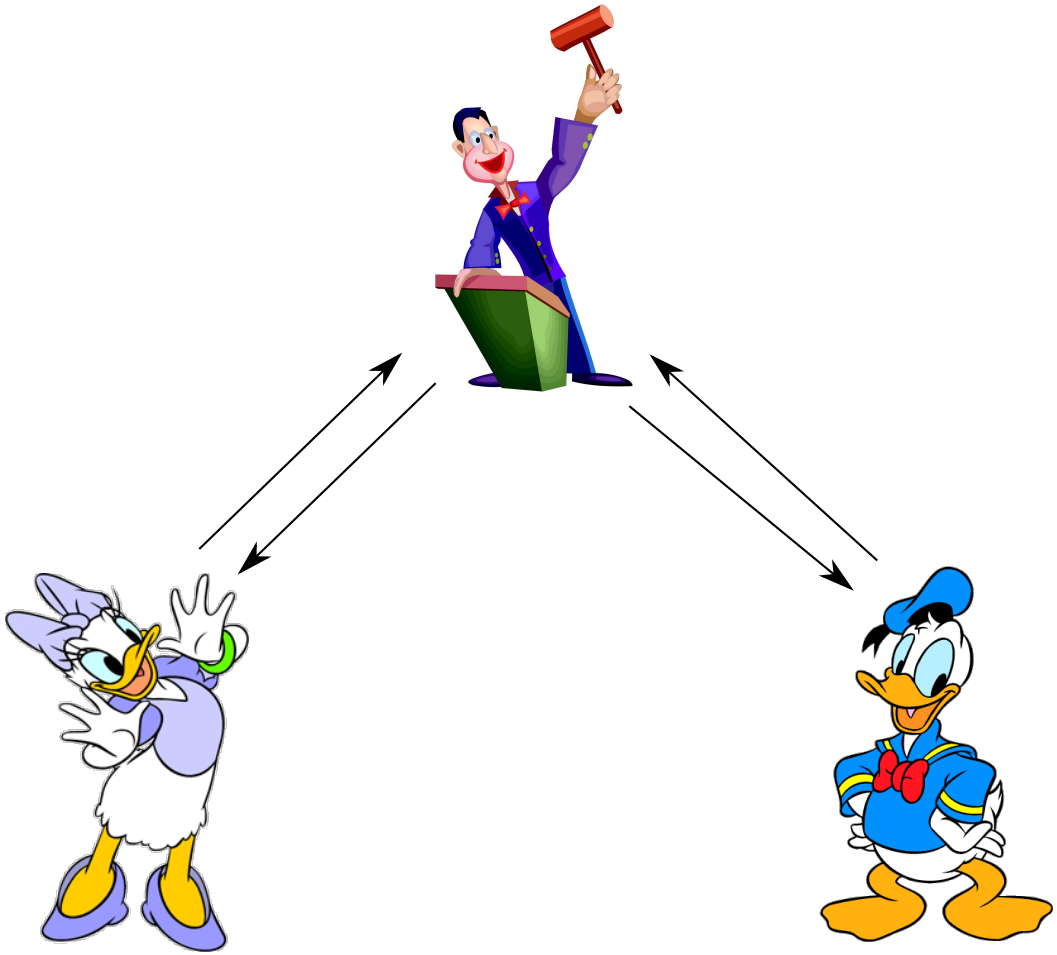
oblivious transfer

coin flip

(trusted, unbiased randomness)

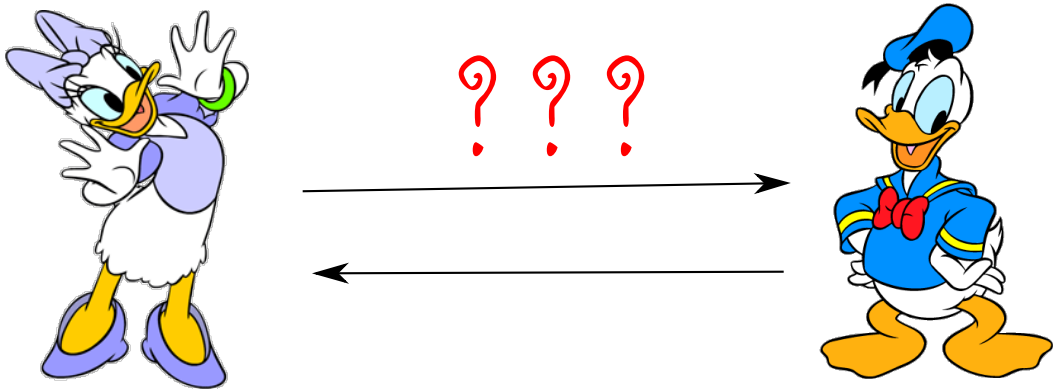
commitment schemes

Auction – motivation for commitment schemes



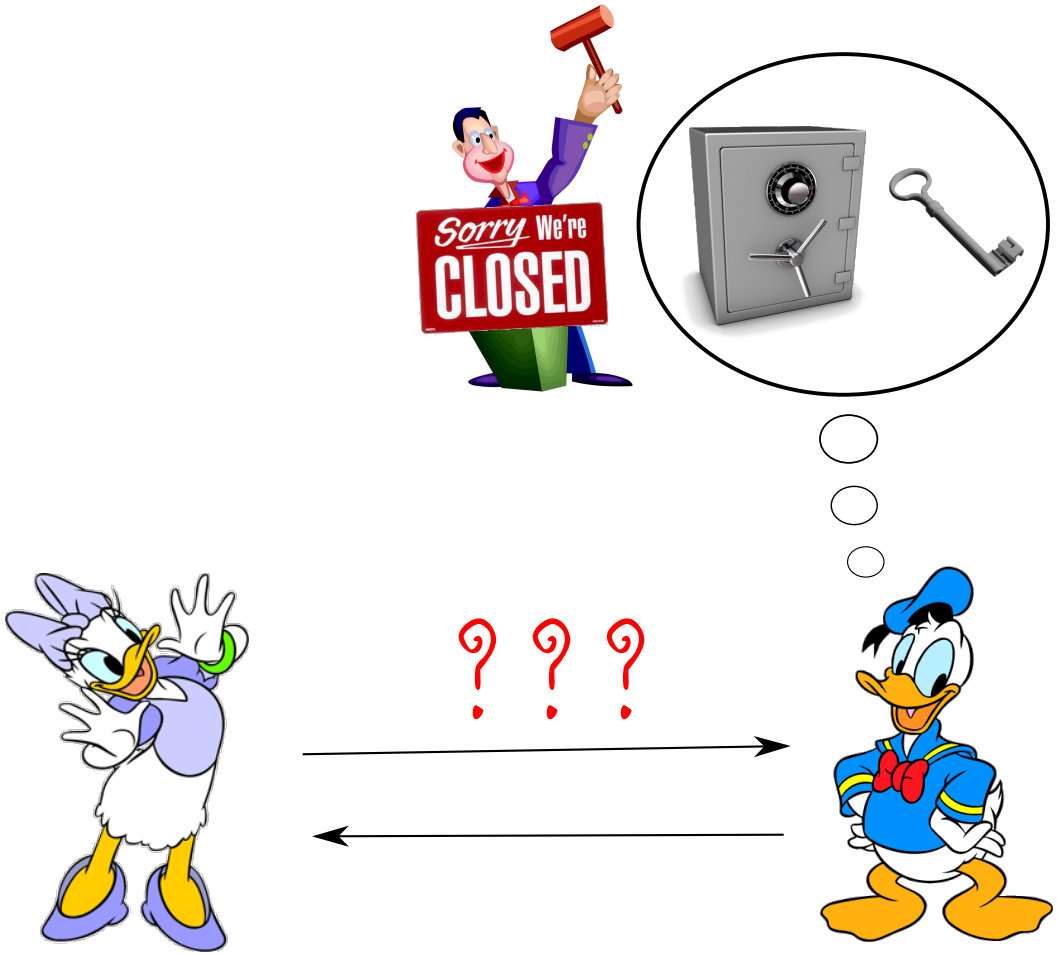
Auctioning is easy if a trusted third-party is available.

Auction – motivation for commitment schemes



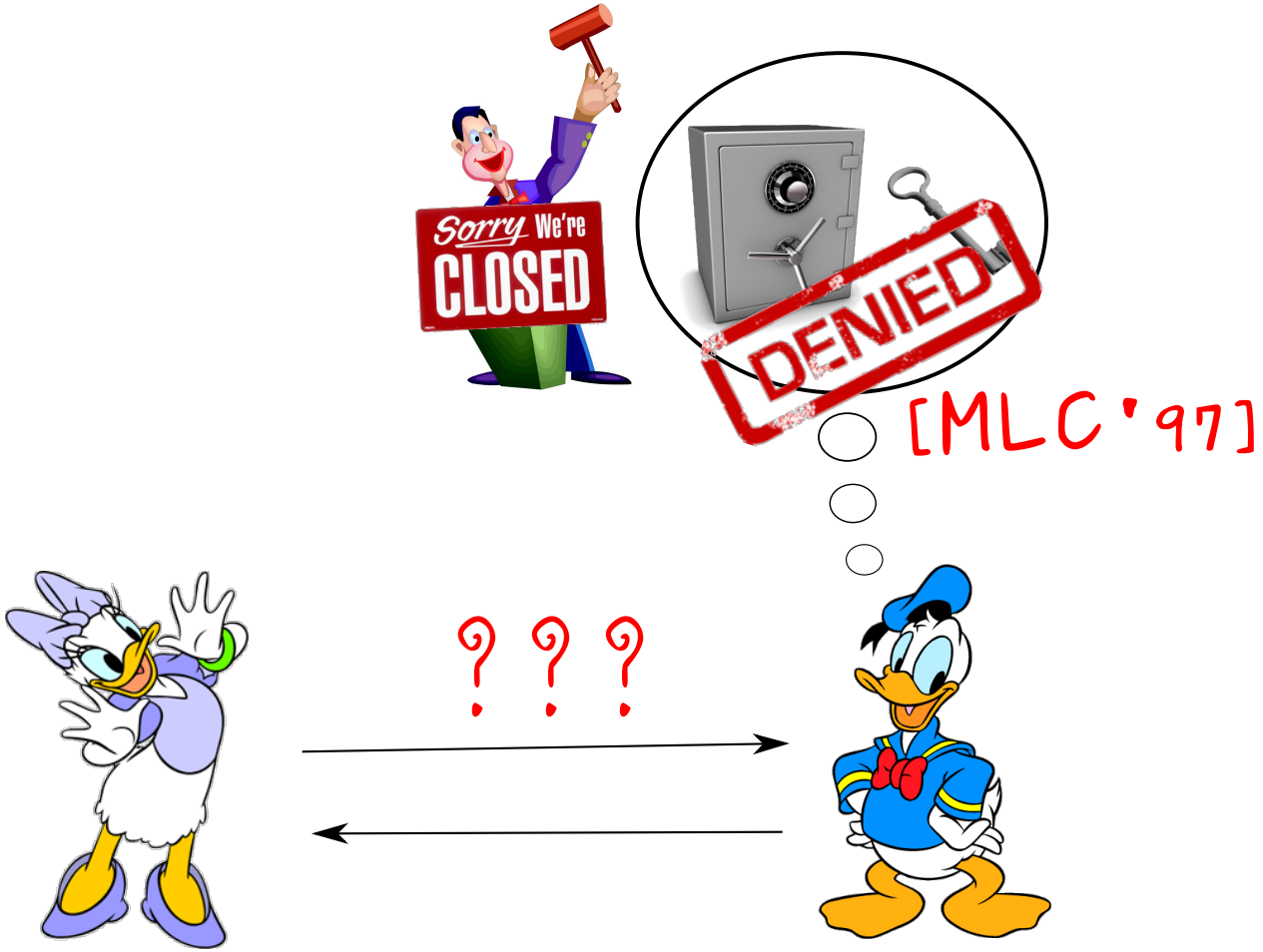
What if there is no trusted third-party? Be paranoid, trust nobody!

Auction – motivation for commitment schemes



We could do it if we had a perfect [information-theoretic] safe.

Auction – motivation for commitment schemes



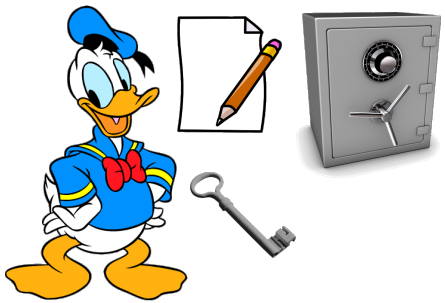
Is this it?

Outline

- Two-party cryptographic primitives
- Classical and quantum bit commitment
- Relativistic setting
- Relativistic bit commitment protocol [Kent'11]
- Security proof
- Summary and open questions

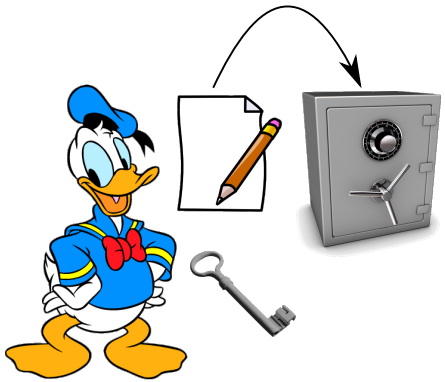
Bit commitment – ideal functionality

Commit phase



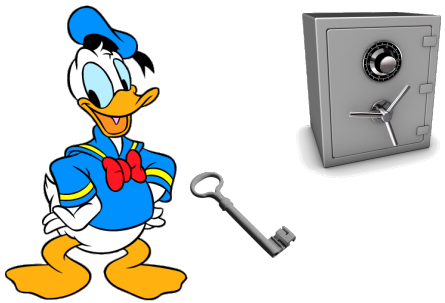
Bit commitment – ideal functionality

Commit phase



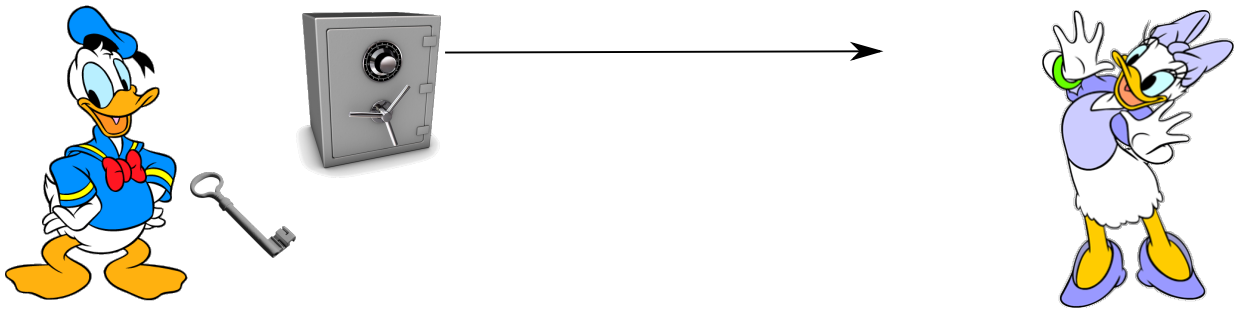
Bit commitment – ideal functionality

Commit phase



Bit commitment – ideal functionality

Commit phase



Bit commitment – ideal functionality

Commit phase

Bit commitment – ideal functionality

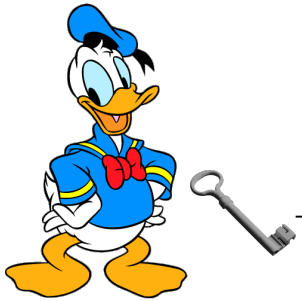
Commit phase

Open phase

Bit commitment – ideal functionality

Commit phase

Open phase



Bit commitment – ideal functionality

Commit phase

Open phase

Bit commitment – ideal functionality

Commit phase

Open phase

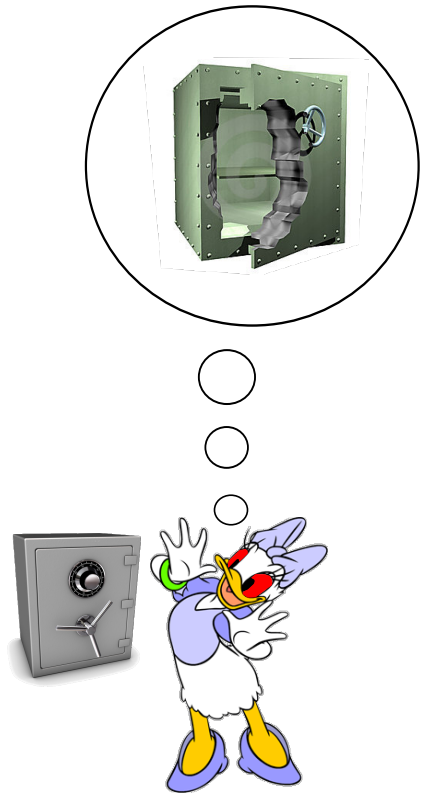
Cheating objectives

The commit phase is over...

Cheating objectives

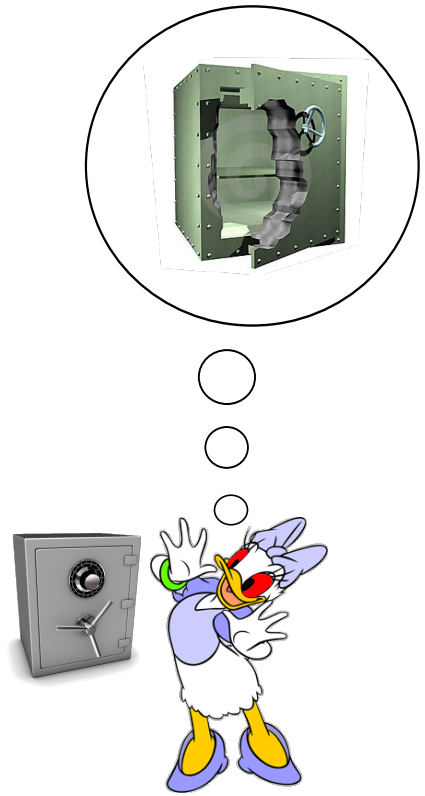
Alice goes mad!

Cheating objectives



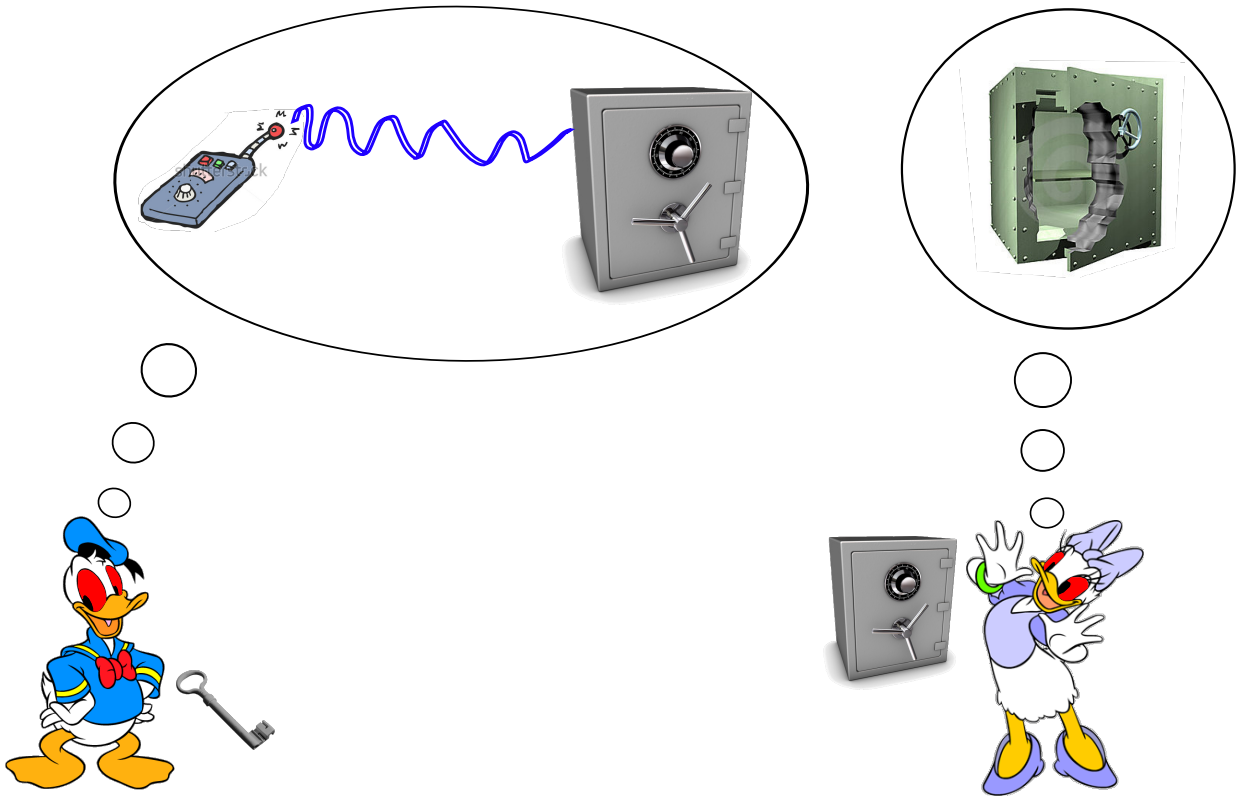
She wants to break the safe and read the message!

Cheating objectives



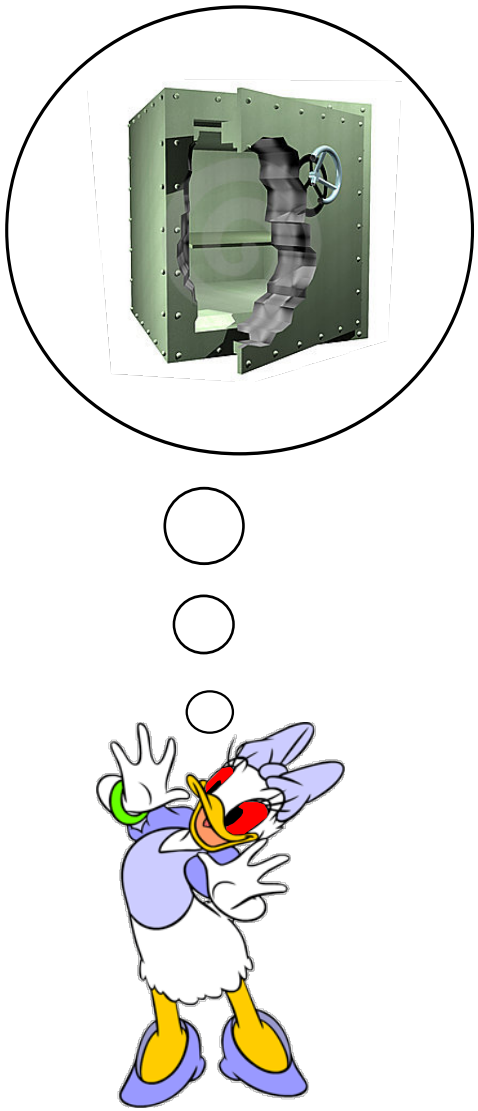
Bob goes mad!

Cheating objectives



He wants to influence the message, he wants to be uncommitted!

Security criteria



The scheme should be **hiding**.

p_{guess} – probability that Alice guesses the commited bit correctly after the commit phase is over

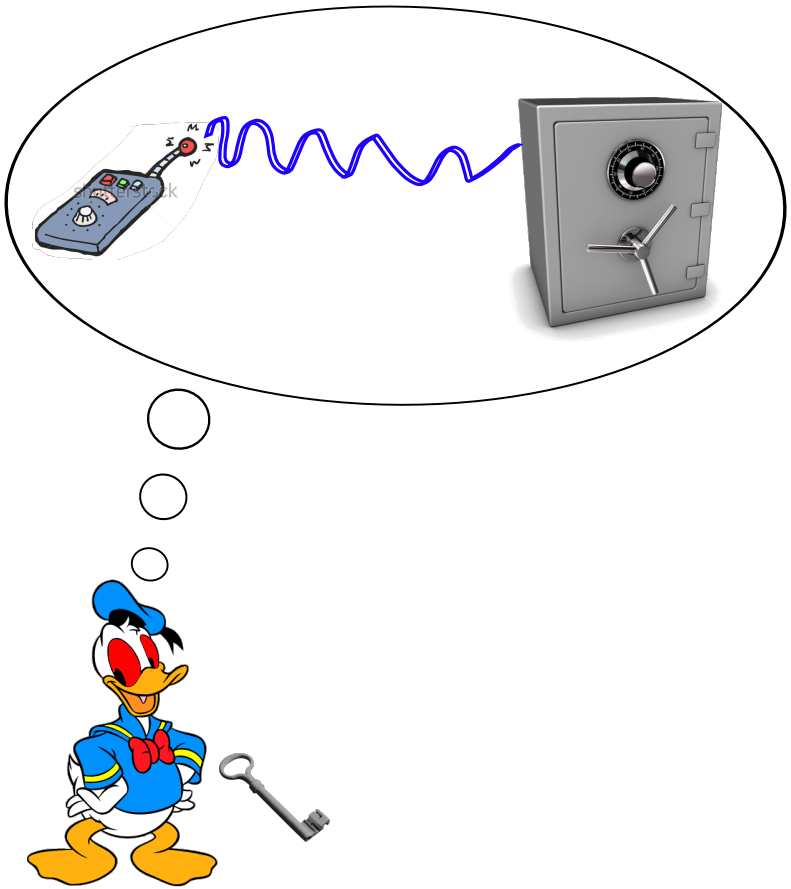
Definition

*A bit commitment protocol is δ-**hiding** if the fact that Bob is honest implies*

$$p_{\text{guess}} \leq \frac{1}{2} + \delta.$$

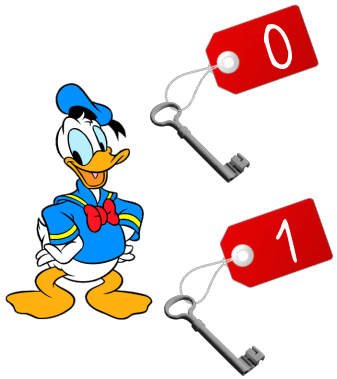
Security criteria

The scheme should be **binding**.



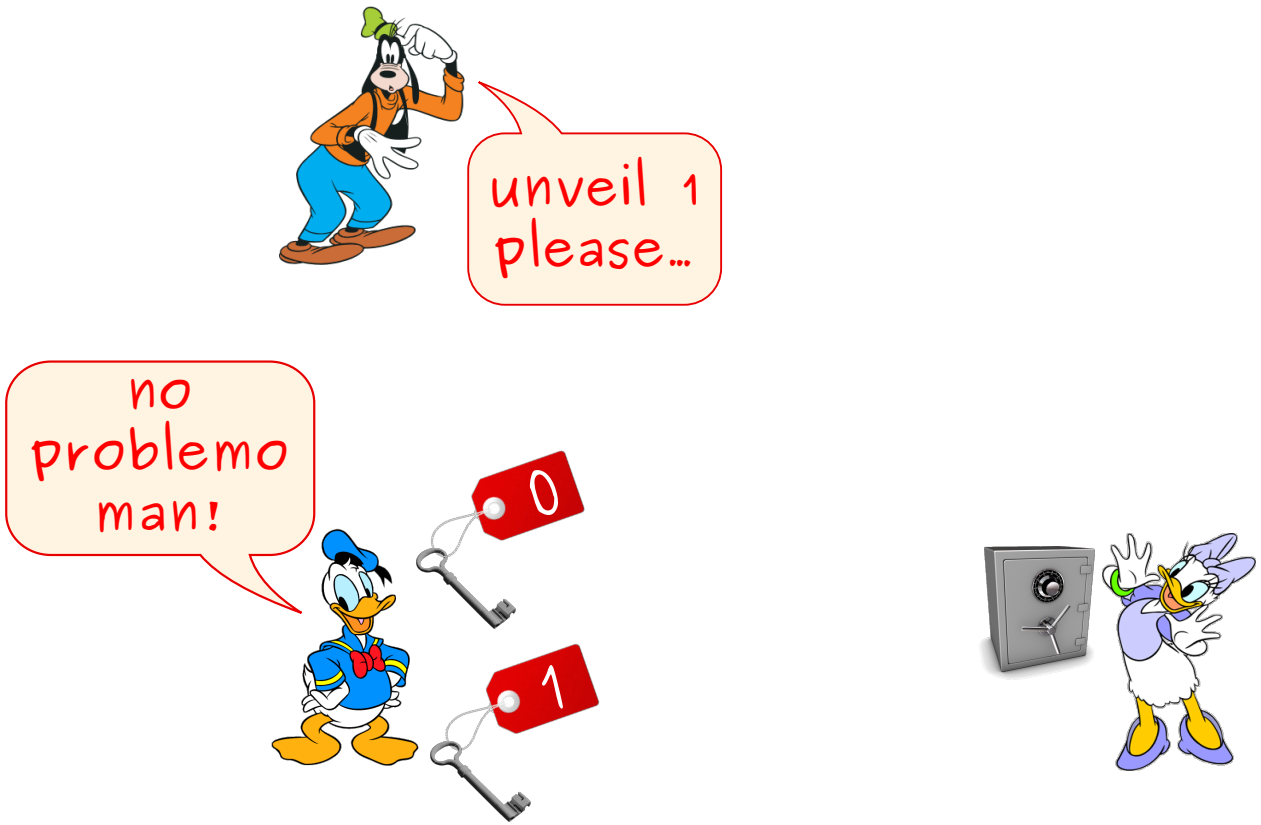
Bob should not be able to change his mind after the commit phase is over.

Security criteria



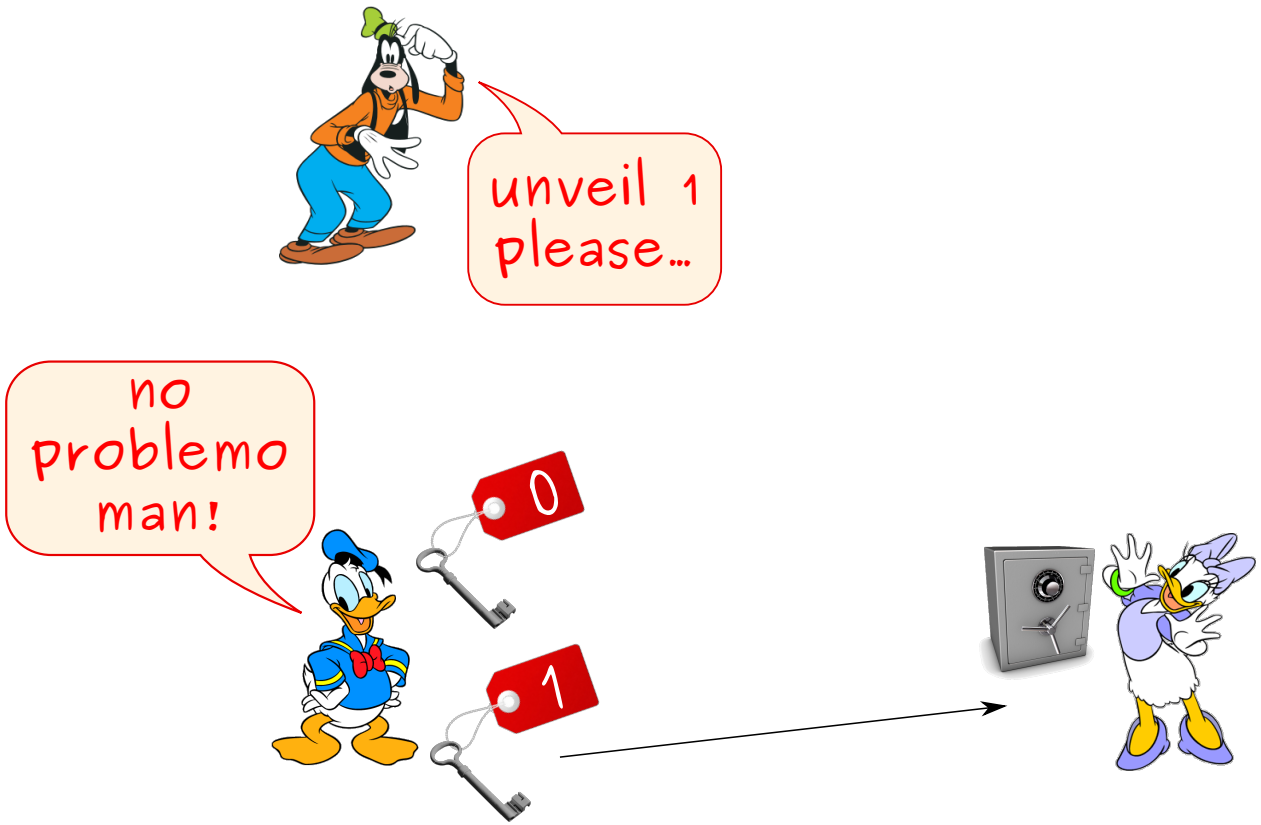
Dishonest Bob will have two different keys...

Security criteria



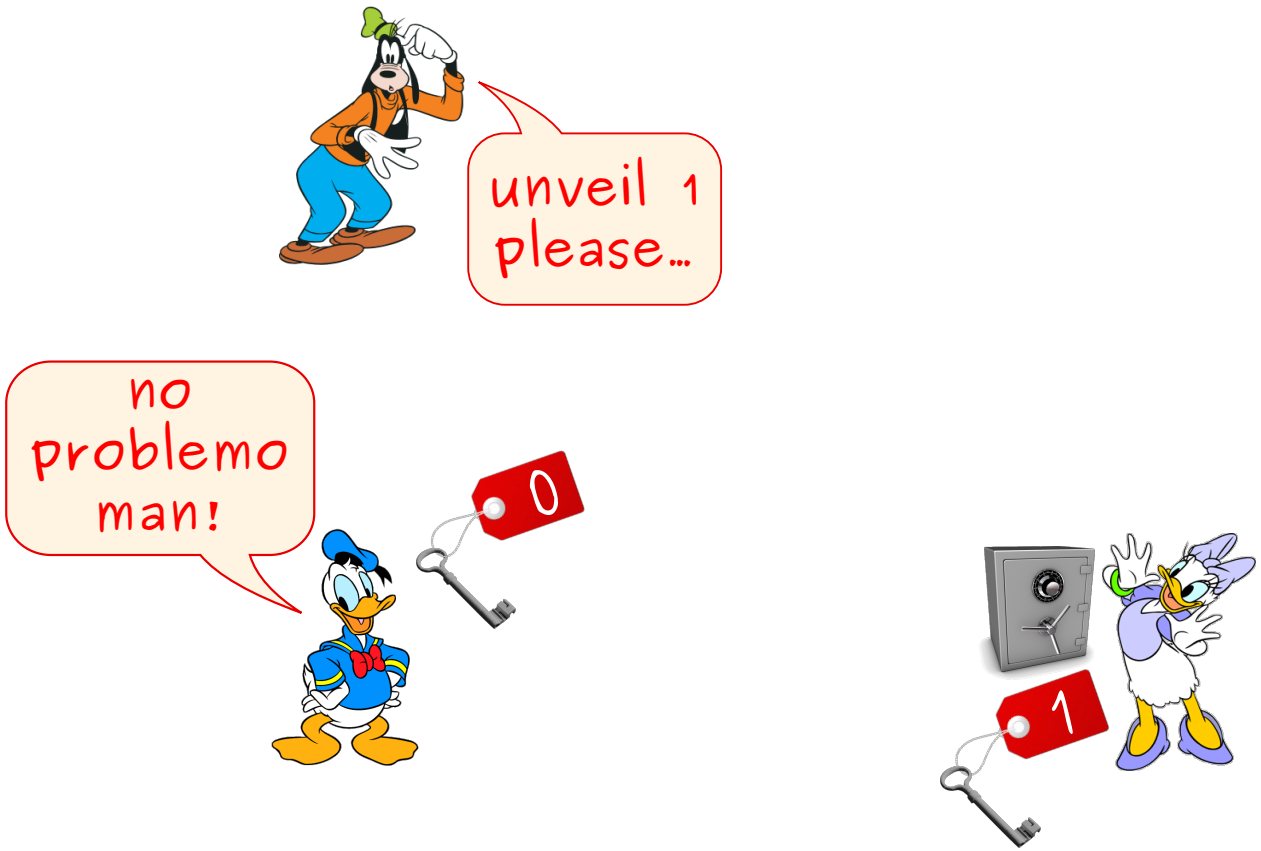
External verifier Victor asks him to unveil 1.

Security criteria



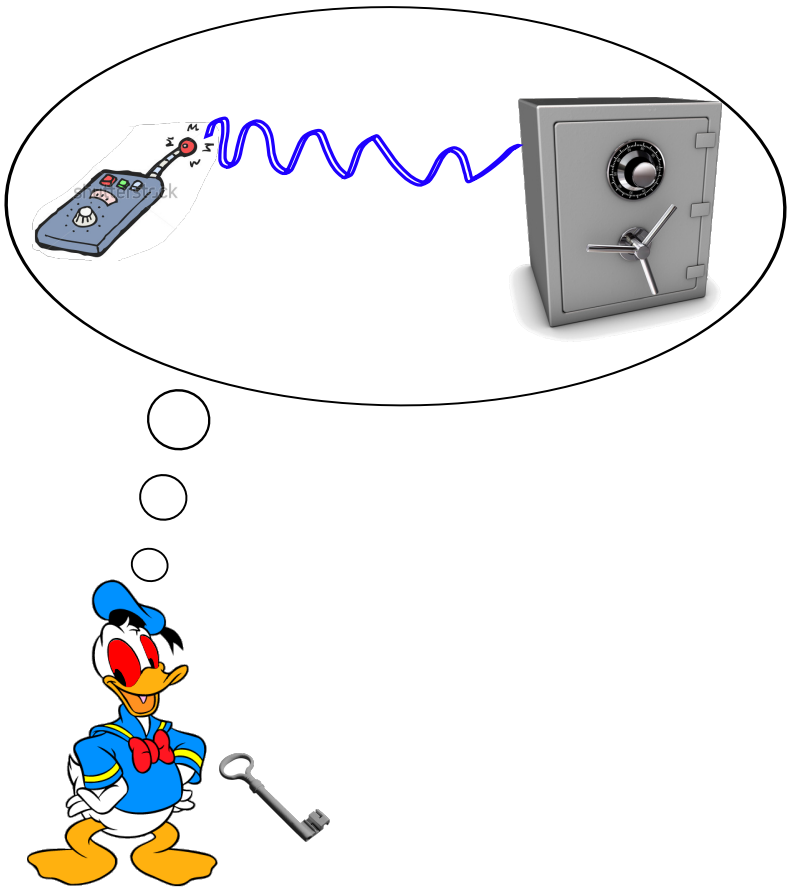
Bob attempts to unveil 1.

Security criteria



p_1 is the probability that Alice accepts the unveiling.

Security criteria



Definition

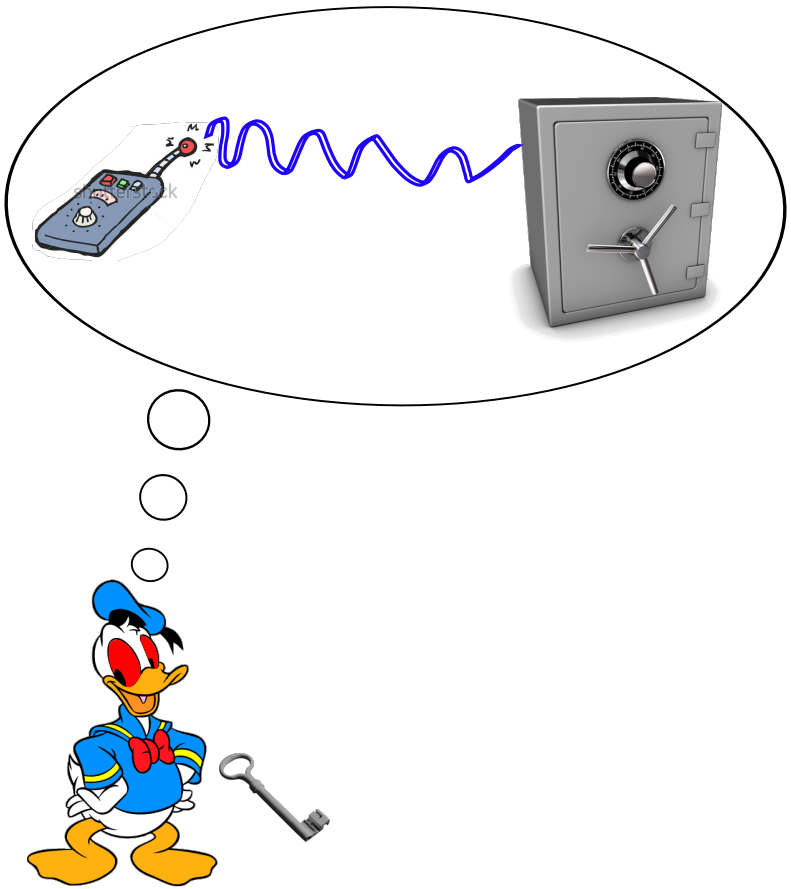
A bit commitment protocol is **ϵ-binding** *if the fact that Alice is honest implies that there exists a bit $c \in \{0, 1\}$ such that $p_c \leq \epsilon$.*

What about **superposition** commitment?

For any protocol Bob can commit to an honest superposition and achieve $p_0 = p_1 = \frac{1}{2}$.

Not satisfiable in the quantum world...

Security criteria



Definition

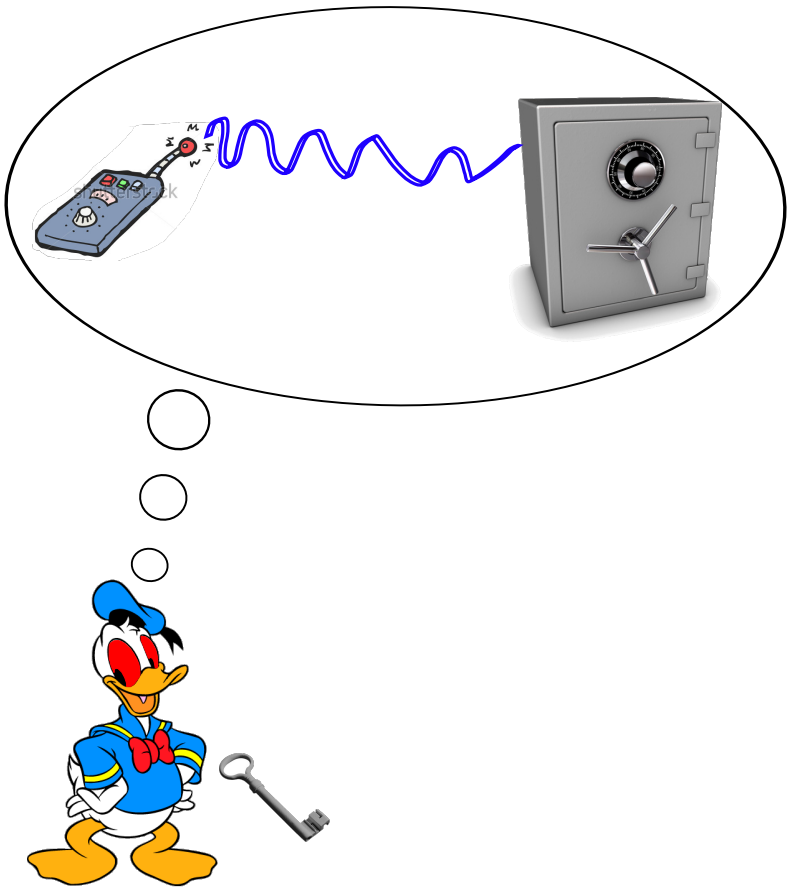
A bit commitment protocol is **ϵ-binding** *if the fact that Alice is honest implies that there exists a bit $c \in \{0, 1\}$ such that $p_c \leq \epsilon$.*

What about **superposition** commitment?

For any protocol Bob can commit to an honest superposition and achieve $p_0 = p_1 = \frac{1}{2}$.

Not satisfiable in the quantum world...

Security criteria



Definition

A bit commitment protocol is **ϵ-binding** *if the fact that Alice is honest implies that there exists a bit $c \in \{0, 1\}$ such that $p_c \leq \epsilon$.*

What about **superposition** commitment?

For any protocol Bob can commit to an honest superposition and achieve $p_0 = p_1 = \frac{1}{2}$.

Not satisfiable in the quantum world...

Security criteria

Definition

A bit commitment protocol is ϵ**-binding** *if the fact that Alice is honest implies that there exists a bit* $c \in \{0, 1\}$ *such that* $p_c \leq \epsilon$.

Definition

A bit commitment protocol is ϵ**-weakly binding** *if the fact that Alice is honest implies that* $p_0 + p_1 \leq 1 + \epsilon$.

Composability? forget it...

Security criteria

Definition

A bit commitment protocol is ϵ**-binding** *if the fact that Alice is honest implies that there exists a bit* $c \in \{0, 1\}$ *such that* $p_c \leq \epsilon$.

Definition

A bit commitment protocol is ϵ**-weakly binding** *if the fact that Alice is honest implies that* $p_0 + p_1 \leq 1 + \epsilon$.

Composability? forget it...

Security criteria

Definition

A bit commitment protocol is ϵ**-binding** *if the fact that Alice is honest implies that there exists a bit* $c \in \{0, 1\}$ *such that* $p_c \leq \epsilon$.

Definition

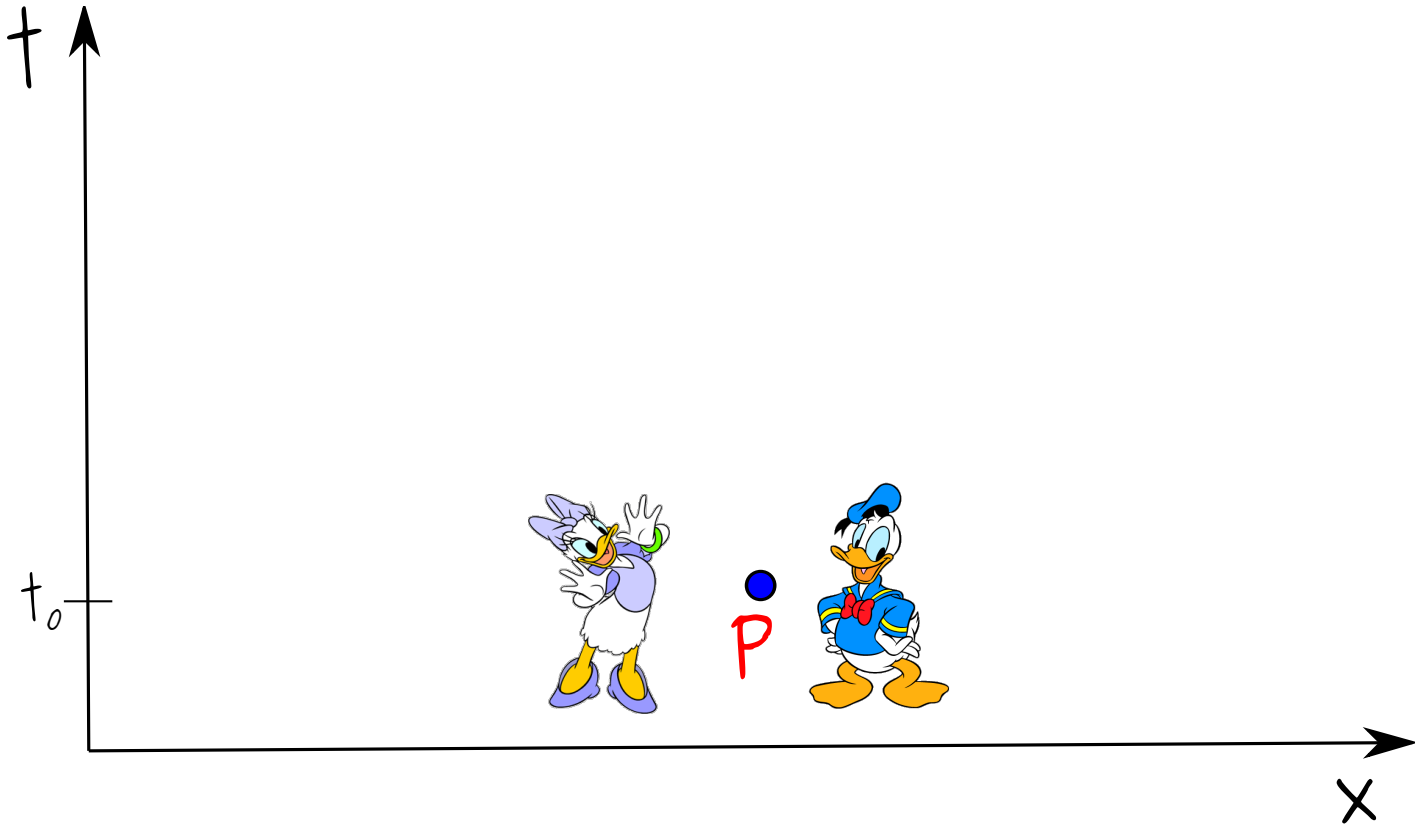
A bit commitment protocol is ϵ**-weakly binding** *if the fact that Alice is honest implies that* $p_0 + p_1 \leq 1 + \epsilon$.

Composability? forget it...

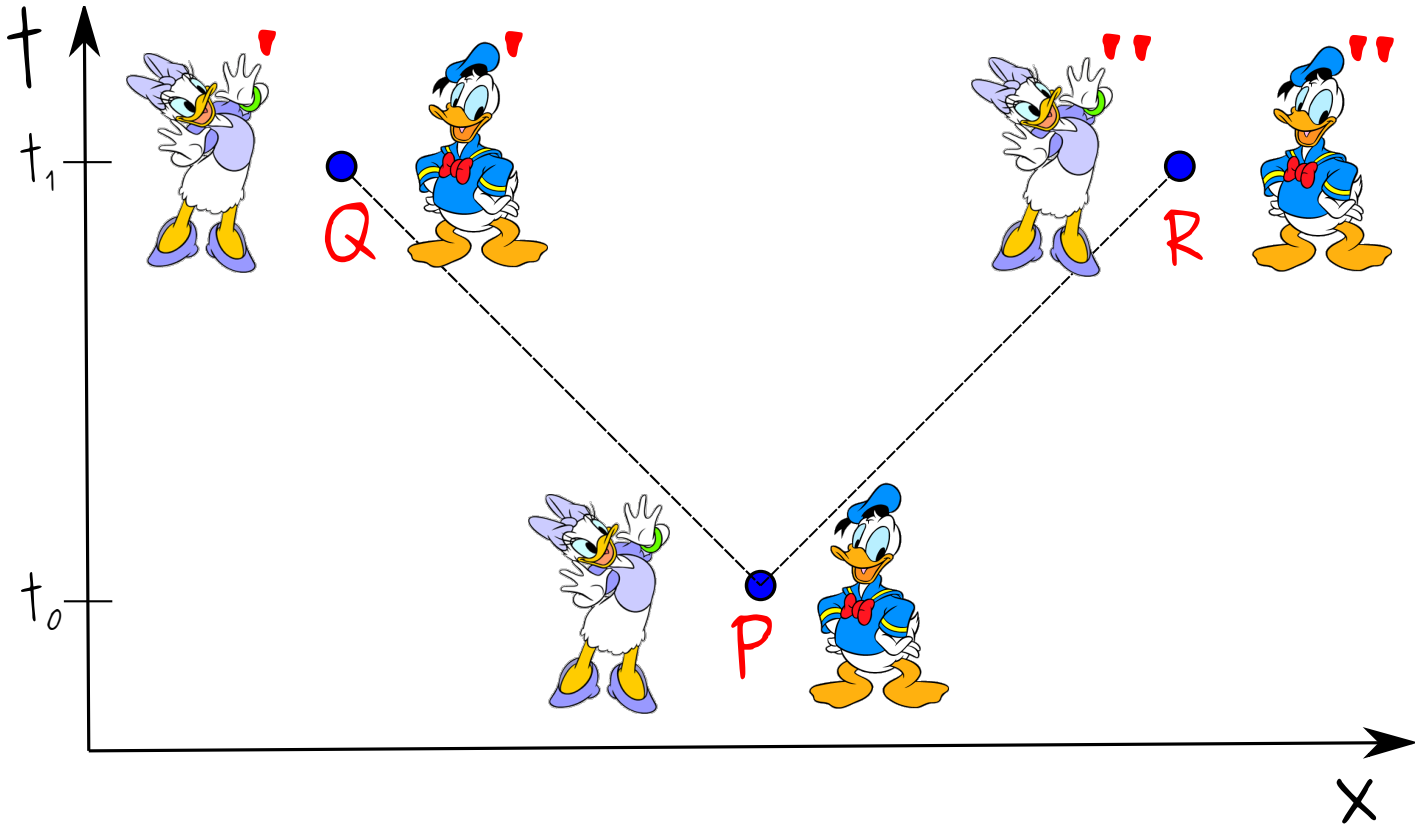
Outline

- Two-party cryptographic primitives
- Classical and quantum bit commitment
- Relativistic setting
- Relativistic bit commitment protocol [Kent'11]
- Security proof
- Summary and open questions

Relativistic setting

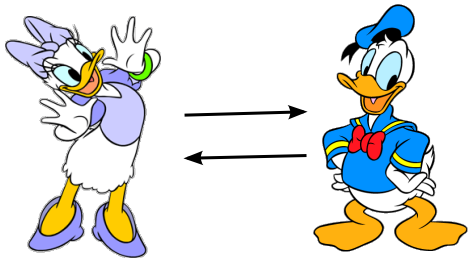


Relativistic setting



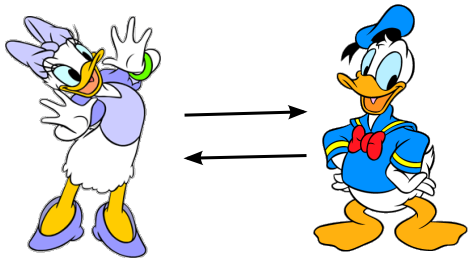
Relativistic setting

Phase 1

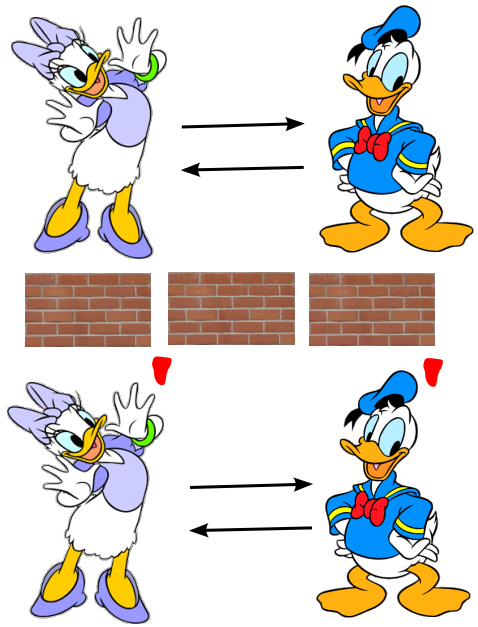


Relativistic setting

Phase 1

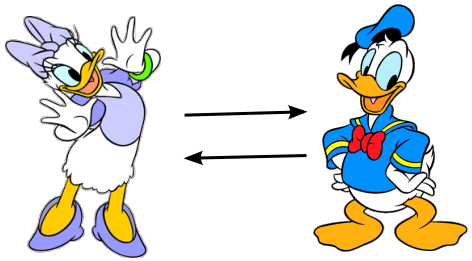


Phase 2

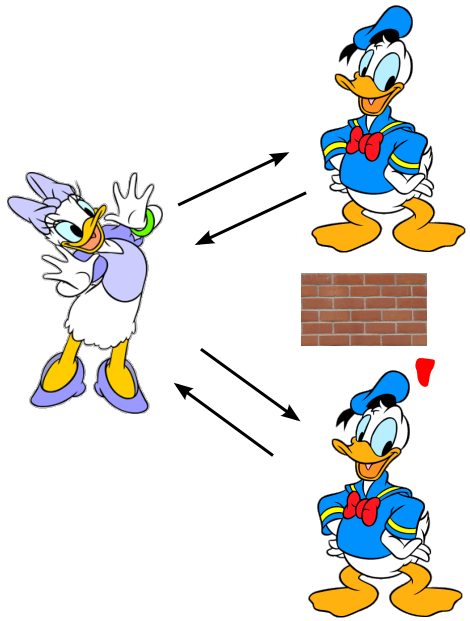


Relativistic setting

Phase 1

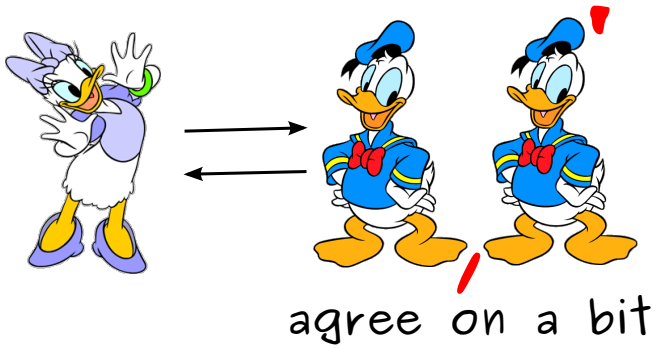


Phase 2

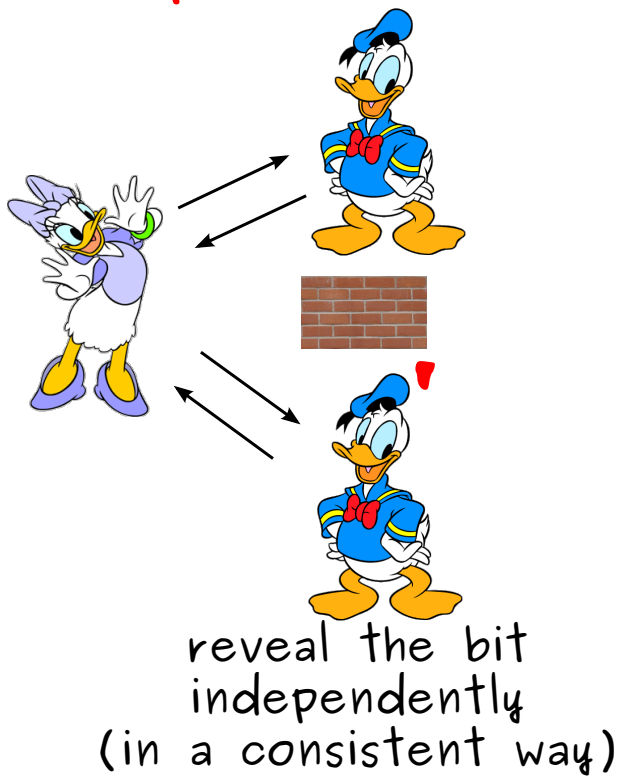


Relativistic setting

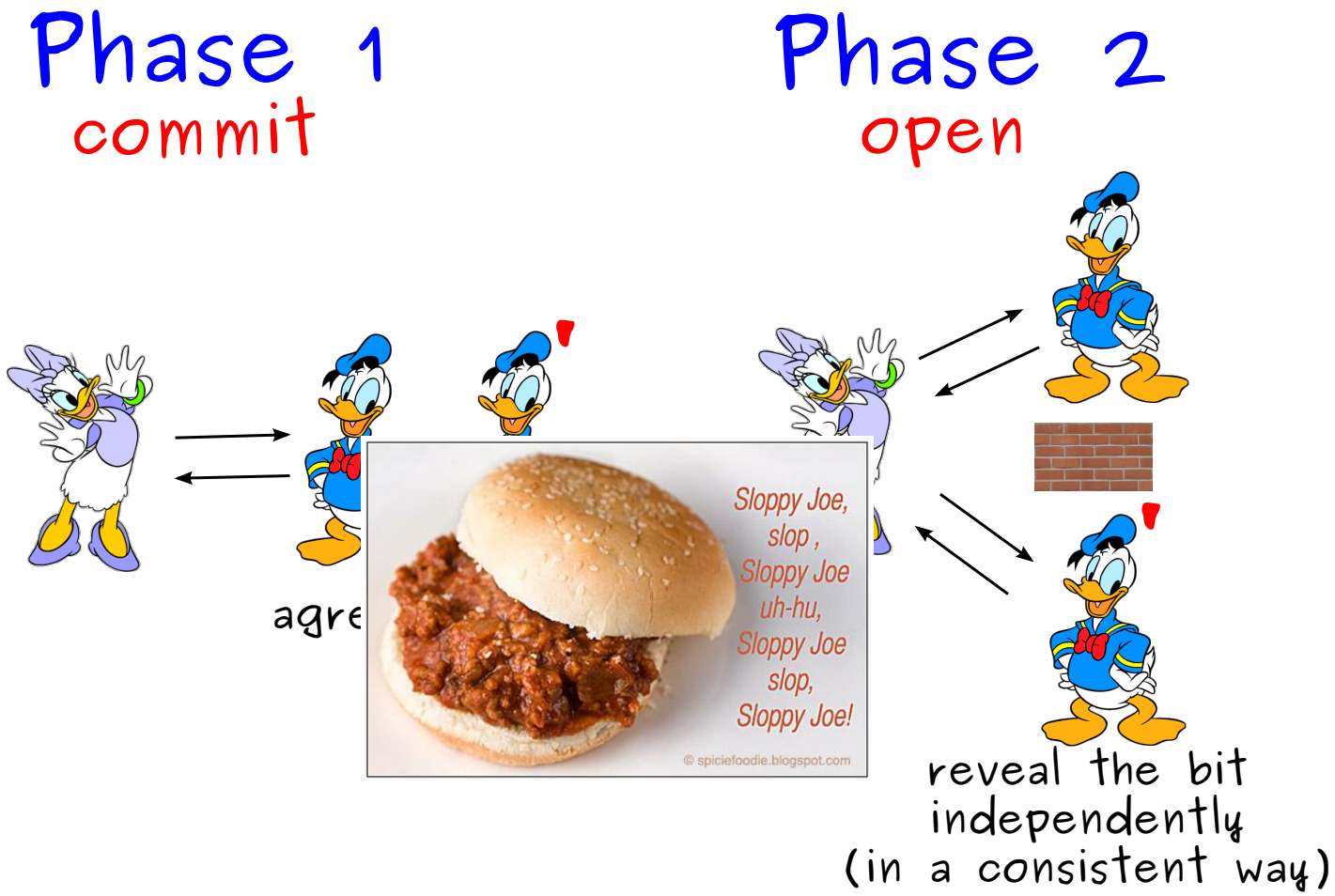
Phase 1
commit



Phase 2
open

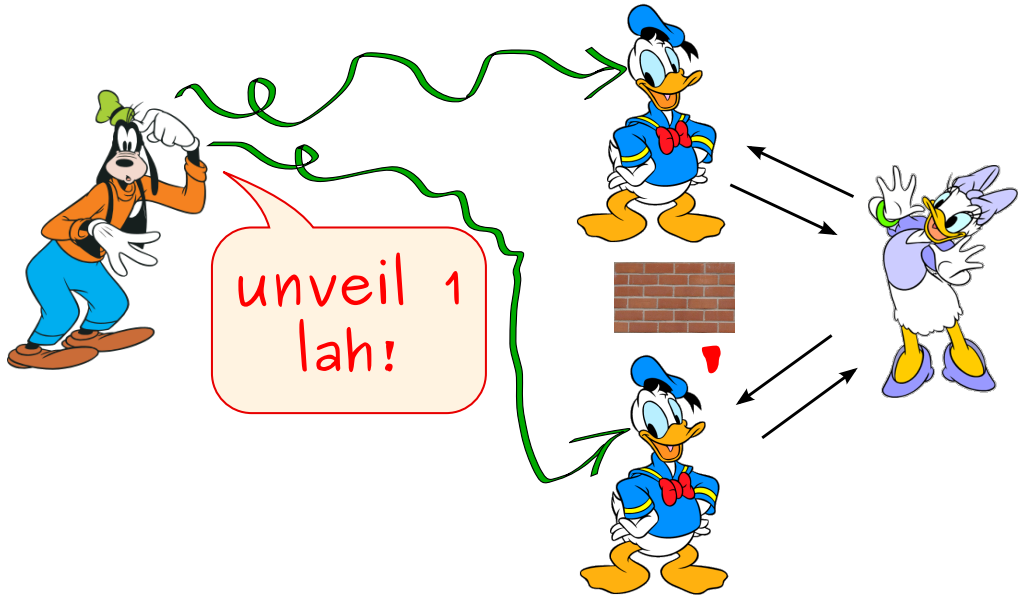


Relativistic setting

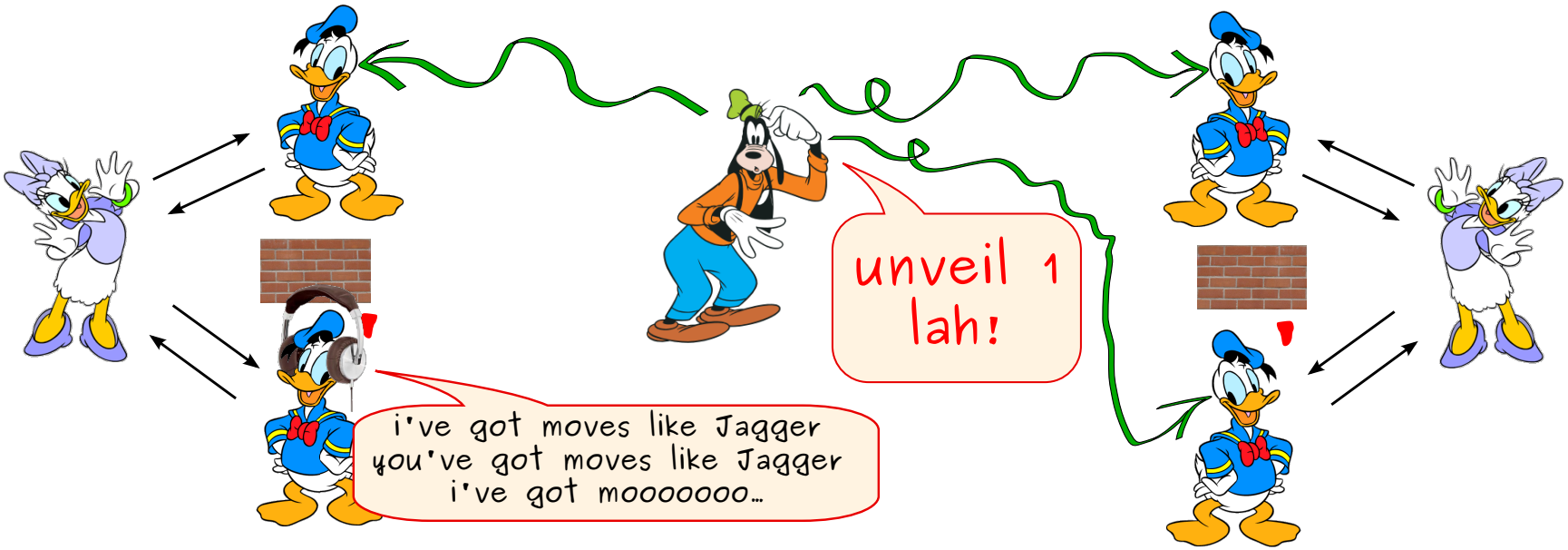


Local vs. global command

Local vs. global command



Local vs. global command



Local vs. global command

Local vs. global command

local command

trivial protocol is secure
(directly from no-signalling)

global command

no classical protocol
can be secure

Local vs. global command

local command

trivial protocol is secure
(directly from no-signalling)

global command

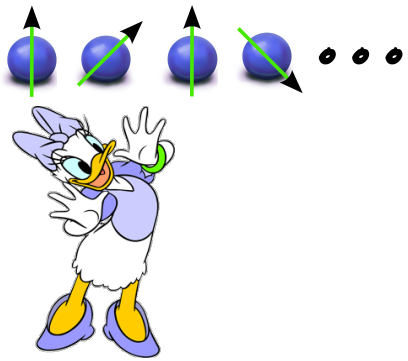
no classical protocol
can be secure

what about quantum ???

Outline

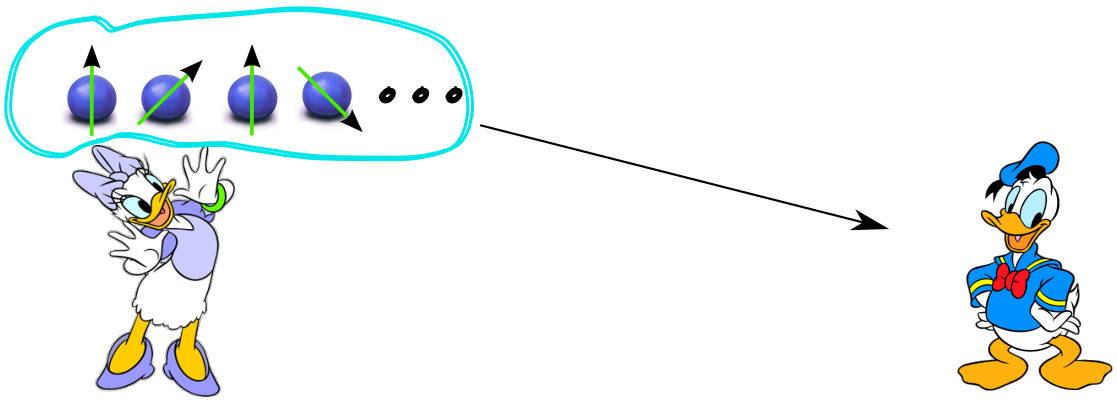
- Two-party cryptographic primitives
- Classical and quantum bit commitment
- Relativistic setting
- Relativistic bit commitment protocol [Kent'11]
- Security proof
- Summary and open questions

RBC [Kent'11] – Commit phase



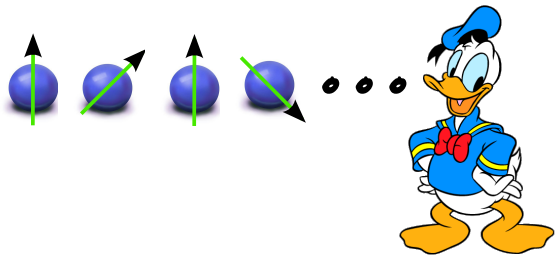
Alice creates n BB84 states ...

RBC [Kent'11] – Commit phase



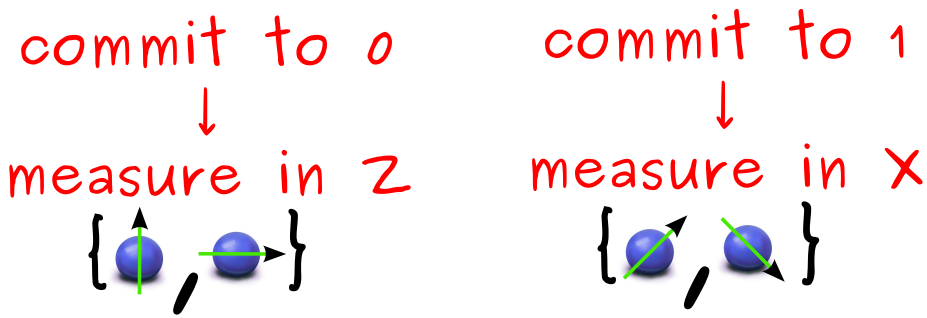
. . . and sends them to Bob.

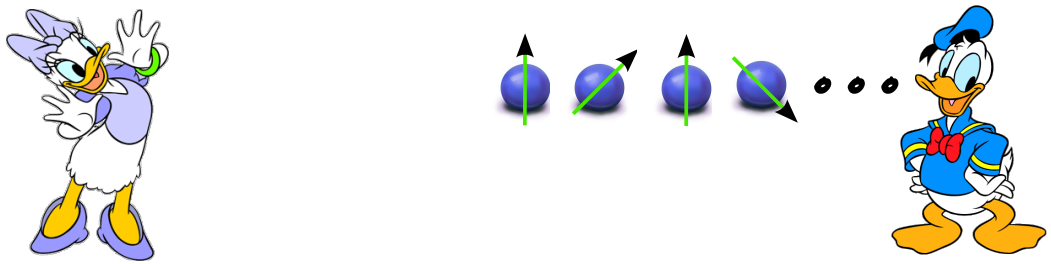
RBC [Kent'11] – Commit phase



Bob receives the qubits . . .

RBC [Kent'11] – Commit phase





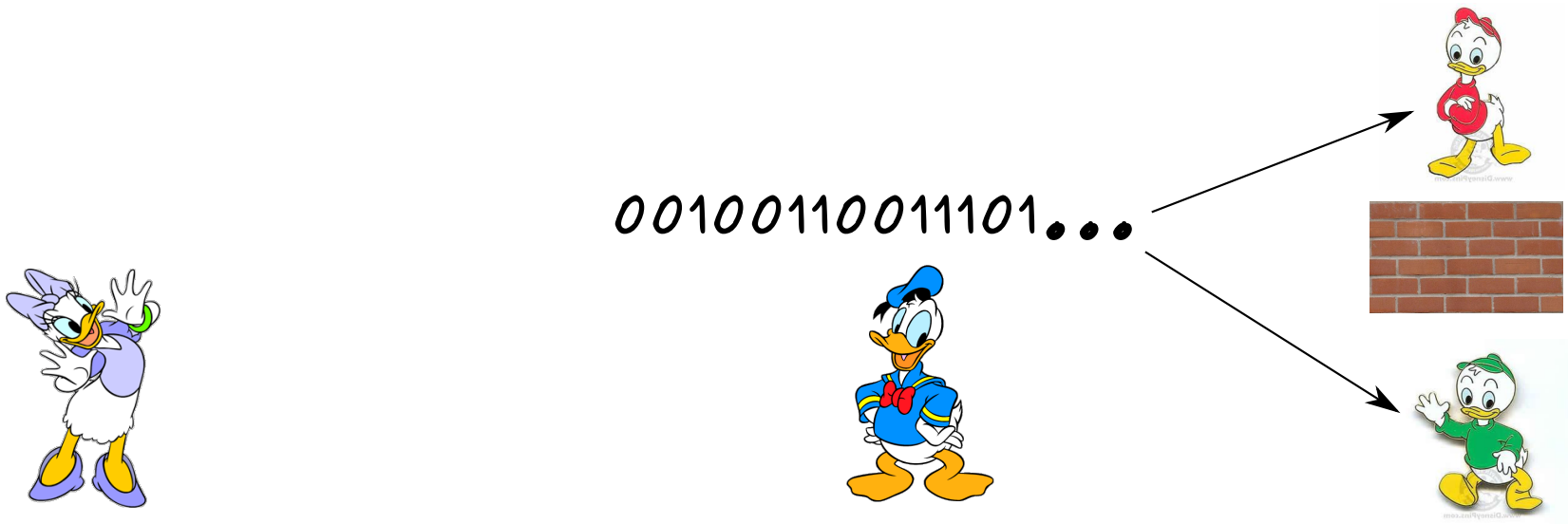
. . . and measures them in either computational or Hadamard basis.

RBC [Kent'11] – Commit phase

00100110011101...

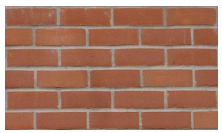
Bob obtains a bit string . . .

RBC [Kent'11] – Commit phase

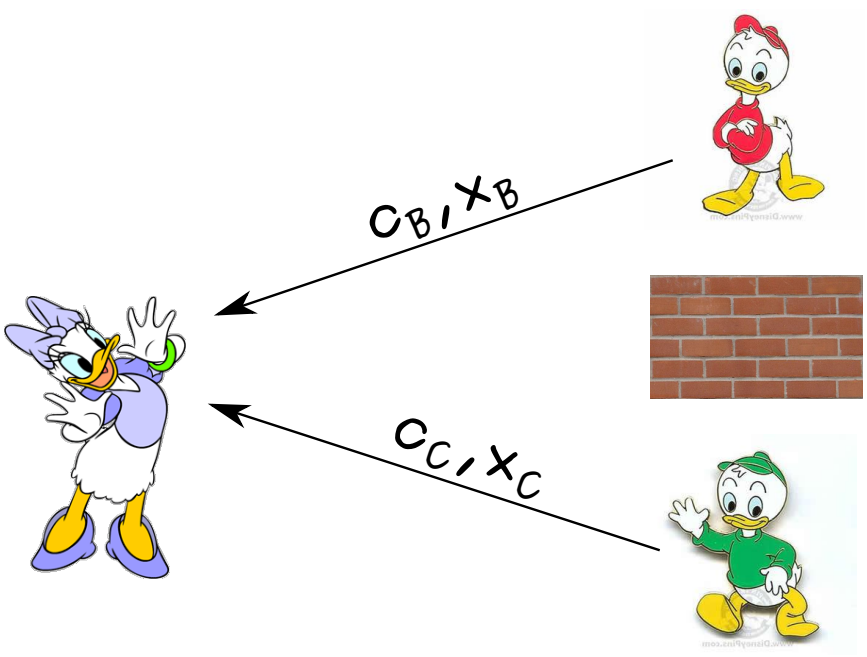


. . . and sends it to his agents.

RBC [Kent'11] – Open phase



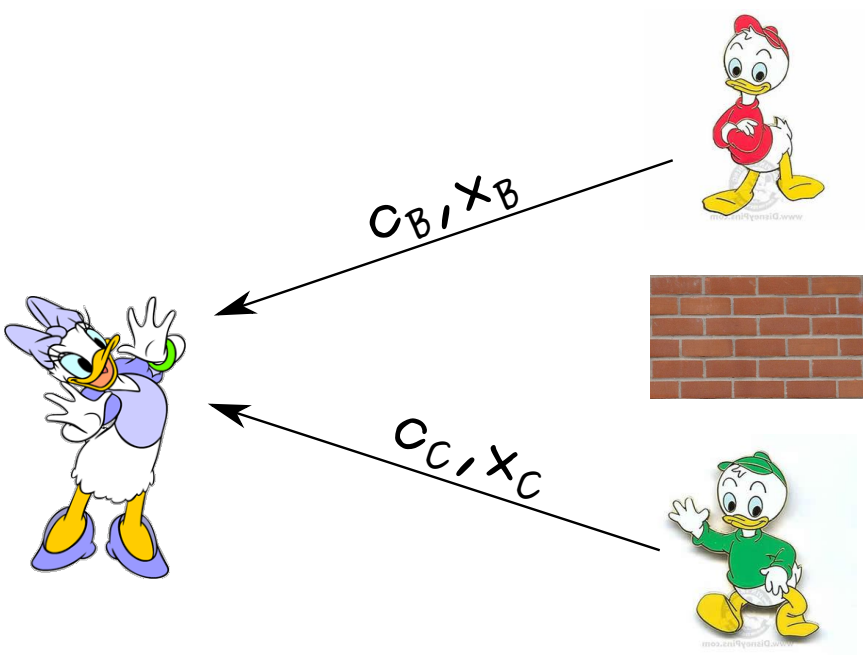
RBC [Kent'11] – Open phase



Alice checks if

- $c_B \stackrel{?}{=} c_C$,
- x_B is consistent with the BB84 states,
- x_C is consistent with the BB84 states.

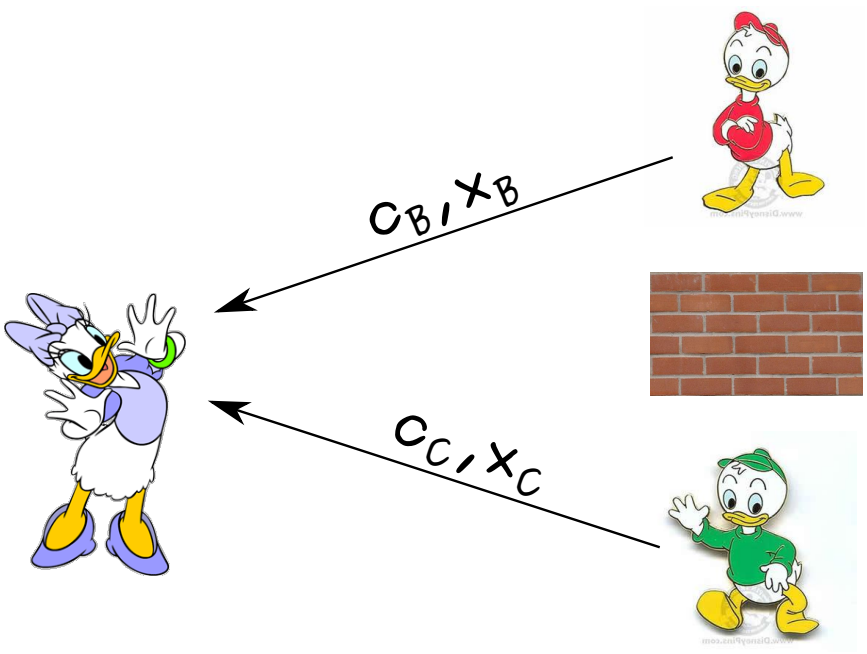
RBC [Kent'11] – Open phase



Alice checks if

- $c_B \stackrel{?}{=} c_C$,
- x_B is consistent with the BB84 states,
- x_C is consistent with the BB84 states.

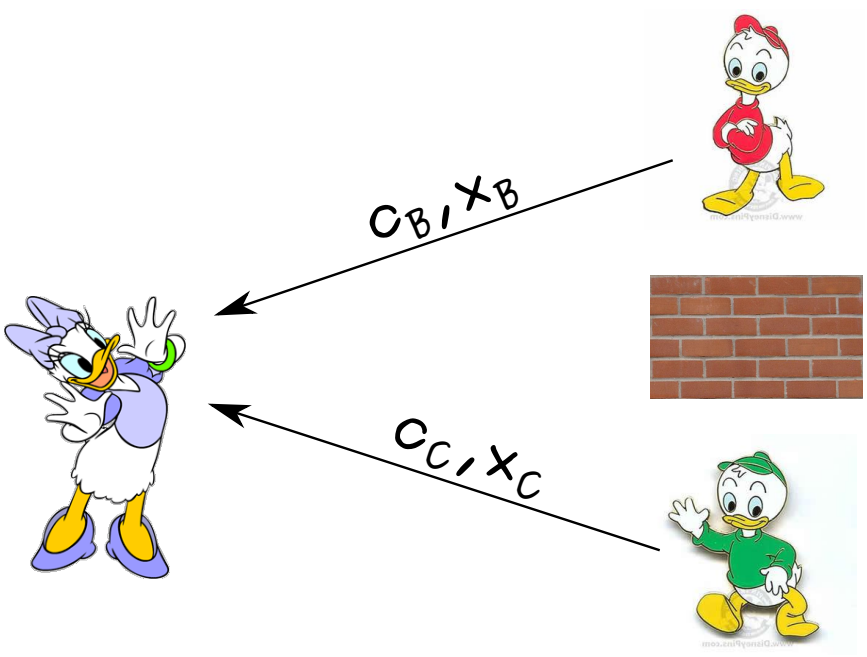
RBC [Kent'11] – Open phase



Alice checks if

- $c_B \stackrel{?}{=} c_C$,
- x_B is consistent with the BB84 states,
- x_C is consistent with the BB84 states.

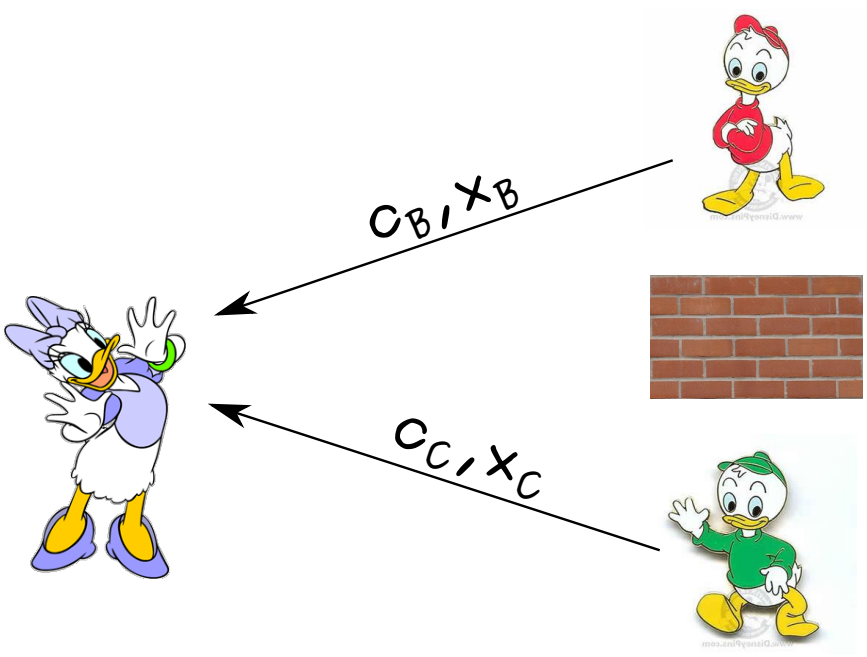
RBC [Kent'11] – Open phase



Alice checks if

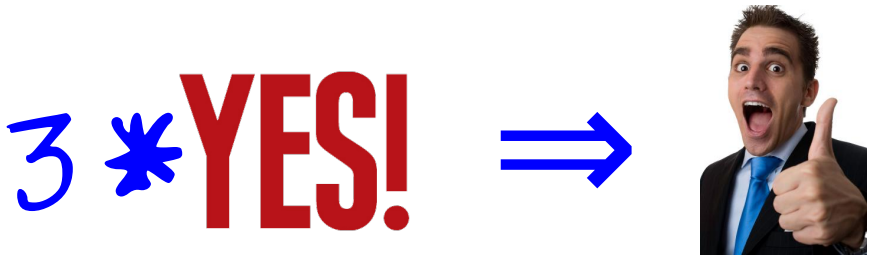
- $c_B \stackrel{?}{=} c_C$,
- x_B is consistent with the BB84 states,
- x_C is consistent with the BB84 states.

RBC [Kent'11] – Open phase

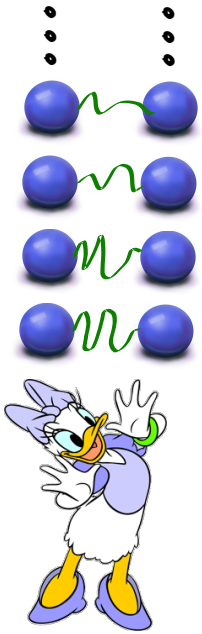


Alice checks if

- $c_B \stackrel{?}{=} c_C$,
- x_B is consistent with the BB84 states,
- x_C is consistent with the BB84 states.

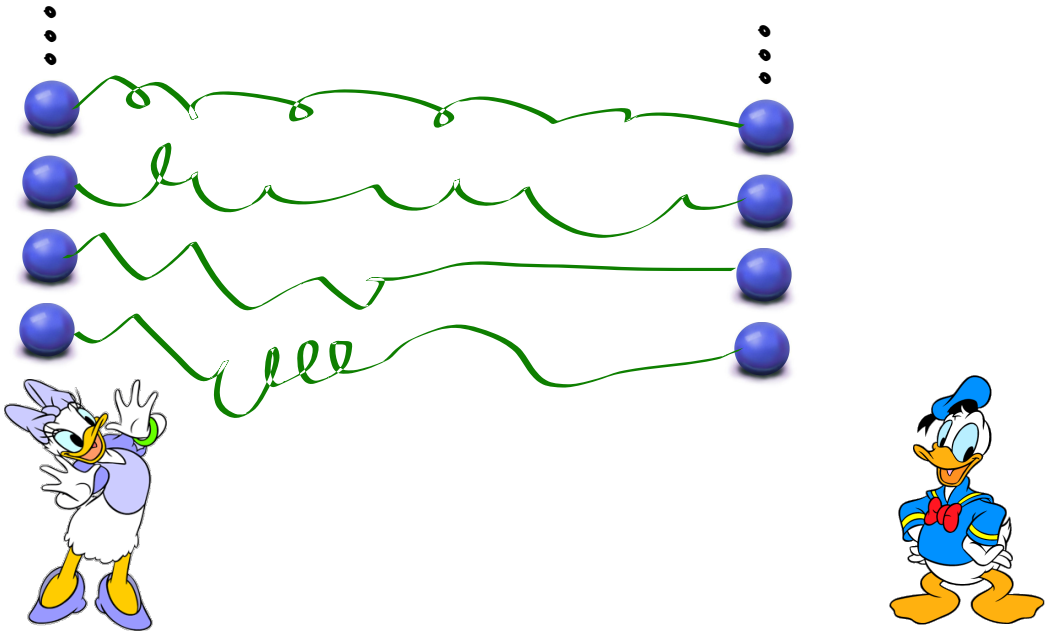


Purified RBC [Kent'11]



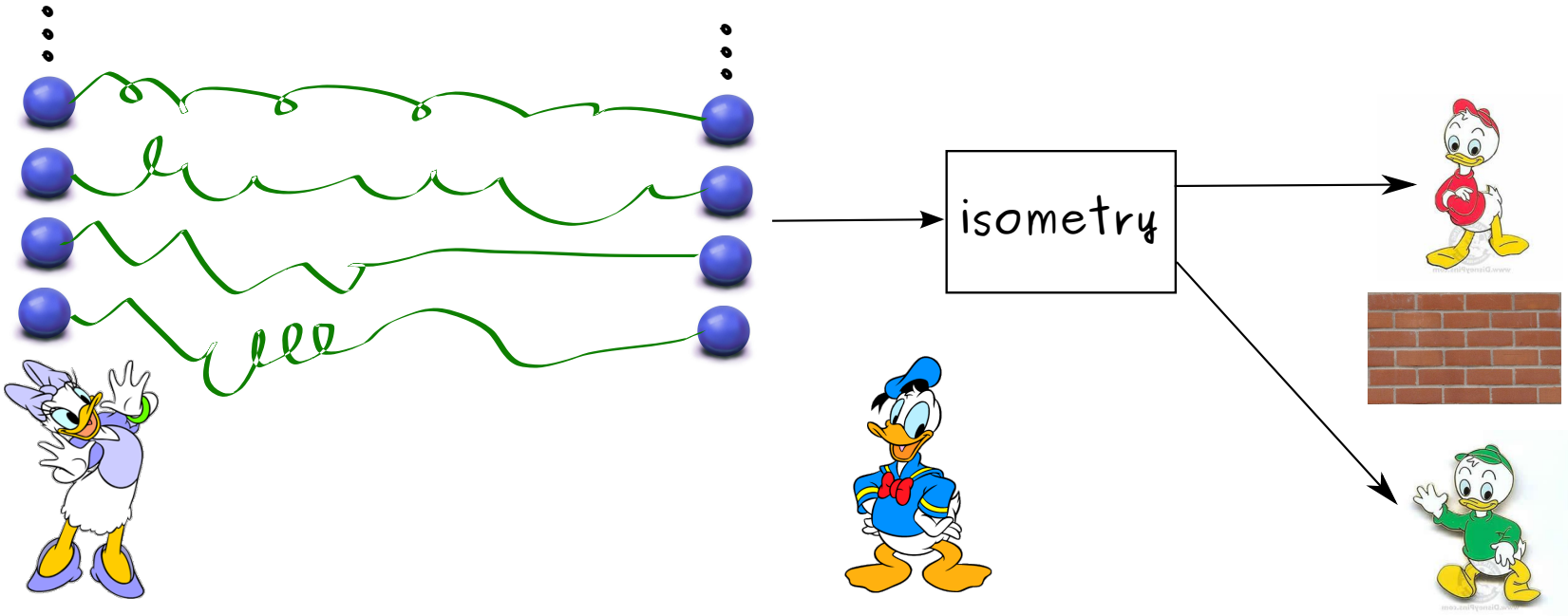
Alice creates n EPR pairs . . .

Purified RBC [Kent'11]



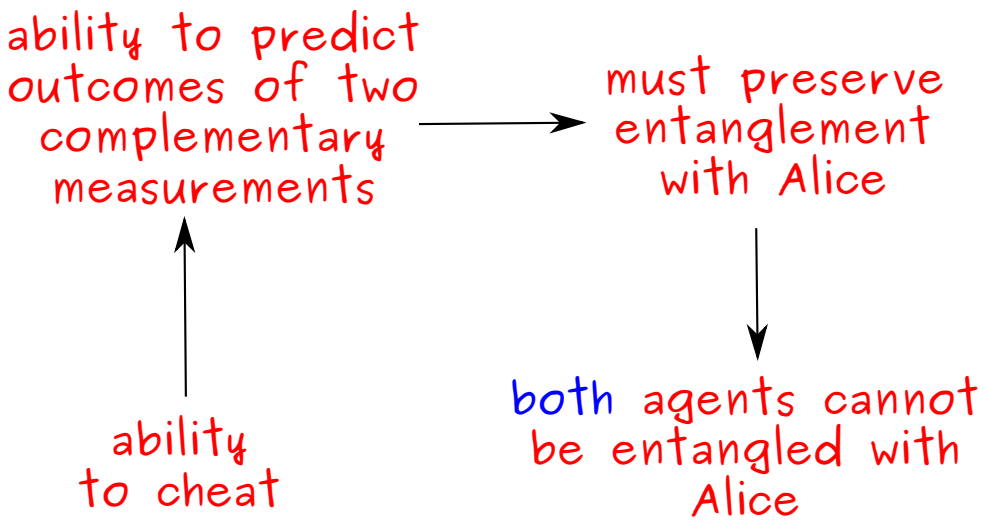
. . . and sends half of each to Bob.

Purified RBC [Kent'11]



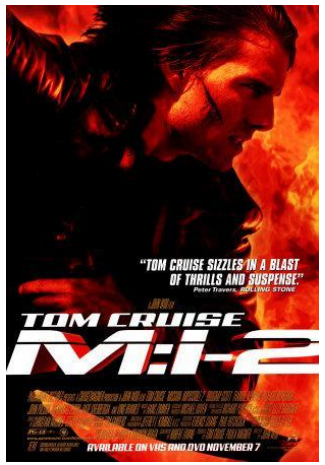
Bob applies an arbitrary isometry which splits the system into two parts. Each agent receives one of them.

RBC [Kent'11] – intuition why it is secure



RBC [Kent'11] – intuition why it is secure

ability to predict outcomes of two complementary measurements → must preserve entanglement with Alice → both agents cannot be entangled with Alice → ability to cheat



Outline

- Two-party cryptographic primitives
- Classical and quantum bit commitment
- Relativistic setting
- Relativistic bit commitment protocol [Kent'11]
- Security proof
- Summary and open questions

Security proof – no-signalling

			Charlie			
			$c = 0$		$c = 1$	
			accept	reject	reject	accept
Bob	$b = 0$	accept	p_0	a_{12}	$\cdot$	α
		reject	a_{21}	a_{22}	a_{23}	a_{24}
	$b = 1$	reject	$\cdot$	$\cdot$	$\cdot$	a_{34}
		accept	$\cdot$	$\cdot$	$\cdot$	p_1

Security proof – no-signalling

			Charlie			
			$c = 0$		$c = 1$	
			accept	reject	reject	accept
Bob	$b = 0$	accept	p_0	a_{12}	$\cdot$	α
		reject	a_{21}	a_{22}	a_{23}	a_{24}
	$b = 1$	reject	$\cdot$	$\cdot$	$\cdot$	a_{34}
		accept	$\cdot$	$\cdot$	$\cdot$	p_1

$$p_0 + p_1 \leq 1 + \alpha$$

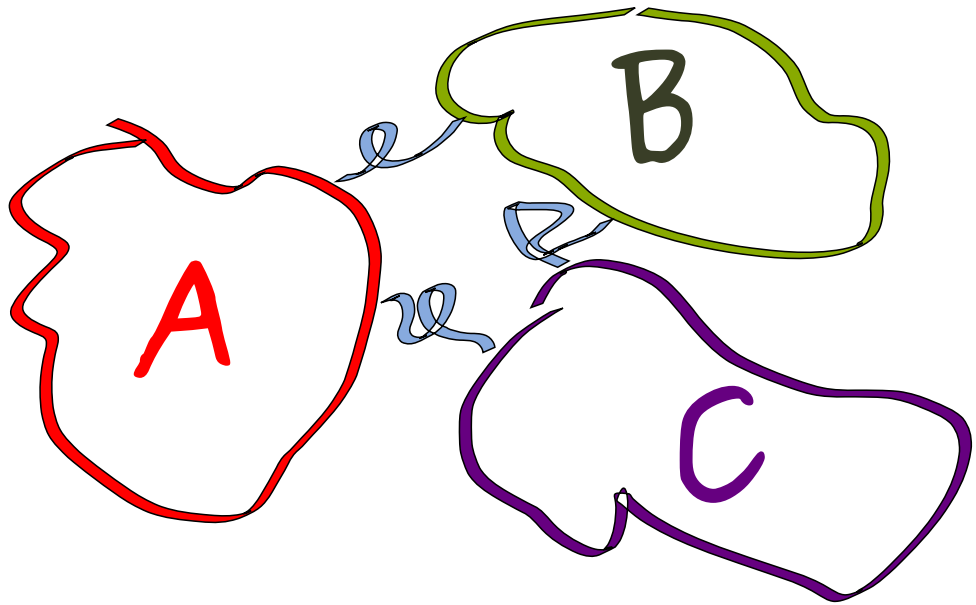
Security proof – no-signalling

			Charlie			
			$c = 0$		$c = 1$	
			accept	reject	reject	accept
Bob	$b = 0$	accept	p_0	a_{12}	$\cdot$	α
		reject	a_{21}	a_{22}	a_{23}	a_{24}
	$b = 1$	reject	$\cdot$	$\cdot$	$\cdot$	a_{34}
		accept	$\cdot$	$\cdot$	$\cdot$	p_1

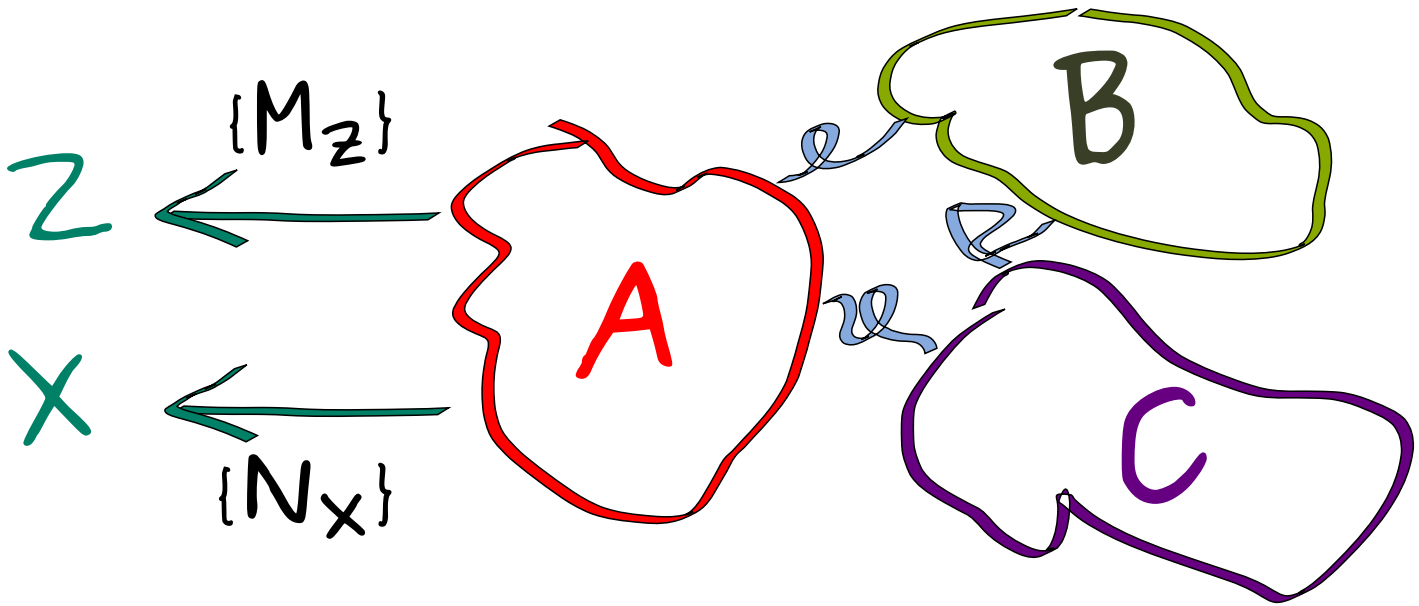
$$p_0 + p_1 \leq 1 + \alpha$$

GOAL : get a bound on α

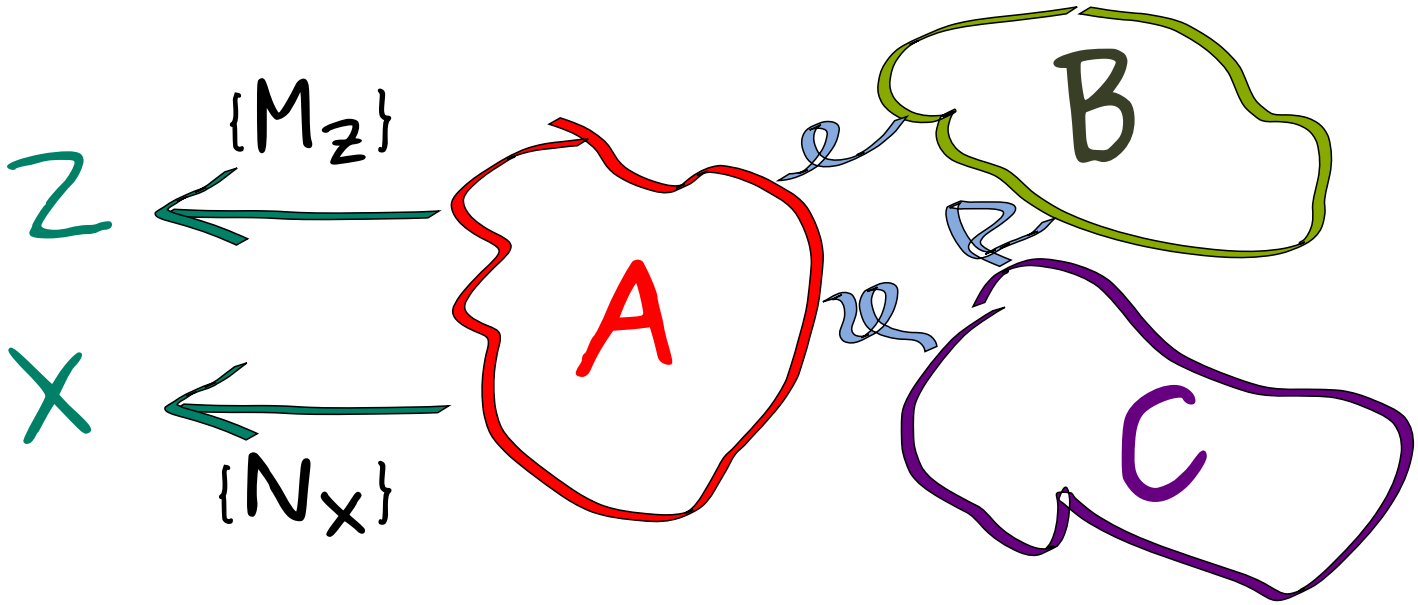
Security proof – uncertainty relation [TR'11]



Security proof – uncertainty relation [TR'11]



Security proof – uncertainty relation [TR'11]



$$\mathrm{H}_{\max}(Z|B) + \mathrm{H}_{\min}(X|C) \geq \log \frac{1}{c},$$

where $c := \max_{z,x} \|\sqrt{M_z}\sqrt{N_x}\|_\infty^2$.

Security proof – sketchy sketch

- red qubits – measure in Z to get Z_r
- green qubits – measured in X to get X_g

Security proof – sketchy sketch

- red qubits – measure in Z to get Z_r
- green qubits – measured in X to get X_g

α = Prob[Bob guesses Z_r AND Charlie guesses X_g]=
Prob[Bob guesses Z_r]*Prob[Charlie guesses X_g | Bob guesses Z_r]

Security proof – sketchy sketch

- red qubits – measure in Z to get Z_r
- green qubits – measured in X to get X_g

α = Prob[Bob guesses Z_r AND Charlie guesses X_g]=
Prob[Bob guesses Z_r]*Prob[Charlie guesses X_g | Bob guesses Z_r]

Bob is able to guess Z_r with high probability → his knowledge about Z_r must be significant

↓ (the sampling is random)

his knowledge about Z_g must also be significant → $H_{max}(Z_g|B)$ must be low

↓ (uncertainty relation)

$H_{min}(X_g|C)$ must be high → Charlie cannot guess X_g

Security proof – main result

Doing the maths properly gives

$$\alpha \leq 2^{1-n(1-h(\delta))} + 2\exp\left(-\frac{1}{2}n\delta^2\right),$$

for any $0 < \delta < \frac{1}{2} \implies$ **exponential decay**.

The fastest decay rate is achieved for $\delta \approx 0.33$, $\alpha \sim 2^{-0.08n}$

$$\alpha \approx 2^{-10} \iff n \approx 125.$$

Security proof – main result

Doing the maths properly gives

$$\alpha \leq 2^{1-n(1-h(\delta))} + 2\exp\left(-\frac{1}{2}n\delta^2\right),$$

for any $0 < \delta < \frac{1}{2}$ $\implies$ **exponential decay**.

The fastest decay rate is achieved for $\delta \approx 0.33$, $\alpha \sim 2^{-0.08n}$

$$\alpha \approx 2^{-10} \iff n \approx 125.$$

Outline

- Two-party cryptographic primitives
- Classical and quantum bit commitment
- Relativistic setting
- Relativistic bit commitment protocol [Kent'11]
- Security proof
- Summary and open questions

Summary

- in the split model with two Bobs in the open phase a new issue of extreme importance arises – **global vs. local command**,
- in the local command a classical, trivial protocol gives unconditional security,
- in the global command no classical protocol can be secure,
- RBC [Kent'11] can be proven **secure in the global command** and we provide explicit security bounds.

Summary

- in the split model with two Bobs in the open phase a new issue of extreme importance arises – **global vs. local command**,
- in the local command a classical, trivial protocol gives unconditional security,
- in the global command no classical protocol can be secure,
- RBC [Kent'11] can be proven **secure in the global command** and we provide explicit security bounds.

Summary

- in the split model with two Bobs in the open phase a new issue of extreme importance arises – **global vs. local command**,
- in the local command a classical, trivial protocol gives unconditional security,
- in the global command no classical protocol can be secure,
- RBC [Kent'11] can be proven **secure in the global command** and we provide explicit security bounds.

Summary

- in the split model with two Bobs in the open phase a new issue of extreme importance arises – **global vs. local command**,
- in the local command a classical, trivial protocol gives unconditional security,
- in the global command no classical protocol can be secure,
- RBC [Kent'11] can be proven **secure in the global command** and we provide explicit security bounds.

Open questions

- the current security bounds are very far from the best attack we can think of... maybe someone could try to **close the gap**?
- what about introducing some **noise tolerance**? (crucial if we think about doing an experiment)
- we know that RBC cannot be universally composable but maybe some weaker notion of **composability** holds. can we get **string commitment** by executing it multiple times (sequentially or in parallel)?

Open questions

- the current security bounds are very far from the best attack we can think of... maybe someone could try to **close the gap**?
- what about introducing some **noise tolerance**? (crucial if we think about doing an experiment)
- we know that RBC cannot be universally composable but maybe some weaker notion of **composability** holds. can we get **string commitment** by executing it multiple times (sequentially or in parallel)?

Open questions

- the current security bounds are very far from the best attack we can think of... maybe someone could try to **close the gap**?
- what about introducing some **noise tolerance**? (crucial if we think about doing an experiment)
- we know that RBC cannot be universally composable but maybe some weaker notion of **composability** holds. can we get **string commitment** by executing it multiple times (sequentially or in parallel)?