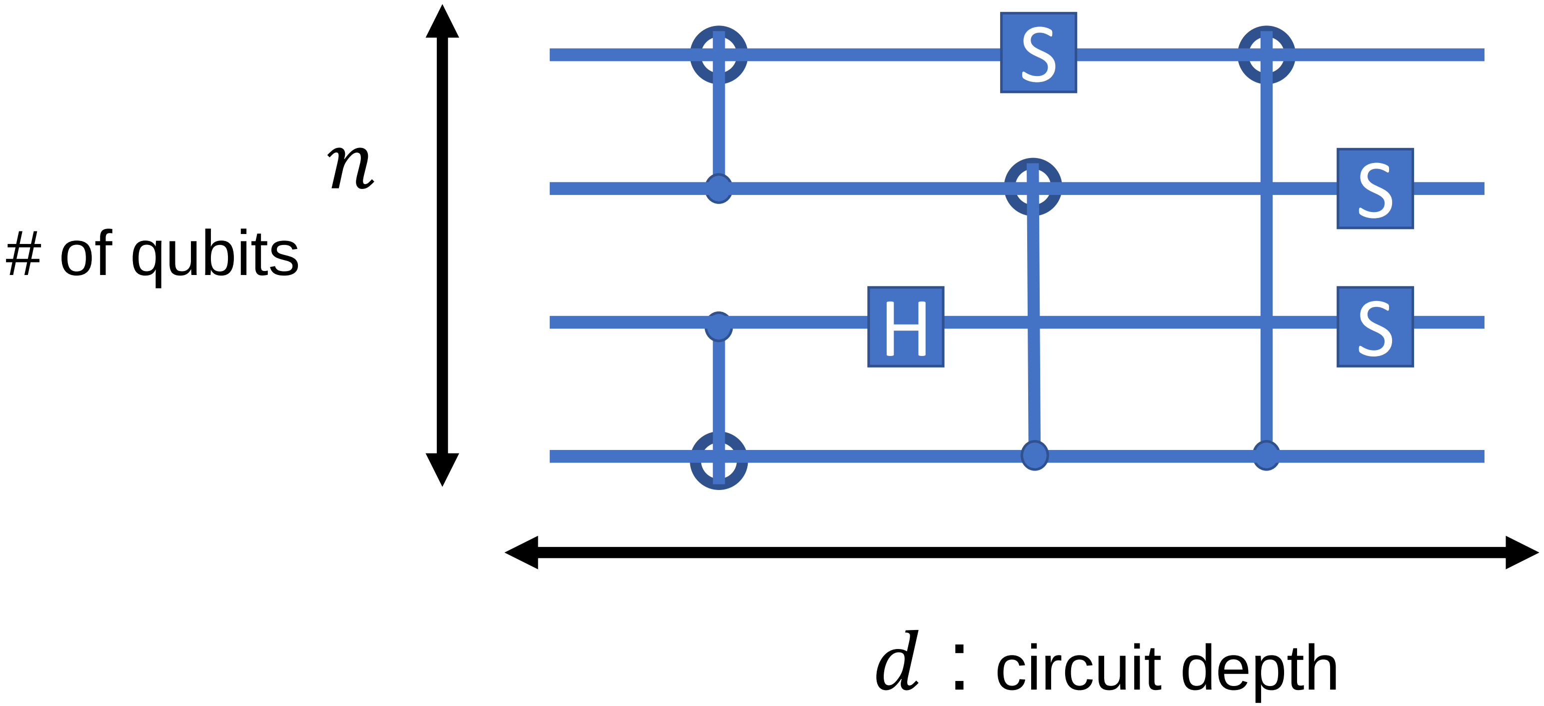
Classical algorithms for quantum mean values

Sergey Bravyi
David Gosset
Ramis Movassagh

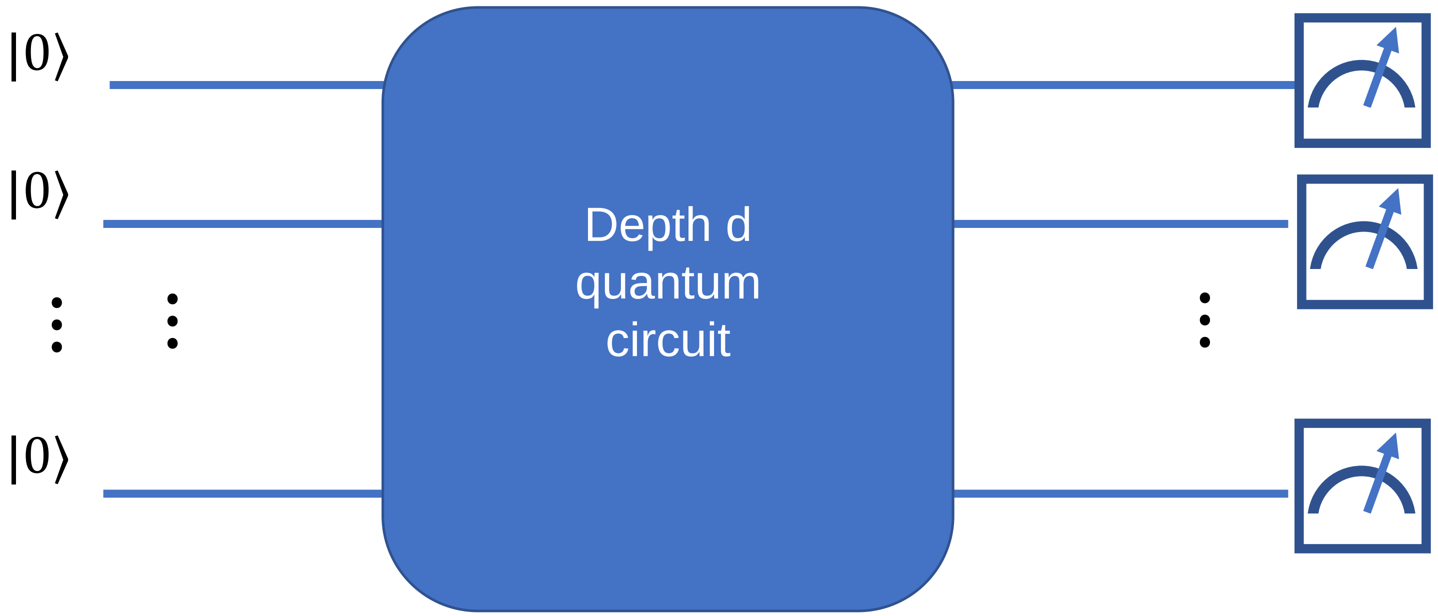
arXiv:1909.11485

Shallow quantum circuits



We are interested in circuits with depth $d = O(1)$.

What are shallow quantum circuits good for?

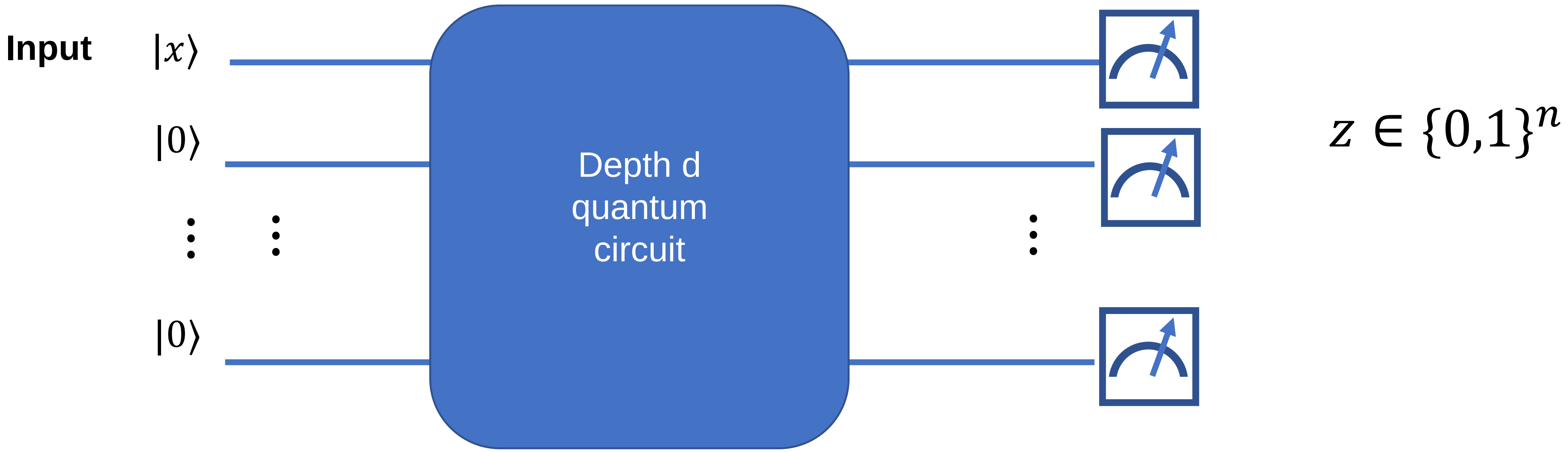


$z \in \{0,1\}^n$

Sample from classically inaccessible probability distributions

[Terhal Divincenzo 2002]
[Gao et al 17]
[Bermejo-Vega et al. 17]

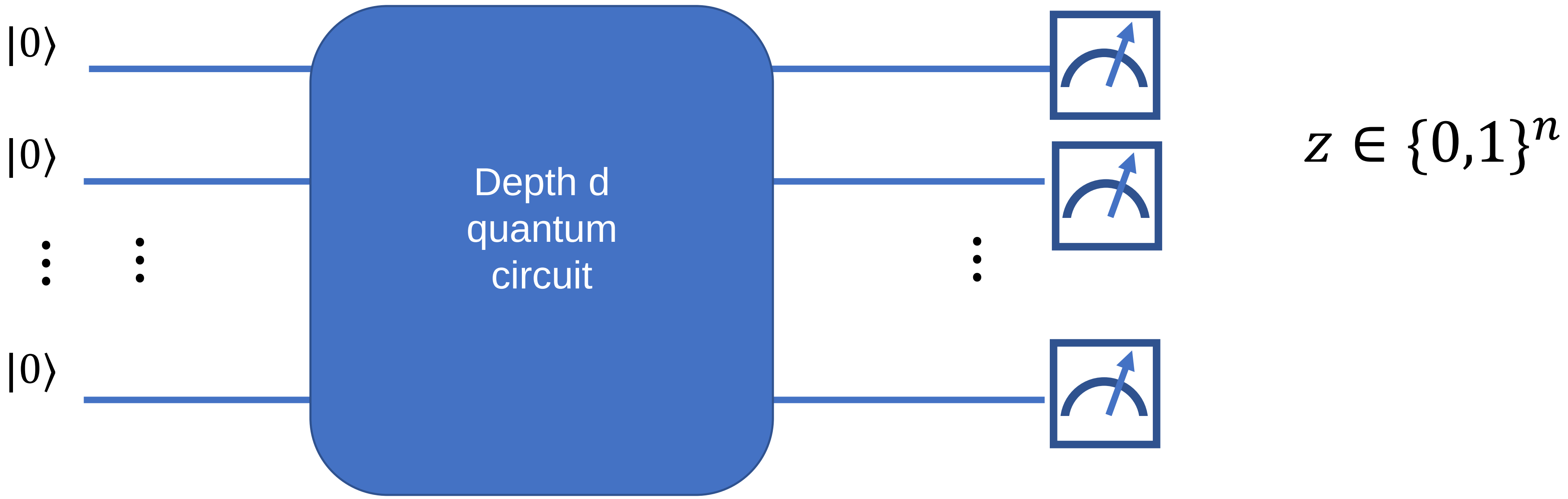
What are shallow quantum circuits good for?



Solve certain linear algebra problems faster than classical algorithms

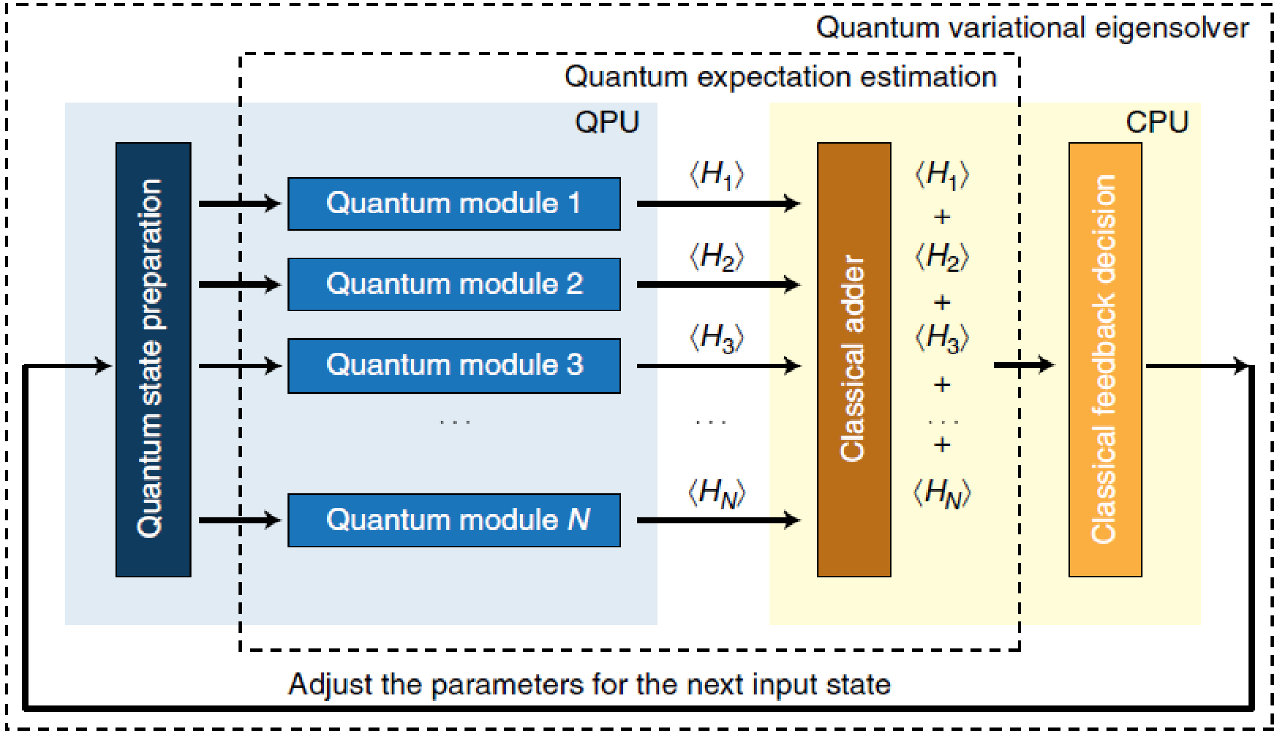
[Bravyi, G., Koenig 18]
[Bene Watts, Kothari, Schaeffer, Tal 19]
[Bravyi, G., Koenig, Tomamichel 19]

What are shallow quantum circuits good for?



…Anything else?

Variational Quantum Algorithms



"Variational Quantum Eigensolver", from
[Peruzzo et al. 2013]

Variational algorithms have recently attracted interest due to their potential for near-term implementations.

Variational Quantum Algorithms

Goal: compute the ground energy of a given Hamiltonian.

$$H = \sum_i P_i \qquad\qquad E_{min} = \min_{\psi} \langle \psi | H | \psi \rangle$$

Variational Quantum Algorithms

Goal: compute the ground energy of a given Hamiltonian.

$$H = \sum_i P_i \qquad\qquad E_{min} = \min_{\psi} \langle \psi | H | \psi \rangle$$

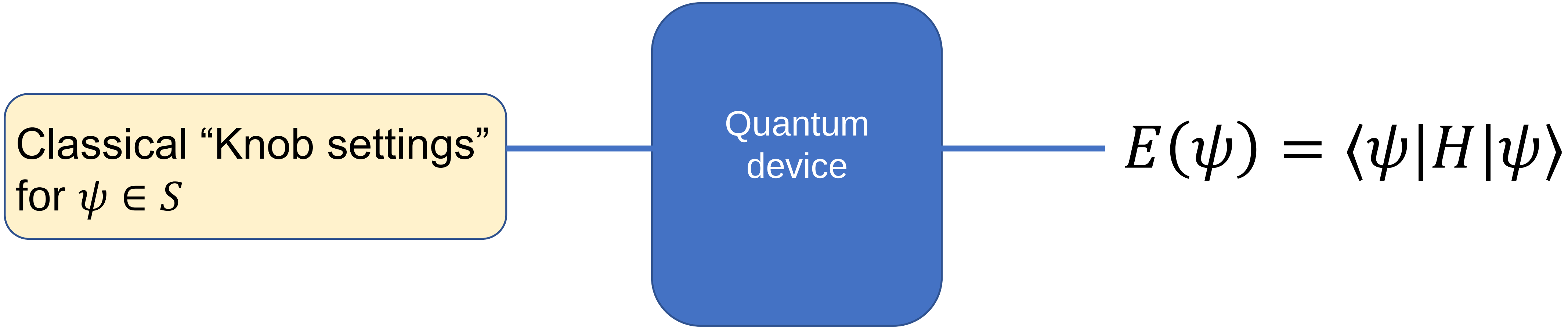
Mild assumption: your quantum device can prepare **some subset $\boldsymbol{S}$ of $\boldsymbol{n}$-qubit states $\boldsymbol{\psi}$** and measure the energy of a given ψ in this set.

Variational Quantum Algorithms

Goal: compute the ground energy of a given Hamiltonian.

$$H = \sum_i P_i \qquad\qquad E_{min} = \min_{\psi} \langle \psi | H | \psi \rangle$$

Mild assumption: your quantum device can prepare **some subset S of n-qubit states ψ** and measure the energy of a given ψ in this set.

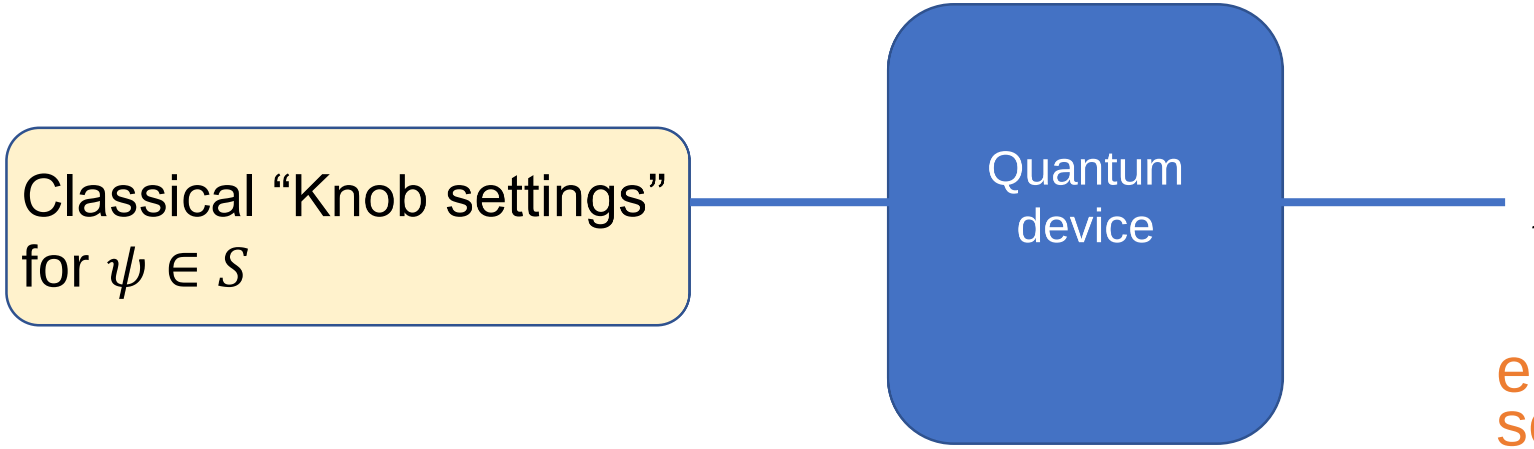


Variational Quantum Algorithms

Goal: compute the ground energy of a given Hamiltonian.

$$H = \sum_i P_i \qquad\qquad E_{min} = \min_{\psi} \langle \psi | H | \psi \rangle$$

Mild assumption: your quantum device can prepare **some subset S of n-qubit states ψ** and measure the energy of a given ψ in this set.



$$E(\psi) = \langle \psi | H | \psi \rangle$$

e.g., by computing $\langle \psi | P_i | \psi \rangle$ separately and then summing

Variational Quantum Algorithms

A variational algorithm aims to compute the minimum energy **over states in $\boldsymbol{S}$**

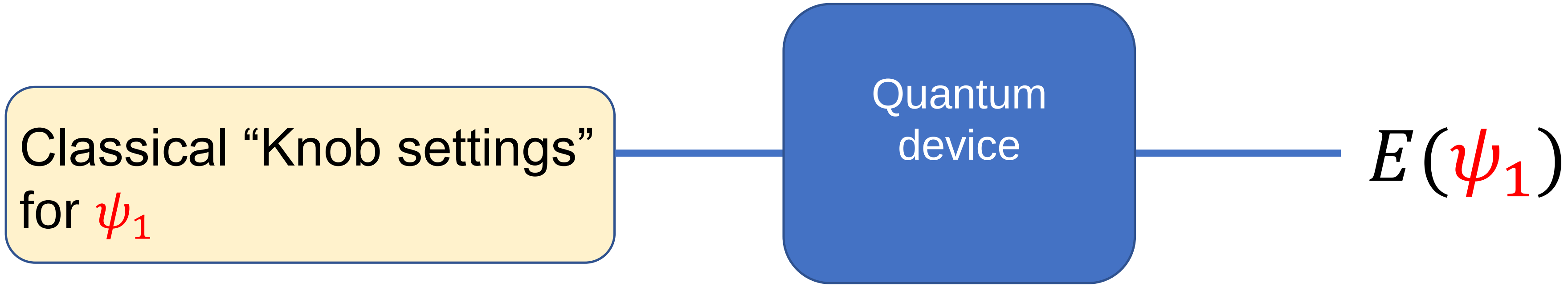
$$\min_{\psi \in S} \langle \psi | H | \psi \rangle$$

Variational Quantum Algorithms

A variational algorithm aims to compute the minimum energy **over states in S**

$$\min_{\psi \in S} \langle \psi | H | \psi \rangle$$

The algorithm uses the quantum device to compute energies and a classical computer to choose the knob settings:

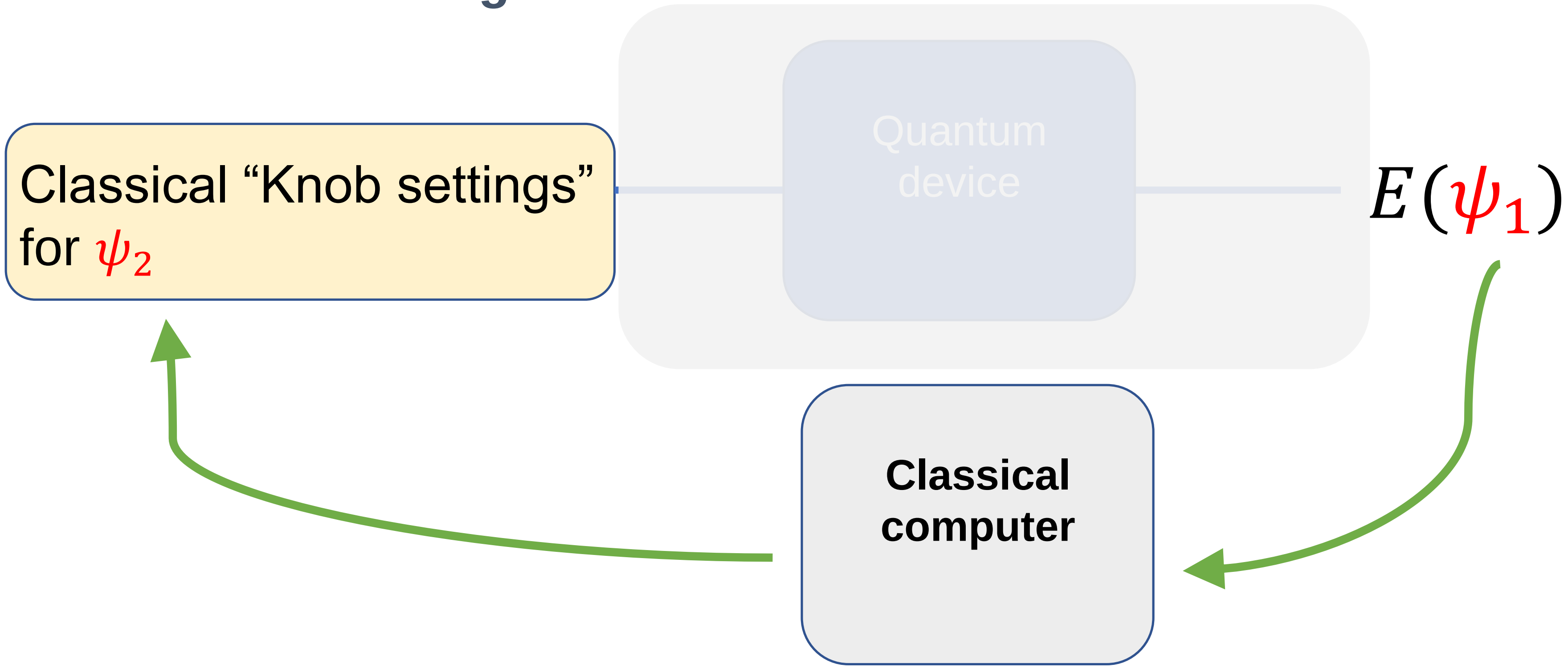


Variational Quantum Algorithms

A variational algorithm aims to compute the minimum energy **over states in S**

$$\min_{\psi \in S} \langle \psi | H | \psi \rangle$$

The algorithm uses the quantum device to compute energies and a classical computer to choose the knob settings:

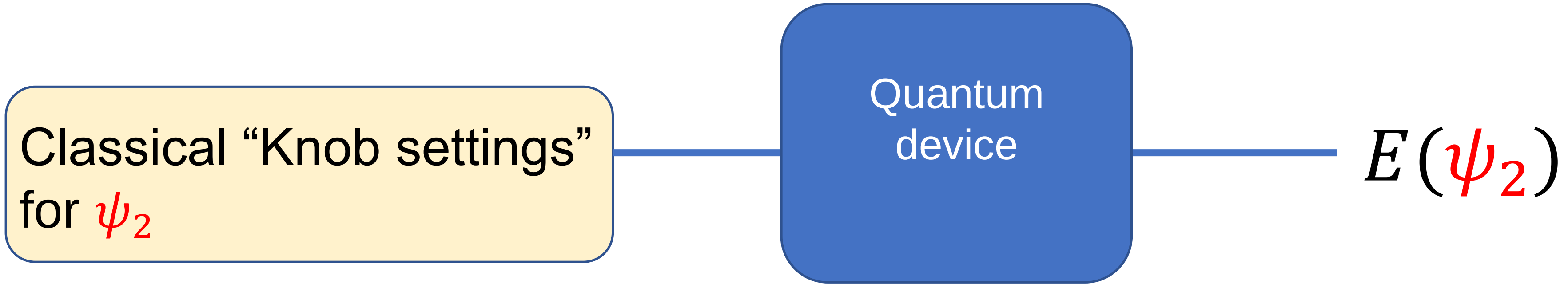


Variational Quantum Algorithms

A variational algorithm aims to compute the minimum energy **over states in S**

$$\min_{\psi \in S} \langle \psi | H | \psi \rangle$$

The algorithm uses the quantum device to compute energies and a classical computer to choose the knob settings:

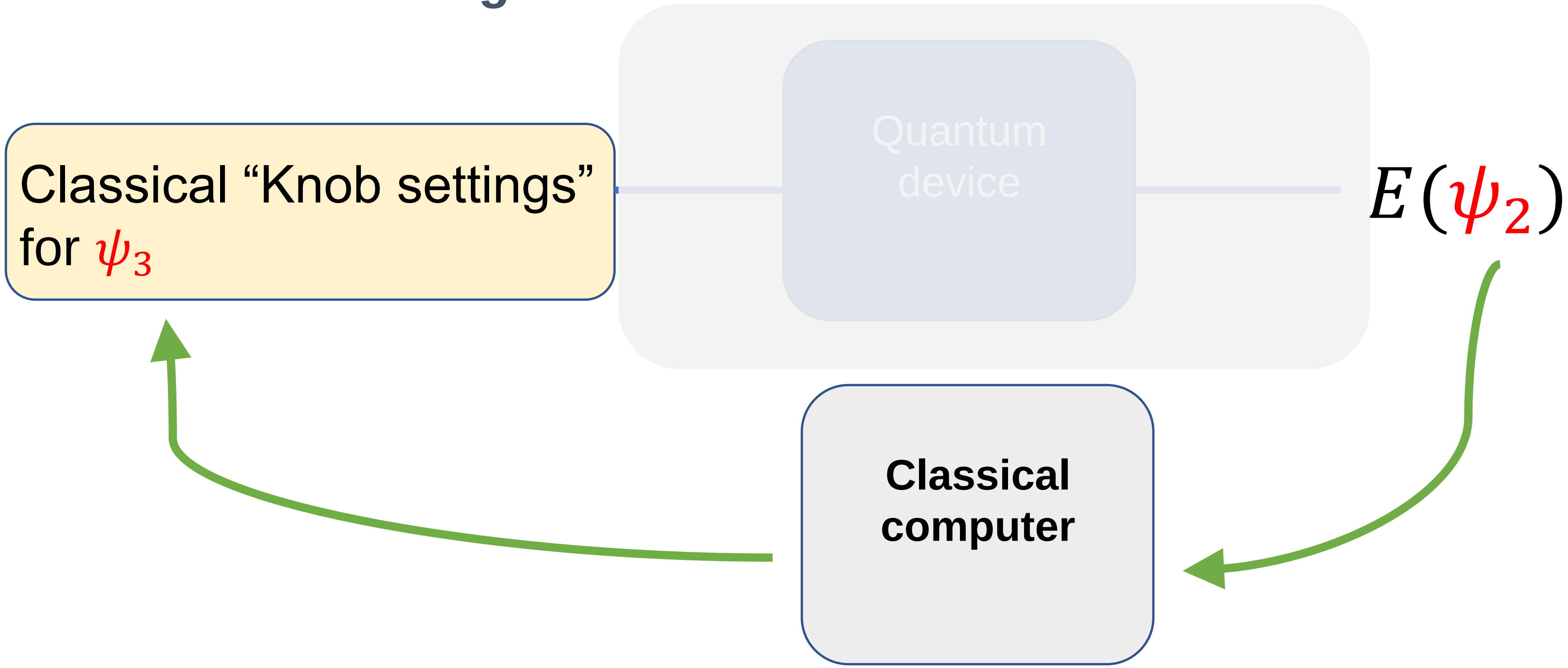


Variational Quantum Algorithms

A variational algorithm aims to compute the minimum energy **over states in S**

$$\min_{\psi \in S} \langle \psi | H | \psi \rangle$$

The algorithm uses the quantum device to compute energies and a classical computer to choose the knob settings:



Variational Quantum Algorithms

There are many proposed applications:

International Workshop on Quantum Technology and Optimization Problems
QTOP 2019: Quantum Technology and Optimization Problems pp 74-85 | Cite as

Variational Quantum Factoring

Authors | Authors and affiliations

Eric Anschuetz, Jonathan Olson, Alán Aspuru-Guzik, Yudong Cao

PHYSICAL REVIEW LETTERS

Highlights Recent Accepted Collections Authors Referees Search Pre

Quantum Machine Learning in Feature Hilbert Spaces

Maria Schuld and Nathan Killoran
Phys. Rev. Lett. **122**, 040504 – Published 1 February 2019

nature
International journal of science

Letter | Published: 13 September 2017

Hardware-efficient variational quantum eigensolver for small molecules and quantum magnets

Abhinav Kandala, Antonio Mezzacapo, Kristan Temme, Maika Takita, Markus Brink, Jerry M. Chow & Jay M. Gambetta

nature
International journal of science

Letter | Published: 13 March 2019

Supervised learning with quantum-enhanced feature spaces

Vojtěch Havlíček, Antonio D. Córcoles, Kristan Temme, Aram W. Harrow, Abhinav Kandala, Jerry M. Chow & Jay M. Gambetta

Unfortunately, variational algorithms generally don’t have performance guarantees.
Do they have any advantage over classical algorithms?...

Variational Quantum Algorithms

The quantum computer is only used to approximate mean values of observables at the output of a quantum computation

$$\mu = \langle 0^n | U^\dagger O U | 0^n \rangle$$

We restrict to: constant-depth quantum circuit U and tensor product observables O

Our question: Can we approximate μ on a classical computer instead?

The mean value problem

Let U be a depth $d = O(1)$ quantum circuit.

Let O be a tensor product of single-qubit Hermitian operators

$$O = O_1 \otimes O_2 \otimes \cdots \otimes O_n$$

Assume $\|O_j\| \leq 1$

We are interested in estimating the mean value

$$\mu = \langle 0^n | U^\dagger O \, U | 0^n \rangle$$

The mean value problem

$$\mu = \langle 0^n | U^\dagger O \, U | 0^n \rangle$$

Interesting special case:

$$O = |x_1\rangle\langle x_1| \otimes |x_2\rangle\langle x_2| \otimes \cdots \otimes |x_n\rangle\langle x_n|$$

Then the mean value is an output probability of the quantum circuit

$$\mu = |\langle x|U|0^n\rangle|^2$$

The mean value problem

$$O = O_1 \otimes O_2 \otimes \cdots \otimes O_n$$

$$\mu = \langle 0^n | U^\dagger O \, U | 0^n \rangle$$

Additive error mean value problem

Given $\epsilon = \frac{1}{poly(n)}$, compute an estimate $\tilde{\mu}$ such that

$$|\tilde{\mu} - \mu| < \epsilon$$

The mean value problem

$$O = O_1 \otimes O_2 \otimes \cdots \otimes O_n$$

$$\mu = \langle 0^n | U^\dagger O\, U | 0^n \rangle$$

Additive error mean value problem

Given $\epsilon = \frac{1}{poly(n)}$, compute an estimate $\tilde{\mu}$ such that

$$|\tilde{\mu} - \mu| < \epsilon$$

The additive error mean value problem can be solved efficiently on a quantum computer.

The mean value problem

$$O = O_1 \otimes O_2 \otimes \cdots \otimes O_n$$

$$\mu = \langle 0^n | U^\dagger O \, U | 0^n \rangle$$

Relative error mean value problem

Given $\epsilon = \frac{1}{poly(n)}$, compute an estimate $\tilde{\mu}$ such that

$$|\tilde{\mu} - \mu| < \epsilon\mu$$

The mean value problem

$$O = O_1 \otimes O_2 \otimes \cdots \otimes O_n$$

$$\mu = \langle 0^n | U^\dagger O\, U | 0^n \rangle$$

Relative error mean value problem

Given $\epsilon = \frac{1}{poly(n)}$, compute an estimate $\tilde{\mu}$ such that

$$|\tilde{\mu} - \mu| < \epsilon\mu$$

The relative error mean value problem is $\#P$-hard.

Complexity of the mean value problem

Quantum circuit U	**Observables O_j**	**Relative error**	**Additive error**
Polynomial size	Pos. semidefinite	#P-hard [1]	BQP-complete
Constant depth	?	?	?

[1] Goldberg Guo 17

In the rest of the talk I will describe 3 cases where the mean value problem is “easy” for classical computers…

Case 1: Single-qubit observables are each close to the identity

Restricted family of tensor product observables

Suppose U is a depth-d quantum circuit and consider an observable

$$O = O_1 \otimes O_2 \otimes \cdots \otimes O_n$$

where

$$\|O_j - I\| \leq \frac{0.001}{2^{5d}}$$

Closeness depends only on depth d

Restricted family of tensor product observables

Suppose U is a depth-d quantum circuit and consider an observable

$$O = O_1 \otimes O_2 \otimes \cdots \otimes O_n \quad \text{where} \quad \|O_j - I\| \leq \frac{0.001}{2^{5d}}$$

Closeness depends only on depth d

In this part we consider the **relative error** MVP for

$$\mu = \langle 0^n | U^\dagger O \, U | 0^n \rangle$$

Restricted family of tensor product observables

Suppose U is a depth-d quantum circuit and consider an observable

$$O = O_1 \otimes O_2 \otimes \cdots \otimes O_n \quad \text{where} \quad \|O_j - I\| \leq \frac{0.001}{2^{5d}}$$

Closeness depends only on depth d

In this part we consider the **relative error** MVP for

$$\mu = \langle 0^n | U^\dagger O \, U | 0^n \rangle$$

When would you ever encounter such observables?...

Restricted family of tensor product observables

Example

Suppose we consider an output probability of a **noisy quantum circuit**

$$\mu' = \langle 0^n | \mathcal{E}^{\otimes n}(U^\dagger |0\rangle\langle 0|^n)\, U | 0^n \rangle.$$

$$\mathcal{E}(\rho) = (1-p)\rho + pX\rho X$$

Flip each bit with probability p

Restricted family of tensor product observables

Example

Suppose we consider an output probability of a **noisy quantum circuit**

$$\mu' = \langle 0^n | \mathcal{E}^{\otimes n} (U^\dagger |0\rangle\langle 0|^n)\, U | 0^n \rangle.$$

$$\mathcal{E}(\rho) = (1-p)\rho + pX\rho X$$ Flip each bit with probability p

The noisy mean value is proportional to an ideal mean value:

$$\mu' = \frac{1}{2^n}\mu$$ with single-qubit observables $$O_j = I + (1-2p)Z$$

Restricted family of tensor product observables

Example

Suppose we consider an output probability of a **noisy quantum circuit**

$$\mu' = \langle 0^n | \mathcal{E}^{\otimes n}(U^\dagger |0\rangle\langle 0|^n)\, U | 0^n \rangle.$$

$$\mathcal{E}(\rho) = (1-p)\rho + pX\rho X$$

Flip each bit with probability p

The noisy mean value is proportional to an ideal mean value:

$$\mu' = \frac{1}{2^n}\mu$$ with single-qubit observables $O_j = I + (1-2p)Z$

This is sufficiently close to the identity in a high noise regime $p \geq \frac{1}{2} - O(2^{-5d})$

Main result

$$\mu = \langle 0^n | U^\dagger O \, U | 0^n \rangle$$

$$O = O_1 \otimes O_2 \otimes \cdots \otimes O_n$$

$$\|O_j - I\| \leq \frac{0.001}{2^{5d}}$$

Theorem

Let $\delta \in \left(0, \frac{1}{2}\right)$ be given. There is a deterministic classical algorithm which outputs an estimate $\tilde{\mu}$ satisfying

$$|\log(\tilde{\mu}) - \log(\mu)| < \delta$$

The runtime of the algorithm is $(n\delta^{-1})^{c \cdot 2^d}$.

Main result

$$\mu = \langle 0^n | U^\dagger O\, U | 0^n \rangle$$

$$O = O_1 \otimes O_2 \otimes \cdots \otimes O_n$$

$$\|O_j - I\| \leq \frac{0.001}{2^{5d}}$$

Theorem

Let $\delta \in \left(0, \frac{1}{2}\right)$ be given. There is a deterministic classical algorithm which outputs an estimate $\tilde{\mu}$ satisfying

$$|\log(\tilde{\mu}) - \log(\mu)| < \delta$$

The runtime of the algorithm is $(n\delta^{-1})^{c\cdot 2^d}$.

Solves the relative error mean value problem for this restricted set of observables
Runtime can be improved for 2D geometrically local circuits

Main result

$$\mu = \langle 0^n | U^\dagger O\, U | 0^n \rangle$$

$$O = O_1 \otimes O_2 \otimes \cdots \otimes O_n$$

$$\|O_j - I\| \le \frac{0.001}{2^{5d}}$$

Theorem

Let $\delta \in \left(0, \frac{1}{2}\right)$ be given. There is a deterministic classical algorithm which outputs an estimate $\tilde{\mu}$ satisfying

$$|\log(\tilde{\mu}) - \log(\mu)| < \delta$$

The runtime of the algorithm is $(n\delta^{-1})^{c \cdot 2^d}$.

Solves the relative error mean value problem for this restricted set of observables
Runtime can be improved for 2D geometrically local circuits

The algorithm is based on a polynomial interpolation method due to Barvinok…

Classical simulation by polynomial interpolation

Define a polynomial

$$f(\epsilon) = \langle 0^n \big| U^\dagger O(\epsilon)\, U \big| 0^n \rangle$$

$$O(\epsilon) = O_1(\epsilon) \otimes O_2(\epsilon) \otimes \cdots \otimes O_n(\epsilon)$$

$$O_j(\epsilon) = (1 - \epsilon) I + \epsilon O$$

Classical simulation by polynomial interpolation

Define a polynomial

$$f(\epsilon) = \langle 0^n \big| U^\dagger O(\epsilon)\, U \big| 0^n \rangle$$

$$O(\epsilon) = O_1(\epsilon) \otimes O_2(\epsilon) \otimes \cdots \otimes O_n(\epsilon)$$

$$O_j(\epsilon) = (1-\epsilon)I + \epsilon O$$

Note that $f(0) = 1$ and we aim to compute $\mu = f(1)$

Classical simulation by polynomial interpolation

Define a polynomial

$$f(\epsilon) = \langle 0^n \big| U^\dagger O(\epsilon)\, U \big| 0^n \rangle \qquad O(\epsilon) = O_1(\epsilon) \otimes O_2(\epsilon) \otimes \cdots \otimes O_n(\epsilon)$$

$$O_j(\epsilon) = (1-\epsilon)I + \epsilon O$$

Note that $f(0) = 1$ and we aim to compute $\mu = f(1)$

Also note that derivatives $f^{(k)}(0)$ can be computed efficiently for small k

Classical simulation by polynomial interpolation

Define a polynomial

$$f(\epsilon) = \langle 0^n | U^\dagger O(\epsilon)\, U | 0^n \rangle$$

$$O(\epsilon) = O_1(\epsilon) \otimes O_2(\epsilon) \otimes \cdots \otimes O_n(\epsilon)$$

$$O_j(\epsilon) = (1-\epsilon)I + \epsilon O$$

Note that $f(0) = 1$ and we aim to compute $\mu = f(1)$

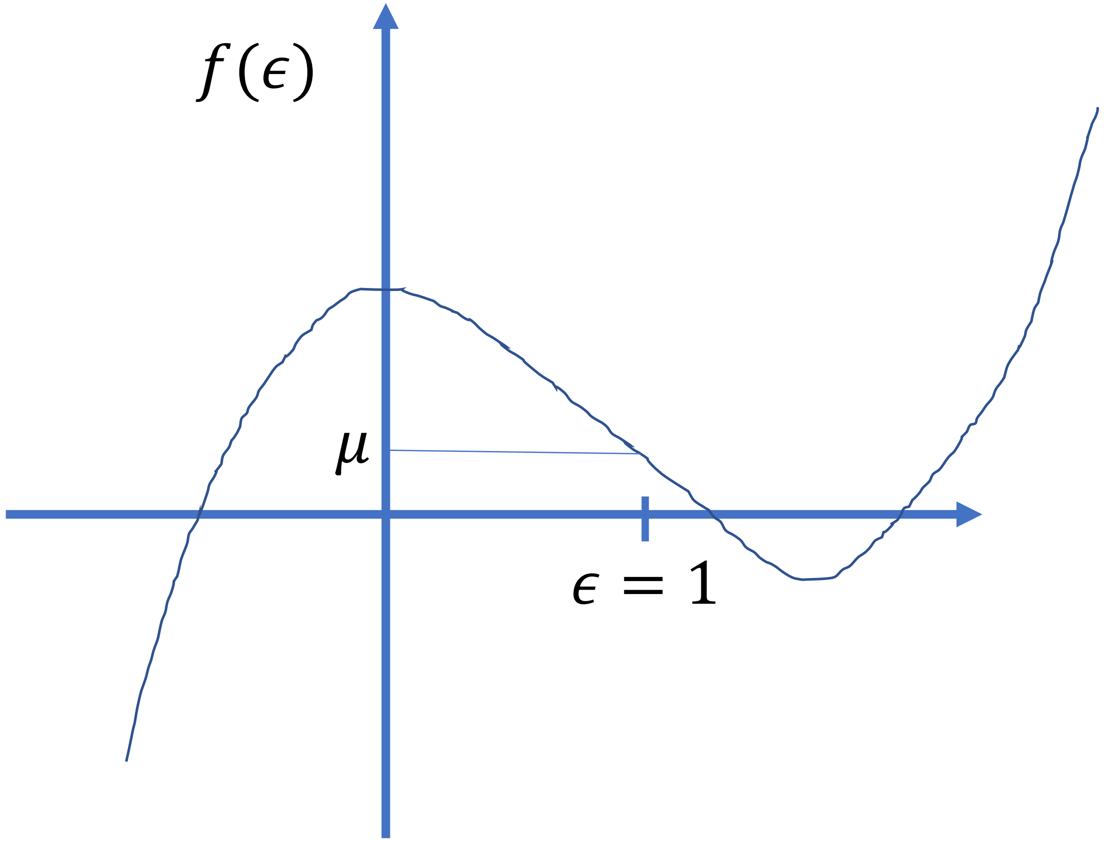
Also note that derivatives $f^{(k)}(0)$ can be computed efficiently for small k

e.g.,

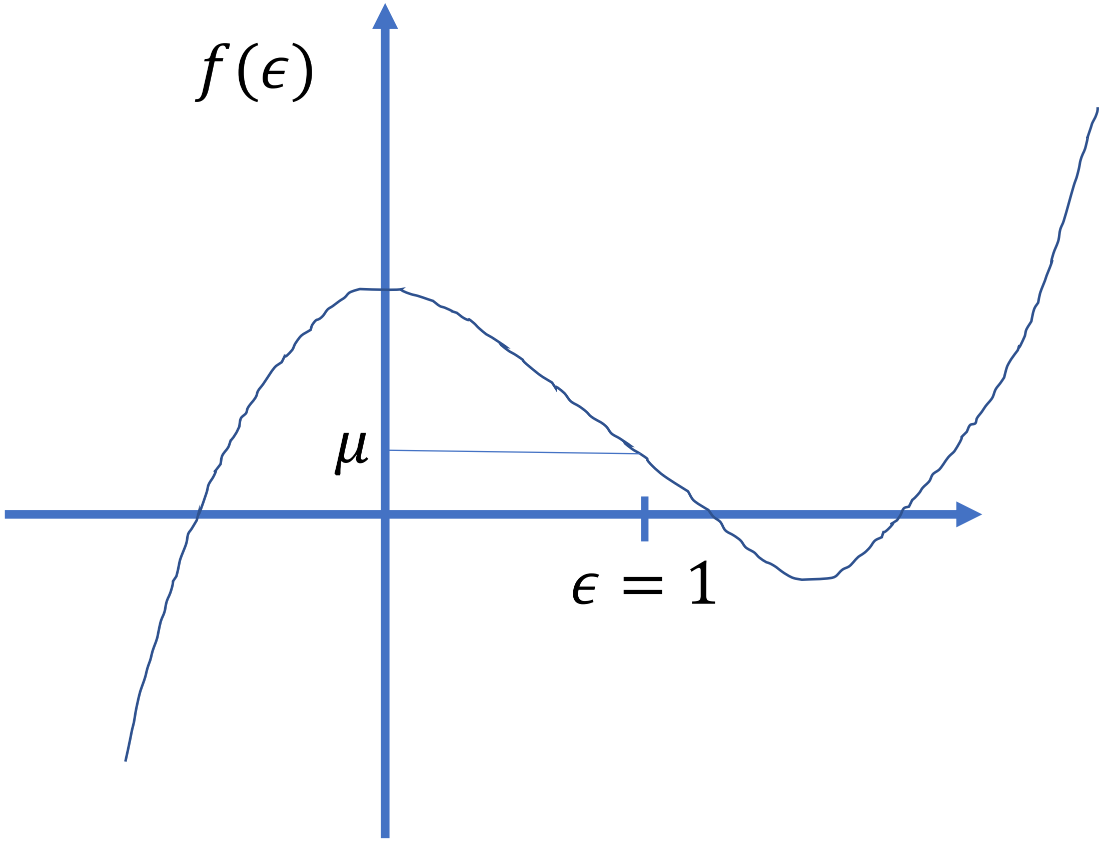
$$f^{(1)}(0) = \sum_{j=1}^{n} \langle 0^n | \underbrace{U^\dagger (O_j - I) U}_{\text{Acts nontrivially on } \leq 2^d \text{ qubits}} | 0^n \rangle$$

Acts nontrivially on $\leq 2^d$ qubits

Classical simulation by polynomial interpolation

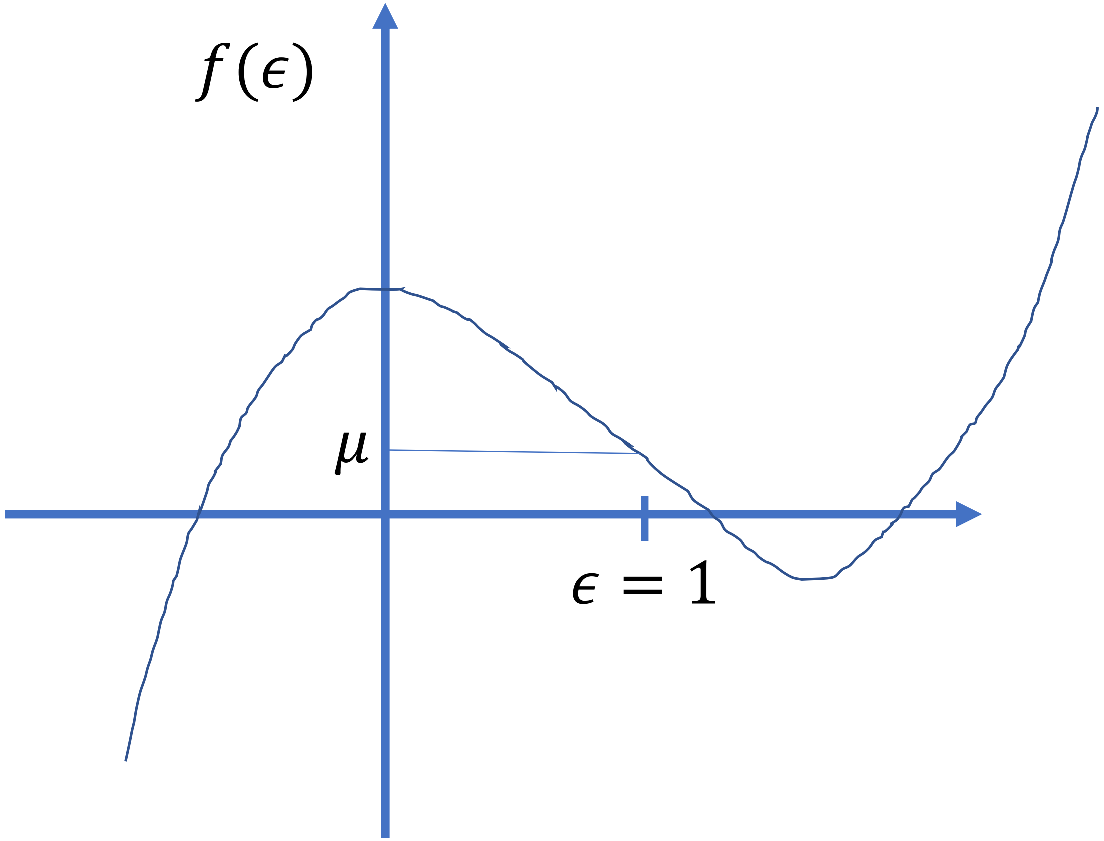


Classical simulation by polynomial interpolation



Since we know the function value and can compute derivatives at $\epsilon = 0$, it is natural to try to use a Taylor series approximation.

Classical simulation by polynomial interpolation



Since we know the function value and can compute derivatives at $\epsilon = 0$, it is natural to try to use a Taylor series approximation.

Barvinok: use Taylor series for the function $g(\epsilon) = \log(f(\epsilon))$ instead…

Classical simulation by polynomial interpolation

Approximate the log by its truncated Taylor series

$$\mathrm{g}(\epsilon) = \log f(\epsilon)$$

We want to compute $g(1)$

$$\mathrm{T}_p(\epsilon) = g(0) + \sum_{k=1}^{p} \frac{\epsilon^k}{k!} g^{(k)}(0)$$

Classical simulation by polynomial interpolation

Approximate the log by its truncated Taylor series

$$\mathrm{g}(\epsilon) = \log f(\epsilon)$$

We want to compute $g(1)$

$$\mathrm{T}_p(\epsilon) = g(0) + \sum_{k=1}^{p} \frac{\epsilon^k}{k!} g^{(k)}(0)$$

Theorem [Barvinok]

If the polynomial $f(\epsilon)$ is zero-free on the disk $|\epsilon| \leq 2$ then

$$|\mathrm{T}_p(\epsilon) - \mathrm{g}(\epsilon)| \leq \frac{n}{(p+1)2^p} \qquad |\epsilon| \leq 1$$

Classical simulation by polynomial interpolation

Approximate the log by its truncated Taylor series

$$\mathrm{g}(\epsilon) = \log f(\epsilon)$$

We want to compute $g(1)$

$$\mathrm{T}_p(\epsilon) = g(0) + \sum_{k=1}^{p} \frac{\epsilon^k}{k!} g^{(k)}(0)$$

Theorem [Barvinok]

If the polynomial $f(\epsilon)$ is zero-free on the disk $|\epsilon| \leq 2$ then

$$|\mathrm{T}_p(\epsilon) - \mathrm{g}(\epsilon)| \leq \frac{n}{(p+1)2^p} \qquad |\epsilon| \leq 1$$

To achieve error δ we need only take $p = O(\log(n\delta^{-1}))$

Classical simulation by polynomial interpolation

$$g(\epsilon) = \log f(\epsilon)$$

To use Barvinok's method we need two ingredients:

Classical simulation by polynomial interpolation

$$\mathrm{g}(\epsilon) = \log f(\epsilon)$$

To use Barvinok's method we need two ingredients:

1) We need to compute derivatives

$$g^{(1)}(0), \ldots, g^{(p)}(0) \qquad\qquad p = O(\log(n\delta^{-1}))$$

These can be computed efficiently from the derivatives $f^{(1)}(0), \ldots, f^{(k)}(0)$.

Classical simulation by polynomial interpolation

$$\mathrm{g}(\epsilon) = \log f(\epsilon)$$

To use Barvinok's method we need two ingredients:

1) We need to compute derivatives

$$g^{(1)}(0), \ldots, g^{(p)}(0) \qquad p = O(\log(n\delta^{-1}))$$

These can be computed efficiently from the derivatives $f^{(1)}(0), \ldots, f^{(k)}(0)$.

2) We need to show that $f(\epsilon)$ is zero-free on the disk $|\epsilon| \leq 2$ …

Zero-free region

$$f(\epsilon) = \langle 0^n \big| U^\dagger O(\epsilon)\, U \big| 0^n \rangle \qquad O(\epsilon) = O_1(\epsilon) \otimes O_2(\epsilon) \otimes \cdots \otimes O_n(\epsilon)$$

$$O_j(\epsilon) = (1-\epsilon)I + \epsilon O$$

Theorem

Suppose $\|O_j - I\| \leq \gamma.$ The polynomial f has no zeros in the disk

$$|\epsilon| \leq \frac{0.001}{\gamma 2^{5d}}$$

Zero-free region

$$f(\epsilon) = \langle 0^n | U^\dagger O(\epsilon)\, U | 0^n \rangle$$

$$O(\epsilon) = O_1(\epsilon) \otimes O_2(\epsilon) \otimes \cdots \otimes O_n(\epsilon)$$

$$O_j(\epsilon) = (1-\epsilon)I + \epsilon O$$

Theorem

Suppose $\|O_j - I\| \leq \gamma$. The polynomial f has no zeros in the disk

$$|\epsilon| \leq \frac{0.001}{\gamma 2^{5d}}$$

Depth of U

Zero-free region

$$f(\epsilon) = \langle 0^n | U^\dagger O(\epsilon)\, U | 0^n \rangle$$

$$O(\epsilon) = O_1(\epsilon) \otimes O_2(\epsilon) \otimes \cdots \otimes O_n(\epsilon)$$

$$O_j(\epsilon) = (1-\epsilon)I + \epsilon O$$

Theorem

Suppose $\|O_j - I\| \le \gamma$. The polynomial f has no zeros in the disk

$$|\epsilon| \le \frac{0.001}{\gamma 2^{5d}}$$

→ Depth of U

Choosing $\gamma = 0.001 \cdot 2^{-5d-1}$ suffices to make the disk radius equal to 2.

Proof sketch (zero-free region)

$$f(\epsilon) = \langle 0^n \middle| U^\dagger O_1(\epsilon) \otimes \cdots O_n(\epsilon)\, U \middle| 0^n \rangle$$

Proof sketch (zero-free region)

$$f(\epsilon) = \langle 0^n \big| U^\dagger O_1(\epsilon) \otimes \cdots O_n(\epsilon)\, U \big| 0^n \rangle$$

Write each 2×2 operator $O_j(\epsilon)$ as the upper left block of a 4×4 **unitary** $\boldsymbol{B_j}(\boldsymbol{\epsilon})$

$$f(\epsilon) = \langle 0^{2n} \big| (U^\dagger \otimes I) B_1(\epsilon) \otimes \cdots B_n(\epsilon) (U \otimes I) \big| 0^{2n} \rangle$$

Proof sketch (zero-free region)

$$f(\epsilon) = \langle 0^n | U^\dagger O_1(\epsilon) \otimes \cdots O_n(\epsilon)\, U | 0^n \rangle$$

Write each 2×2 operator $O_j(\epsilon)$ as the upper left block of a 4×4 **unitary** $\boldsymbol{B_j(\epsilon)}$

$$f(\epsilon) = \langle 0^{2n} | (U^\dagger \otimes I) B_1(\epsilon) \otimes \cdots B_n(\epsilon)(U \otimes I) | 0^{2n} \rangle$$

Define $V_j(\epsilon) = (U^\dagger \otimes I) B_j(\epsilon)(U \otimes I)$

The $V_j(\epsilon)$ each act on 2^{d+1} qubits

Proof sketch (zero-free region)

$$f(\epsilon) = \langle 0^n \middle| U^\dagger O_1(\epsilon) \otimes \cdots O_n(\epsilon)\, U \middle| 0^n \rangle$$

Write each 2×2 operator $O_j(\epsilon)$ as the upper left block of a 4×4 **unitary** $\boldsymbol{B_j}(\epsilon)$

$$f(\epsilon) = \langle 0^{2n} \middle| (U^\dagger \otimes I) B_1(\epsilon) \otimes \cdots B_n(\epsilon)(U \otimes I) \middle| 0^{2n} \rangle$$

Define $V_j(\epsilon) = (U^\dagger \otimes I) B_j(\epsilon)(U \otimes I)$

The $V_j(\epsilon)$ each act on 2^{d+1} qubits

Then $f(\epsilon) = \langle 0^{2n} | V_1(\epsilon) V_2(\epsilon) \ldots V_n(\epsilon) | 0^{2n} \rangle$

A constant depth circuit
Each gate is close to identity

Proof sketch (zero-free region)

$f(\epsilon) = \langle 0^{2n} | V(\epsilon) | 0^{2n} \rangle$ $\quad V(\epsilon) = V_1(\epsilon) V_2(\epsilon) \ldots V_n(\epsilon)$

A constant-depth circuit
Each gate is close to identity

Now consider a probability distribution over $2n$-bit strings defined by

$$p_\epsilon(z) = \left| \langle z | V(\epsilon) | 0^{2n} \rangle \right|^2$$

Our goal is to show that $p_\epsilon\left(0^{2n}\right) > 0$ for all ϵ in the disk.

We establish this using the **Lovasz Local Lemma** (see paper for details).

Can the bound on zero-free radius be improved?

In the **worst case** the zero-free radius can be exponentially small in the depth.
There is a depth d circuit and single qubit observables such that

$$f(\epsilon) = \langle 0^{2^d} | O_1(\epsilon) \otimes \cdots \otimes O_{2^d}(\epsilon) | 0 \rangle^{2^d}$$
$$= \frac{1}{2}\left((1+\epsilon)^{2^d} + (1-\epsilon)^{2^d} \right)$$

Has a root at
$$\epsilon_0 \approx \frac{i\pi}{2^{d+1}}$$

Can the bound on zero-free radius be improved?

In the **worst case** the zero-free radius can be exponentially small in the depth.
There is a depth d circuit and single qubit observables such that

$$f(\epsilon) = \langle 0^{2^d} | O_1(\epsilon) \otimes \cdots \otimes O_{2^d}(\epsilon) | 0 \rangle^{2^d}$$
$$= \frac{1}{2}\left((1+\epsilon)^{2^d} + (1-\epsilon)^{2^d}\right)$$

Has a root at
$$\epsilon_0 \approx \frac{i\pi}{2^{d+1}}$$

For random circuits the zero free radius is **typically much larger.**
We show that it scales as $1 - O\left(\frac{\log(n)}{n}\right)$ for an n-qubit circuit drawn from any 2-design.

Complexity of the mean value problem

Quantum circuit U	**Observables O_j**	**Relative error**	**Additive error**
Polynomial size	Pos. semidefinite	#P-hard [1]	BQP-complete
Constant depth	Close to I	P	P

[1] Goldberg Guo 17

Case 2: Positive semidefinite observables

Subexponential time classical algorithm

$$O = O_1 \otimes O_2 \otimes \cdots \otimes O_n \qquad \|O_j\| = 1$$

Theorem

Let $\delta \in \left(0, \frac{1}{2}\right)$ be given. There is a deterministic classical algorithm which outputs an estimate $\tilde{\mu}$ satisfying

$$\left|\tilde{\mu} - |\langle 0^n | U^\dagger O U | 0^n \rangle|\right| < \delta$$

The runtime of the algorithm is $e^{\tilde{O}(4^d \sqrt{n \cdot \log(\delta^{-1})})}$.

Subexponential time classical algorithm

$$O = O_1 \otimes O_2 \otimes \cdots \otimes O_n \qquad \|O_j\| = 1$$

Theorem

Let $\delta \in \left(0, \frac{1}{2}\right)$ be given. There is a deterministic classical algorithm which outputs an estimate $\tilde{\mu}$ satisfying

$$\left|\tilde{\mu} - |\langle 0^n | U^\dagger O U | 0^n \rangle|\right| < \delta$$

The runtime of the algorithm is $e^{\tilde{O}(4^d \sqrt{n \cdot \log(\delta^{-1})})}$.

In general, the algorithm estimates the absolute value of the mean.

Subexponential time classical algorithm

$$O = O_1 \otimes O_2 \otimes \cdots \otimes O_n \qquad \|O_j\| = 1$$

Theorem

Let $\delta \in \left(0, \frac{1}{2}\right)$ be given. There is a deterministic classical algorithm which outputs an estimate $\tilde{\mu}$ satisfying

$$\left|\tilde{\mu} - |\langle 0^n | U^\dagger O U | 0^n \rangle|\right| < \delta$$

The runtime of the algorithm is $e^{\tilde{O}(4^d \sqrt{n \cdot \log(\delta^{-1})})}$.

In general, the algorithm estimates the absolute value of the mean.

Solves the additive error MVP for pos. semidefinite observables.

Subexponential time classical algorithm

$$O = O_1 \otimes O_2 \otimes \cdots \otimes O_n \qquad \|O_j\| = 1$$

Theorem

Let $\delta \in \left(0, \frac{1}{2}\right)$ be given. There is a deterministic classical algorithm which outputs an estimate $\tilde{\mu}$ satisfying

$$\left|\tilde{\mu} - |\langle 0^n|U^\dagger O U|0^n\rangle|\right| < \delta$$

The runtime of the algorithm is $e^{\tilde{O}(4^d \sqrt{n \cdot \log(\delta^{-1})})}$.

Proof idea:

First reduce to the case where $O = |0^n\rangle\langle 0^n|$ (easy)

Then we want

$$\left|\tilde{\mu} - |\langle 0^n|U|0^n\rangle|^2\right| < \delta$$

Subexponential time classical algorithm

Proof idea continued:

Consider a local Hamiltonian $H = \sum_{j=1}^{n} U|1\rangle\langle 1|_j U^\dagger$

Unique zero energy ground state is the state of interest $U|0^n\rangle$

Subexponential time classical algorithm

Proof idea continued:

Consider a local Hamiltonian $H = \sum_{j=1}^{n} U|1\rangle\langle 1|_j U^\dagger$

Unique zero energy ground state is the state of interest $U|0^n\rangle$

Fact: There is a polynomial p of degree $O(\sqrt{\boldsymbol{n} \cdot \log(\boldsymbol{\delta}^{-1})})$ s.t

$$\left\| p(H) - U|0^n\rangle\langle 0^n|U^\dagger \right\| \leq \delta$$

Obtained from a quantum query algorithm [Buhrman, Cleve, de Wolf, Zalka 1999]

Subexponential time classical algorithm

Proof idea continued:

Consider a local Hamiltonian $H = \sum_{j=1}^{n} U|1\rangle\langle 1|_j U^\dagger$

Unique zero energy ground state is the state of interest $U|0^n\rangle$

Fact: There is a polynomial p of degree $O(\sqrt{\boldsymbol{n} \cdot \log(\boldsymbol{\delta}^{-1})})$ s.t

$$\left\| p(H) - U|0^n\rangle\langle 0^n|U^\dagger \right\| \leq \delta$$

Obtained from a quantum query algorithm [Buhrman, Cleve, de Wolf, Zalka 1999]

The desired approximation is

$$\tilde{\mu} = \langle 0^n|p(H)|0^n\rangle$$

A k-local operator with $k = O\left(\boldsymbol{2^d}\sqrt{\boldsymbol{n} \cdot \log(\boldsymbol{\delta}^{-1})}\right)$.

Can be computed in time $e^{\tilde{O}(k)}$

Complexity of the mean value problem

Quantum circuit U	**Observables O_j**	**Relative error**	**Additive error**
Polynomial size	Pos. semidefinite	#P-hard [1]	BQP-complete
Constant depth	Close to I	P	P
Constant depth	Pos. semidefinite	#P-hard [1,2]	BQP Subexp. Classical alg.

[1] Goldberg Guo 17
[2] Terhal Divincenzo 02

Case 3: 2D shallow circuits

2D shallow circuits

Suppose the qubits are located at the vertices of a 2D grid, and U is a depth d quantum circuit where each gate acts between nearest-neighbors.

2D shallow circuits

Suppose the qubits are located at the vertices of a 2D grid, and U is a depth d quantum circuit where each gate acts between nearest-neighbors.

$$O = O_1 \otimes O_2 \otimes \cdots \otimes O_n \qquad \mu = \langle 0^n | U^\dagger O \, U | 0^n \rangle$$

Theorem

Let $\delta \in \left(0, \frac{1}{2}\right)$ be given. There is a randomized classical algorithm which, with probability at least 2/3, outputs an estimate $\tilde{\mu}$ satisfying

$$|\mu - \tilde{\mu}| \leq \delta$$

The runtime is $O(n\delta^{-2} 2^{O(d^2)})$.

2D shallow circuits

Suppose the qubits are located at the vertices of a 2D grid, and U is a depth d quantum circuit where each gate acts between nearest-neighbors.

$$O = O_1 \otimes O_2 \otimes \cdots \otimes O_n \qquad \mu = \langle 0^n | U^\dagger O\, U | 0^n \rangle$$

Theorem

Let $\delta \in \left(0, \frac{1}{2}\right)$ be given. There is a randomized classical algorithm which, with probability at least 2/3, outputs an estimate $\tilde{\mu}$ satisfying

$$|\mu - \tilde{\mu}| \leq \delta$$

The runtime is $O(n\delta^{-2} 2^{O(d^2)})$. Linear time!

2D shallow circuits

Suppose the qubits are located at the vertices of a 2D grid, and U is a depth d quantum circuit where each gate acts between nearest-neighbors.

$$O = O_1 \otimes O_2 \otimes \cdots \otimes O_n \qquad\qquad \mu = \langle 0^n | U^\dagger O\, U | 0^n \rangle$$

Theorem

Let $\delta \in \left(0, \frac{1}{2}\right)$ be given. There is a randomized classical algorithm which, with probability at least 2/3, outputs an estimate $\tilde{\mu}$ satisfying

$$|\mu - \tilde{\mu}| \leq \delta$$

The runtime is $O(n\delta^{-2}2^{O(d^2)})$. Linear time!

Algorithm is based on an MPS representation and Monte Carlo method…

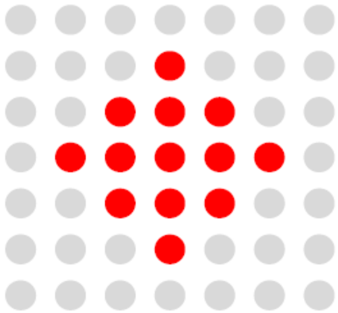
2D shallow circuit simulation

Express mean value as amplitude of a 2D constant depth circuit with **commuting gates**

$$\mu = \left\langle 0^n \middle| U^\dagger O_1 \otimes O_2 \otimes \cdots \otimes O_n \, U \middle| 0^n \right\rangle$$

$$= \langle 0^n | Q_n Q_{n-1} \dots Q_1 | 0^n \rangle \qquad Q_n = U^\dagger O_j \mathrm{U}$$

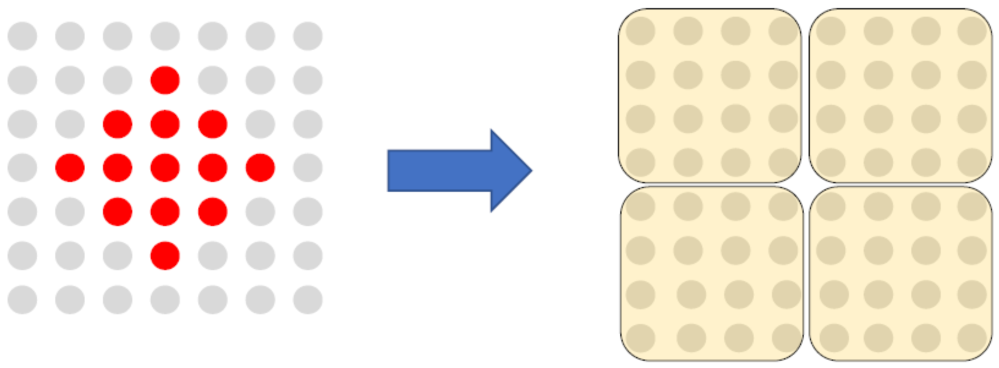
Each gate Q_j is supported on a $2d \times 2d$ square region centred at qubit j



2D shallow circuit simulation

$$\mu = \langle 0^n | Q_n Q_{n-1} \dots Q_1 | 0^n \rangle$$

Coarse-grain: group the qubits into supersites of size $2d \times 2d$



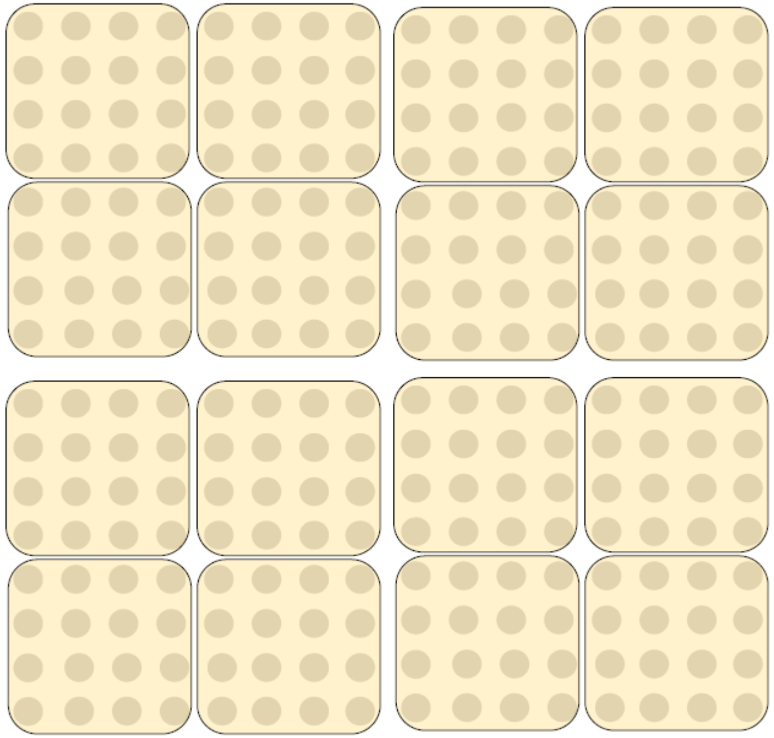
Each gate now acts nontrivially on 1 plaquette consisting of 4 supersites

2D shallow circuit simulation

Express mean value as inner product between two Matrix Product states

$$\mu = \langle 0^n | Q_n Q_{n-1} \dots Q_1 | 0^n \rangle$$

$$= \langle \Phi_{even} | \Phi_{odd} \rangle$$

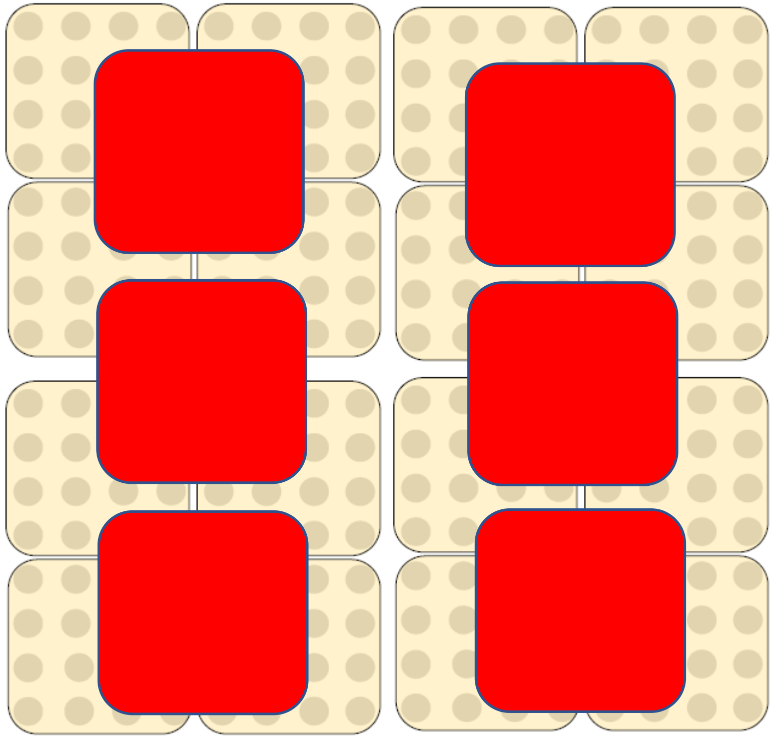


2D shallow circuit simulation

Express mean value as inner product between two Matrix Product states

$$\mu = \langle 0^n | Q_n Q_{n-1} \dots Q_1 | 0^n \rangle$$

$$= \langle \Phi_{even} | \Phi_{odd} \rangle$$

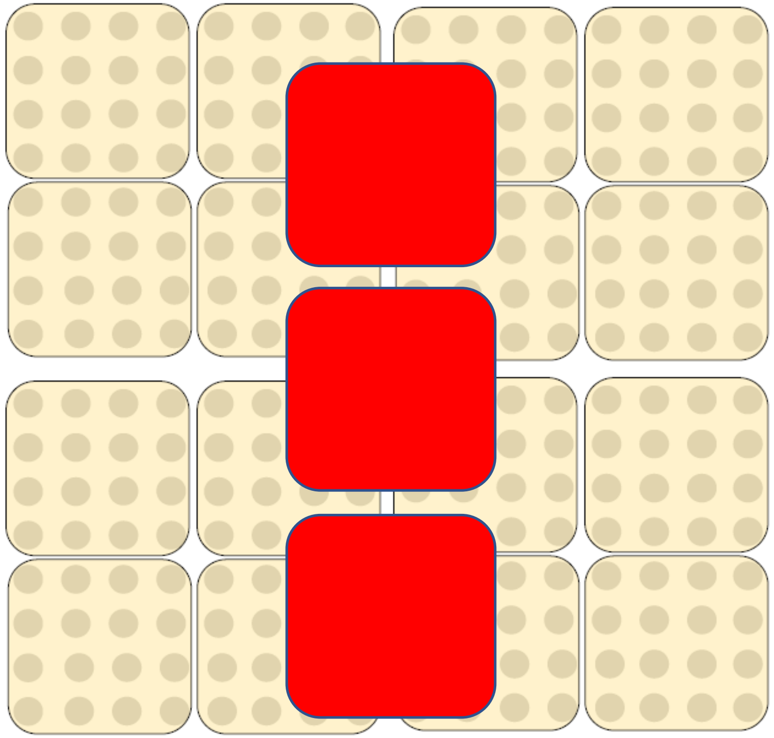


2D shallow circuit simulation

Express mean value as inner product between two Matrix Product states

$$\mu = \langle 0^n | Q_n Q_{n-1} \ldots Q_1 | 0^n \rangle$$

$$= \langle \Phi_{even} | \Phi_{odd} \rangle$$

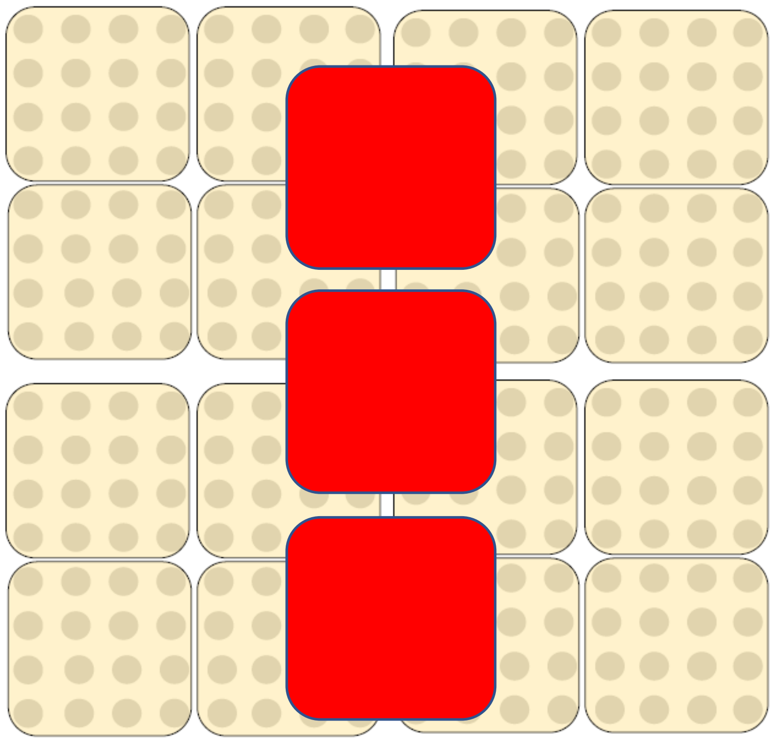


2D shallow circuit simulation

Express mean value as inner product between two Matrix Product states

$$\mu = \langle 0^n | Q_n Q_{n-1} \dots Q_1 | 0^n \rangle$$

$$= \langle \Phi_{even} | \Phi_{odd} \rangle$$



Inner product between MPS can be estimated in polynomial time using a Monte Carlo method [Van den Nest 2009]

Complexity of the mean value problem

Quantum circuit U	**Observables O_j**	**Relative error**	**Additive error**
Polynomial size	Pos. semidefinite	#P-hard [1]	BQP-complete
Constant depth	Close to I	P	P
Constant depth	Pos. semidefinite	#P-hard [1,2]	BQP Subexp. Classical alg.
2D Constant depth	Hermitian	#P-hard Subexp. classical [3]	BPP

[1] Goldberg Guo 17
[2] Terhal Divincenzo 02
[3] Markov, Shi 05

Open problems

Big question: what is the complexity of the additive-error mean value problem for constant-depth circuits?

Can the subexponential-time algorithm be generalized to the case of observables which may not be positive semidefinite?

Can the 2D algorithm be generalized to higher dimensional lattices?

Other applications of the zero-free region for quantum circuits?