Rank–dependent complexity in estimating traces of quantum state powers

[2408.00314] Rank Is All You Need: Estimating the Trace of Powers of Density Matrices

Junseo Lee

Collaborators: Myeongjin Shin, Seungwoo Lee, Kabgyun Jeong

Dec 13, 2024 @ SNU QST Seminar

Overview

- Trace of powers & Literature review
- Mathematical intuitions
- Main results: algorithm, lemmas, theorems, corollaries
- Numerical simulations
- Computational hardness of $\mathrm{TsallisQED}_q$ and $\mathrm{TsallisQEA}_q$
- Applications
- Concluding remarks

Overview

- **Trace of powers & Literature review**
- Mathematical intuitions
- Main results: algorithm, lemmas, theorems, corollaries
- Numerical simulations
- Computational hardness of TsallisQED$_q$ and TsallisQEA$_q$
- Applications
- Concluding remarks

Property testing

- **(Property testing)** Let $\mathcal{X}$ be a set of objects and $d : \mathcal{X} \times \mathcal{X} \to [0,1]$ be a distance measure on $\mathcal{X}$.
  - A subset $\mathcal{P} \subseteq \mathcal{X}$ is called a *property*.
  - An object $x \in \mathcal{X}$ is ε-*far* from $\mathcal{P}$ if $d(x, y) \geq \varepsilon$ for all $y \in \mathcal{P}$; x is ε-*close* to $\mathcal{P}$ if there is a $y \in \mathcal{P}$ such that $d(x, y) \leq \varepsilon$.
  - An ε-property tester (sometimes abbreviated to ε-tester) for $\mathcal{P}$ is an algorithm that receives as input either an $x \in \mathcal{P}$ or an x that is ε-far from $\mathcal{P}$.
  - In the former case, the algorithm accepts with probability at least 2/3; in the latter case, the algorithm rejects with probability at least 2/3.

Property testing

- One can view classical property testing algorithms in two ways:
  - either as testing **properties of data** (such as graph isomorphism),
  - or **properties of functions** (such as linearity).

Property testing

- One can view classical property testing algorithms in two ways:
  - either as testing **properties of data** (such as graph isomorphism),
  - or **properties of functions** (such as linearity).

- If one wishes to generalize property testing to the quantum realm, one is thus naturally led to two different generalizations:
  - testing **properties of quantum states**,
  - and **properties of quantum operations**.

Property testing

- **(Quantum testing of quantum properties)** Consider cases where $\mathscr{X}$ is a set of quantum objects and our tester is also quantum.

[1310.2035] A Survey of Quantum Property Testing

Property testing

- **(Quantum testing of quantum properties)** Consider cases where $\mathscr{X}$ is a set of quantum objects and our tester is also quantum.
- For example, testing properties of mixed states is analogous to the classical idea of testing properties of probability distributions.

[1310.2035] A Survey of Quantum Property Testing

Property testing

- **(Quantum testing of quantum properties)** Consider cases where $\mathscr{X}$ is a set of quantum objects and our tester is also quantum.
- For example, testing properties of mixed states is analogous to the classical idea of testing properties of probability distributions.
- Some quantum properties, however, have no simple classical analog (such as properties relating to entanglement).

[1310.2035] A Survey of Quantum Property Testing

Property testing

- **(Quantum testing of quantum properties)** Consider cases where $\mathscr{X}$ is a set of quantum objects and our tester is also quantum.
- For example, testing properties of mixed states is analogous to the classical idea of testing properties of probability distributions.
- Some quantum properties, however, have no simple classical analog (such as properties relating to entanglement).

	Coherent	**Incoherent**
Static	Pure state (§4.1)	Mixed state (§4.2)
Dynamic	Unitary operator (§5.1)	Quantum channel (§5.2)

Table 1: The taxonomy of quantum properties

[1310.2035] A Survey of Quantum Property Testing

Property testing

- **(Quantum testing of quantum properties)** Consider cases where $\mathscr{X}$ is a set of quantum objects and our tester is also quantum.
- For example, testing properties of mixed states is analogous to the classical idea of testing properties of probability distributions.
- Some quantum properties, however, have no simple classical analog (such as properties relating to entanglement).

	Coherent	Incoherent
Static	Pure state (§4.1)	Mixed state (§4.2)
Dynamic	Unitary operator (§5.1)	Quantum channel (§5.2)

Table 1: The taxonomy of quantum properties

Equality, Separability

Unitarily invariant properties

(Purity, Mixedness, …)

[1310.2035] A Survey of Quantum Property Testing

Trace of powers

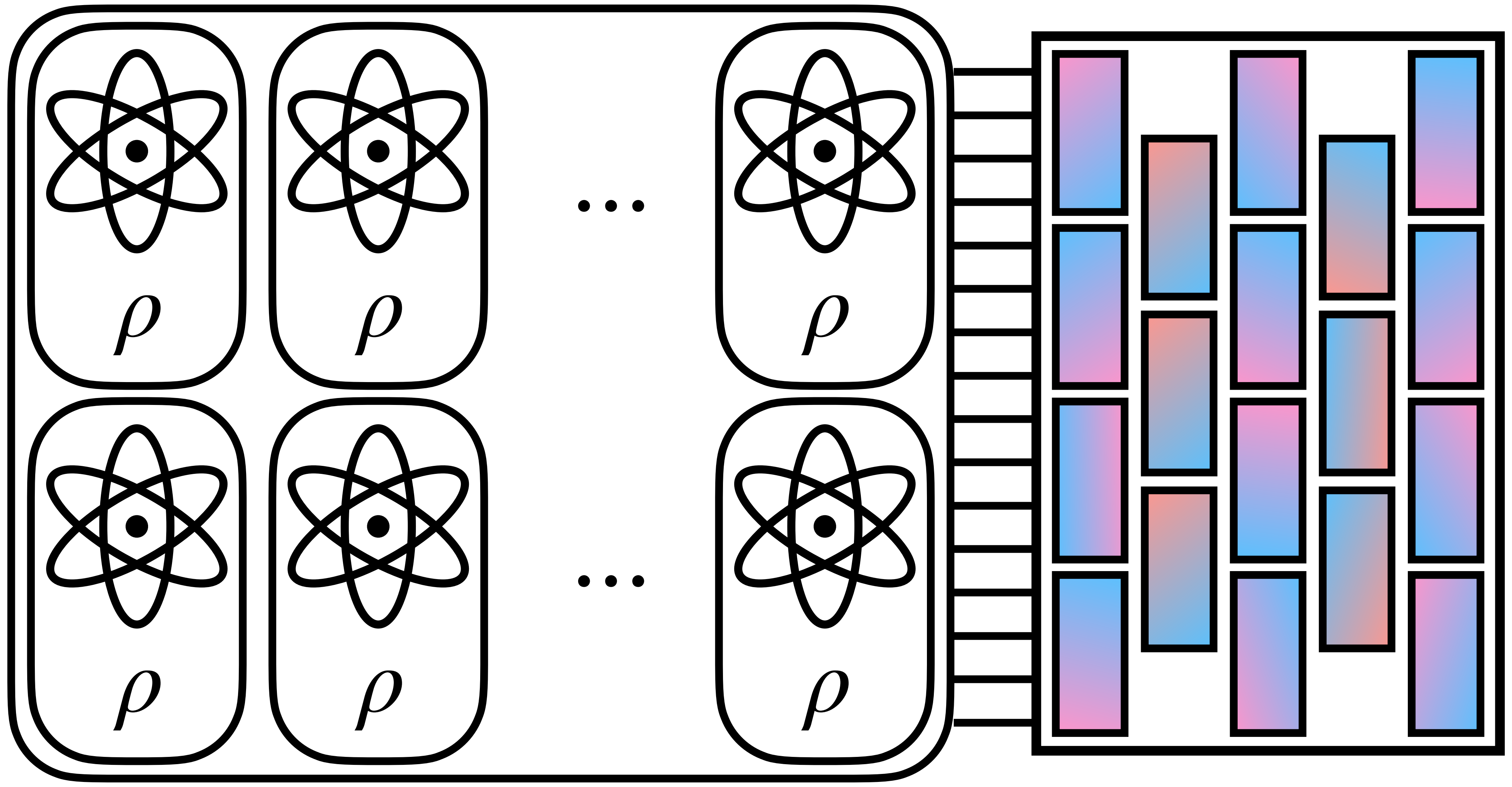
How can we estimate the value of $\mathrm{Tr}\left(\rho^k\right)$

when given access to copies of a quantum state ρ? (for large $k \in \mathbb{N}$)

Trace of powers

How can we estimate the value of $\mathrm{Tr}\left(\rho^k\right)$

when given access to copies of a quantum state ρ? (for large $k \in \mathbb{N}$)



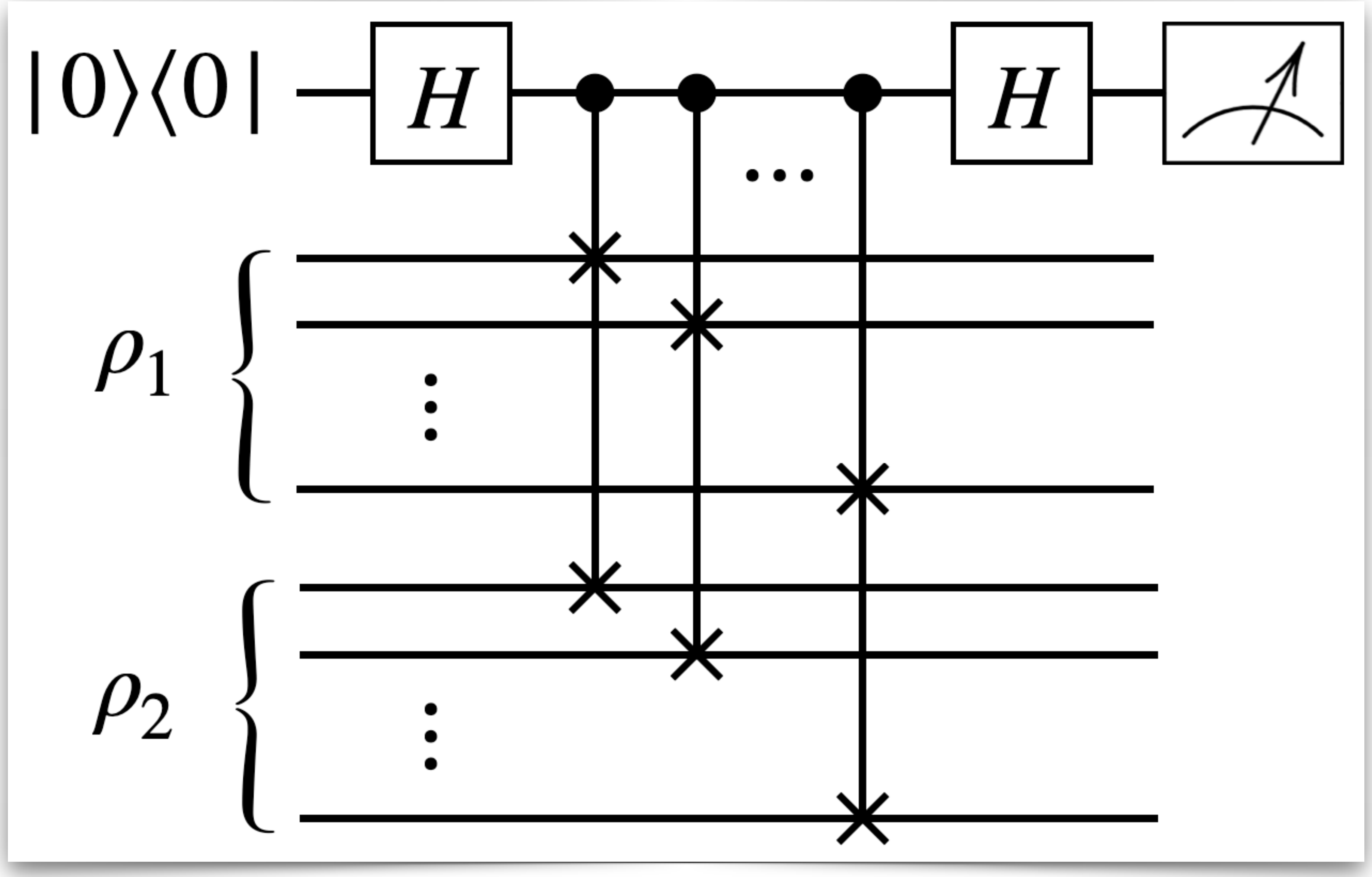
Given $\rho^{\otimes k}$

(efficient) quantum protocol
& some post processing

$\Rightarrow$ ε-approximate $\mathrm{Tr}\left(\rho^k\right)$

- Nonlinear function calculations
- Entanglement spectroscopy
- Quantum error mitigation
- Quantum Gibbs state preparation

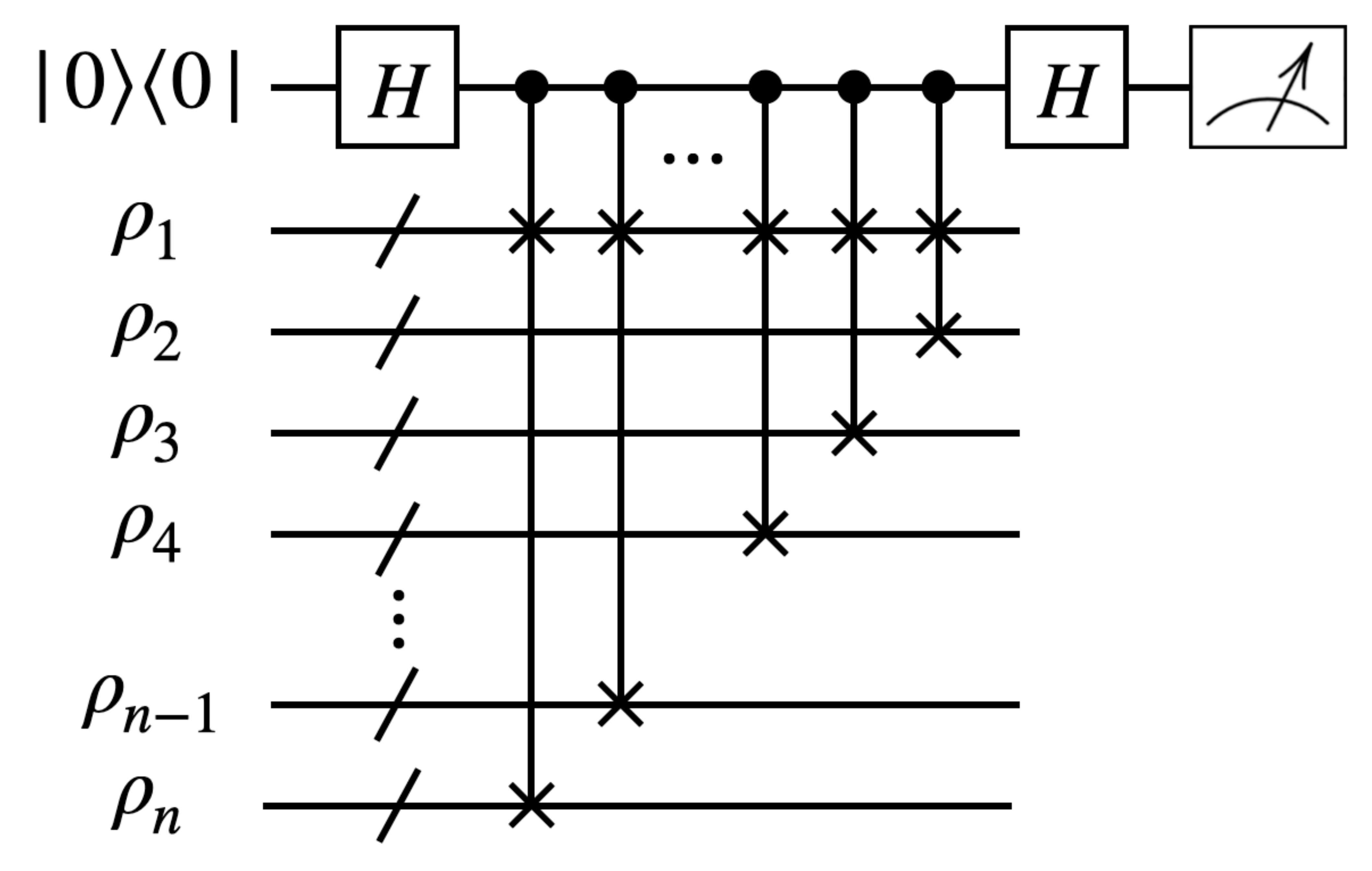
Swap test



$$\mathrm{Tr}\left(S\left(\rho_1 \otimes \rho_2\right)\right) = \mathrm{Tr}\left(\rho_1 \rho_2\right)$$

S = swap operator

Generalized swap test



To estimate $\mathrm{Tr}\left(\rho^k\right)$

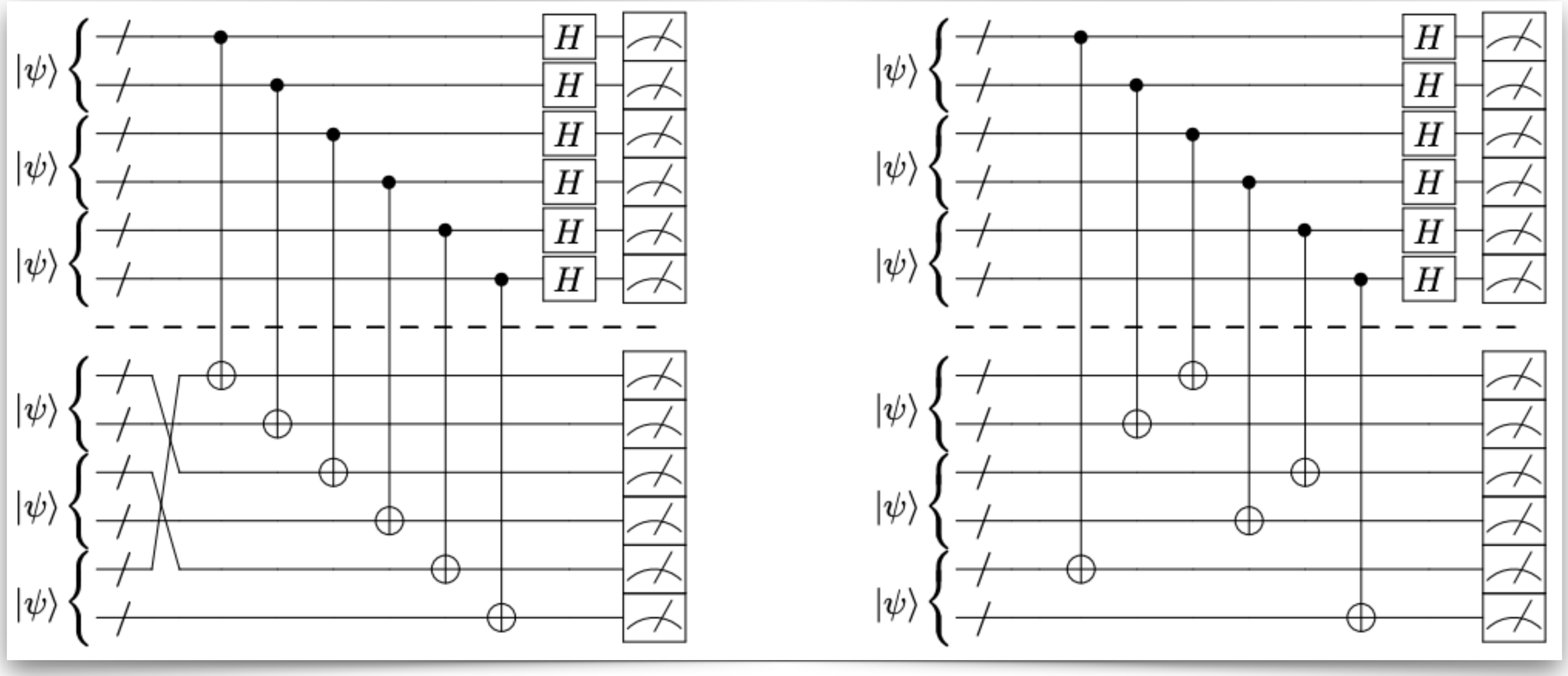
* Depth: $\mathcal{O}(k)$
* Width: $\mathcal{O}(k)$
* Copies: $\mathcal{O}(k)$
* Multi-qubit gates: $\mathcal{O}(k)$

$$\mathrm{Tr}\left(W^\pi\left(\rho_1 \otimes \rho_2 \otimes \cdots \otimes \rho_{n-1} \otimes \rho_n\right)\right) = \mathrm{Tr}\left(\rho_1\rho_2\cdots\rho_{n-1}\rho_n\right)$$

W^π = cyclic shift permutation operator

"Direct estimations of linear and nonlinear functionals of a quantum state". Physical Review Letters 88, 217901 (2002).

Two copy test



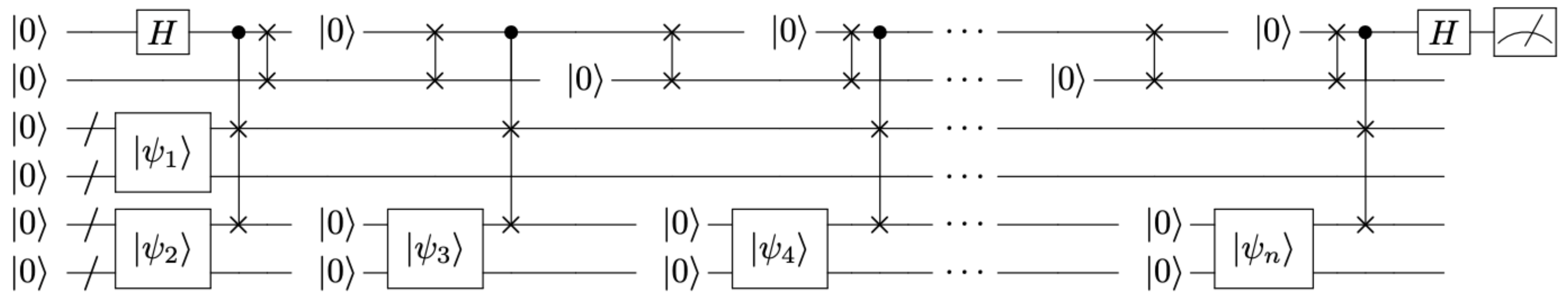
To estimate Tr $\left(\rho^k\right)$

* **Depth: $\mathcal{O}(1)$**
* Width: $\mathcal{O}(k)$
* Copies: $\mathcal{O}(k)$
* Multi-qubit gates: $\mathcal{O}(k)$

* Note that original entangled pure state $|\psi_{AB}\rangle$ needed, where $\rho_A = \mathrm{Tr}_B\left(|\psi\rangle\langle\psi|_{AB}\right)$

"Entanglement spectroscopy with a depth-two quantum circuit". Journal of Physics A: Mathematical and Theoretical 52, 044001 (2019).

Qubit-efficient two copy test



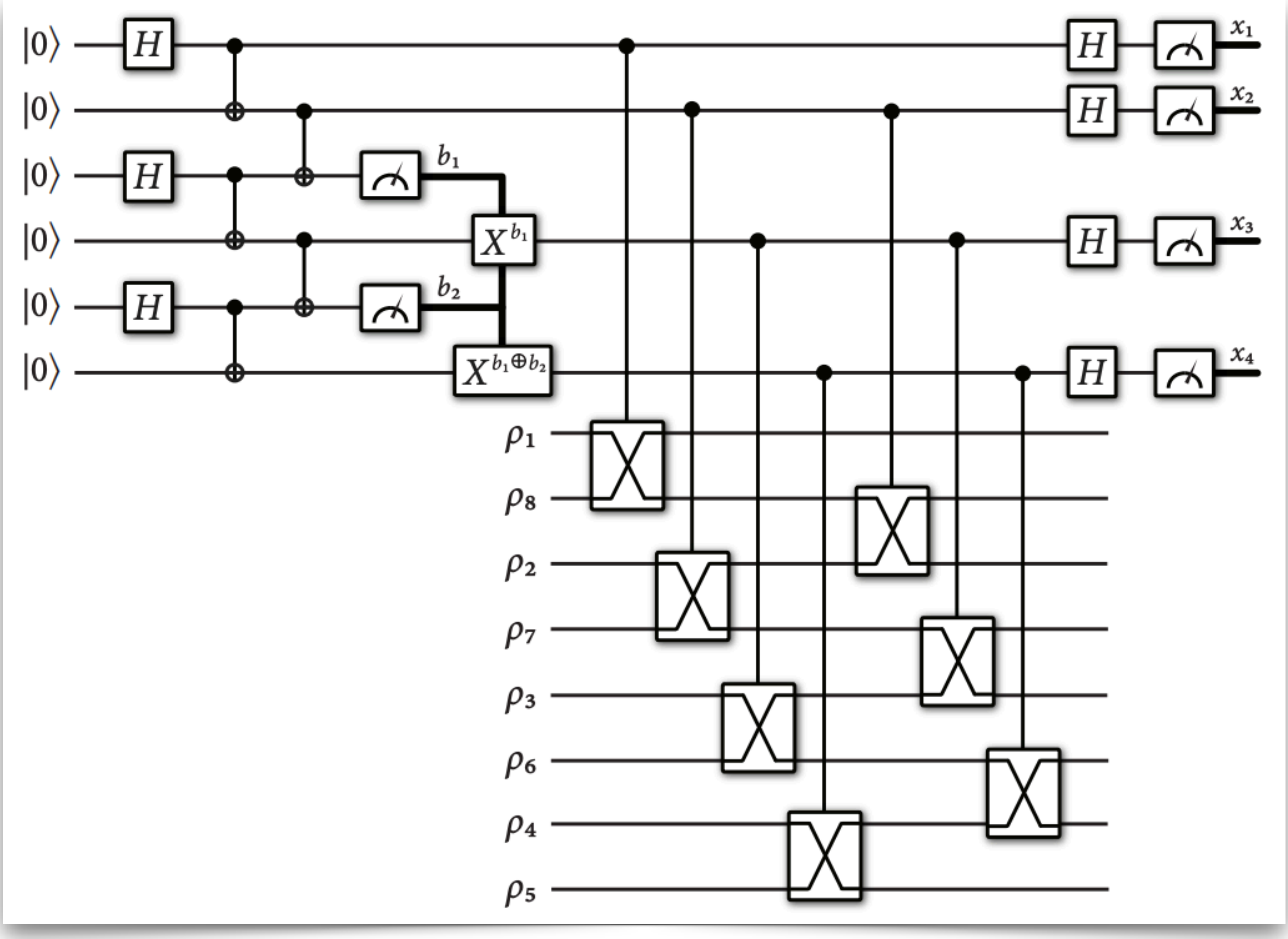
To estimate $\mathrm{Tr}\left(\rho^k\right)$

* Depth: $\mathcal{O}(k)$
* **Width: $\mathcal{O}(1)$**
* Copies: $\mathcal{O}(k)$
* Multi-qubit gates: $\mathcal{O}(k)$

* Using qubit-reset strategies
* Original entangled pure state $|\psi_{AB}\rangle$ needed

"Qubit- efficient entanglement spectroscopy using qubit resets". Quantum 5, 535 (2021).

Multivariate trace estimation algorithm



To estimate $\mathrm{Tr}\left(\rho^k\right)$

- **Depth: $\mathcal{O}(1)$**
- Width: $\mathcal{O}(k)$
- Copies: $\mathcal{O}(k)$
- Multi-qubit gates: $\mathcal{O}(k)$

* Inspired by Shor's error correction code

"Multivariate trace estimation in constant quantum depth". Quantum 8, 1220 (2024)

Comparison

Summary of resources required by different algorithms to estimate the values of $\{\mathrm{Tr}(\rho^i)\}_{i=1}^k$ within an error margin of ϵ

Algorithm	# Depth	# Qubits	# CSWAP	# Copies	Original $\lvert\psi\rangle$
Generalized swap test	$\mathcal{O}(k)$	$\mathcal{O}(k)$	$\mathcal{O}(k)$	$\mathcal{O}\left(\frac{k^2}{\epsilon^2}\right)$	***NOT*** required
Hadamard test	$\mathcal{O}(k)$	$\mathcal{O}(k)$	$\mathcal{O}(k)$	$\mathcal{O}\left(\frac{k^2}{\epsilon^2}\right)$	Required
Two copy test	$\mathcal{O}(1)$	$\mathcal{O}(k)$	$\mathcal{O}(k)$	$\mathcal{O}\left(\frac{k^2}{\epsilon^2}\right)$	Required
Qubit-efficient two copy test	$\mathcal{O}(k)$	$\mathcal{O}(1)$	$\mathcal{O}(k)$	$\mathcal{O}\left(\frac{k^2}{\epsilon^2}\right)$	Required
Multivariate trace estimation	$\mathcal{O}(1)$	$\mathcal{O}(k)$	$\mathcal{O}(k)$	$\mathcal{O}\left(\frac{k^2}{\epsilon^2}\right)$	***NOT*** required
Ours (this work)	$\mathcal{O}(1)$	$\mathcal{O}(r)$	$\mathcal{O}(r)$	$\mathcal{O}\left(\frac{k^2 r^4 \ln^2 r}{\epsilon^2}\right)$, $p_1 \approx 1$ $\mathcal{O}\left(\frac{r^2 \ln^2 r}{\epsilon^2}\right)$, otherwise	***NOT*** required

Overview

- Trace of powers & Literature review
- **Mathematical intuitions**
- Main results: algorithm, lemmas, theorems, corollaries
- Numerical simulations
- Computational hardness of TsallisQED$_q$ and TsallisQEA$_q$
- Applications
- Concluding remarks

Intuition

Consider two quantum states: $\rho = \sum_{i=1}^{r} p_i |\psi_i\rangle\langle\psi_i|$, $\sigma = \sum_{i=1}^{r} q_i |\phi_i\rangle\langle\phi_i|$

(assume descending order $p_1 \geq p_2 \geq \ldots \geq p_{r-1} \geq p_r \geq 0$ and also for q_i)

Intuition

Consider two quantum states: $\rho = \sum_{i=1}^{r} p_i |\psi_i\rangle\langle\psi_i|$, $\sigma = \sum_{i=1}^{r} q_i |\phi_i\rangle\langle\phi_i|$

(assume descending order $p_1 \geq p_2 \geq \ldots \geq p_{r-1} \geq p_r \geq 0$ and also for q_i)

If $\begin{cases} \mathrm{Tr}(\rho^1) = \mathrm{Tr}(\sigma^1) \\ \mathrm{Tr}(\rho^2) = \mathrm{Tr}(\sigma^2) \\ \vdots \\ \mathrm{Tr}(\rho^{r-1}) = \mathrm{Tr}(\sigma^{r-1}) \\ \mathrm{Tr}(\rho^r) = \mathrm{Tr}(\sigma^r) \end{cases}$ then, $\begin{cases} p_1 = q_1 \\ p_2 = q_2 \\ \vdots \\ p_{r-1} = q_{r-1} \\ p_r = q_r \end{cases}$?

Intuition

Consider two quantum states: $\rho = \sum_{i=1}^{r} p_i |\psi_i\rangle\langle\psi_i|, \ \sigma = \sum_{i=1}^{r} q_i |\phi_i\rangle\langle\phi_i|$

(assume descending order $p_1 \geq p_2 \geq \ldots \geq p_{r-1} \geq p_r \geq 0$ and also for q_i)

$$\text{If} \quad \begin{cases} \mathrm{Tr}(\rho^1) = \mathrm{Tr}(\sigma^1) \\ \mathrm{Tr}(\rho^2) = \mathrm{Tr}(\sigma^2) \\ \vdots \\ \mathrm{Tr}(\rho^{r-1}) = \mathrm{Tr}(\sigma^{r-1}) \\ \mathrm{Tr}(\rho^r) = \mathrm{Tr}(\sigma^r) \end{cases} \quad \text{then,} \quad \begin{cases} p_1 = q_1 \\ p_2 = q_2 \\ \vdots \\ p_{r-1} = q_{r-1} \\ p_r = q_r \end{cases} \quad ?$$

Answer: YES

Newton-Girard method

Key observation: exactly knowing $\{\mathrm{Tr}(\rho^i)\}_{i=1}^r$ is equivalent to knowing $\{p_i\}_{i=1}^r$.

Newton-Girard method

Key observation: exactly knowing $\{\mathrm{Tr}(\rho^i)\}_{i=1}^r$ is equivalent to knowing $\{p_i\}_{i=1}^r$.

Consider the equation having these eigenvalues as root in the form of

$$\prod_{m=1}^{r}(x - p_m) = 0.$$

Newton-Girard method

Key observation: exactly knowing $\{\mathrm{Tr}(\rho^i)\}_{i=1}^r$ is equivalent to knowing $\{p_i\}_{i=1}^r$.

Consider the equation having these eigenvalues as root in the form of

$$\prod_{m=1}^{r}(x - p_m) = 0.$$

The values of $\mathrm{Tr}(\rho^i)$ are now the i-th power sum of the roots. Denote the power sum as

$$P_i := \sum_{m=1}^{r} p_m^i = \mathrm{Tr}(\rho^i)\,.$$

Newton-Girard method

Simply expanding the terms to get: $\prod_{m=1}^{r}(x - p_m) = \sum_{k=0}^{r}(-1)^k a_k x^{r-k},$

Newton-Girard method

Simply expanding the terms to get: $\prod_{m=1}^{r}(x - p_m) = \sum_{k=0}^{r}(-1)^k a_k x^{r-k}$,

where

$$
\begin{aligned}
a_0 &= 1, \\
a_1 &= p_1 + p_2 + \cdots + p_r = \sum_{1 \le \alpha \le r} p_\alpha, \\
a_2 &= p_1 p_2 + p_1 p_3 + \cdots + p_{r-1} p_r = \sum_{1 \le \alpha < \beta \le r} p_\alpha p_\beta, \\
a_3 &= \sum_{1 \le \alpha < \beta < \gamma \le r} p_\alpha p_\beta p_\gamma, \\
&\vdots \\
a_r &= \prod_{i=1}^{r} p_i \,.
\end{aligned}
$$

Newton-Girard method

The Newton-Girard method states the relationship between

the elementary symmetric polynomials and the power sums recursively.

For all $r \geq k \geq 1, \;\; a_k = \dfrac{1}{k}\sum_{i=1}^{k}(-1)^{i-1}a_{k-i}P_i.$

Newton-Girard method

The Newton-Girard method states the relationship between
the elementary symmetric polynomials and the power sums recursively.

For all $r \geq k \geq 1, \quad a_k = \frac{1}{k} \sum_{i=1}^{k} (-1)^{i-1} a_{k-i} P_i.$

Uniquely defined $P_i = \mathrm{Tr}(\rho^i)$

Newton-Girard method

The Newton-Girard method states the relationship between

the elementary symmetric polynomials and the power sums recursively.

For all $r \geq k \geq 1, \quad a_k = \frac{1}{k} \sum_{i=1}^{k} (-1)^{i-1} a_{k-i} P_i.$

Also unique

Uniquely defined $P_i = \mathrm{Tr}(\rho^i)$

Newton-Girard method

The Newton-Girard method states the relationship between

the elementary symmetric polynomials and the power sums recursively.

For all $r \geq k \geq 1, \quad a_k = \frac{1}{k} \sum_{i=1}^{k} (-1)^{i-1} a_{k-i} P_i.$

Also unique

Uniquely defined $P_i = \mathrm{Tr}(\rho^i)$

Therefore, the set of eigenvalues is **uniquely determined**

as the roots of the equation $\prod_{m=1}^{r} (x - p_m)$.

Estimation with errors

Unfortunately in real-world situations,

we cannot exactly calculate the trace of powers.

Estimation with errors

Unfortunately in real-world situations,
we cannot exactly calculate the trace of powers.

Next challenge: If the error of estimated power sums $P_i = \mathrm{Tr}(\rho^i)$ is small,
are the roots obtained by the Newton-Girard method close to the eigenvalues of ρ?

Estimation with errors

Unfortunately in real-world situations,
we cannot exactly calculate the trace of powers.

Next challenge: If the error of estimated power sums $P_i = \mathrm{Tr}(\rho^i)$ is small,
are the roots obtained by the Newton-Girard method close to the eigenvalues of ρ?

Answer: NO

Estimation with errors

Unfortunately in real-world situations,
we cannot exactly calculate the trace of powers.

Next challenge: If the error of estimated power sums $P_i = \mathrm{Tr}(\rho^i)$ is small,
are the roots obtained by the Newton-Girard method close to the eigenvalues of ρ?

Answer: NO

Counterexample – *Wilkinson's polynomial*

* The location of the roots can be very sensitive to perturbations
in the coefficients of the polynomial

Wilkinson's polynomial

$$w(x) = \prod_{i=1}^{20} (x - i) = (x - 1)(x - 2)\cdots(x - 20)$$

Expanding the polynomial, one finds: $w(x) = x^{20} - 210x^{19} + 20615x^{18} - \cdots$

Wilkinson's polynomial

$$w(x) = \prod_{i=1}^{20} (x - i) = (x-1)(x-2)\cdots(x-20)$$

Expanding the polynomial, one finds: $w(x) = x^{20} - 210x^{19} + 20615x^{18} - \cdots$

If the coefficient of x^{19} is decreased from -210 by 2^{-23} to -210.000001192

Wilkinson's polynomial

$$w(x) = \prod_{i=1}^{20}(x - i) = (x-1)(x-2)\cdots(x-20)$$

Expanding the polynomial, one finds: $w(x) = x^{20} - 210x^{19} + 20615x^{18} - \cdots$

If the coefficient of x^{19} is decreased from -210 by 2^{-23} to -210.0000001192

The 20 roots become:

1.00000	2.00000	3.00000	4.00000	5.00000
6.00001	6.99970	8.00727	8.91725	20.84691
$10.09527 \pm 0.64350i$	$11.79363 \pm 1.65233i$	$13.99236 \pm 2.51883i$	$16.73074 \pm 2.81262i$	$19.50244 \pm 1.94033i$

Wilkinson's polynomial

$$w(x) = \prod_{i=1}^{20} (x - i) = (x-1)(x-2)\cdots(x-20)$$

Expanding the polynomial, one finds: $w(x) = x^{20} - 210x^{19} + 20615x^{18} - \cdots$

If the coefficient of x^{19} is decreased from -210 by 2^{-23} to -210.0000001192

The 20 roots become:

1.00000	2.00000	3.00000	4.00000	5.00000
6.00001	6.99970	8.00727	8.91725	20.84691
$10.09527 \pm 0.64350i$	$11.79363 \pm 1.65233i$	$13.99236 \pm 2.51883i$	$16.73074 \pm 2.81262i$	$19.50244 \pm 1.94033i$

* Some of the roots are greatly displaced, even though the change to the coefficient is tiny and the original roots seem widely spaced.

Overview

- Trace of powers & Literature review
- Mathematical intuitions
- **Main results: algorithm, lemmas, theorems, corollaries**
- Numerical simulations
- Computational hardness of TsallisQED$_q$ and TsallisQEA$_q$
- Applications
- Concluding remarks

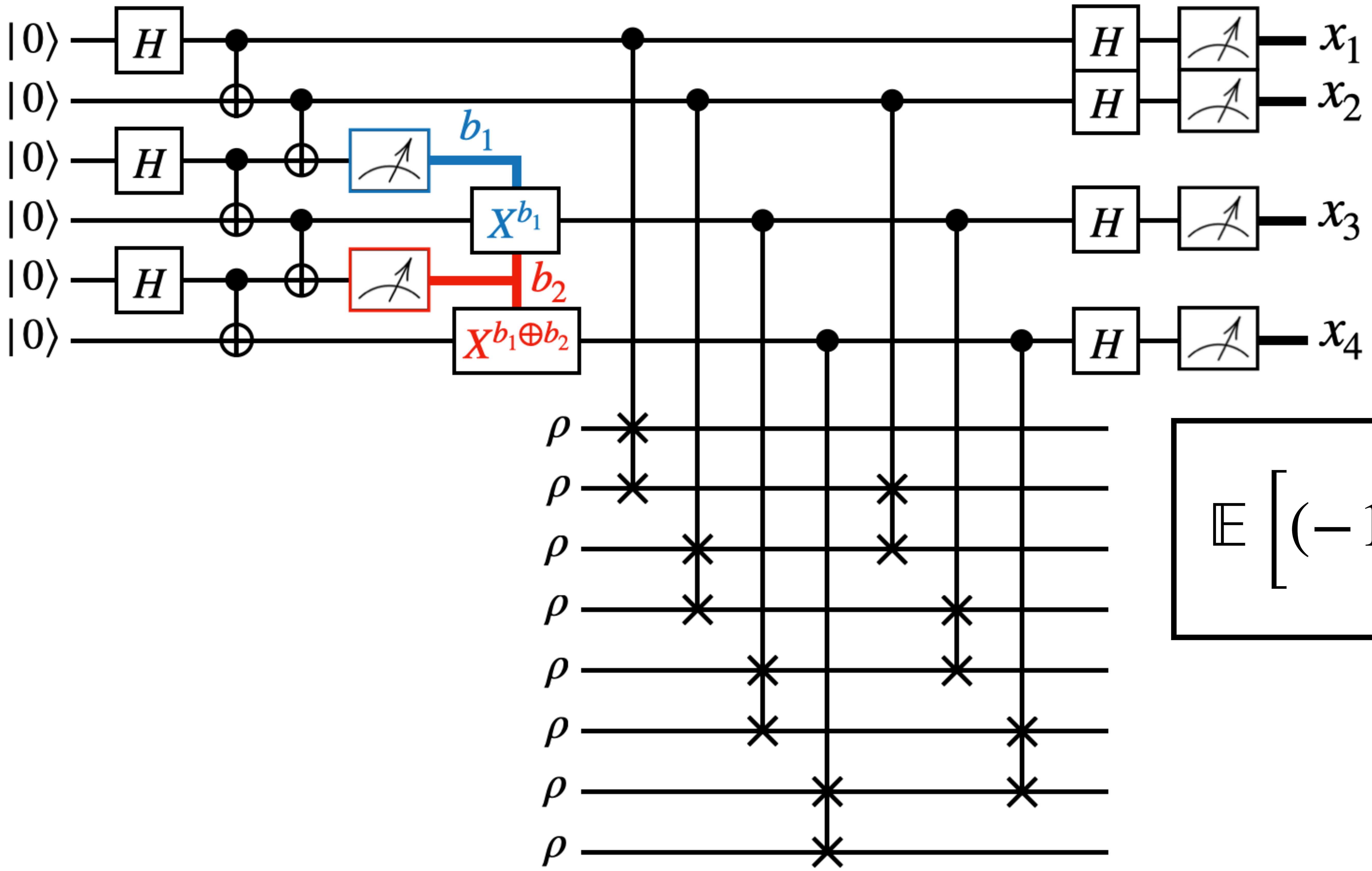
Main result: iterative $\mathrm{Tr}(\rho^i)$ estimation algorithm

[1] Estimate $P_i = \mathrm{Tr}(\rho^i)$ for $i = 1,2,\ldots,r$, using a constant-depth quantum circuit consisting of $\mathcal{O}(i)$ qubits and $\mathcal{O}(i)$ CSWAP operations using multivariate trace estimation, where r is the rank of ρ, and denote the estimated value as Q_i. $\rightarrow \quad Q_1, \ldots, Q_r$

Main result: iterative $\mathrm{Tr}(\rho^i)$ estimation algorithm

[1] Estimate $P_i = \mathrm{Tr}(\rho^i)$ for $i = 1,2,\ldots,r$, using a constant-depth quantum circuit consisting of $\mathcal{O}(i)$ qubits and $\mathcal{O}(i)$ CSWAP operations using multivariate trace estimation, where r is the rank of ρ, and denote the estimated value as Q_i. $\rightarrow\ Q_1,\ldots,Q_r$

Example ($i = 8$)



$$\mathbb{E}\left[(-1)^{\sum_{i=1}^{4} x_i}\right] = \mathrm{Re}\left(\mathrm{Tr}(\rho^8)\right)$$

Main result: iterative $\mathrm{Tr}(\rho^i)$ estimation algorithm

[1] Estimate $P_i = \mathrm{Tr}(\rho^i)$ for $i = 1,2,\ldots,r$, using a constant-depth quantum circuit consisting of $\mathcal{O}(i)$ qubits and $\mathcal{O}(i)$ CSWAP operations using multivariate trace estimation, where r is the rank of ρ, and denote the estimated value as Q_i.

$\rightarrow \; Q_1, \ldots, Q_r$

[2] Calculate the elementary symmetric polynomial $b_i = \frac{1}{i}\sum_{\ell=1}^{i}(-1)^{\ell-1} b_{i-\ell} Q_\ell, \; b_1 = 1, \; 1 \le i \le r$.

$\rightarrow \; b_1, \ldots, b_r$

Main result: iterative $\mathrm{Tr}(\rho^i)$ estimation algorithm

[1] Estimate $P_i = \mathrm{Tr}(\rho^i)$ for $i = 1,2,\ldots,r$, using a constant-depth quantum circuit consisting of $\mathcal{O}(i)$ qubits and $\mathcal{O}(i)$ CSWAP operations using multivariate trace estimation, where r is the rank of ρ, and denote the estimated value as Q_i.

$\rightarrow\ Q_1, \ldots, Q_r$

[2] Calculate the elementary symmetric polynomial $b_i = \frac{1}{i}\sum_{\ell=1}^{i}(-1)^{\ell-1} b_{i-\ell} Q_\ell,\ b_1 = 1,\ 1 \le i \le r$.

$\rightarrow\ b_1, \ldots, b_r$

[3] Calculate the estimated value Q_i $(i > r)$ by $Q_i = \sum_{\ell=1}^{r}(-1)^{\ell-1} b_\ell Q_{i-\ell} \sim \mathrm{Tr}(\rho^i)$.

$\rightarrow\ Q_{r+1}, \ldots$

Main result: iterative $\mathrm{Tr}(\rho^i)$ estimation algorithm

[1] Estimate $P_i = \mathrm{Tr}(\rho^i)$ for $i = 1,2,\ldots,r$, using a constant-depth quantum circuit consisting of $\mathcal{O}(i)$ qubits and $\mathcal{O}(i)$ CSWAP operations using multivariate trace estimation, where r is the rank of ρ, and denote the estimated value as Q_i.

$\rightarrow\ Q_1, \ldots, Q_r$

[2] Calculate the elementary symmetric polynomial $b_i = \frac{1}{i}\sum_{\ell=1}^{i}(-1)^{\ell-1} b_{i-\ell} Q_\ell,\ b_1 = 1,\ 1 \le i \le r$.

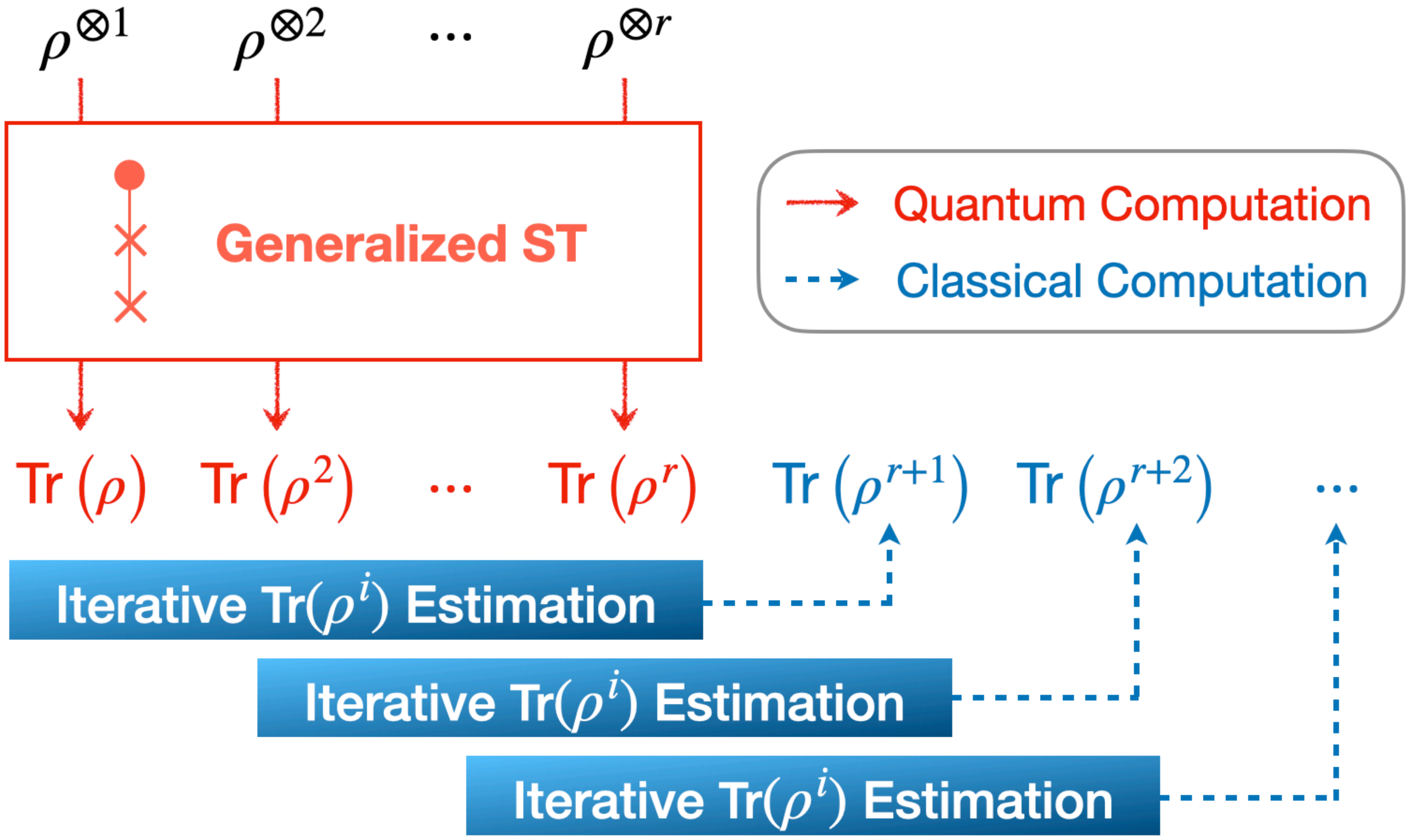
$\rightarrow\ b_1, \ldots, b_r$

[3] Calculate the estimated value Q_i $(i > r)$ by $Q_i = \sum_{\ell=1}^{r}(-1)^{\ell-1} b_\ell Q_{i-\ell} \sim \mathrm{Tr}(\rho^i)$.

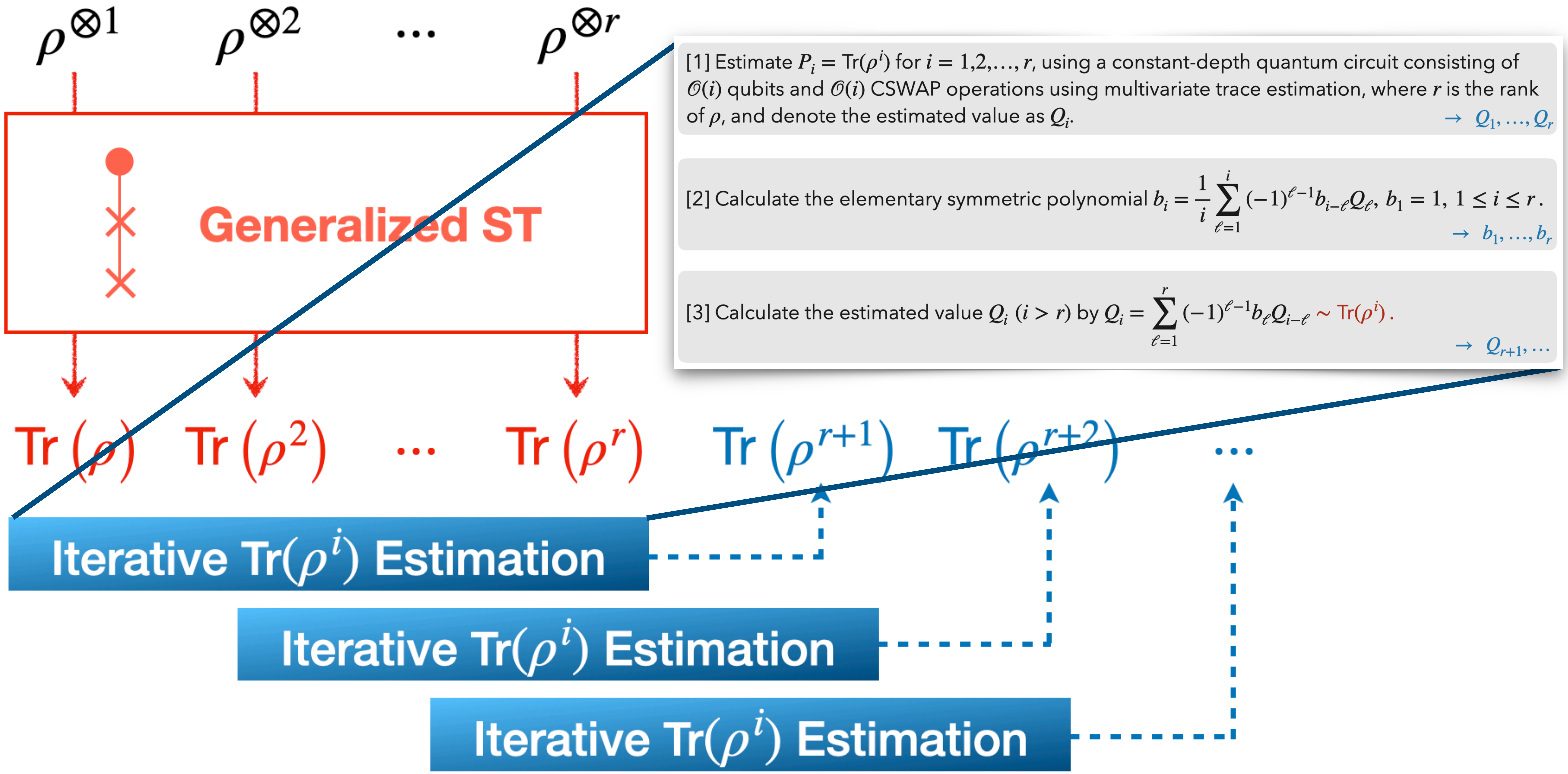
$\rightarrow\ Q_{r+1}, \ldots$

The output of our algorithm Q_i guarantees an *ϵ-approximate trace of powers* $\mathrm{Tr}(\rho^i)$

Main result: iterative $\mathrm{Tr}(\rho^i)$ estimation algorithm



Main result: iterative $\mathrm{Tr}(\rho^i)$ estimation algorithm



Rank is all you need — Lemma 1

- $a_k, b_k =$ the elementary symmetric polynomials corresponding to each P_i and Q_i.
- Q_i is defined as the estimated value of $P_i = \mathrm{Tr}(\rho^i)$ on a quantum device for $i \leq r$ otherwise $Q_i = \sum_{\ell=1}^{r} (-1)^{\ell-1} b_\ell Q_{i-\ell}$.

Rank is all you need – Lemma 1

- $a_k, b_k =$ the elementary symmetric polynomials corresponding to each P_i and Q_i.
- Q_i is defined as the estimated value of $P_i = \mathrm{Tr}(\rho^i)$ on a quantum device for $i \leq r$ otherwise $Q_i = \sum_{\ell=1}^{r} (-1)^{\ell-1} b_\ell Q_{i-\ell}$.

Lemma 1

Let $t_k := b_k - a_k$, then the following holds: $|t_k| \leq \sum_{j=1}^{k} \frac{|\epsilon_j|}{j}$

where $\epsilon_j = Q_j - P_j$ is the error that occurred by the estimation of $P_j = \mathrm{Tr}(\rho^j)$.

Rank is all you need — Lemma 1

- $a_k, b_k =$ the elementary symmetric polynomials corresponding to each P_i and Q_i.
- Q_i is defined as the estimated value of $P_i = \mathrm{Tr}(\rho^i)$ on a quantum device for $i \le r$ otherwise $Q_i = \sum_{\ell=1}^{r} (-1)^{\ell-1} b_\ell Q_{i-\ell}$.

Lemma 1

Let $t_k := b_k - a_k$, then the following holds: $|t_k| \le \sum_{j=1}^{k} \frac{|\epsilon_j|}{j}$

where $\epsilon_j = Q_j - P_j$ is the error that occurred by the estimation of $P_j = \mathrm{Tr}(\rho^j)$.

Proof. By strong mathematical induction logic.

Rank is all you need – Theorem 1

Theorem 1

Suppose that

$$\varepsilon_i := |\epsilon_i| = |P_i - Q_i| < \frac{\epsilon}{2T \ln r}$$

holds for $i = 1,2,\ldots,r$, where T is defined as:

$$T = \sum_{i=1}^{r} \frac{p_i(1 - p_i^k)(1 - p_i^r)}{(1 - p_i)^2} \leq kr\,.$$

Then the following relation always holds:

$$|\epsilon_i| = |P_i - Q_i| \leq \epsilon$$

for $i = 1,2,\ldots,k$.

Rank is all you need – Theorem 1

Theorem 1

Suppose that

$$\varepsilon_i := |\epsilon_i| = |P_i - Q_i| < \frac{\epsilon}{2T \ln r}$$

holds for $i = 1,2,\ldots,r$, where T is defined as:

$$T = \sum_{i=1}^{r} \frac{p_i(1-p_i^k)(1-p_i^r)}{(1-p_i)^2} \leq kr\,.$$

Then the following relation always holds:

$$|\epsilon_i| = |P_i - Q_i| \leq \epsilon$$

for $i = 1,2,\ldots,k$.

Proof. By applying **Lemma 1** + long calculation with some mathematical trick.

Rank is all you need – Corollary 1

Corollary 1

To estimate $\mathrm{Tr}(\rho^i)$ for all $i \leq k$ within an additive error of ϵ and with a success probability of at least $1-\delta$, where $\delta \in (0,1)$, it is necessary to estimate each $\mathrm{Tr}(\rho^j)$ for $j \leq r$ within an additive error of ε_j, as defined in **Theorem 1.** This can be achieved by using

$$\mathcal{O}\left(\frac{T^2}{\epsilon^2}\ln^2 r \ln\left(\frac{1}{\delta}\right)\right)$$

runs on a constant-depth quantum circuit consisting of $\mathcal{O}(j)$ qubits and $\mathcal{O}(j)$ CSWAP operations. Here, T is defined in **Theorem 1.**

Rank is all you need – Corollary 1

Corollary 1

To estimate $\mathrm{Tr}(\rho^i)$ for all $i \leq k$ within an additive error of ϵ and with a success probability of at least $1-\delta$, where $\delta \in (0,1)$, it is necessary to estimate each $\mathrm{Tr}(\rho^j)$ for $j \leq r$ within an additive error of ε_j, as defined in **Theorem 1.** This can be achieved by using

$$\mathcal{O}\left(\frac{T^2}{\epsilon^2}\ln^2 r \ln\left(\frac{1}{\delta}\right)\right)$$

runs on a constant-depth quantum circuit consisting of $\mathcal{O}(j)$ qubits and $\mathcal{O}(j)$ CSWAP operations. Here, T is defined in **Theorem 1.**

* Note that T scales from a constant to kr, mainly depending on the largest eigenvalue p_1 of ρ.
* If p_1 is not close to 1, then $T = \mathcal{O}(1)$. (This implies that if ρ is far from a pure state.)

Rank is all you need – Theorem 2

Extension: Estimating $\mathrm{Tr}(M\rho^k)$, the trace of powers with arbitrary observables M.

Rank is all you need – Theorem 2

Extension: Estimating $\mathrm{Tr}(M\rho^k)$, the trace of powers with arbitrary observables M.

Theorem 2

Suppose that $\varepsilon_{i,M} := |\epsilon_{i,M}| = |P_{i,M} - Q_{i,M}| < \frac{\epsilon}{2}$, and $\varepsilon_i = |\epsilon_i| = |P_i - Q_i| < \frac{\epsilon}{2T\|M\|_\infty \ln r}$, holds for $i = 1,2,\ldots,r$, where the operator norm $\|M\|_\infty$ is defined corresponding to the ∞-norm for vectors $\|x\|_\infty$, as $\|M\|_\infty = \sup_{x\neq 0} \frac{\|Mx\|_\infty}{\|x\|_\infty}$, $T = \sum_{i=1}^{r} \frac{p_i(1-p_i^k)(1-p_i^r)}{(1-p_i)^2} \leq kr$.

Then the following holds:

$$|\epsilon_{i,M}| = |P_{i,M} - Q_{i,M}| \leq \epsilon$$

for $i = 1,2,\ldots,k$.

Rank is all you need – Corollary 2

Corollary 2

Suppose there is an efficient decomposition $M = \sum_{\ell=1}^{N_M} a_\ell P_\ell$, where $a_\ell \in \mathbb{R}$ and $P_\ell = \sigma_{\ell_1} \otimes \cdots \otimes \sigma_{\ell_n}$ are tensor products of Pauli operators $\sigma_{\ell_1}, \ldots, \sigma_{\ell_n} \in \{\sigma_x, \sigma_y, \sigma_z, I\}$. The quantity $\sum_{\ell=1}^{N_M} |a_\ell| = \mathcal{O}(c)$ is bounded by a constant c.

To estimate $\mathrm{Tr}(M\rho^i)$ for all $i \leq k$ within an additive error of ϵ and with a success probability of at least $1-\delta$, where $\delta \in (0,1)$, it is necessary to estimate each $\mathrm{Tr}(M\rho^j)$ for $j \leq r$ within an additive error of $\varepsilon_{j,M}$.

This can be achieved by using $\mathcal{O}\left(\frac{c^2 N_M}{\epsilon^2} \ln\left(\frac{1}{\delta}\right)\right)$ runs on a constant-depth quantum circuit consisting of $\mathcal{O}(j)$ qubits and $\mathcal{O}(j)$ CSWAP operations, and estimating each $\mathrm{Tr}(\rho^{j'})$ for $j' \leq r$ within an additive error of $\varepsilon_{j'}$, by using $\mathcal{O}\left(\frac{T^2}{\epsilon^2} \ln^2 r \ln\left(\frac{1}{\delta}\right)\right)$ runs on a constant-depth quantum circuit consisting of $\mathcal{O}(j')$ qubits and $\mathcal{O}(j')$ CSWAP operations. Here, $\varepsilon_{j,M}$, $\varepsilon_{j'}$ and T are defined in **Theorem 2.**

(Effective) Rank is all you need – Lemma 2

$$\mathcal{O}\left(\frac{T^2}{\epsilon^2}\ln^2 r \ln\left(\frac{1}{\delta}\right)\right)$$

* If p_1 is not close to 1, then $T = \mathcal{O}(1)$.
* ***Our algorithm may perform suboptimally on pure states.***

(Effective) Rank is all you need – Lemma 2

$$\mathcal{O}\left(\frac{T^2}{\epsilon^2}\ln^2 r \ln\left(\frac{1}{\delta}\right)\right)$$

- If p_1 is not close to 1, then $T = \mathcal{O}(1)$.
- ***Our algorithm may perform suboptimally on pure states.***

Definition (Effective rank)

Effective rank with error ϵ is defined as the minimum value r_ϵ, which satisfies $\sum_{i=1}^{r_\epsilon} p_i > 1 - \epsilon$.

(Effective) Rank is all you need – Lemma 2

$$\mathcal{O}\left(\frac{T^2}{\epsilon^2}\ln^2 r \ln\left(\frac{1}{\delta}\right)\right)$$

- * If p_1 is not close to 1, then $T = \mathcal{O}(1)$.
- * ***Our algorithm may perform suboptimally on pure states.***

Definition (Effective rank)

Effective rank with error ϵ is defined as the minimum value r_ϵ, which satisfies $\sum_{i=1}^{r_\epsilon} p_i > 1 - \epsilon$.

Lemma 2

For every integer $k \geq 2$, the following holds: $\left|\sum_{i=1}^{r} p_i^k - \sum_{i=1}^{r_\epsilon} p_i^k\right| < \epsilon^2$.

(Effective) Rank is all you need – Lemma 2

$$\mathcal{O}\left(\frac{T^2}{\epsilon^2}\ln^2 r \ln\left(\frac{1}{\delta}\right)\right)$$

* If p_1 is not close to 1, then $T = \mathcal{O}(1)$.
* ***Our algorithm may perform suboptimally on pure states.***

Definition (Effective rank)

Effective rank with error ϵ is defined as the minimum value r_ϵ, which satisfies $\sum_{i=1}^{r_\epsilon} p_i > 1 - \epsilon$.

Lemma 2

For every integer $k \geq 2$, the following holds: $\left|\sum_{i=1}^{r} p_i^k - \sum_{i=1}^{r_\epsilon} p_i^k\right| < \epsilon^2$.

Lemma 2 suggests that we only need $\{\mathrm{Tr}(\rho^i)\}_{i=1}^{r_\epsilon}$ for the estimation.

(Effective) Rank is all you need – Lemma 2

$$\mathcal{O}\left(\frac{T^2}{\epsilon^2}\ln^2 r \ln\left(\frac{1}{\delta}\right)\right)$$

* If p_1 is not close to 1, then $T = \mathcal{O}(1)$.
* ***Our algorithm may perform suboptimally on pure states.***

Definition (Effective rank)

Effective rank with error ϵ is defined as the minimum value r_ϵ, which satisfies $\sum_{i=1}^{r_\epsilon} p_i > 1 - \epsilon$.

Lemma 2

For every integer $k \geq 2$, the following holds: $\left| \sum_{i=1}^{r} p_i^k - \sum_{i=1}^{r_\epsilon} p_i^k \right| < \epsilon^2$.

Lemma 2 suggests that we only need $\{\mathrm{Tr}(\rho^i)\}_{i=1}^{r_\epsilon}$ for the estimation.

As the maximum eigenvalue of ρ approaches 1, the r_ϵ decreases!

(Possible to resolve issues with pure states)

Finding the $r = \text{rank}(\rho)$

- To proceed with our algorithm, we need to know the rank of quantum state.
- Due to the concept of effective rank, **the estimated rank doesn't have to be precise.**
- Several approaches:
  - Variational algorithms via purity minimization
  - Binary search-based heuristic

Variational algorithms via purity minimization

$$C(\theta) = \mathrm{Tr}\left[\left[\frac{\eta(\theta)^k \rho \eta(\theta)^k}{\mathrm{Tr}\left(\eta(\theta)^k \rho \eta(\rho)^k\right)}\right]^2\right]$$

[Phys. Rev. Research **3**, 033251]

- where k is any positive integer, ρ is the density matrix of a possibly unnormalized mixed state, and $\eta(\theta)$ is some normalized ansatz state. The goal is to minimize the cost function $C(\theta)$ by varying θ in order to find the optimal ansatz $\eta(\theta^*)$.
- Purity optimization allows one to find a lower bound estimate of the rank.
- This comes directly from the fact that $r = \mathrm{rank}(\rho) \approx \dfrac{1}{C(\theta^*)}$ and that in general for any θ, we have $d \geq \dfrac{1}{C(\theta)}$.

Binary search-based heuristic

1. Choose r to be reasonably large.

2. Apply our algorithm to obtain estimated values for $\mathrm{Tr}(\rho^i)$, for $r+1 \leq i \leq 2r$.

3. Analyze the difference between the estimated values and the measurements for $r+1 \leq i \leq 2r$.

4. Do the following:

   A. If the difference is significant, double the value of r.

   B. Otherwise, keep the current value of r.

Overview

- Trace of powers & Literature review
- Mathematical intuitions
- Main results: algorithm, lemmas, theorems, corollaries
- **Numerical simulations**
- Computational hardness of $\mathrm{TsallisQED}_q$ and $\mathrm{TsallisQEA}_q$
- Applications
- Concluding remarks

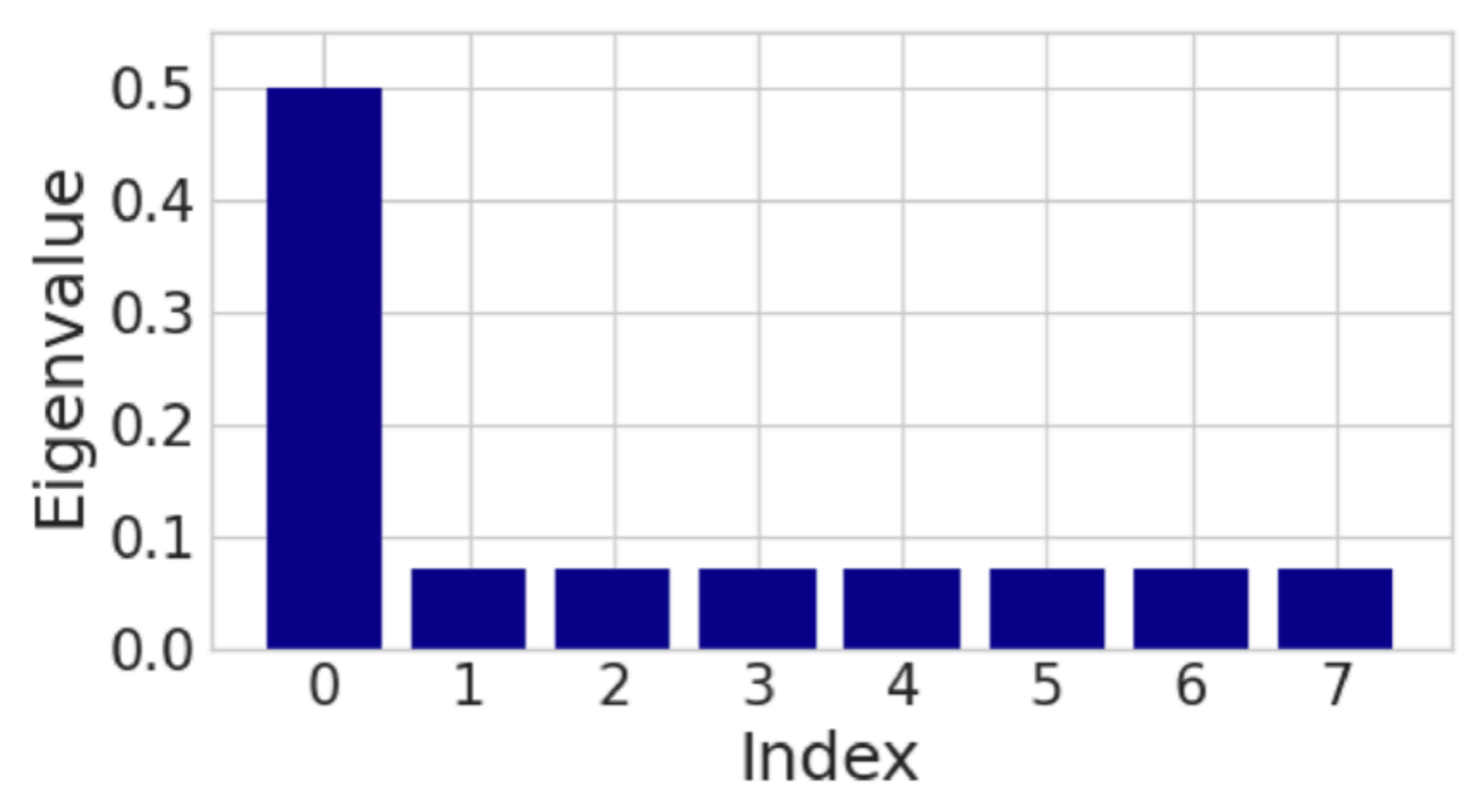
Reduction of required runs

Analyze the behavior of $T = \sum_{i=1}^{r} \frac{p_i(1-p_i^k)(1-p_i^r)}{(1-p_i)^2}$ as the eigenvalues of ρ change.

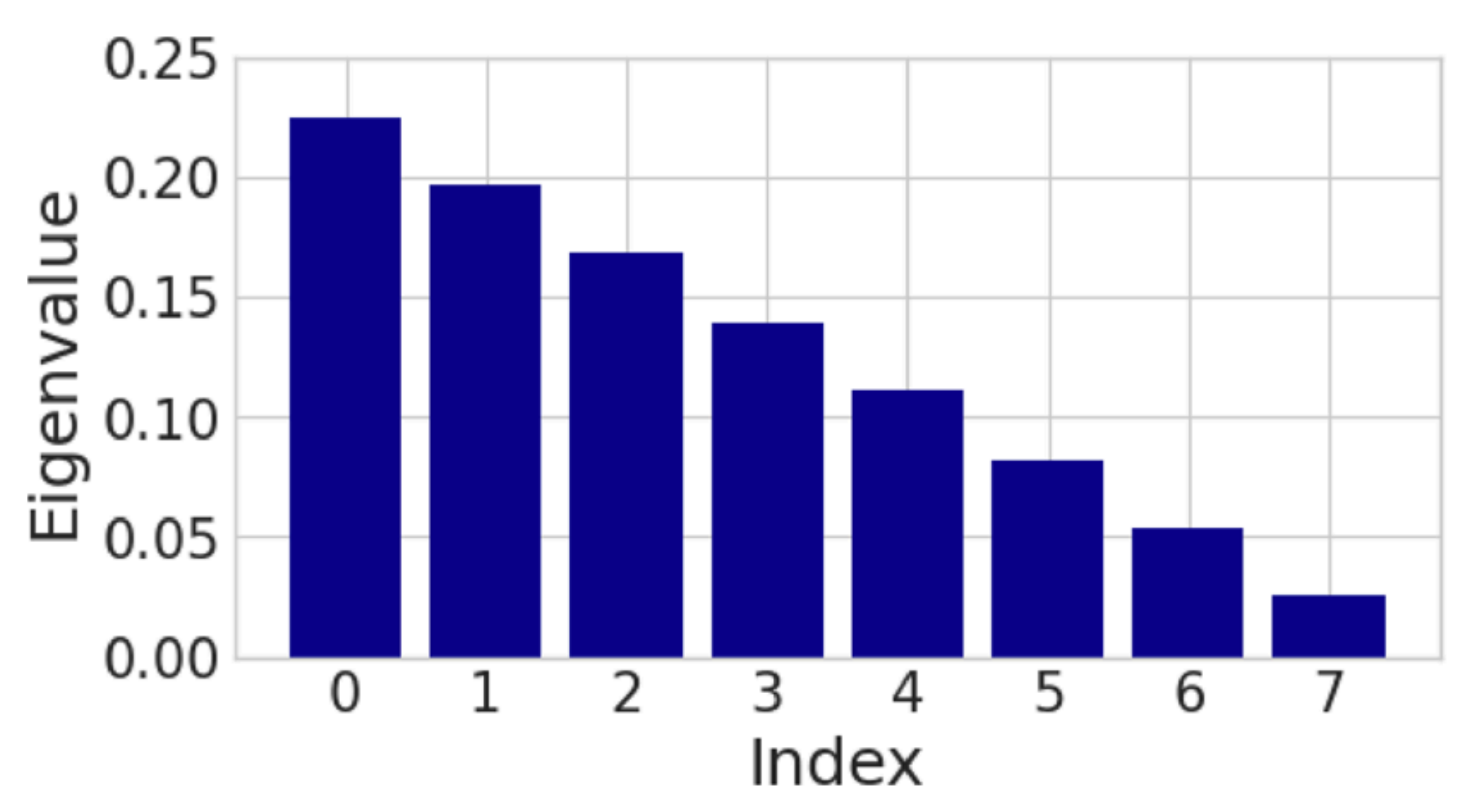
Reduction of required runs

Analyze the behavior of $T = \sum_{i=1}^{r} \frac{p_i(1-p_i^k)(1-p_i^r)}{(1-p_i)^2}$ as the eigenvalues of ρ change.

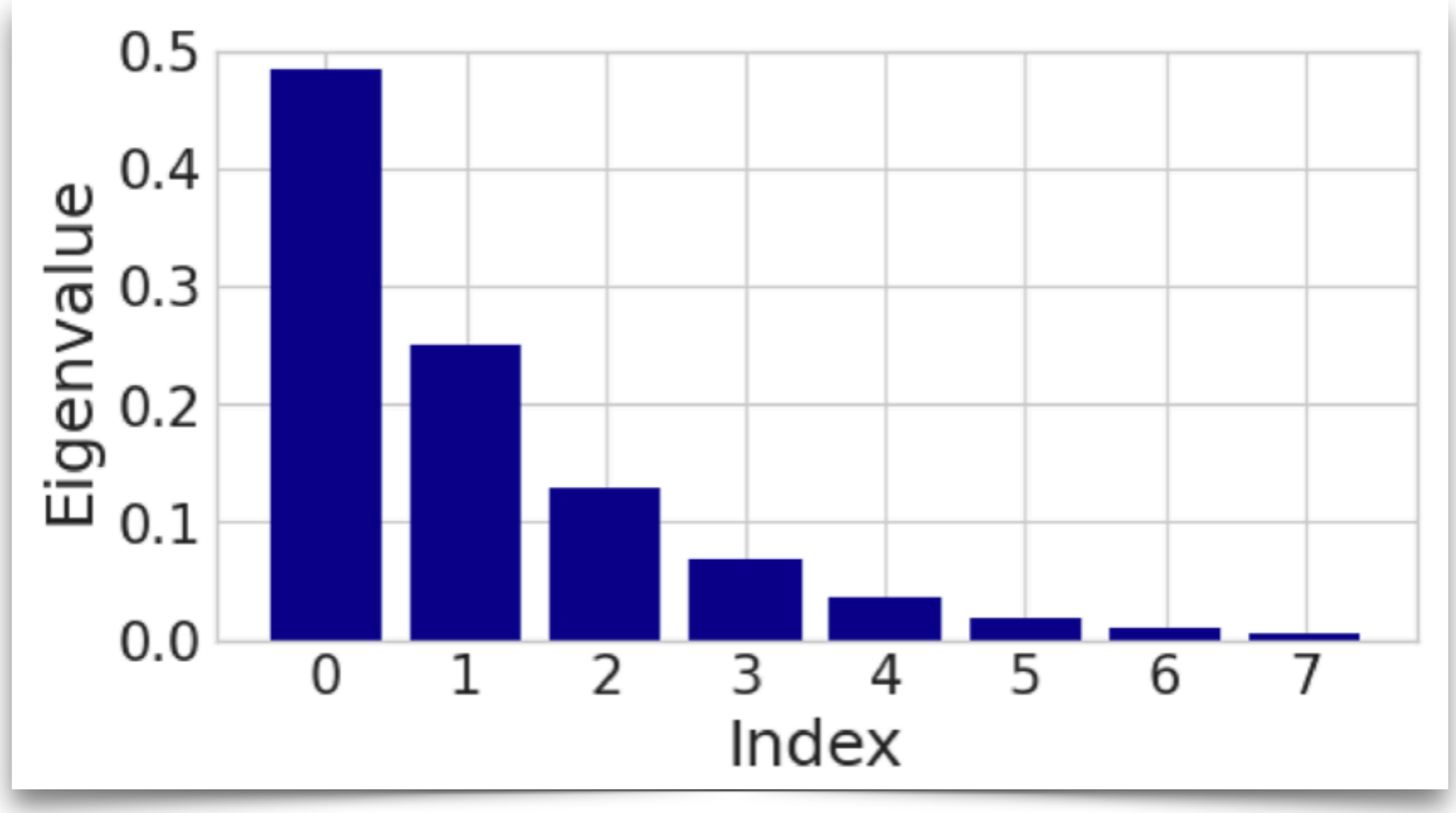
Since we don't know the exact distribution of the eigenvalues of an arbitrary density matrix ρ, we consider several typical cases.



Uniform non-maximum

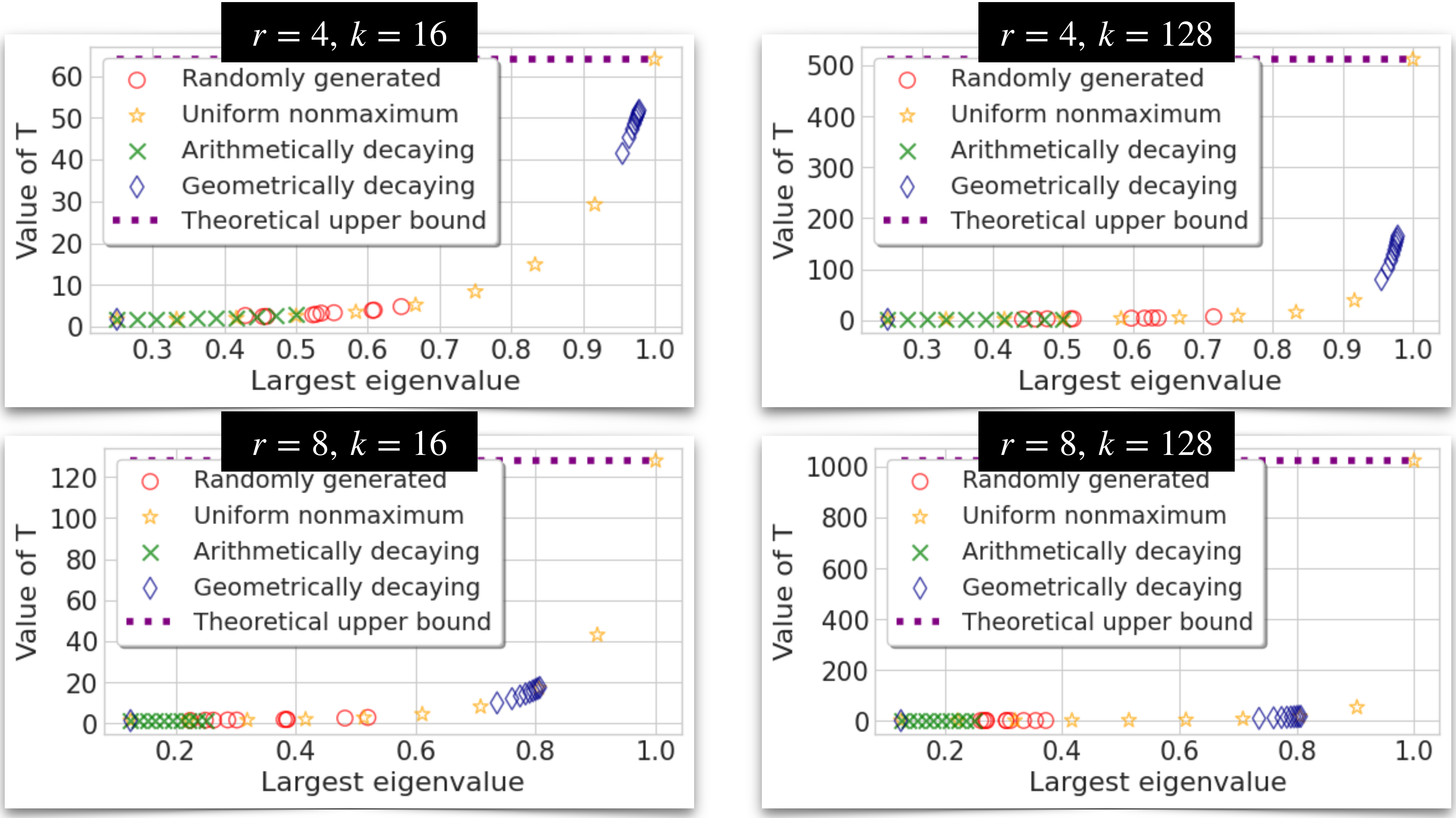


Arithmetically decaying

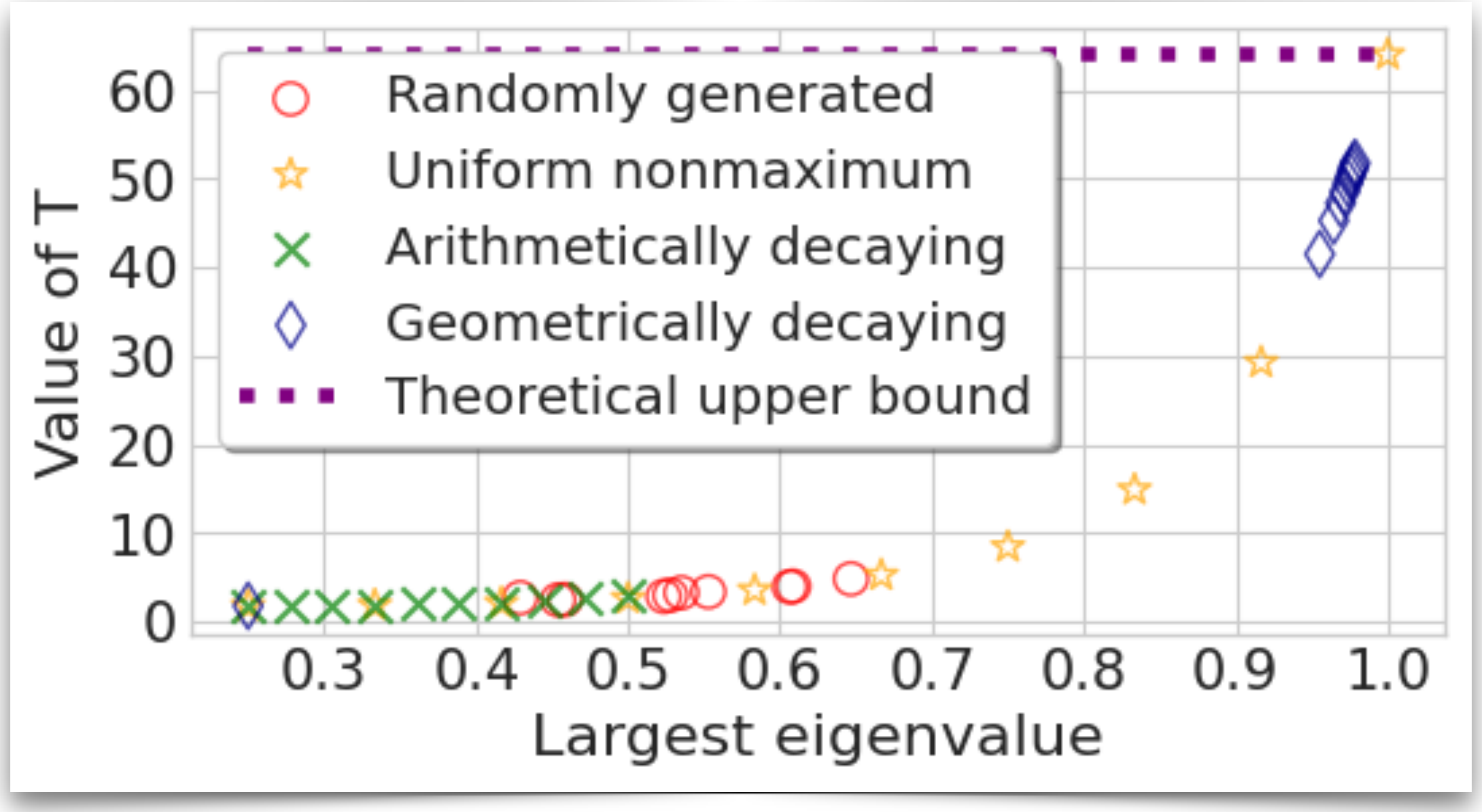


Geometrically decaying

Reduction of required runs

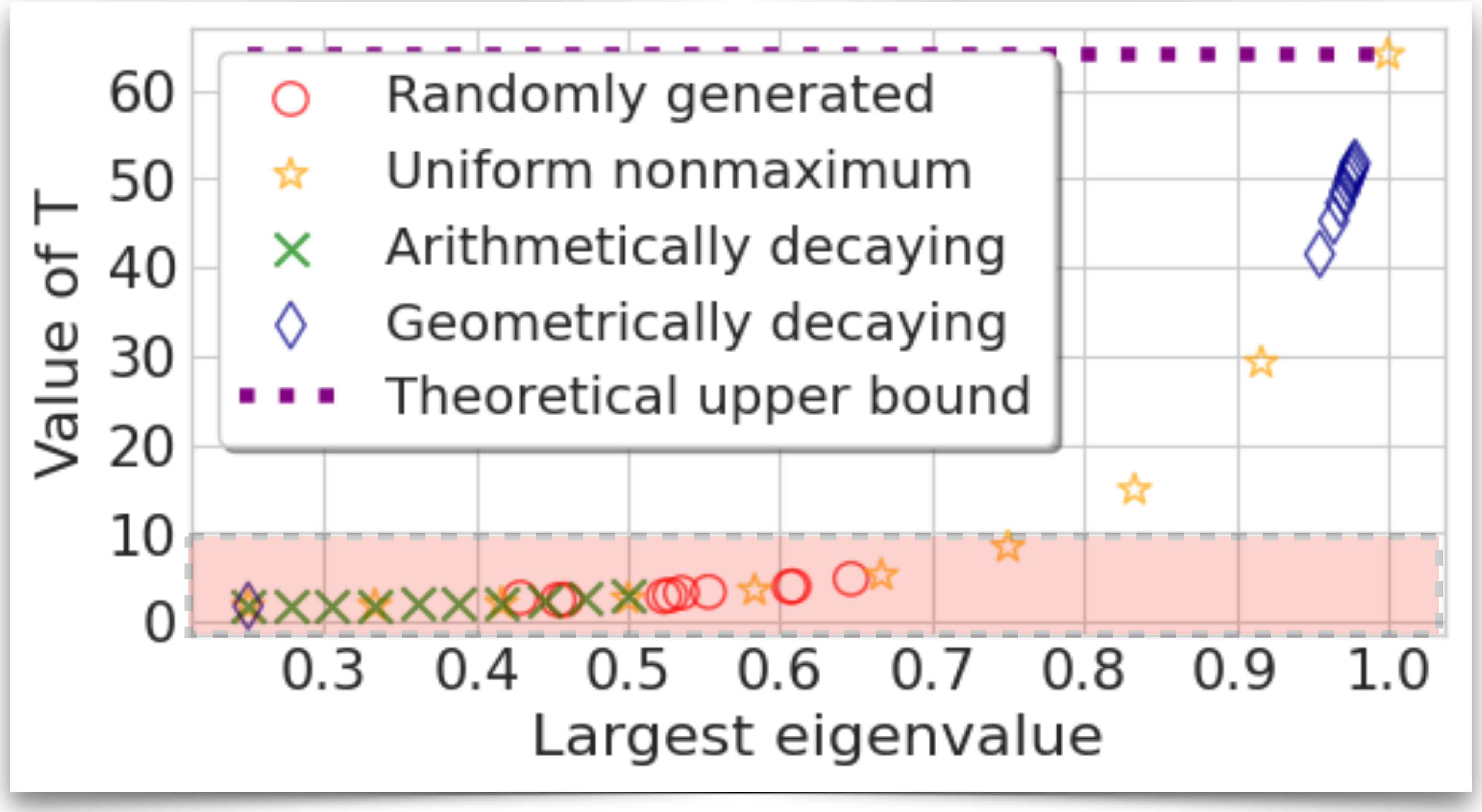


Reduction of required runs



$$T = \sum_{i=1}^{r} \frac{p_i(1 - p_i^k)(1 - p_i^r)}{(1 - p_i)^2}$$

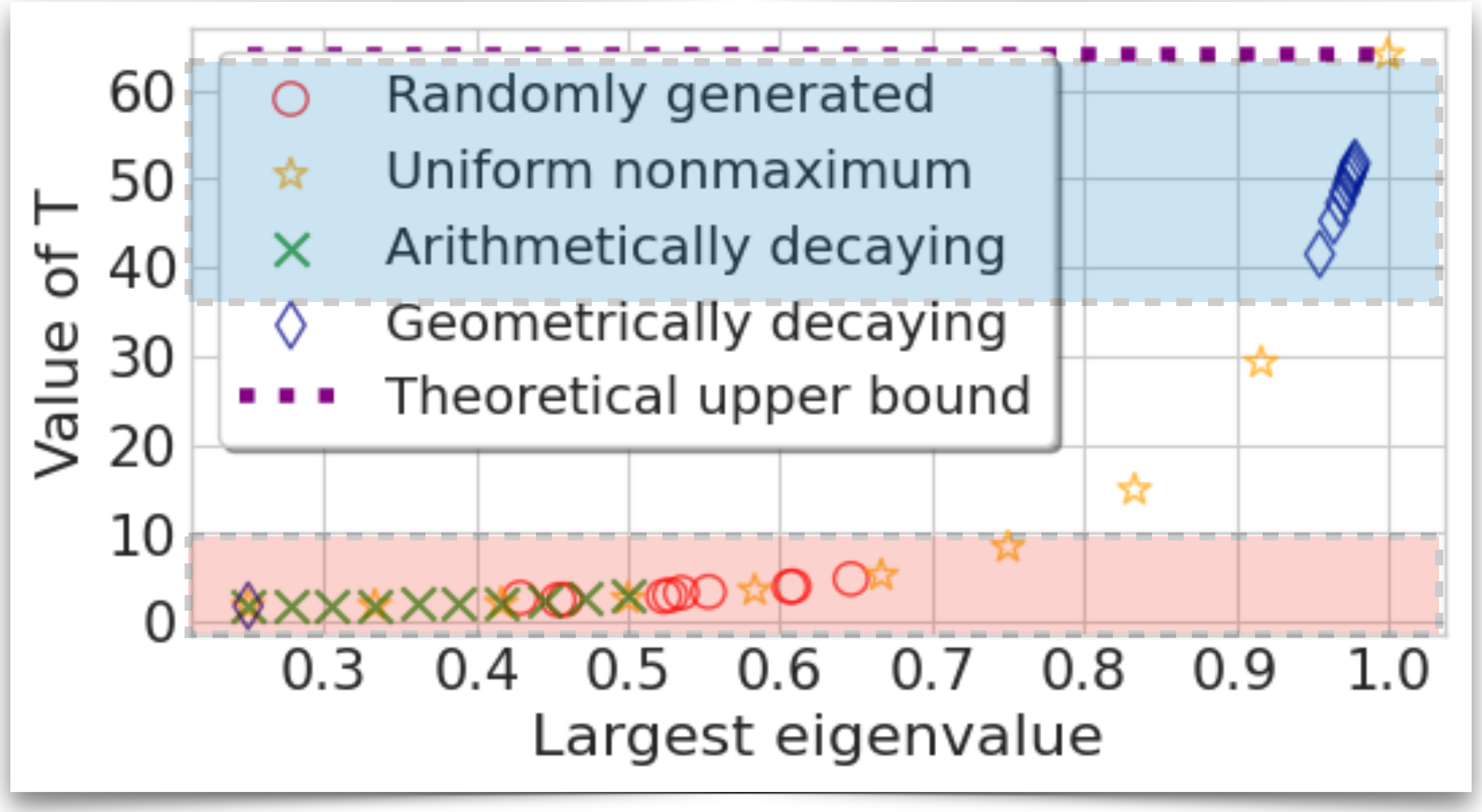
Reduction of required runs



~ constant

$$\mathcal{O}(1) \leq T = \sum_{i=1}^{r} \frac{p_i(1-p_i^k)(1-p_i^r)}{(1-p_i)^2}$$

Reduction of required runs

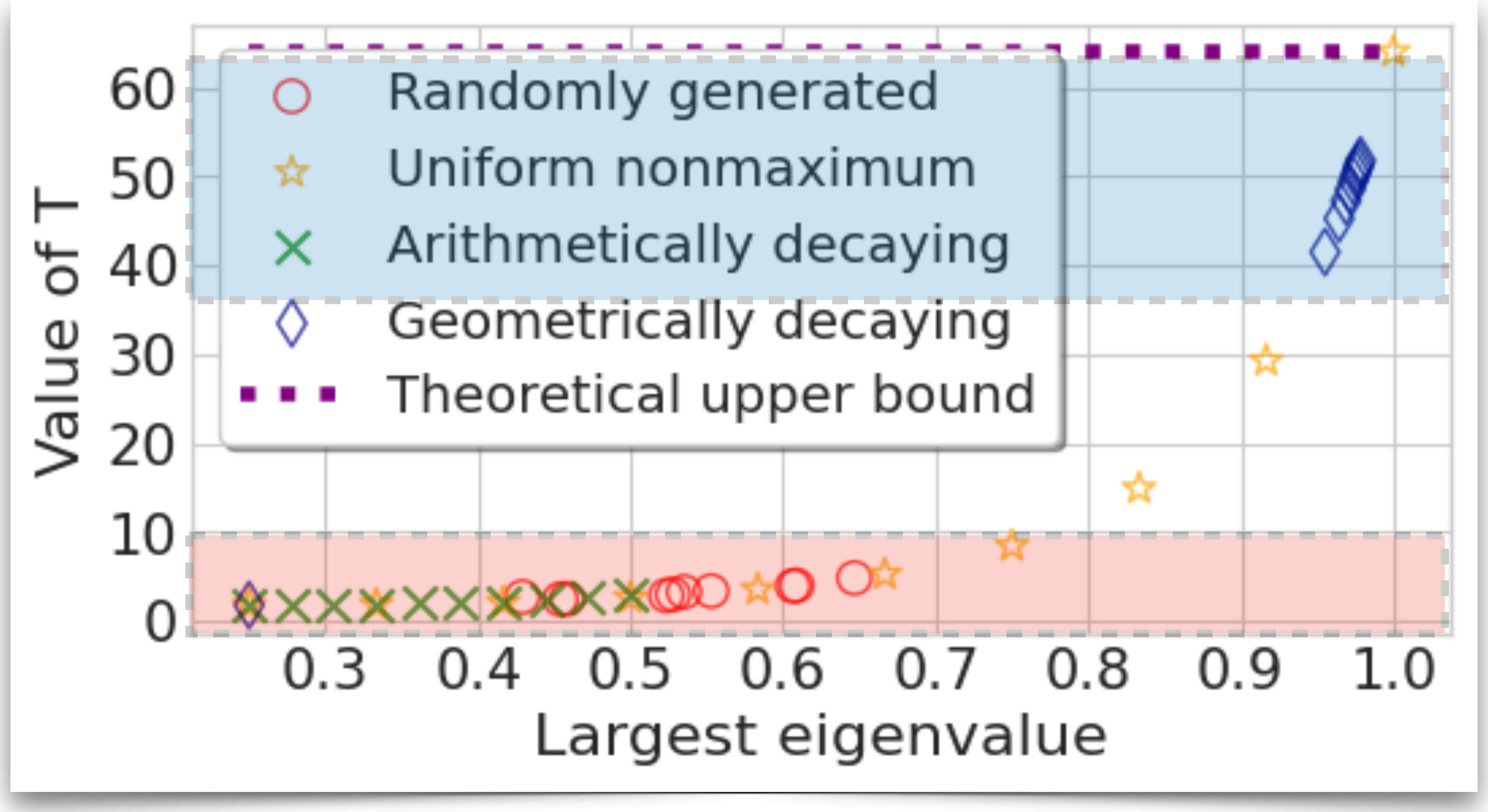


$\sim \mathcal{O}(kr)$

$$\mathcal{O}(1) \leq T = \sum_{i=1}^{r} \frac{p_i(1-p_i^k)(1-p_i^r)}{(1-p_i)^2} \leq kr$$

$\sim$ constant

Reduction of required runs



$\sim \mathcal{O}(kr)$

$\sim$ constant

$$\mathcal{O}(1) \leq T = \sum_{i=1}^{r} \frac{p_i(1-p_i^k)(1-p_i^r)}{(1-p_i)^2} \leq kr$$

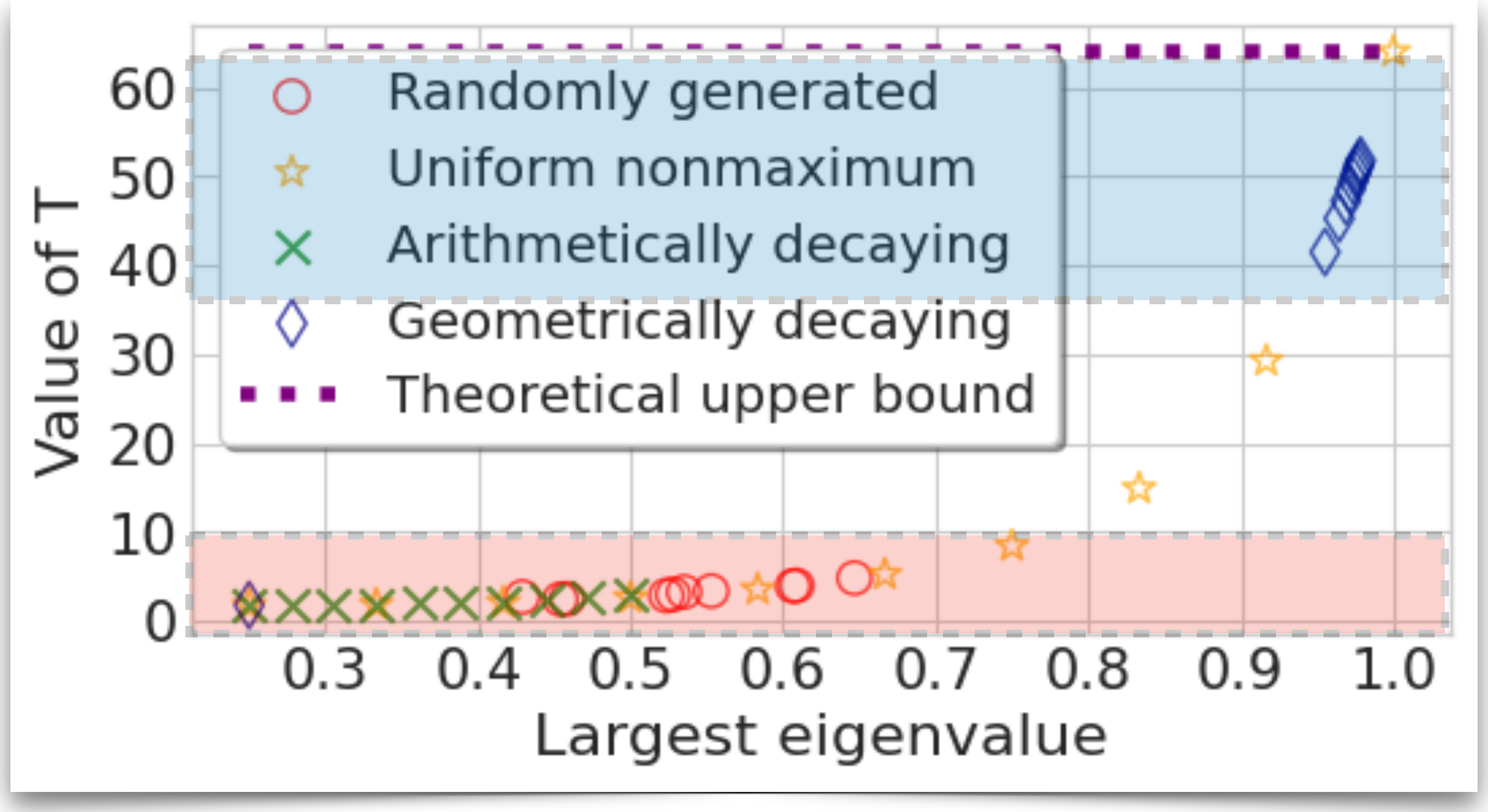
Corollary 1

To estimate $\mathrm{Tr}(\rho^i)$ for all $i \leq k$ within an additive error of ϵ and with a success probability of at least $1-\delta$, where $\delta \in (0,1)$, it is necessary to estimate each $\mathrm{Tr}(\rho^j)$ for $j \leq r$ within an additive error of ε_j, as defined in **Theorem 1.** This can be achieved by using

$$\mathcal{O}\left(\frac{T^2}{\epsilon^2}\ln^2 r \ln\left(\frac{1}{\delta}\right)\right)$$

runs on a constant-depth quantum circuit consisting of $\mathcal{O}(j)$ qubits and $\mathcal{O}(j)$ CSWAP operations. Here, T is defined in **Theorem 1.**

Reduction of required runs



$\sim \mathcal{O}(kr)$

$$\mathcal{O}(1) \leq T = \sum_{i=1}^{r} \frac{p_i(1-p_i^k)(1-p_i^r)}{(1-p_i)^2} \leq kr$$

$\sim$ constant

Corollary 1

To estimate $\mathrm{Tr}(\rho^i)$ for all $i \leq k$ within an additive error of ϵ and with a success probability of at least $1-\delta$, where $\delta \in (0,1)$, it is necessary to estimate each $\mathrm{Tr}(\rho^j)$ for $j \leq r$ within an additive error of ε_j, as defined in **Theorem 1.** This can be achieved by using

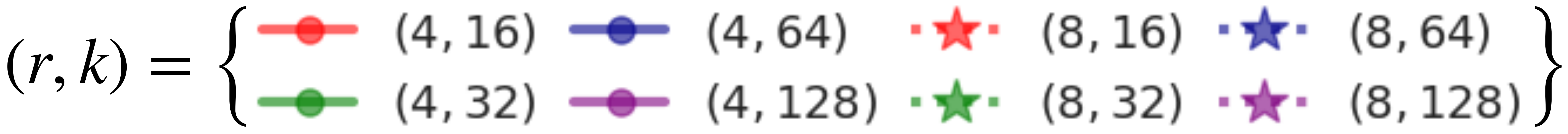
$$\mathcal{O}\left(\frac{T^2}{\epsilon^2}\ln^2 r \ln\left(\frac{1}{\delta}\right)\right)$$

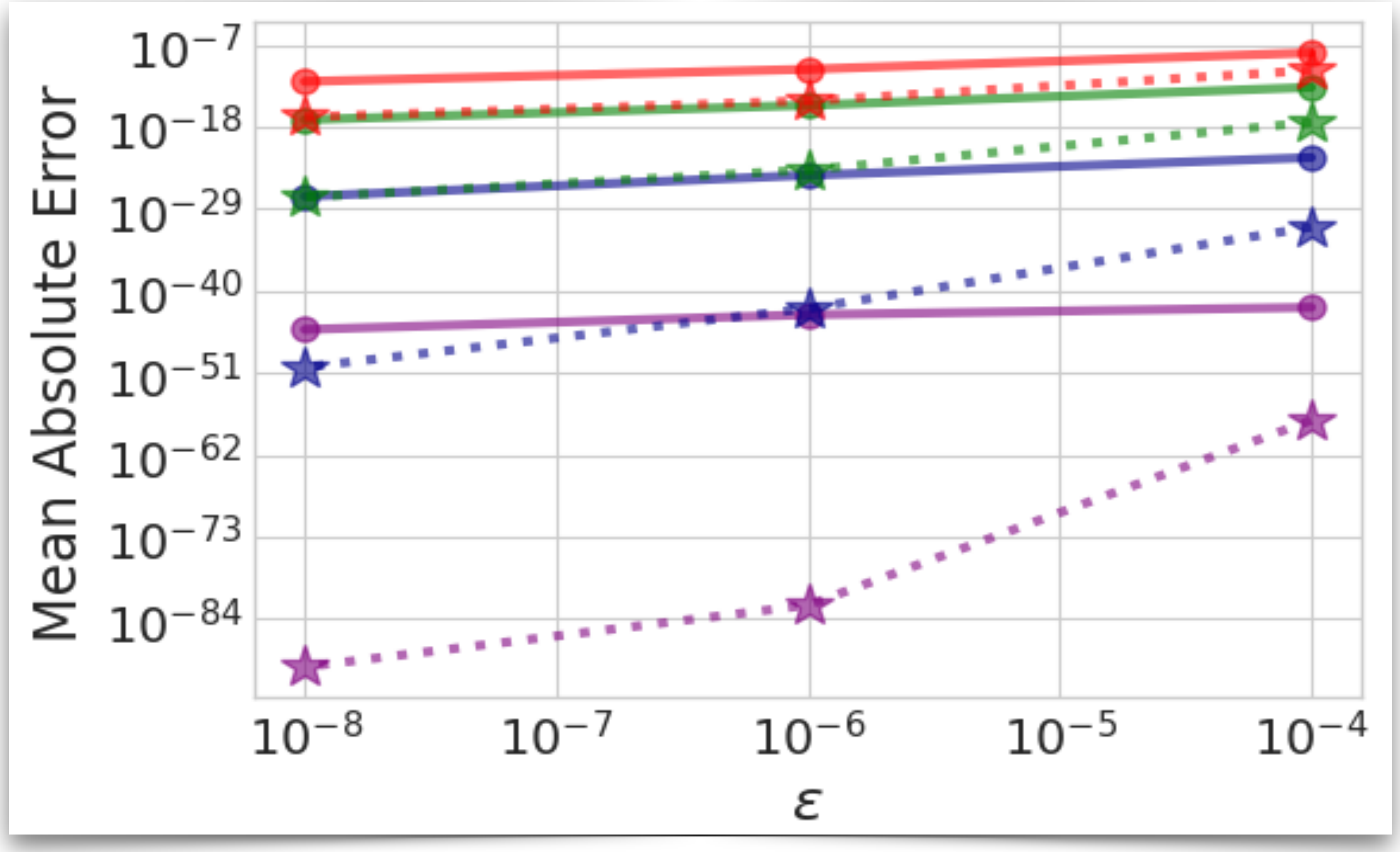
runs on a constant-depth quantum circuit consisting of $\mathcal{O}(j)$ qubits and $\mathcal{O}(j)$ CSWAP operations. Here, T is defined in **Theorem 1.**

The value of T is significantly less than the kr in most cases.

Especially when the state is *mixed* (e.g., the largest eigenvalue is small), the system is large (e.g., r is large), or k grows large, the expected advantage becomes more dramatic compared to the upper bound.

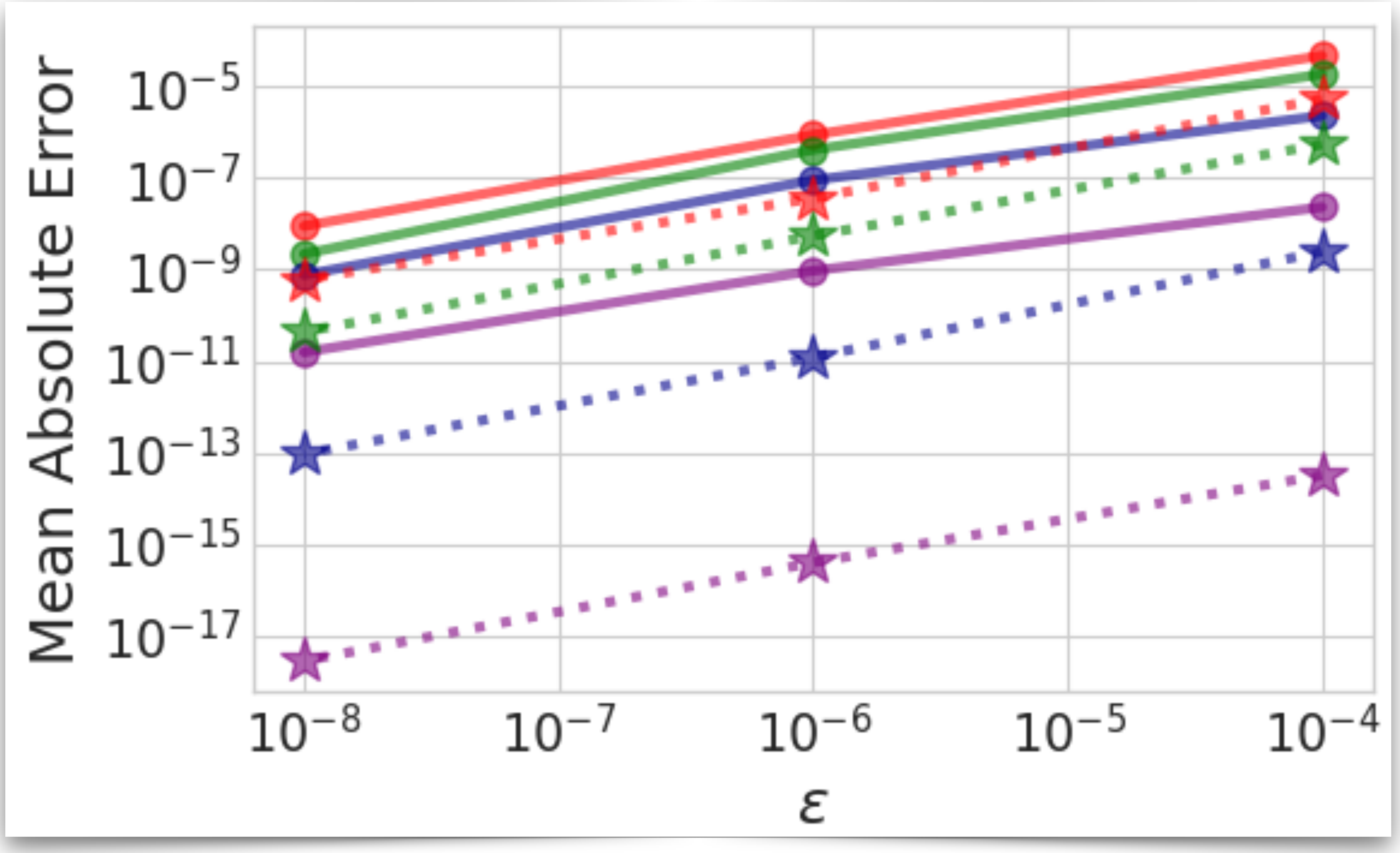
Numerical simulation





Arithmetically decaying

$T \sim \mathcal{O}(1)$



Geometrically decaying

$T \sim \mathcal{O}(kr)$

Overview

- Trace of powers & Literature review
- Mathematical intuitions
- Main results: algorithm, lemmas, theorems, corollaries
- Numerical simulations
- **Computational hardness of $\mathrm{TsallisQED}_q$ and $\mathrm{TsallisQEA}_q$**
- Applications
- Concluding remarks

Computational hardness of TsallisQED$_q$ and TsallisQEA$_q$

$$S_q(\rho) = \frac{1 - \mathrm{Tr}(\rho^q)}{q-1}$$

- **Quantum q-Tsallis Entropy Difference Problem (TsallisQED$_q$):**
  - Two polynomial-size quantum circuits, denoted as Q_0 and Q_1, prepare n-qubit quantum states ρ_0 and ρ_0.
  - The goal is to decide whether the difference between their q-Tsallis entropy values, $S_q(\rho_0) - S_q(\rho_1)$, is:
    - At least 0.001, or
    - At most −0.001.
  - The problem involves access to the descriptions of the quantum circuits Q_0 and Q_1.

[arXiv:2410.13559]

Computational hardness of TsallisQED$_q$ and TsallisQEA$_q$

$$S_q(\rho) = \frac{1 - \mathrm{Tr}(\rho^q)}{q - 1}$$

- **Quantum q-Tsallis Entropy Approximation Problem (TsallisQEA$_q$)**
  - Similar to TsallisQED$_q$, but involves only a single n-qubit quantum state ρ.
  - The task is to decide whether the difference between the q-Tsallis entropy of ρ and a known threshold $t(n)$, $S_q(\rho) - t(n)$is:
    - At least 0.001, or
    - At most -0.001.

[arXiv:2410.13559]

Computational hardness of TsallisQED$_q$ and TsallisQEA$_q$

- Given quantum query access to the state-preparation circuit of an n-qubit quantum state ρ, for any $q \geq 1 + \Omega(1)$, there is a quantum algorithm for estimating $S_q(\rho)$ to additive error 0.001 with query complexity $O(1)$.
- Moreover, if the description of the state-preparation circuit is of size poly(n), then the time complexity of the quantum algorithm is poly(n).

∴ For any $q \geq 1 + \Omega(1)$, TsallisQEC$_q$ and TsallisQEA$_q$ are in BQP.

Computational hardness of TsallisQED$_q$ and TsallisQEA$_q$

	$q = 1$	$1 < q \leq 1 + \frac{1}{n-1}$	$1+\Omega(1) \leq q \leq 2$	$q > 2$
TsallisQED$_q$	QSZK-complete [BASTS10]	QSZK-hard Theorem 1.2(2)	BQP-complete Theorem 1.1 and Theorem 1.2(1)	in BQP Theorem 1.1
TsallisQEA$_q$	NIQSZK-complete [BASTS10, CCKV08]	NIQSZK-hard* Theorem 1.2(2)	BQP-complete Theorem 1.1 and Theorem 1.2(1)	in BQP Theorem 1.1

Leading to hardness of approximating von Neumann entropy

because $S_q(\rho) \leq S(\rho)$, as long as **BQP** $\subsetneq$ **QSZK**

Purity Estimation = BQP-complete

QSZK $\subseteq$ QIP(2) $\subseteq$ PSPACE = IP

NIQSZK $\subseteq$ qq-QAM $\subseteq$ QIP(2) $\subseteq$ PSPACE = IP

Computational hardness of TsallisQED_q and TsallisQEA_q

- In terms of *quantitative bounds* on quantum query and sample complexities, **QSZK-hard or NIQSZK-hard correspond to rank-dependent complexities.**

Regime of q	Query Complexity		Sample Complexity	
	Upper Bound	Lower Bound	Upper Bound	Lower Bound
$q \geq 1 + \Omega(1)$	$O(1/\epsilon^{1+\frac{1}{q-1}})$ Theorem 3.2	$\Omega(1/\sqrt{\epsilon})$ Theorem 5.12	$\widetilde{O}(1/\epsilon^{3+\frac{2}{q-1}})$ Theorem 3.3	$\Omega(1/\epsilon)$ Theorem 5.14
$1 < q \leq 1 + \frac{1}{n-1}$	$\widetilde{O}(r/\epsilon^2)$ [WZL24]	$\Omega(r^{1/3})$ Theorem 5.13	$\widetilde{O}(r^2/\epsilon^5)$[7] [WZ24b]	$\Omega(r^{0.51-c})$[8] Theorem 5.15
$q = 1$	$\widetilde{O}(r/\epsilon^2)$[9] [WGL+24]	$\widetilde{\Omega}(\sqrt{r})$ [BKT20]	$\widetilde{O}(r^2/\epsilon^5)$[7] [WZ24b]	$\Omega(r/\epsilon)$ [WZ24b]

Computational hardness of $\mathrm{TsallisQED}_q$ and $\mathrm{TsallisQEA}_q$

The quantum query complexities are also extensively studied. For von Neumann entropy estimation, the dimension-dependence was studied in [GL20], the dependence on the reciprocal of the minimum non-zero eigenvalue of the quantum state was studied in [CLW20], the multiplicative error-dependence was studied in [GHS21], and the rank-dependence was studied in [WGL+24]. In [SLLJ24], they presented a rank-dependent estimator for the q-Tsallis entropy with integer q larger than the rank of quantum states. For Rényi entropy estimation, the query complexity was first studied in [SH21], the rank-dependence was studied in [WGL+24], and was later improved in [WZL24]. Other problems include tomography [vACGN23], and the estimations of fidelity and trace distance [WZC+23, WGL+24, GP22, WZ24a, Wan24].

In [GH20], the Quantum Entropy Difference Problem (with respect to von Neumann entropy) with shallow circuits was shown to have (conditional) hardness. The computational complexity of the space-bounded versions of the Quantum Entropy Difference Problem and Quantum State Distinguishability Problem were studied in [LGLW23].

[3] Calculate the estimated value Q_i $(i > r)$ by $Q_i = \sum_{\ell=1}^{r} (-1)^{\ell-1} b_\ell Q_{i-\ell} \sim \mathrm{Tr}(\rho^i)$.

Overview

- Trace of powers & Literature review
- Mathematical intuitions
- Main results: algorithm, lemmas, theorems, corollaries
- Numerical simulations
- Computational hardness of TsallisQED$_q$ and TsallisQEA$_q$
- **Applications**
- Concluding remarks

Applications

Nonlinear functions

Computation of the **partition function** and other **thermodynamical** variables for the systems with finite energy levels and finite # of non interacting particles

e.g.

$$g(x) = e^{\beta x},\ \beta \in \mathbb{R}$$

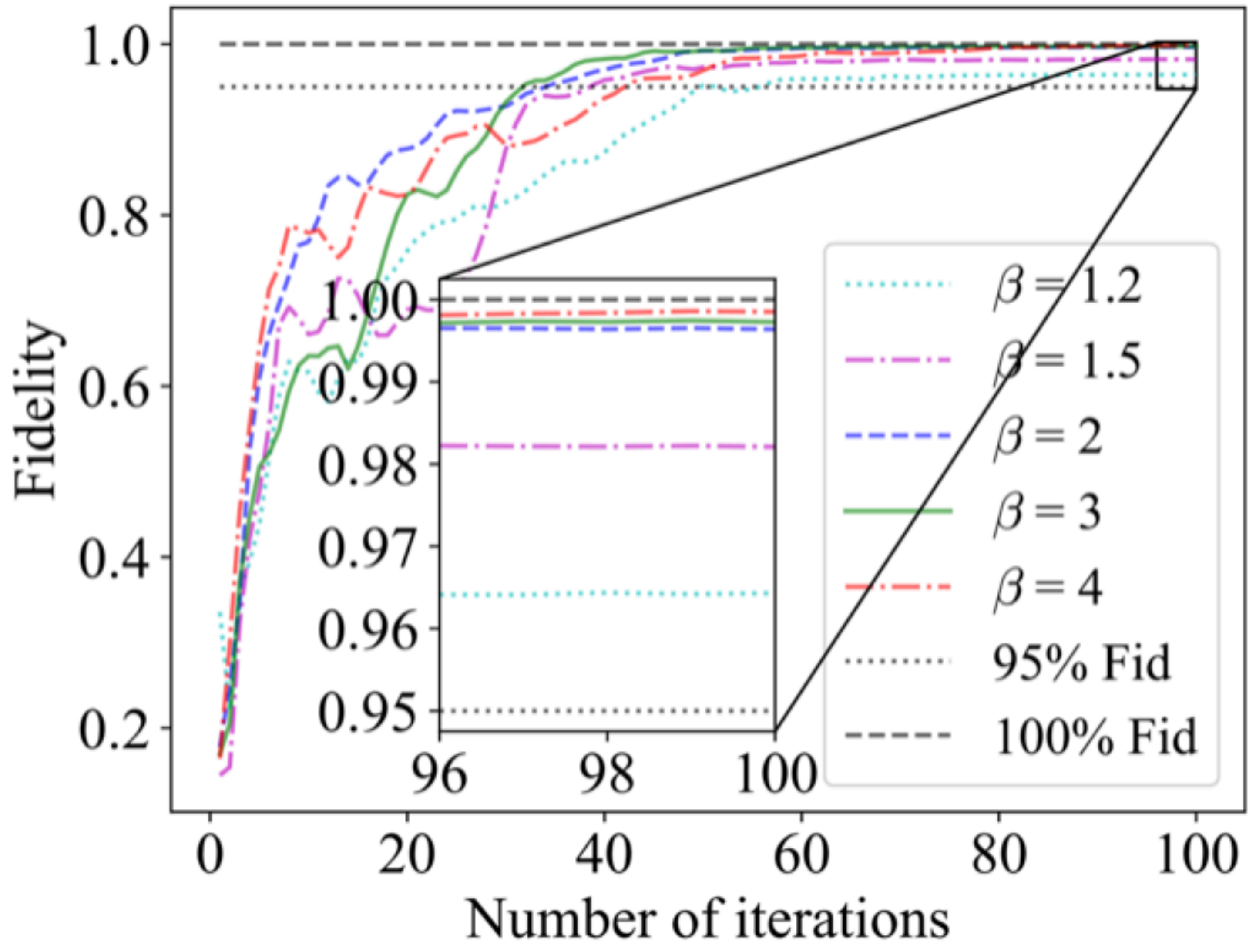
$$g(x) = (1+x)^{\alpha},\ \alpha \in \mathbb{R}^{+}$$

and more

Quantum Gibbs states

Variational quantum Gibbs state preparation with a **truncated Taylor series**

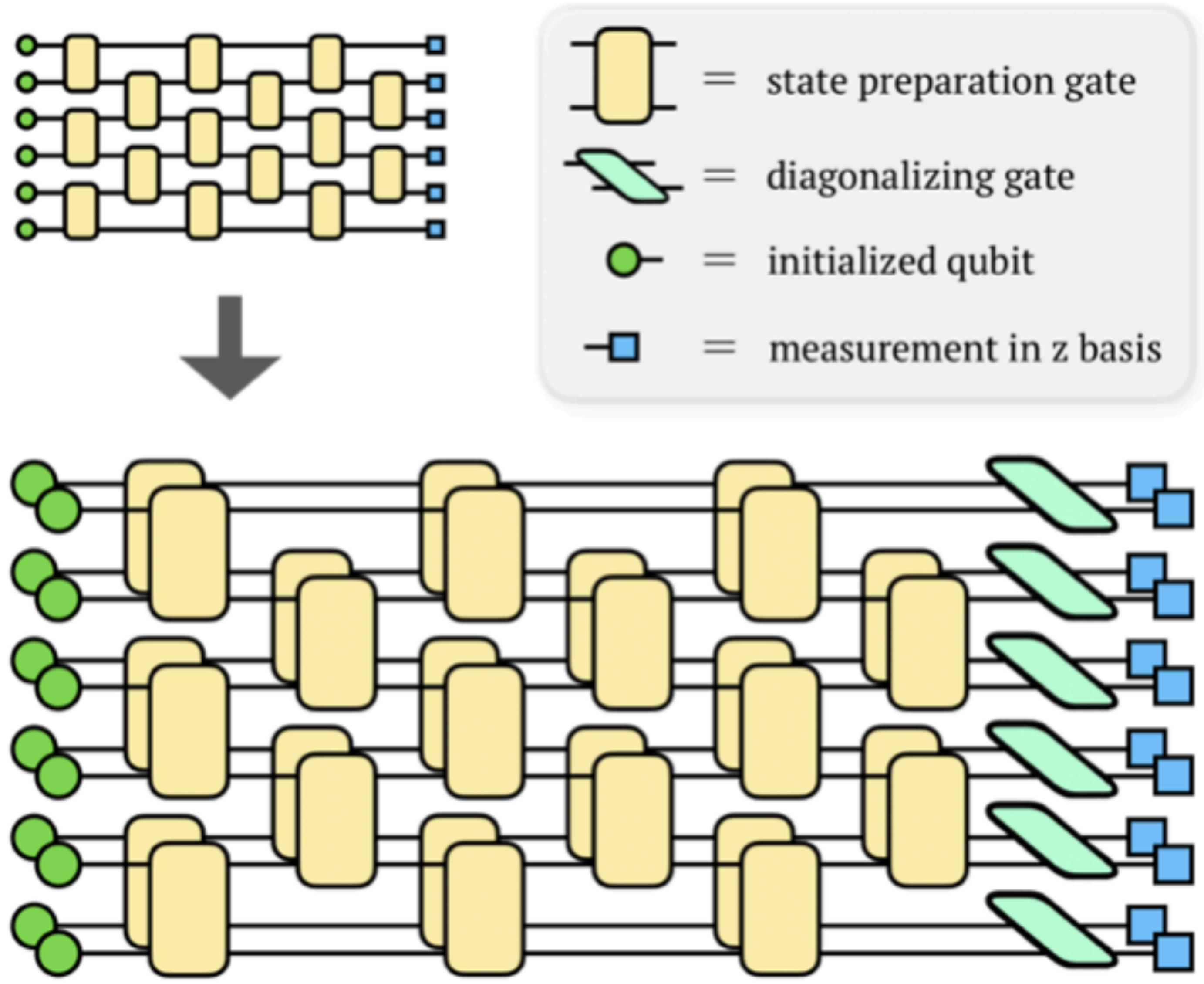
[Phys. Rev. Applied 16, 054035 (2021)]



Quantum error mitigation

Virtual distillation for quantum error mitigation

[Phys. Rev. X 11, 041036 (2021)]



Applications

Nonlinear functions

Computation of the **partition function** and other **thermodynamical** variables for the systems with finite energy levels and finite # of non interacting particles

e.g.

$$g(x) = e^{\beta x},\ \beta \in \mathbb{R}$$

$$g(x) = (1 + x)^{\alpha},\ \alpha \in \mathbb{R}^{+}$$

and more

Quantum Gibbs states

Variational quantum Gibbs state preparation with a **truncated Taylor series**

[Phys. Rev. Applied 16, 054035 (2021)]

Quantum error mitigation

Virtual distillation for quantum error mitigation

[Phys. Rev. X 11, 041036 (2021)]

Calculating the nonlinear functions of quantum state

Let ρ be a quantum state with rank r.

Suppose there exist $\epsilon > 0$ and a slowly-growing function C (as a function of m) such that $g : \mathbb{R} \to \mathbb{R}$ is approximated by a degree m polynomial $f(x) = \sum_{k=0}^{m} c_k x^k$ on the interval $[0,1]$, in the sense that $\sup_{x \in [0,1]} |g(x) - f(x)| < \frac{\epsilon}{2r}$, and $\sum_{k=0}^{m} |c_k| < C$.

Calculating the nonlinear functions of quantum state

Then estimating Tr$(g(\rho))$ within an ϵ additive error and with a success probability of at least $1-\delta$, where $\delta \in (0,1)$ requires

Calculating the nonlinear functions of quantum state

Then estimating $\mathrm{Tr}(g(\rho))$ within an ϵ additive error and with a success probability of at least $1-\delta$, where $\delta \in (0,1)$ requires

$\mathcal{O}\left(\frac{T^2C^2r^2}{\epsilon^2}\ln^2 r \ln\left(\frac{1}{\delta}\right)\right)$ copies of ρ and

Calculating the nonlinear functions of quantum state

Then estimating $\mathrm{Tr}(g(\rho))$ within an ϵ additive error and

with a success probability of at least $1-\delta$, where $\delta \in (0,1)$ requires

$\mathcal{O}\left(\frac{T^2C^2r^2}{\epsilon^2}\ln^2 r \ln\left(\frac{1}{\delta}\right)\right)$ copies of ρ and

$\mathcal{O}\left(\frac{T^2C^2r}{\epsilon^2}\ln^2 r \ln\left(\frac{1}{\delta}\right)\right)$ runs on a constant-depth quantum circuit

Calculating the nonlinear functions of quantum state

Then estimating $\mathrm{Tr}(g(\rho))$ within an ϵ additive error and with a success probability of at least $1-\delta$, where $\delta \in (0,1)$ requires

$\mathcal{O}\left(\frac{T^2C^2r^2}{\epsilon^2}\ln^2 r \ln\left(\frac{1}{\delta}\right)\right)$ copies of ρ and

$\mathcal{O}\left(\frac{T^2C^2r}{\epsilon^2}\ln^2 r \ln\left(\frac{1}{\delta}\right)\right)$ runs on a constant-depth quantum circuit

consisting of $\mathcal{O}(r)$ qubits and $\mathcal{O}(r)$ CSWAP operations.

Calculating the nonlinear functions of quantum state

Previous: $\mathcal{O}\left(\frac{C^2m^2}{\epsilon^2}\log\left(\frac{1}{\delta}\right)\right)$ copies of ρ and $\mathcal{O}\left(\frac{C^2m}{\epsilon^2}\log\left(\frac{1}{\delta}\right)\right)$ runs on a circuit consisted of $\mathcal{O}(1)$ depth, $\mathcal{O}(m)$ qubits and $\mathcal{O}(m)$ CSWAP operations.

Calculating the nonlinear functions of quantum state

Previous: $\mathcal{O}\left(\frac{C^2m^2}{\epsilon^2}\log\left(\frac{1}{\delta}\right)\right)$ copies of ρ and $\mathcal{O}\left(\frac{C^2m}{\epsilon^2}\log\left(\frac{1}{\delta}\right)\right)$ runs on a circuit consisted of $\mathcal{O}(1)$ depth, $\mathcal{O}(m)$ qubits and $\mathcal{O}(m)$ CSWAP operations.

Ours: $\mathcal{O}\left(\frac{T^2C^2r^2}{\epsilon^2}\ln^2 r\ln\left(\frac{1}{\delta}\right)\right)$ copies of ρ and $\mathcal{O}\left(\frac{T^2C^2r}{\epsilon^2}\ln^2 r\ln\left(\frac{1}{\delta}\right)\right)$ runs on a circuit consisted of $\mathcal{O}(1)$ depth, $\mathcal{O}(r)$ qubits and $\mathcal{O}(r)$ CSWAP operations.

Calculating the nonlinear functions of quantum state

Previous: $\mathcal{O}\left(\frac{C^2m^2}{\epsilon^2}\log\left(\frac{1}{\delta}\right)\right)$ copies of ρ and $\mathcal{O}\left(\frac{C^2m}{\epsilon^2}\log\left(\frac{1}{\delta}\right)\right)$ runs on a circuit consisted of $\mathcal{O}(1)$ depth, $\mathcal{O}(m)$ qubits and $\mathcal{O}(m)$ CSWAP operations.

Typically, $m \gg r$,

so our enhanced theorem offers advantages for estimating $g(\rho)$.

Ours: $\mathcal{O}\left(\frac{T^2C^2r^2}{\epsilon^2}\ln^2 r\ln\left(\frac{1}{\delta}\right)\right)$ copies of ρ and $\mathcal{O}\left(\frac{T^2C^2r}{\epsilon^2}\ln^2 r\ln\left(\frac{1}{\delta}\right)\right)$ runs on a circuit consisted of $\mathcal{O}(1)$ depth, $\mathcal{O}(r)$ qubits and $\mathcal{O}(r)$ CSWAP operations.

Calculating the nonlinear functions of quantum state

Previous: $\mathcal{O}\left(\frac{C^2m^2}{\epsilon^2}\log\left(\frac{1}{\delta}\right)\right)$ copies of ρ and $\mathcal{O}\left(\frac{C^2m}{\epsilon^2}\log\left(\frac{1}{\delta}\right)\right)$ runs on a circuit consisted of $\mathcal{O}(1)$ depth, $\mathcal{O}(m)$ qubits and $\mathcal{O}(m)$ CSWAP operations.

Typically, $m \gg r$,

so our enhanced theorem offers advantages for estimating $g(\rho)$.

Ours: $\mathcal{O}\left(\frac{T^2C^2r^2}{\epsilon^2}\ln^2 r \ln\left(\frac{1}{\delta}\right)\right)$ copies of ρ and $\mathcal{O}\left(\frac{T^2C^2r}{\epsilon^2}\ln^2 r \ln\left(\frac{1}{\delta}\right)\right)$ runs on a circuit consisted of $\mathcal{O}(1)$ depth, $\mathcal{O}(r)$ qubits and $\mathcal{O}(r)$ CSWAP operations.

When $g(x) = e^{\beta x}$, C becomes $e^{|\beta|}$. We can efficiently estimate $\mathrm{Tr}(e^{\beta\rho})$ which has applications in thermodynamics and the density exponentiation algorithm.

Applications

Nonlinear functions

Computation of the **partition function** and other **thermodynamical** variables for the systems with finite energy levels and finite # of non interacting particles

e.g.

$$g(x) = e^{\beta x},\ \beta \in \mathbb{R}$$

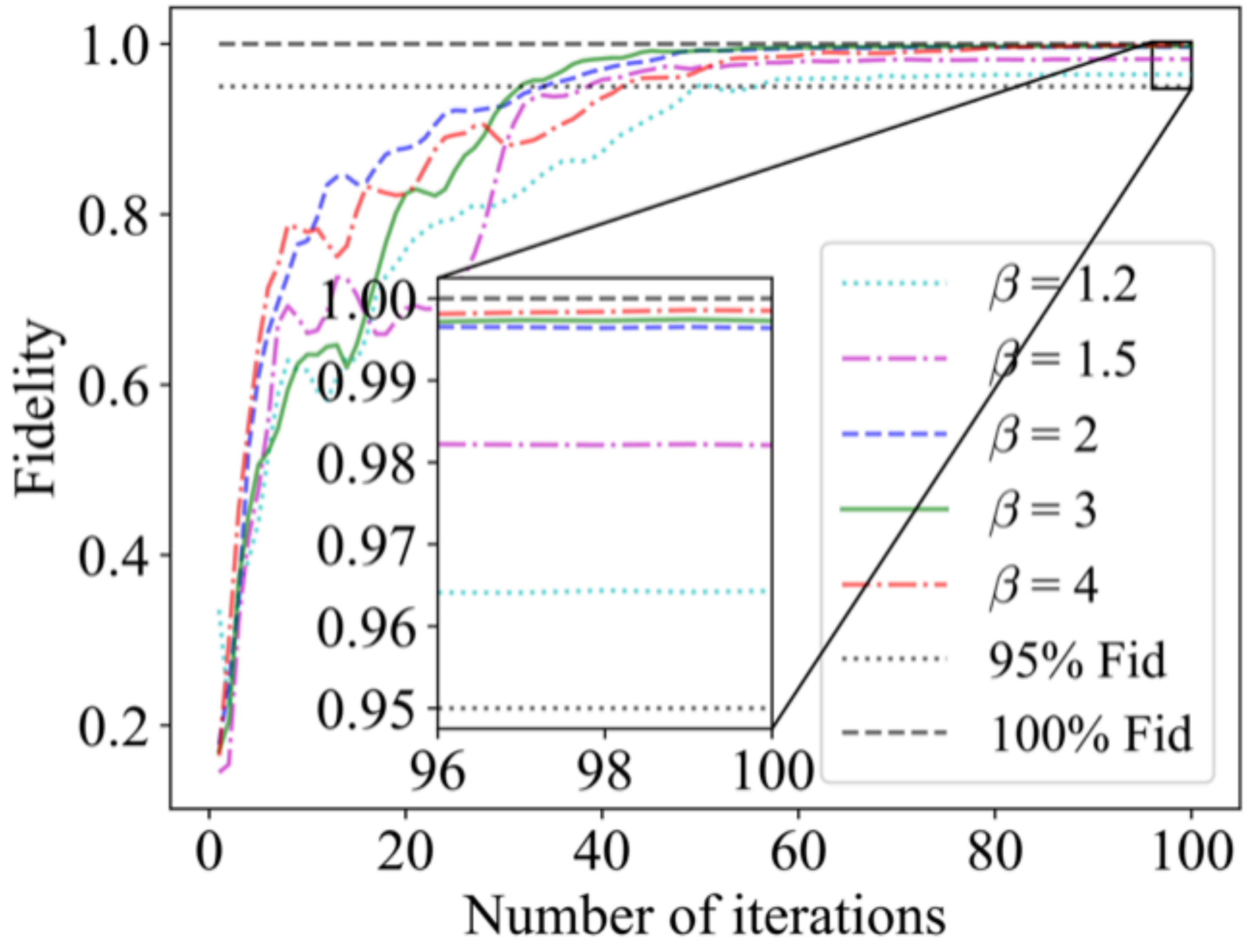
$$g(x) = (1+x)^{\alpha},\ \alpha \in \mathbb{R}^{+}$$

and more

Quantum Gibbs states

Variational quantum Gibbs state preparation with a **truncated Taylor series**

[Phys. Rev. Applied 16, 054035 (2021)]



Quantum error mitigation

Virtual distillation for quantum error mitigation

[Phys. Rev. X 11, 041036 (2021)]

Quantum Gibbs state preparation

The truncated Taylor series $S_k(\rho) = \sum_{i=1}^{k} \mathrm{Tr}\left(\left(\rho - I\right)^k \rho\right)$

is used as the **cost function for variational quantum Gibbs state preparation.**

Quantum Gibbs state preparation

The truncated Taylor series $S_k(\rho) = \sum_{i=1}^{k} \mathrm{Tr}\left(\left(\rho - I\right)^k \rho\right)$

is used as the **cost function for variational quantum Gibbs state preparation.**

It is shown that the fidelity $F\left(\rho(\theta_0), \rho_G\right)$ between the optimized state $\rho(\theta_0)$ and the Gibbs state ρ_G is bounded by $F(\rho(\theta_0), \rho_G) \geq 1 - \sqrt{2\left(\beta\epsilon + \frac{2r}{k+1}(1-\Delta)^{k+1}\right)}$

where β is the inverse temperature of the system,

and Δ is a constant that satisfies $-\Delta \ln(\Delta) < \frac{1}{k+1}(1-\Delta)^{k+1}$.

Quantum Gibbs state preparation

By using the inequality $D(\rho(\theta_0), \rho_G) < \sqrt{1 - F\left(\rho\left(\theta_0\right), \rho_G\right)}$, to achieve

$T\left(\rho(\theta_0), \rho_G\right) < \epsilon$, we need to set $k = \mathcal{O}\left(\,\cdot\,\right)$, where T is the trace distance.

Quantum Gibbs state preparation

By using the inequality $D(\rho(\theta_0), \rho_G) < \sqrt{1 - F\left(\rho\left(\theta_0\right), \rho_G\right)}$, to achieve $T\left(\rho(\theta_0), \rho_G\right) < \epsilon$, we need to set $k = \mathcal{O}\,(\,\cdot\,)$, where T is the trace distance.

Previous: $k = \mathcal{O}\left(\frac{r}{\epsilon^4}\right)$ qubits and CSWAP operations are needed.

Quantum Gibbs state preparation

By using the inequality $D(\rho(\theta_0), \rho_G) < \sqrt{1 - F\left(\rho\left(\theta_0\right), \rho_G\right)}$, to achieve $T\left(\rho(\theta_0), \rho_G\right) < \epsilon$, we need to set $k = \mathcal{O}\left(\,\cdot\,\right)$, where T is the trace distance.

Previous: $k = \mathcal{O}\left(\frac{r}{\epsilon^4}\right)$ qubits and CSWAP operations are needed.

Ours: $k = \mathcal{O}\left(r\right)$ qubits and CSWAP operations are needed.

Quantum Gibbs state preparation

By using the inequality $D(\rho(\theta_0), \rho_G) < \sqrt{1 - F\left(\rho\left(\theta_0\right), \rho_G\right)}$, to achieve $T\left(\rho(\theta_0), \rho_G\right) < \epsilon$, we need to set $k = \mathcal{O}(\,\cdot\,)$, where T is the trace distance.

Previous: $k = \mathcal{O}\left(\frac{r}{\epsilon^4}\right)$ qubits and CSWAP operations are needed.

Significantly reduces the number of qubits and CSWAP operations.

* Independent of the desired error bound ϵ.

Ours: $k = \mathcal{O}(r)$ qubits and CSWAP operations are needed.

Applications

Nonlinear functions

Computation of the **partition function** and other **thermodynamical** variables for the systems with finite energy levels and finite # of non interacting particles

e.g.

$$g(x) = e^{\beta x},\ \beta \in \mathbb{R}$$

$$g(x) = (1+x)^{\alpha},\ \alpha \in \mathbb{R}^{+}$$

and more

Quantum Gibbs states

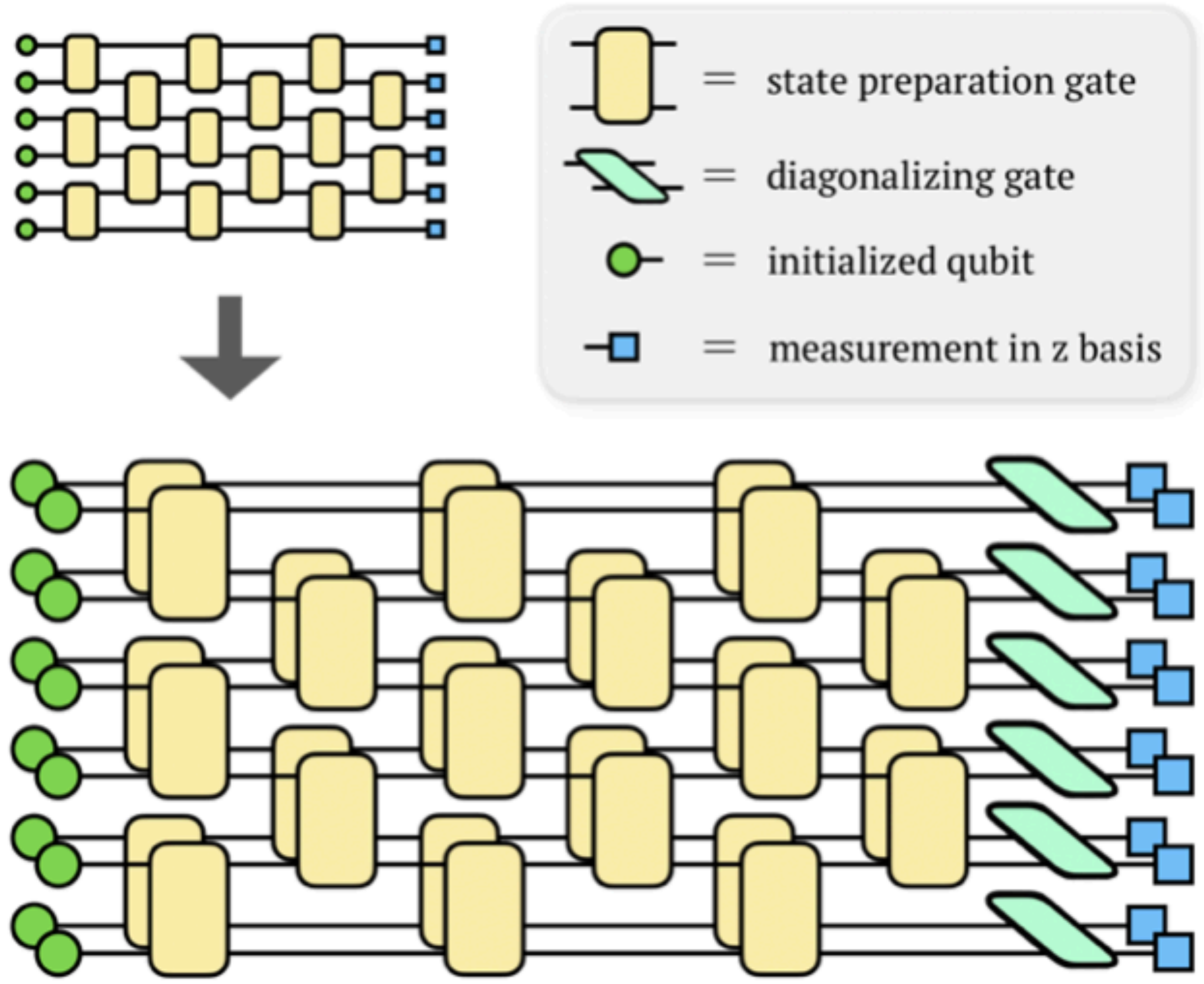
Variational quantum Gibbs state preparation with a **truncated Taylor series**

[Phys. Rev. Applied 16, 054035 (2021)]

Quantum error mitigation

Virtual distillation for quantum error mitigation

[Phys. Rev. X 11, 041036 (2021)]



Quantum error mitigation

The expected value of a Hermitian operator M is given by $\langle M \rangle = \mathrm{Tr}(M |\psi\rangle\langle\psi|)$.

Due to noise, this value becomes $\langle M \rangle_{\text{noise}} = \mathrm{Tr}(M\rho) \neq \langle M \rangle$.

Quantum error mitigation

The expected value of a Hermitian operator M is given by $\langle M \rangle = \mathrm{Tr}(M|\psi\rangle\langle\psi|)$.

Due to noise, this value becomes $\langle M \rangle_{\text{noise}} = \mathrm{Tr}(M\rho) \neq \langle M \rangle$.

Virtual distillation protocol offers a method to address this issue.

The protocol involves using collective measurements of k copies of the mixed state ρ to suppress incoherent errors.

Quantum error mitigation

The expected value of a Hermitian operator M is given by $\langle M \rangle = \mathrm{Tr}(M|\psi\rangle\langle\psi|)$.

Due to noise, this value becomes $\langle M \rangle_{\text{noise}} = \mathrm{Tr}(M\rho) \neq \langle M \rangle$.

Virtual distillation protocol offers a method to address this issue.

The protocol involves using collective measurements of k copies of the mixed state ρ to suppress incoherent errors.

This approach approximates the error-free expectation value $\langle M \rangle_{\text{vd}}^{(k)} = \dfrac{\mathrm{Tr}(M\rho^k)}{\mathrm{Tr}(\rho^k)}$,

where k denotes the number of copies used.

Quantum error mitigation

$$\langle M \rangle_{\mathrm{vd}}^{(k)} = \frac{\mathrm{Tr}(M\rho^k)}{\mathrm{Tr}(\rho^k)}$$

Quantum error mitigation

$$\langle M \rangle_{\mathrm{vd}}^{(k)} = \frac{\mathrm{Tr}(M\rho^k)}{\mathrm{Tr}(\rho^k)}$$

Corollary 1

To estimate $\mathrm{Tr}(\rho^i)$ for all $i \leq k$ within an additive error of ϵ and with a success probability of at least $1 - \delta$, where $\delta \in (0,1)$, it is necessary to estimate each $\mathrm{Tr}(\rho^j)$ for $j \leq r$ within an additive error of ε_j, as defined in **Theorem 1.** This can be achieved by using

$$\mathcal{O}\left(\frac{T^2}{\epsilon^2} \ln^2 r \ln\left(\frac{1}{\delta}\right)\right)$$

runs on a constant-depth quantum circuit consisting of $\mathcal{O}(j)$ qubits and $\mathcal{O}(j)$ CSWAP operations. Here, T is defined in **Theorem 1.**

Quantum error mitigation

$$\langle M \rangle^{(k)}_{\text{vd}} = \frac{\mathrm{Tr}(M\rho^k)}{\mathrm{Tr}(\rho^k)}$$

Corollary 2

Suppose there is an efficient decomposition $M = \sum_{\ell=1}^{N_M} a_\ell P_\ell$, where $a_\ell \in \mathbb{R}$ and $P_\ell = \sigma_{\ell_1} \otimes \cdots \otimes \sigma_{\ell_n}$ are tensor products of Pauli operators $\sigma_{\ell_1}, \ldots, \sigma_{\ell_n} \in \{\sigma_x, \sigma_y, \sigma_z, I\}$. The quantity $\sum_{\ell=1}^{N_M} |a_\ell| = \mathcal{O}(c)$ is bounded by a constant c.

To estimate $\mathrm{Tr}(M\rho^i)$ for all $i \leq k$ within an additive error of ϵ and with a success probability of at least $1-\delta$, where $\delta \in (0,1)$, it is necessary to estimate each $\mathrm{Tr}(M\rho^j)$ for $j \leq r$ within an additive error of $\varepsilon_{j,M}$.

This can be achieved by using $\mathcal{O}\left(\frac{c^2 N_M}{\epsilon^2} \ln\left(\frac{1}{\delta}\right)\right)$ runs on a constant-depth quantum circuit consisting of $\mathcal{O}(j)$ qubits and $\mathcal{O}(j)$ CSWAP operations, and estimating each $\mathrm{Tr}(\rho^{j'})$ for $j' \leq r$ within an additive error of $\varepsilon_{j'}$, by using $\mathcal{O}\left(\frac{T^2}{\epsilon^2} \ln^2 r \ln\left(\frac{1}{\delta}\right)\right)$ runs on a constant-depth quantum circuit consisting of $\mathcal{O}(j')$ qubits and $\mathcal{O}(j')$ CSWAP operations. Here, $\varepsilon_{j,M}$, $\varepsilon_{j'}$ and T are defined in **Theorem 2.**

Corollary 1

To estimate $\mathrm{Tr}(\rho^i)$ for all $i \leq k$ within an additive error of ϵ and with a success probability of at least $1-\delta$, where $\delta \in (0,1)$, it is necessary to estimate each $\mathrm{Tr}(\rho^j)$ for $j \leq r$ within an additive error of ε_j, as defined in **Theorem 1.** This can be achieved by using

$$\mathcal{O}\left(\frac{T^2}{\epsilon^2} \ln^2 r \ln\left(\frac{1}{\delta}\right)\right)$$

runs on a constant-depth quantum circuit consisting of $\mathcal{O}(j)$ qubits and $\mathcal{O}(j)$ CSWAP operations. Here, T is defined in **Theorem 1.**

Overview

- Trace of powers & Literature review
- Mathematical intuitions
- Main results: algorithm, lemmas, theorems, corollaries
- Numerical simulations
- Computational hardness of $\mathrm{TsallisQED}_q$ and $\mathrm{TsallisQEA}_q$
- Applications
- **Concluding remarks**

Concluding remarks

- Our main contribution lies in proving that the **error increases linearly at most** when applying the Newton-Girard method with a recursive strategy.
- We also generalize the result to traces of powers with observables M, which are represented as $\mathrm{Tr}(M\rho^k)$.
- Our work can enhance any previous algorithms, including $\mathrm{Tr}(\rho^k)$ and/or $\mathrm{Tr}(M\rho^k)$.
- We can estimate the trace of powers with $\mathcal{O}(1)$-depth, $\mathcal{O}(r)$-width, and only $\mathcal{O}(r)$ -CSWAP operations.
- Our method also provides advantages in copy complexity when estimating the trace of large powers **with low-rank states or sufficiently mixed states.**

Future work

- It remains open for future work to find more applications that can take advantage of our work.
- Generalizing this result to multivariate trace estimation, or even Tr$(\rho^k \sigma^l)$, can open up more possibilities, such as calculating functions that satisfy the data-processing inequality under unital quantum channels, which can be an alternative tool for distance measures.
- Tightening the bounds on Theorems 1 and 2 is an interesting future research topic.
- **More algorithms for low-rank quantum states.**

Thank you for listening!

Junseo Lee (harris.junseo@gmail.com)

arXiv:2408.00314