Private Proofs of When and Where

Feasibility and Applications of Zero-Knowledge Position Verification

Leo Orshansky (Columbia)

Uma Girish

Columbia

Grzegorz Gluch

UC Berkeley/MIT

Shafi Goldwasser

UC Berkeley/MIT

Tal Malkin

Columbia

Henry Yuen

Columbia

Talk Teaser

Previous PV Protocols

Our ZK-PV Protocol

Talk Teaser

Previous PV Protocols

Can prove one thing: instantaneous position.

"I am in location XYZ right now."

Our ZK-PV Protocol

Many new types of statements —

"I am **either** in New York or in Sydney."

Talk Teaser

Previous PV Protocols

Can prove one thing: instantaneous position.

"I am in location XYZ right now."

Our ZK-PV Protocol

Many new types of statements —

"I am **either** in New York or in Sydney."

"I was **not** near location XYZ anytime today"

Talk Teaser

Previous PV Protocols

Can prove one thing: instantaneous position.

"I am in location XYZ right now."

Only a **single point** in spacetime is allowed.

Our ZK-PV Protocol

Many new types of statements —

"I am **either** in New York or in Sydney."

"I was **not** near location XYZ anytime today"

General **spacetime region** of allowed points.

Talk Teaser

Previous PV Protocols

Can prove one thing: instantaneous position.

"I am in location XYZ right now."

Only a **single point** in spacetime is allowed.

Prover's location is a public parameter of the protocol.

Our ZK-PV Protocol

Many new types of statements —

"I am **either** in New York or in Sydney."

"I was **not** near location XYZ anytime today"

General **spacetime region** of allowed points.

Prover's location is totally hidden from (honest) verifiers.

Background: Position Verification

- What's the problem we are solving?
- How does quantum help us?

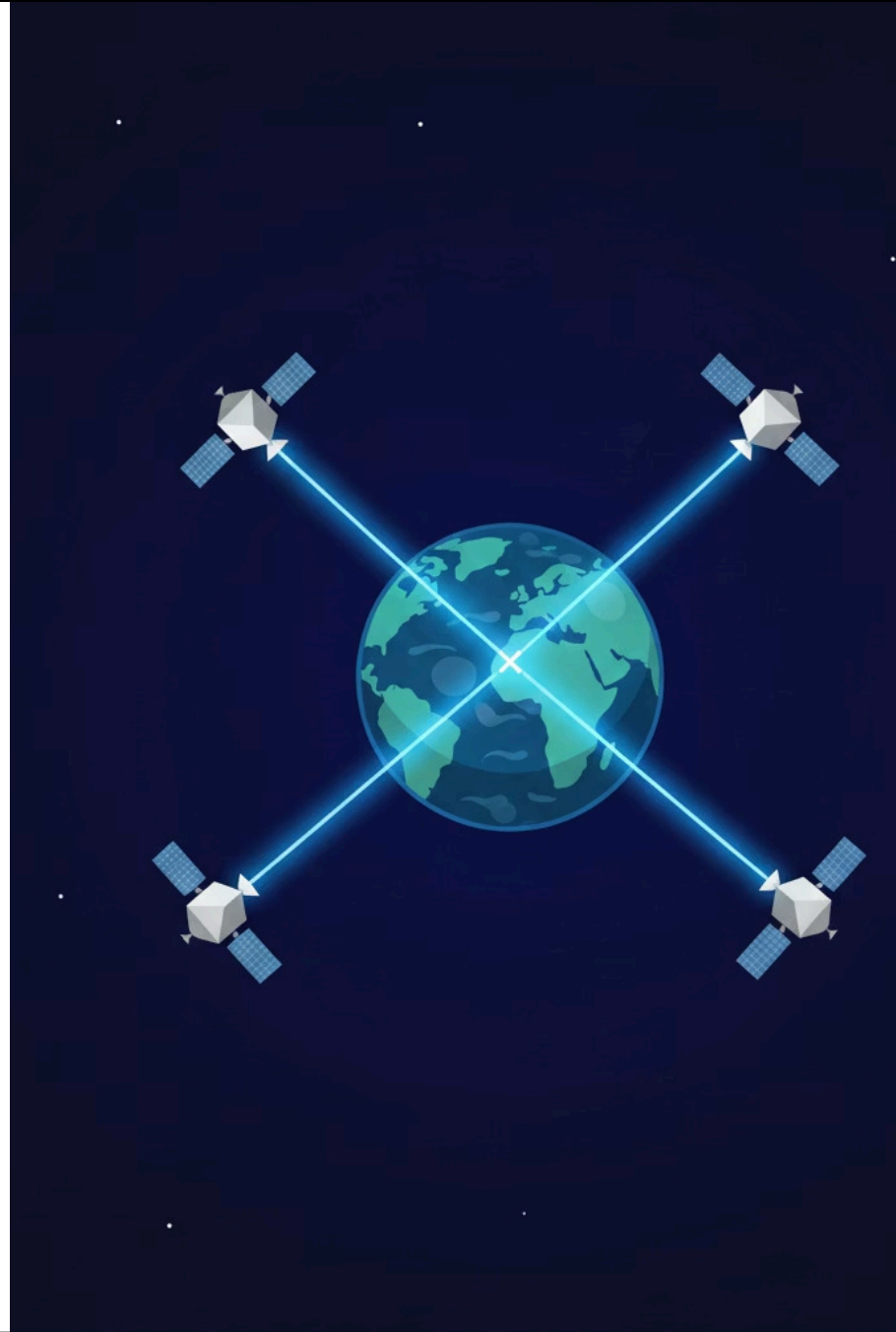
Position Verification [CGMO'09]

Provably Secure GPS / Triangulation

- An untrusted **prover** interacts with a trusted set of **verifiers** to prove its whereabouts.
- Light-speed timings ensure that the prover cannot lie.

Anti-Spoofing Security

- A malicious coalition of **spoofers** might try to pretend they are a single prover in a **different location**.
- **Hard Part:** Achieving soundness against arbitrary prover coalitions



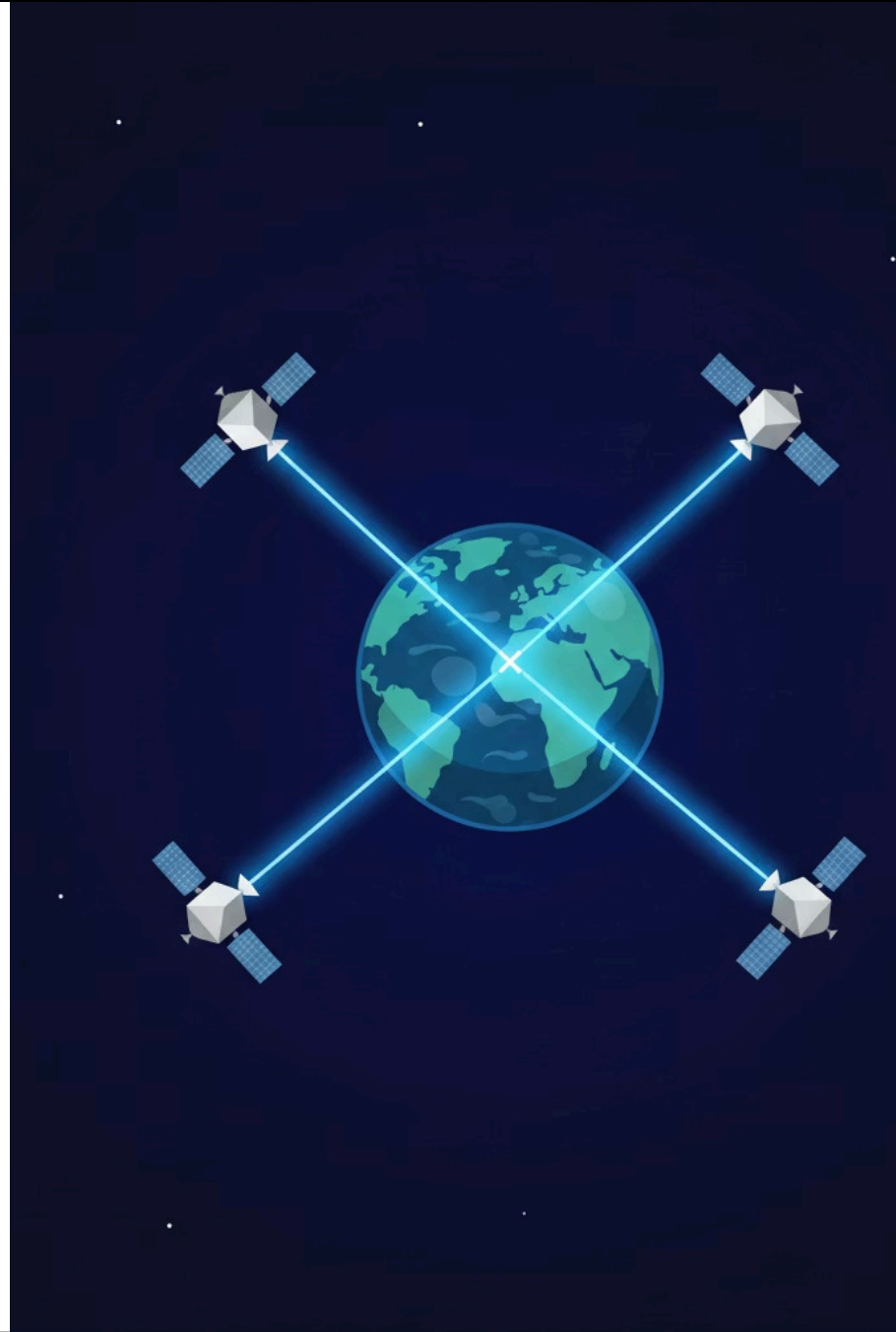
Position Verification [CGMO'09]

Provably Secure GPS / Triangulation

- An untrusted **prover** interacts with a trusted set of **verifiers** to prove its whereabouts.
- Light-speed timings ensure that the prover cannot lie.

Anti-Spoofing Security

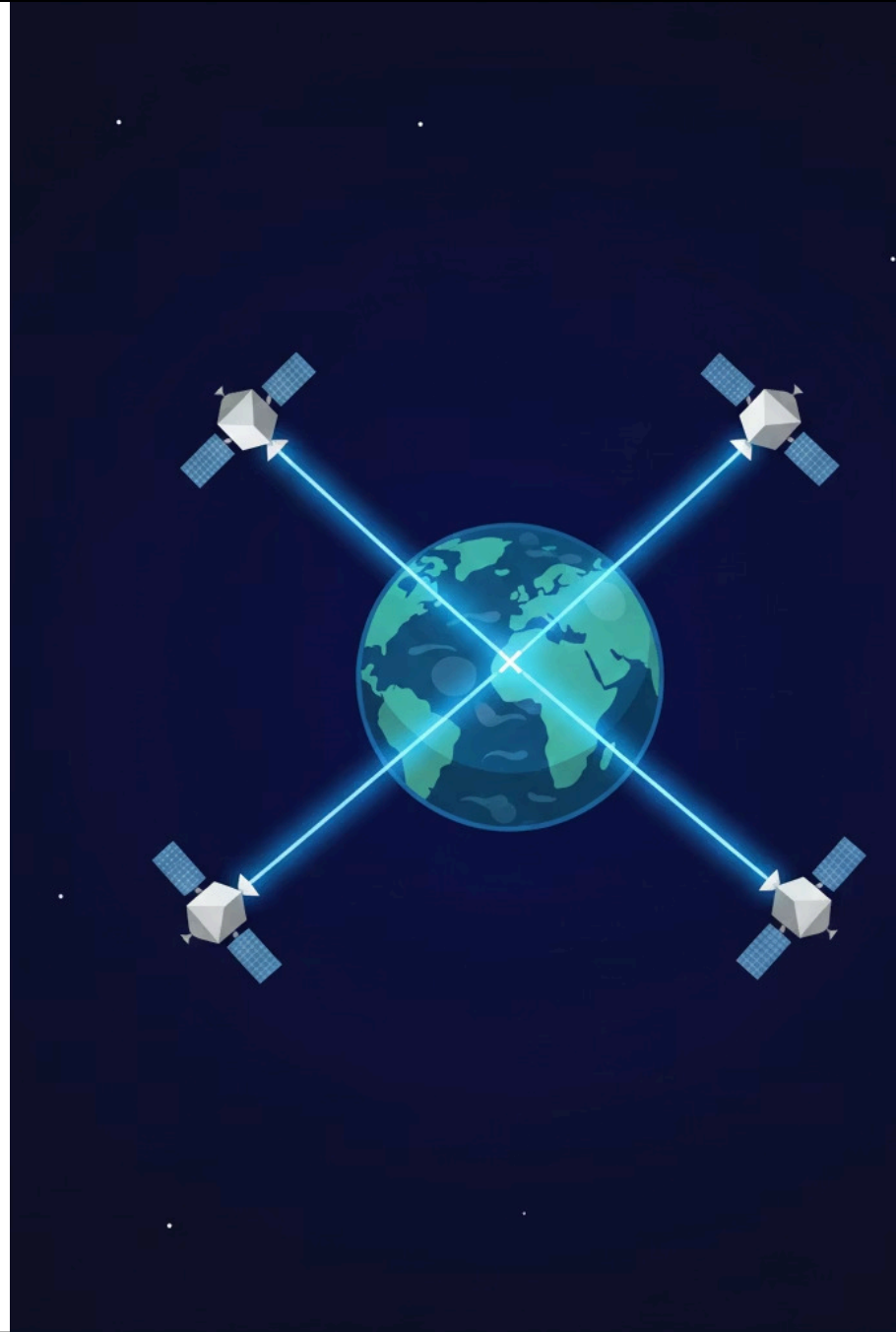
- A malicious coalition of **spoofers** might try to pretend they are a single prover in a **different location**.
- **Hard Part:** Achieving soundness against arbitrary prover coalitions



Position Verification [CGMO'09]

Provably Secure Triangulation

- Use light-speed timing constraints to narrow down the prover's position to a single point in space.
- A malicious coalition of **spoofers** might try to impersonate a single prover elsewhere.
- **Hard Part:** Achieving soundness against spoofer coalitions with arbitrary resources



Position-Based Cryptography

Position-Based Authentication [BCFGGOS'10, U'14]

Messages are "signed" with the sender's true position

Position-Based Key Agreement [CGMO'09]

- A secret key accessible only to someone in a particular location
- The verifiers can then encrypt messages with this key, which can only be decrypted by someone physically in that location

Position-Based Cryptography

Position-Based Authentication [BCFGGOS'10, U'14]

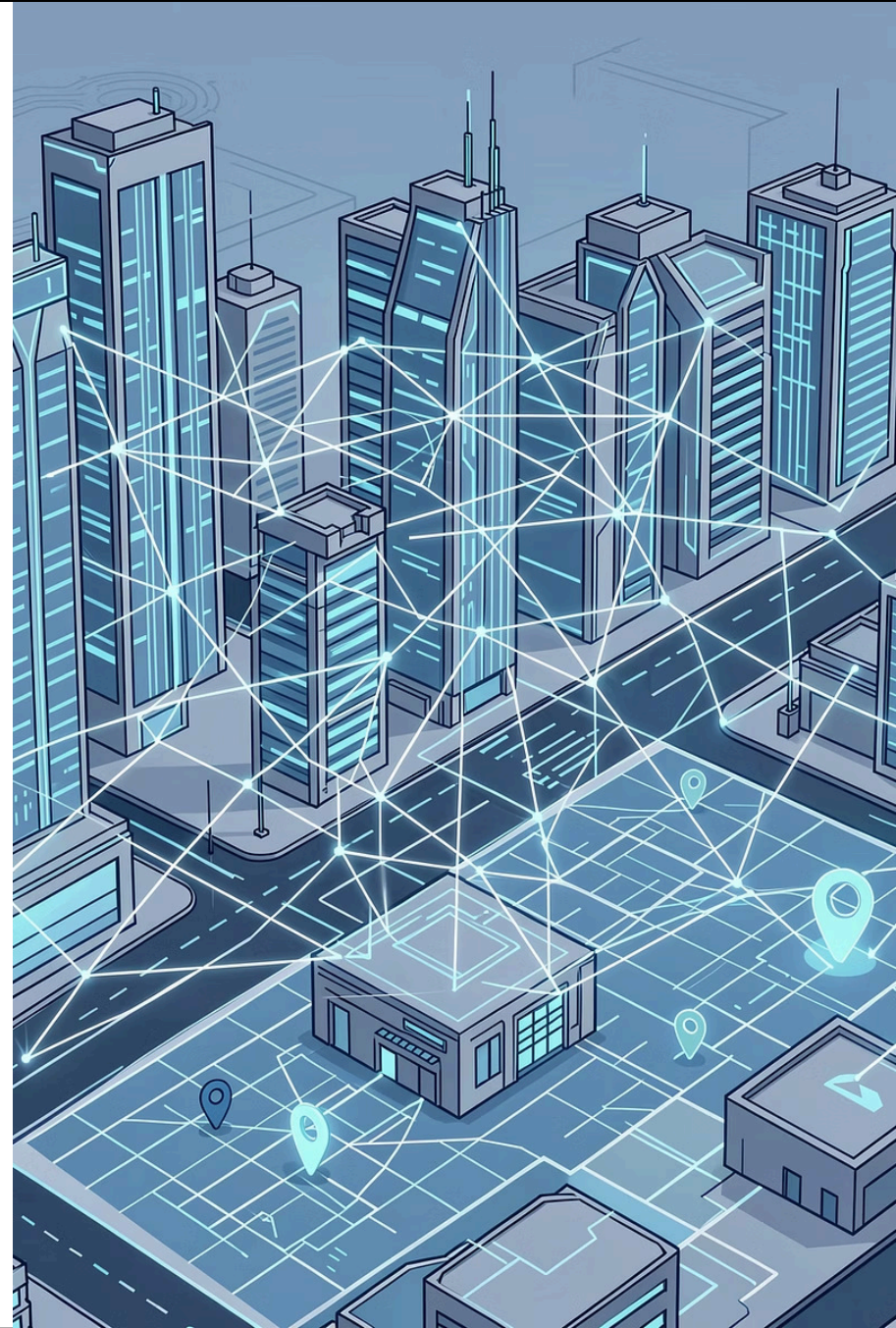
Messages are "signed" with the sender's true position

Position-Based Key Agreement [CGMO'09]

- A secret key accessible only to someone in a particular location
- The verifiers can then encrypt messages with this key, which can only be decrypted by someone physically in that location

Wide-Ranging Applications

- **Authentication/encryption for secure facilities**
- Manufacturer geo-restriction of devices (e.g. GPUs)
- Certifiably original news reporting

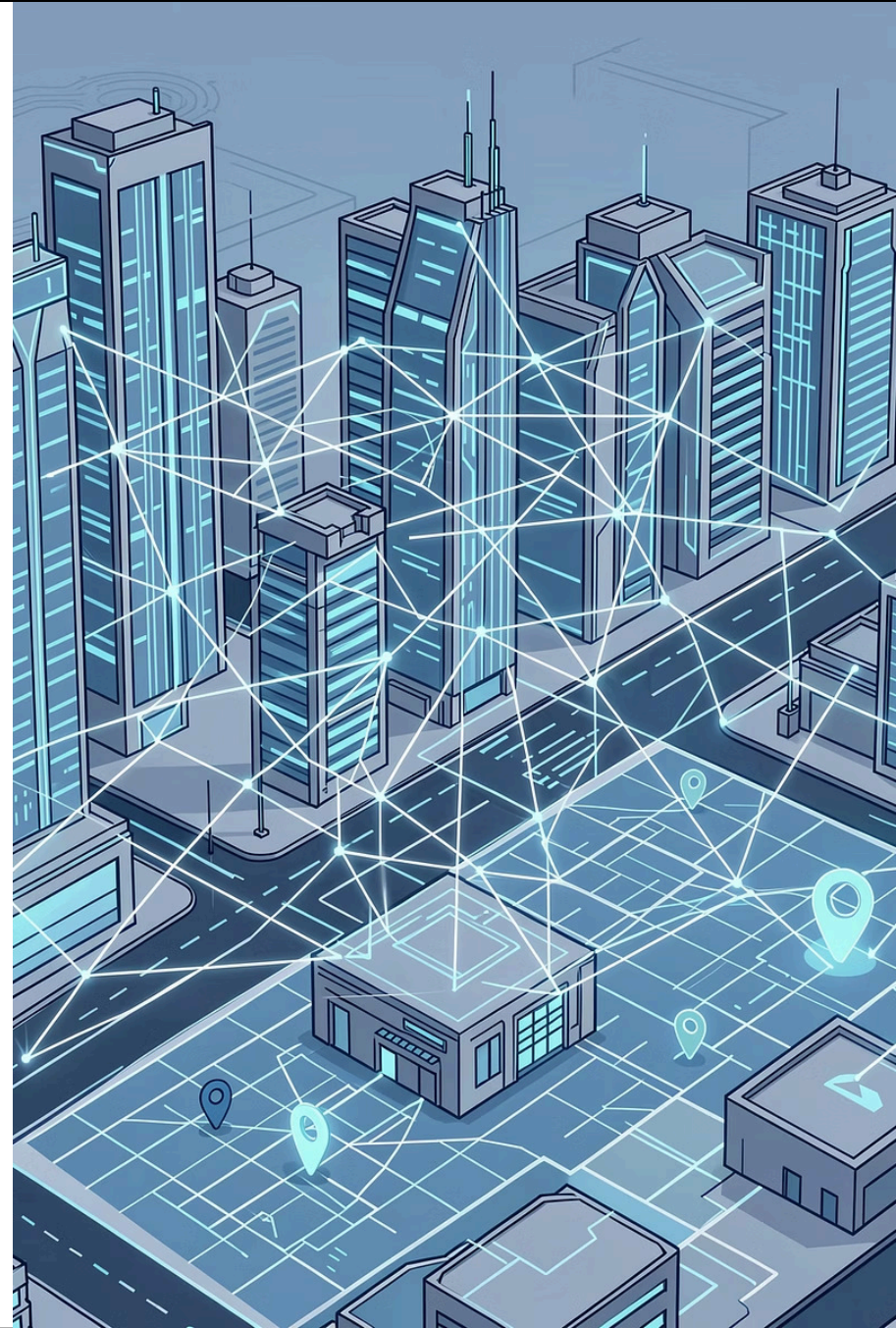


Wide-Ranging Applications

Authentication/encryption for secure facilities

Manufacturer geo-restriction of devices (e.g. GPUs)

Certifiably original news reporting



Wide-Ranging Applications

Authentication/encryption for secure facilities

Manufacturer geo-restriction of devices (e.g. GPUs)

Certifiably original news reporting

(Quantum) Position Verification

As it turns out, secure position verification cannot be constructed from classical means!

Classical No-Go

In any classical scheme, spoofers can:

1. Surround the claimed location.
2. Relay all messages **across** the claimed point to their colluding partners.
3. Solve the verifiers' challenges **at a number of distant points simultaneously**, in order to meet timing constraints.

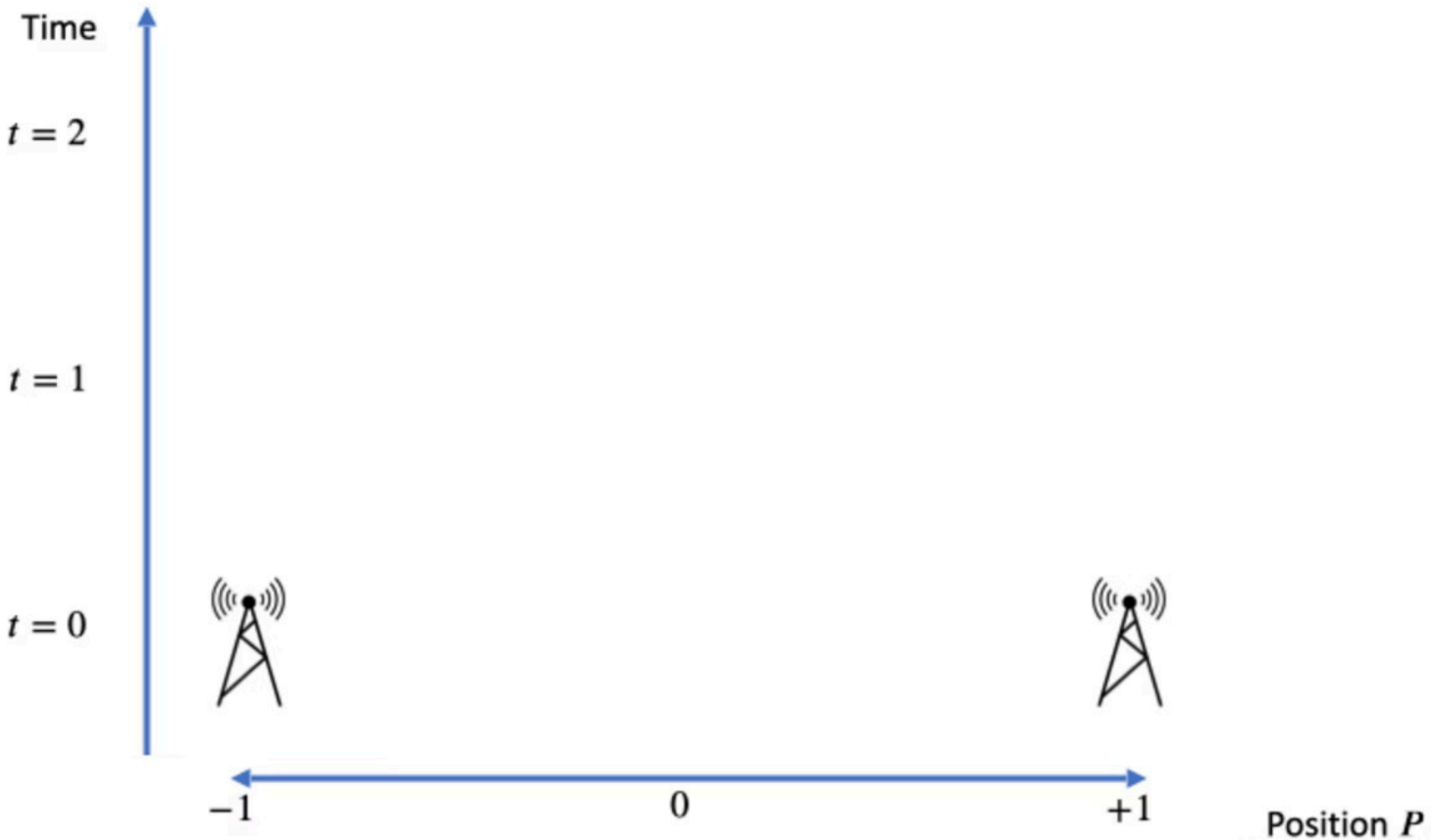
Quantum Solution: Unclonability

In the aforementioned attack, a key tool is the ability to **replicate** information across points in space.

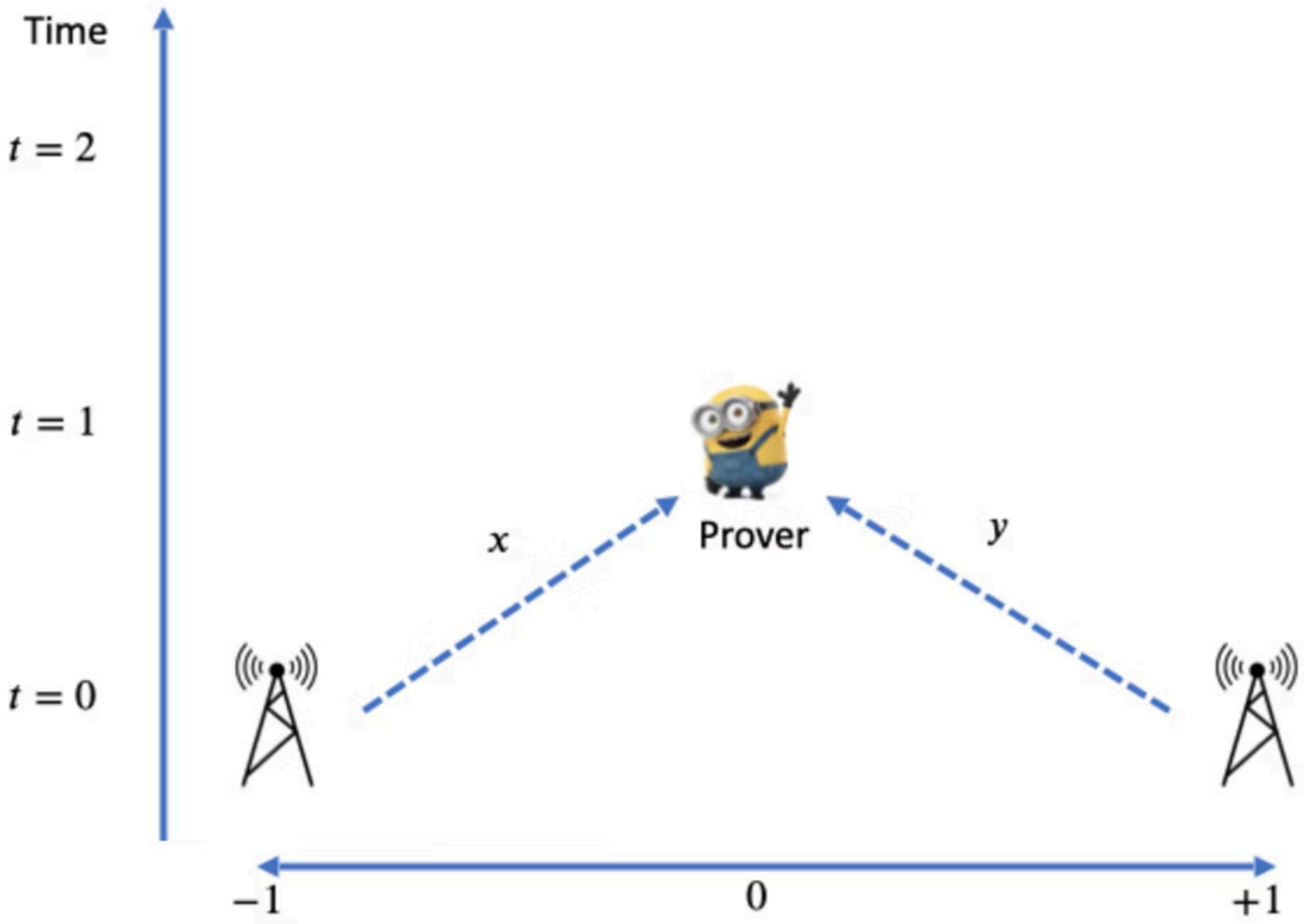
Quantum No-Cloning Theorem:

There is no physical procedure to produce two copies of an unknown quantum state.

Example: 1D position verification



Example: 1D position verification

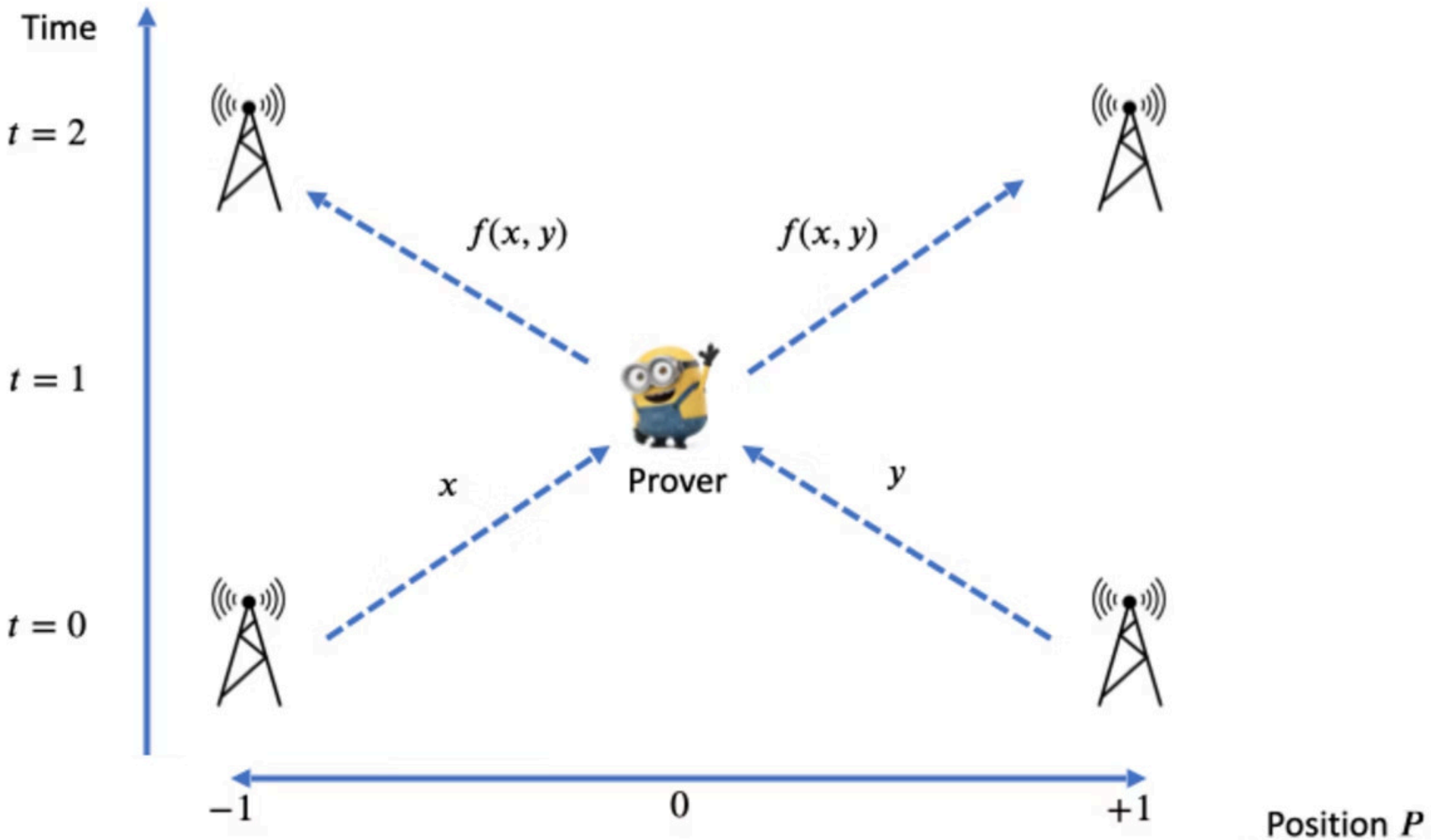


Model Assumptions

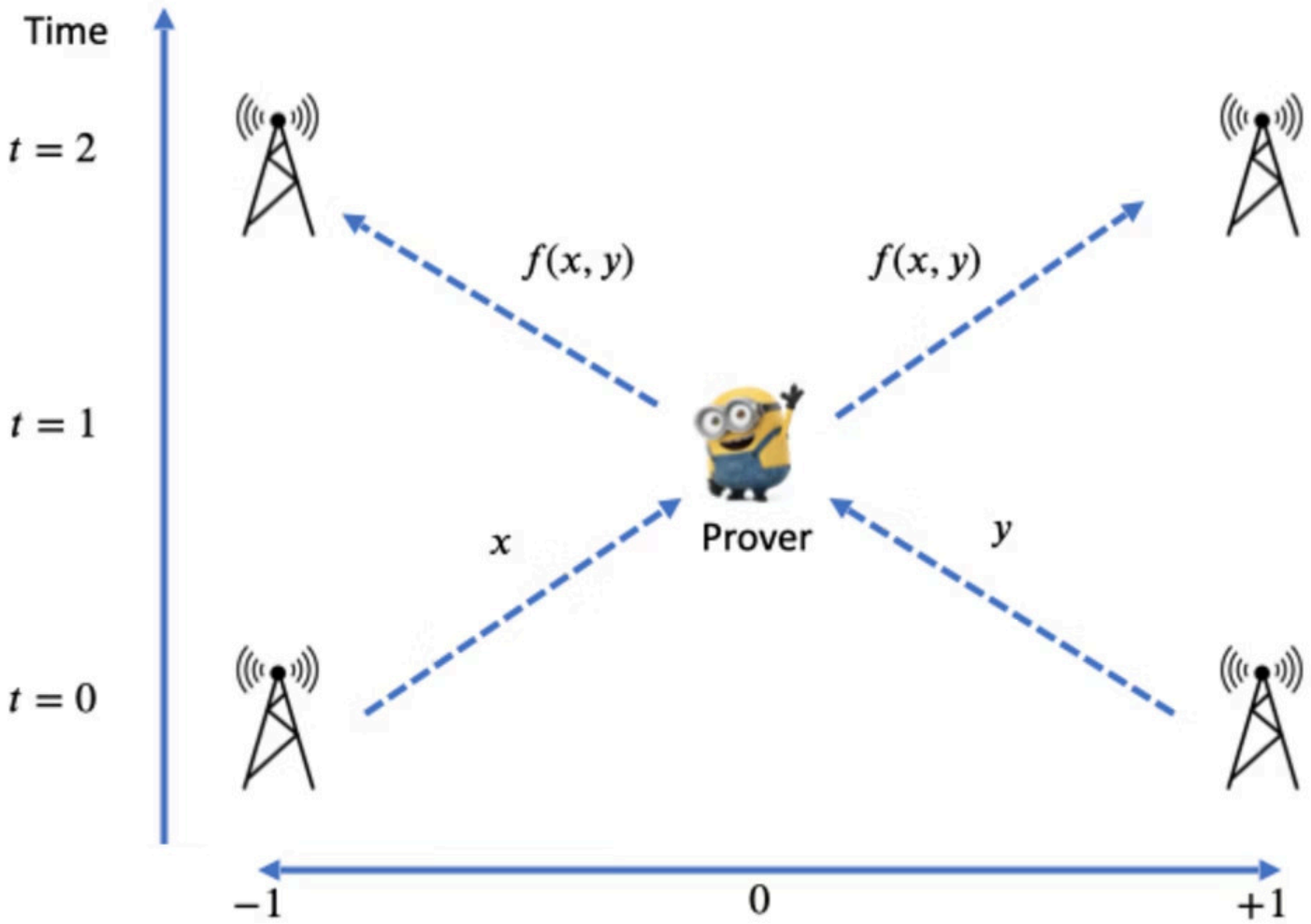
All parties are equipped with:

1. Light speed communication
2. Instantaneous computation
3. Synchronized and perfectly real-time clocks
4. Perfect aiming of "laser beam" signals

Example: 1D position verification



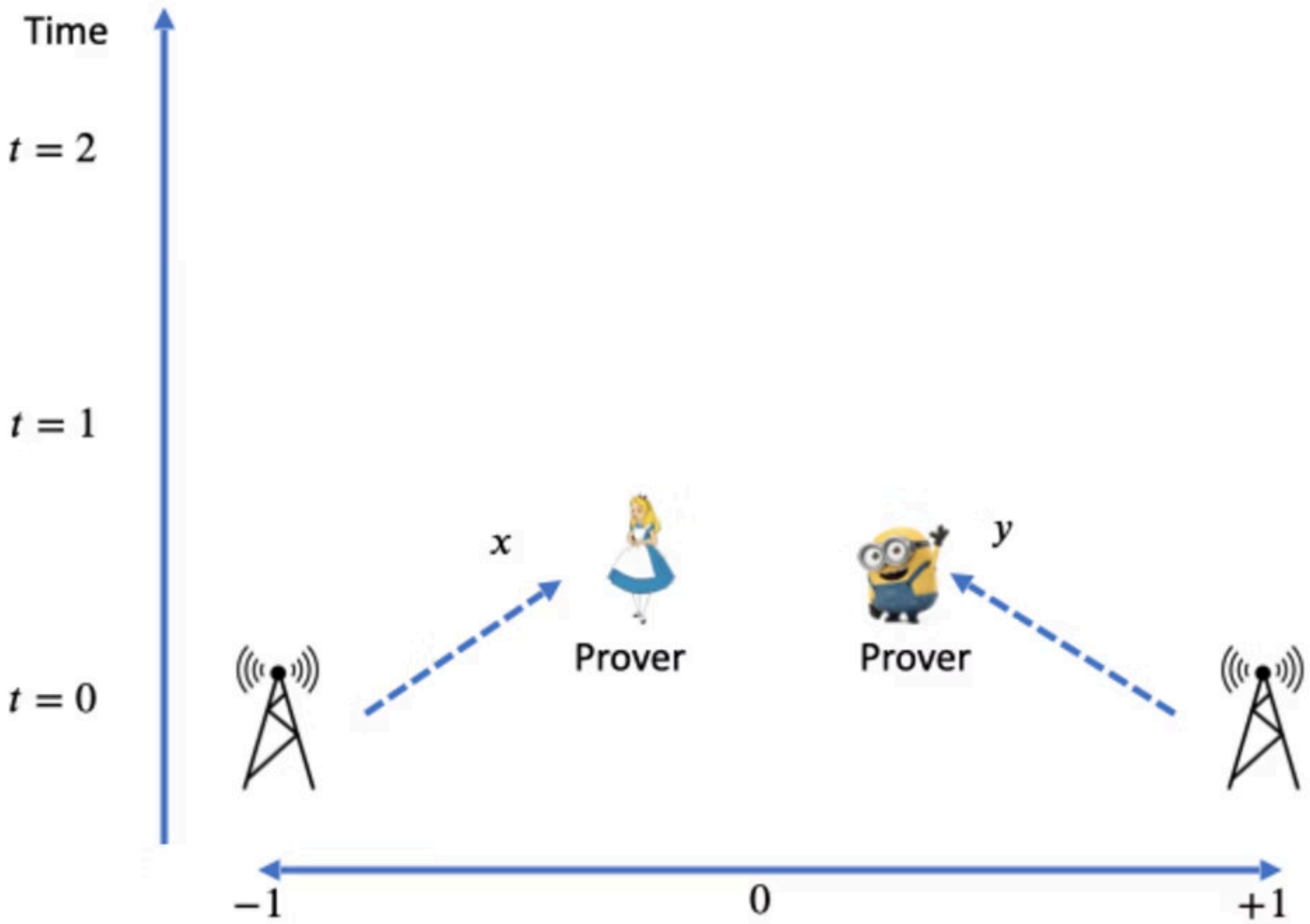
Example: 1D position verification



Verifiers check that prover's responses combine both challenges, and satisfy timing constraint.

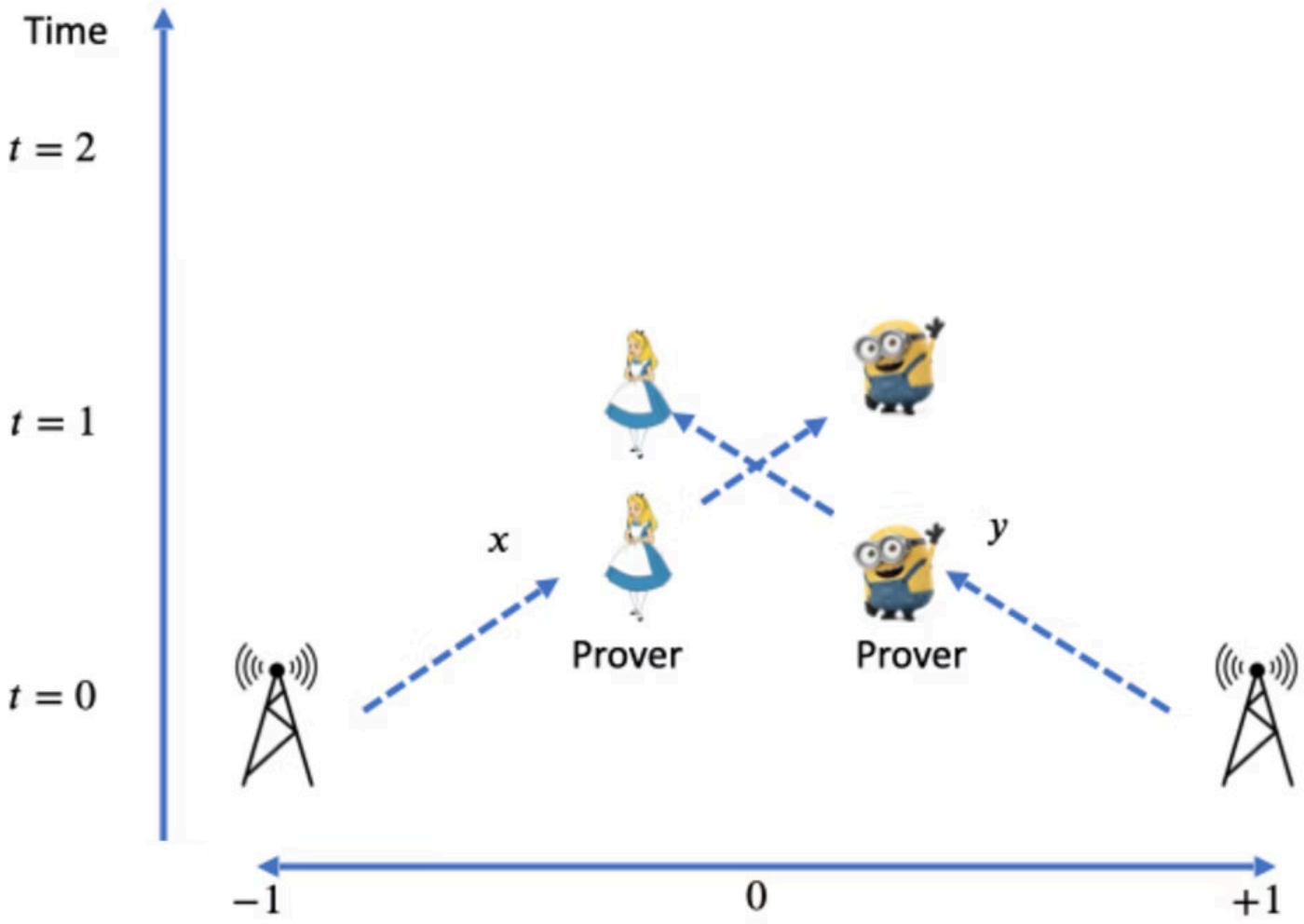
Idea: the only way for prover to satisfy challenges is to be in the correct position.

Classical impossibility



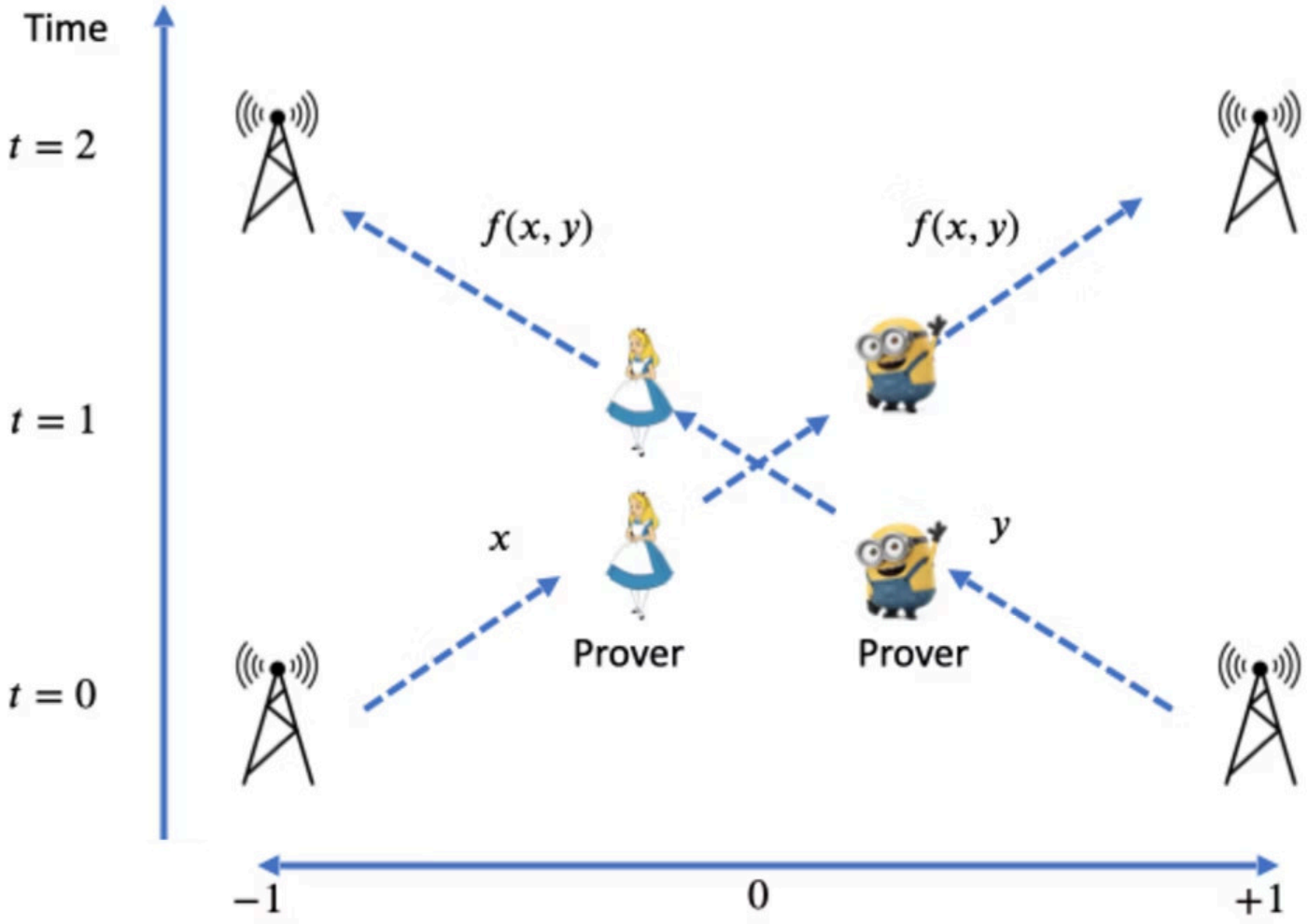
Theorem: every classical position verification protocol can be spoofed by **colluding provers**. This is because challenges can always be **copied** and **forwarded**.

Classical impossibility



Theorem: every classical position verification protocol can be spoofed by **colluding provers**. This is because challenges can always be **copied** and **forwarded**.

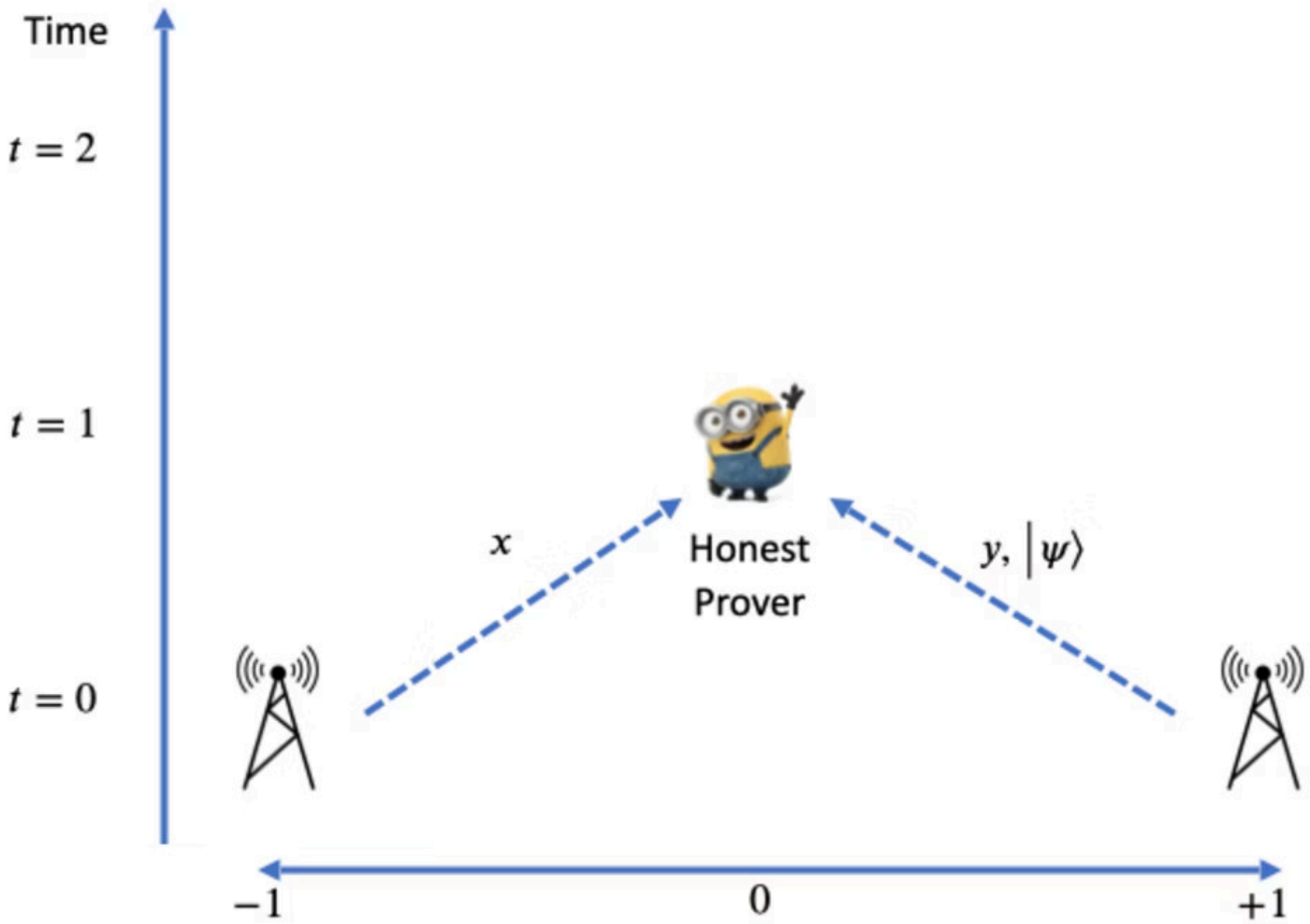
Classical impossibility



Theorem: every classical position verification protocol can be spoofed by **colluding provers.** This is because challenges can always be **copied** and **forwarded**.

Quantum position verification

Idea: use quantum resources, and leverage unclonability.

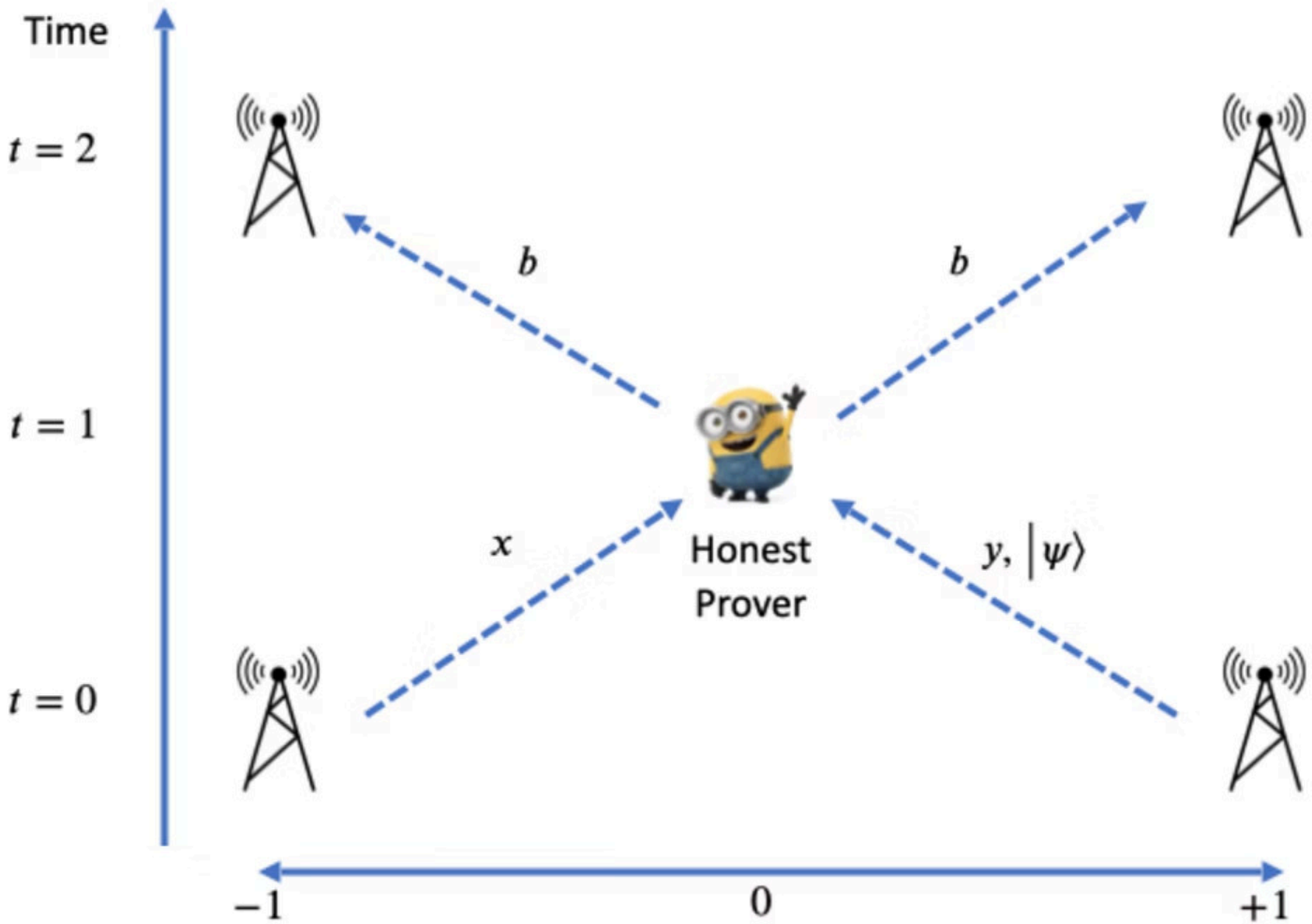


f-BB84:

1. Verifiers secretly sample $x, y \in \{0,1\}^n$ and $b \in \{0,1\}$
2. Left verifier sends x
3. Right verifier sends y and BB84 state $|\psi\rangle = H^{f(x,y)}|b\rangle$

Quantum position verification

Idea: use quantum resources, and leverage unclonability.

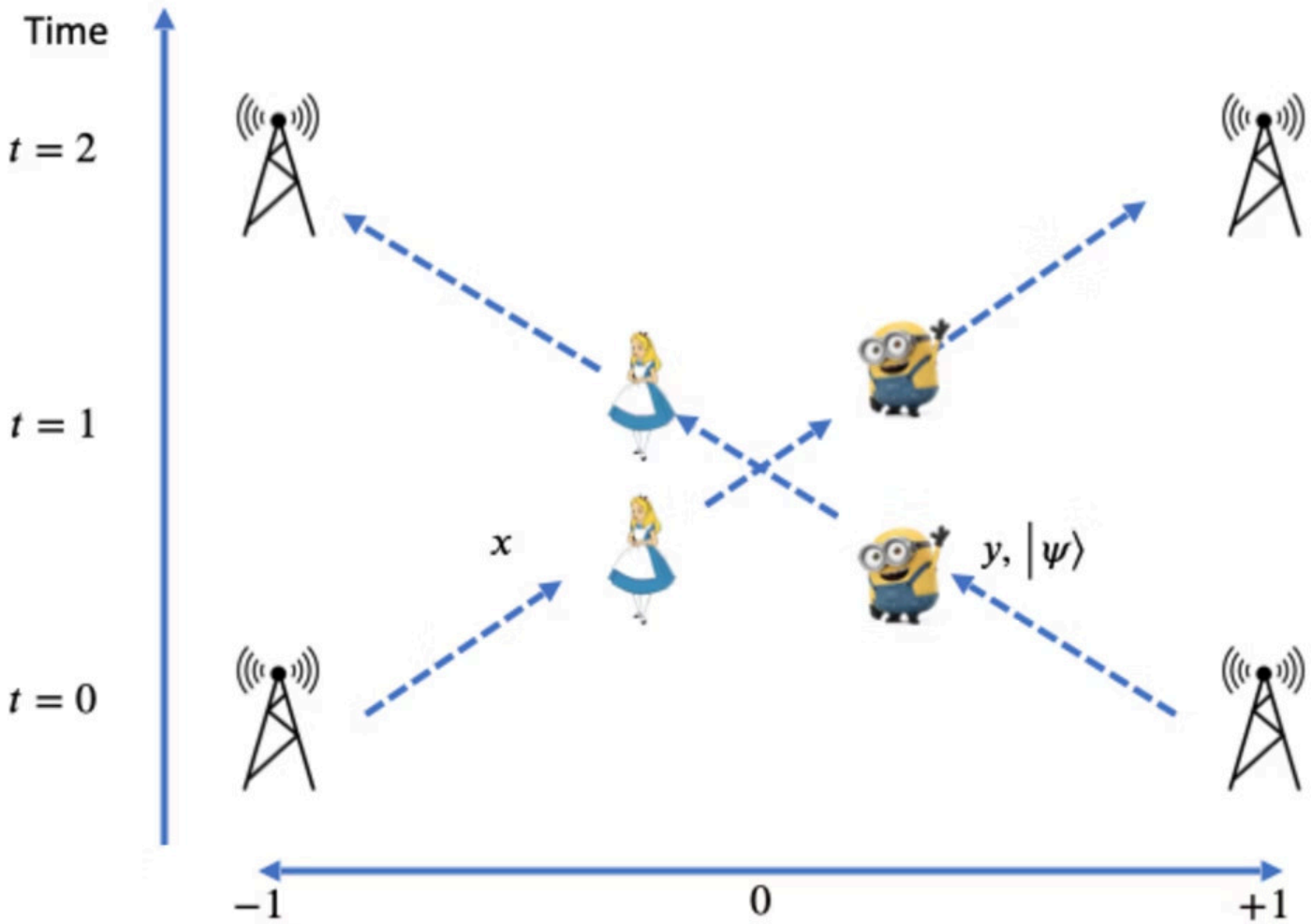


f-BB84:

1. Verifiers secretly sample $x, y \in \{0,1\}^n$ and $b \in \{0,1\}$
2. Left verifier sends x
3. Right verifier sends y and BB84 state $|\psi\rangle = H^{f(x,y)}|b\rangle$
4. Honest prover recovers $|b\rangle$ and sends back to verifiers.

Quantum position verification

Idea: use quantum resources, and leverage unclonability.



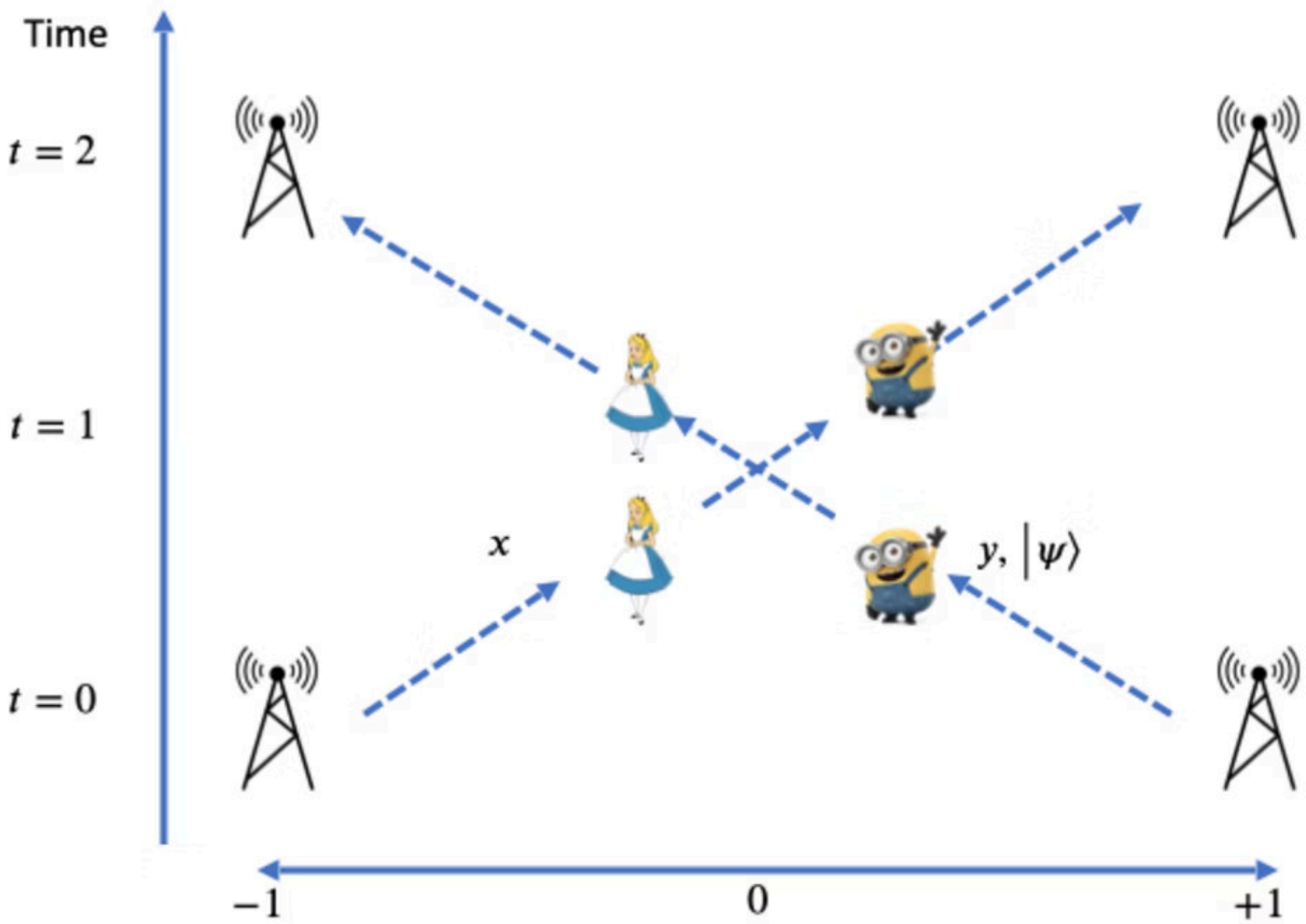
f-BB84:

1. Verifiers secretly sample $x, y \in \{0,1\}^n$ and $b \in \{0,1\}$
2. Left verifier sends x
3. Right verifier sends y and BB84 state $|\psi\rangle = H^{f(x,y)}|b\rangle$
4. Honest prover recovers $|b\rangle$ and sends back to verifiers.

Bob has to decide what to do with $|\psi\rangle$.

If he sends it to Alice, then he can't recover $|b\rangle$. If he keeps it, Alice can't recover $|b\rangle$.

Quantum position verification



Theorem: f-BB84 can be spoofed if Alice and Bob share $\Theta(n)$ EPR pairs for $f = IP$.

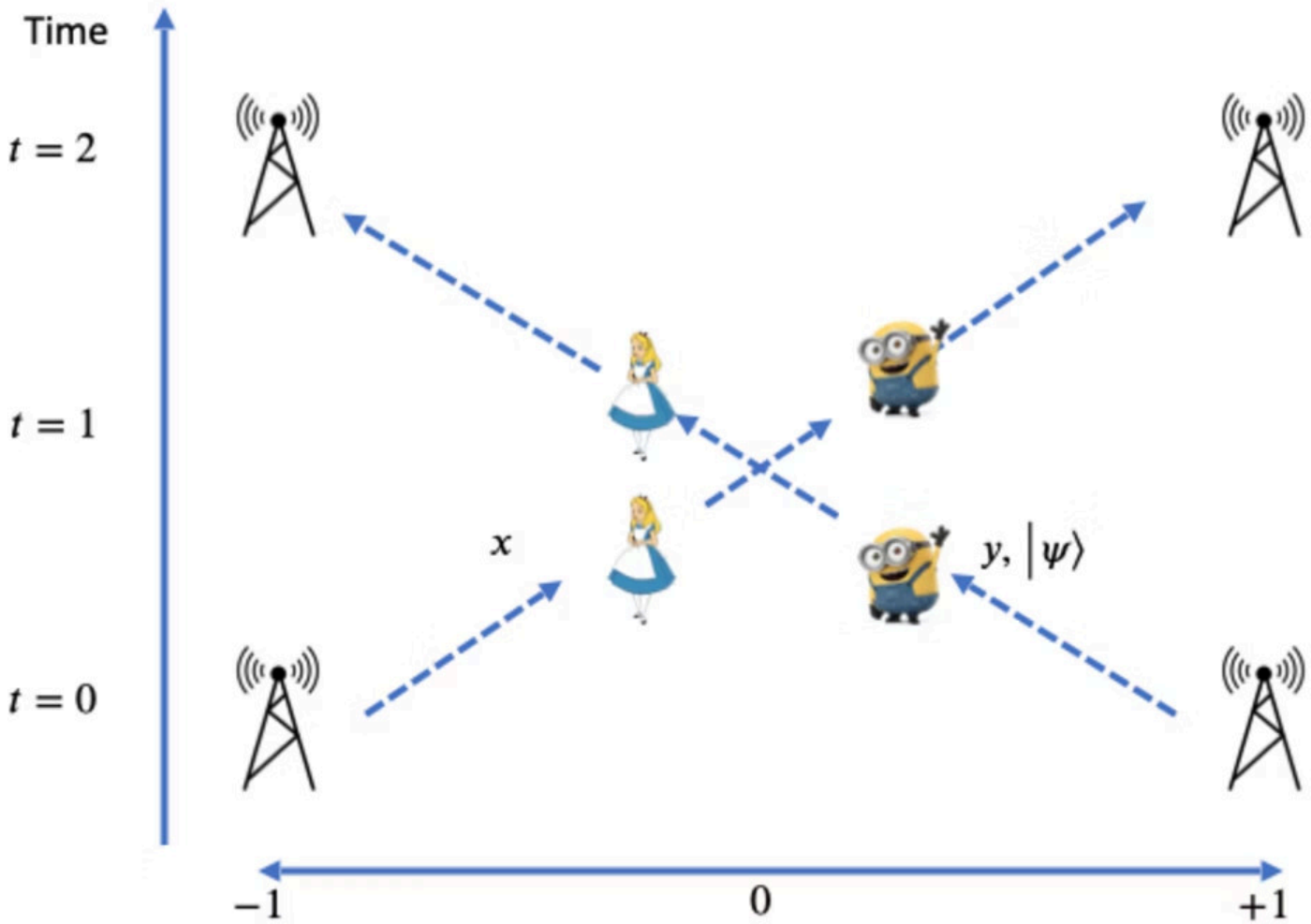
(Kent-Munro-Spiller '11, Lau-Lo '11)

Theorem: f-BB84 is secure against spoofers with $o(\log n)$ qubits of entanglement, for $f = IP$

(Bluhm, Christandl, Speelman, '21)

Security: If no spoofers are in correct position, then verifiers accept with low probability.

Quantum position verification

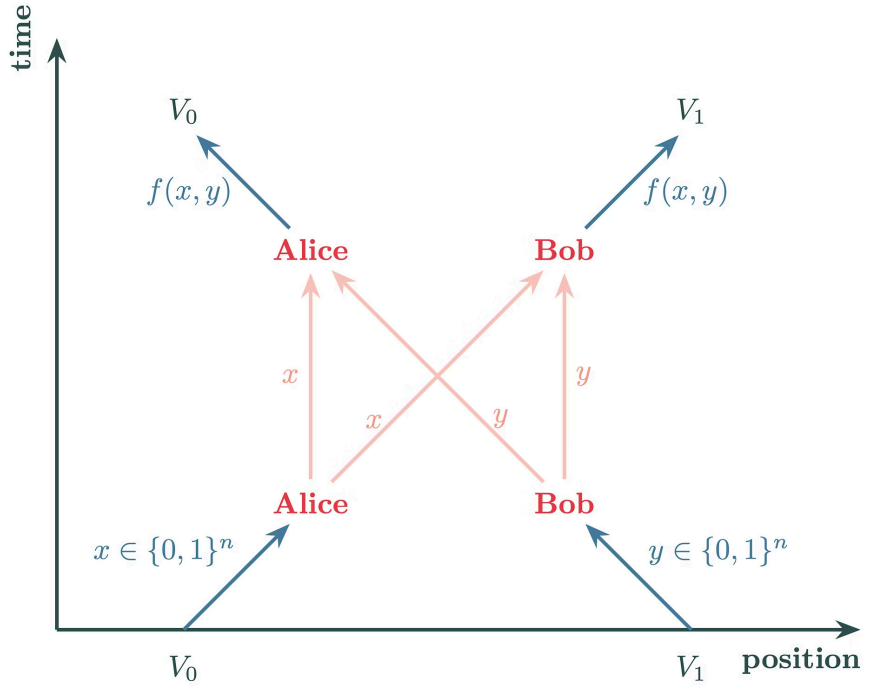


Theorem: **Every** position verification protocol can be spoofed if Alice and Bob share $\exp(n)$ EPR pairs.

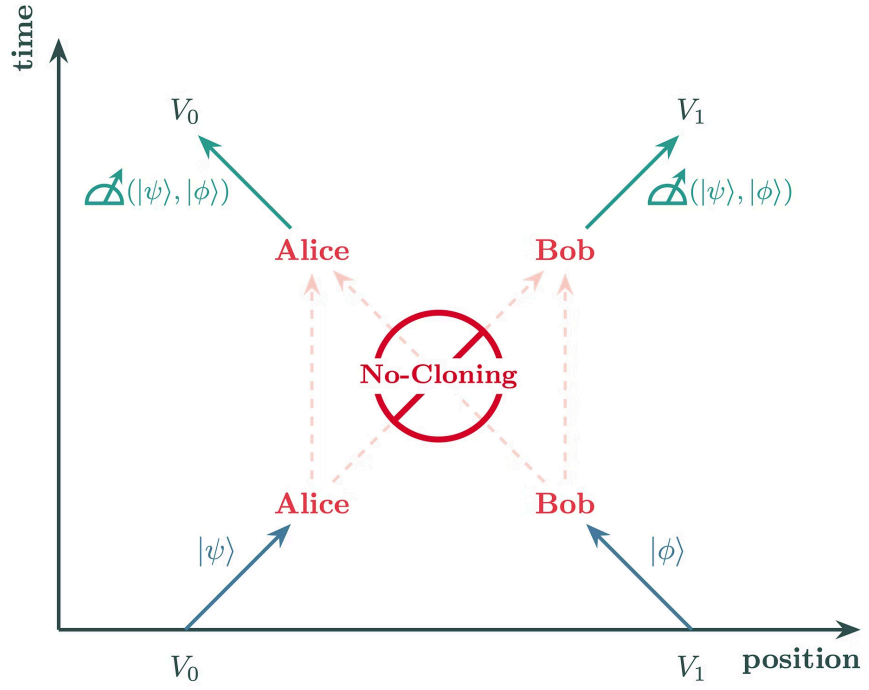
(Buhrman, et al. '11, Beigi-König '11)

Open question: Is there a position verification protocol secure against spoofers with superpoly qubits of entanglement?

Recap: Classical vs. Quantum PV

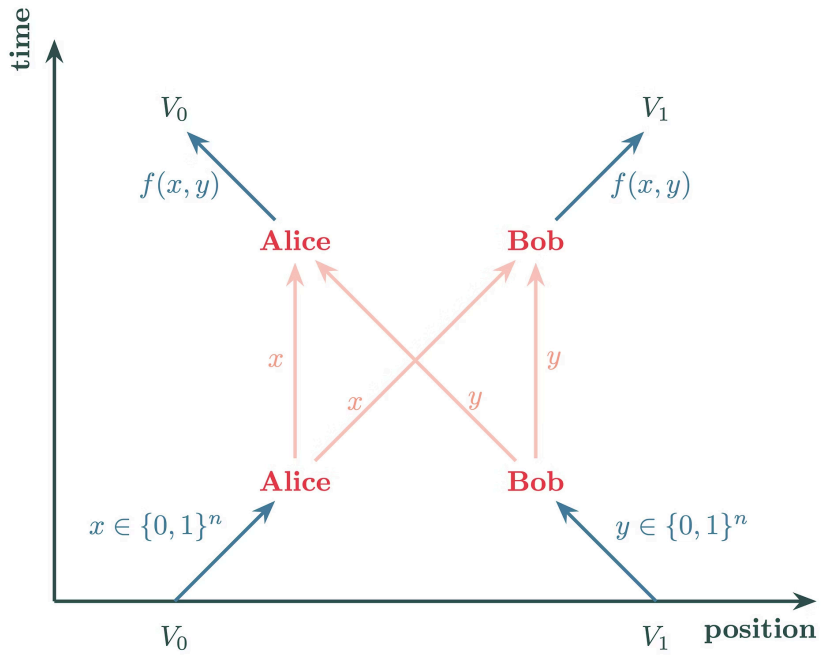


Any information send by V0 and V1 can be fully replicated by both spoofers, and the response computed in parallel.

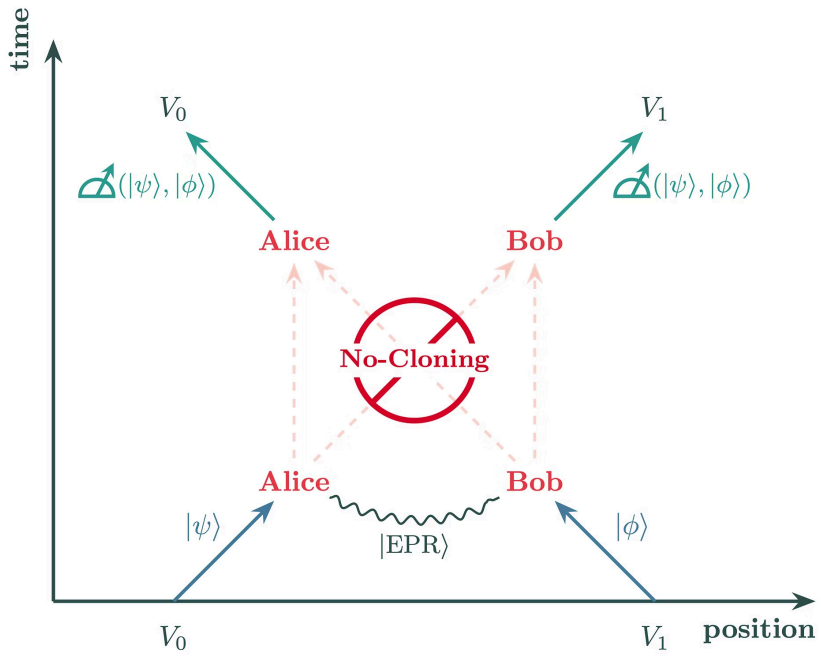


But with quantum challenges, there is no way for Alice & Bob to create two full copies of the challenge information.

Recap: Classical vs. Quantum PV



Any information send by V0 and V1 can be fully replicated by both spoofers, and the response computed in parallel.



That is, not unless the attackers pre-share entanglement!

Not Just a Theoretical Pipe Dream!

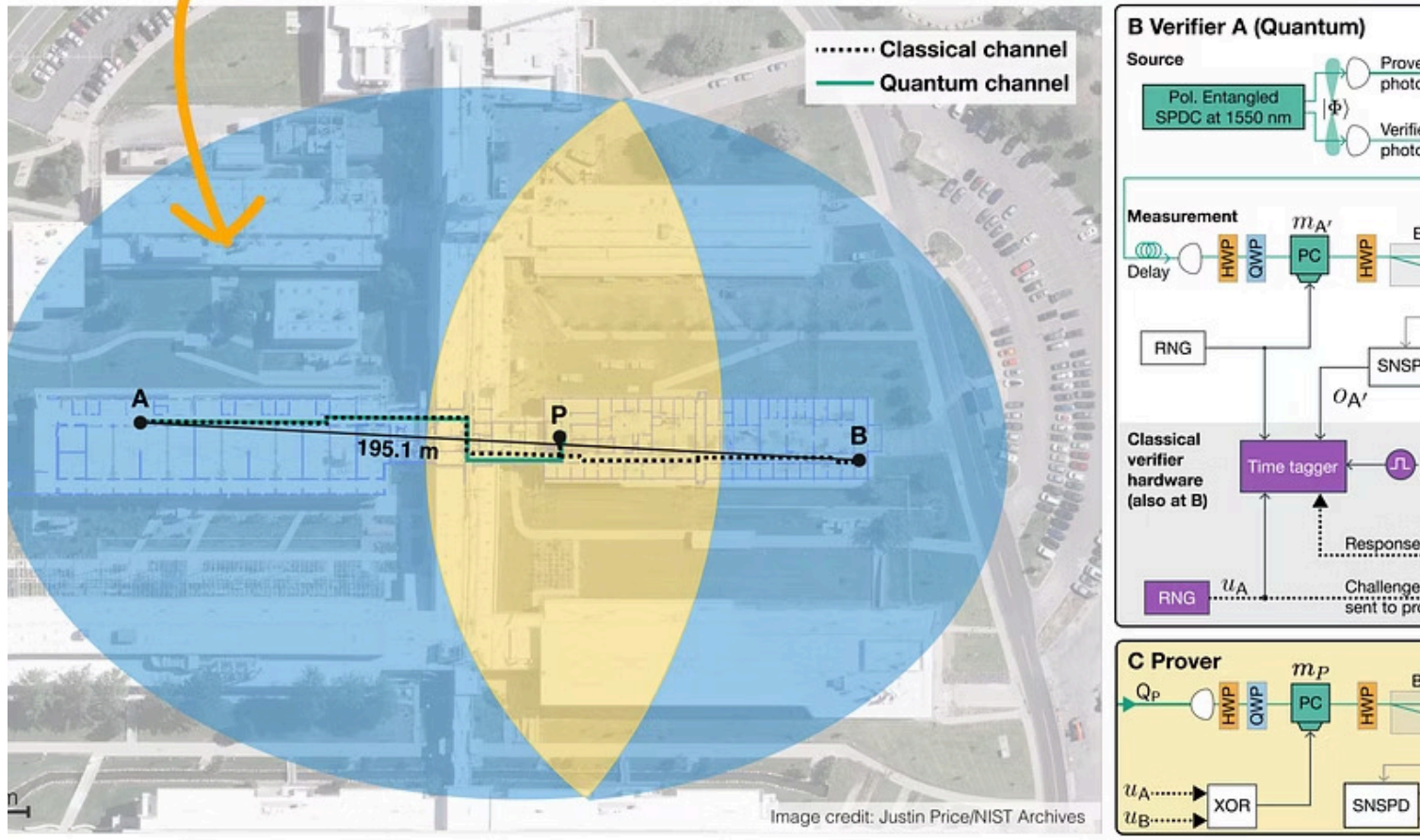
- Many experimentalists, including from the US, Europe, and China, are racing to implement secure and practical QPV.

Quantum Position Verification with Remote Untrusted Devices

Gautam A. Kavuri,[1,2,*] Yanbao Zhang,[3,†] Abigail R. Gookin,[4,2] Soumyadip Patra,[5] Joshua C. Bienfang,[6] Honghao Fu,[7] Yusuf Alnawakhtha,[8] Dileep V. Reddy,[1,2] Michael D. Mazurek,[1,2] Carlos Abellán,[9] Waldimar Amaya,[9] Morgan W. Mitchell,[10,11] Sae Woo Nam,[1,12] Carl A. Miller,[13,8] Richard P. Mirin,[12] Martin J. Stevens,[12] Scott Glancy,[14,1] Emanuel Knill,[14,1,15] and Lynden K. Shalm[1,12,16,‡]

[1]*Department of Physics, University of Colorado, Boulder, CO, 80309, USA*
[2]*Associate of the National Institute of Standards and Technology, Boulder, CO, 80305, USA*
[3]*Quantum Information Science Section, Computational Sciences and Engineering Division, Oak Ridge National Laboratory, Oak Ridge, TN, 37831, USA*

NIST (Maryland, USA)



3. Figure depicting the experimental setup and verifier and prover stations used fo
ıstration. Subfigure A depicts the quantum target region E of our protocol in yellow, an
:al target region of a comparable classical protocol in blue overlaid on an aerial photog
NIST building, where the experiment is housed. Subfigure B shows a schematic o

This Work: Verifying More General Position Properties

This Work: Verifying More General Position Properties, Privately

Privacy: A New, Far-Reaching Goal

Location Privacy is Crucial

Our position can be a **useful cryptographic credential**, with many exciting applications.

However, we may be **unwilling** to give out our **exact locations**. Leaked to the wrong hands, it can mean **life or death!**

Example: Private Alibis

How would somebody prove to a judge, "I wasn't at the crime scene at any point in the last month", without having to reveal the **entire month's location history**?

A Host of Other Applications

- Nuclear treaty enforcement
- Privacy-preserving georestrictions
- Strava posts which don't leak location

Privacy: A New, Far-Reaching Goal

Location Privacy is Crucial

Our position can be a **useful cryptographic credential**, with many exciting applications.

However, we may be **unwilling** to give out our **exact locations**. Leaked to the wrong hands, it can mean **life or death!**

Example: Private Alibis

How would somebody prove to a judge, "I wasn't at the crime scene at any point in the last month", without having to reveal the **entire month's location history**?

A Host of Other Applications

- Nuclear treaty enforcement
- Privacy-preserving georestrictions
- Strava posts which don't leak location

Privacy: A New, Far-Reaching Goal

Location Privacy is Crucial

Our position can be a **useful cryptographic credential**, with many exciting applications.

However, we may be **unwilling** to give out our **exact locations**. Leaked to the wrong hands, it can mean **life or death!**

Example: Private Alibis

How would somebody prove to a judge, "I wasn't at the crime scene at any point in the last month", without having to reveal the **entire month's location history**?

A Host of Other Applications

- Nuclear treaty enforcement
- Privacy-preserving georestrictions
- Strava posts which don't leak location

Strava Security— Seriously!

- Want to prove properties of our spacetime **trajectory**, such as how far we traveled.
- Easy to do with continuous location tracking, but this completely sacrifices privacy!

FRANCE • STRAVALEAKS

How Emmanuel Macron can be tracked: Watch the first episode of StravaLeaks

By Antoine Schirer, Sébastien Bourdon, Le Monde's video investigation team and Sinead McCausland

Published on October 27, 2024, at 6:21 pm (Paris), updated on November 6, 2024, at 5:24 pm

Zero-Knowledge Position Verification

Take any spacetime region R. A ZK-PV protocol for R has three conditions:

1. **Completeness**
   A prover located anywhere in R will almost always pass the protocol.

2. **Position Security (Soundness)**
   A coalition of provers outside of R, who pre-share at most a bounded amount of entanglement, will almost always fail.

3. **(Honest-Verifier) Zero Knowledge**
   Honest-but-curious verifiers, who are polynomially bounded, can learn nothing about the prover's true position.

Zero-Knowledge Position Verification

Take any spacetime region R. A ZK-PV protocol for R has three conditions:

1. **Completeness**
   A prover located anywhere in R will almost always pass the protocol.

2. **Position Security (Soundness)**
   A coalition of provers outside of R, who pre-share at most a bounded amount of entanglement, will almost always fail.

3. **(Honest-Verifier) Zero Knowledge**
   Honest-but-curious verifiers, who are polynomially bounded, can learn nothing about the prover's true position.

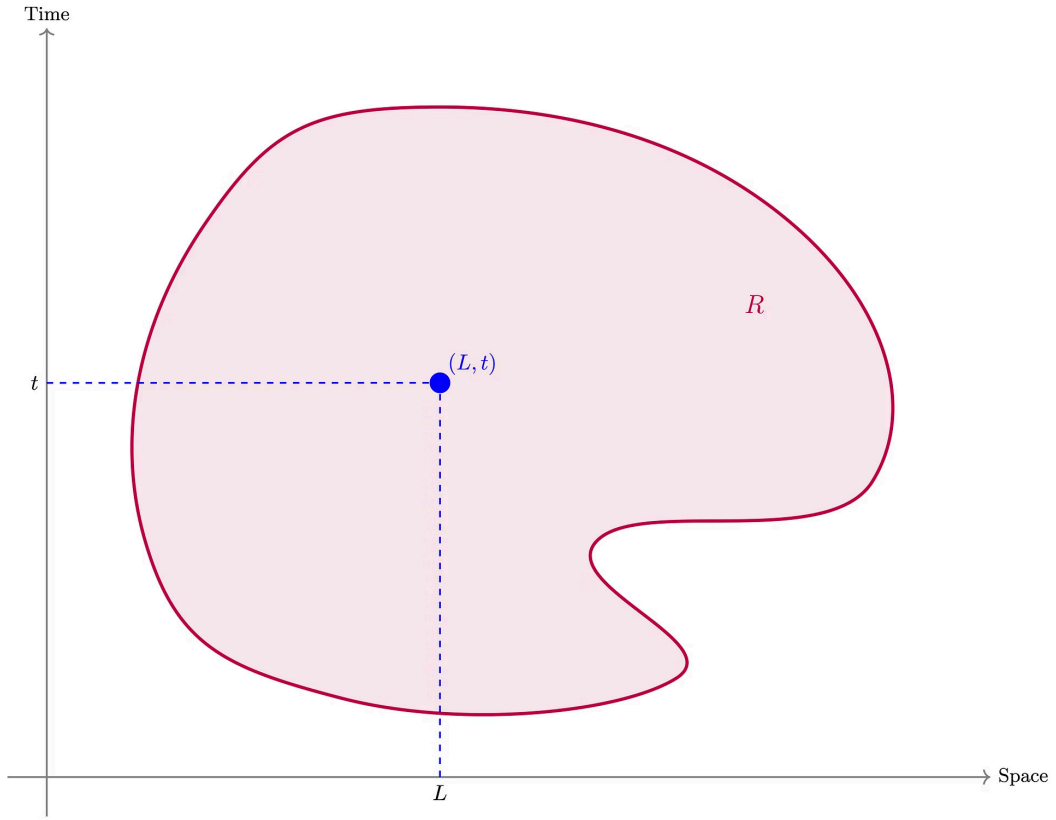
Main Theorem (Constructing ZK-PV):

Assuming secure position verification, and post-quantum private-key cryptography, there is a Zero-Knowledge Position Verification protocol for any finite region R.

Generalizing PV to Spacetime Regions

Standard PV: the prover must be at a specific point at a specific time.

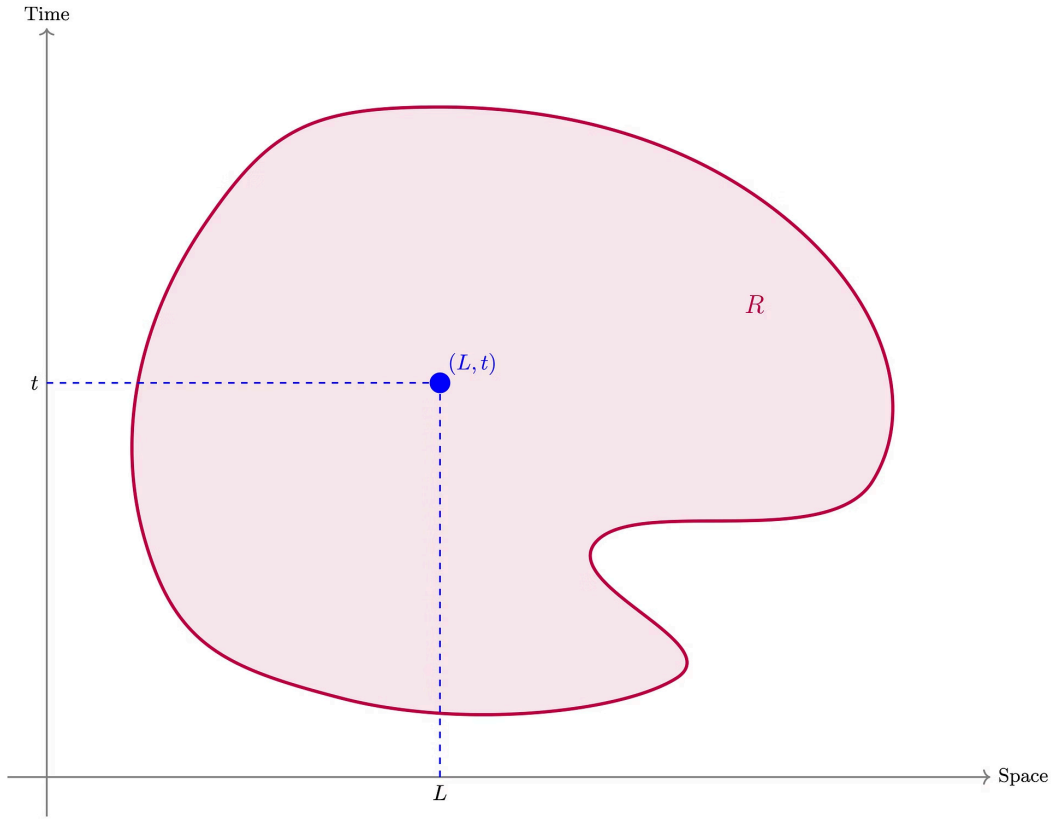
Our generalization: The prover's position must only belong to some arbitrary region of spacetime, R.



Generalizing PV to Spacetime Regions

Standard PV: the prover must be at a specific point at a specific time.

Our generalization: The prover's position must only belong to some arbitrary region of spacetime, R.



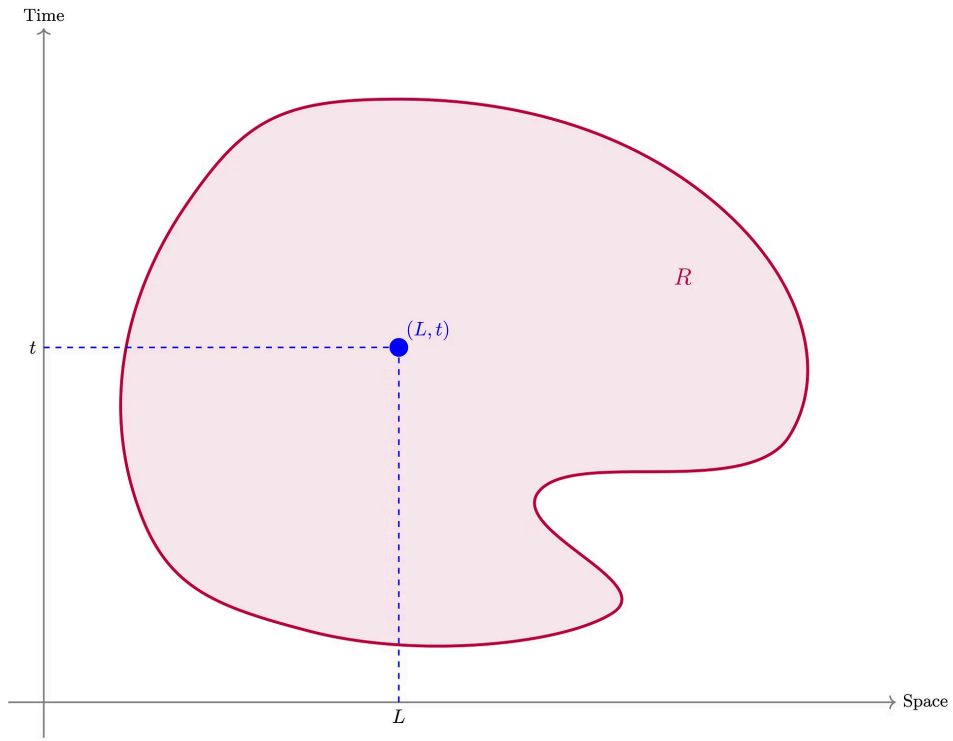
Generalizing PV to Spacetime Regions

Standard PV: the prover must be at a specific point at a specific time.

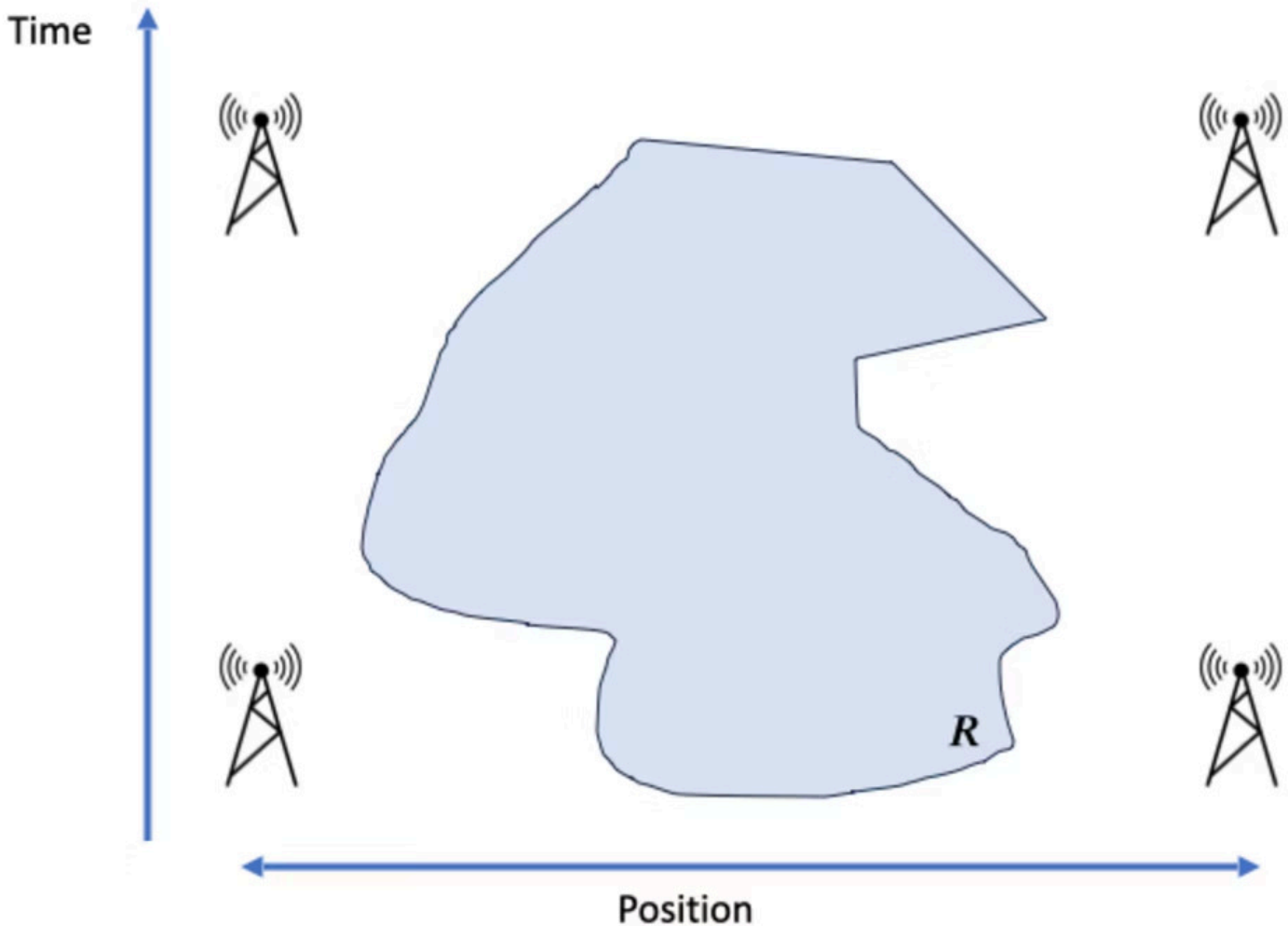
Our generalization: The prover's position must only belong to some arbitrary region of spacetime, R.

Note: If we set $|R| = 1$, then the definition of Zero-Knowledge PV collapses to that of standard PV:

Zero-Knowledge condition is vacuous.

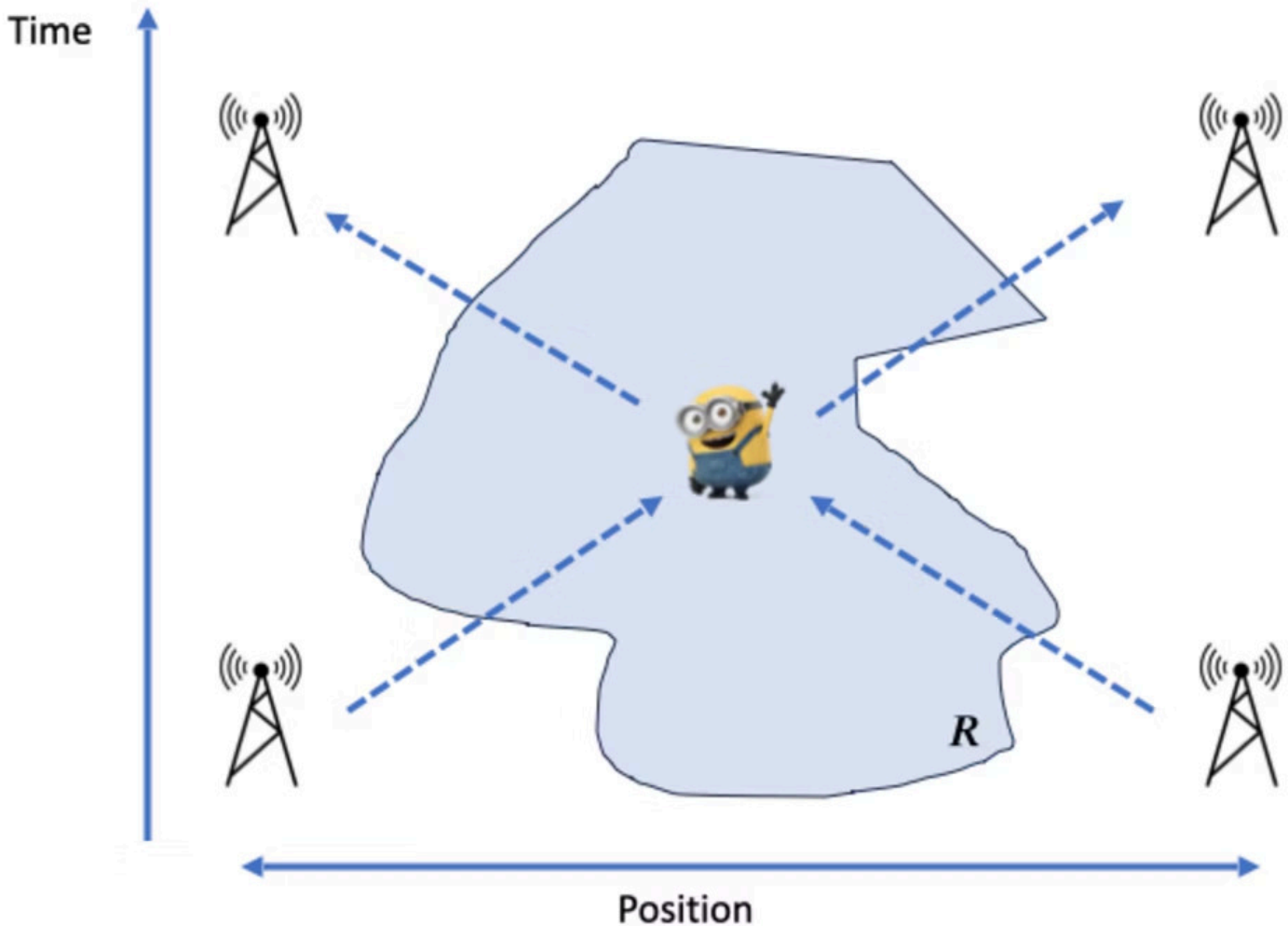


Zero-knowledge position verification



Definition: A zero-knowledge position verification protocol for a spacetime region $R \subseteq \mathbb{R} \times \mathbb{Z}$ satisfies

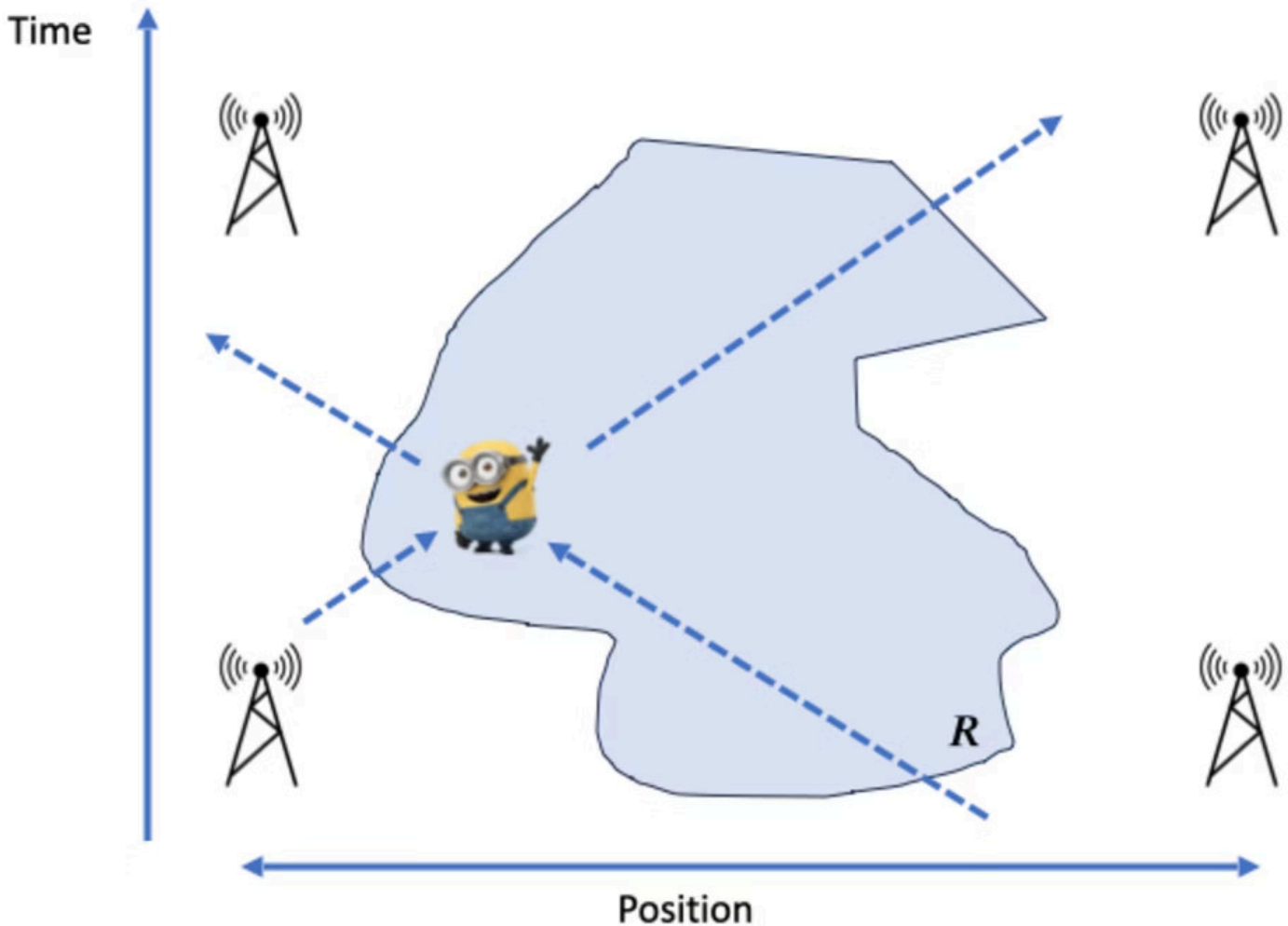
Zero-knowledge position verification



Definition: A zero-knowledge position verification protocol for a spacetime region $R \subseteq \mathbb{R} \times \mathbb{Z}$ satisfies

Completeness: For all $(L, t) \in R$ there exists an honest prover P at (L, t) that convinces verifiers to accept whp.

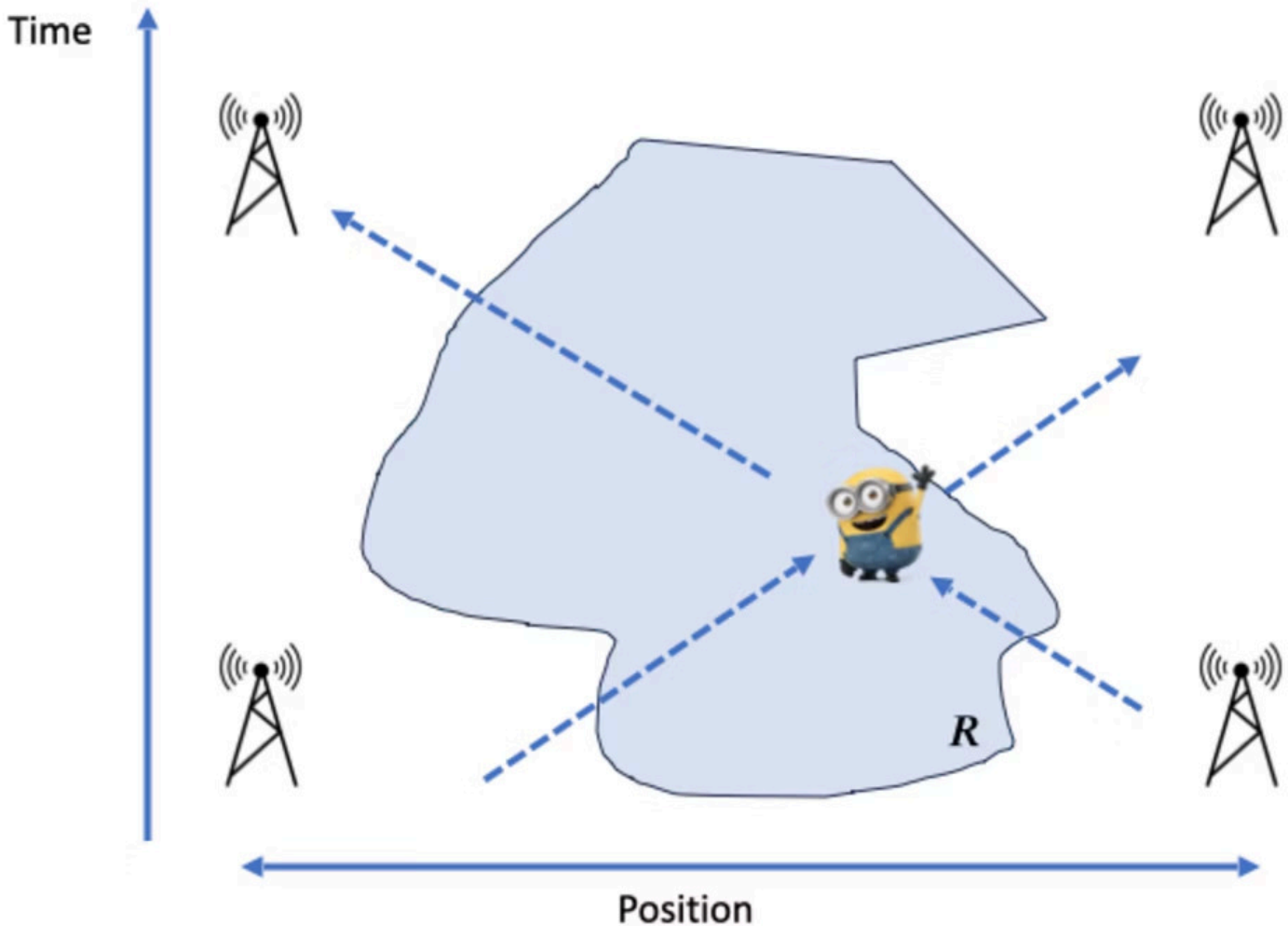
Zero-knowledge position verification



Definition: A zero-knowledge position verification protocol for a spacetime region $R \subseteq \mathbb{R} \times \mathbb{Z}$ satisfies

Completeness: For all $(L, t) \in R$ there exists an honest prover P at (L, t) that convinces verifiers to accept whp.

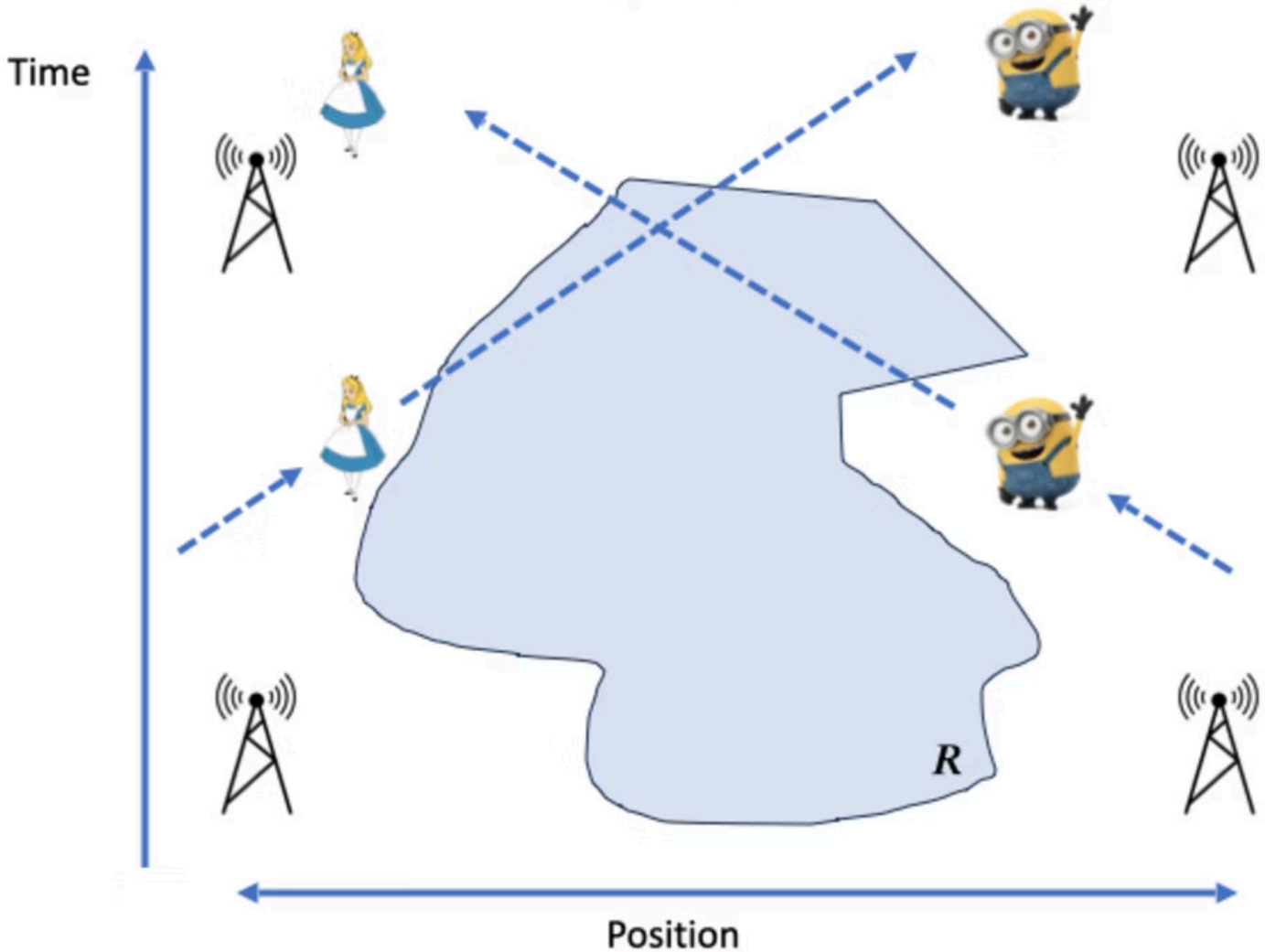
Zero-knowledge position verification



Definition: A zero-knowledge position verification protocol for a spacetime region $R \subseteq \mathbb{R} \times \mathbb{Z}$ satisfies

Completeness: For all $(L, t) \in R$ there exists an honest prover P at (L, t) that convinces verifiers to accept whp.

Zero-knowledge position verification

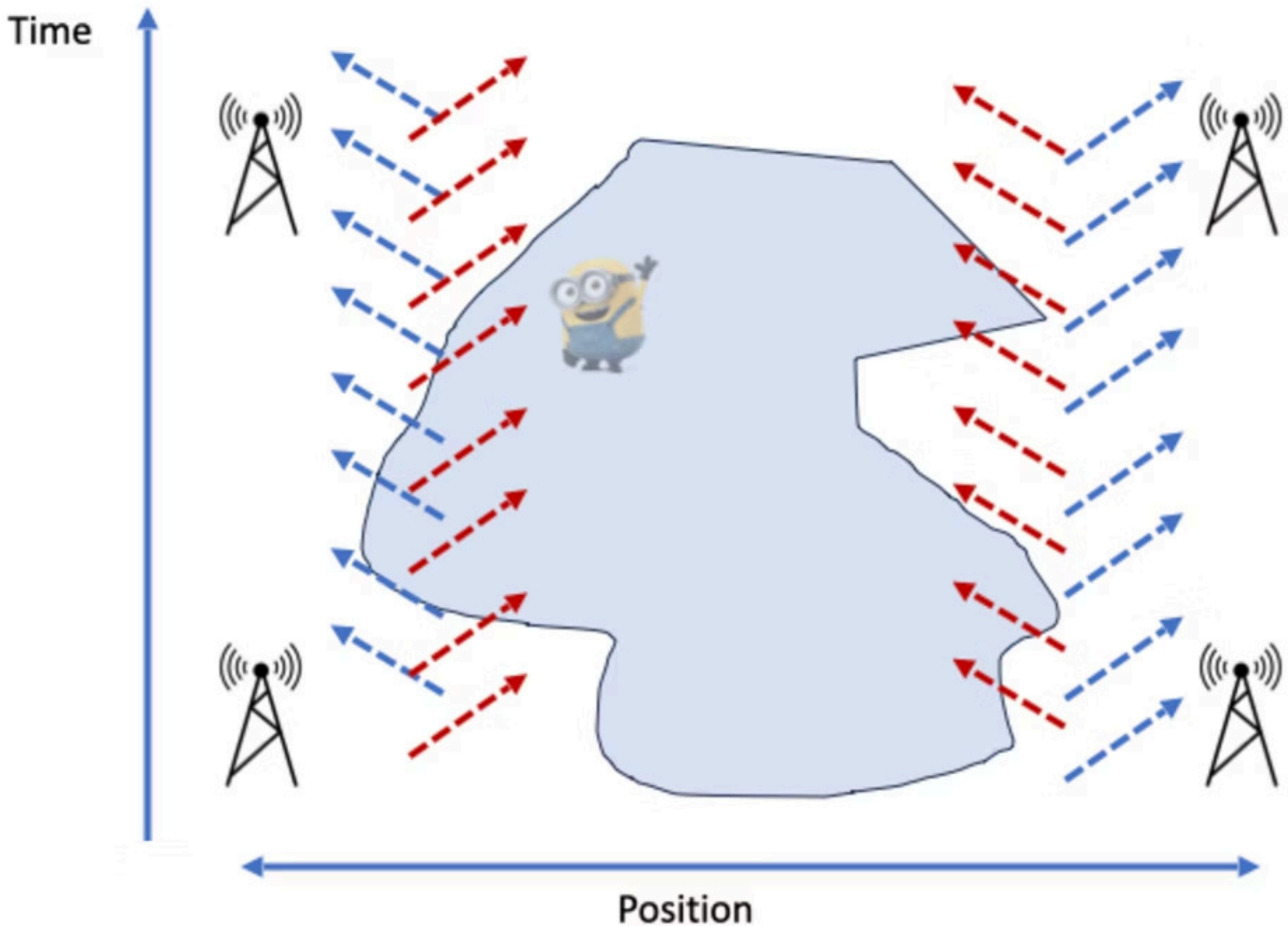


Definition: A zero-knowledge position verification protocol for a spacetime region $R \subseteq \mathbb{R} \times \mathbb{Z}$ satisfies

Position security: For all spoofing provers $P = (P_1, \ldots, P_k)$ from a class $\mathscr{C}$ of adversaries, if none of them are located at R then verifiers reject whp.

E.g., class $\mathscr{C}$ of adversaries can be spoofers with bounded entanglement and computational power.

Zero-knowledge position verification

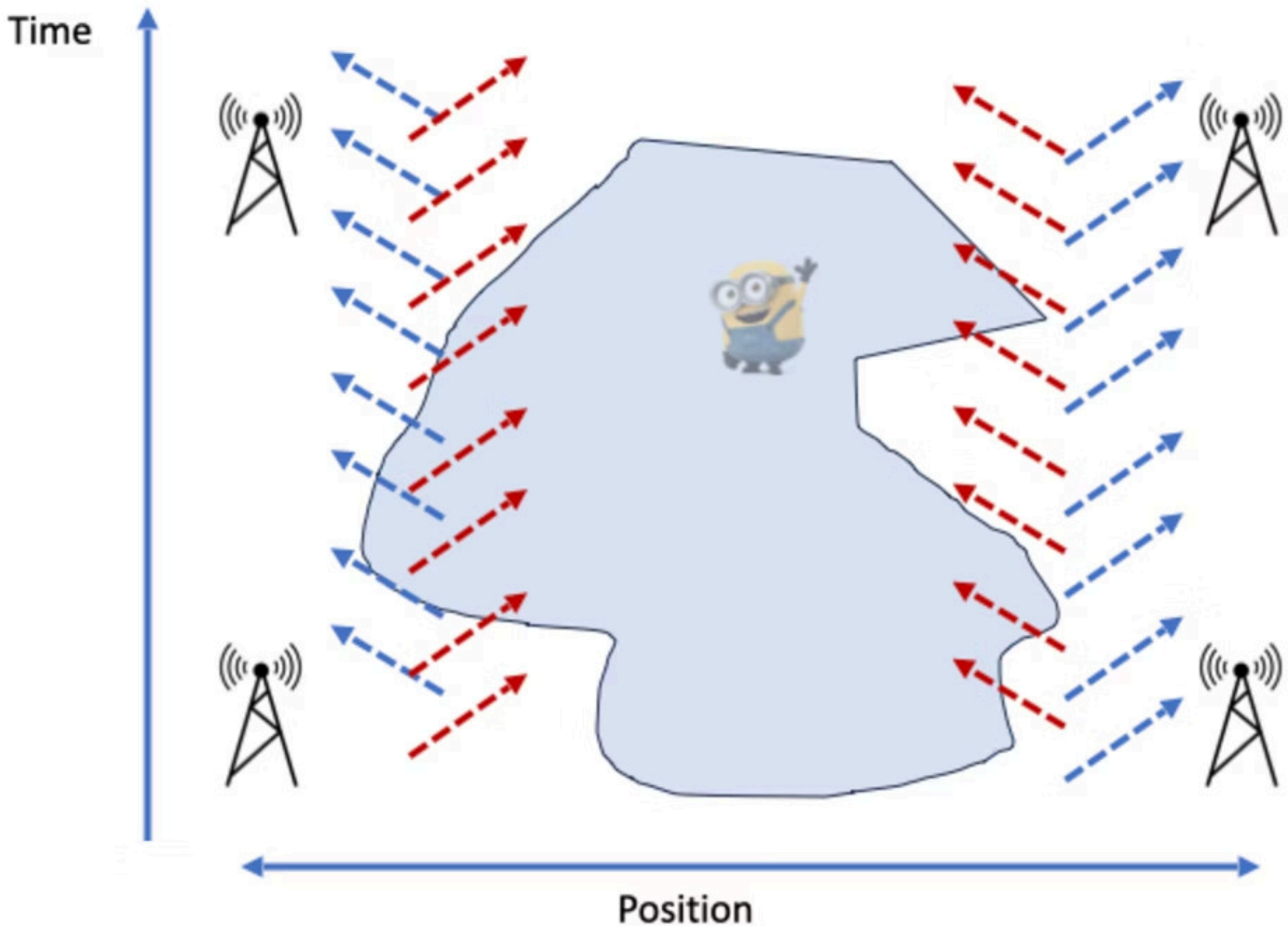


Definition: A zero-knowledge position verification protocol for a spacetime region $R \subseteq \mathbb{R} \times \mathbb{Z}$ satisfies

Privacy: If the honest prover is located in region R, the view of the verifiers can be efficiently simulated at each step of the protocol.

Analogous to quantum ZK.

Zero-knowledge position verification

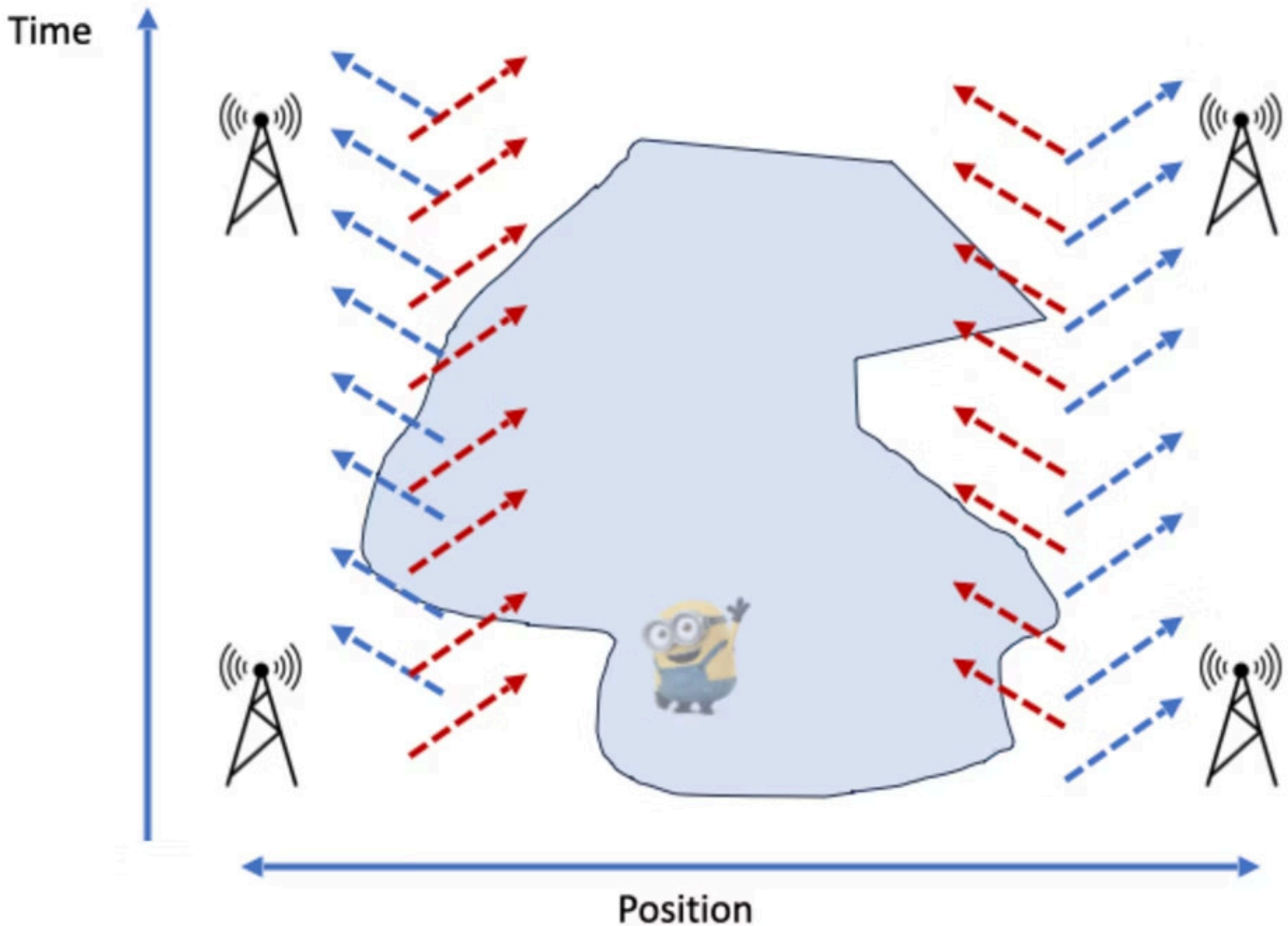


Definition: A zero-knowledge position verification protocol for a spacetime region $R \subseteq \mathbb{R} \times \mathbb{Z}$ satisfies

Privacy: If the honest prover is located in region R, the view of the verifiers can be efficiently simulated at each step of the protocol.

Analogous to quantum ZK.

Zero-knowledge position verification

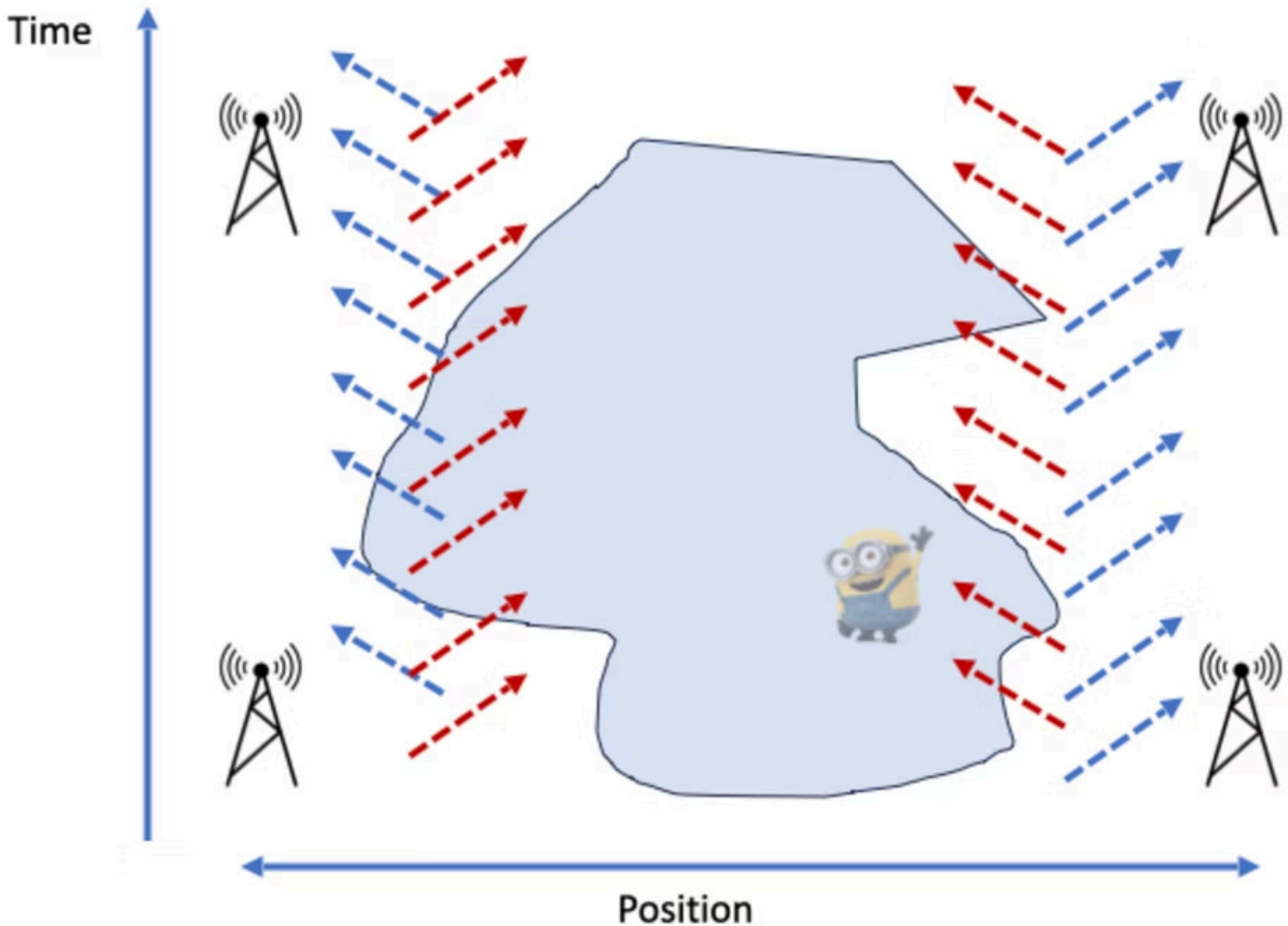


Definition: A zero-knowledge position verification protocol for a spacetime region $R \subseteq \mathbb{R} \times \mathbb{Z}$ satisfies

Privacy: If the honest prover is located in region R, the view of the verifiers can be efficiently simulated at each step of the protocol.

Analogous to quantum ZK.

Zero-knowledge position verification



Definition: A zero-knowledge position verification protocol for a spacetime region $R \subseteq \mathbb{R} \times \mathbb{Z}$ satisfies

Privacy: If the honest prover is located in region R, the view of the verifiers can be efficiently simulated at each step of the protocol.

Analogous to quantum ZK.

Constructing Zero-Knowledge Position Proofs

Detour: Classical Cryptography — Commitments

01

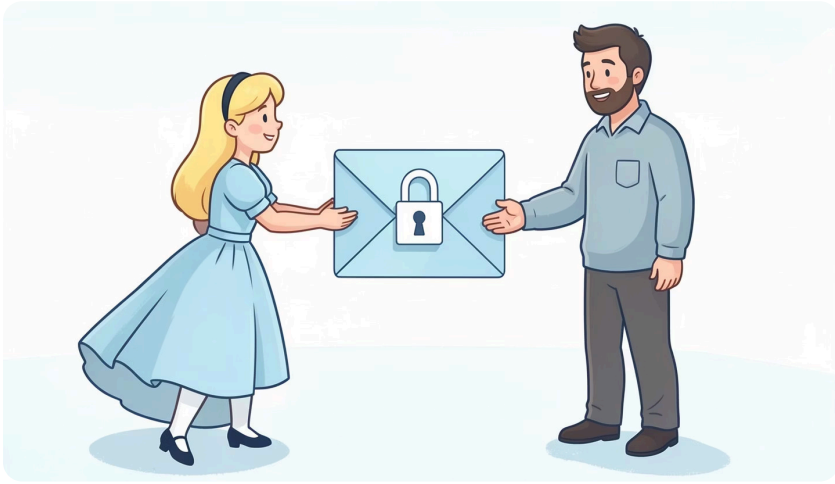
Commit

- Alice sends Bob a message "in a padlocked envelope".
- The message is completely hidden from Bob, but he knows that Alice cannot tamper with his copy.

02

Reveal

- Later on, Alice wants to reveal her message to Bob
- She sends Bob a secret string (the "padlock key") which opens the commitment to her message.



Security Properties

Hiding

- Bob learns nothing about the message until Alice chooses to reveal.

Binding

- Once Alice sends the commitment, there is **only one** value which it is possible for her to reveal.

Detour: Classical Cryptography — Commitments

01

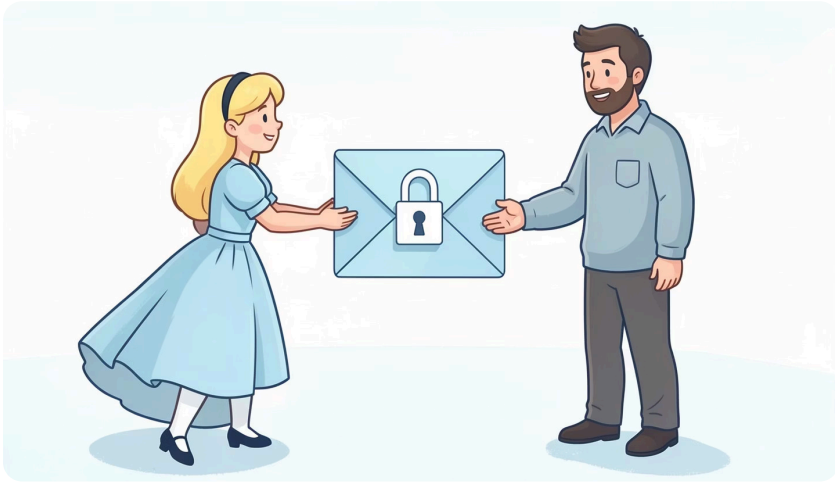
Commit

- Alice sends Bob a message "in a padlocked envelope".
- The message is completely hidden from Bob, but he knows that Alice cannot tamper with his copy.

02

Reveal

- Later on, Alice wants to reveal her message to Bob
- She sends Bob a secret string (the "padlock key") which opens the commitment to her message.



Security Properties

Hiding

- Bob learns nothing about the message until Alice chooses to reveal.

Binding

- Once Alice sends the commitment, there is **only one** value which it is possible for her to reveal.

Detour: Classical Cryptography — Zero Knowledge Proofs

Zero-Knowledge Proofs (ZKP)

- Alice wants to prove to Bob, "I possess a string x with property P".
- After being convinced that x satisfies P, Bob should still know absolutely nothing else about x.

Detour: Classical Cryptography — Zero Knowledge Proofs

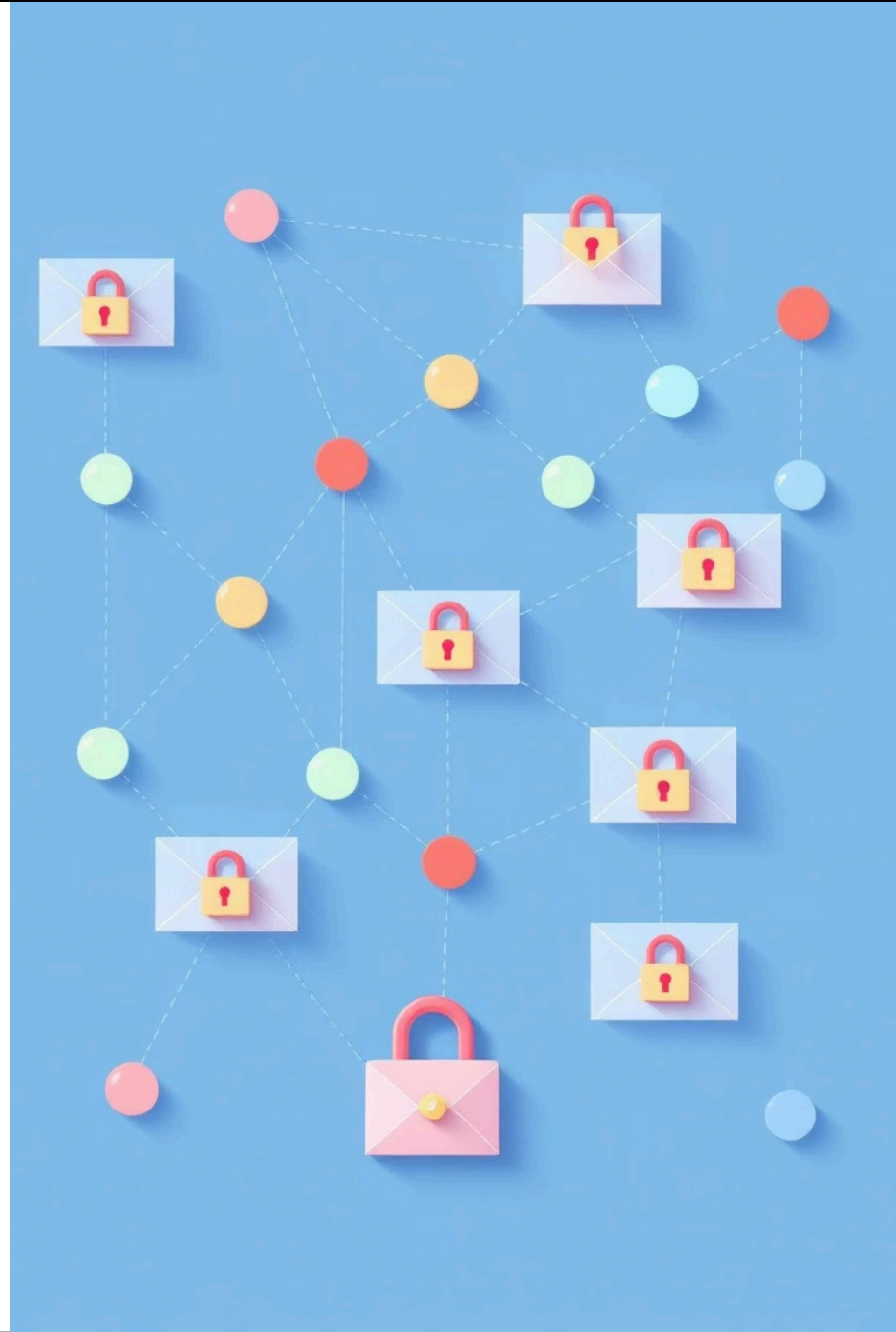
Zero-Knowledge Proofs (ZKP)

- Alice wants to prove to Bob, "I possess a string x with property P".
- After being convinced that x satisfies P, Bob should still know absolutely nothing else about x.

Assuming cryptographic commitments exist, there is a ZKP protocol for any NP predicate (GMW'87).

Example Zero-Knowledge Proof: 3-Coloring

1. **Setup**: Alice claims to have a valid 3-coloring of a graph.
2. **Commit**: Alice sends Bob a commitment to the color of each node.
3. **Challenge**:
    a. Bob randomly picks an edge, and tells Alice to reveal its coloring.
    b. Alice reveals the colors of its two endpoints by sending the corresponding reveal strings.
4. **Verify**: Bob checks the following:
    a. That the commitment opened to the correct colors, and
    b. that the two nodes were indeed colored differently.



New Tool: Position Commitments

01

Commit Phase

Prover at spacetime point (L, t) interacts with verifiers to create a position commitment C.

02

Prover Reveals

At any later time, the prover can announce their location (L, t), along with some reveal string, to the verifiers.

03

Verification

The verifiers decide whether to accept or reject the prover's claimed location.

Security Properties

(Honest-Verifier) Hiding Property

- (Honest-but-curious) verifiers learn no information about (L, t) during the Commit phase.
- No matter where the prover was in R, their view is exactly the same.

Position Binding

- A commitment which **successfully opens** to (L, t) can only be produced by a prover **located at** (L, t).
- Each commitment has a **unique position** which it can open to, even among multiple points the prover may have occupied.

New Tool: Position Commitments

01

Commit Phase

Prover at spacetime point (L, t) interacts with verifiers to create a position commitment C.

02

Prover Reveals

At any later time, the prover can announce their location (L, t), along with some reveal string, to the verifiers.

03

Verification

The verifiers decide whether to accept or reject the prover's claimed location.

Security Properties

(Honest-Verifier) Hiding Property

- (Honest-but-curious) verifiers learn no information about (L, t) during the Commit phase.
- No matter where the prover was in R, their view is exactly the same.

Position Binding

- A commitment which **successfully opens** to (L, t) can only be produced by a prover **located at** (L, t).
- Each commitment has a **unique position** which it can open to, even among multiple points the prover may have occupied.

New Tool: Position Commitments

01

Commit Phase

Prover at spacetime point (L, t) interacts with verifiers to create a position commitment C.

02

Prover Reveals

At any later time, the prover can announce their location (L, t), along with some reveal string, to the verifiers.

03

Verification

The verifiers decide whether to accept or reject the prover's claimed location.

Security Properties

(Honest-Verifier) Hiding Property

- (Honest-but-curious) verifiers learn no information about (L, t) during the Commit phase.
- No matter where the prover was in R, their view is exactly the same.

Position Binding

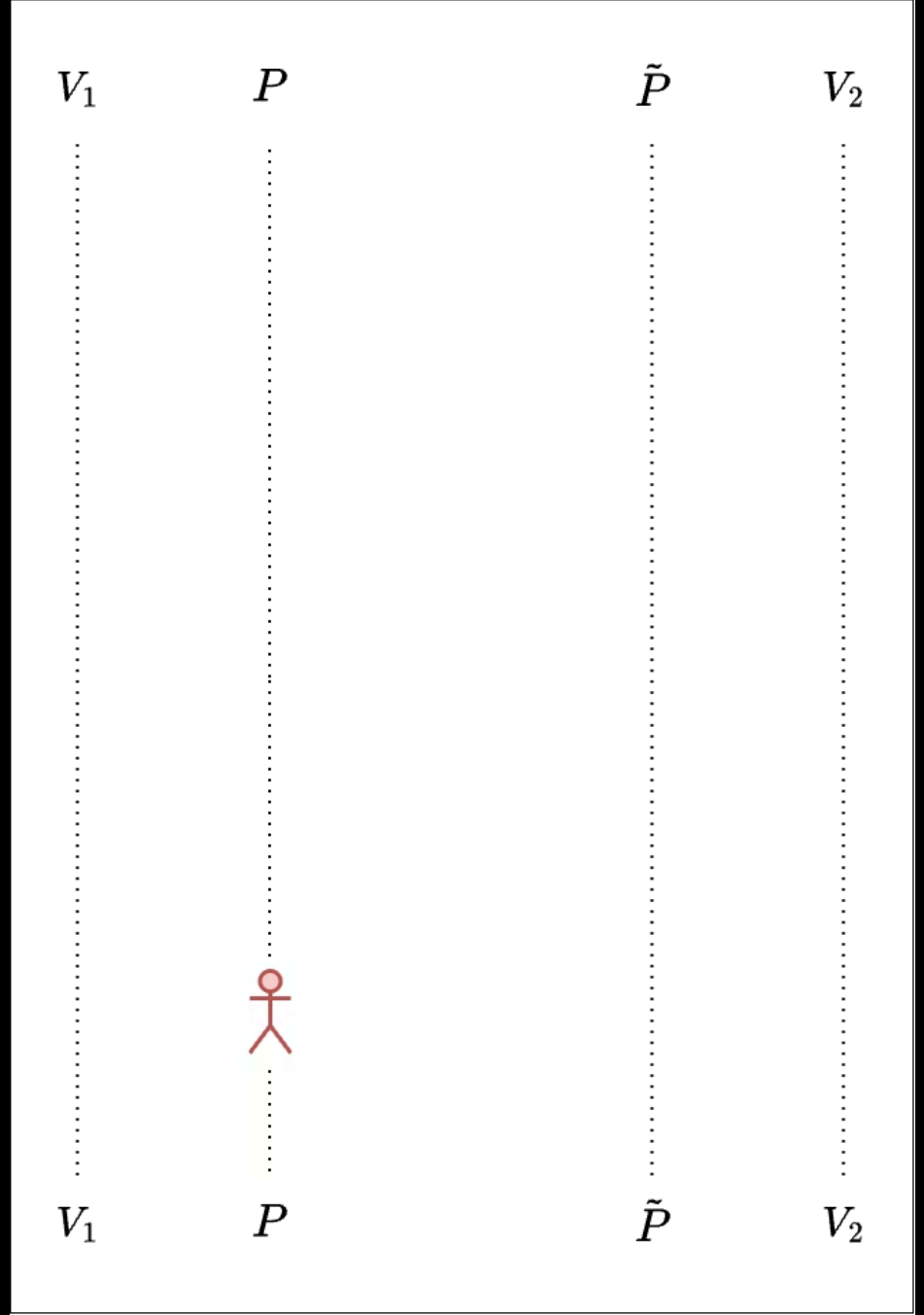
- A commitment which **successfully opens** to (L, t) can only be produced by a prover **located at** (L, t).
- Each commitment has a **unique position** which it can open to, even among multiple points the prover may have occupied.

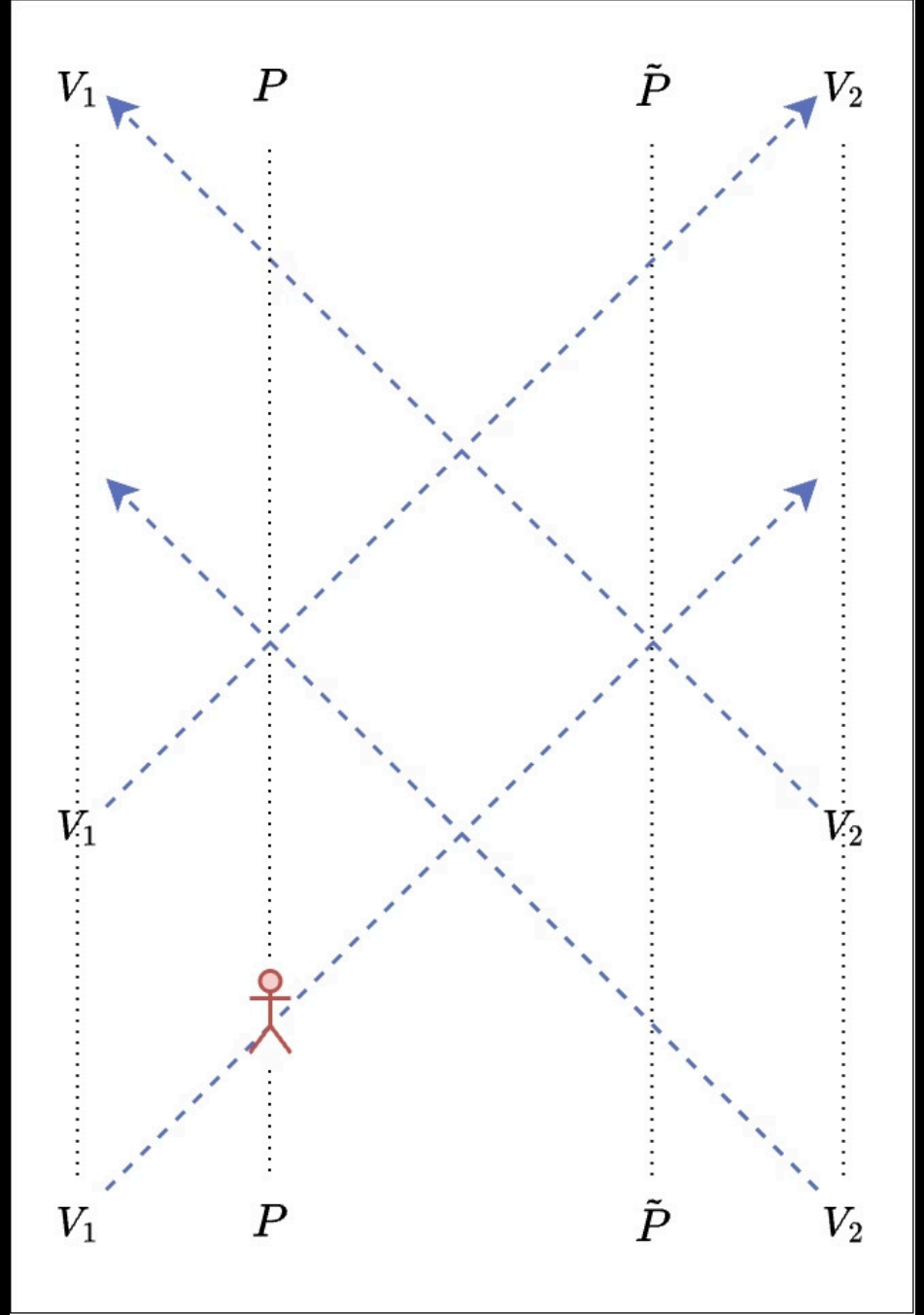
How Should We Build These?

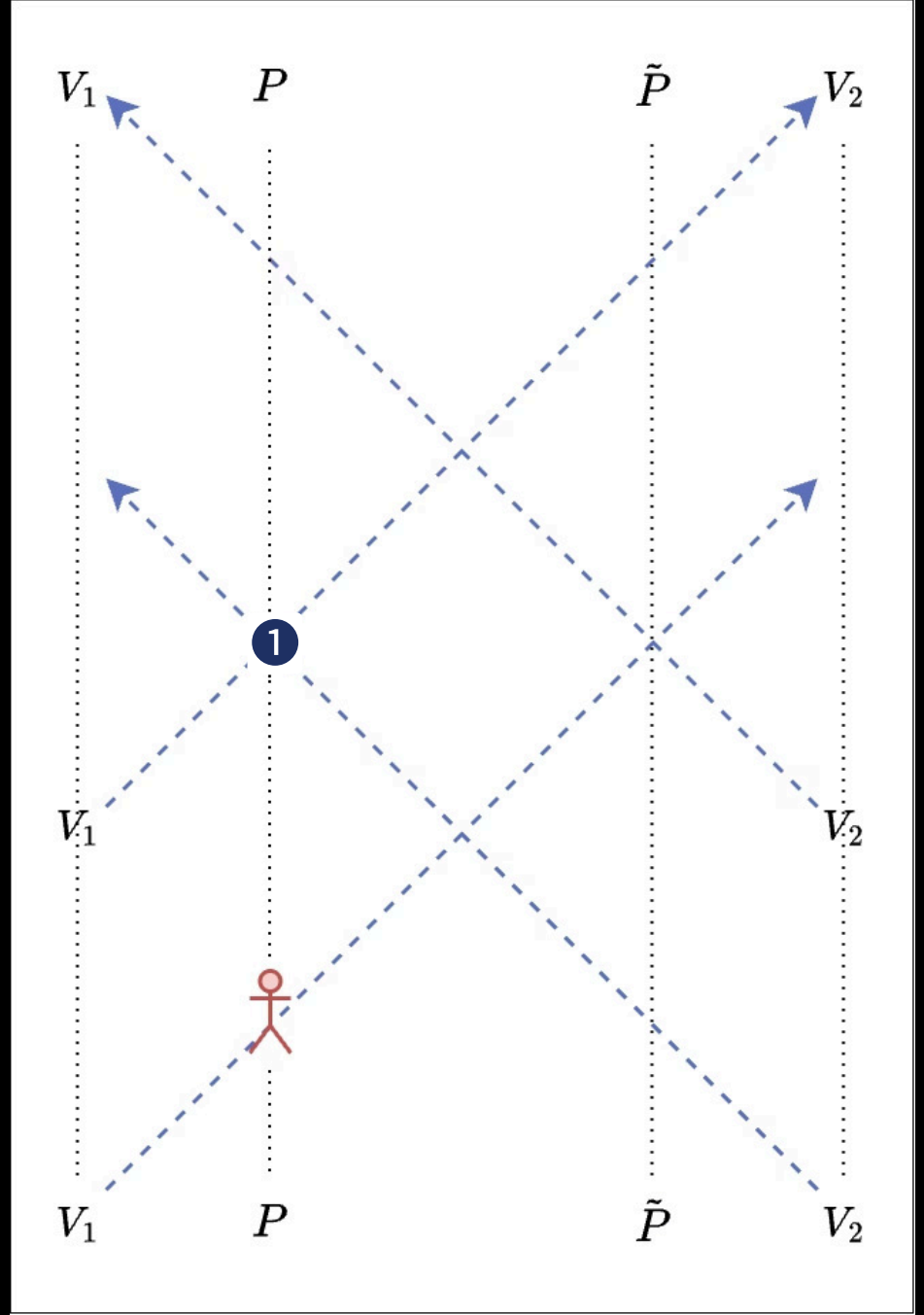
Two Properties in Tension:

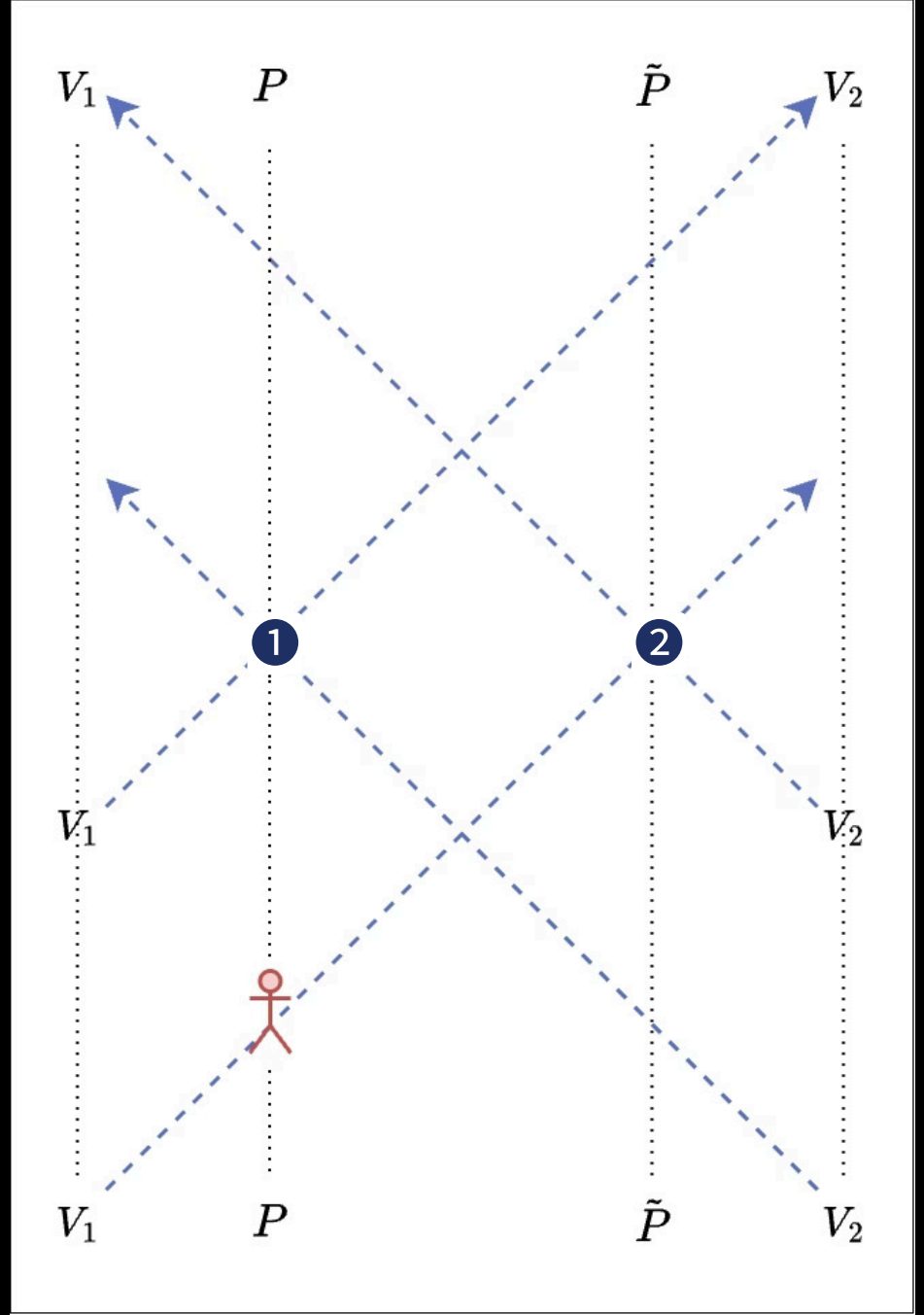
- **Soundness** of QPV requires **precise timing constraints** on the prover's messages, corresponding to their exact position.
- On the other hand, **zero-knowledge** requires that the verifiers see the **same view** regardless of the prover's position.

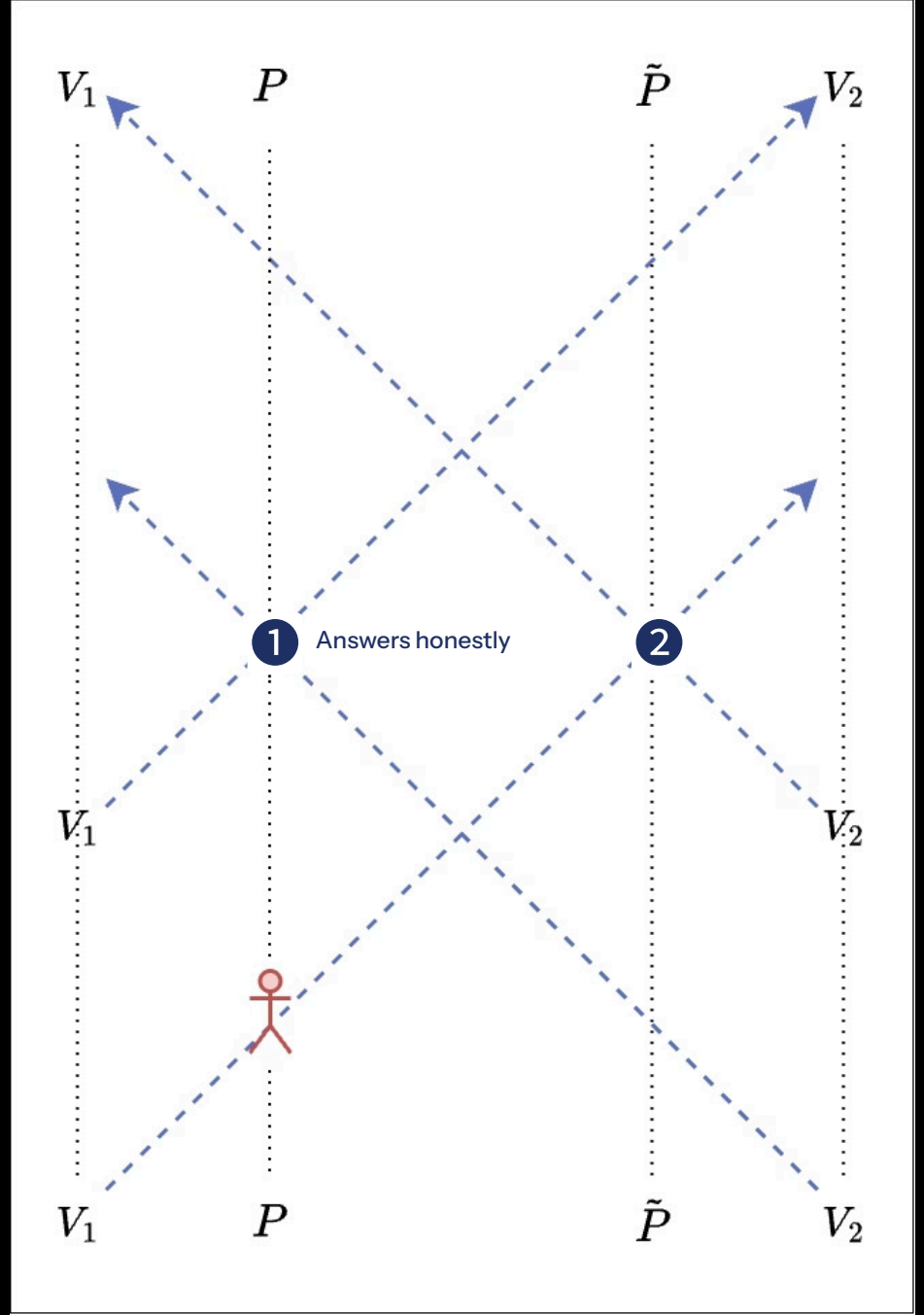
Example: Committing to Either Position P or $\tilde{P}$

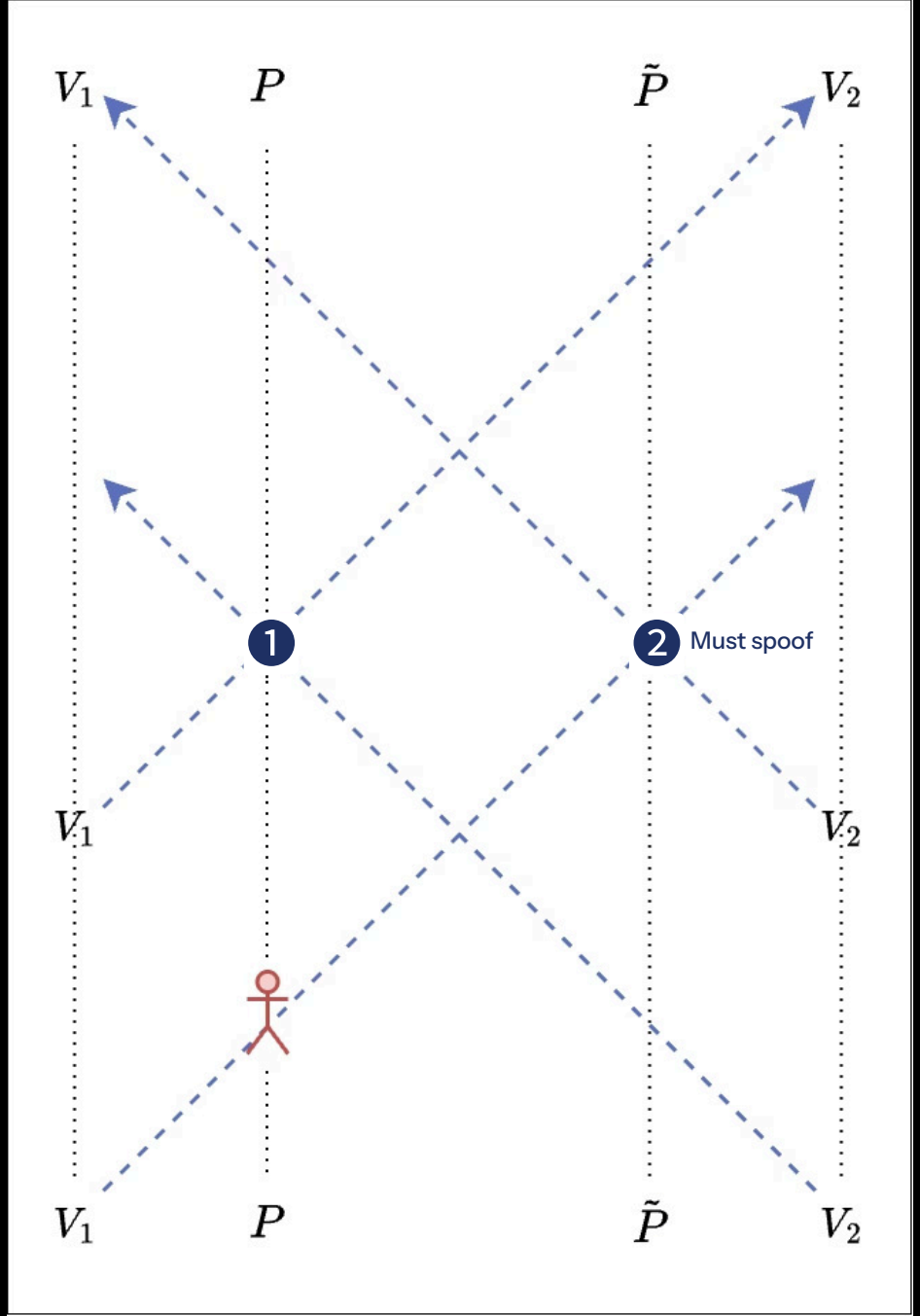

V_1
P
$\tilde{P}$
V_2
V_1
P
$\tilde{P}$
V_2

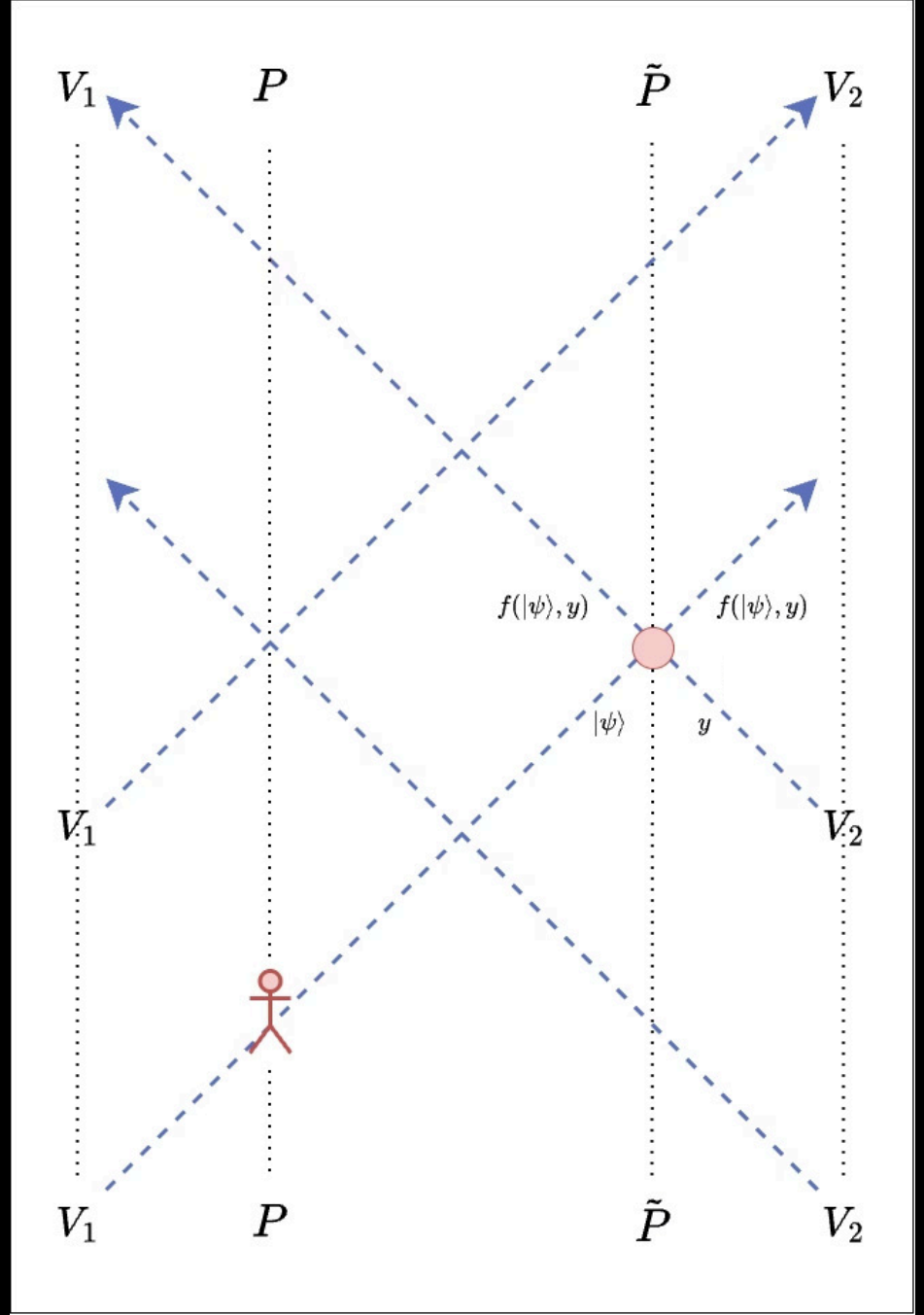

V_1
P
$\tilde{P}$
V_2
V_1
V_2
V_1
P
$\tilde{P}$
V_2

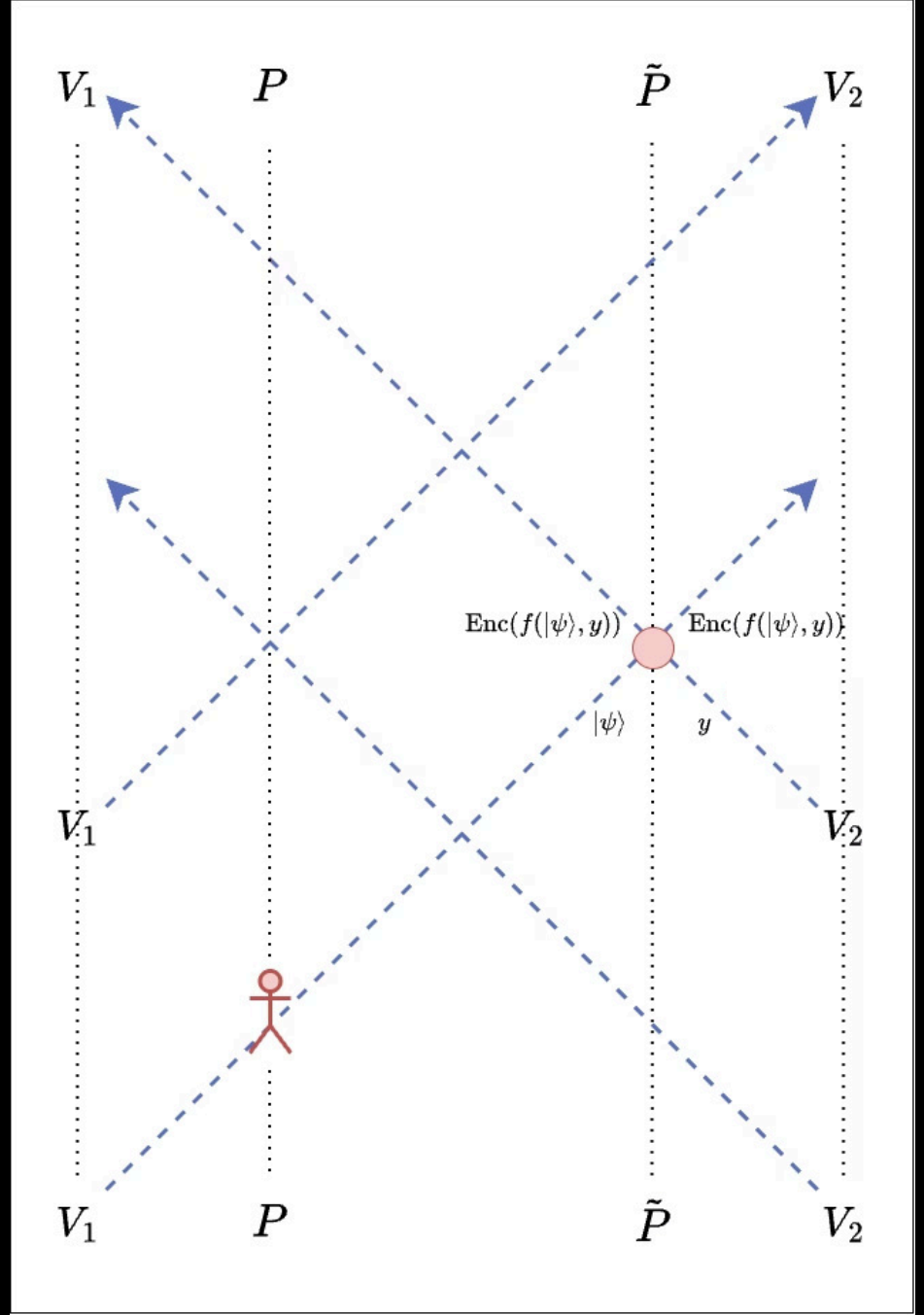

V_1
P
$\tilde{P}$
V_2
1
V_1
V_2
V_1
P
$\tilde{P}$
V_2

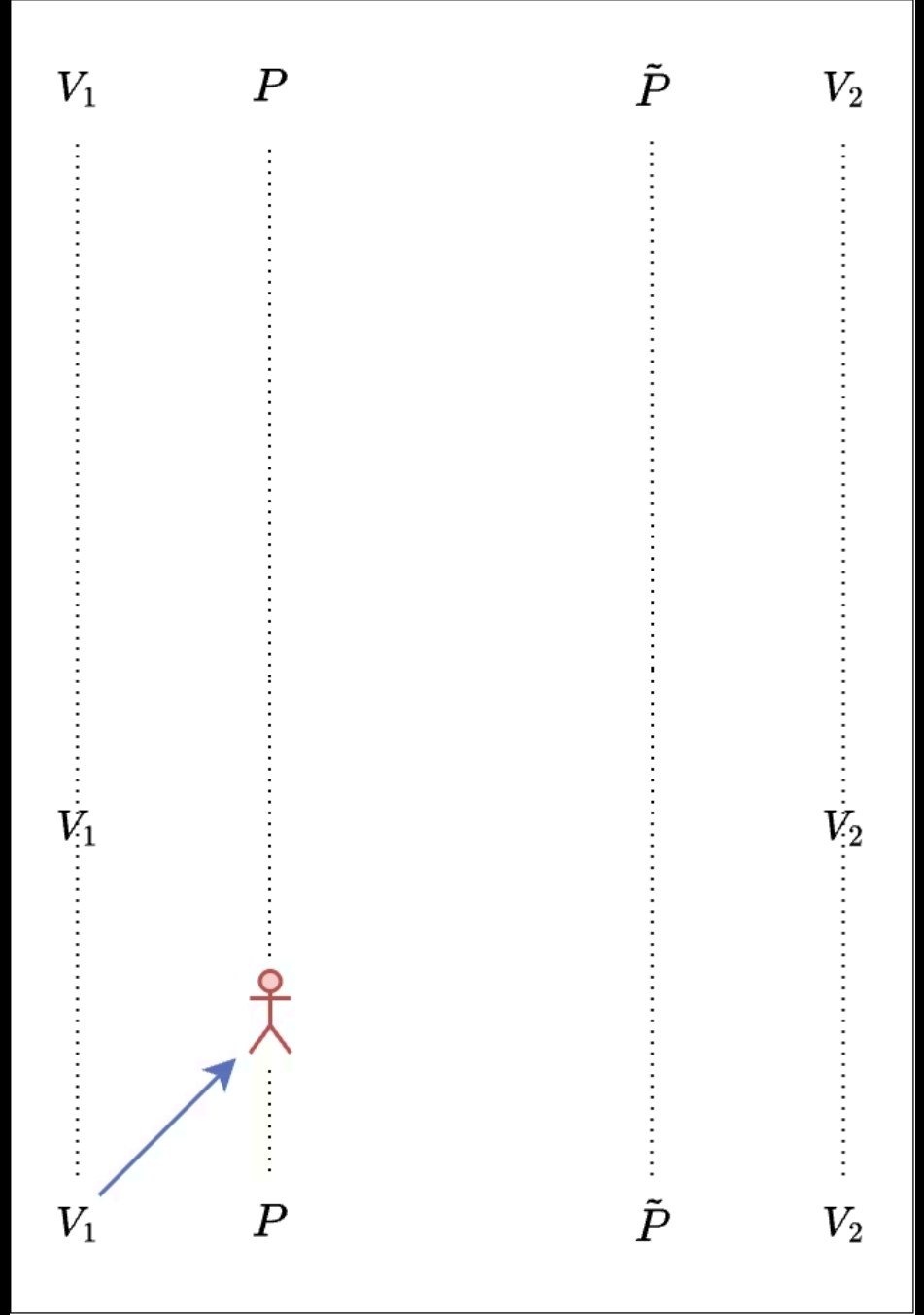

V_1
P
$\tilde{P}$
V_2
1
2
V_1
V_2
V_1
P
$\tilde{P}$
V_2

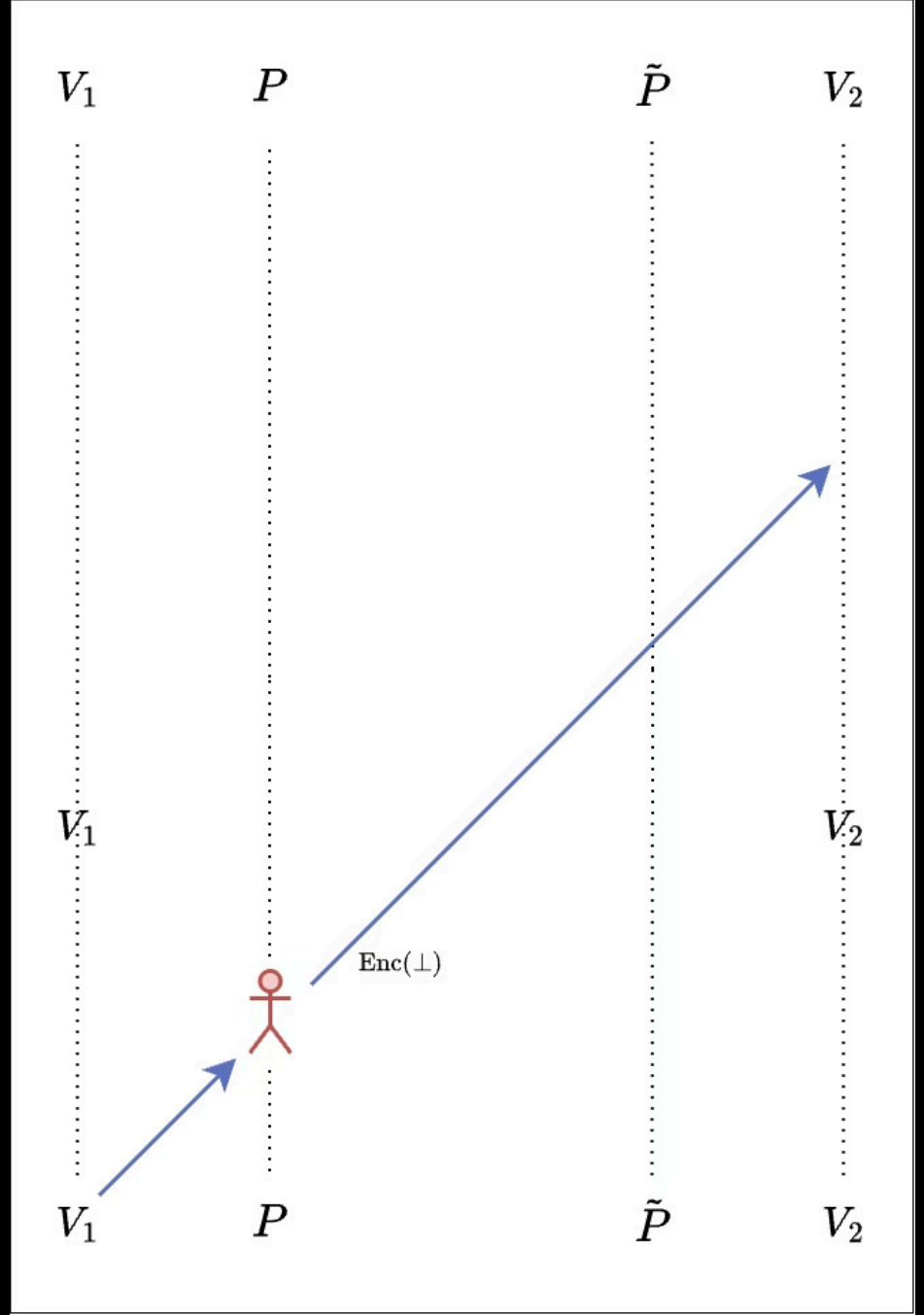

V_1
P
$\tilde{P}$
V_2
1
Answers honestly
2
V_1
V_2
V_1
P
$\tilde{P}$
V_2

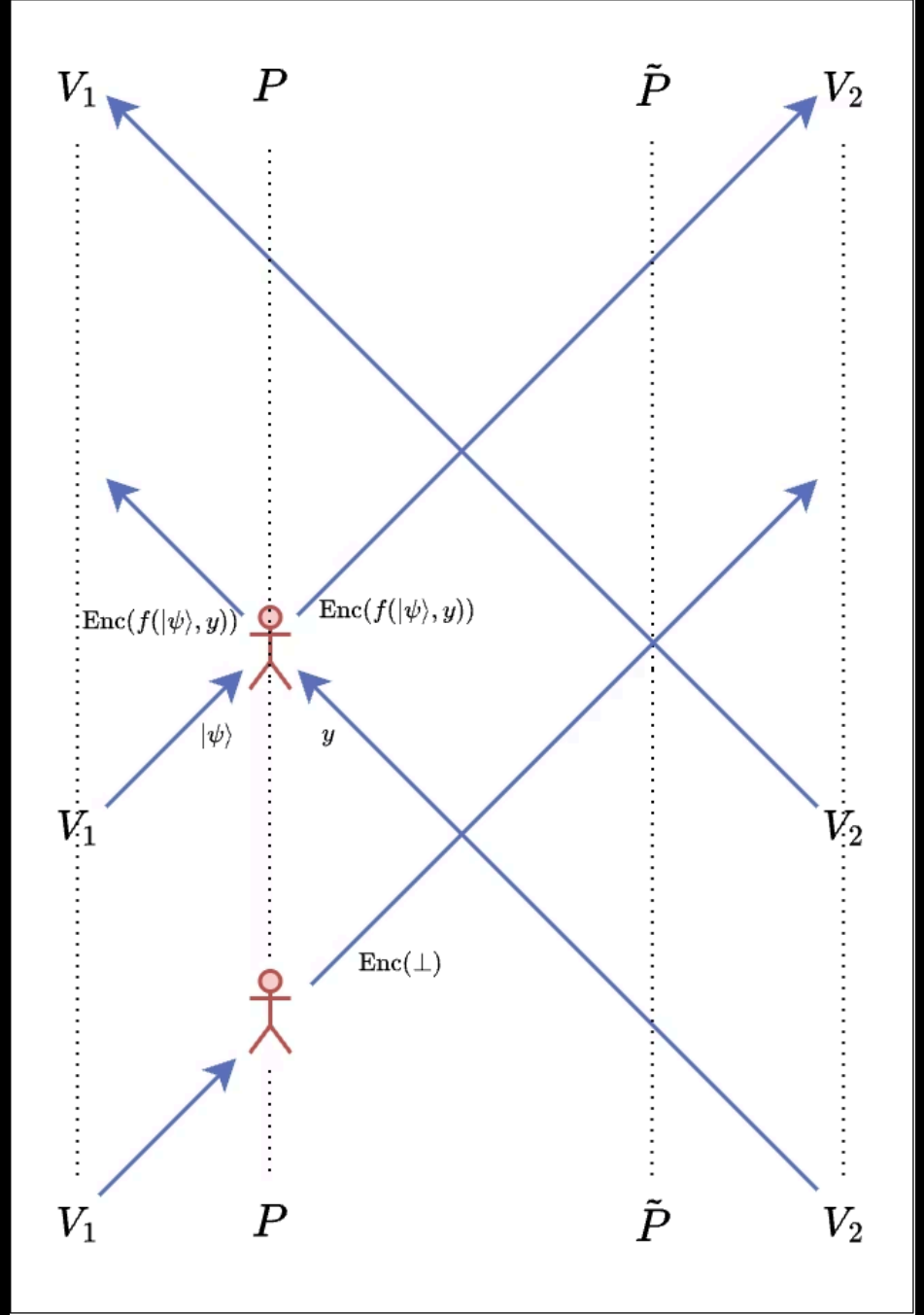

V_1
P
$\tilde{P}$
V_2
1
2
Must spoof
V_1
V_2
V_1
P
$\tilde{P}$
V_2

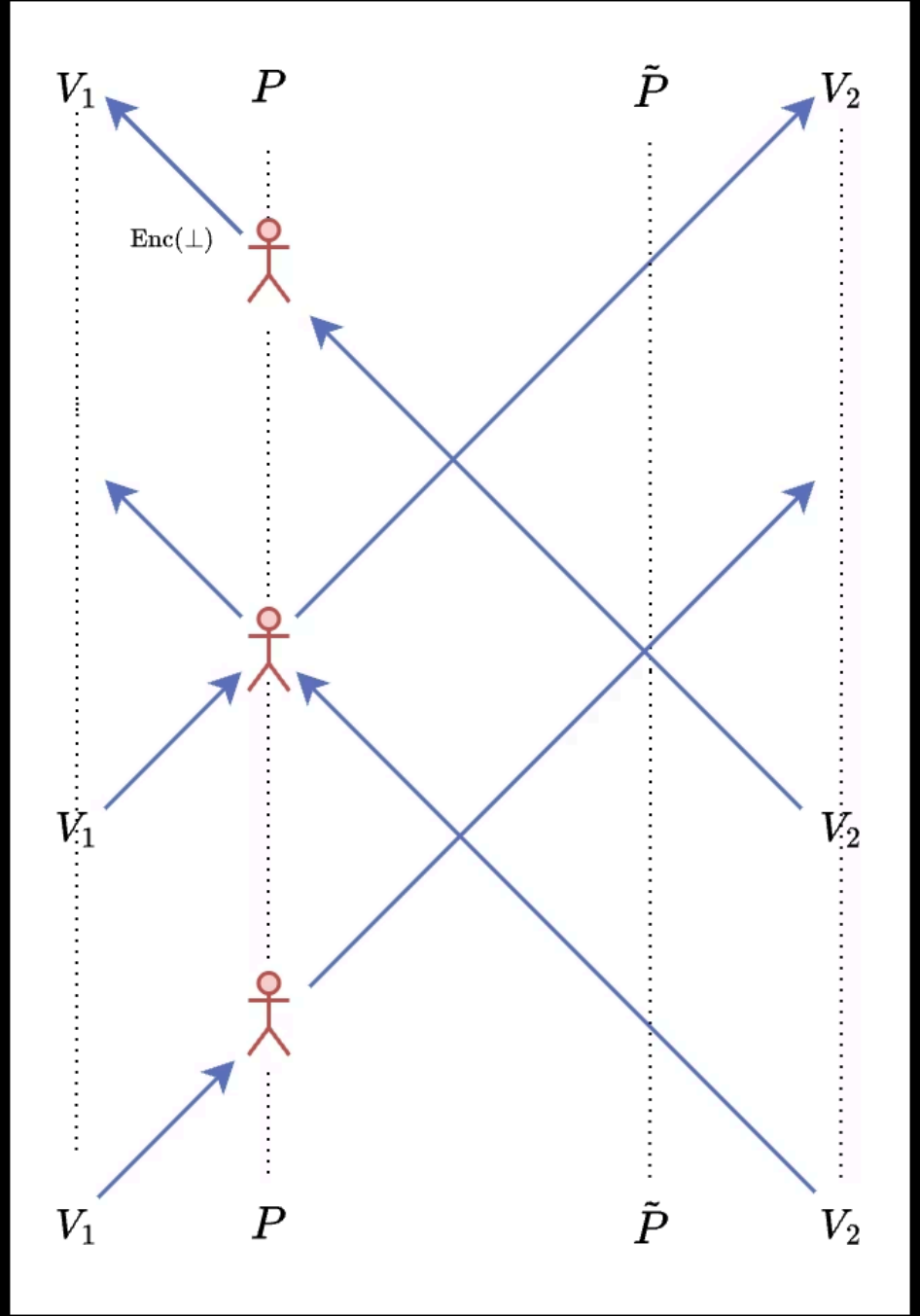

V_1
P
$\tilde{P}$
V_2
$f(|\psi\rangle, y)$
$f(|\psi\rangle, y)$
$|\psi\rangle$
y
V_1
V_2
V_1
P
$\tilde{P}$
V_2

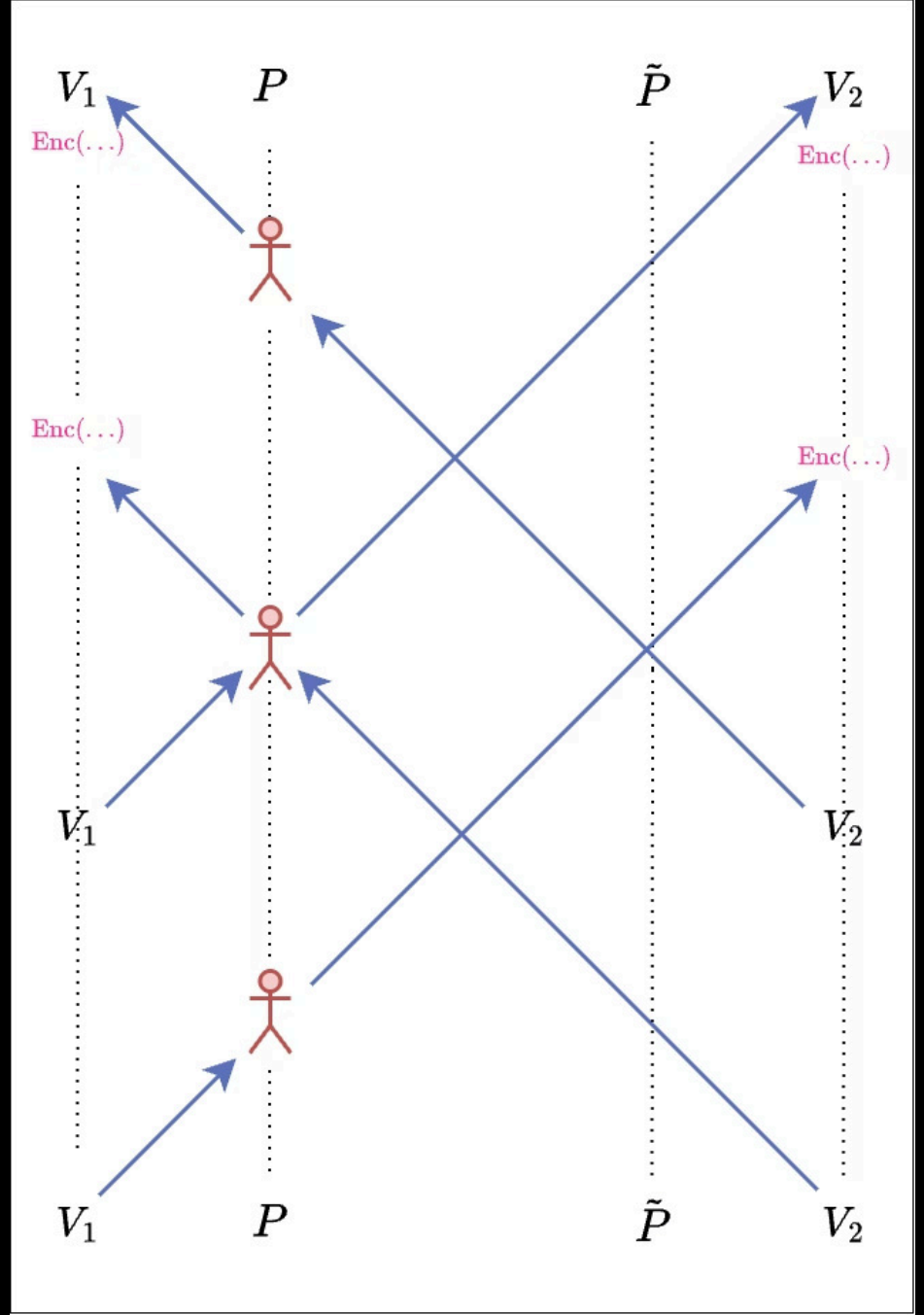

V_1
P
$\tilde{P}$
V_2
$\mathrm{Enc}(f(|\psi\rangle, y))$
$\mathrm{Enc}(f(|\psi\rangle, y))$
$|\psi\rangle$
y
V_1
V_2
V_1
P
$\tilde{P}$
V_2


V_1
P
$\tilde{P}$
V_2
V_1
V_2
V_1
P
$\tilde{P}$
V_2


V_1
P
$\tilde{P}$
V_2
V_1
V_2
Enc(⊥)
V_1
P
$\tilde{P}$
V_2


V_1
P
$\tilde{P}$
V_2
$\mathrm{Enc}(f(|\psi\rangle, y))$
$\mathrm{Enc}(f(|\psi\rangle, y))$
$|\psi\rangle$
y
V_1
V_2
$\mathrm{Enc}(\perp)$
V_1
P
$\tilde{P}$
V_2


V_1
P
$\tilde{P}$
V_2
Enc(⊥)
V_1
V_2
V_1
P
$\tilde{P}$
V_2


V_1
P
$\tilde{P}$
V_2
Enc(...)
Enc(...)
Enc(...)
Enc(...)
V_1
V_2
V_1
P
$\tilde{P}$
V_2

Position Commitments

1

- For **each point** $x \in R$
- Verifiers send a PV challenge to x.

2

- If prover is at x: send back a PV response
- If prover is not at x: dummy messages

3

- Prover encrypts all messages under a secret key
- Prover gets a head start on the verifiers, in order to meet all timing constraints

Position Commitments

Privacy: No matter where the prover is, the verifiers' view during the Commit phase is the same:

They see a stream of encrypted messages, one for each PV timing constraint of points in R.

Position Commitments

Privacy: No matter where the prover is, the verifiers' view during the Commit phase is the same:

They see a stream of encrypted messages, one for each PV timing constraint of points in R.

Security: In order to spoof the reveal, the prover must have cheated some actual PV challenge.

We can prove this by showing a reduction from the position commitment experiment to the PV experiment.

Position Commitment Construction: In Detail

Protecting The Prover's Whereabouts: Two Key Tools

- Prover's **head start** on the verifiers lets it meet timing constraints for every point in the region giving **plausible deniability** of having answered all challenges
- Dummy responses are **hidden** with SKE: prover simply sends $\mathrm{Enc}(\perp)$

Covering The Entire Region In Parallel

- The verifiers send challenges to **every spacetime point** in the region **simultaneously**.
- The prover sends the verifiers a **continuous stream of encrypted responses**.

Verification

- To reveal, the prover sends the verifiers its position (L, t) and its secret key.
- The verifiers decrypt their transcript. They then check that there's a successful PV response corresponding to (L, t), and that all other responses were dummy messages.

Position Commitment Construction: In Detail

Protecting The Prover's Whereabouts: Two Key Tools

- Prover's **head start** on the verifiers lets it meet timing constraints for every point in the region giving **plausible deniability** of having answered all challenges
- Dummy responses are **hidden** with SKE: prover simply sends $\mathrm{Enc}(\perp)$

Covering The Entire Region In Parallel

- The verifiers send challenges to **every spacetime point** in the region **simultaneously**.
- The prover sends the verifiers a **continuous stream of encrypted responses**.

Verification

- To reveal, the prover sends the verifiers its position (L, t) and its secret key.
- The verifiers decrypt their transcript. They then check that there's a successful PV response corresponding to (L, t), and that all other responses were dummy messages.

Position Commitment Construction: In Detail

Protecting The Prover's Whereabouts: Two Key Tools

- Prover's **head start** on the verifiers lets it meet timing constraints for every point in the region giving **plausible deniability** of having answered all challenges
- Dummy responses are **hidden** with SKE: prover simply sends $\mathrm{Enc}(\perp)$

Covering The Entire Region In Parallel

- The verifiers send challenges to **every spacetime point** in the region **simultaneously**.
- The prover sends the verifiers a **continuous stream of encrypted responses**.

Verification

- To reveal, the prover sends the verifiers its position (L, t) and its secret key.
- The verifiers decrypt their transcript. They then check that there's a successful PV response corresponding to (L, t), and that all other responses were dummy messages.

From Position Commitments to Zero-Knowledge Position Verification

Classical ZK Proofs

Any NP statement (an efficiently-checkable "there exists" statement) can be proved in zero knowledge.

Putting it Together

Create a position commitment C. Prove in zero knowledge, "the position is in R."

Position Commitment

A string for which **there exists** an opening, such that it's **efficient to check** whether the revealed location is in R

Tangent: Models of QPV

Bounded-Entanglement Model

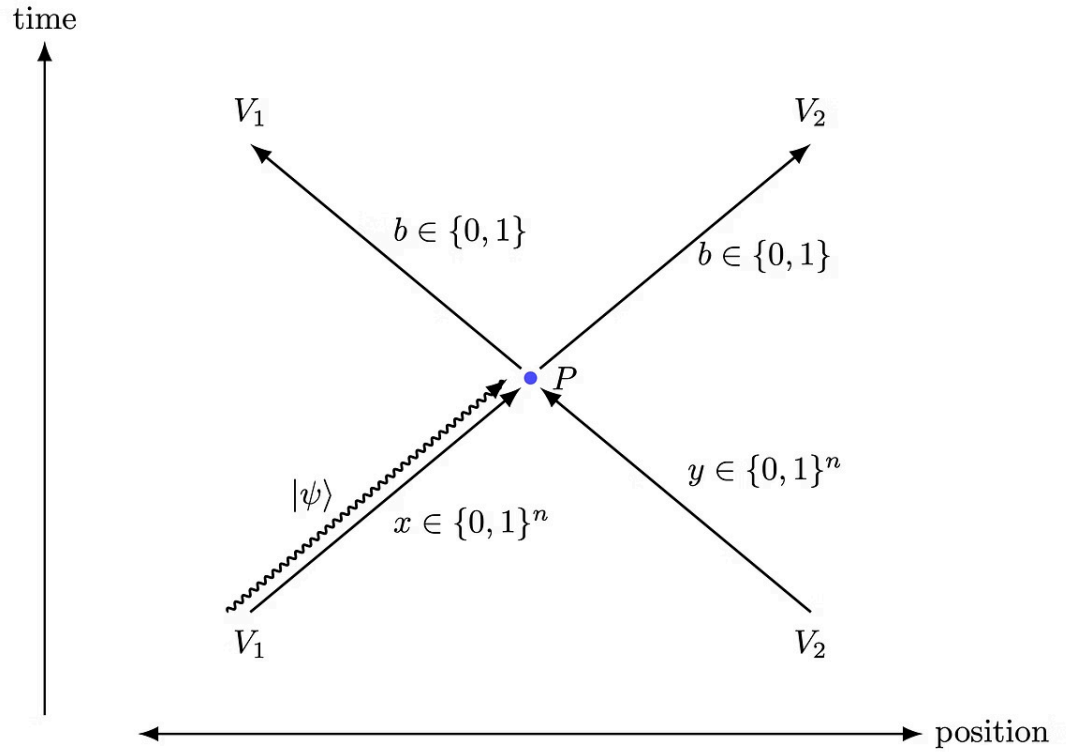
The f-BB84 protocol is provably secure against attackers with $O(\log n)$ ebits of entanglement (Bluhm et al., 2022).

Random Oracle Model (ROM)

In the ROM, it is possible to get security against **fully unbounded adversaries** (Unruh, 2014).

Cryptographic Plain Model

Assuming strong versions of LWE, there are secure PV protocols **with classical verifiers** (but still quantum prover!) (Liu et al., 2022).



Summing Up: Our Main Result

Formal Statement

Given post-quantum one-way functions and secure position verification, there exist Honest-Verifier Zero-Knowledge Position Verifications protocol for every finite spacetime region.

A Generic Transformation

We can "upgrade" essentially all QPV protocols to have honest-verifier zero knowledge, assuming the existence of private-key cryptography.

Security Bounds

In the bounded-entanglement model, our transformation roughly preserves the security of the original position verification protocol — up to a factor of two. We believe this likely also holds for other models, such as the ROM or the cryptographic plain model.

Future Directions

Security Against Malicious Verifiers

- We assume verifiers are honest-but-curious.
- Can we extend this to arbitrary verifiers? Seems difficult.

Capturing Even More Applications

- Notion of **identity** seems to be a fascinating direction to explore.
- How should we think about **linking** between multiple position proofs?

Efficiency and More Realistic Modeling

- The typical QPV model is somewhat idealistic, from an experimental POV.
- In follow up work, we hope to relax some of these assumptions and move toward practical implementability.

Future Directions

Security Against Malicious Verifiers

- We assume verifiers are honest-but-curious.
- Can we extend this to arbitrary verifiers? Seems difficult.

Capturing Even More Applications

- Notion of **identity** seems to be a fascinating direction to explore.
- How should we think about **linking** between multiple position proofs?

Efficiency and More Realistic Modeling

- The typical QPV model is somewhat idealistic, from an experimental POV.
- In follow up work, we hope to relax some of these assumptions and move toward practical implementability.

Future Directions

Security Against Malicious Verifiers

- We assume verifiers are honest-but-curious.
- Can we extend this to arbitrary verifiers? Seems difficult.

Capturing Even More Applications

- Notion of **identity** seems to be a fascinating direction to explore.
- How should we think about **linking** between multiple position proofs?

Efficiency and More Realistic Modeling

- The typical QPV model is somewhat idealistic, from an experimental POV.
- In follow up work, we hope to relax some of these assumptions and move toward practical implementability.

Thank You!

Any questions?