Quantum Pseudorandom Scramblers

Chuhan Lu
Portland State University

Minglong Qin
Nanjing University

Fang Song
Portland State University

Penghui Yao
Nanjing University
Hefei National Lab.

Mingnan Zhao
Nanjing University

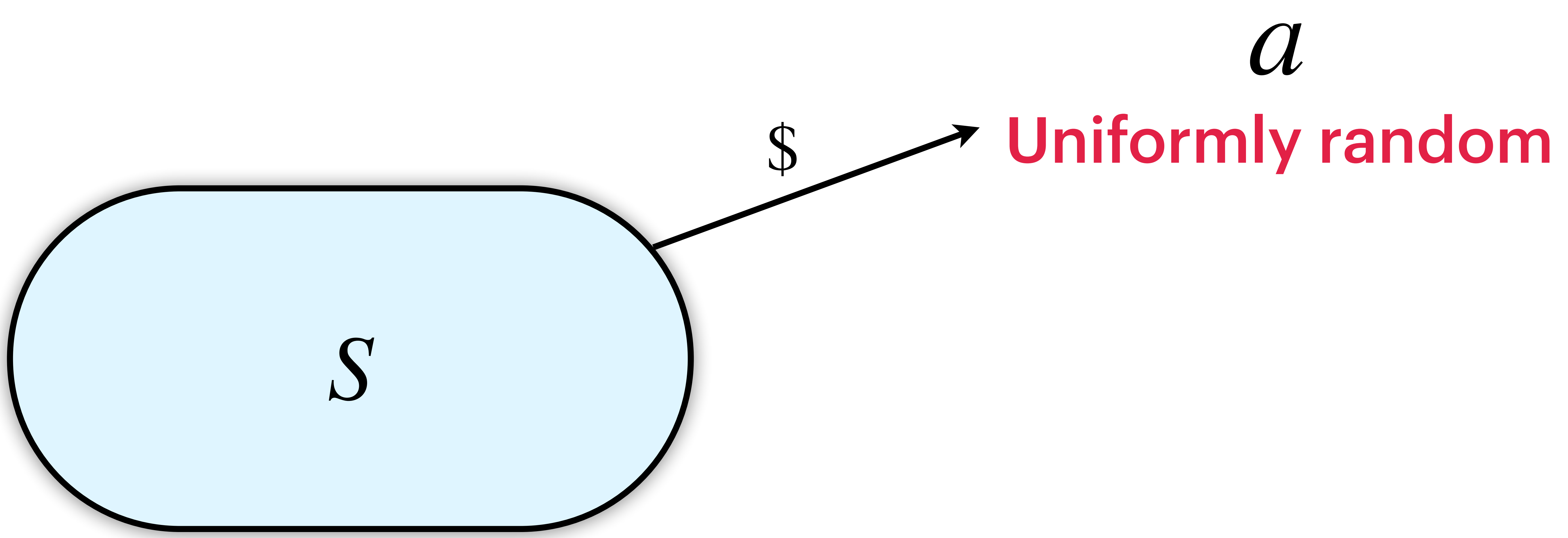
04/12/2024 @ TCC 2024, Milan, Italy

Pseudorandomness

- Generating true randomness is usually **costly**:
  - $n2^n$ random bits for a random function

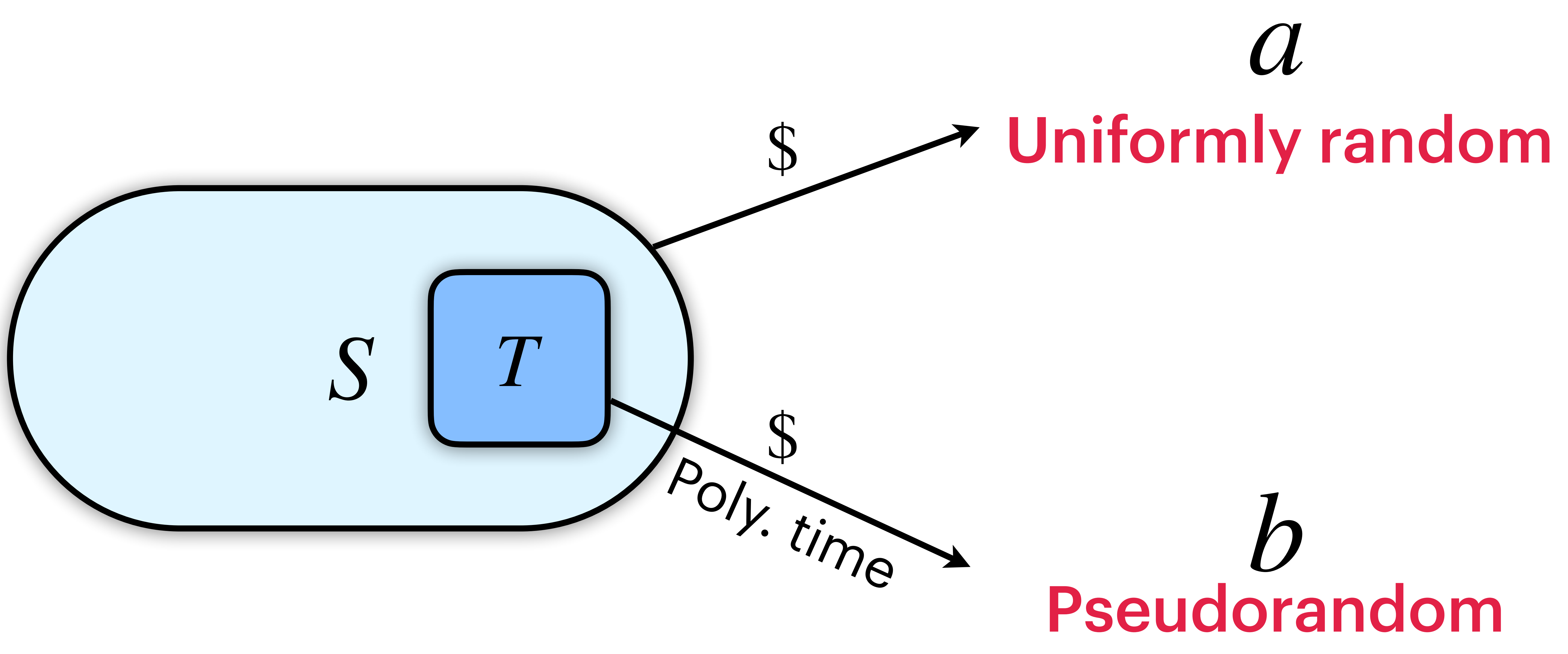
Pseudorandomness

- Generating true randomness is usually **costly**:
  - $n2^n$ random bits for a random function



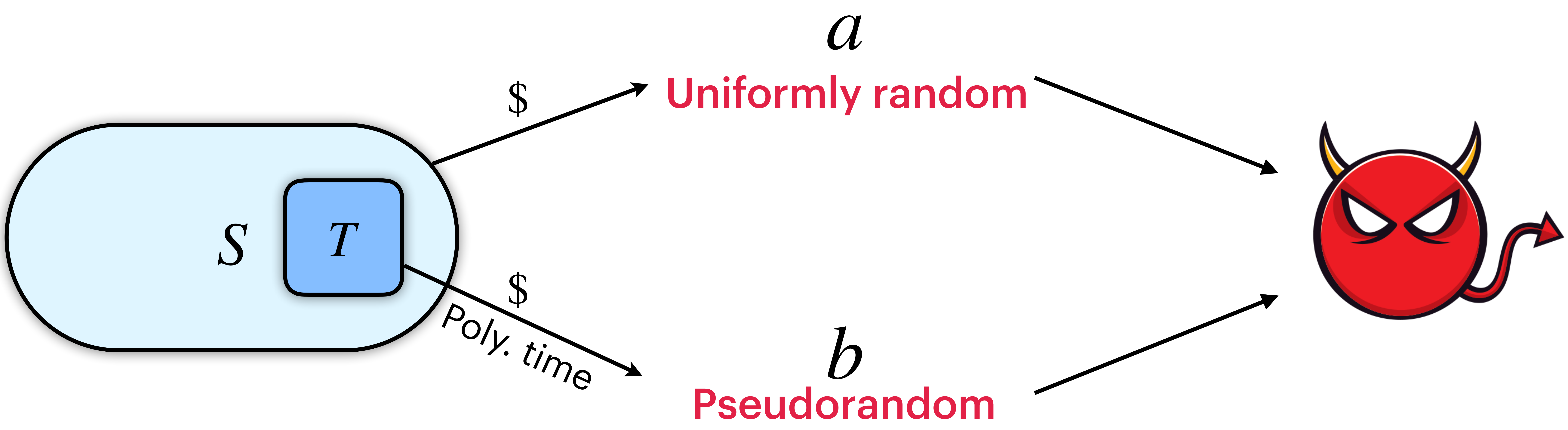
Pseudorandomness

- Generating true randomness is usually **costly**:
  - $n2^n$ random bits for a random function



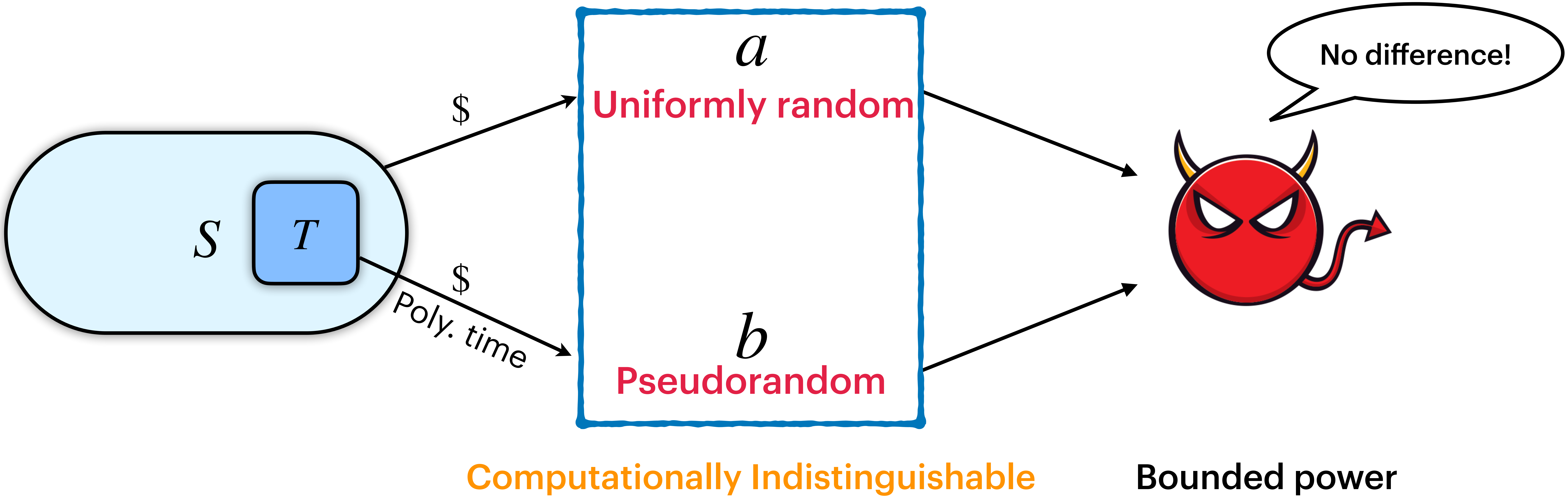
Pseudorandomness

- Generating true randomness is usually **costly**:
  - $n2^n$ random bits for a random function



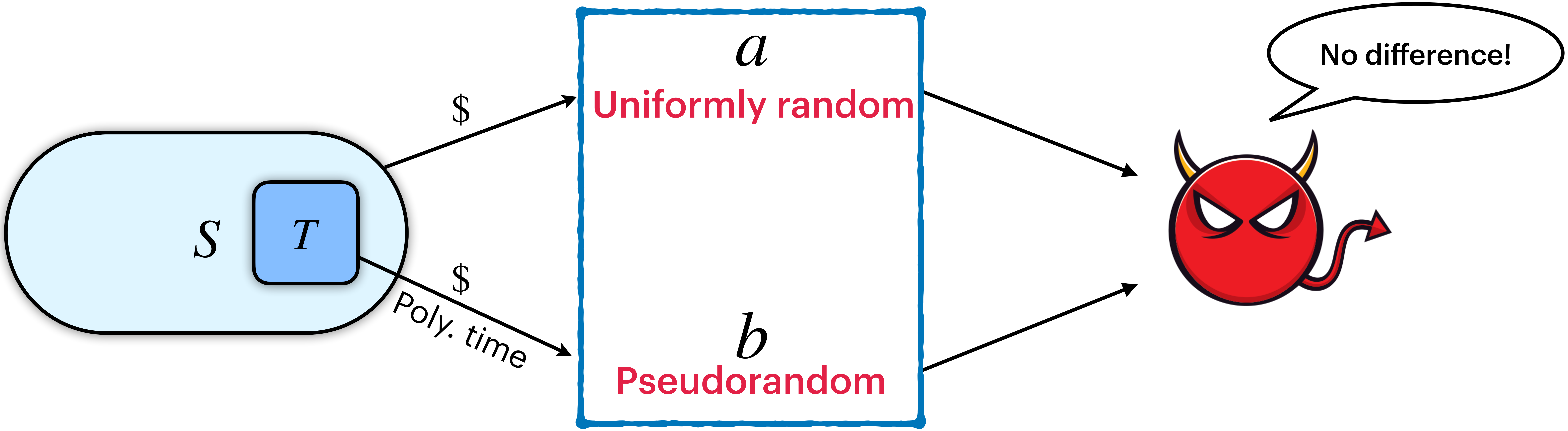
Pseudorandomness

- Generating true randomness is usually **costly**:
  - $n2^n$ random bits for a random function



Pseudorandomness

- Generating true randomness is usually **costly**:
  - $n2^n$ random bits for a random function



Computationally Indistinguishable — Bounded power

Statistically Indistinguishable — Unlimited power

Quantum Pseudorandomness

Pseudorandom State Generator (PRSG) [JLS18]

Quantum Pseudorandomness

Pseudorandom State Generator (PRSG) [JLS18]

$$k \longrightarrow \boxed{G} \longrightarrow |\phi_k\rangle$$

Quantum Pseudorandomness

Pseudorandom State Generator (PRSG) [JLS18]

$$k \longrightarrow \boxed{G} \longrightarrow |\phi_k\rangle$$

$$|\psi\rangle \longleftarrow \mu$$ **Haar random state**

Quantum Pseudorandomness

Pseudorandom State Generator (PRSG) [JLS18]

$$k \longrightarrow \boxed{G} \longrightarrow |\phi_k\rangle \qquad\qquad |\psi\rangle \longleftarrow \mu$$

Haar random state

Comp. indist. given any poly. copies

Quantum Pseudorandomness

Pseudorandom State Generator (PRSG) [JLS18]

Pseudorandom States

$k \longrightarrow \boxed{G} \longrightarrow |\phi_k\rangle \qquad\qquad |\psi\rangle \longleftarrow \mu$ Haar random state

Comp. indist. given any poly. copies

Quantum Pseudorandomness

Pseudorandom State Generator (PRSG) [JLS18]

Pseudorandom States

$k \longrightarrow \boxed{G} \longrightarrow |\phi_k\rangle$ $\qquad\qquad$ $|\psi\rangle \longleftarrow \mu$ Haar random state

Comp. indist. given any poly. copies

PRSGs exist assuming the existence of QPRFs. [JLS18, BS19]

Quantum Pseudorandomness

Pseudorandom State Generator (PRSG) [JLS18]

Pseudorandom States

$k \longrightarrow \boxed{G} \longrightarrow |\phi_k\rangle$ $\qquad\qquad$ $|\psi\rangle \longleftarrow \mu$ **Haar random state**

Comp. indist. given any poly. copies

$$|\phi_k\rangle = \frac{1}{\sqrt{N}} \sum_{x \in \{0,1\}^n} (-1)^{\mathrm{PRF}_k(x)} |x\rangle$$

$$|\phi_k\rangle = \frac{1}{\sqrt{N}} \sum_{x \in \{0,1\}^n} \omega_N^{\mathrm{PRF}_k(x)} |x\rangle$$

PRSGs exist assuming the existence of QPRFs. [JLS18, BS19]

Quantum Pseudorandomness

Pseudorandom State Generator (PRSG) [JLS18]

Pseudorandom States

$k \longrightarrow \boxed{G} \longrightarrow |\phi_k\rangle$ $\qquad\qquad$ $|\psi\rangle \longleftarrow \mu$ **Haar random state**

Comp. indist. given any poly. copies

$$|\phi_k\rangle = \frac{1}{\sqrt{N}} \sum_{x \in \{0,1\}^n} (-1)^{\mathrm{PRF}_k(x)} |x\rangle$$

PRSGs exist assuming the existence of QPRFs. [JLS18, BS19]

$$|\phi_k\rangle = \frac{1}{\sqrt{N}} \sum_{x \in \{0,1\}^n} \omega_N^{\mathrm{PRF}_k(x)} |x\rangle$$

Applications:

- Quantum money [JLS18]
- Digital signature [MY22]
- Data encryption [AGQY23]
- Quantum bit commitment [AQY22, AGQY23]
- Quantum trapdoor function [Col23]
- Quantum gravity theory [BFV20]

Quantum Pseudorandomness

Pseudorandom Unitary Operators (PRU) [JLS18]

Quantum Pseudorandomness

Pseudorandom Unitary Operators (PRU) [JLS18]

Pseudorandom Unitary Operators

$$k \longrightarrow \boxed{G} \longrightarrow U_k$$

Quantum Pseudorandomness

Pseudorandom Unitary Operators (PRU) [JLS18]

Pseudorandom Unitary Operators

$$k \longrightarrow \boxed{G} \longrightarrow U_k \qquad\qquad U \longleftarrow \mu$$

Haar random unitary

Quantum Pseudorandomness

Pseudorandom Unitary Operators (PRU) [JLS18]

Pseudorandom Unitary Operators

$k \longrightarrow \boxed{G} \longrightarrow U_k$ $\qquad\qquad$ $U \longleftarrow \mu$ Haar random unitary

Comp. indist. given oracle access

Quantum Pseudorandomness

Pseudorandom Unitary Operators (PRU) [JLS18]

Pseudorandom Unitary Operators

$k \longrightarrow \boxed{G} \longrightarrow U_k$ $\qquad$ $U \longleftarrow \mu$ Haar random unitary

Comp. indist. given oracle access

- Quantum analogue of PRFs.
- PRUs exist even if BQP=QMA [Kre21].

Quantum Pseudorandomness

Pseudorandom Unitary Operators (PRU) [JLS18]

Pseudorandom Unitary Operators

$k \longrightarrow \boxed{G} \longrightarrow U_k$ $\qquad\qquad$ $U \longleftarrow \mu$ Haar random unitary

Comp. indist. given oracle access

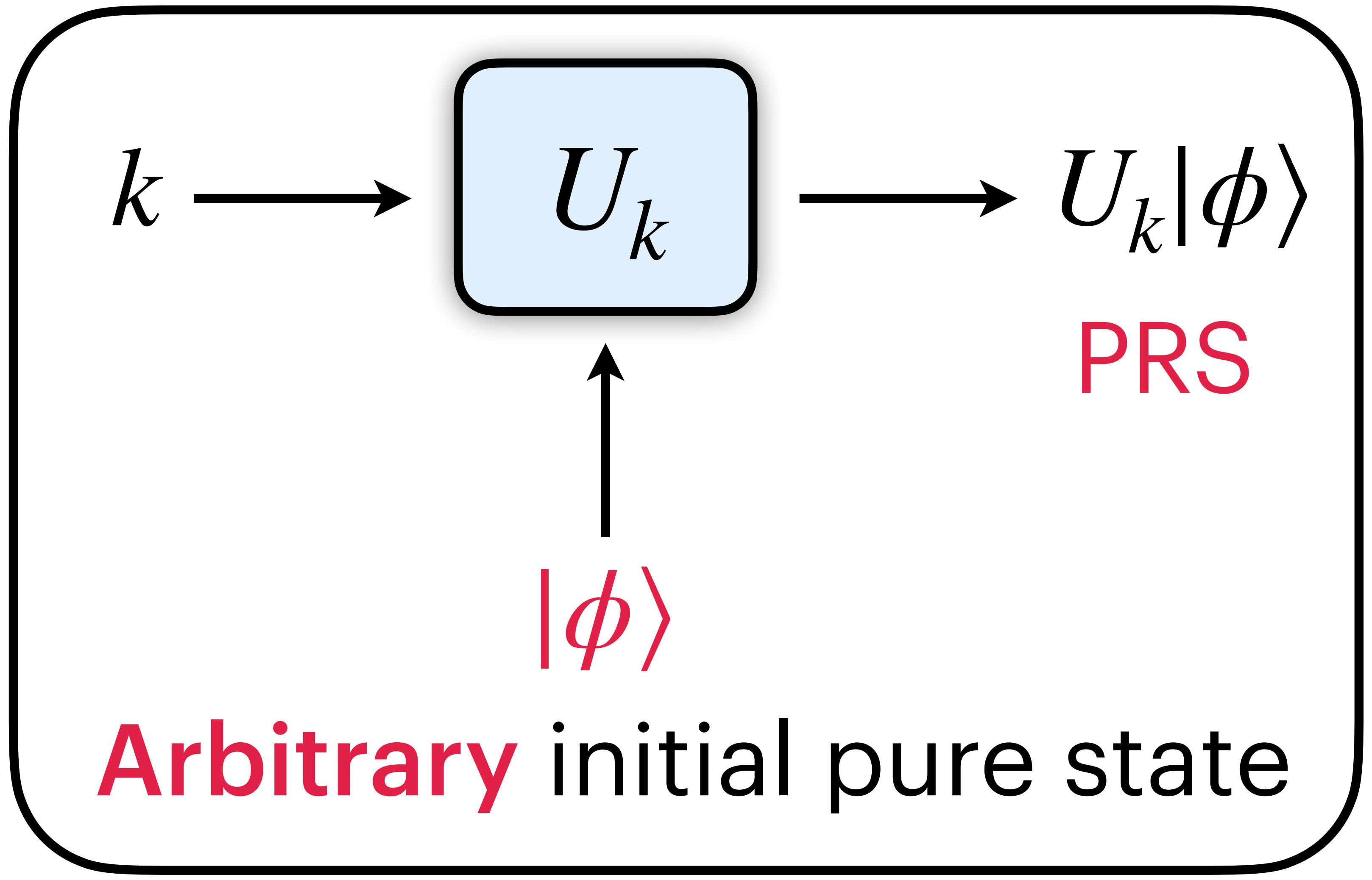
- Quantum analogue of PRFs.
- PRUs exist even if BQP=QMA [Kre21].

Construction (from OWFs)?

An open problem until very recently.

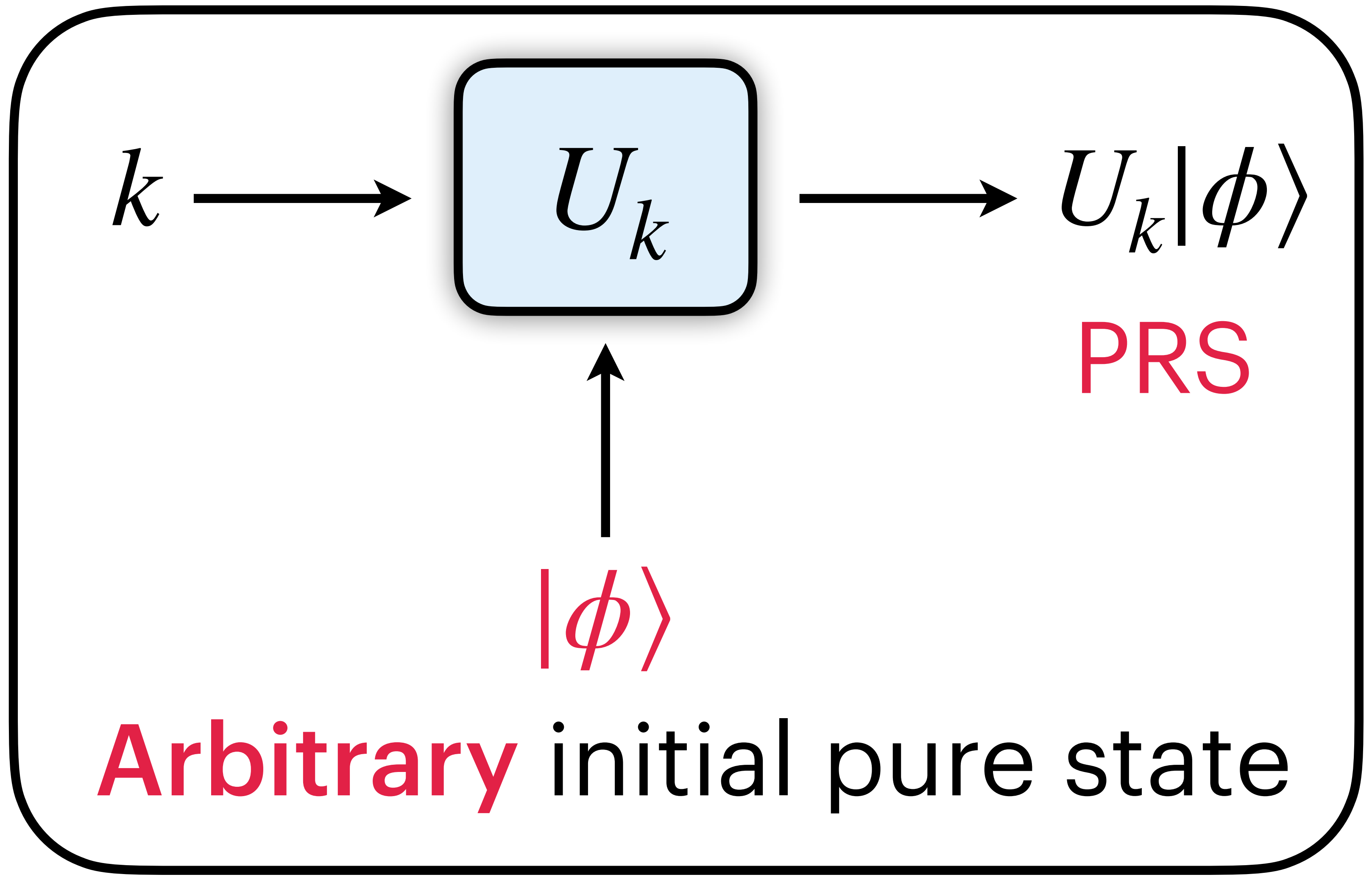
Quantum Pseudorandomness

Quantum Pseudorandomness

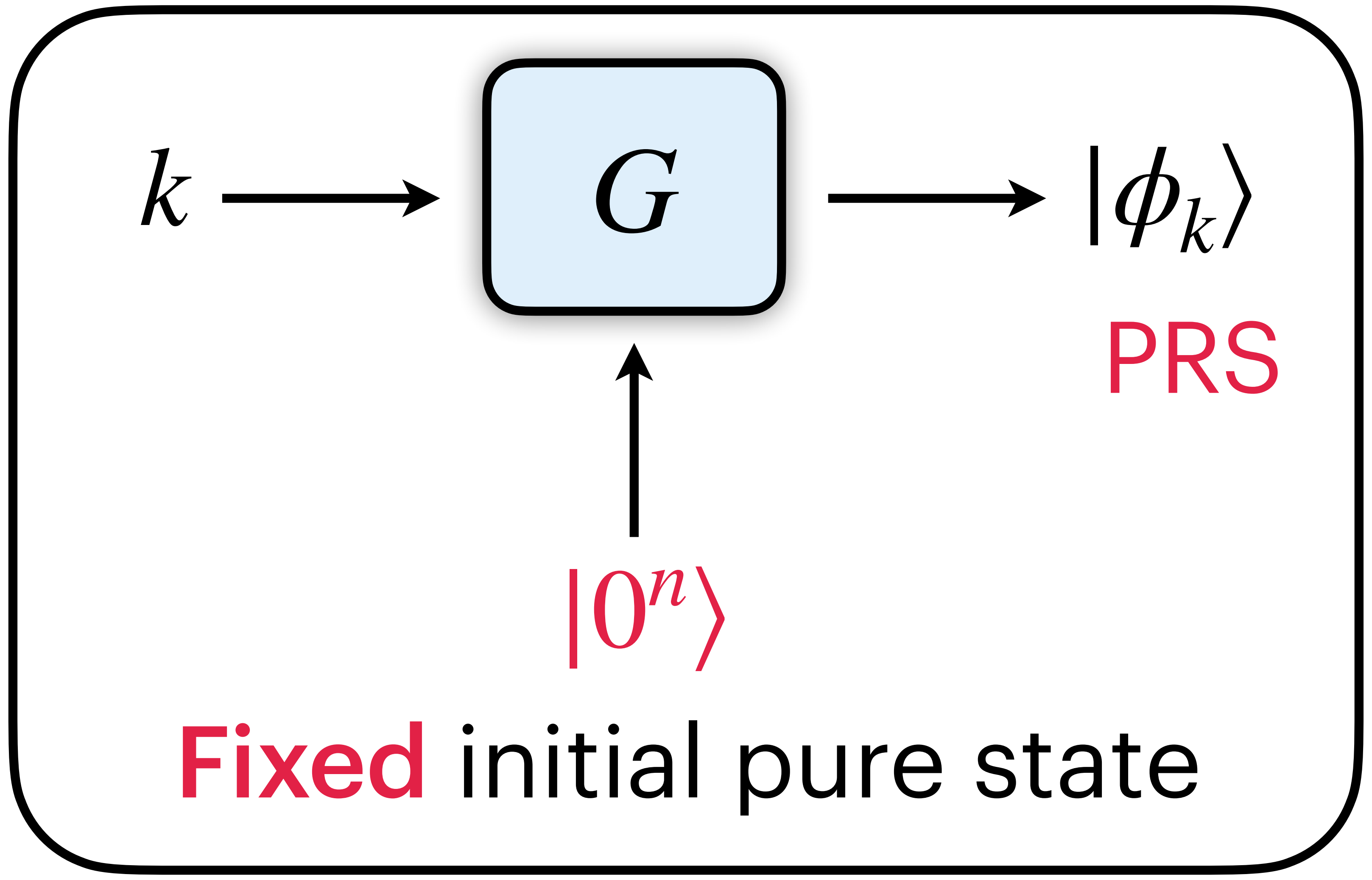


PRU

Quantum Pseudorandomness

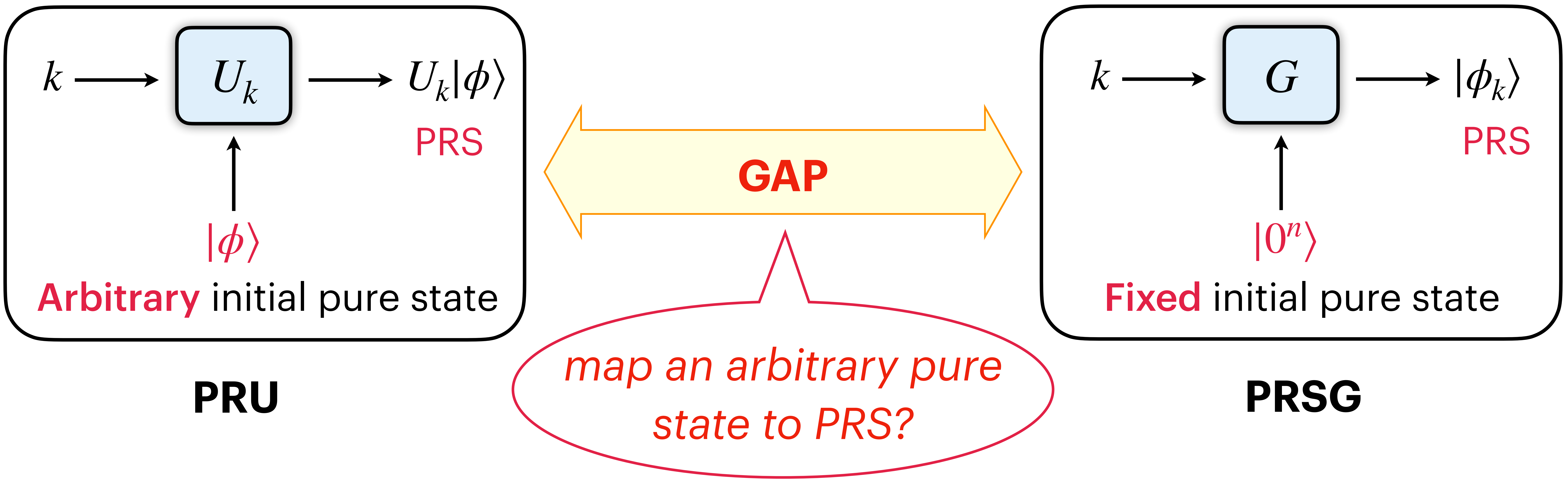


PRU

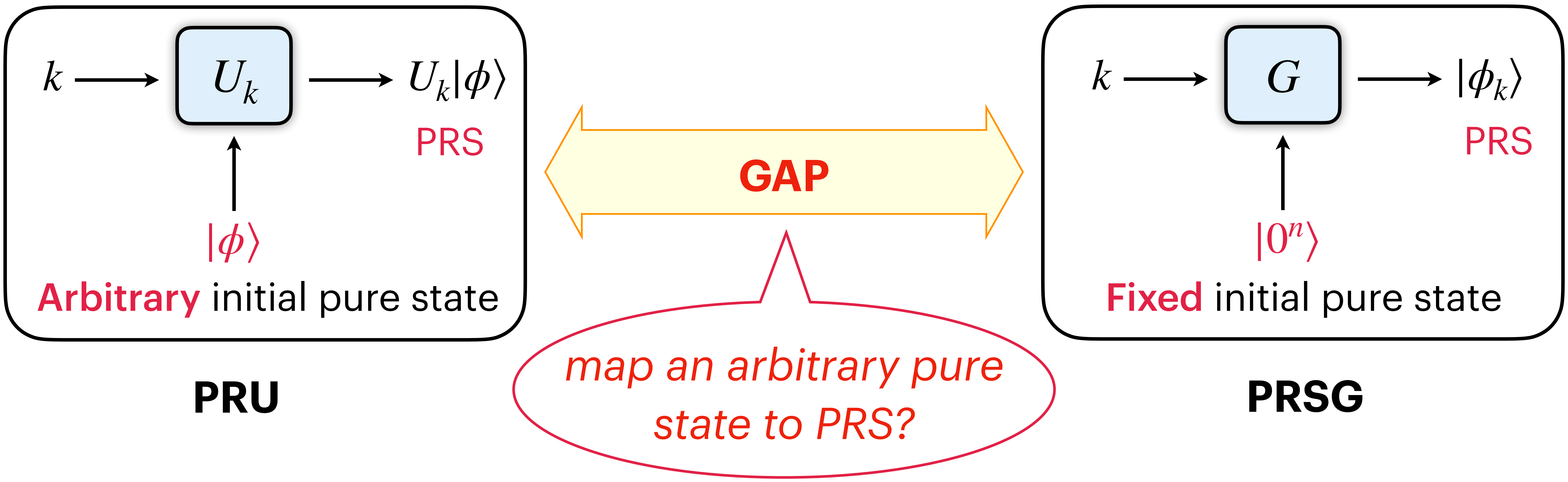


PRSG

Quantum Pseudorandomness

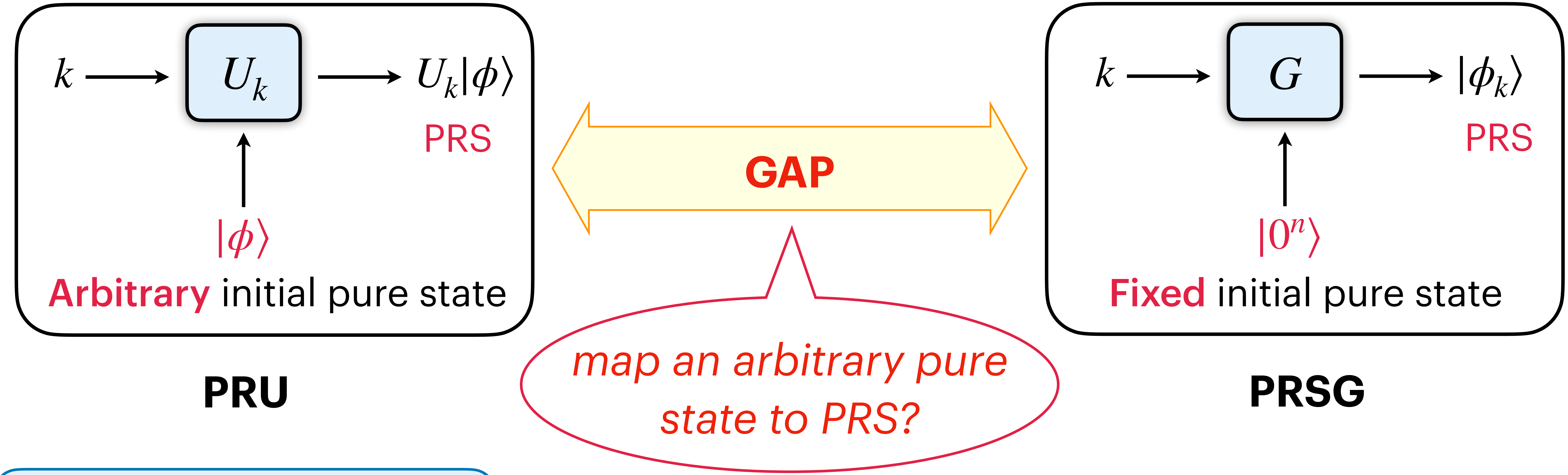


Quantum Pseudorandomness



Pseudorandom State Scramblers

Quantum Pseudorandomness



PRU

PRSG

PRUs exist assuming the existence of any quantum-secure OWF. [MH24]

Pseudorandom State Scramblers

Our Contributions

Our Contributions

⇨ **Defining Quantum Pseudorandom State Scramblers (PRSS)**

- capture the property of scrambling an arbitrary pure state

Our Contributions

⇨ **Defining Quantum Pseudorandom State Scramblers (PRSS)**

- capture the property of scrambling an arbitrary pure state

⇨ **Construction of PRSS from QPRP and QPRF**

- A new random walk: the parallel Kac's walk
- the parallel Kac's walk converges in poly(n) time, an exponential speed-up

Our Contributions

⇨ **Defining Quantum Pseudorandom State Scramblers (PRSS)**

- capture the property of scrambling an arbitrary pure state

⇨ **Construction of PRSS from QPRP and QPRF**

- A new random walk: the parallel Kac's walk
- the parallel Kac's walk converges in poly(n) time, an exponential speed-up

⇨ **A Dispersing Property**

- the output states of our scrambler form an ε-net on the sphere

Our Contributions

⇨ **Defining Quantum Pseudorandom State Scramblers (PRSS)**

- capture the property of scrambling an arbitrary pure state

⇨ **Construction of PRSS from QPRP and QPRF**

- A new random walk: the parallel Kac's walk
- the parallel Kac's walk converges in $\mathrm{poly}(n)$ time, an exponential speed-up

⇨ **A Dispersing Property**

- the output states of our scrambler form an ε-net on the sphere

⇨ **Applications**

- compact quantum encryption
- succinct quantum state commitment

Pseudorandom State Scrambler (PRSS)

Pseudorandom State Scrambler (PRSS)

Let $\mathscr{H}$ be a Hilbert spaces of dimension 2^n and λ be a security parameter. A family of unitary operators $\left\{R_k : \mathscr{H} \to \mathscr{H}\right\}_{k \in \mathscr{K}}$ is a PRSS, if

1. **Bounded key length**: $\log \left|\mathscr{K}\right| = \mathrm{poly}(n, \lambda)$
2. $\exists$ **efficient implementation** of R_k
3. **Comp. Indist.**: $\forall \, |\phi\rangle \in \mathcal{S}(\mathscr{H})$, $\forall$ poly-time quantum $\mathscr{A}$, $\forall \, t = \mathrm{poly}(\lambda)$

$$\left| \Pr_{k \leftarrow \mathscr{K}} \left[\mathscr{A}\left(\left(R_k|\phi\rangle\right)^{\otimes t}\right) = 1\right] - \Pr_{|\psi\rangle \leftarrow \mu} \left[\mathscr{A}\left(|\psi\rangle^{\otimes t}\right) = 1\right] \right| = \mathrm{negl}(\lambda)$$

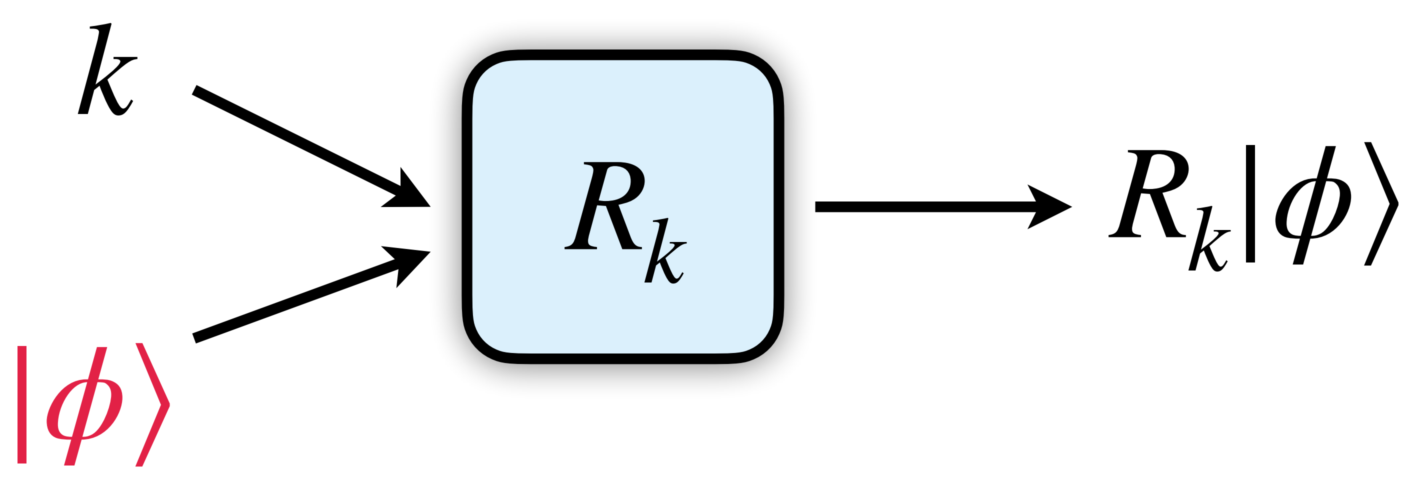
Pseudorandom State Scrambler (PRSS)

Let $\mathscr{H}$ be a Hilbert spaces of dimension 2^n and λ be a security parameter. A family of unitary operators $\left\{R_k : \mathscr{H} \to \mathscr{H}\right\}_{k \in \mathscr{K}}$ is a PRSS, if

1. **Bounded key length**: $\log \left|\mathscr{K}\right| = \mathrm{poly}(n, \lambda)$
2. $\exists$ **efficient implementation** of R_k
3. **Comp. Indist.**: $\forall\ |\phi\rangle \in \mathcal{S}(\mathscr{H})$, $\forall$ poly-time quantum $\mathscr{A}$, $\forall\ t = \mathrm{poly}(\lambda)$

$$\left| \Pr_{k \leftarrow \mathscr{K}}\left[\mathscr{A}\left(\left(R_k|\phi\rangle\right)^{\otimes t}\right) = 1\right] - \Pr_{|\psi\rangle \leftarrow \mu}\left[\mathscr{A}\left(|\psi\rangle^{\otimes t}\right) = 1\right] \right| = \mathrm{negl}(\lambda)$$

PRSS



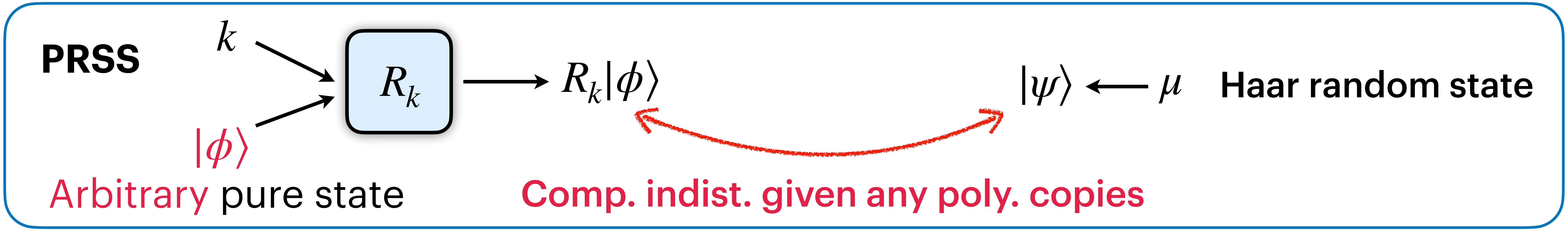
Arbitrary pure state

Pseudorandom State Scrambler (PRSS)

Let $\mathscr{H}$ be a Hilbert spaces of dimension 2^n and λ be a security parameter. A family of unitary operators $\left\{R_k : \mathscr{H} \to \mathscr{H}\right\}_{k \in \mathscr{K}}$ is a PRSS, if

1. **Bounded key length**: $\log \left| \mathscr{K} \right| = \mathrm{poly}(n, \lambda)$
2. $\exists$ **efficient implementation** of R_k
3. **Comp. Indist.**: $\forall\ |\phi\rangle \in \mathcal{S}(\mathscr{H})$, $\forall$ poly-time quantum $\mathscr{A}$, $\forall\ t = \mathrm{poly}(\lambda)$

$$\left| \Pr_{k \leftarrow \mathscr{K}} \left[\mathscr{A}\Big(\big(R_k|\phi\rangle\big)^{\otimes t} \Big) = 1 \right] - \Pr_{|\psi\rangle \leftarrow \mu} \left[\mathscr{A}\Big(|\psi\rangle^{\otimes t} \Big) = 1 \right] \right| = \mathrm{negl}(\lambda)$$

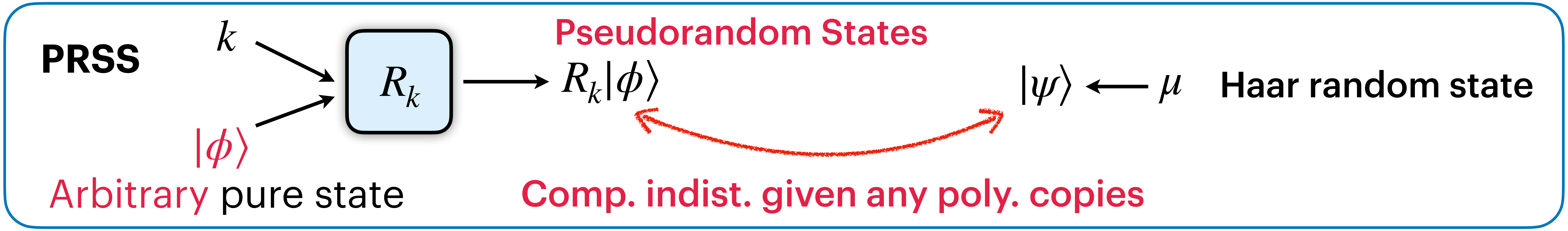


Pseudorandom State Scrambler (PRSS)

Let $\mathscr{H}$ be a Hilbert spaces of dimension 2^n and λ be a security parameter. A family of unitary operators $\left\{R_k : \mathscr{H} \to \mathscr{H}\right\}_{k \in \mathscr{K}}$ is a PRSS, if

1. **Bounded key length**: $\log \left| \mathscr{K} \right| = \mathrm{poly}(n, \lambda)$
2. $\exists$ **efficient implementation** of R_k
3. **Comp. Indist.**: $\forall\ |\phi\rangle \in \mathcal{S}(\mathscr{H})$, $\forall$ poly-time quantum $\mathscr{A}$, $\forall\ t = \mathrm{poly}(\lambda)$

$$\left| \Pr_{k \leftarrow \mathscr{K}} \left[\mathscr{A}\left(\left(R_k |\phi\rangle\right)^{\otimes t} \right) = 1 \right] - \Pr_{|\psi\rangle \leftarrow \mu} \left[\mathscr{A}\left(|\psi\rangle^{\otimes t} \right) = 1 \right] \right| = \mathrm{negl}(\lambda)$$



Pseudorandom State Scrambler (PRSS)

PRSS

k, $|\phi\rangle$ $\rightarrow$ $\boxed{R_k}$ $\longrightarrow$ $R_k|\phi\rangle$

Arbitrary pure state

Pseudorandom States

$|\psi\rangle \longleftarrow \mu$ Haar random state

Comp. indist. given any poly. copies

Pseudorandom State Scrambler (PRSS)

PRSS

k

$|\phi\rangle$

Arbitrary pure state

R_k

$$R_k|\phi\rangle \qquad\qquad |\psi\rangle \longleftarrow \mu$$

Pseudorandom States

Haar random state

Comp. indist. given any poly. copies

Random State Scrambler (RSS)

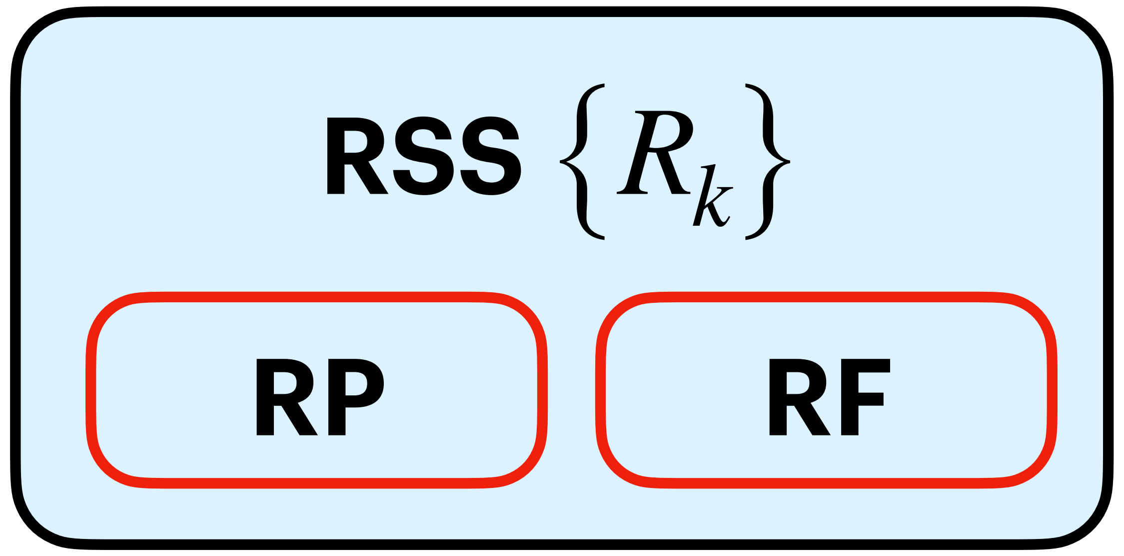
A family of unitary operators $\left\{R_k : \mathscr{H} \to \mathscr{H}\right\}_{k\in\mathscr{K}}$ is an RSS, if

1. **Stat. Indist.**: $\forall\ |\phi\rangle \in \mathcal{S}(\mathscr{H})$, $\forall\ t = \mathrm{poly}(\lambda)$

$$\left\| \mathop{\mathbb{E}}_{k\leftarrow\mathscr{K}}\left[\left(R_k|\phi\rangle\langle\phi|R_k^\dagger\right)^{\otimes t}\right] - \mathop{\mathbb{E}}_{|\psi\rangle\leftarrow\mu}\left[\left(|\psi\rangle\langle\psi|\right)^{\otimes t}\right] \right\|_1 = \mathrm{negl}(\lambda)$$

Constructing PRSS

Constructing PRSS



Constructing PRSS

RSS $\{R_k\}$

RP

RF

$$\left\{R_k|\phi\rangle\right\}$$

Stat. indist.

Haar random state

Constructing PRSS

RSS $\{R_k\}$

RP

RF

PRSS $\{R'_k\}$

QPRP

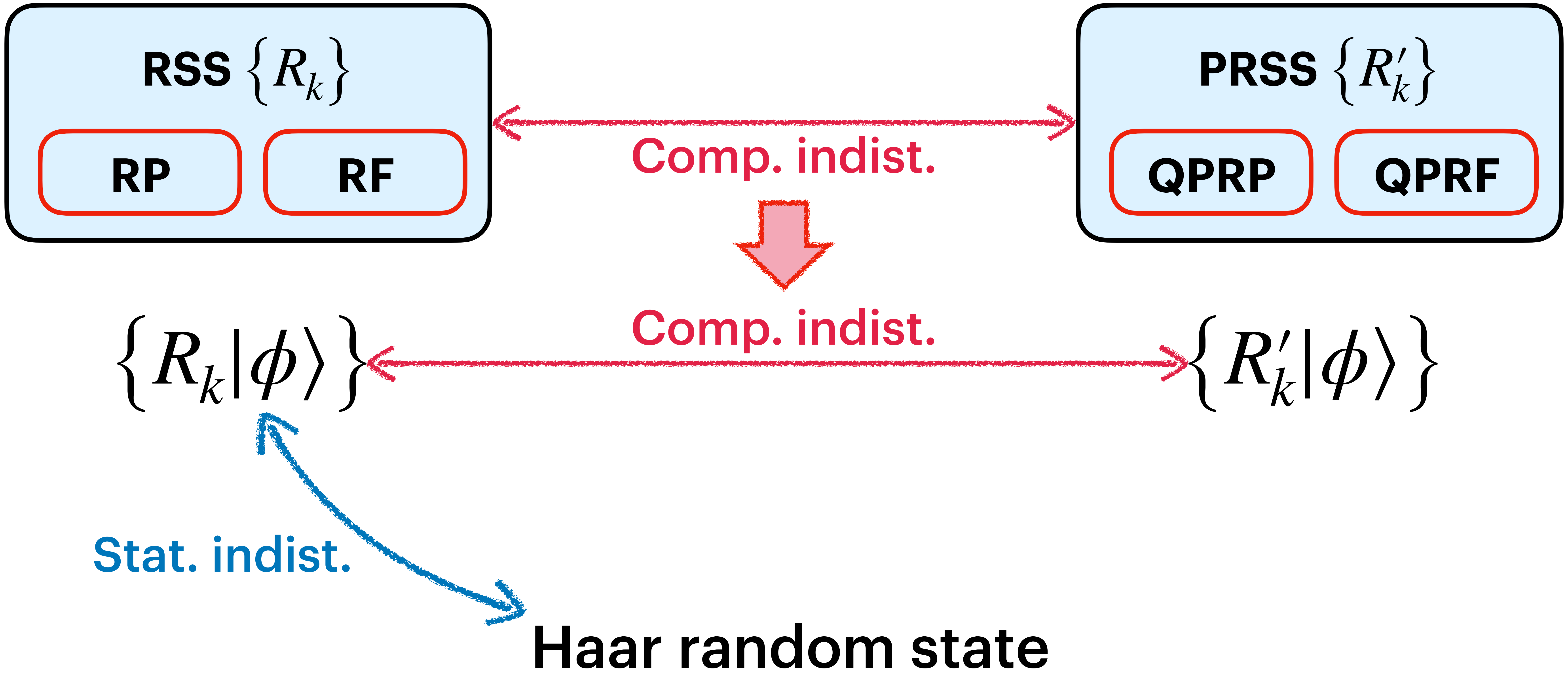
QPRF

$$\{R_k|\phi\rangle\}$$

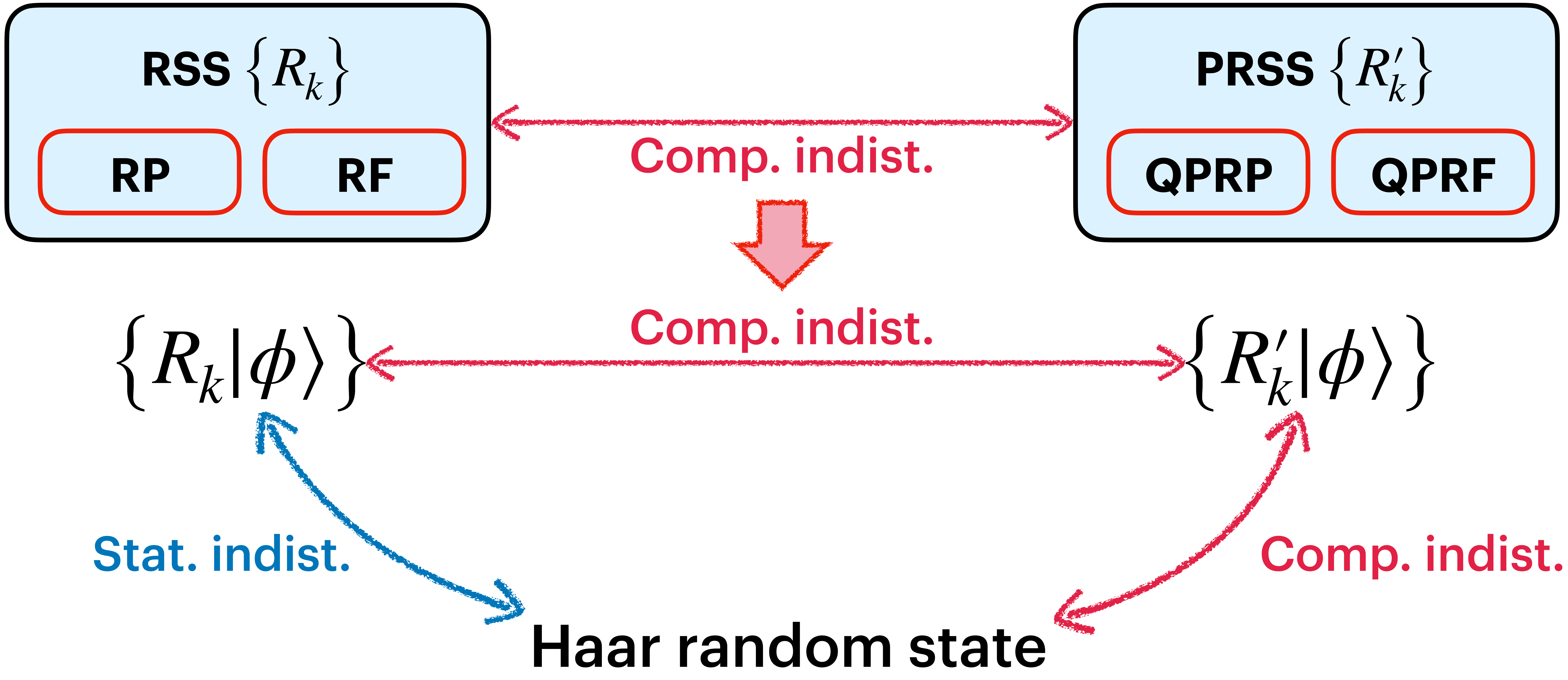
Stat. indist.

Haar random state

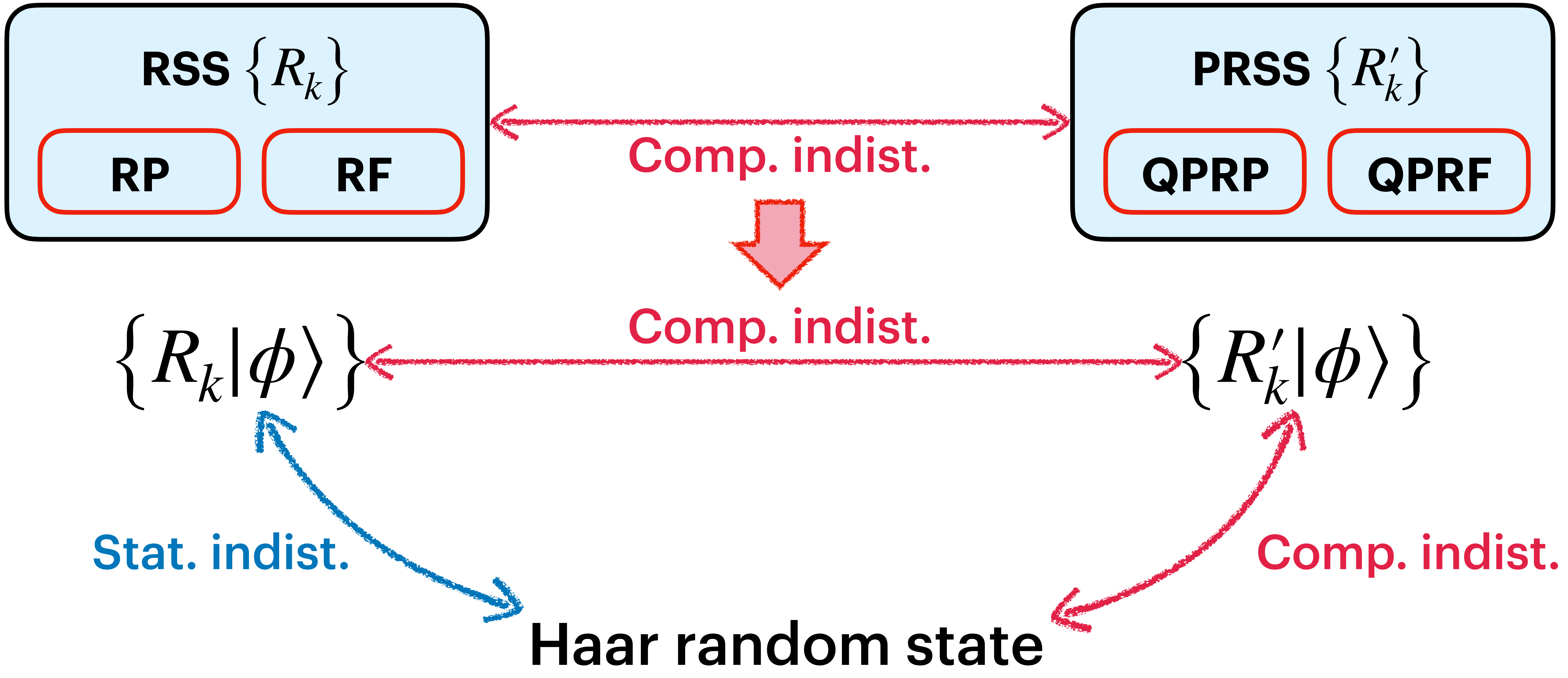
Constructing PRSS



Constructing PRSS



Constructing PRSS



OUR APPROACH:

1. a novel **random walk** yields a state that is **stat. indist.** from a Haar random state.
2. an efficient quantum circuit simulates one step of the random walk.

Constructing RSS $(\mathscr{H} = \mathbb{R}^{2^n})$

Scrambling a quantum state

Constructing RSS ($\mathscr{H} = \mathbb{R}^{2^n}$)

Scrambling a quantum state

Kac's Walk (on $x \in \mathcal{S}(\mathbb{R}^{2^n})$)

Constructing RSS ($\mathscr{H} = \mathbb{R}^{2^n}$)

Scrambling a quantum state

Kac's Walk (on $x \in \mathcal{S}(\mathbb{R}^{2^n})$)

$$(i, j) \quad \begin{bmatrix} x_1 \\ \vdots \\ x_i \\ \vdots \\ x_j \\ \vdots \\ x_{2^n} \end{bmatrix}$$

Constructing RSS ($\mathscr{H} = \mathbb{R}^{2^n}$)

Scrambling a quantum state

Kac's Walk (on $x \in \mathcal{S}(\mathbb{R}^{2^n})$)

$$(i,j) \quad \begin{bmatrix} x_1 \\ \vdots \\ x_i \\ \vdots \\ x_j \\ \vdots \\ x_{2^n} \end{bmatrix}$$

$\theta \in [0,2\pi)$

Constructing RSS ($\mathscr{H} = \mathbb{R}^{2^n}$)

Scrambling a quantum state

Kac's Walk (on $x \in \mathcal{S}(\mathbb{R}^{2^n})$)

$$(i,j) \quad \begin{bmatrix} x_1 \\ \vdots \\ x_i \\ \vdots \\ x_j \\ \vdots \\ x_{2^n} \end{bmatrix} \xrightarrow{\theta \in [0,2\pi)} \begin{bmatrix} x_1 \\ \vdots \\ x_i' \\ \vdots \\ x_j' \\ \vdots \\ x_{2^n} \end{bmatrix}$$

$$\begin{pmatrix} x_i' \\ x_j' \end{pmatrix} \longleftarrow \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \begin{pmatrix} x_i \\ x_j \end{pmatrix}$$

Constructing RSS ($\mathscr{H} = \mathbb{R}^{2^n}$)

Scrambling a quantum state

Kac's Walk (on $x \in \mathcal{S}(\mathbb{R}^{2^n})$)

$$(i,j) \quad \begin{bmatrix}x_1\\ \vdots\\ x_i\\ \vdots\\ x_j\\ \vdots\\ x_{2^n}\end{bmatrix} \xrightarrow{\theta \in [0,2\pi)} \begin{bmatrix}x_1\\ \vdots\\ x_i'\\ \vdots\\ x_j'\\ \vdots\\ x_{2^n}\end{bmatrix}$$

$$\begin{pmatrix}x_i'\\ x_j'\end{pmatrix} \longleftarrow \begin{bmatrix}\cos\theta & -\sin\theta\\ \sin\theta & \cos\theta\end{bmatrix}\begin{pmatrix}x_i\\ x_j\end{pmatrix}$$

- A model for Boltzmann gas [Kac56].

Constructing RSS ($\mathscr{H} = \mathbb{R}^{2^n}$)

Scrambling a quantum state

Kac's Walk (on $x \in \mathcal{S}(\mathbb{R}^{2^n})$)

$$(i,j) \quad \begin{bmatrix} x_1 \\ \vdots \\ x_i \\ \vdots \\ x_j \\ \vdots \\ x_{2^n} \end{bmatrix} \xrightarrow[\begin{pmatrix} x_i' \\ x_j' \end{pmatrix} \longleftarrow \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \begin{pmatrix} x_i \\ x_j \end{pmatrix}]{\theta \in [0, 2\pi)} \begin{bmatrix} x_1 \\ \vdots \\ x_i' \\ \vdots \\ x_j' \\ \vdots \\ x_{2^n} \end{bmatrix}$$

- A model for Boltzmann gas [Kac56].
- Converges in $O(n2^n)$ steps [PS17]

Constructing RSS $(\mathscr{H} = \mathbb{R}^{2^n})$

Scrambling a quantum state

Kac's Walk (on $x \in \mathcal{S}(\mathbb{R}^{2^n})$)

$$(i,j) \quad \begin{bmatrix} x_1 \\ \vdots \\ x_i \\ \vdots \\ x_j \\ \vdots \\ x_{2^n} \end{bmatrix} \xrightarrow{\theta \in [0,2\pi)} \begin{bmatrix} x_1 \\ \vdots \\ x_i' \\ \vdots \\ x_j' \\ \vdots \\ x_{2^n} \end{bmatrix}$$

$$\begin{pmatrix} x_i' \\ x_j' \end{pmatrix} \longleftarrow \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \begin{pmatrix} x_i \\ x_j \end{pmatrix}$$

- A model for Boltzmann gas [Kac56].
- Converges in $O(n2^n)$ steps [PS17]
- Kac's walk on $\mathrm{SO}(N)$:
  - Spectral gap [DSC00, Jan03, CCL03]
  - L^2 W-distance: $O(N^2 \log N)$ [Oli09]
  - TV distance: $O(N^2) \sim O(N^4 \log N)$ [PS18]

Constructing RSS ($\mathscr{H} = \mathbb{R}^{2^n}$)

Scrambling a quantum state

Our construction: Parallel Kac's Walk

1. Select a random perfect matching of $\{1,\cdots,2^n\}$

$$P = \left\{(i_1,j_1), \ldots, (i_{2^{n-1}},j_{2^{n-1}})\right\}$$

2. For each pair $(i_k,j_k) \in P$, sample an angle

$$\theta_k \in [0,2\pi)$$

and set

$$\begin{pmatrix} x_{i_k} \\ x_{j_k} \end{pmatrix} \longleftarrow \begin{bmatrix} \cos\theta_k & -\sin\theta_k \\ \sin\theta_k & \cos\theta_k \end{bmatrix} \begin{pmatrix} x_{i_k} \\ x_{j_k} \end{pmatrix}$$

Constructing RSS ($\mathscr{H} = \mathbb{R}^{2^n}$)

Scrambling a quantum state

Our construction: Parallel Kac's Walk

1. Select a random perfect matching of $\{1,\cdots,2^n\}$
$$P = \left\{(i_1,j_1),\ldots,(i_{2^{n-1}},j_{2^{n-1}})\right\}$$
2. For each pair $(i_k,j_k) \in P$, sample an angle
$$\theta_k \in [0,2\pi)$$
and set
$$\begin{pmatrix} x_{i_k} \\ x_{j_k} \end{pmatrix} \longleftarrow \begin{bmatrix} \cos\theta_k & -\sin\theta_k \\ \sin\theta_k & \cos\theta_k \end{bmatrix} \begin{pmatrix} x_{i_k} \\ x_{j_k} \end{pmatrix}$$

One step of the parallel Kac's walk

$\approx$

2^{n-1} steps of the original Kac's walk

Constructing RSS ($\mathscr{H} = \mathbb{R}^{2^n}$)

Scrambling a quantum state

Our construction: Parallel Kac's Walk

1. Select a random perfect matching of $\{1,\cdots,2^n\}$

$$P = \left\{(i_1,j_1),\ldots,(i_{2^{n-1}},j_{2^{n-1}})\right\}$$

2. For each pair $(i_k,j_k) \in P$, sample an angle

$$\theta_k \in [0,2\pi)$$

and set

$$\begin{pmatrix} x_{i_k} \\ x_{j_k} \end{pmatrix} \longleftarrow \begin{bmatrix} \cos\theta_k & -\sin\theta_k \\ \sin\theta_k & \cos\theta_k \end{bmatrix} \begin{pmatrix} x_{i_k} \\ x_{j_k} \end{pmatrix}$$

One step of the parallel Kac's walk

$\mathrel{\rotatebox{90}{$\approx$}}$

2^{n-1} steps of the original Kac's walk

Theorem

Let $\{|\phi_t\rangle \in \mathbb{R}^{2^n}\}_{t\geq 0}$ be a parallel Kac's walk. For $T = 10(\lambda+1)n$,

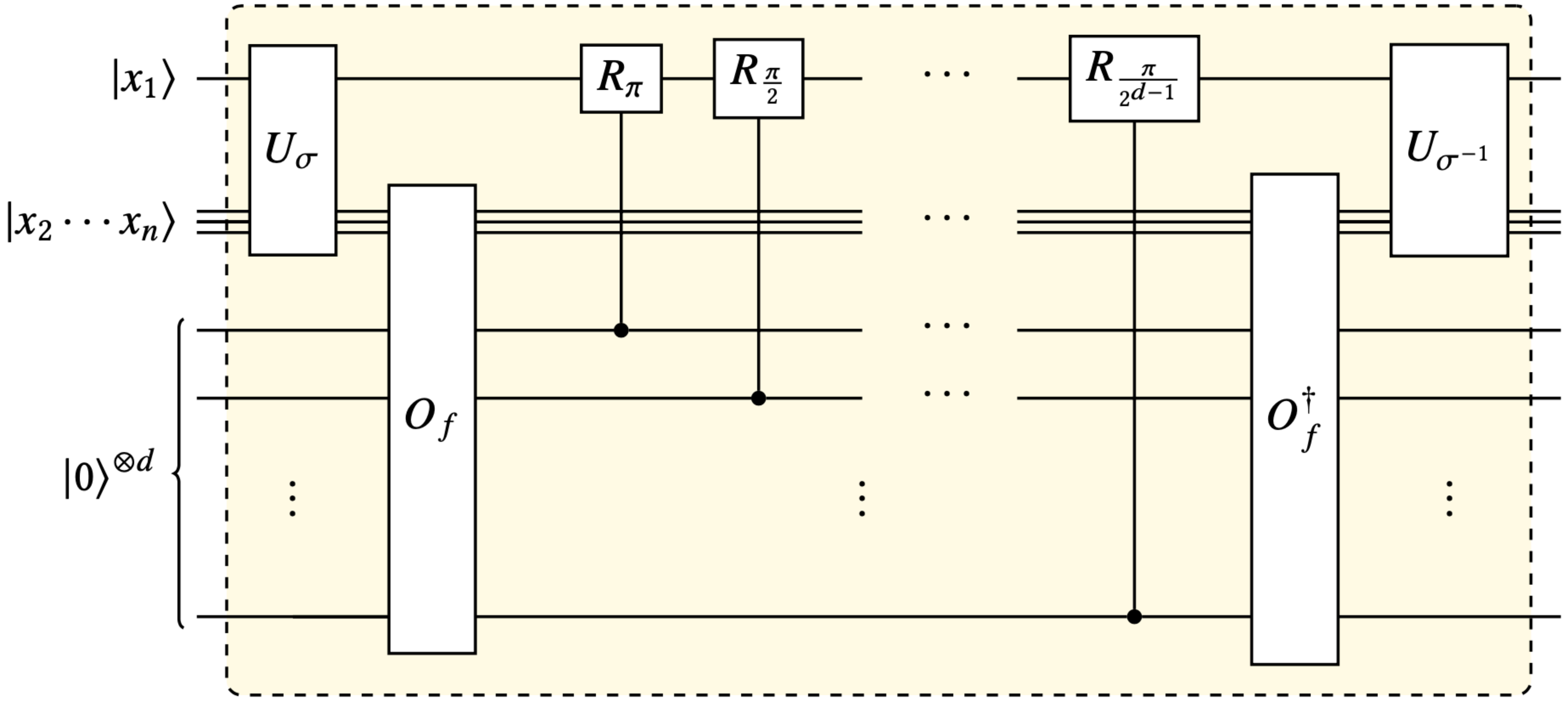
$$\left(|\phi_T\rangle\langle\phi_T|\right)^{\otimes l} \approx_s \left(|\psi\rangle\langle\psi|\right)^{\otimes l}$$

Constructing RSS ($\mathscr{H} = \mathbb{R}^{2^n}$)

Implementing via quantum circuit

$K_{\sigma,f}$ simulates one step of Kac's walk.

$\sigma : \{0,1\}^n \to \{0,1\}^n \quad f : \{0,1\}^{n-1} \to \{0,1\}^d$



Constructing RSS ($\mathscr{H} = \mathbb{R}^{2^n}$)

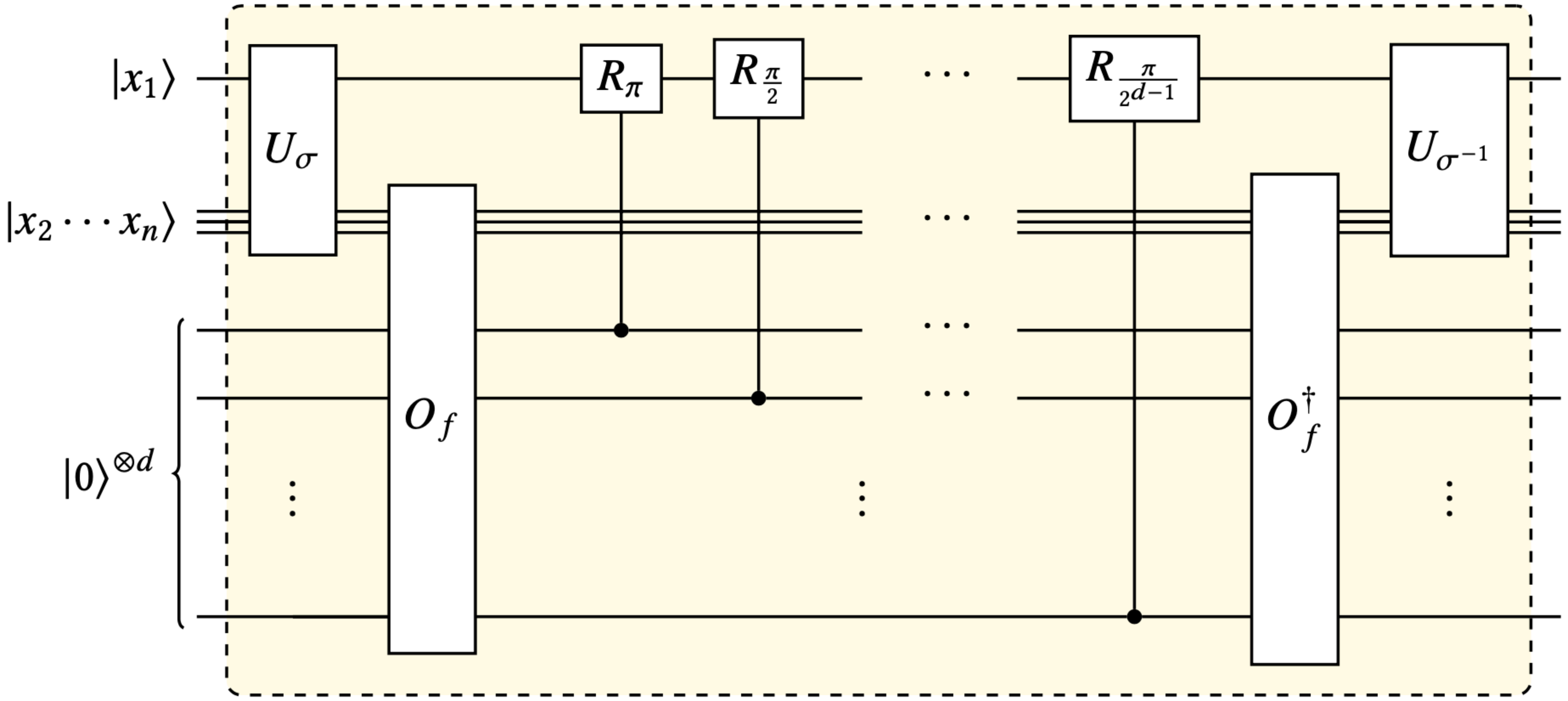
Implementing via quantum circuit

$K_{\sigma,f}$ simulates one step of Kac's walk.

$\sigma : \{0,1\}^n \to \{0,1\}^n \quad f : \{0,1\}^{n-1} \to \{0,1\}^d$

σ: partition the basis into 2^{n-1} pairs

f: choose an independently random angle



Constructing RSS ($\mathscr{H} = \mathbb{R}^{2^n}$)

Implementing via quantum circuit

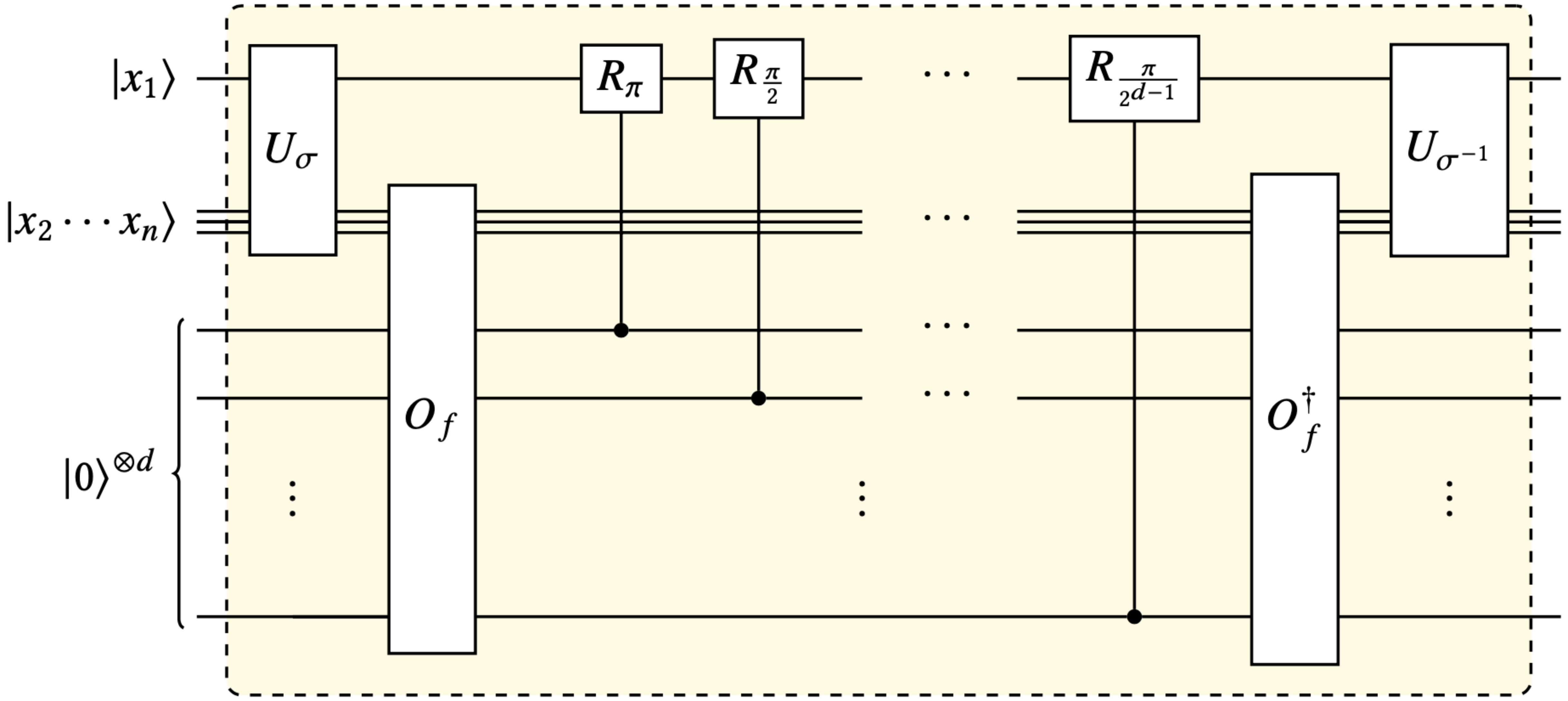
$K_{\sigma,f}$ simulates one step of Kac's walk.

$\sigma : \{0,1\}^n \to \{0,1\}^n \quad f : \{0,1\}^{n-1} \to \{0,1\}^d$

σ: partition the basis into 2^{n-1} pairs

f: choose an independently random angle

$$C_{\sigma_1,\ldots,\sigma_T,f_1,\ldots,f_T} = K_{\sigma_T,f_T} \cdots K_{\sigma_2,f_2} K_{\sigma_1,f_1}$$



Constructing RSS ($\mathscr{H} = \mathbb{R}^{2^n}$)

Implementing via quantum circuit

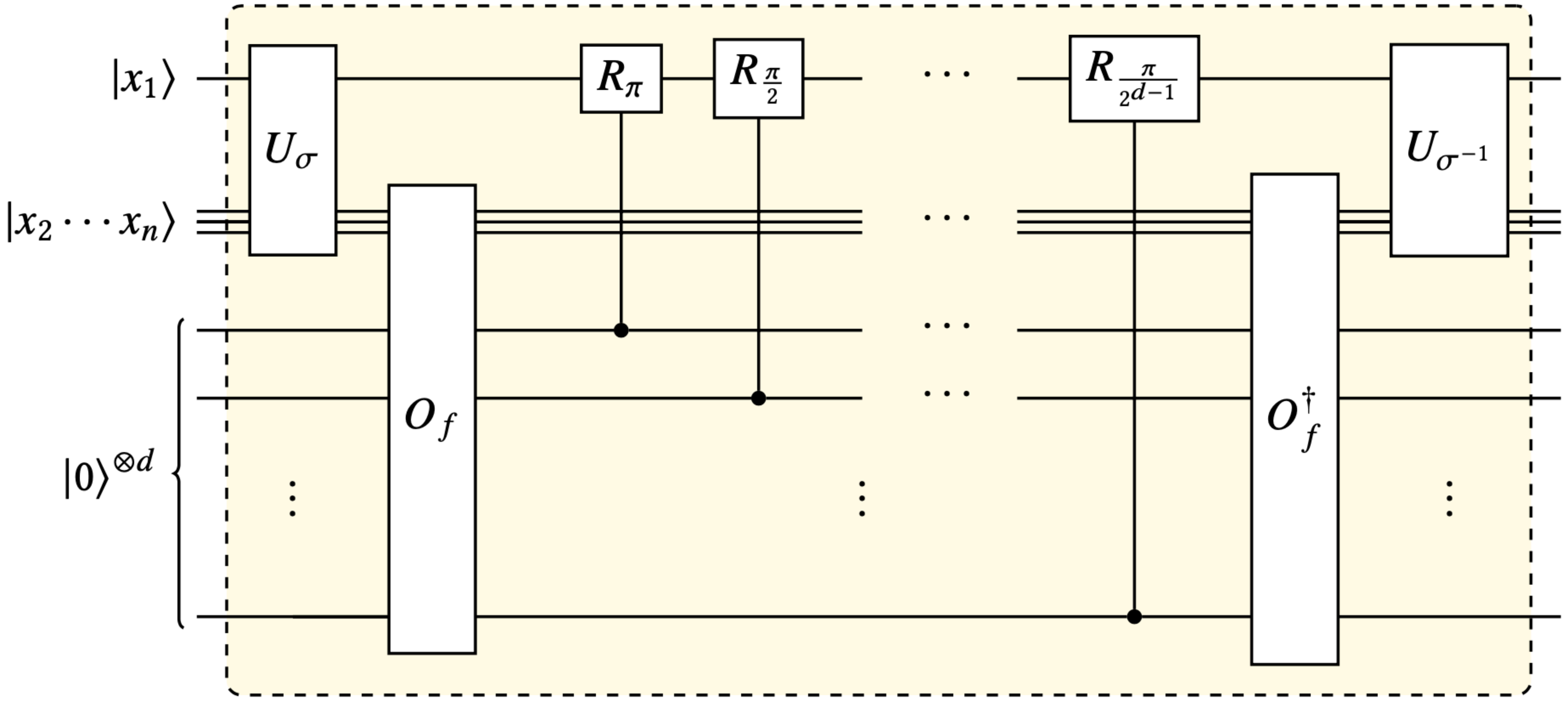
$K_{\sigma,f}$ simulates one step of Kac's walk.

$\sigma : \{0,1\}^n \to \{0,1\}^n \quad f : \{0,1\}^{n-1} \to \{0,1\}^d$

σ: partition the basis into 2^{n-1} pairs

f: choose an independently random angle

$$C_{\sigma_1,\ldots,\sigma_T,f_1,\ldots,f_T} = K_{\sigma_T,f_T} \cdots K_{\sigma_2,f_2} K_{\sigma_1,f_1}$$

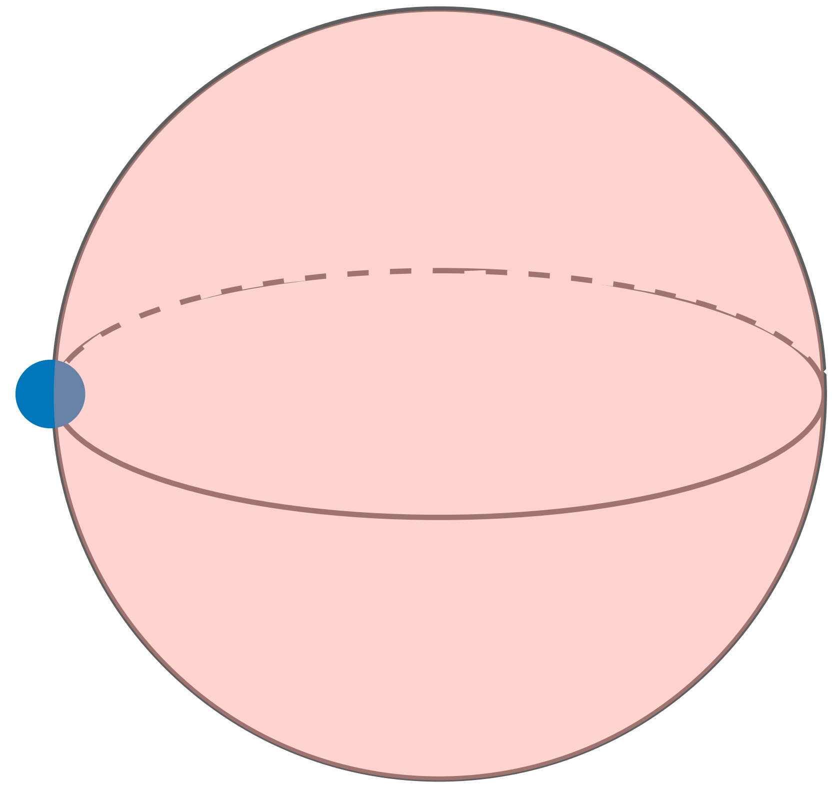


Theorem

Let $d = \log^2 \lambda + \log^2 n$ and $T = 10(\lambda + 1)n$. $\left\{ C_{\sigma_1,\ldots,\sigma_T,f_1,\ldots,f_T} \right\}$ is an RSS.

A Dispersing Property

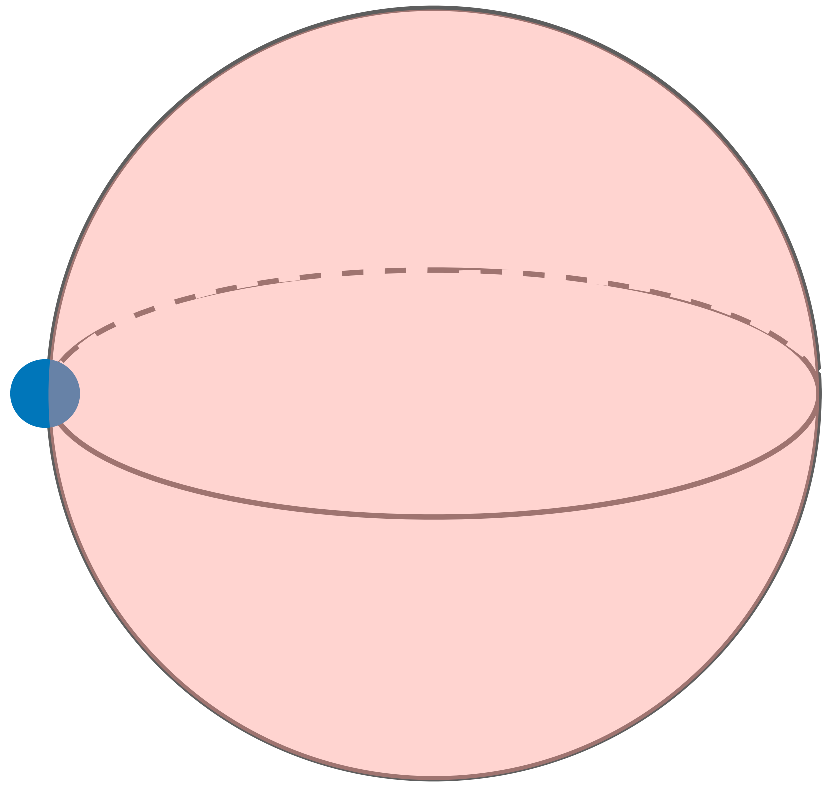
A Dispersing Property



Parallel Kac's walk

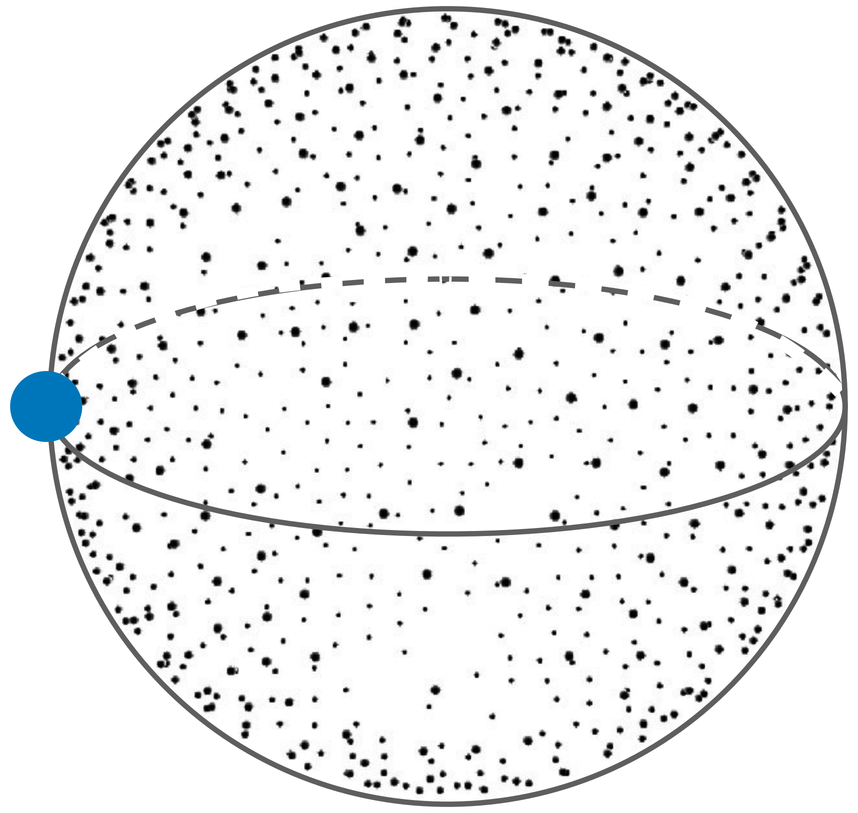
Close to Haar in total variation distance

A Dispersing Property



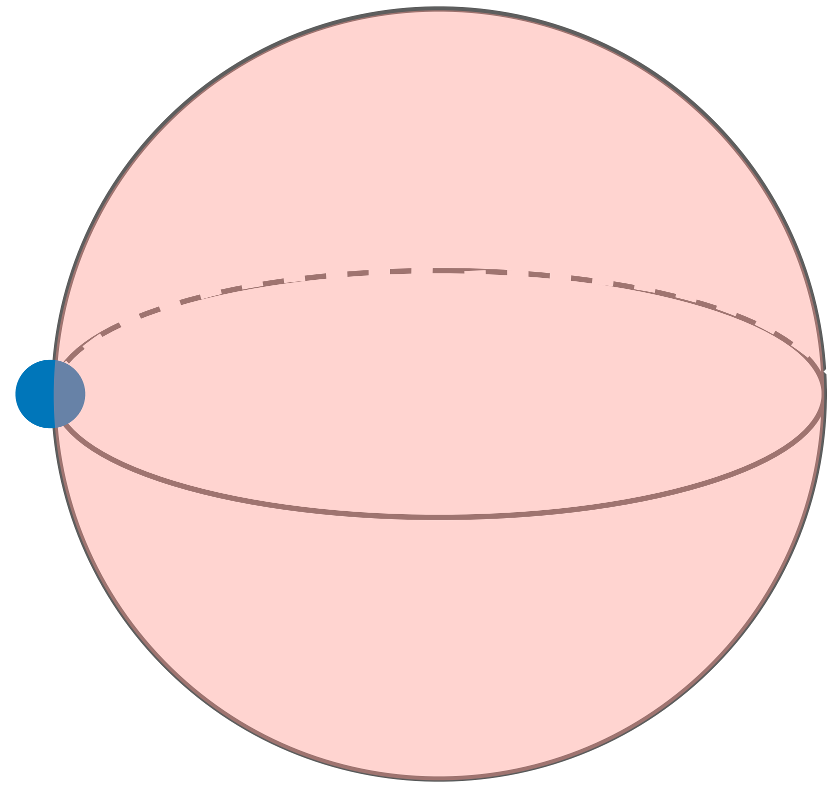
Parallel Kac's walk

Close to Haar in total variation distance



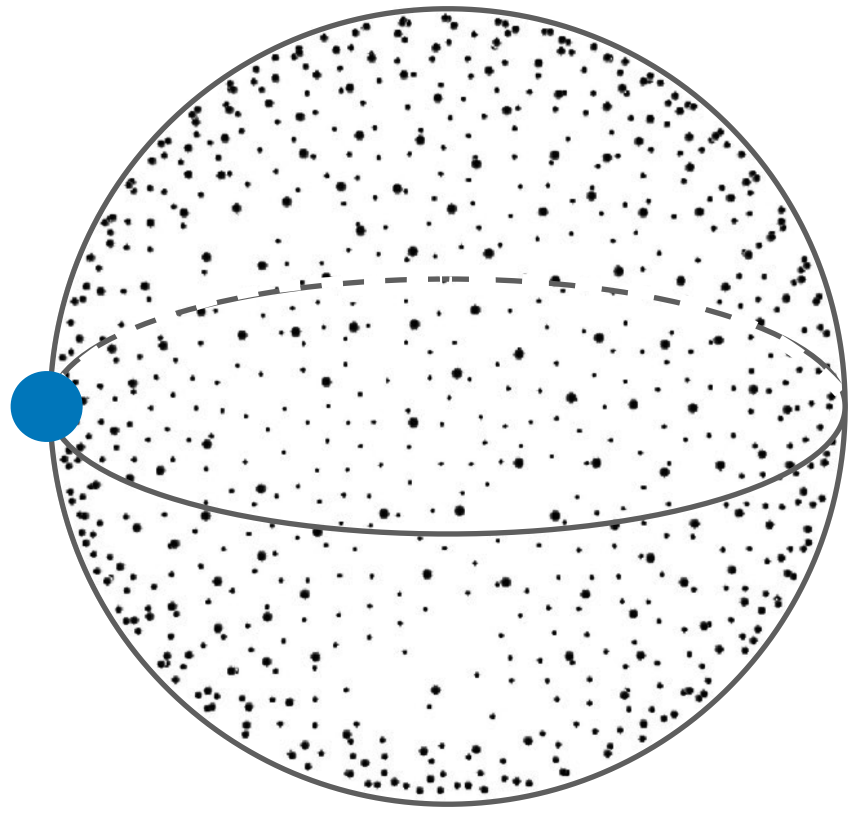
Our RSS

A Dispersing Property



Parallel Kac's walk

Close to Haar in total variation distance

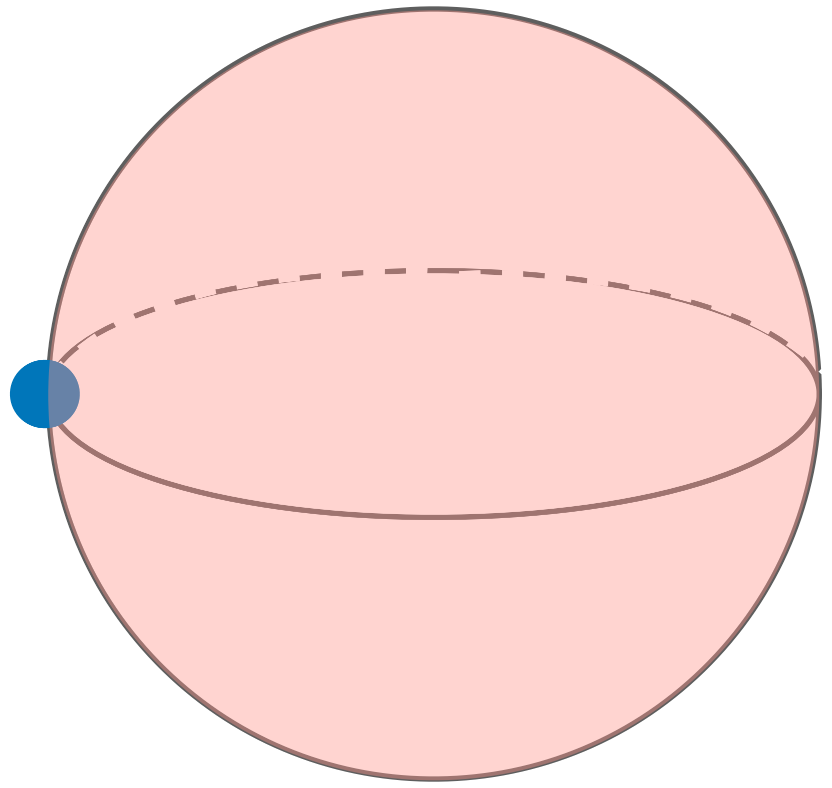


Our RSS

Output states span an ε-net of the state space

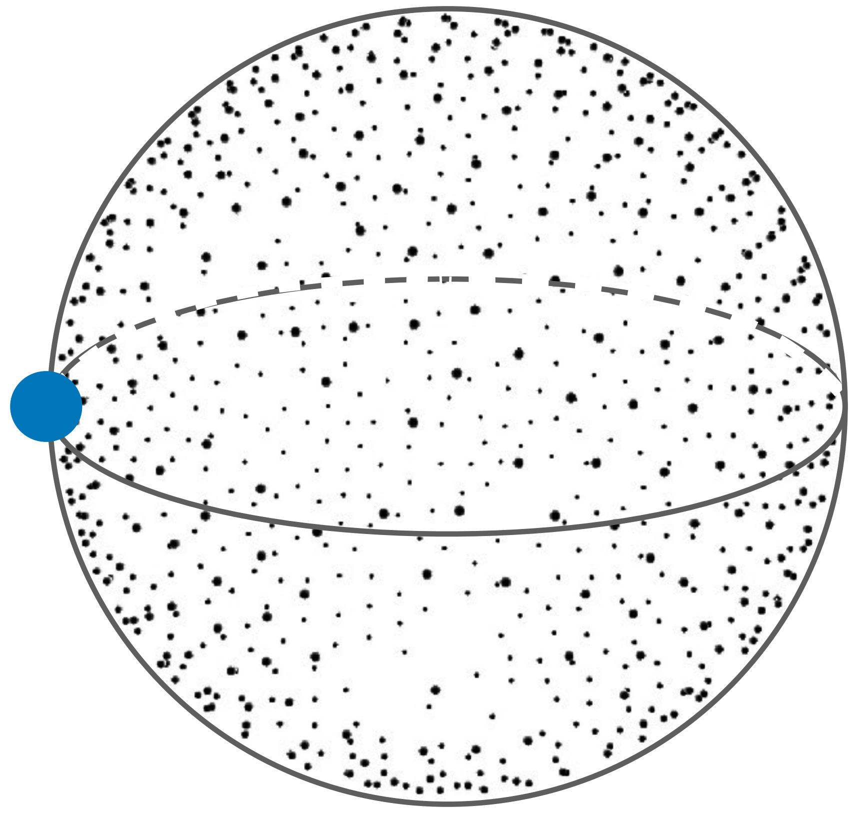
Close to Haar in Wasserstein distance

A Dispersing Property



Parallel Kac's walk

Close to Haar in total variation distance



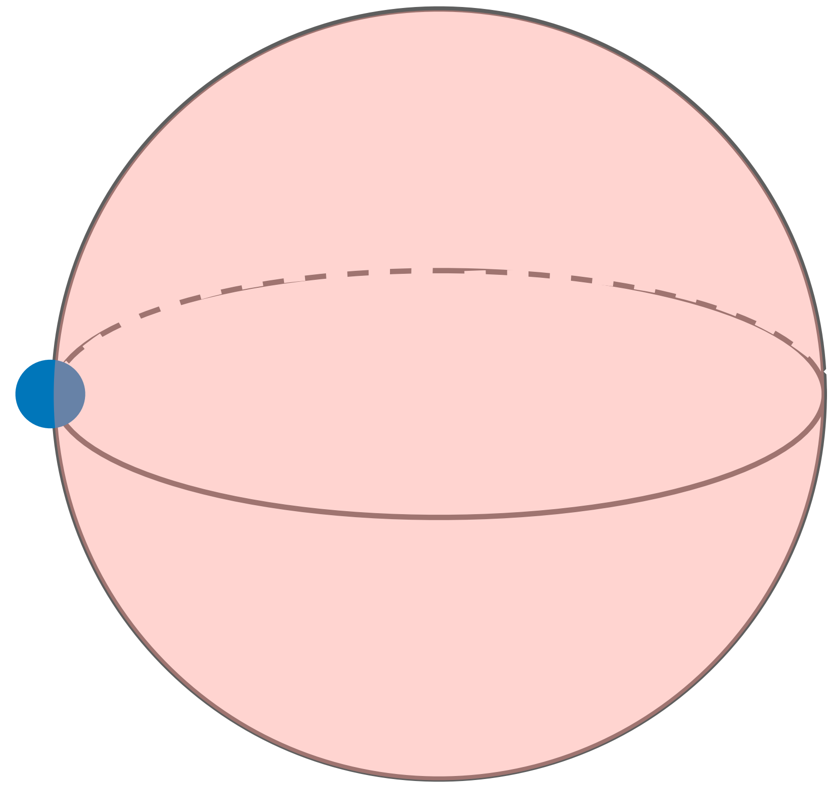
Our RSS

Output states span an ε-net of the state space

Close to Haar in Wasserstein distance

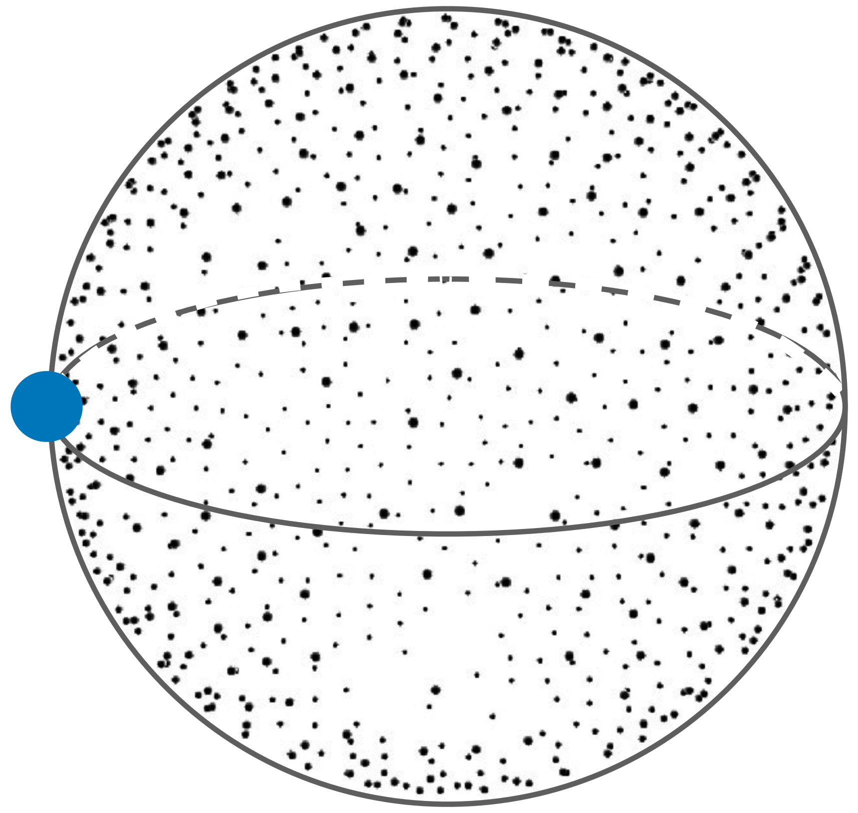
Not a necessary property by definition, even for PRU

A Dispersing Property



Parallel Kac's walk

Close to Haar in total variation distance

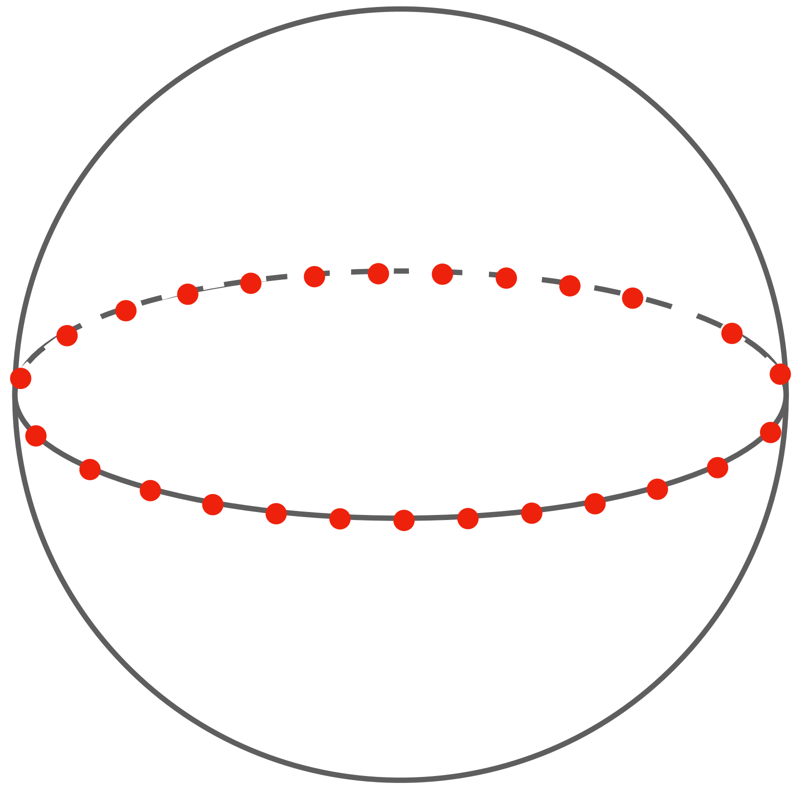


Our RSS

Output states span an ε-net of the state space

Close to Haar in Wasserstein distance

Not a necessary property by definition, even for PRU



Random phase states

$$|\phi_k\rangle = \frac{1}{\sqrt{N}} \sum_{x \in \{0,1\}^n} \omega_N^{f_k(x)} |x\rangle$$

$$|\phi_k\rangle = \frac{1}{\sqrt{N}} \sum_{x \in \{0,1\}^n} (-1)^{f_k(x)} |x\rangle$$

Close to Haar in average

Final Remarks and Open Questions

$U_k \approx$ Haar random unitary

Able to scramble an arbitrary pure state

PRU $\{U_k\}$

Scramble an arbitrary pure state

$$\left(R_k|\phi\rangle\right)^{\otimes l(n)} \approx |\psi\rangle^{\otimes l(n)}$$

Haar random state

PRSS $\{R_k\}$

Scramble an fixed initial state, e.g. $|0^n\rangle$

$$\left(G(1^n, k)|0^n\rangle\right)^{\otimes l(n)} \approx |\psi\rangle^{\otimes l(n)}$$

Haar random state

PRSG G

Final Remarks and Open Questions

$U_k \approx$ Haar random unitary
Able to scramble an arbitrary pure state

PRU $\{U_k\}$

Q1: Is our scrambler a PRU?

Scramble an arbitrary pure state

$$\left(R_k|\phi\rangle\right)^{\otimes l(n)} \approx \underbrace{|\psi\rangle^{\otimes l(n)}}_{\text{Haar random state}}$$

PRSS $\{R_k\}$

Scramble an fixed initial state, e.g. $|0^n\rangle$

$$\left(G(1^n,k)|0^n\rangle\right)^{\otimes l(n)} \approx \underbrace{|\psi\rangle^{\otimes l(n)}}_{\text{Haar random state}}$$

PRSG G

Final Remarks and Open Questions

$U_k \approx$ Haar random unitary

Able to scramble an arbitrary pure state

PRU $\{U_k\}$

Q1: Is our scrambler a PRU?

Yes! Using the compressed purification technique in [MH24].

Scramble an arbitrary pure state

$$\left(R_k|\phi\rangle\right)^{\otimes l(n)} \approx |\psi\rangle^{\otimes l(n)}$$

Haar random state

PRSS $\{R_k\}$

Scramble an fixed initial state, e.g. $|0^n\rangle$

$$\left(G(1^n, k)|0^n\rangle\right)^{\otimes l(n)} \approx |\psi\rangle^{\otimes l(n)}$$

Haar random state

PRSG G

Final Remarks and Open Questions

$U_k \approx$ Haar random unitary
Able to scramble an arbitrary pure state

PRU $\{U_k\}$

Q1: Is our scrambler a PRU?

Yes! Using the compressed purification technique in [MH24].

Scramble an arbitrary pure state

$$\big(R_k|\phi\rangle\big)^{\otimes l(n)} \approx \underbrace{|\psi\rangle^{\otimes l(n)}}_{\text{Haar random state}}$$

PRSS $\{R_k\}$

Q2: More applications?

Scramble an fixed initial state, e.g. $|0^n\rangle$

$$\big(G(1^n,k)|0^n\rangle\big)^{\otimes l(n)} \approx \underbrace{|\psi\rangle^{\otimes l(n)}}_{\text{Haar random state}}$$

PRSG G

Final Remarks and Open Questions

$U_k \approx$ Haar random unitary

Able to scramble an arbitrary pure state

PRU $\{U_k\}$

Q1: Is our scrambler a PRU?

Yes! Using the compressed purification technique in [MH24].

Scramble an arbitrary pure state

$$\left(R_k|\phi\rangle\right)^{\otimes l(n)} \approx \underbrace{|\psi\rangle^{\otimes l(n)}}_{\text{Haar random state}}$$

PRSS $\{R_k\}$

Q2: More applications?

Q3: Simplify the construction?

Scramble an fixed initial state, e.g. $|0^n\rangle$

$$\left(G(1^n, k)|0^n\rangle\right)^{\otimes l(n)} \approx \underbrace{|\psi\rangle^{\otimes l(n)}}_{\text{Haar random state}}$$

PRSG G