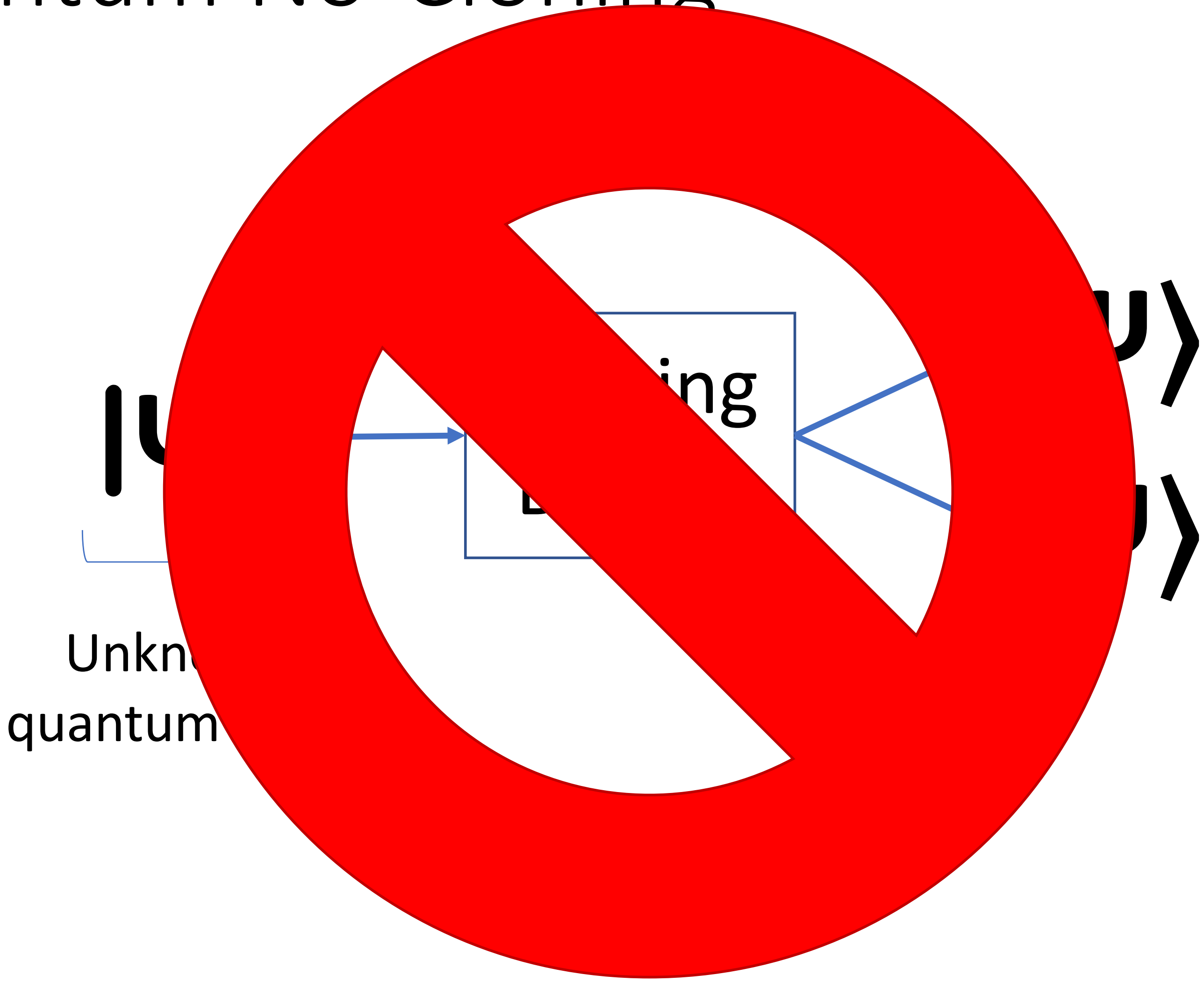
Quantum Lightning Never Strikes the Same State Twice

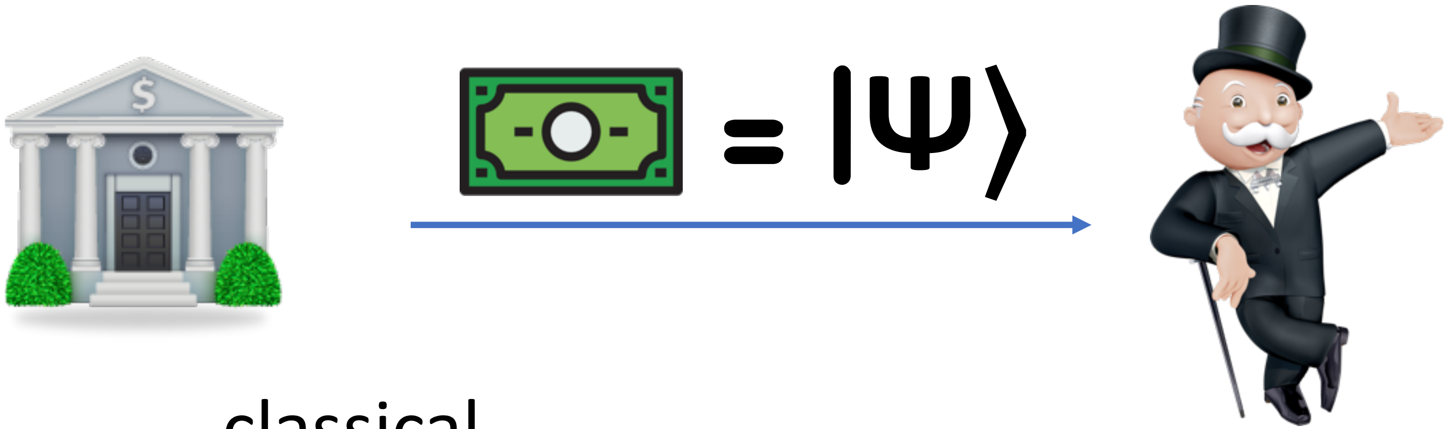
Mark Zhandry

Princeton University

Quantum No-Cloning



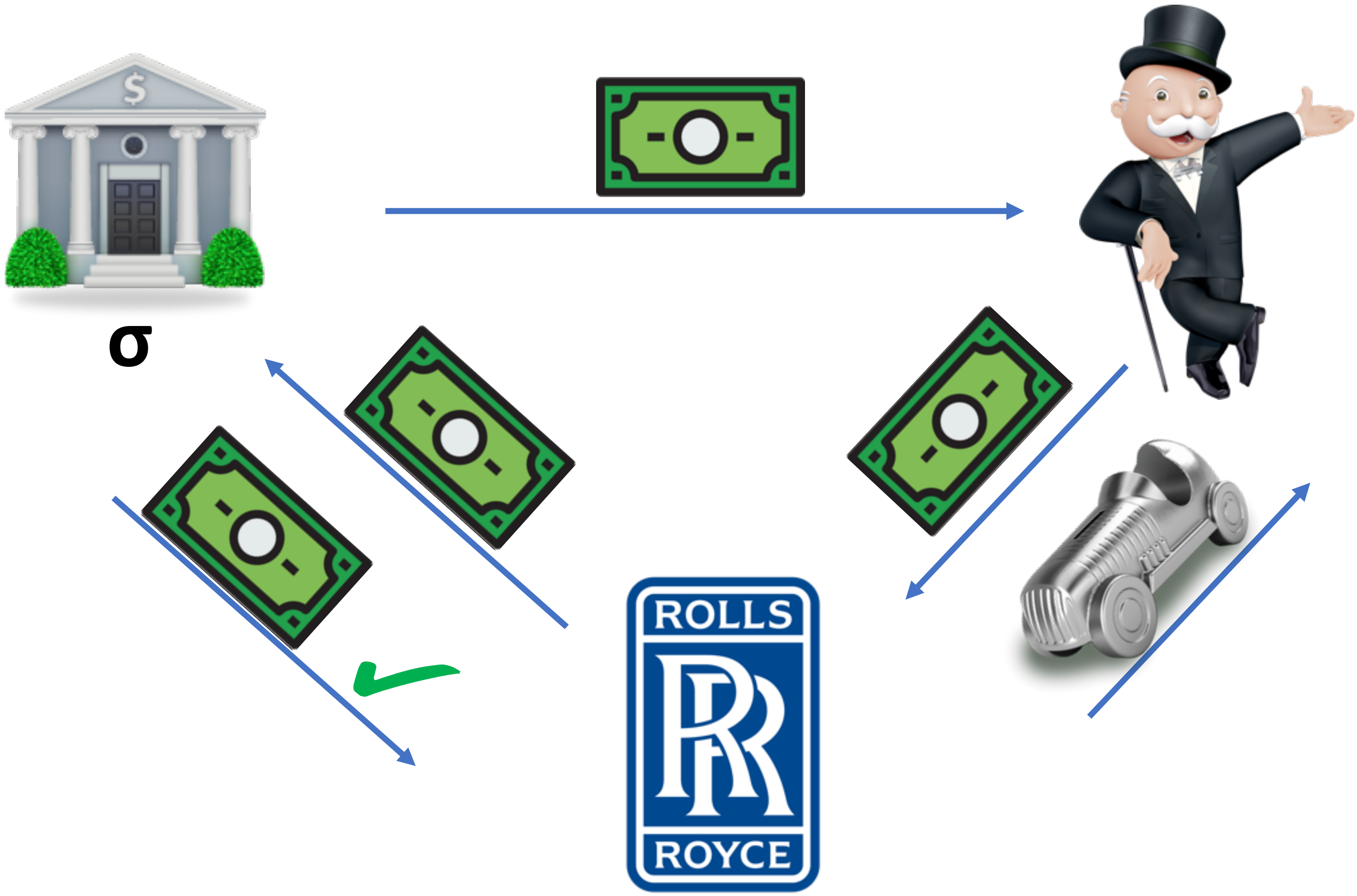
No-Cloning = Quantum Money [Wiesner'70]



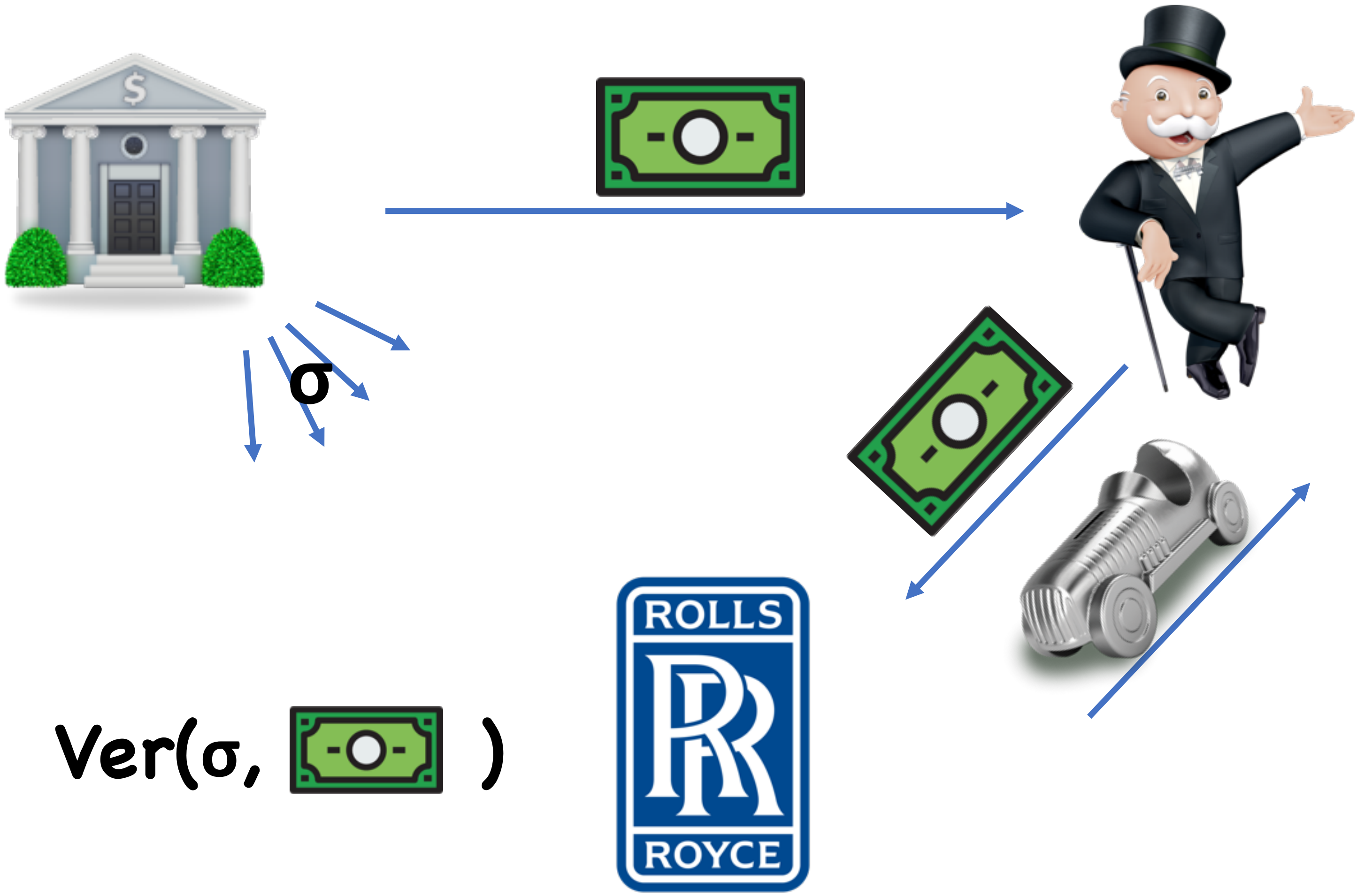
💵 = $|\Psi\rangle$

$$\underbrace{\text{Serial \# = classical description}}_{\text{Kept secret}}$$

Limits of (Plain) Quantum Money



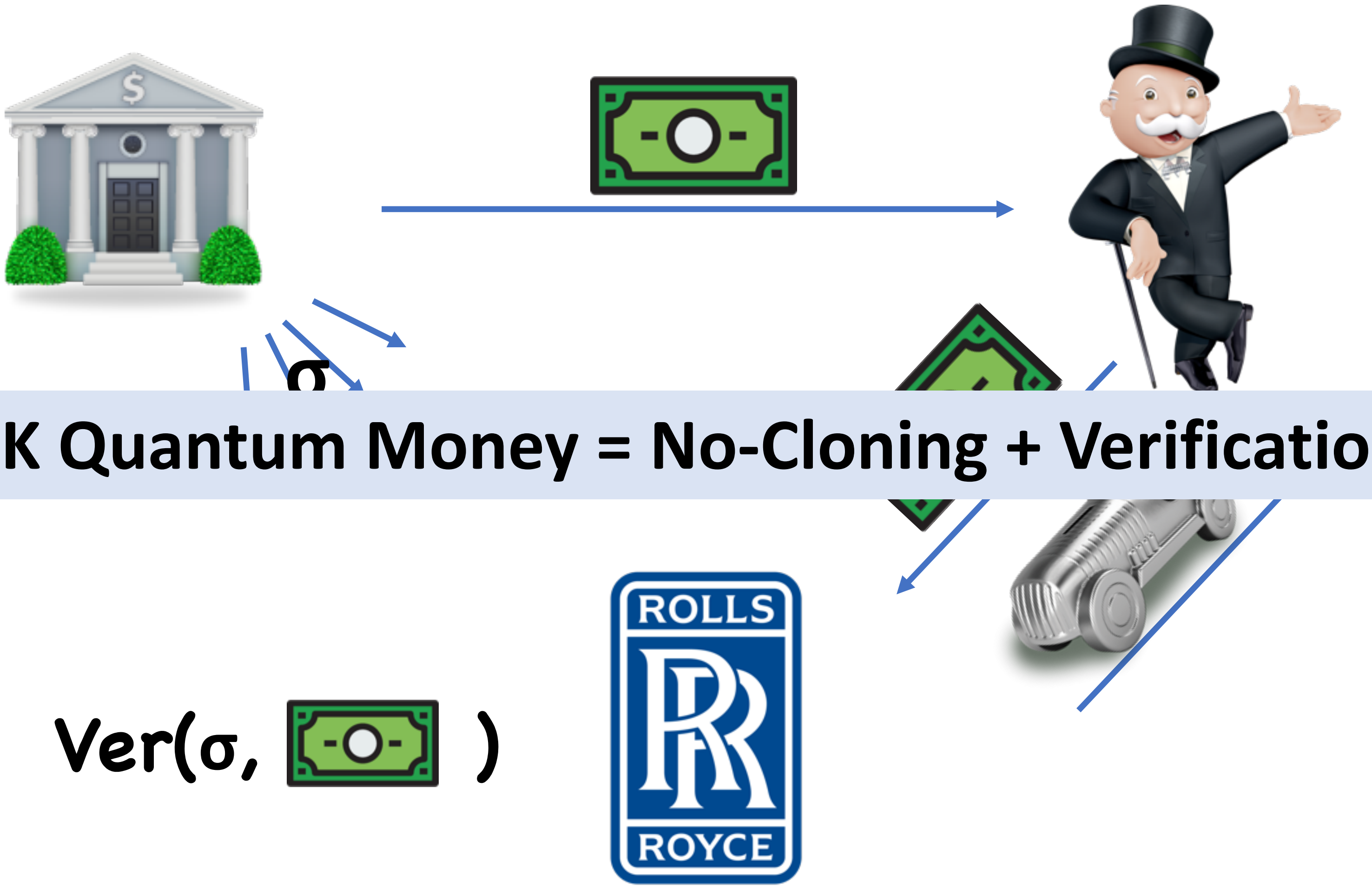
Public Key Quantum Money [Aaronson'09]



σ

Ver(σ, 💵)

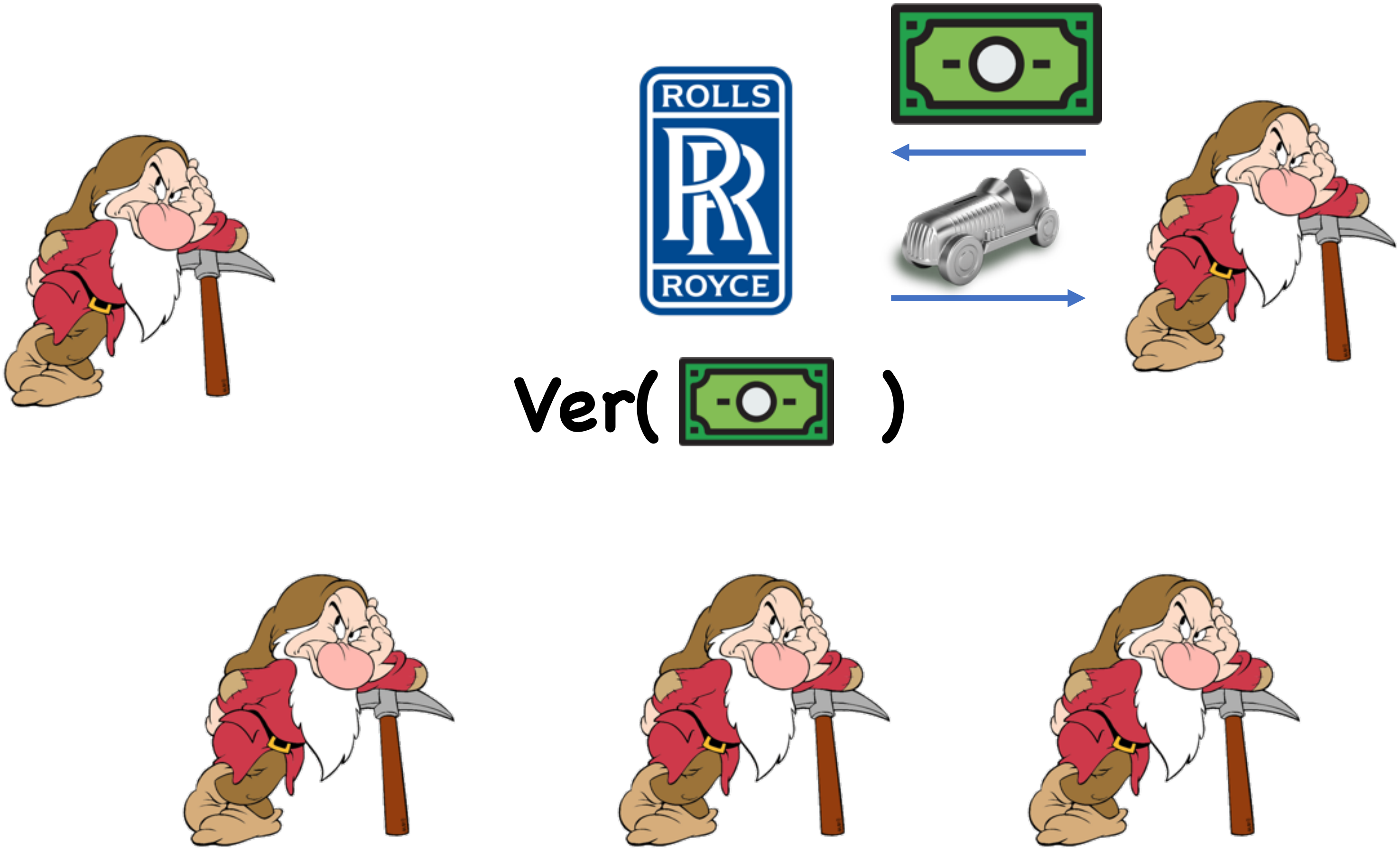
Public Key Quantum Money [Aaronson'09]



PK Quantum Money = No-Cloning + Verification

Ver(σ, 🟩)

Bitcoin sans Blockchain?

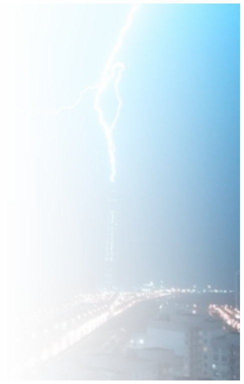


Ver(💵)

Lightning Never Strikes Same Place Twice?

Let's pretend

Of course, ca



Quantum lig **adversarial thunderstor**

- Impossible c ditions

Did Germany already invent quantum lightning?

Quantum Lightning

Applications:

- PK Quantum money

- Decentralized currency

💵 = ⚡ s.t. $H(\sigma)=0^n\{0,1\}^*$

- Provable min-entropy

⚡ proves that **σ** has min-entropy

Constructions?

PK quantum money?

- [Aaronson’09]: (1) relative to **Quantum** oracle, (2) concrete candidate instantiation
    - (2) broken by [Lutomirski-Aaronson-Farhi-Gosset-Kelner-Hassidim-Shor’10]
- [Farhi-Gosset-Hassidim-Lutomirski-Shor’12]: from knots
- [Aaronson-Christiano’12]: (1) relative to **Classical** oracle, (2) concrete candidate instantiation
    - (2) broken by [Pena-Faugère-Perret’15]

Quantum Lightning?

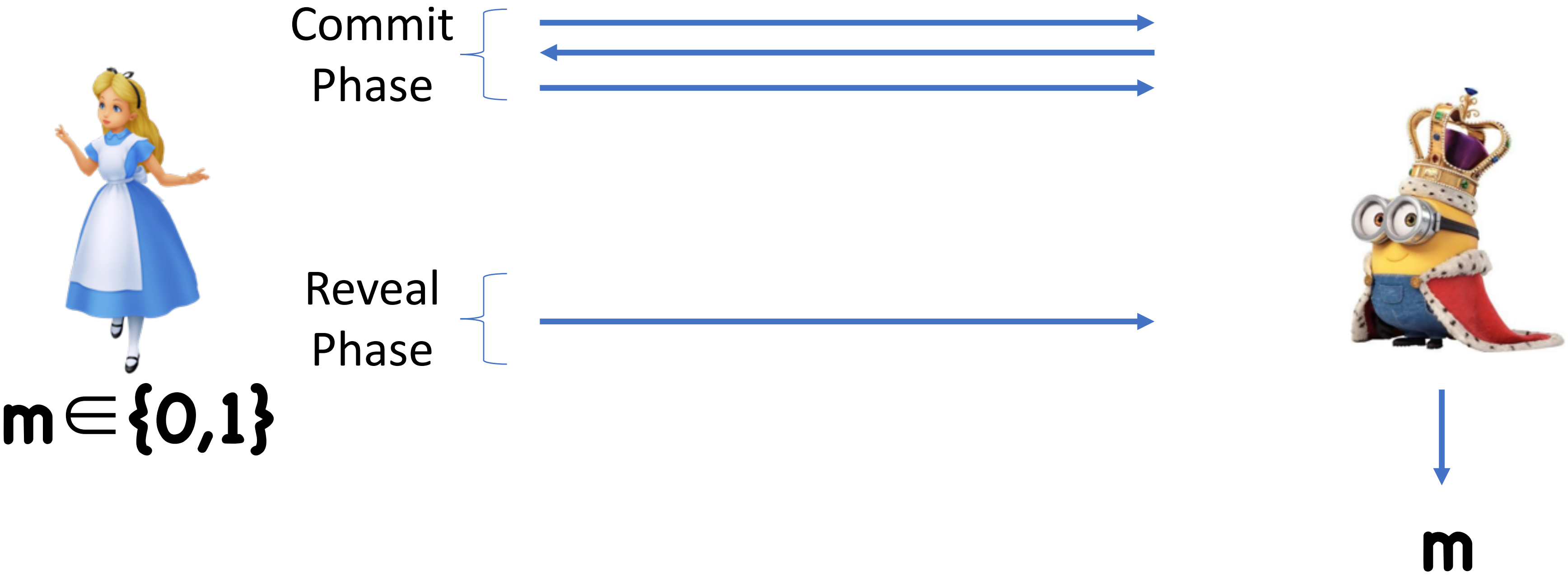
- [Lutomirski-Aaronson-Farhi- Gosset-Hassidim-Kelner-Shor’09]: “collision-free” QM
    - Already believed insecure

This work: study strong variants of no cloning

- **New constructions**
- **Connections to post-quantum security**

Detour:
Classical crypto in a quantum world

(Bit) Commitment Schemes



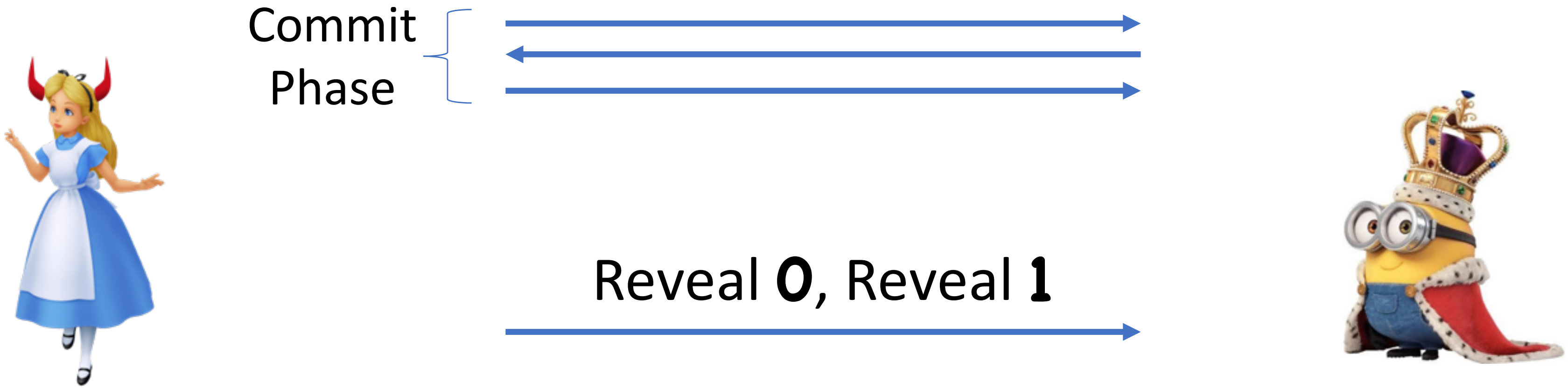
Commit Phase

Reveal Phase

m∈{0,1}

m

Binding



Commit Phase

Reveal **0**, Reveal **1**

Limitation

Security goal: once Alice commits, there is a unique message she can de-commit to

Typical security notion: once Alice commits, she cannot *simultaneously* de-commit to both $\mathbf{0}$ and $\mathbf{1}$

Classically, these two goals are the same (use rewinding), but quantumly, they may not be

Limitation: Quantum Rewinding

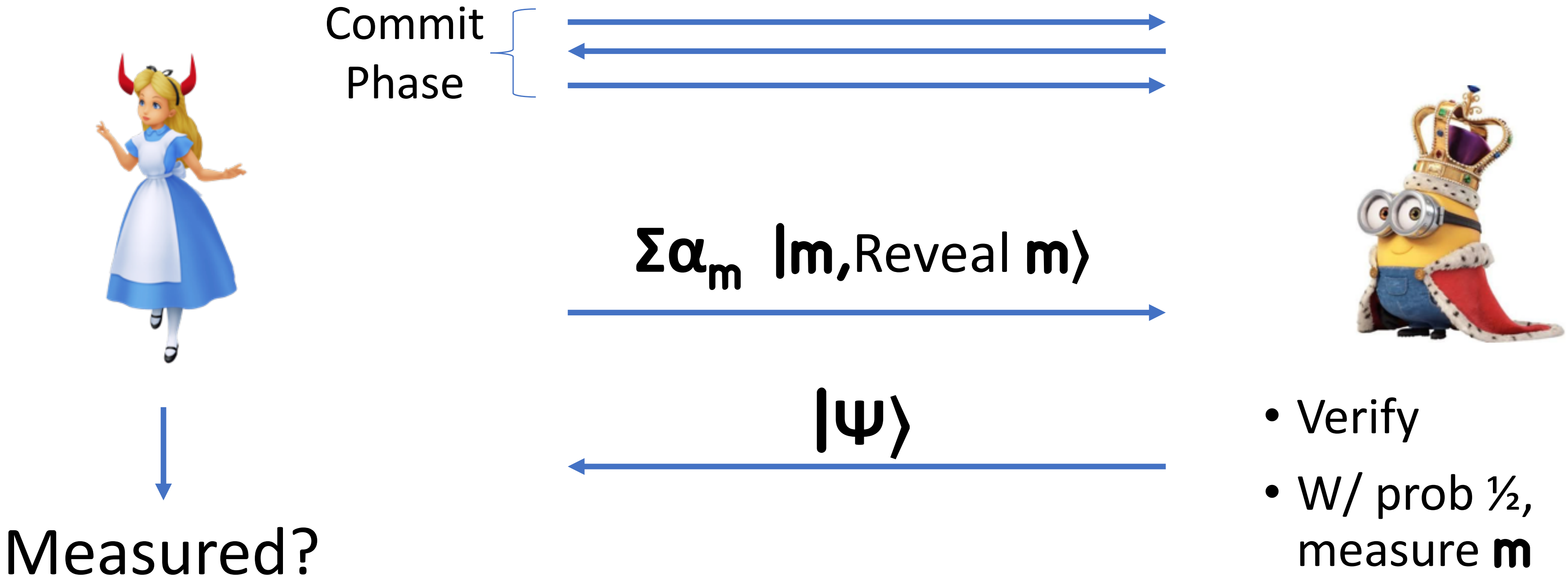
Intuition:

- Alice may keep a quantum state that allows her to decommit to either **0** or **1**

- Once she decommits to, say, **0**, she must measure to get classical decommitment ⇒ state collapses

- Cannot no longer rewind to evaluate on **1**

Solution: Collapse-Binding [Unruh'16]



Commit Phase

$\Sigma\alpha_m\ |\mathbf{m}, \text{Reveal } \mathbf{m}\rangle$

$|\psi\rangle$

Measured?

- Verify
- W/ prob ½, measure **m**

Is this really a problem?

Thm [Ambainis-Rosmanis-Unruh’14**]:** Relative to a quantum oracle, there exists a commitment scheme that is classically binding, but an efficient quantum adversary can de-commit to either **0** or **1**

What’s this got to do with no-cloning?

Either/Or Results

Thm (Informal): A **binding** commitment is either **collapse binding**, or can be used to build public key quantum money.

Thm (Informal): A *non-interactive* **binding** commitment is either **collapse binding**, or can be used to build quantum lightning.

Also show analogous statements for digital signatures, hash functions

Intuition

Thm (Informal): A **binding** commitment is either **collapse binding**, or can be used to build public key quantum money.

What if we could clone adversary's state?

- Then no need to rewind, definitions equivalent

So any separation inherently uses no-cloning

- Banknote/bolt = adversary's state
- For verification, check that adversary breaks collapse-binding

Takeaways

Two possible interpretations:

(1) Quantum money/lightning is hard, so probably don't have to worry about these quantum security issues for most schemes

(At this point, still no concrete separation)

(2) Possible route toward building quantum money/lightning

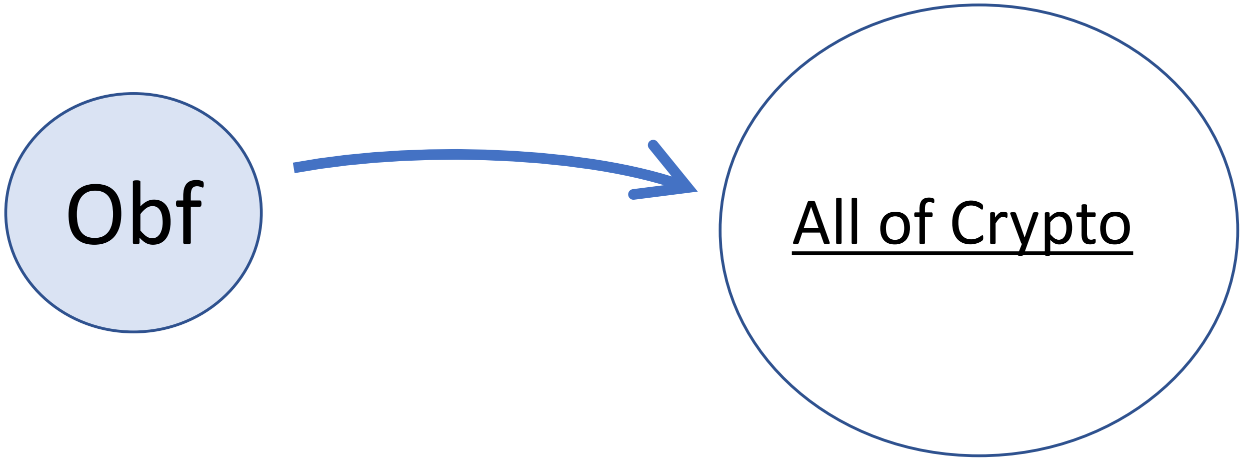
New Constructions of Quantum Money/Lightning

Program Obfuscation

“Scramble” a program

- Hide implementation details
- Maintain functionality

Golden goose of crypto, believed by many to be “crypto complete”

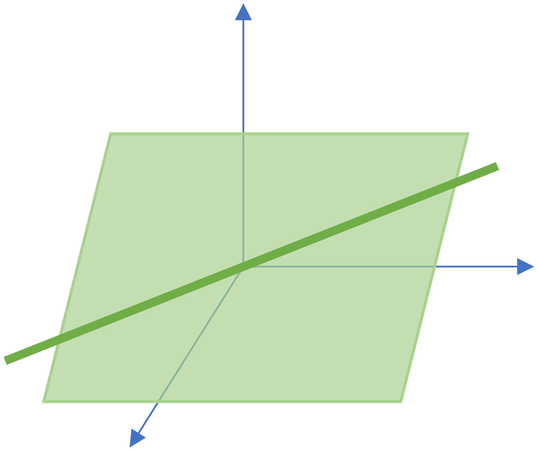


PKQM from Obfuscation

Thm: Indistinguishability obfuscation ⇒ PKQM

= **Lem:** Subspace hiding obfuscation ⇒ PKQM **+** **Lem:** Indist. obf ⇒ Sub. hiding obf

Subspace hiding obfuscation:



T = random subspace of $\mathbf{F}^n$

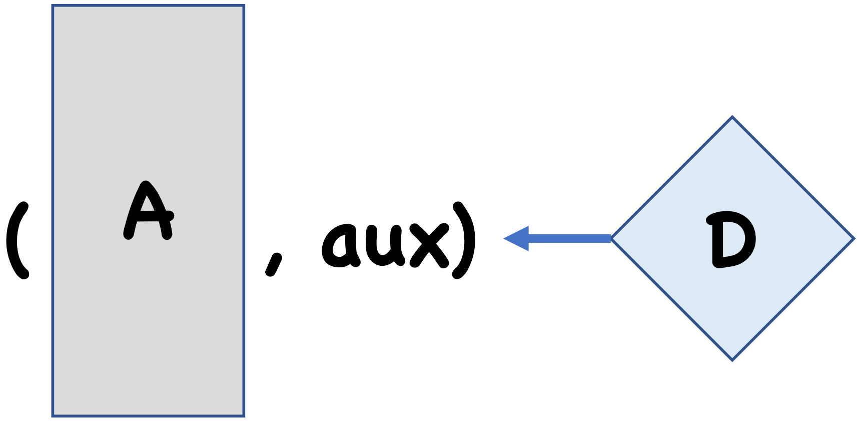
S = random subspace of **T**

$$(\mathbf{S}, \mathbf{Obf}(\mathbf{S})) \approx_c (\mathbf{S}, \mathbf{Obf}(\mathbf{T}))$$

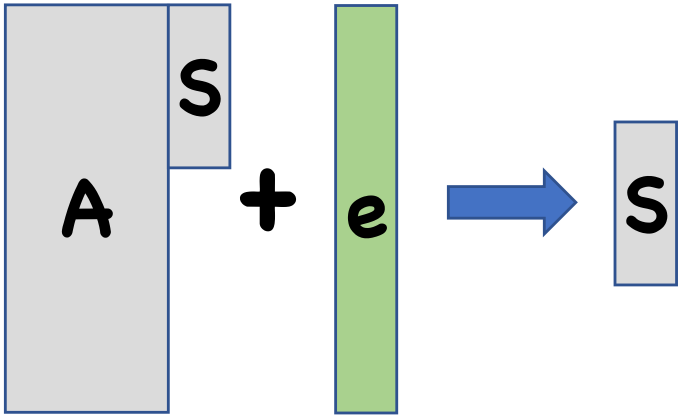
Quantum Lightning from LWE?

Lem: “Gap LWE” $\Rightarrow$ Quantum Lightning

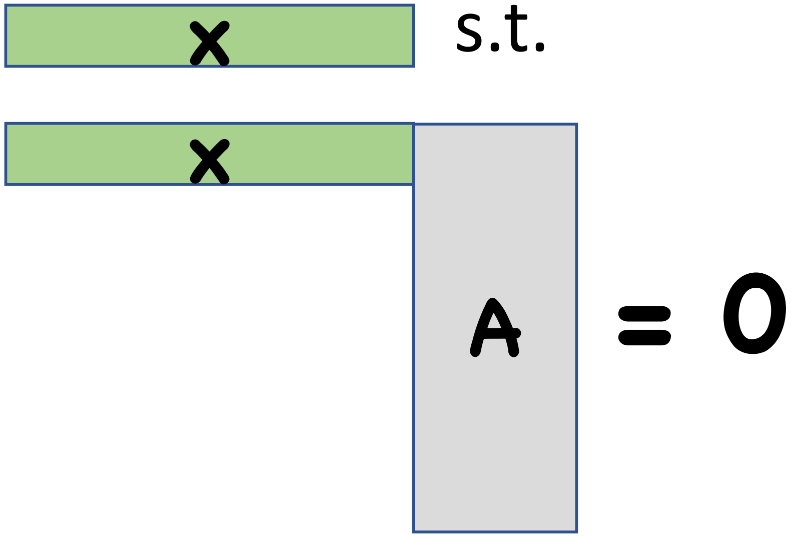
"Gap LWE"



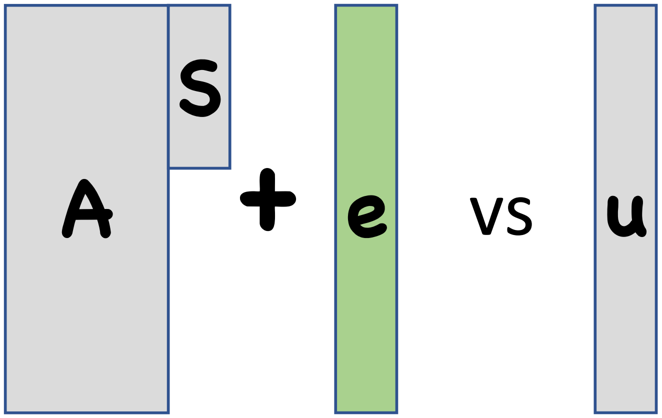
S-LWE **Hard**:



SIS **Hard**:



D-LWE **Easy**:



🟩 = short vector

Constructing Quantum Lightning

Don’t know how to construct “gap LWE”

Instead, give candidate modification where L2 norm is replaced with “rank norm”

- Rank norm SIS is actually easy [Ding-Yang’08, Applebaum-Haramaty-Ishai-Kushilevitz-Vaikuntanathan’17]
- Many annoying details to get plausible instantiations
- Broken in some settings [Leander-Rasoolzadeh-Wiemer’19, Roberts’19], more work needed to find and verify secure instance

Thanks!