A convex view of query (and control) algorithms

Duyal Yolcu

February 20, 2024

Overview

- We know some quantum and classical algorithms.

Overview

- We know some quantum and classical algorithms.
- Usual way to understand them: Keep track of memory states for each possible input.

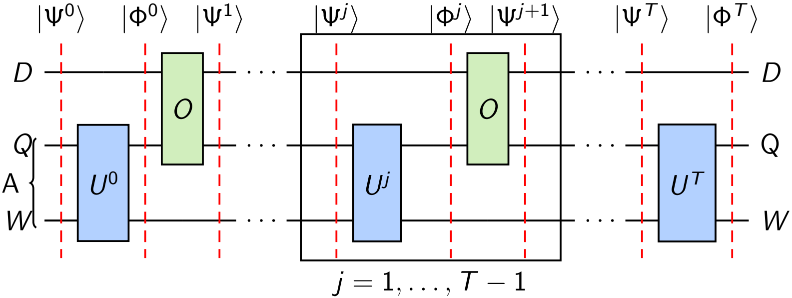
Overview

- We know some quantum and classical algorithms.
- Usual way to understand them: Keep track of memory states for each possible input.
- We will discuss an alternative, where we keep track of *collections of memory states, modulo equivalences*.

Overview

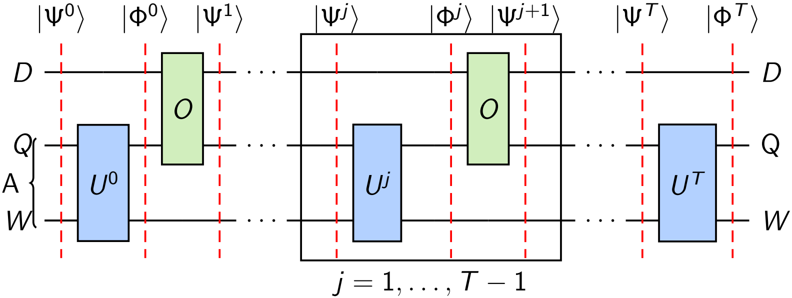
- We know some quantum and classical algorithms.
- Usual way to understand them: Keep track of memory states for each possible input.
- We will discuss an alternative, where we keep track of *collections of memory states, modulo equivalences*.
- Formally (pure-state quantum): The matrix of inner products (*Gram matrix*) of wavefunctions for different possible inputs.

The general view: Control



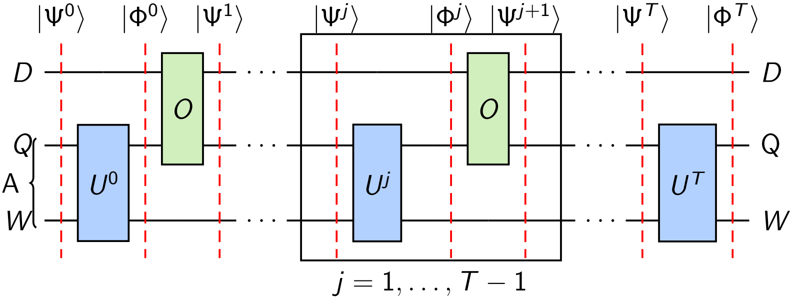
- An **agent** interacts with its **environment** in discrete time

The general view: Control



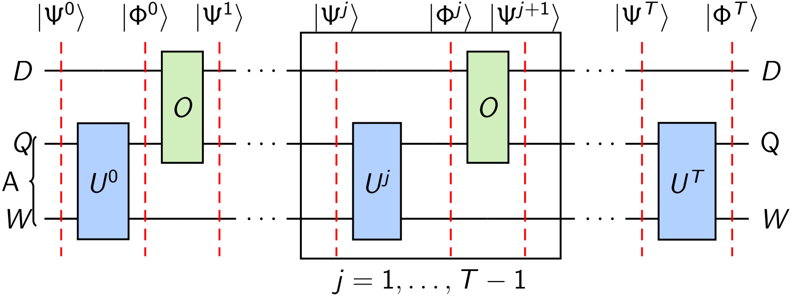
- An **agent** interacts with its **environment** in discrete time
- Goal: Determine a policy (sequences of transformations) that converts given inital state $|\Psi^0\rangle$ into a desirable target state $|\Phi^T\rangle$ (or lower-bound T necessary)

The general view: Control



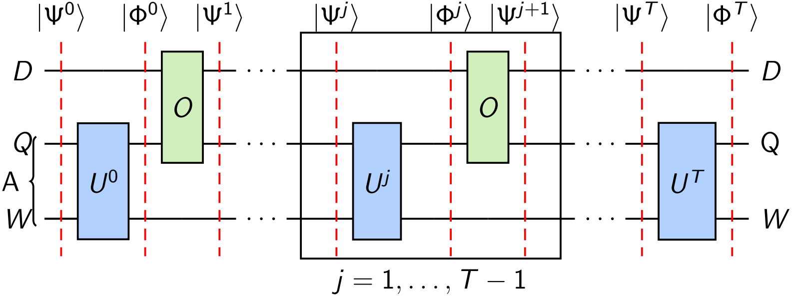
- An **agent** interacts with its **environment** in discrete time
- Goal: Determine a policy (sequences of transformations) that converts given inital state $|\Psi^0\rangle$ into a desirable target state $|\Phi^T\rangle$ (or lower-bound T necessary)
- $|\Psi^0\rangle$ and $|\Phi^T\rangle$ contain information about both state of environment and knowledge about environment

The general view: Control



U^i, O are matrices/linear maps, we don't restrict to norm-preserving for technical reasons

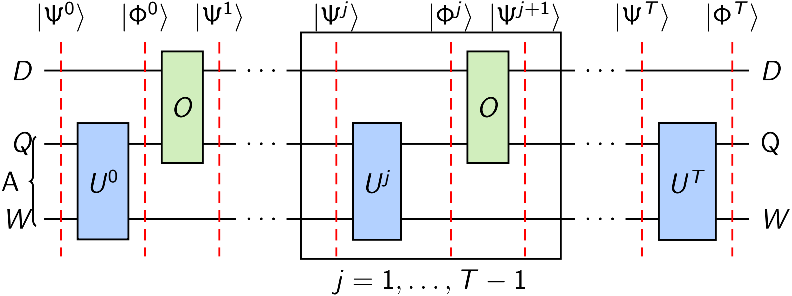
The general view: Control



U^i, O are matrices/linear maps, we don't restrict to norm-preserving for technical reasons

- classical: substochastic ($\mathbb{R}_{\geq 0}$-valued, L_1-nonincreasing),

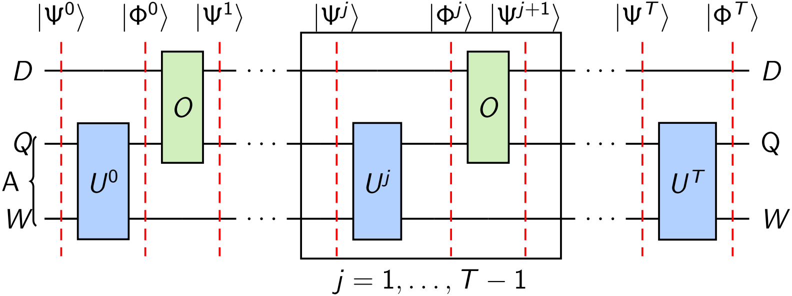
The general view: Control



U^i, O are matrices/linear maps, we don't restrict to norm-preserving for technical reasons

- classical: substochastic ($\mathbb{R}_{\geq 0}$-valued, L_1-nonincreasing),
- pure-state quantum: contractions ($\mathbb{C}$-valued, L_2-nonincreasing),

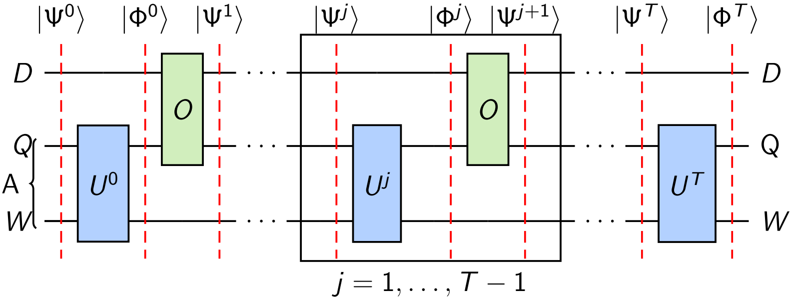
The general view: Control



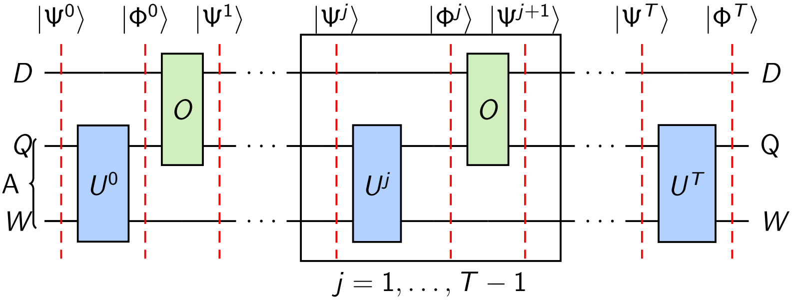
U^i, O are matrices/linear maps, we don't restrict to norm-preserving for technical reasons

- classical: substochastic ($\mathbb{R}_{\geq 0}$-valued, L_1-nonincreasing),
- pure-state quantum: contractions ($\mathbb{C}$-valued, L_2-nonincreasing),
- mixed-state quantum: completely positive trace-nonincreasing

The general view: Control

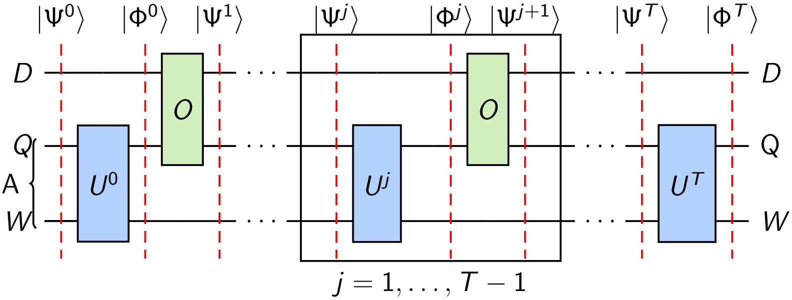


The general view: Control



Assumption: **Internal computation much cheaper than interaction:**

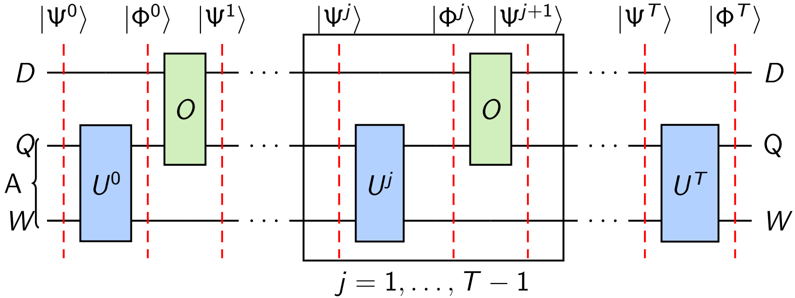
The general view: Control



Assumption: **Internal computation much cheaper than interaction:**

- Workspace W arbitrarily large,

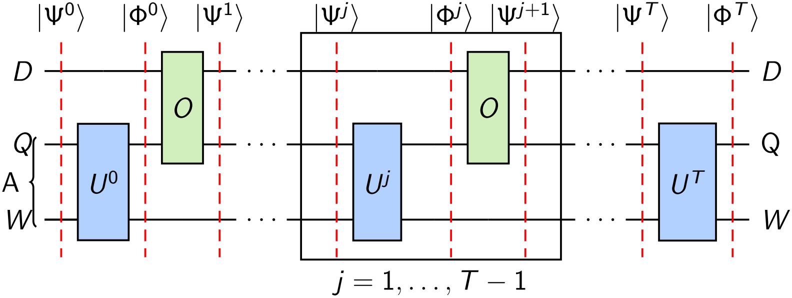
The general view: Control



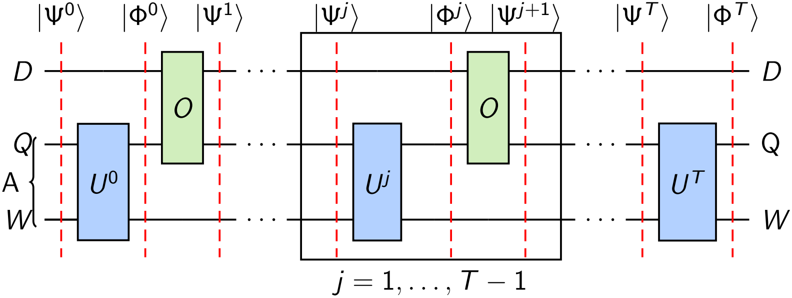
Assumption: **Internal computation much cheaper than interaction:**

- Workspace W arbitrarily large,
- Circuit complexity of U^i not counted.

Control problems: Examples

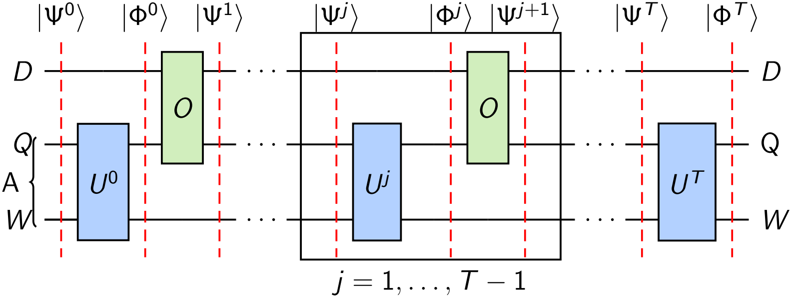


Control problems: Examples



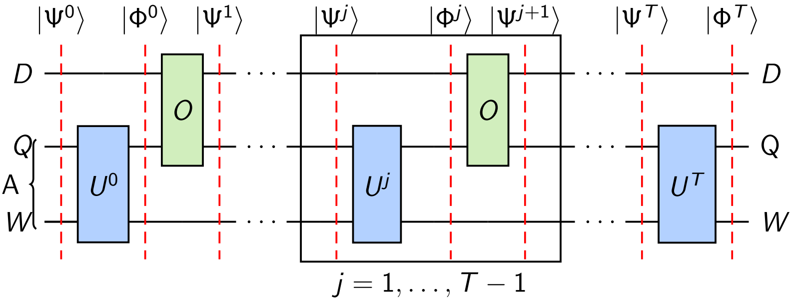
- Explore-exploit tradeoff/Multi-armed bandit

Control problems: Examples



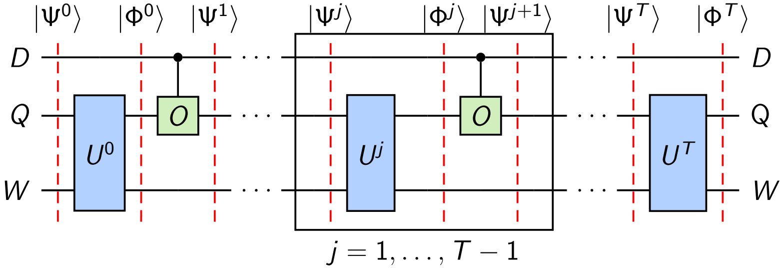
- Explore-exploit tradeoff/Multi-armed bandit
- Quantum/classical time-optimal control

Control problems: Examples



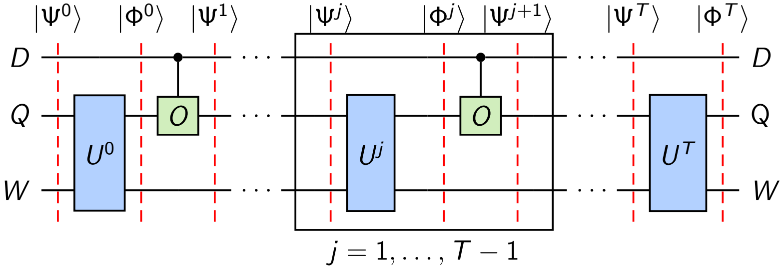
- Explore-exploit tradeoff/Multi-armed bandit
- Quantum/classical time-optimal control
- Robot cooks from fridge

Special case: Query problems



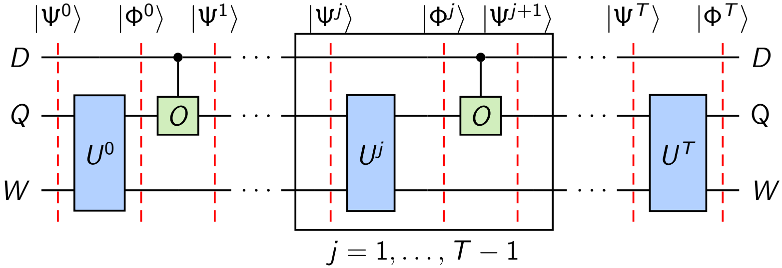
- evolution law O allows "read-only" access to state ($O = \bigoplus_{x \in D} O_x$ block-diagonal in D)

Special case: Query problems



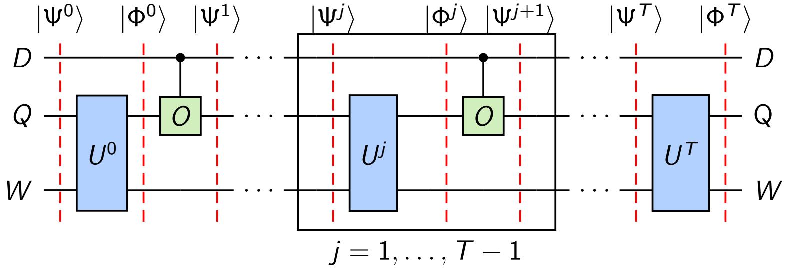
- evolution law O allows "read-only" access to state ($O = \bigoplus_{x \in D} O_x$ block-diagonal in D)
- Goal is to learn something about environment $x \in D$.
- Whether this is achieved can be judged by the final state $|\Phi^T\rangle$, where the agent's state (on QW) should be correlated to the environment's state

Special case: Query problems

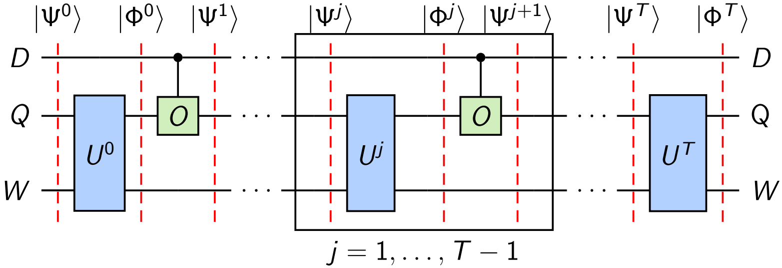


- evolution law O allows "read-only" access to state ($O = \bigoplus_{x \in D} O_x$ block-diagonal in D)
- Goal is to learn something about environment $x \in D$.
- Whether this is achieved can be judged by the final state $|\Phi^T\rangle$, where the agent's state (on QW) should be correlated to the environment's state

Query problems: Examples

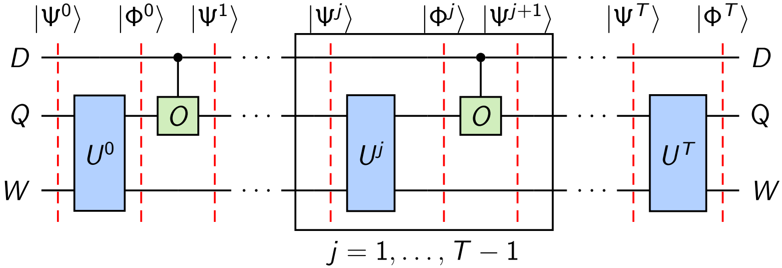


Query problems: Examples



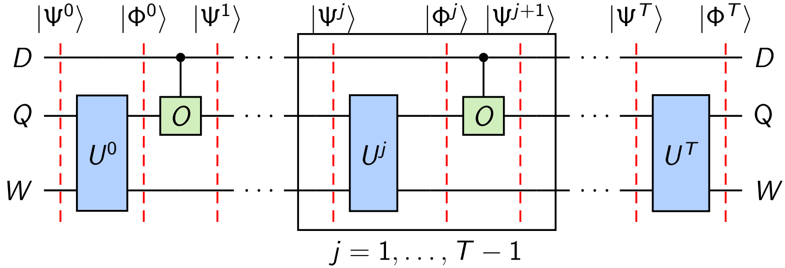
- Query algorithms, e.g. Grover's algorithm (as internal computations are cost-free)

Query problems: Examples



- Query algorithms, e.g. Grover's algorithm (as internal computations are cost-free)
- Learning of quantum oracles,

Query problems: Examples



- Query algorithms, e.g. Grover's algorithm (as internal computations are cost-free)
- Learning of quantum oracles,
- Classical: Active learning/Optimal experimental design

Example

- Search: O_x is a "harddisk" with n bits, input is address i, output is bit x_i, 1 bit set to 1, the others are 0.

Example

- Search: O_x is a "harddisk" with n bits, input is address i, output is bit x_i, 1 bit set to 1, the others are 0.
- Classically: Need to query half the bits in the harddisk to find the 1.

Example

- Search: O_x is a "harddisk" with n bits, input is address i, output is bit x_i, 1 bit set to 1, the others are 0.
- Classically: Need to query half the bits in the harddisk to find the 1.
- Quantumly, $\sim \sqrt{n}$ queries are **necessary and sufficient** by querying in superposition (lower bound for unstructured search/Grover's algorithm).

Example

- Search: O_x is a "harddisk" with n bits, input is address i, output is bit x_i, 1 bit set to 1, the others are 0.
- Classically: Need to query half the bits in the harddisk to find the 1.
- Quantumly, $\sim \sqrt{n}$ queries are **necessary and sufficient** by querying in superposition (lower bound for unstructured search/Grover's algorithm).

Example/Why query model?

- Noone can prove time complexity lower bounds for NP search problems, but proving $n/2$ lower bound for classical or $\sim \sqrt{n}$ lower bound for quantum search isn't too hard.

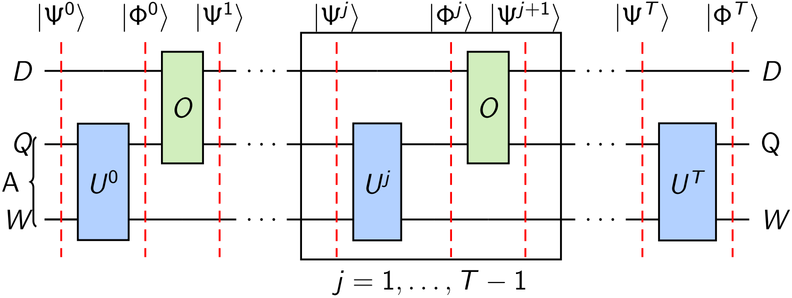
Example/Why query model?

- Noone can prove time complexity lower bounds for *NP* search problems, but proving $n/2$ lower bound for classical or $\sim \sqrt{n}$ lower bound for quantum search isn't too hard.
- But it still pertains to optimization/search problems! We can rule out that a quantum algorithm that treats the search space as a "black box" attains more than a quadratic speedup.

Example/Why query model?

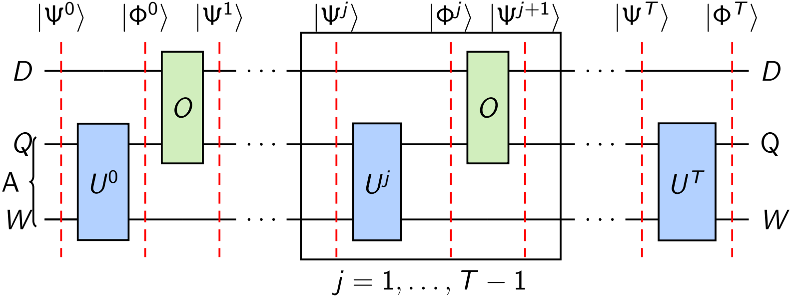
- Noone can prove time complexity lower bounds for *NP* search problems, but proving $n/2$ lower bound for classical or $\sim \sqrt{n}$ lower bound for quantum search isn't too hard.
- But it still pertains to optimization/search problems! We can rule out that a quantum algorithm that treats the search space as a "black box" attains more than a quadratic speedup.
- The **adversary bound** is a lower-bound method generalizing the unstructured search lower bound to other problems.

States, state collections, states of knowledge



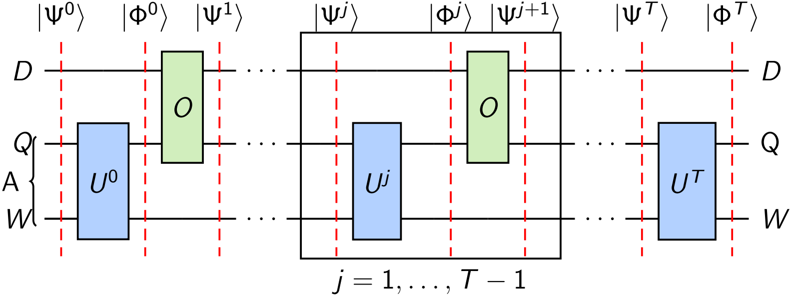
- $|\Psi\rangle$ determines next-round accessible states, agent's knowledge about D

States, state collections, states of knowledge



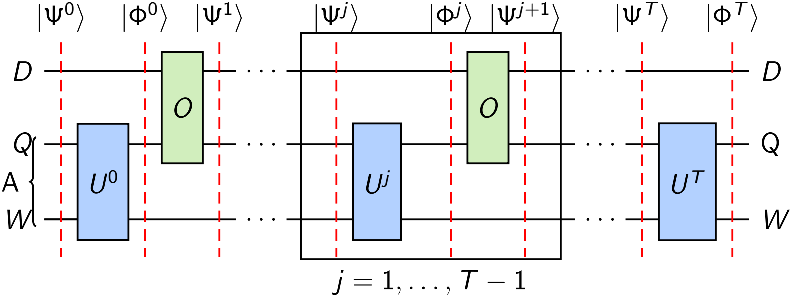
- $|\Psi\rangle = \sum_{x \in D} |x\rangle \otimes |\psi_x\rangle$ determines next-round accessible states, agent's knowledge about D

States, state collections, states of knowledge



- $|\Psi\rangle = \sum_{x \in D} |x\rangle \otimes |\psi_x\rangle \hat{=} (|\psi_x\rangle)_{x \in D}$ determines next-round accessible states, agent's knowledge about D

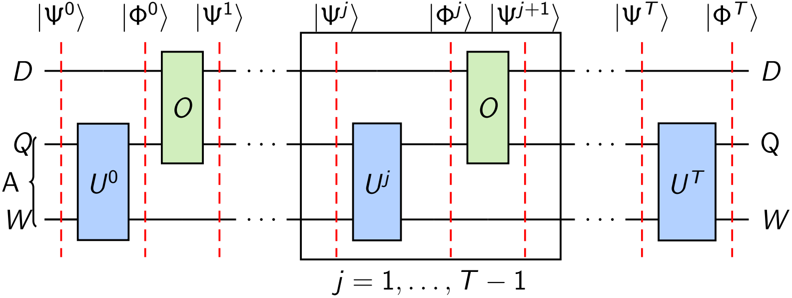
States, state collections, states of knowledge



- $|\Psi\rangle = \sum_{x \in D} |x\rangle \otimes |\psi_x\rangle \hat{=} (|\psi_x\rangle)_{x \in D}$ determines next-round accessible states, agent's knowledge about D
- **Equivalence relation**: $(|\psi_x\rangle)_{x \in D}$ equivalent to $(|\psi'_x\rangle)_{x \in D}$ iff agent can transform states into each other in memory with allowed U', U:

$$(|\psi_x\rangle)_{x \in D} = (U' |\psi'_x\rangle)_{x \in D} ,$$
$$(|\psi'_x\rangle)_{x \in D} = (U |\psi_x\rangle)_{x \in D}$$

States, state collections, states of knowledge



- Equivalence class $\left[(|\psi_x\rangle)_{x\in D}\right]_\sim$ determines next-round accessible states, agent's knowledge about D
- **Equivalence relation**: $(|\psi_x\rangle)_{x\in D}$ equivalent to $(|\psi'_x\rangle)_{x\in D}$ iff agent can transform states into each other in memory with allowed U', U:

$$(|\psi_x\rangle)_{x\in D} = (U' |\psi'_x\rangle)_{x\in D} ,$$
$$(|\psi'_x\rangle)_{x\in D} = (U |\psi_x\rangle)_{x\in D}$$

States, state collections, states of knowledge

- Equivalence class $\left[(|\psi_x\rangle)_{x\in D}\right]_\sim$ determines next-round accessible states, agent's knowledge about D
- **Equivalence relation**: $(|\psi_x\rangle)_{x\in D}$ equivalent to $(|\psi'_x\rangle)_{x\in D}$ iff agent can transform states into each other in memory with allowed U', U:

$$(|\psi_x\rangle)_{x\in D} = (U'\,|\psi'_x\rangle)_{x\in D}\,,$$
$$(|\psi'_x\rangle)_{x\in D} = (U\,|\psi_x\rangle)_{x\in D}$$

- Math fact, only for pure-state quantum: $(|\psi_x\rangle)_{x\in D}\,, (|\psi'_x\rangle)_{x\in D}$ are equivalent iff **Gram matrices** (matrices of inner products) are equal:

$$(\langle\psi_x|\psi_y\rangle)_{x,y\in D} = \left(\langle\psi'_x|\psi'_y\rangle\right)_{x,y\in D} \qquad (1)$$

States, state collections, states of knowledge

- Equivalence class $\left[(|\psi_x\rangle)_{x\in D}\right]_\sim$ determines next-round accessible states, agent's knowledge about D
- **Equivalence relation**: $(|\psi_x\rangle)_{x\in D}$ equivalent to $(|\psi'_x\rangle)_{x\in D}$ iff agent can transform states into each other in memory with allowed U', U:

$$(|\psi_x\rangle)_{x\in D} = (U'\,|\psi'_x\rangle)_{x\in D}\,,$$
$$(|\psi'_x\rangle)_{x\in D} = (U\,|\psi_x\rangle)_{x\in D}$$

- Math fact, only for pure-state quantum: $(|\psi_x\rangle)_{x\in D}\,,(|\psi'_x\rangle)_{x\in D}$ are equivalent iff **Gram matrices** (matrices of inner products) are equal:

$$(\langle\psi_x|\psi_y\rangle)_{x,y\in D} = \left(\langle\psi'_x|\psi'_y\rangle\right)_{x,y\in D} \tag{1}$$

- So info in $\left[(|\psi_x\rangle)_{x\in D}\right]_\sim$ is equivalent to $(\langle\psi_x|\psi_y\rangle)_{x,y\in D}$, and $\mathcal{S}^{+D}$ is equivalent to possible Gram matrices, i.e. set of positive semidefinite matrices

States, state collections, states of knowledge

- Equivalence class $\left[(|\psi_x\rangle)_{x\in D}\right]_\sim$ determines next-round accessible states, agent's knowledge about D
- **Equivalence relation**: $(|\psi_x\rangle)_{x\in D}$ equivalent to $(|\psi'_x\rangle)_{x\in D}$ iff agent can transform states into each other in memory with allowed U', U:

$$(|\psi_x\rangle)_{x\in D} = (U'\,|\psi'_x\rangle)_{x\in D}\,,$$
$$(|\psi'_x\rangle)_{x\in D} = (U\,|\psi_x\rangle)_{x\in D}$$

- Math fact, only for pure-state quantum: $(|\psi_x\rangle)_{x\in D}\,,(|\psi'_x\rangle)_{x\in D}$ are equivalent iff **Gram matrices** (matrices of inner products) are equal:

$$(\langle\psi_x|\psi_y\rangle)_{x,y\in D} = \left(\langle\psi'_x|\psi'_y\rangle\right)_{x,y\in D} \tag{1}$$

- So info in $\left[(|\psi_x\rangle)_{x\in D}\right]_\sim$ is equivalent to $(\langle\psi_x|\psi_y\rangle)_{x,y\in D}$, and $\mathcal{S}^{+D}$ is equivalent to possible Gram matrices, i.e. set of positive semidefinite matrices

Convex mixtures of SOKs

Given $p, p' \in \mathbb{R}_{\geq 0}$ and $\left[(|\psi_x\rangle)_{x \in D}\right]_\sim, \left[(|\psi'_x\rangle)_{x \in D}\right]_\sim \in \mathcal{S}^{+D}$, define (for pure quantum), define:

Convex mixtures of SOKs

Given $p, p' \in \mathbb{R}_{\geq 0}$ and $\left[(|\psi_x\rangle)_{x\in D}\right]_\sim, \left[(|\psi'_x\rangle)_{x\in D}\right]_\sim \in \mathcal{S}^{+D}$, define (for pure quantum), define:

$$p\left[(|\psi_x\rangle)_{x\in D}\right]_\sim + p'\left[(|\psi'_x\rangle)_{x\in D}\right]_\sim := \left[\left(\sqrt{p}\,|\psi_x\rangle \oplus \sqrt{p'}\,|\psi'_x\rangle\right)_{x\in D}\right]_\sim$$

$$\hat{=} \left[\left(\sqrt{p}\,|0\rangle \otimes |\psi_x\rangle + \sqrt{p'}\,|1\rangle \otimes |\psi'_x\rangle\right)_{x\in D}\right]_\sim$$

(for classical: $\sqrt{p} \to p$, $\sqrt{p'} \to p'$).

Convex mixtures of SOKs

Given $p, p' \in \mathbb{R}_{\geq 0}$ and $\left[(|\psi_x\rangle)_{x \in D}\right]_\sim, \left[(|\psi'_x\rangle)_{x \in D}\right]_\sim \in \mathcal{S}^{+D}$, define (for pure quantum), define:

$$p\left[(|\psi_x\rangle)_{x \in D}\right]_\sim + p'\left[(|\psi'_x\rangle)_{x \in D}\right]_\sim := \left[\left(\sqrt{p}\,|\psi_x\rangle \oplus \sqrt{p'}\,|\psi'_x\rangle\right)_{x \in D}\right]_\sim$$

$$\hat{=} \left[\left(\sqrt{p}\,|0\rangle \otimes |\psi_x\rangle + \sqrt{p'}\,|1\rangle \otimes |\psi'_x\rangle\right)_{x \in D}\right]_\sim$$

(for classical: $\sqrt{p} \to p$, $\sqrt{p'} \to p'$).

- well-defined, $+$ commutative, allows **convex combinations**

Convex mixtures of SOKs

Given $p, p' \in \mathbb{R}_{\geq 0}$ and $\left[(|\psi_x\rangle)_{x\in D}\right]_\sim, \left[(|\psi'_x\rangle)_{x\in D}\right]_\sim \in \mathcal{S}^{+D}$, define (for pure quantum), define:

$$p\left[(|\psi_x\rangle)_{x\in D}\right]_\sim + p'\left[(|\psi'_x\rangle)_{x\in D}\right]_\sim := \left[\left(\sqrt{p}\,|\psi_x\rangle \oplus \sqrt{p'}\,|\psi'_x\rangle\right)_{x\in D}\right]_\sim$$

$$\hat{=} \left[\left(\sqrt{p}\,|0\rangle \otimes |\psi_x\rangle + \sqrt{p'}\,|1\rangle \otimes |\psi'_x\rangle\right)_{x\in D}\right]_\sim$$

(for classical: $\sqrt{p} \to p$, $\sqrt{p'} \to p'$).

- well-defined, $+$ commutative, allows **convex combinations**
- For Gram matrices (pure-state quantum): Regular convex combinations

Convex mixtures of SOKs

Given $p, p' \in \mathbb{R}_{\geq 0}$ and $\left[(|\psi_x\rangle)_{x\in D}\right]_\sim, \left[(|\psi'_x\rangle)_{x\in D}\right]_\sim \in \mathcal{S}^{+D}$, define (for pure quantum), define:

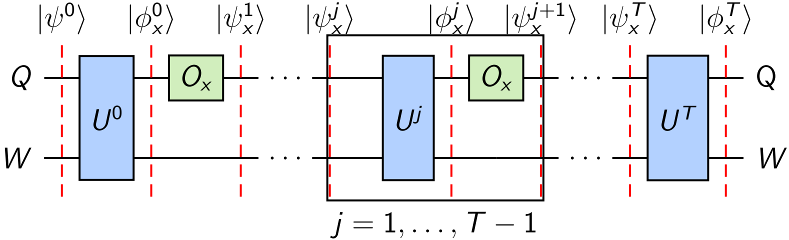
$$p\left[(|\psi_x\rangle)_{x\in D}\right]_\sim + p'\left[(|\psi'_x\rangle)_{x\in D}\right]_\sim := \left[\left(\sqrt{p}\,|\psi_x\rangle \oplus \sqrt{p'}\,|\psi'_x\rangle\right)_{x\in D}\right]_\sim$$

$$\hat{=} \left[\left(\sqrt{p}\,|0\rangle \otimes |\psi_x\rangle + \sqrt{p'}\,|1\rangle \otimes |\psi'_x\rangle\right)_{x\in D}\right]_\sim$$

(for classical: $\sqrt{p} \to p$, $\sqrt{p'} \to p'$).

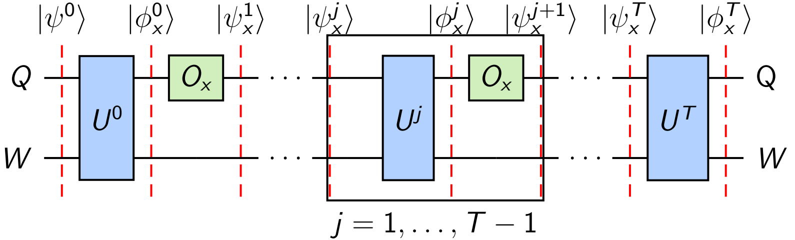
- well-defined, $+$ commutative, allows **convex combinations**
- For Gram matrices (pure-state quantum): Regular convex combinations
- If $G_1 \in \mathcal{S}^{+D}$ is reachable from $G_0 \in \mathcal{S}^{+D}$ in k queries, and $G'_1 \in \mathcal{S}^{+D}$ is reachable from G'_0 in k queries, then $pG_1 + p'G'_1$ is reachable from $pG_0 + p'G'_0$ in k queries as well by conditional execution

State conversion and the Gram matrix picture



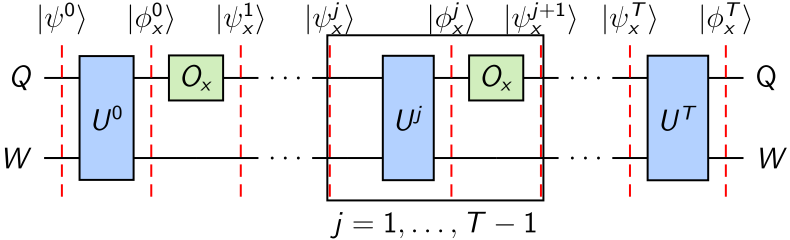
- To understand what function evaluation problems can be solved with error $1/3$ in T queries, we need to answer two questions:

State conversion and the Gram matrix picture



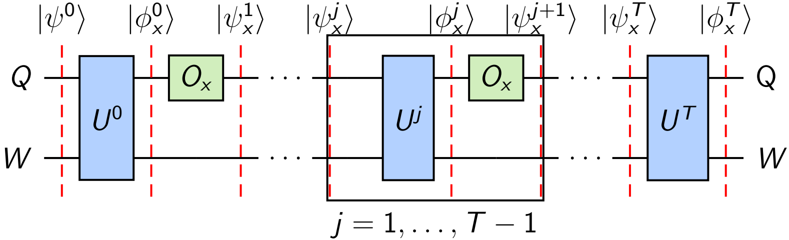
- To understand what function evaluation problems can be solved with error $1/3$ in T queries, we need to answer two questions:
  1. What combinations of final states $\left(|\phi_x^T\rangle\right)_{x\in D}$ are reachable in T queries?

State conversion and the Gram matrix picture



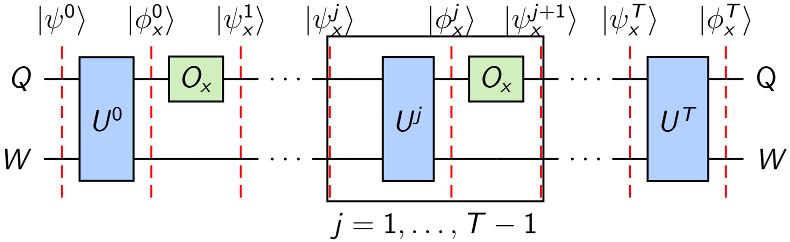
- To understand what function evaluation problems can be solved with error $1/3$ in T queries, we need to answer two questions:
  1. What combinations of final states $\left(|\phi_x^T\rangle\right)_{x\in D}$ are reachable in T queries?
  2. **Output condition:** What combinations of final states $\left(|\phi_x^T\rangle\right)_{x\in D}$ would permit an accurate measurement of some function f?

State conversion and the Gram matrix picture



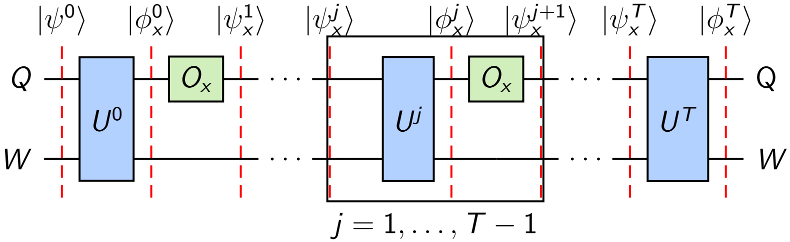
- To understand what function evaluation problems can be solved with error $1/3$ in T queries, we need to answer two questions:
  1. What combinations of final states $\left(|\phi_x^T\rangle\right)_{x\in D}$ are reachable in T queries?
  2. **Output condition:** What combinations of final states $\left(|\phi_x^T\rangle\right)_{x\in D}$ would permit an accurate measurement of some function f?
- **State conversion problem** (Lee et al. 2011 [1]): (How) can we convert initial states $\left(|\phi_x^0\rangle\right)_{x\in D}$ to final states $\left(|\phi_x^T\rangle\right)_{x\in D}$ in T queries?

State conversion and the Gram matrix picture



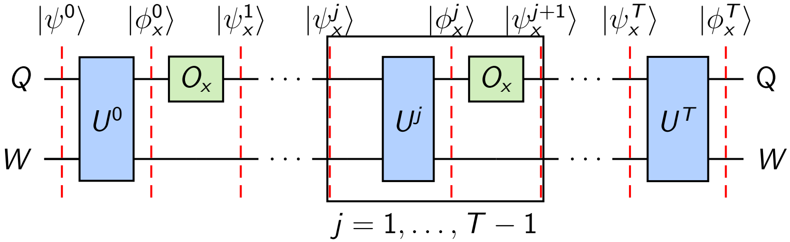
- Collection of states

State conversion and the Gram matrix picture



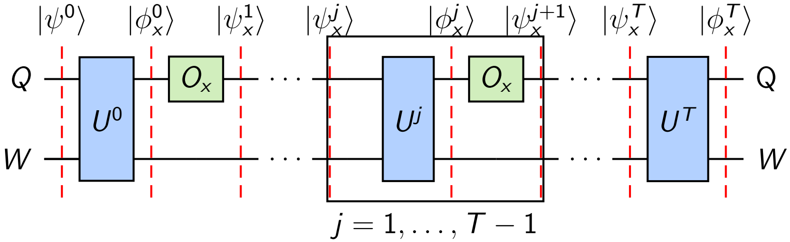
- Collection of states
  1. What combinations of final states $\left(|\phi_x^T\rangle\right)_{x \in D}$ are reachable in T queries?

State conversion and the Gram matrix picture



- Collection of states
  1. What combinations of final states $\left(|\phi_x^T\rangle\right)_{x\in D}$ are reachable in T queries?
  2. **Output condition:** What combinations of final states $\left(|\phi_x^T\rangle\right)_{x\in D}$ would permit an accurate measurement of some function f?

State conversion and the Gram matrix picture



- Collection of states
  1. What combinations of final states $\left(|\phi_x^T\rangle\right)_{x \in D}$ are reachable in T queries?
  2. **Output condition:** What combinations of final states $\left(|\phi_x^T\rangle\right)_{x \in D}$ would permit an accurate measurement of some function f?
- **State conversion problem** (Lee et al. 2011 [1]): (How) can we convert initial states $\left(|\phi_x^0\rangle\right)_{x \in D}$ to final states $\left(|\phi_x^T\rangle\right)_{x \in D}$ in T queries? ¡- focus on this

Adversary bound

- Now talking about Gram matrices, but this can be generalized to general states of knowledge as above

Adversary bound

- Now talking about Gram matrices, but this can be generalized to general states of knowledge as above
- Suppose that G_T can be reached from G_0 in T queries, i.e. $G_0 \to G_1$, $G_1 \to G_2$, $\ldots$, $G_{T-1} \to G_T$ is possible in 1 query each.

Adversary bound

- Now talking about Gram matrices, but this can be generalized to general states of knowledge as above
- Suppose that G_T can be reached from G_0 in T queries, i.e. $G_0 \to G_1$, $G_1 \to G_2$, ..., $G_{T-1} \to G_T$ is possible in 1 query each.
- Add initial/final Gram matrices (convexity): $\sum_{j=0}^{T-1} G_j \to \sum_{j=0}^{T-1} G_j + G_T - G_0$ is possible in 1 query as well, achieving a Gram matrix change of U_1

Adversary bound

- Now talking about Gram matrices, but this can be generalized to general states of knowledge as above
- Suppose that G_T can be reached from G_0 in T queries, i.e. $G_0 \to G_1$, $G_1 \to G_2$, …, $G_{T-1} \to G_T$ is possible in 1 query each.
- Add initial/final Gram matrices (convexity): $\sum_{j=0}^{T-1} G_j \to \sum_{j=0}^{T-1} G_j + G_T - G_0$ is possible in 1 query as well, achieving a Gram matrix change of U_1
- Let $\overline{G} := \sum_{j=0}^{T-1} G_j$. This is a Gram matrix as well (of direct sum of intermediate states) and fulfills

Adversary bound

- Now talking about Gram matrices, but this can be generalized to general states of knowledge as above
- Suppose that G_T can be reached from G_0 in T queries, i.e. $G_0 \to G_1$, $G_1 \to G_2$, ..., $G_{T-1} \to G_T$ is possible in 1 query each.
- Add initial/final Gram matrices (convexity): $\sum_{j=0}^{T-1} G_j \to \sum_{j=0}^{T-1} G_j + G_T - G_0$ is possible in 1 query as well, achieving a Gram matrix change of U_1
- Let $\overline{G} := \sum_{j=0}^{T-1} G_j$. This is a Gram matrix as well (of direct sum of intermediate states) and fulfills
  1. $\overline{G} \to \overline{G} + G_T - G_0$ possible in 1 query,

Adversary bound

- Now talking about Gram matrices, but this can be generalized to general states of knowledge as above
- Suppose that G_T can be reached from G_0 in T queries, i.e. $G_0 \to G_1$, $G_1 \to G_2$, ..., $G_{T-1} \to G_T$ is possible in 1 query each.
- Add initial/final Gram matrices (convexity): $\sum_{j=0}^{T-1} G_j \to \sum_{j=0}^{T-1} G_j + G_T - G_0$ is possible in 1 query as well, achieving a Gram matrix change of U_1
- Let $\overline{G} := \sum_{j=0}^{T-1} G_j$. This is a Gram matrix as well (of direct sum of intermediate states) and fulfills
  1. $\overline{G} \to \overline{G} + G_T - G_0$ possible in 1 query,
  2. Diagonal entries $\overline{G}[x, x] \leq T$.

Adversary bound

- Let $\overline{G} := \sum_{j=0}^{T-1} G_j$. Then $\overline{G}$ fulfills
  1. $\overline{G} \to \overline{G} + G_T - G_0$ possible in 1 query,
  2. Diagonal entries $\overline{G}[x,x] \leq T \langle \psi_x^0 | \psi_x^0 \rangle = T$.

Adversary bound

- Let $\overline{G} := \sum_{j=0}^{T-1} G_j$. Then $\overline{G}$ fulfills
  1. $\overline{G} \to \overline{G} + G_T - G_0$ possible in 1 query,
  2. Diagonal entries $\overline{G}[x, x] \leq T \langle \psi_x^0 | \psi_x^0 \rangle = T$.
- So the following is a lower bound on the number of queries needed to transform G_0 to G_T:

$$\text{minimize } \max_{x \in D} \overline{G}[x, x] \tag{2}$$

$$\text{subject to } \overline{G} \to \overline{G} + G_T - G_0 \text{ possible in 1 query} \tag{3}$$

$$\text{where } \overline{G} \text{ is GM of some state collection.} \tag{4}$$

Convex optimization duality

$$\text{minimize } \max_{x \in D} \overline{G}[x, x] \tag{5}$$

$$\text{subject to } \overline{G} \rightarrow \overline{G} + G_T - G_0 \text{ possible in 1 query} \tag{6}$$

$$\text{where } \overline{G} \text{ is GM of some state collection.} \tag{7}$$

Convex optimization duality

$$\text{minimize } \max_{x \in D} \overline{G}[x, x] \tag{5}$$

$$\text{subject to } \overline{G} \to \overline{G} + G_T - G_0 \text{ possible in 1 query} \tag{6}$$

$$\text{where } \overline{G} \text{ is GM of some state collection.} \tag{7}$$

- It turns out this is a **convex optimization problem**, more specifically a **semidefinite program**. By the theory of **convex duality**, the optimal value is equal to the optimal value of a related **maximization problem**.

Convex optimization duality

$$\text{minimize } \max_{x \in D} \overline{G}[x, x] \tag{5}$$
$$\text{subject to } \overline{G} \to \overline{G} + G_T - G_0 \text{ possible in 1 query} \tag{6}$$
$$\text{where } \overline{G} \text{ is GM of some state collection.} \tag{7}$$

- It turns out this is a **convex optimization problem**, more specifically a **semidefinite program**. By the theory of **convex duality**, the optimal value is equal to the optimal value of a related **maximization problem**.
- This means that finding any feasible solution to the maximization problem yields a lower bound on the number of queries needed.

Convex optimization duality

$$\text{minimize } \max_{x \in D} \overline{G}[x, x] \tag{5}$$

$$\text{subject to } \overline{G} \to \overline{G} + G_T - G_0 \text{ possible in 1 query} \tag{6}$$

$$\text{where } \overline{G} \text{ is GM of some state collection.} \tag{7}$$

- It turns out this is a **convex optimization problem**, more specifically a **semidefinite program**. By the theory of **convex duality**, the optimal value is equal to the optimal value of a related **maximization problem**.
- This means that finding any feasible solution to the maximization problem yields a lower bound on the number of queries needed.
- For classical probability theory/mixed-state quantum: Not so easy, because space is infinite-dimensional

(Newer) universal algorithm

- Now assume we have *some* valid $\overline{G}$ for the optimization problem (for which $\overline{G} \to \overline{G} + G_T - G_0$ is possible in 1 query), and try to turn it into an algorithm.

(Newer) universal algorithm

- Now assume we have *some* valid $\overline{G}$ for the optimization problem (for which $\overline{G} \to \overline{G} + G_T - G_0$ is possible in 1 query), and try to turn it into an algorithm.
- Assume also that "doing nothing in 1 query", i.e. transforming $G \to G$, is possible for all G.

(Newer) universal algorithm

- Now assume we have *some* valid $\overline{G}$ for the optimization problem (for which $\overline{G} \to \overline{G} + G_T - G_0$ is possible in 1 query), and try to turn it into an algorithm.
- Assume also that "doing nothing in 1 query", i.e. transforming $G \to G$, is possible for all G.
- For any integer $T' > 0$, we can transform $G_0 + \overline{G}/T'$ to $G_T + \overline{G}/T'$ in T' steps by going in a "straight line":

$$G_0 + \frac{\overline{G}}{T'} \to G_0 + \frac{\overline{G} + (G_T - G_0)}{T'}$$
$$\to G_0 + \frac{\overline{G} + 2(G_T - G_0)}{T'} \to \ldots \to G_T + \frac{\overline{G}}{T'}.$$

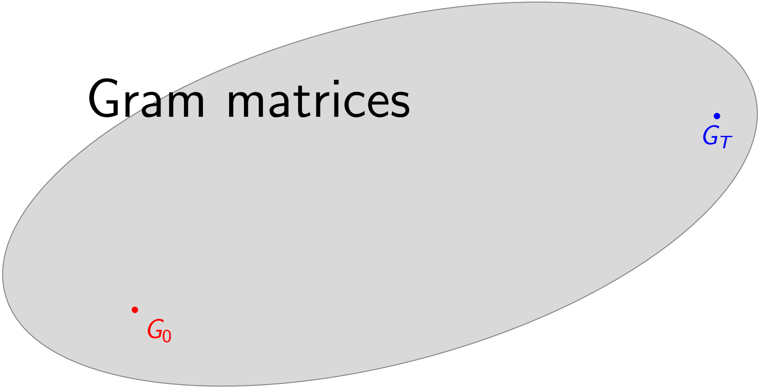
(Newer) universal algorithm

- Now assume we have *some* valid $\overline{G}$ for the optimization problem (for which $\overline{G} \to \overline{G} + G_T - G_0$ is possible in 1 query), and try to turn it into an algorithm.
- Assume also that "doing nothing in 1 query", i.e. transforming $G \to G$, is possible for all G.
- For any integer $T' > 0$, we can transform $G_0 + \overline{G}/T'$ to $G_T + \overline{G}/T'$ in T' steps by going in a "straight line":

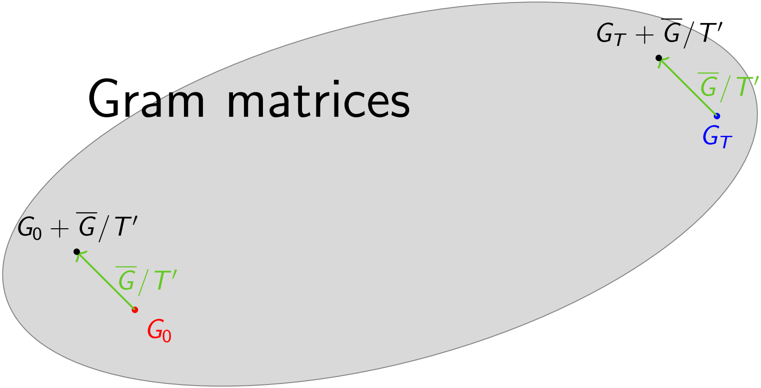
$$
\begin{aligned}
& G_0 + \frac{\overline{G}}{T'} \to G_0 + \frac{\overline{G} + (G_T - G_0)}{T'} \\
\to G_0 + \frac{\overline{G} + 2(G_T - G_0)}{T'} & \to \ldots \to G_T + \frac{\overline{G}}{T'}.
\end{aligned}
$$

- As $T' \to \infty$, initial and final states converge to G_0 or G_T.
$\to$ Approximate solution of state conversion problem $G_0 \to G_T$

Picture

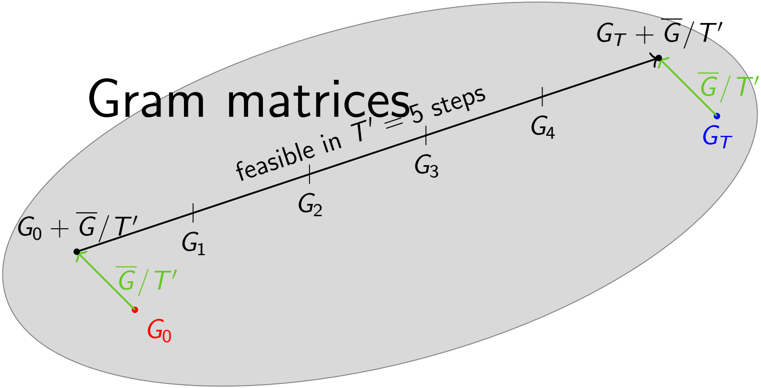


Picture



Gram matrices
$G_T + \overline{G}/T'$
$\overline{G}/T'$
G_T
$G_0 + \overline{G}/T'$
$\overline{G}/T'$
G_0

Picture



Gram matrices
feasible in $T' = 5$ steps
$G_T + \overline{G}/T'$
$\overline{G}/T'$
G_T
G_4
G_3
G_2
G_1
$G_0 + \overline{G}/T'$
$\overline{G}/T'$
G_0

Conclusion (Universal algorithm)

- Instead of as a sequence of states/transformations, we can understand a quantum (query) algorithm as a sequence of feasible matrices of inner products (Gram matrices) or equivalence classes of states (states of knowledge).

Conclusion (Universal algorithm)

- Instead of as a sequence of states/transformations, we can understand a quantum (query) algorithm as a sequence of feasible matrices of inner products (Gram matrices) or equivalence classes of states (states of knowledge).
- We may consider/optimize over a "coarse-grained version" of an algorithm, $\overline{G}$.

Conclusion (Universal algorithm)

- Instead of as a sequence of states/transformations, we can understand a quantum (query) algorithm as a sequence of feasible matrices of inner products (Gram matrices) or equivalence classes of states (states of knowledge).
- We may consider/optimize over a "coarse-grained version" of an algorithm, $\overline{G}$.
- Knowing $\overline{G}$ allows algorithms that approximate the desired target state.

Conclusion (Universal algorithm)

- Instead of as a sequence of states/transformations, we can understand a quantum (query) algorithm as a sequence of feasible matrices of inner products (Gram matrices) or equivalence classes of states (states of knowledge).
- We may consider/optimize over a "coarse-grained version" of an algorithm, $\overline{G}$.
- Knowing $\overline{G}$ allows algorithms that approximate the desired target state.
- Compared to previous universal algorithms: Generalizes to control algorithms, better prefactor, avoids some restrictions of previous approaches.

Conclusion (Universal algorithm)

- Instead of as a sequence of states/transformations, we can understand a quantum (query) algorithm as a sequence of feasible matrices of inner products (Gram matrices) or equivalence classes of states (states of knowledge).
- We may consider/optimize over a "coarse-grained version" of an algorithm, $\overline{G}$.
- Knowing $\overline{G}$ allows algorithms that approximate the desired target state.
- Compared to previous universal algorithms: Generalizes to control algorithms, better prefactor, avoids some restrictions of previous approaches.
- Idea of the new universal algorithm, compared to older ones based on states-and-unitaries:

Conclusion (Universal algorithm)

- Instead of as a sequence of states/transformations, we can understand a quantum (query) algorithm as a sequence of feasible matrices of inner products (Gram matrices) or equivalence classes of states (states of knowledge).
- We may consider/optimize over a "coarse-grained version" of an algorithm, $\overline{G}$.
- Knowing $\overline{G}$ allows algorithms that approximate the desired target state.
- Compared to previous universal algorithms: Generalizes to control algorithms, better prefactor, avoids some restrictions of previous approaches.
- Idea of the new universal algorithm, compared to older ones based on states-and-unitaries:
    1. Change initial states to more convenient ones in the "Gram matrix picture",

Conclusion (Universal algorithm)

- Instead of as a sequence of states/transformations, we can understand a quantum (query) algorithm as a sequence of feasible matrices of inner products (Gram matrices) or equivalence classes of states (states of knowledge).
- We may consider/optimize over a "coarse-grained version" of an algorithm, $\overline{G}$.
- Knowing $\overline{G}$ allows algorithms that approximate the desired target state.
- Compared to previous universal algorithms: Generalizes to control algorithms, better prefactor, avoids some restrictions of previous approaches.
- Idea of the new universal algorithm, compared to older ones based on states-and-unitaries:
  1. Change initial states to more convenient ones in the "Gram matrix picture",
  2. bound error incurred by that in "states-and-unitaries

Open problems

Open problems

- Find other query algorithms from "Gram matrix picture"

Open problems

- Find other query algorithms from "Gram matrix picture"
- Make SOK algebra for classical states lower-dimensional (e.g. by law of large numbers)

Open problems

- Find other query algorithms from "Gram matrix picture"
- Make SOK algebra for classical states lower-dimensional (e.g. by law of large numbers)
- Applications, e.g. re-prove quantum Chernoff bound

Open problems

- Find other query algorithms from "Gram matrix picture"
- Make SOK algebra for classical states lower-dimensional (e.g. by law of large numbers)
- Applications, e.g. re-prove quantum Chernoff bound
- Any ideas? Let's have another call :)
- fi1w+qudent@outlook.com
- Thanks to Alexander Belov

Bonus: Algebra of SOKs/Formalizing reachability

$$\left[(|\psi_{x,q}\rangle)_{x\in D}\right]_{\sim} + \left[(|\psi'_x\rangle)_{x\in D}\right]_{\sim} := \left[(|\psi_x\rangle \oplus |\psi'_x\rangle)_{x\in D}\right]_{\sim}$$
$$\left[(|\psi_x\rangle)_{x\in D}\right]_{\sim} * \left[(|\psi'_x\rangle)_{x\in D}\right]_{\sim} := \left[(|\psi_x\rangle \otimes |\psi'_x\rangle)_{x\in D}\right]_{\sim}$$

- Well-defined, commutative, associative and distributive on the equivalence classes.
- On Gram matrices: $+*$ are entrywise sums/products
- Suppose our oracle O_x just emits a new state in each query, i.e. $O_x = |\theta_x\rangle$, maps $|\psi_x\rangle \to |\theta_x\rangle \otimes |\psi_x\rangle$
- This corresponds to transforming SOKs/Gram matrices $G_\psi \to G_\psi * G_\theta$

Bonus: Algebra of SOKs/Formalizing reachability

$$\sum_{q\in Q}\left[(|\psi_{x,q}\rangle)_{x\in D}\right]_\sim := \left[\left(\sum_{q\in Q}|q\rangle\otimes|\psi_x\rangle\right)_{x\in D}\right]_\sim$$

$$\left[(|\psi_x\rangle)_{x\in D}\right]_\sim * \left[(|\psi'_x\rangle)_{x\in D}\right]_\sim := \left[(|\psi_x\rangle\otimes|\psi'_x\rangle)_{x\in D}\right]_\sim$$

- Suppose our oracle O_x emits a new state in each query based on query, i.e. maps $O_x = \sum_{q\in Q}|q\rangle\langle q|\otimes|\theta_{x,q}\rangle$
- Then states reachable by reversible transformations are

$$\left\{\sum_{q\in Q} G_{\psi,q} * G_{\theta,q} \mid \sum_{q\in Q} G_{\psi,q} = G_\psi\right\}$$

Bonus: Algebra of SOKs/Formalizing reachability

$$\sum_{q \in Q} \left[(|\psi_{x,q}\rangle)_{x \in D} \right]_\sim := \left[\left(\sum_{q \in Q} |q\rangle \otimes |\psi_x\rangle \right)_{x \in D} \right]_\sim$$

$$\left[(|\psi_x\rangle)_{x \in D} \right]_\sim * \left[(|\psi'_x\rangle)_{x \in D} \right]_\sim := \left[(|\psi_x\rangle \otimes |\psi'_x\rangle)_{x \in D} \right]_\sim$$

- Suppose our oracle O_x emits a new state in each query based on query, i.e. maps $O_x = \sum_{q \in Q} |q\rangle \langle q| \otimes |\theta_{x,q}\rangle$
- Then states reachable by reversible transformations are

$$\left\{ \sum_{q \in Q} G_{\psi,q} * G_{\theta,q} \mid \sum_{q \in Q} G_{\psi,q} = G_\psi \right\}$$

- For control theory: need to generalize partial trace of Gram matrices

References

- Barnum-Saks-Szegedy "Quantum query complexity and semi-definite programming": does this specifically for Gram matrices/pure-state quantum query algorithms
- https://arxiv.org/abs/2212.04606 formalizes the algebra of knowledge in detail (draft)
- https://arxiv.org/abs/2211.16293 discusses the universal algorithm for control theory in the Gram matrix picture (equivalent to reduced density matrices)
- Belovs-Y. https://arxiv.org/abs/2301.02003 discusses the algorithm in a more "traditional" way as part of the introduction of "Las Vegas complexity"

Troy Lee, Rajat Mittal, Ben W Reichardt, Robert Špalek, and Mario Szegedy.
Quantum query complexity of state conversion.
In *2011 IEEE 52nd Annual Symposium on Foundations of Computer Science*, pages 344–353. IEEE, 2011.