Quantum Algorithms for Finite-horizon Markov Decision Processes

Bin Luo (Robin)
Supervisor: **John C.S. Lui**
April 28, 2025

香港中文大學
The Chinese University of Hong Kong

Table of Contents

1 Introduction

▶ Introduction

▶ Preliminaries

▶ Exact Dynamics Setting

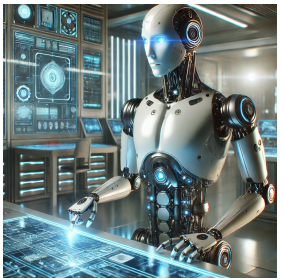
▶ Generative Model Setting

▶ Conclusion

▶ Reference

- Markov Decision Process (MDP) is a framework used for modeling decision-making in various environments. They are capable of obtaining optimal or near-optimal policies in a stochastic dynamic.

(a) Autonomous driving



(b) Robotics

(c) Operation research

(d) Reinforcement learning

Figure: Applications of MDP in different areas.

The Challenge of MDPs

- Curse of dimensionality will occur when the number of possible states in the system grows exponentially with the number of variables or components being modeled.

Figure: Autonomous driving

In the autonomous driving, we may need to consider

- vehicle position
- velocity
- orientation
- weather outside the car
- positions and velocities of other vehicles
- ...

If each variable has n possible values, the total size of the state space S grows as n^d, where d is the number of state variables.

- The time complexity of the classical algorithm becomes exponential in d.

For certain problems, quantum computing demonstrates a significant speedup over classical computing in terms of time complexity.

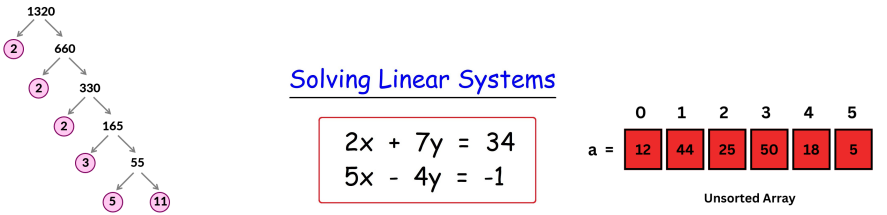
(a). factorizing an integer N: quantum $O(\log N)$ vs. classical $O\big(\exp\big(1.9(\log N)^{1/3}\big)(\log\log N)^{2/3}\big)$;

(b). solving a system of N linear equations: quantum $O(\log N)$ vs. classical $\Omega(N)$;

— Suppose $N = 2^{20}$: Quantum: $\approx$ 20 hours vs. Classical: $\approx$ 119.7 years!

(c). unstructured search within N items: quantum $\Theta(\sqrt{N})$ vs. classical $O(N)$.

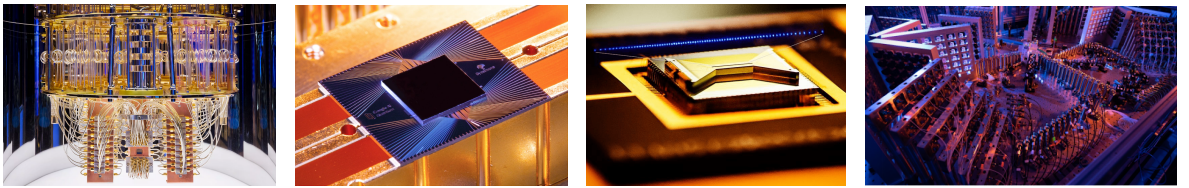
— Suppose $N = 1,000,000$: Quantum: 1000 hours $\approx$ 42 days vs. Classical: $1,000,000$ hours $\approx$ 114 years!



(a) Integer factorization (b) Solving linear systems (c) Unstructured search

Figure: A small set of problems that can show quantum supremacy.

- Quantum computers exploit quantum-mechanical phenomena, such as superposition and entanglement, to perform computation.
  - Google's Willow: It takes less than 5 minutes to finish random circuit sampling (RCS) task.
  - Classical supercomputer: 10^{25} years!



(a) IBM Condor (b) Google Willow (c) IonQ Forte (d) USTC Jiuzhang

Figure: The most advanced quantum computers/chips in the world.

Quantum for MDPs

Many researchers have explored various quantum algorithms to reduce the time complexity of solving MDPs.

- Lack a concrete quantum algorithm/rigorious theoretical analysis;
- Only apply for a specific class of finite-horizon MDPs;
- Require exponential time complexity for general finite-horizon MDPs problem;
- Only tailored to infinite-horizon problems with a time-invariant value function.
  - infinite-horizon MDPs: The process continues indefinitely vs. Finite-horizon MDPs: The process ends at a finite and fixed number of time steps.
  - Time-dependent MDPs: The environment changes as time progresses vs. Time-independent MDPs: The environment is consistent across the time.

Quantum for MDPs

Many researchers have explored various quantum algorithms to reduce the time complexity of solving MDPs.

- Lack a concrete quantum algorithm/rigorious theoretical analysis;
- Only apply for a specific class of finite-horizon MDPs;
- Require exponential time complexity for general finite-horizon MDPs problem;
- Only tailored to infinite-horizon problems with a time-invariant value function.
  - infinite-horizon MDPs: The process continues indefinitely vs. Finite-horizon MDPs: The process ends at a finite and fixed number of time steps.
  - Time-dependent MDPs: The environment changes as time progresses vs. Time-independent MDPs: The environment is consistent across the time.

Can one design quantum algorithms that are more efficient than classical algorithms in solving general "time-dependent" and "finite-horizon" MDPs?

Quantum for MDPs

Many researchers have explored various quantum algorithms to reduce the time complexity of solving MDPs.

- Lack a concrete quantum algorithm/rigorious theoretical analysis;
- Only apply for a specific class of finite-horizon MDPs;
- Require exponential time complexity for general finite-horizon MDPs problem;
- Only tailored to infinite-horizon problems with a time-invariant value function.
  - infinite-horizon MDPs: The process continues indefinitely vs. Finite-horizon MDPs: The process ends at a finite and fixed number of time steps.
  - Time-dependent MDPs: The environment changes as time progresses vs. Time-independent MDPs: The environment is consistent across the time.

Can one design quantum algorithms that are more efficient than classical algorithms in solving general "time-dependent" and "finite-horizon" MDPs?

Yes!

- Exact dynamics setting: The environment's dynamics is fully known.
- Generative model setting: The environment's dynamics is unknown.

Table of Contents

2 Preliminaries

- ▶ Introduction
- ▶ Preliminaries
- ▶ Exact Dynamics Setting
- ▶ Generative Model Setting
- ▶ Conclusion
- ▶ Reference

MDP Preliminaries

2 Preliminaries

We define a time-dependent and finite-horizon MDP as a $5-$tuple $\mathcal{M} = (\mathcal{S}, \mathcal{A}, \{P_h\}_{h=0}^{H-1}, \{r_h\}_{h=0}^{H-1}, H)$.

- State space $\mathcal{S}$ and action space $\mathcal{A}$ are discrete and finite sets.
- The total time step H is a finite positive integer.
- $P_h(s_{h+1}|s_h, a_h)$ is a transition probability.
  - Fix h, s_h and a_h, one can view $P_h(s_{h+1}|s_h, a_h)$ as a vector $P_{h|s_h,a_h}(s_{h+1})$.
- A reward $r_h(s_h, a_h)$ is a scalar in $[0, 1]$.

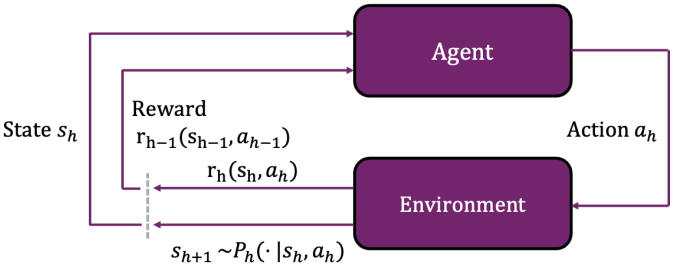


Figure: An abstract illustration of time-dependent and finite-horizon MDP dynamics.

Optimization goal:

- A policy π is a mapping from $\mathcal{S} \times [H]$ to $\mathcal{A}$, where $[H] \coloneqq \{0, 1, \ldots, H-1\}$.
- The policy space is defined as $\Pi \coloneqq \mathcal{A}^{\mathcal{S} \times [H]}$.
- Find a policy π that maximizes the expected cumulative reward (V-value function) over H time horizon for an initial state $s \in \mathcal{S}$.

$$\underset{\pi \in \Pi}{\operatorname{argmax}}\, V_h^{\pi}(s) = \mathbb{E}\Big[\sum_{t=h}^{H-1} r_t(s_t, a_t) | \pi, s_h = s\Big]. \tag{1}$$

Optimization goal:

- A policy π is a mapping from $\mathcal{S} \times [H]$ to $\mathcal{A}$, where $[H] \coloneqq \{0, 1, \ldots, H-1\}$.
- The policy space is defined as $\Pi \coloneqq \mathcal{A}^{\mathcal{S} \times [H]}$.
- Find a policy π that maximizes the expected cumulative reward (V-value function) over H time horizon for an initial state $s \in \mathcal{S}$.

$$\operatorname*{argmax}_{\pi \in \Pi} V_h^\pi(s) = \mathbb{E}\Big[\sum_{t=h}^{H-1} r_t(s_t, a_t) | \pi, s_h = s\Big]. \tag{1}$$

- Define the optimal value of an initial state $s \in \mathcal{S}$ at each time step $h \in [H]$ of the finite-horizon MDP $\mathcal{M}$ as $V_h^*(s) \coloneqq \max_{\pi \in \Pi} V_h^\pi(s)$.
- A policy π is an optimal policy π^* if $V_0^\pi = V_0^*$.
- Similarly, Q-value function $Q_h^\pi : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$ is defined as

$$Q_h^\pi(s, a) \coloneqq \mathbb{E}\Big[\sum_{t=h}^{H-1} r_t(s_t, a_t) \Big| \pi, s_h = s, a_h = a\Big], \tag{2}$$

and $Q_h^*(s, a) \coloneqq \max_{\pi \in \Pi} Q_h^\pi(s, a)$.

MDP Preliminaries: Finding the Shortest Path in a Maze

- States: Positions in the maze.
- Actions: Movements (up, down, left, right).
- Transition probabilities: It captures how reliable the robot's movements are.
- Reward function: $r_h(s_h, a_h) = 0$ if s_h is the exit; otherwise, $r_h(s_h, a_h) = -1$.
- Total time horizon: The total number of time steps the robot is allowed to act before the game ends.
- Optimization goal: Find a policy $\pi \in \Pi$ that minimizes the expected number of steps to reach the exit.

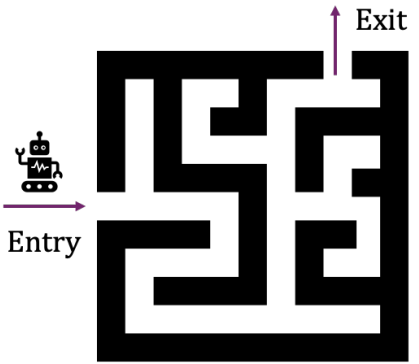


Figure: Robot-in-Maze Example: Find the shortest path.

Qubits (Quantum Bits)

- A qubit $|\psi\rangle$ is the basic unit of quantum information (vs. classical bit 0 or 1).
- Superposition property: $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle = \alpha\begin{bmatrix}1\\0\end{bmatrix} + \beta\begin{bmatrix}0\\1\end{bmatrix}$, where $\alpha, \beta \in \mathbb{C}$ are amplitudes satisfying $|\alpha|^2 + |\beta|^2 = 1$.
- Measurement: observe $|0\rangle$ or $|1\rangle$ with $|\alpha|^2$ or $|\beta|^2$ probability.

Unitary Operators

- Quantum computations are performed using unitary operators U, where $U^\dagger U = I$.
- Example: Hadamard gate ($H = \frac{1}{\sqrt{2}}\begin{bmatrix}1 & 1\\1 & -1\end{bmatrix}$);

$$H|0\rangle = \frac{1}{\sqrt{2}}\begin{bmatrix}1 & 1\\1 & -1\end{bmatrix}\begin{bmatrix}1\\0\end{bmatrix} = \frac{1}{\sqrt{2}}\begin{bmatrix}1\\1\end{bmatrix} = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle).$$

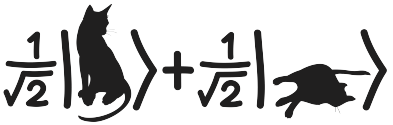


Figure: A cat that is 50% likely dead and 50% likely alive.

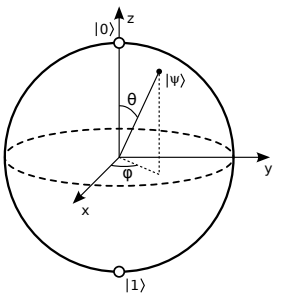


Figure: A geometrical representation of a qubit: bloch sphere.

How to encode a real number in quantum computing?

- For any non-negative real number k, the fixed-point binary representation of k would be written as

$$\mathrm{Bi}[k] = k_1 2^{q-p-1} + \cdots k_{q-p} 2^0 + k_{q-p+1} 2^{-1} + \cdots + k_q 2^{-p} = k_1 k_2 \cdots k_{q-p}.k_{q-p+1} \cdots k_q,$$

  where $k_i \in \{0, 1\}$ for all $1 \leq i \leq q$.
  - Example: When $q = 7, p = 4$, then $\mathrm{Bi}[5.75] = 101.1100$.
- Then we encode the real number k with q qubits based on $\mathrm{Bi}[k]$ and write it as

$$|\mathrm{Bi}[k]\rangle_q = |k_1\rangle |k_2\rangle \cdots |k_q\rangle \in \mathbb{C}^{2^q}.$$

  For simplicity, we often omit the index q when writing the ket.
  - Example: $|\mathrm{Bi}[5.75]\rangle = |1\rangle \otimes |0\rangle \otimes |1\rangle \otimes |1\rangle \otimes |1\rangle \otimes |0\rangle \otimes |0\rangle = |1\rangle |0\rangle |1\rangle |1\rangle |1\rangle |0\rangle |0\rangle$.
- We assume that q and p are large enough for the problem we consider so that there is no overflow in storing real number.

How to encode a series of real numbers in quantum computing?

Definition (Quantum oracle for functions and vectors)

Let Ω be a finite set of size N and $f \in \mathbb{R}^{\Omega}$ (equivalently $f : \Omega \to \mathbb{R}$) where each entry of f is represented with a precision of 2^{-p}. A quantum oracle encoding f is a unitary matrix $B_f : \mathbb{C}^N \otimes \mathbb{C}^{2^q} \to \mathbb{C}^N \otimes \mathbb{C}^{2^q}$ such that

$$B_f : |i\rangle \otimes |0\rangle \mapsto |i\rangle \otimes |\mathrm{Bi}[f(i)]\rangle \tag{3}$$

for all $i \in [N]$, where $\mathrm{Bi}[f(i)]$ is the binary representation of $f(i)$ with precision 2^{-p}.

- B_f is often referred to as a binary oracle for the function/vector f.

Table of Contents

3 Exact Dynamics Setting

- ▶ Introduction
- ▶ Preliminaries
- ▶ Exact Dynamics Setting
- ▶ Generative Model Setting
- ▶ Conclusion
- ▶ Reference

Under this setting, it is assumed that the dynamics of the environment is fully known to the agent.

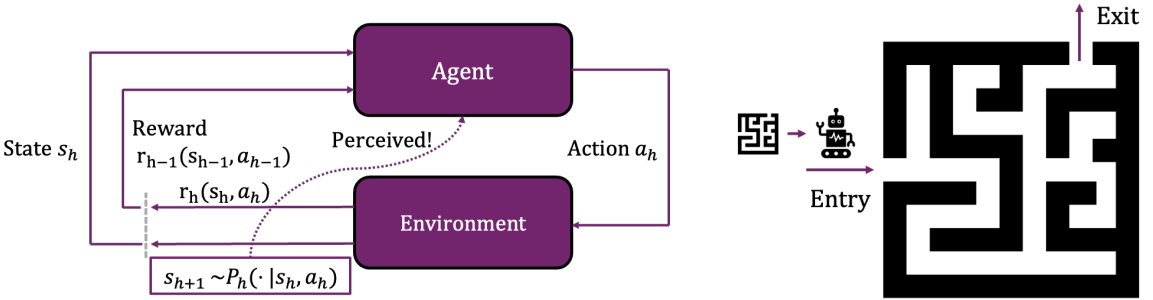


Figure: An illustration and an example of time-dependent and finite-horizon MDP dynamics in the exact dynamics setting.

Under this setting, it is assumed that the dynamics of the environment is fully known to the agent.

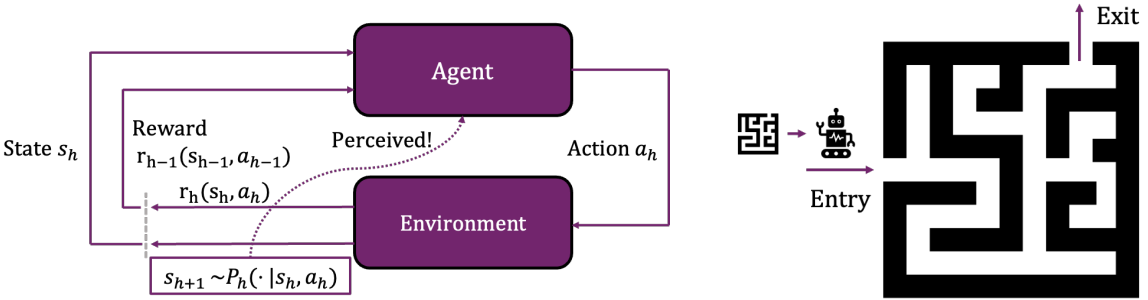


Figure: An illustration and an example of time-dependent and finite-horizon MDP dynamics in the exact dynamics setting.

Definition (Classical oracle of time-dependent and finite-horizon MDP)

We define a classical oracle $O_{\mathcal{M}} : \mathcal{S} \times \mathcal{A} \times [H] \times \mathcal{S} \to [0,1] \times [0,1]$ for time-dependent and finite-horizon MDPs

$$O_{\mathcal{M}} : (s, a, h, s') \mapsto \big(r_h(s,a), P_{h|s,a}(s')\big). \tag{4}$$

The Bellman optimality value operator $\mathcal{T}^h : \mathbb{R}^{\mathcal{S}} \to \mathbb{R}^{\mathcal{S}}$ is defined as

$$[\mathcal{T}^h(V_{h+1})]_s := \max_{a \in \mathcal{A}} \{r_h(s,a) + P_{h|s,a}^{\mathrm{T}} V_{h+1}\}. \tag{5}$$

Theorem: Bellman Optimality Equations [Bellman, 1957]

Suppose that $V_H = \mathbf{0}$. The V-value functions satisfy $V_h = V_h^*$ for all $h \in [H]$ if and only if:

$$V_h = \mathcal{T}^h(V_{h+1}), \quad \forall h \in [H]. \tag{6}$$

Furthermore, the policy:

$$\pi(s,h) = \operatorname*{argmax}_{a \in \mathcal{A}} \left\{ r_h(s,a) + P_{h|s,a}^{\mathrm{T}} V_{h+1} \right\} \tag{7}$$

is an optimal policy.

Classical Algorithm for Finite-horizon MDPs

Algorithm 1 Value Iteration (Backward Induction) Algorithm for Finite Horizon MDPs [Bellman, 1957]

Require: MDP $\mathcal{M}$.
Initialize: $V_H \leftarrow \mathbf{0}$
for $h := H-1, \ldots, 0$ **do**
  for each $s \in \mathcal{S}$ **do**
    for each $a \in \mathcal{A}$ **do**
      $Q_h(s,a) = r_h(s,a) + \sum_{s' \in \mathcal{S}} P_{h|s,a}(s') V_{h+1}(s')$
    end for
    $\pi(s,h) = \operatorname{argmax}_{a \in \mathcal{A}} Q_h(s,a)$
    $V_h(s) = Q_h\big(s, \pi(s,h)\big)$
  end for
end for
Return: π, V_0

Classical Algorithm for Finite-horizon MDPs

Definition (Classical oracle of time-dependent and finite-horizon MDP)

We define a classical oracle $O_{\mathcal{M}} : \mathcal{S} \times \mathcal{A} \times [H] \times \mathcal{S} \to [0,1] \times [0,1]$ for time-dependent and finite-horizon MDPs

$$O_{\mathcal{M}} : (s, a, h, s') \mapsto \big(r_h(s,a), P_{h|s,a}(s')\big). \tag{8}$$

- The classical value iteration algorithm requires

$$O(S^2AH) \tag{9}$$

queries to the oracle $O_{\mathcal{M}}$.
  - Taking maximum over the whole action space: $O(A)$.
  - Computing the inner product $P_{h|s,a}^{\mathrm{T}} V_{h+1}$: $O(S)$.
  - Updating all the values in V_h: $O(S)$.
  - Updating H time horizons: $O(H)$.
- Assuming that it takes $O(1)$ time to query the oracle $O_{\mathcal{M}}$ once, the time complexity of the classical value iteration algorithm is $O(S^2AH)$.

Classical Algorithm for Finite-horizon MDPs

Definition (Classical oracle of time-dependent and finite-horizon MDP)

We define a classical oracle $O_{\mathcal{M}} : \mathcal{S} \times \mathcal{A} \times [H] \times \mathcal{S} \to [0,1] \times [0,1]$ for time-dependent and finite-horizon MDPs

$$O_{\mathcal{M}} : (s, a, h, s') \mapsto \big(r_h(s,a), P_{h|s,a}(s')\big). \tag{8}$$

- The classical value iteration algorithm requires

$$O(S^2AH) \tag{9}$$

  queries to the oracle $O_{\mathcal{M}}$.
  - Taking maximum over the whole action space: $O(A)$.
  - Computing the inner product $P_{h|s,a}^{\mathrm{T}} V_{h+1}$: $O(S)$.
  - Updating all the values in V_h: $O(S)$.
  - Updating H time horizons: $O(H)$.
- Assuming that it takes $O(1)$ time to query the oracle $O_{\mathcal{M}}$ once, the time complexity of the classical value iteration algorithm is $O(S^2AH)$.

Can we design a quantum algorithm to reduce the time complexity of solving finite-horizon MDP, i.e., computing an optimal policy π and optimal V-value function V_0^*?

- Note that quantum computation are performed using unitary operators!

Definition (Classical oracle of time-dependent and finite-horizon MDP)

We define a classical oracle $O_{\mathcal{M}} : \mathcal{S} \times \mathcal{A} \times [H] \times \mathcal{S} \to [0,1] \times [0,1]$ for time-dependent and finite-horizon MDPs

$$O_{\mathcal{M}} : (s, a, h, s') \mapsto \big(r_h(s,a), P_{h|s,a}(s')\big). \tag{10}$$

Definition (Quantum oracle of time-dependent and finite-horizon MDP)

Let $\mathcal{M}$ be a time-dependent and finite-horizon MDP. A quantum oracle of such an MDP is a unitary matrix $O_{\mathcal{QM}} : \mathbb{C}^S \otimes \mathbb{C}^A \otimes \mathbb{C}^H \otimes \mathbb{C}^S \otimes \mathbb{C}^{2^q} \otimes \mathbb{C}^{2^q} \to \mathbb{C}^S \otimes \mathbb{C}^A \otimes \mathbb{C}^H \otimes \mathbb{C}^S \otimes \mathbb{C}^{2^q} \otimes \mathbb{C}^{2^q}$ such that

$$O_{\mathcal{QM}} : |s\rangle |a\rangle |h\rangle |s'\rangle |0\rangle |0\rangle \mapsto |s\rangle |a\rangle |h\rangle |s'\rangle |\mathrm{Bi}[r_h(s,a)]\rangle |\mathrm{Bi}[P_{h|s,a}(s')]\rangle , \tag{11}$$

for all $(s, a, h, s') \in \mathcal{S} \times \mathcal{A} \times [H] \times \mathcal{S}$, where $\mathrm{Bi}[r_h(s,a)]$ and $\mathrm{Bi}[P_{h|s,a}(s')]$ denote the fixed-point binary representation of $r_h(s,a)$ and $P_{h|s,a}(s')$.

Quantum Maximum Searching Algorithm

3 Exact Dynamics Setting

- Problem Formulation: For an unsorted vector $f \in \mathbb{R}^N$, one wants to find the index i such that $f(i) = \max_{j \in [N]} f(j)$.

Quantum Maximum Searching Algorithm

- Problem Formulation: For an unsorted vector $f \in \mathbb{R}^N$, one wants to find the index i such that $f(i) = \max_{j \in [N]} f(j)$.
- Classical algorithm: $\Theta(N)$ queries to the vector f.
- Quantum maximum searching algorithm [Durr and Hoyer, 1999]: $\Theta(\sqrt{N})$ queries to a quantum oracle B_f!
  - Suppose $N = 1,000,000$: Quantum: $\approx$ 42 days vs. Classical: $\approx$ 114 years!
- We use $\mathbf{QMS}_\delta\{f(i) : i \in [N]\}$ to denote the process of finding the index of the maximum value of a vector f with a success probability at least $1 - \delta$.

Revisit the Classical Value Iteration Algorithm

Algorithm 2 Value Iteration (Backward Induction) Algorithm for Finite Horizon MDPs [Bellman, 1957]

Require: MDP $\mathcal{M}$.
Initialize: $V_H \leftarrow \mathbf{0}$
for $h := H-1, \ldots, 0$ **do**
  for each $s \in \mathcal{S}$ **do**
    for each $a \in \mathcal{A}$ **do**
      $Q_h(s,a) = r_h(s,a) + \sum_{s' \in \mathcal{S}} P_{h|s,a}(s') V_{h+1}(s')$
    end for
    $\pi(s,h) = \operatorname{argmax}_{a \in \mathcal{A}} Q_h(s,a)$ ▷ Can we incorporate **QMS** in this step?
    $V_h(s) = Q_h\big(s, \pi(s,h)\big)$
  end for
end for
Return: π, V_0

Algorithm 3 Quantum Value Iteration Algorithm **QVI-1**$(\mathcal{M}, \delta)$

Require: MDP $\mathcal{M}$, quantum oracle $O_{\mathcal{QM}}$, maximum failure probability $\delta \in (0, 1)$.
Initialize: $\zeta \leftarrow \delta/(SH)$, $\hat{V}_H \leftarrow \mathbf{0}$.
for $h := H-1, \ldots, 0$ **do**
  create a quantum oracle $B_{\hat{V}_{h+1}}$ for vector $\hat{V}_{h+1} \in \mathbb{R}^{\mathcal{S}}$
  $\forall s \in \mathcal{S}$: create a quantum oracle $B_{\hat{Q}_{h,s}}$ encoding vector $\hat{Q}_{h,s} \in \mathbb{R}^{\mathcal{A}}$ with $O_{\mathcal{QM}}$ and $B_{\hat{V}_{h+1}}$ satisfying
    $\hat{Q}_{h,s}(a) \leftarrow r_h(s,a) + P_{h|s,a}^{\mathrm{T}} \hat{V}_{h+1}$
  $\forall s \in \mathcal{S}$: $\hat{\pi}(s,h) \leftarrow$ **QMS**$_\zeta\{\hat{Q}_{h,s}(a) : a \in \mathcal{A}\}$ ▷ We apply **QMS** now!
  $\forall s \in \mathcal{S}$: $\hat{V}_h(s) \leftarrow \hat{Q}_{h,s}\big(\hat{\pi}(s,h)\big)$
end for
Return: $\hat{\pi}$, $\hat{V}_0$

Theoretical Analysis on QVI-1$(\mathcal{M}, \delta)$

Precise case

Theorem (Correctness of QVI-1)

The outputs $\hat{\pi}$ and $\hat{V}_0$ satisfy that $\hat{\pi} = \pi^*$ and $\hat{V}_0 = V_0^*$ with a success probability at least $1 - \delta$.

- **QVI-1** can obtain optimal policy and V-value function.

Theorem (Complexity of QVI-1)

*The quantum query complexity of **QVI-1** in terms of the quantum oracle of MDPs $O_{\mathcal{QM}}$ is*

$$O(S^2\sqrt{A}H\log(SH/\delta)).$$

- Classical value iteration algorithm: $O(S^2AH)$

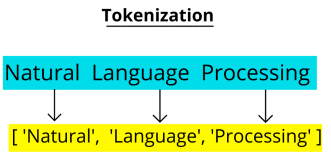
Potential Problems in QVI-1

QVI-1 is advantageous for problems with a large action space.

- Natural language processing (NLP): Each text in a large dictionary corresponds to a distinct action.

For the problems that have large state spaces, **QVI-1** become infeasible, because of its complexity of $O(S^2)$.

- Chess or Go: Each position in a vast board is represented as a state.
- Computing the inner product $P_{h|s,a}^{\mathrm{T}} \hat{V}_{h+1}$: $O(S)$.
- Updating all values in $\hat{V}_h$: $O(S)$.



(a) NLP

(b) Chess

(c) Go

Figure: Applications of **QVI-1**.

Improvement on QVI-1

Observation: for obtaining an "ϵ-*estimation of the mean*" of n Boolean variables, quantum algorithms only need $\Theta(\min\{\epsilon^{-1}, n\})$ queries to a binary oracle [Nayak and Wu, 1999, Beals et al., 2001].

- A quantum speedup is possible when estimating inner product $P_{h|s,a}^{\mathrm{T}} \hat{V}_{h+1}$.
- We can only obtain a near-optimal policy.

Improvement on QVI-1

Observation: for obtaining an "ϵ-*estimation of the mean*" of n Boolean variables, quantum algorithms only need $\Theta(\min\{\epsilon^{-1}, n\})$ queries to a binary oracle [Nayak and Wu, 1999, Beals et al., 2001].

- A quantum speedup is possible when estimating inner product $P_{h|s,a}^{\mathrm{T}} \hat{V}_{h+1}$.
- We can only obtain a near-optimal policy.

Question

Does there exist an error-bounded quantum algorithm that can obtain ϵ-optimal policy $\hat{\pi}$ and ϵ-optimal values $\{\hat{V}_h\}_{h=0}^{H-1}$ for an MDP $\mathcal{M}$ but only requires

$$\tilde{O}\left(S^c \mathrm{poly}(\sqrt{A}, H, 1/\epsilon)\right) \tag{12}$$

queries to the quantum oracle $O_{\mathcal{QM}}$, where $0 < c < 2$?

Definition (ϵ-optimal value and policy)

- We define values $\{V_h\}_{h=0}^{H-1}$ are ϵ-optimal if $\left\|V_h^* - V_h\right\|_\infty \leq \epsilon$ for all $h \in [H]$.
- A policy π is ϵ-optimal if $\left\|V_h^* - V_h^\pi\right\|_\infty \leq \epsilon$.

Can we use existing quantum mean estimation algorithms [Montanaro, 2015, Cornelissen et al., 2022]?

- They require a probability oracle U_p that encodes the probability distribution in the amplitude.
- We only have a binary oracle $O_{\mathcal{QM}}$ that encodes the probability distribution in the ket $|\cdot\rangle$.

Definition (Quantum oracle for probability distribution)

Let Ω be a finite set of size N and $p = (p_x)_{x\in\Omega}$ a discrete probability distribution on Ω. A quantum oracle encoding a probability distribution p is a unitary matrix $U_p : \mathbb{C}^N \otimes \mathbb{C}^J \to \mathbb{C}^N \otimes \mathbb{C}^J$ such that

$$U_p : |0\rangle \otimes |0\rangle \mapsto \sum_{x\in\Omega} \sqrt{p_x}\,|x\rangle \otimes |w_x\rangle\,, \tag{13}$$

where $0 \leq J \in \mathbb{Z}$ is arbitrary and $|w_x\rangle \in \mathbb{C}^J$ are arbitrary junk state.

Theorem (Quantum Mean Estimation with Binary Oracle)

Let Ω be a finite set with cardinality N, $p = (p_x)_{x\in\Omega}$ a discrete probability distribution over Ω, and $f : \Omega \to \mathbb{R}$ a function. Suppose we have access to

- *a binary oracle B_p encoding the probability distribution p,*
- *a binary oracle B_f encoding the function f.*

*If the function f satisfies $f(x) \in [0, 1]$ for all $x \in \Omega$, then the algorithm **QMEBO** requires $O((\frac{\sqrt{N}}{\epsilon} + \sqrt{\frac{N}{\epsilon}})\log(1/\delta))$ queries to B_p and B_f to put an estimate $\hat{\mu}$ of*

$$\mu = \mathbb{E}[f(x)|x \sim p] = p^{\mathrm{T}} f \tag{14}$$

such that $Pr(|\tilde{\mu} - \mu| < \epsilon) > 1 - \delta$ for any $\delta > 0$.

- We denote **QMEBO**$_\delta(p^{\mathrm{T}} f, B_p, B_f, \epsilon)$ as an estimation of $\mathbb{E}[f(x)|x \sim p]$, to error less than ϵ with probability at least $1 - \delta$, using **QMEBO**.
- **QMEBO**$_\delta(P^{\mathrm{T}}_{h|s,a}\hat{V}_{h+1}, O_{\mathcal{QM}}, B_{\hat{V}_{h+1}}, \epsilon)$ requires $O(\frac{\sqrt{S}}{\epsilon})$ queries to $O_{\mathcal{QM}}$.
  - Computing precise value $P^{\mathrm{T}}_{h|s,a}\hat{V}_{h+1}$ requires $O(S)$ queries to $O_{\mathcal{QM}}$.

Revisit the Quantum Value Iteration Algorithm QVI-1$(\mathcal{M}, \delta)$

Algorithm 4 Quantum Value Iteration Algorithm **QVI-1**$(\mathcal{M}, \delta)$

Require: MDP $\mathcal{M}$, quantum oracle $O_{\mathcal{QM}}$, maximum failure probability $\delta \in (0, 1)$.
Initialize: $\zeta \leftarrow \delta/(SH)$, $\hat{V}_H \leftarrow \mathbf{0}$.
for $h := H-1, \ldots, 0$ **do**
  create a quantum oracle $B_{\hat{V}_{h+1}}$ for vector $\hat{V}_{h+1} \in \mathbb{R}^{\mathcal{S}}$
  $\forall s \in \mathcal{S}$: create a quantum oracle $B_{\hat{Q}_{h,s}}$ encoding vector $\hat{Q}_{h,s} \in \mathbb{R}^{\mathcal{A}}$ with $O_{\mathcal{QM}}$ and $B_{\hat{V}_{h+1}}$ satisfying
    $\hat{Q}_{h,s}(a) \leftarrow r_h(s,a) + P_{h|s,a}^{\mathrm{T}} \hat{V}_{h+1}$ ▷ Can we incorporate **QMEBO** in this step?
  $\forall s \in \mathcal{S}$: $\hat{\pi}(s,h) \leftarrow \mathbf{QMS}_{\zeta}\{\hat{Q}_{h,s}(a) : a \in \mathcal{A}\}$
  $\forall s \in \mathcal{S}$: $\hat{V}_h(s) \leftarrow \hat{Q}_{h,s}\big(\hat{\pi}(s,h)\big)$
end for
Return: $\hat{\pi}$, $\hat{V}_0$

Algorithm 5 Quantum Value Iteration Algorithm **QVI-2**$(\mathcal{M}, \epsilon, \delta)$

Require: MDP $\mathcal{M}$, quantum oracle $O_{\mathcal{QM}}$, maximum error $\epsilon \in (0, H]$, failure probability $\delta \in (0, 1)$.
Initialize: $\zeta \leftarrow \delta/\big(4\tilde{c}SA^{1.5}H\log(1/\delta)\big)$, $\hat{V}_H \leftarrow \mathbf{0}$.
for $h := H-1, \ldots, 0$ **do**
  create a quantum oracle $B_{\tilde{V}_{h+1}}$ encoding $\tilde{V}_{h+1} \in [0,1]^{\mathcal{S}}$ defined by $\tilde{V}_{h+1} \leftarrow \hat{V}_{h+1}/H$
  $\forall s \in \mathcal{S}$: create a quantum oracle $B_{z_{h,s}}$ encoding $z_{h,s} \in \mathbb{R}^{\mathcal{A}}$ defined by
    $z_{h,s}(a) \leftarrow H \cdot \textbf{QMEBO}_{\zeta}(P^{\mathrm{T}}_{h|s,a}\tilde{V}_{h+1}, O_{\mathcal{QM}}, B_{\tilde{V}_{h+1}}, \frac{\epsilon}{2H^2}) - \frac{\epsilon}{2H}$
  $\forall s \in \mathcal{S}$: create quantum oracle $B_{\hat{Q}_{h,s}}$ encoding $\hat{Q}_{h,s} \in \mathbb{R}^{\mathcal{A}}$ with $O_{\mathcal{QM}}$ and $B_{z_{h,s}}$ satisfying
    $\hat{Q}_{h,s}(a) \leftarrow \max\{r_h(s,a) + z_{h,s}(a), 0\}$
  $\forall s \in \mathcal{S}$: $\hat{\pi}(s,h) \leftarrow \textbf{QMS}_{\delta}\{\hat{Q}_{h,s}(a) : a \in \mathcal{A}\}$
  $\forall s \in \mathcal{S}$: $\hat{V}_h(s) \leftarrow \hat{Q}_{h,s}\big(\hat{\pi}(s,h)\big)$
end for
Return: $\hat{\pi}$, $\{\hat{V}_h\}_{h=0}^{H-1}$

- $z_{h,s}(a)$ can be regarded as an $\frac{\epsilon}{H}$-approximation of $P^{\mathrm{T}}_{h|s,a}\hat{V}_{h+1}$.

High-level Idea of QVI-2$(\mathcal{M}, \epsilon, \delta)$

Note that the classical value iteration algorithm and **QVI-1** follows the same idea:

- Initialize $V_H = \mathbf{0}$.
- Repeatedly apply the Bellman recursion $V_h = \mathcal{T}^h(V_{h+1})$ for all $h \in [H]$, where

$$[\mathcal{T}^h(V_{h+1})]_s = \max_{a \in \mathcal{A}} \{r_h(s,a) + P_{h|s,a}^{\mathrm{T}} V_{h+1}\}, \forall s \in \mathcal{S}. \tag{15}$$

Idea of **QVI-2**:

- The Monotonicity Technique: Instead of computing the precise value of $P_{h|s,a}^{T} V_{h+1}$, **QMEBO** computes an estimate $z_{h,s}(a)$ with one-sided error satisfying

$$P_{h|s,a}^{\mathrm{T}} V_{h+1} - \frac{\epsilon}{H} \leq z_{h,s}(a) \leq P_{h|s,a}^{\mathrm{T}} V_{h+1}. \tag{16}$$

- Control the error in each step to be $\frac{\epsilon}{H}$ so that the total error after H steps remains ϵ.

The quantum speedup of **QVI-2**:

- **QMEBO**: $O(\sqrt{S})$ vs. precise value: $O(S)$.
- **QMS**: $O(\sqrt{A})$ vs. Classical: $O(A)$.

Theoretical Analysis on QVI-2

Theorem (Correctness of QVI-2$(\mathcal{M}, \epsilon, \delta)$)

The outputs $\hat{\pi}$ and $\{\hat{V}_h\}_{h=0}^{H-1}$ satisfy that

$$V_h^* - \epsilon \leq \hat{V}_h \leq V_h^{\hat{\pi}} \leq V_h^* \tag{17}$$

for all $h \in [H]$ with a success probability at least $1 - \delta$.

- The inequality $\hat{V}_h \leq V_h^{\hat{\pi}}$ comes from the one-sided error, i.e. the monotonicity technique.

Theorem (Complexity of QVI-2$(\mathcal{M}, \epsilon, \delta)$)

The quantum query complexity of ***QVI-2****$(\mathcal{M}, \epsilon, \delta)$ in terms of the quantum oracle of MDPs $O_{\mathcal{QM}}$ is*

$$O\Big(\frac{S^{1.5}\sqrt{A}H^3 \log\big(SA^{1.5}H/\delta\big)}{\epsilon}\Big). \tag{18}$$

- **QVI-2**$(\mathcal{M}, \epsilon, \delta)$ successfully achieves our optimization goal!
- **QVI-2** achieves significantly higher computational efficiency than the classical value iteration algorithm, particularly in problems characterized by a large state and action space but a short time horizon H.

Classical Lower Bound

Theorem (Classical Lower Bound in the Exact Dynamics Setting)

Let $\mathcal{S}$ and $\mathcal{A}$ be finite sets of states and actions. Let $H \geq 2$ be a positive integer and $\epsilon \in (0, \frac{H-1}{4})$ be an error parameter. We consider the following time-dependent and finite-horizon MDP $\mathcal{M} = (\mathcal{S}, \mathcal{A}, \{P_h\}_{h=0}^{H-1}, \{r_h\}_{h=0}^{H-1}, H)$, where $r_h \in [0,1]^{\mathcal{S}\times\mathcal{A}}$ for all $h \in [H]$.

- *Given access to a classical oracle $O_{\mathcal{M}}$, any algorithm $\mathcal{K}$, which takes $\mathcal{M}$ as an input and outputs ϵ-approximations of $\{V_h^*\}_{h=0}^{H-1}$ or π^* with probability at least 0.9, must call the classical oracle $O_{\mathcal{M}}$ at least*

$$\Omega(S^2A) \tag{19}$$

times on the worst case of input $\mathcal{M}$.

- Provided H and ϵ are constants, the quantum query complexities of **QVI-1** and **QVI-2** are $O(S^2\sqrt{A})$ and $O(S^{1.5}\sqrt{A})$, respectively.
- Quantum algorithms can solve finite-horizon MDPs with query complexity in terms of S and A that lies in a regime provably inaccessible to any classical algorithm!

Summary

3 Exact Dynamics Setting

Goal:	Query Complexity		
	Classical		Quantum Upper Bound
	Upper bound	Lower bound	
optimal π^*, V_0^*	S^2AH	S^2A	$S^2\sqrt{AH}$ **[QVI-1]**
ϵ-accurate estimate of π^* and $\{V_h^*\}_{h=0}^{H-1}$	S^2AH	S^2A	$\frac{S^{1.5}\sqrt{AH^3}}{\epsilon}$ **[QVI-2]**

Table: Classical and quantum query complexities for different algorithms solving time-dependent and finite-horizon MDPs in the exact dynamics setting. All quantum upper bounds are $\tilde{O}(\cdot)$ assuming a constant failure probability δ. The range of error term ϵ is $(0, H]$. The classical upper bounds are $O(\cdot)$, derived from the value iteration algorithm in Section 4.5 in [Bellman, 1957].

Table of Contents

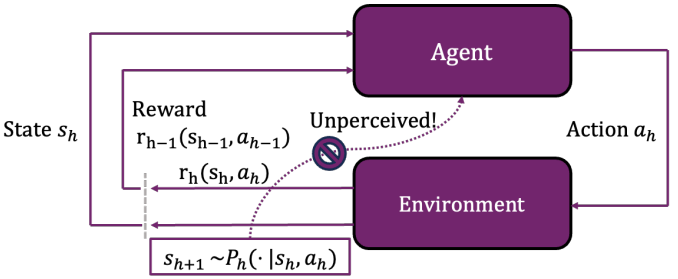
4 Generative Model Setting

- ▶ Introduction
- ▶ Preliminaries
- ▶ Exact Dynamics Setting
- ▶ Generative Model Setting
- ▶ Conclusion
- ▶ Reference

Background

4 Generative Model Setting

- The prior exact dynamics model is not always readily available in a complex environment.
- In this setting, it is assumed that the dynamics of the environment are unknown to the agent.



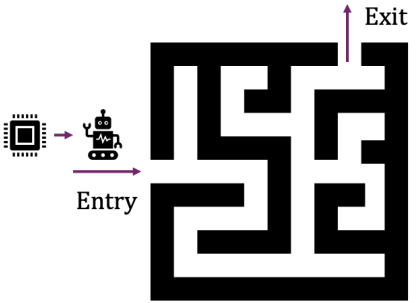


Figure: An illustration and an example of time-dependent and finite-horizon MDP dynamics in the generative model setting.

- The prior exact dynamics model is not always readily available in a complex environment.
- In this setting, it is assumed that the dynamics of the environment are unknown to the agent.

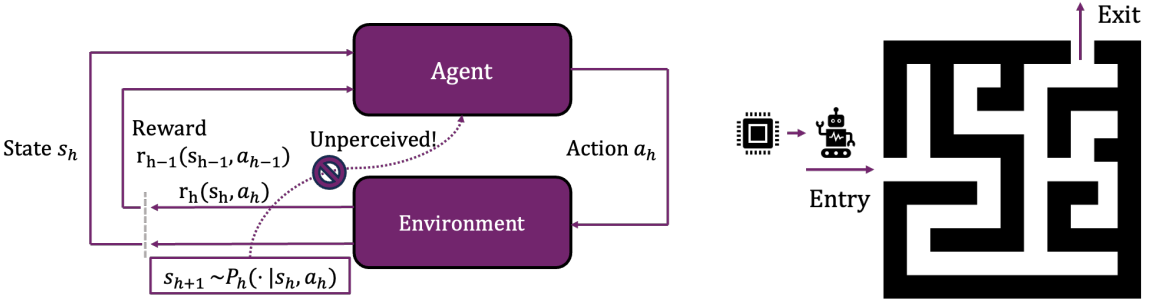


Figure: An illustration and an example of time-dependent and finite-horizon MDP dynamics in the generative model setting.

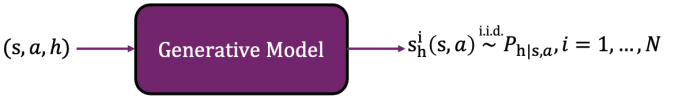


Figure: The agent can query a generative model to sample transitions for specific state-action pairs in each time horizon $h \in [H]$.

Classical and Quantum Generative Oracle

Generative Model Setting

- A classical generative oracle for the finite-horizon MDP is able to generate N independent samples for each triple $(s, a, h) \in \mathcal{S} \times \mathcal{A} \times [H]$ as follows

$$s_h^i(s,a) \overset{i.i.d.}{\sim} P_h(\cdot|s,a), \quad i = 1, \ldots, N. \tag{20}$$

Classical and Quantum Generative Oracle

Generative Model Setting

- A classical generative oracle for the finite-horizon MDP is able to generate N independent samples for each triple $(s,a,h) \in \mathcal{S} \times \mathcal{A} \times [H]$ as follows

$$s_h^i(s,a) \overset{i.i.d.}{\sim} P_h(\cdot|s,a), \quad i = 1,...,N. \tag{20}$$

- A quantum generative oracle for the finite-horizon MDP is defined as follows.

Definition (Quantum generative oracle of an MDP)

The quantum generative oracle of a time-dependent and finite-horizon MDP $\mathcal{M}$ is a unitary matrix $\mathcal{G} : \mathbb{C}^S \otimes \mathbb{C}^A \otimes \mathbb{C}^H \otimes \mathbb{C}^S \otimes \mathbb{C}^J \to \mathbb{C}^S \otimes \mathbb{C}^A \otimes \mathbb{C}^H \otimes \mathbb{C}^S \otimes \mathbb{C}^J$ satisfying

$$\mathcal{G} : |s\rangle \otimes |a\rangle \otimes |h\rangle \otimes |0\rangle \otimes |0\rangle \mapsto |s\rangle \otimes |a\rangle \otimes |h\rangle \left(\sum_{s'} \sqrt{P_{h|s,a}(s')}\, |s'\rangle \otimes |w_{s'}\rangle\right), \tag{21}$$

where $0 \leq J \in \mathbb{Z}$ is arbitrary and $|w_{s'}\rangle \in \mathbb{C}^J$ are arbitrary.

- Optimization goal: Given the generated data samples, we want to obtain ϵ-optimal policy $\hat{\pi}$, V-value functions $\{\hat{V}_h\}_{h=0}^{H-1}$ and Q-value functions $\{\hat{Q}_h\}_{h=0}^{H-1}$.

Quantum Mean Estimation

Generative Model Setting

Theorem (Quantum mean estimation [Montanaro, 2015])

There are two quantum algorithms, denoted as **QME1** *and* **QME2**, *with the following properties. Let* Ω *be a finite set,* $p = (p_x)_{x\in\Omega}$ *a discrete probability distribution over* Ω, *and* $f : \Omega \to \mathbb{R}$ *a function. Assume access to*

- *a probability oracle* U_p *for the probability distribution* p;
- *a binary oracle* B_f *for the function* f.

Then,

1. *For a function* f *satisfying* $0 \le f(x) \le u$ *for all* $x \in \Omega$, **QME1** *requires* $O\left(\frac{u}{\epsilon} + \sqrt{\frac{u}{\epsilon}}\right)$ *invocations of* U_p *and* B_f,
2. *For a function* f *satisfying* $Var[f(x) \mid x \sim p] \le \sigma^2$, **QME2** *needs* $O\left(\frac{\sigma}{\epsilon}\log^2\left(\frac{\sigma}{\epsilon}\right)\right)$ *invocations of* U_p *and* B_f,

to output an estimate $\tilde{\mu}$ *of* $\mu = \mathbb{E}[f(x) \mid x \sim p] = p^T f$ *satisfying* $\Pr(|\tilde{\mu} - \mu| > \epsilon) < 1/3$. *Furthermore, by repeating either* **QME1** *or* **QME2** *a total of* $O(\log(1/\delta))$ *times and taking the median of the outputs, one can obtain another estimate* $\hat{\mu}$ *of* μ *such that* $\Pr(|\hat{\mu} - \mu| < \epsilon) > 1 - \delta$.

We denote **QME**$\{i\}_\delta(p^T v, \epsilon)$ as an estimate of the mean $f(x)$, with x distributed as p, to error less than ϵ with probability at least $1 - \delta$, using **QME**$\{i\}$ for $i \in \{1, 2\}$.

Quantum Mean Estimation QME1

Generative Model Setting

For a random variable $X \in [0, u]$, one wants to obtain an ϵ-estimation of $\mathbb{E}[X]$, where $\epsilon \in (0, u]$.

- Hoeffding's inequality implies that $O(u^2/\epsilon^2)$ classical samples are required.
- **QME1** only requires $O(u/\epsilon)$ quantum samples.
- **QME1** is a quantum version of Hoeffding's inequality.

Lemma: Hoeffding's inequality

Let $X_1, X_2, \ldots, X_n$ be independent and identically distributed random variables such that $0 \leq X_i \leq u$ and true mean $\mathbb{E}[X_i] = \mu$ for all i. Let $\hat{X}_n = \frac{1}{n}(X_1 + X_2 + \cdots + X_n)$ be the sample mean. Then the Hoeffding's inequality states:

$$P(|\hat{X}_n - \mu| \geq \epsilon) \leq 2 \exp\left(-\frac{2n\epsilon^2}{u^2}\right). \tag{22}$$

Quantum Mean Estimation QME2

Generative Model Setting

For a random variable X with finite non-zero variance σ^2, one wants to obtain an ϵ-estimation of $\mathbb{E}[X]$, where $\epsilon \in (0, \sigma]$.

- Chebyshev's inequality implies that $O(\sigma^2/\epsilon^2)$ classical samples are required.
- **QME2** only requires $\tilde{O}(\sigma/\epsilon)$ quantum samples.
- **QME2** is a quantum version of Chebyshev's inequality.

Lemma: Chebyshev's inequality

Let $X_1, X_2, \ldots, X_n$ be independent and identically distributed random variables such that true mean $\mathbb{E}[X_i] = \mu$ and true variance $Var[X_i] = \sigma^2$ for all i. Let $\hat{X}_n = \frac{1}{n}(X_1 + X_2 + \cdots + X_n)$ be the sample mean. Then the Chebyshev's inequality states:

$$P(|\hat{X}_n - \mu| \geq \epsilon) \leq \frac{Var[\hat{X}_n]}{\epsilon^2} = \frac{\sigma^2}{n\epsilon^2}. \tag{23}$$

Quantum Value Iteration Algorithm QVI-3$(\mathcal{M}, \epsilon, \delta)$

4 Generative Model Setting

Algorithm 6 Quantum Value Iteration Algorithm **QVI-3**$(\mathcal{M}, \epsilon, \delta)$

Require: MDP $\mathcal{M}$, generative model $\mathcal{G}$, maximum error $\epsilon \in (0, H]$, maximum failure probability $\delta \in (0, 1)$.
Initialize: $\zeta \leftarrow \delta/\big(4\tilde{c}SA^{1.5}H\log(1/\delta)\big)$, $\hat{V}_H \leftarrow \mathbf{0}$.
for $h := H-1, \ldots, 0$ **do**
  create a quantum oracle $B_{\hat{V}_{h+1}}$ encoding $\hat{V}_{h+1} \in \mathbb{R}^{\mathcal{S}}$
  $\forall s \in \mathcal{S}$: create a quantum oracle $B_{z_{h,s}}$ encoding $z_{h,s} \in \mathbb{R}^{\mathcal{A}}$ with $\mathcal{G}$ and $B_{\hat{V}_{h+1}}$ satisfying
    $z_{h,s}(a) \leftarrow$ **QME1**$_\zeta\big((P^{\mathrm{T}}_{h|s,a}\hat{V}_{h+1}), \frac{\epsilon}{2H}\big) - \frac{\epsilon}{2H}$ ▷ We replace **QMEBO** with **QME1**.
  create a quantum oracle B_{r_h} encoding $r_h \in \mathbb{R}^{\mathcal{S}\times\mathcal{A}}$
  $\forall s \in \mathcal{S}$: create a quantum oracle $B_{\hat{Q}_{h,s}}$ encoding $\hat{Q}_{h,s} \in \mathbb{R}^{\mathcal{A}}$ with B_{r_h} and $B_{z_{h,s}}$ satisfying
    $\hat{Q}_{h,s}(a) \leftarrow \max\{r_h(s,a) + z_{h,s}(a), 0\}$
  $\forall s \in \mathcal{S} : \hat{\pi}(s,h) \leftarrow$ **QMS**$_\delta\{\hat{Q}_{h,s}(a) : a \in \mathcal{A}\}$
  $\forall s \in \mathcal{S} : \hat{V}_h(s) \leftarrow \hat{Q}_{h,s}\big(\hat{\pi}(s,h)\big)$
end for
Return: $\hat{\pi}$, $\{\hat{V}_h\}_{h=0}^{H-1}$

High-level Idea of QVI-3$(\mathcal{M}, \epsilon, \delta)$

QVI-3 shares a similar idea as **QVI-2**:

- Initialize $V_H = \mathbf{0}$.
- Repeatedly apply the Bellman recursion $V_h = \mathcal{T}^h(V_{h+1})$ for all $h \in [H]$, where

$$[\mathcal{T}^h(V_{h+1})]_s = \max_{a \in \mathcal{A}} \{r_h(s,a) + P_{h|s,a}^{\mathrm{T}} V_{h+1}\}, \forall s \in \mathcal{S}. \tag{24}$$

- The Monotonicity Technique: Instead of computing the precise value of $P_{h|s,a}^{T} V_{h+1}$, **QME1** computes an estimate $z_{h,s}(a)$ with one-sided eror satisfying

$$P_{h|s,a}^{\mathrm{T}} V_{h+1} - \frac{\epsilon}{H} \leq z_{h,s}(a) \leq P_{h|s,a}^{\mathrm{T}} V_{h+1}. \tag{25}$$

- Control the error in each step to be $\frac{\epsilon}{H}$ so that the total error after H steps remains ϵ.
- Apply **QMS** to find the action $\pi(s,h) = \mathrm{argmax}_{a \in \mathcal{A}} \{r_h(s,a) + P_{h|s,a}^{\mathrm{T}} V_{h+1}\}$.

The quantum speedup of **QVI-3**:

- **QME1**: $O(\sqrt{\frac{H^2}{\epsilon^2/H^2}}) = O(\frac{H^2}{\epsilon})$ vs. Hoeffding's inequality: $O(\frac{H^2}{\epsilon^2/H^2}) = O(\frac{H^4}{\epsilon^2})$.
- **QMS**: $O(\sqrt{A})$ vs. Classical: $O(A)$.

Theoretical Analysis on QVI-3$(\mathcal{M}, \epsilon, \delta)$

4 Generative Model Setting

Theorem (Correctness of QVI-3$(\mathcal{M}, \epsilon, \delta)$)

The outputs $\hat{\pi}$ and $\{\hat{V}_h\}_{h=0}^{H}$ satisfy that

$$V_h^* - \epsilon \leq \hat{V}_h \leq V_h^{\hat{\pi}} \leq V_h^* \tag{26}$$

for all $h \in [H]$ with a success probability at least $1 - \delta$.

- The inequality $\hat{V}_h \leq V_h^{\hat{\pi}}$ comes from the one-sided error technique, i.e. the monotonicity technique.

Theorem (Complexity of QVI-3$(\mathcal{M}, \epsilon, \delta)$)

*The quantum query complexity of **QVI-3**$(\mathcal{M}, \epsilon, \delta)$ in terms of the quantum generative oracle of MDPs $\mathcal{G}$ is*

$$O\Big(\frac{S\sqrt{A}H^3 \log\big(SA^{1.5}H/\delta\big)}{\epsilon}\Big). \tag{27}$$

- A classical algorithm [Sidford et al., 2023] requires $\tilde{O}(\frac{SAH^5}{\epsilon^2})$ queries to the classical generative model G.
- The state-of-the-art (SOTA) classical algorithm [Li et al., 2020] requires $\tilde{O}(\frac{SAH^4}{\epsilon^2})$ queries to the classical generative model G.

Improvement on QVI-3

Note that **QVI-3** only outputs ϵ-optimal policy and V-value functions.

- Can we obtain ϵ-optimal Q-value functions with **QVI-3**?
- Yes, but $\tilde{O}(\frac{S\sqrt{A}H^3}{\epsilon}) \to \tilde{O}(\frac{SAH^3}{\epsilon})$, because Q-value functions $Q_h \in \mathbb{R}^{\mathcal{S}\times\mathcal{A}}, h \in [H]$.
- Our quantum lower bounds also confirms that the $O(A)$ dependence of the quantum sample complexity is unavoidable.

QVI-4: (a) outputs the ϵ-optimal policy, V-value functions, and Q-value functions; (b) achieves a better dependence on H than **QVI-3** by adapting the following classical techniques [Sidford et al., 2018] in a quantum setting.

- The monotonicity technique
- The variance reduction technique
- The total-variance technique

Variance Reduction

Generative Model Setting

- Main Idea: Enhance efficiency over standard value iteration
- Goal: Achieve target error ϵ with $K = O(\log(H/\epsilon))$ epochs
- Strategy:
  - Decrease error: $\epsilon_k = \epsilon_{k-1}/2$, ending at $\epsilon_K = \epsilon$.
  - Outputs per epoch k: ϵ_k-optimal $V_{k,h}$, $Q_{k,h}$, and policy π_k.
  - Only increase a log term in query complexity.
- Rewrite the Bellman recursion:
  - Standard Bellman recursion: (1) Initialize $V_H = \mathbf{0}$; (2) Repeatedly apply the Bellman recursion $V_h = \mathcal{T}^h(V_{h+1})$, where $\mathcal{T}^h : \mathbb{R}^{\mathcal{S}} \to \mathbb{R}^{\mathcal{S}}$ is defined as

$$[\mathcal{T}^h(V_{h+1})]_s := \max_{a \in \mathcal{A}} \{r_h(s,a) + P_{h|s,a}^{\mathrm{T}} V_{h+1}\}, \tag{28}$$

    for all $s \in \mathcal{S}$.
  - Rewriting: (1) Repeat the standard Bellman recursion for K times: $V_h \to V_{k,h}$; (2) Rewrite the Bellman recursion:

$$P_{h|s,a}^{\mathrm{T}} V_{k,h+1} = P_{h|s,a}^{\mathrm{T}} (V_{k,h+1} - V_{k,h+1}^{(0)}) + P_{h|s,a}^{\mathrm{T}} V_{k,h+1}^{(0)}, \tag{29}$$

    where $V_{k,h+1}^{(0)}$ is the initial V-value from epoch $k-1$.

Variance Reduction

Generative Model Setting

- Estimation approach: Individually estimate the two terms of the RHS of Eq. (29) with an error $\epsilon_k/(2H)$.
- $P_{h|s,a}^{\mathrm{T}}(V_{k,h+1} - V_{k,h+1}^{(0)})$:
  - Condition: $\mathbf{0} \leq V_{k,h+1} - V_{k,h+1}^{(0)} \leq \tilde{c}\epsilon_k$
  - Classical: $O(H^2)$ samples — Quantum: $O(H)$ samples
- $P_{h|s,a}^{\mathrm{T}} V_{k,h+1}^{(0)}$:
  - Condition: $\mathbf{0} \leq V_{k,h+1}^{(0)} \leq H$
  - Classical: $O(H^4/\epsilon_k^2)$ — Quantum: $O(H^2/\epsilon_k)$
- Overall complexity:
  - Classical: $\tilde{O}(SAH^5/\epsilon_k^2)$
  - Quantum: $\tilde{O}(SAH^3/\epsilon_k)$

Variance Reduction

Generative Model Setting

- Estimation approach: Individually estimate the two terms of the RHS of Eq. (29) with an error $\epsilon_k/(2H)$.
- $P^{\mathrm{T}}_{h|s,a}(V_{k,h+1} - V^{(0)}_{k,h+1})$:
  - Condition: $\mathbf{0} \leq V_{k,h+1} - V^{(0)}_{k,h+1} \leq \tilde{c}\epsilon_k$
  - Classical: $O(H^2)$ samples — Quantum: $O(H)$ samples
- $P^{\mathrm{T}}_{h|s,a}V^{(0)}_{k,h+1}$:
  - Condition: $\mathbf{0} \leq V^{(0)}_{k,h+1} \leq H$
  - Classical: $O(H^4/\epsilon_k^2)$ — Quantum: $O(H^2/\epsilon_k)$
- Overall complexity:
  - Classical: $\tilde{O}(SAH^5/\epsilon_k^2)$
  - Quantum: $\tilde{O}(SAH^3/\epsilon_k)$

- Key advantage: Quantum subroutine **QME1** reduces complexity ($H^5 \to H^3$ and $1/\epsilon_k^2 \to 1/\epsilon_k$).
- Limitation: No A to $\sqrt{A}$ speedup (estimates all Q-values)
- Comparison: No additional H speedup vs. **QVI-3**
- Future benefit: Combines with total variance technique for greater gains

Total Variance Technique

Generative Model Setting

- Core insight: The propagation of errors across the H steps is smaller than assumed!
- Previous error: $\epsilon_k/(2H)$ per step for $\mu_{k,h}^{s,a} = P_{h|s,a}^{\mathrm{T}} V_{k,h+1}^{(0)} \to$ accumulated error over H steps is $\epsilon_k/2$.
- New error: Relax to $\epsilon_k \sigma_{k,h}^{s,a}/(2H^{1.5})$, where $\sigma_{k,h}^{s,a} = [\sigma_h(V_{k,h+1}^{(0)})](s,a)$
  - Max error: $\epsilon_k/(2\sqrt{H})$
  - Since $\epsilon_k \sigma_{k,h}^{s,a}/(2H^{1.5}) > \epsilon_k/(2H)$, the sample complexity can be reduced.
- Total error over H steps: Still bounded by $\epsilon_k/2$ (via Lemma on total variance upper bound: $\sum_{h=0}^{H-1} \sigma_{k,h}^{s,a} \leq H^{1.5}$)
- Classical sample complexity [Sidford et al., 2018]:
  - Chebyshev's inequality: $O(SA(\sigma_{k,h}^{s,a})^2(\epsilon\sigma_{k,h}^{s,a}/H^{1.5})^{-2}) = O(SAH^3/\epsilon^2)$ samples per time step and $\tilde{O}(SAH^4/\epsilon^2)$ overall.
  - Classical sample complexity without total variance technique: $\tilde{O}(SAH^5/\epsilon^2)$.
- Quantum sample complexity:
  - **QME2**: $\tilde{O}(SAH^{1.5}/\epsilon)$ samples per time step and $\tilde{O}(SAH^{2.5}/\epsilon)$ overall.

Quantum Value Iteration Algorithm QVI-4$(\mathcal{M}, \epsilon, \delta)$

Generative Model Setting

Algorithm 7 Quantum Value Iteration Algorithm **QVI-4**$(\mathcal{M}, \epsilon, \delta)$

Require: MDP $\mathcal{M}$, generative model $\mathcal{G}$, maximum error $\epsilon \in (0, \sqrt{H}]$, maximum failure probability $\delta \in (0, 1)$.
Initialize: $K \leftarrow \lceil \log_2(H/\epsilon) \rceil + 1$, $\zeta \leftarrow \delta/4KHSA$, $c = 0.001$, $b = 1$
Initialize: $\forall h \in [H] : V_{0,h}^{(0)} \leftarrow \mathbf{0}$; $\forall s \in \mathcal{S}, h \in [H] : \pi_0^{(0)}(s,h) \leftarrow$ arbitrary action $a \in \mathcal{A}$.
for $k = 0, \ldots, K-1$ **do**
  $\epsilon_k \leftarrow H/2^k$, $V_{k,H} \leftarrow \mathbf{0}$, $V_{k,H}^{(0)} \leftarrow \mathbf{0}$
  $\forall (s,a,h) \in \mathcal{S} \times \mathcal{A} \times [H] : y_{k,h}(s,a) \leftarrow \max\left\{\textbf{QME1}_\zeta\left(P_{h|s,a}^{\mathrm{T}}(V_{k,h+1}^{(0)})^2, b\right) - \left(\textbf{QME1}_\zeta(P_{h|s,a}^{\mathrm{T}} V_{k,h+1}^{(0)}, b/H)\right)^2, 0\right\}$
  $\forall (s,a,h) \in \mathcal{S} \times \mathcal{A} \times [H] : x_{k,h}(s,a) \leftarrow \textbf{QME2}_\zeta\left(P_{h|s,a}^{\mathrm{T}} V_{k,h+1}^{(0)}, cH^{-1.5}\epsilon\sqrt{y_{k,h}(s,a) + 4b}\right) - cH^{-1.5}\epsilon\sqrt{y_{k,h}(s,a) + 4b}$
  for $h := H-1, \ldots, 0$ **do**
    $\forall (s,a) \in \mathcal{S} \times \mathcal{A} : g_{k,h}(s,a) \leftarrow \textbf{QME1}_\zeta\left(P_{h|s,a}^{\mathrm{T}}(V_{k,h+1} - V_{k,h+1}^{(0)}), cH^{-1}\epsilon_k\right) - cH^{-1}\epsilon_k$
    $\forall (s,a) \in \mathcal{S} \times \mathcal{A} : Q_{k,h}(s,a) \leftarrow \max\{r_h(s,a) + x_{k,h}(s,a) + g_{k,h}(s,a), 0\}$
    $\forall s \in \mathcal{S} : \tilde{V}_{k,h}(s) \leftarrow V_{k,h}(s) \leftarrow [V(Q_{k,h})]_s, \tilde{\pi}_k(s,h) \leftarrow \pi_k(s,h) \leftarrow [\pi(Q_{k,h})]_s$
    $\forall s \in \mathcal{S}$: if $\tilde{V}_{k,h}(s) \leq V_{k,h}^{(0)}(s)$, then $V_{k,h}(s) \leftarrow V_{k,h}^{(0)}(s)$ and $\pi_k(s,h) \leftarrow \pi_k^{(0)}(s,h)$
  end for
  $\forall h \in [H] : V_{k+1,h}^{(0)} \leftarrow V_{k,h}$ and $\pi_{k+1}^{(0)}(\cdot,h) \leftarrow \pi_k(\cdot,h)$
end for
Return: $\hat{\pi} := \pi_{K-1}$, $\{\hat{V}_h\}_{h=0}^{H-1} := \{V_{K-1,h}\}_{h=0}^{H-1}$, $\{\hat{Q}_h\}_{h=0}^{H-1} := \{Q_{K-1,h}\}_{h=0}^{H-1}$

Analysis of QVI-4$(\mathcal{M}, \epsilon, \delta)$

Generative Model Setting

Theorem (Correctness of QVI-4$(\mathcal{M}, \epsilon, \delta)$)

The outputs $\hat{\pi}$, $\{\hat{V}_h\}_{h=0}^H$ and $\{\hat{Q}_h\}_{h=0}^H$ satisfy that

$$V_h^* - \epsilon \leq \hat{V}_h \leq V_h^{\hat{\pi}} \leq V_h^* \tag{30}$$

$$Q_h^* - \epsilon \leq \hat{Q}_h \leq Q_h^{\hat{\pi}} \leq Q_h^* \tag{31}$$

for all $h \in [H]$ with a success probability at least $1 - \delta$.

Theorem (Complexity of QVI-4$(\mathcal{M}, \epsilon, \delta)$)

*The quantum query complexity of **QVI-4**$(\mathcal{M}, \epsilon, \delta)$ in terms of the quantum generative oracle of MDPs $\mathcal{G}$ is*

$$O\left(SA\left(\frac{H^{2.5}}{\epsilon} + H^3\right)\log^2\left(\frac{H^{1.5}}{\epsilon}\right)\log\left(\log\left(\frac{H}{\epsilon}\right)HSA/\delta\right)\right). \tag{32}$$

- The best classical algorithm [Li et al., 2020] requires $\tilde{O}(\frac{SAH^4}{\epsilon^2})$ queries to a classical generative model G.

Lower Bounds for time-dependent and finite-horizon MDP

Generative Model Setting

Theorem (Classical lower bound for finite-horizon MDPs)

Let $\mathcal{S}$ and $\mathcal{A}$ be finite sets of states and actions. Let $H > 0$ be a positive integer and $\epsilon \in (0, 1/2)$ be an error parameter. We consider the following time-dependent and finite-horizon MDP $\mathcal{M} = (\mathcal{S}, \mathcal{A}, \{P_h\}_{h=0}^{H-1}, \{r_h\}_{h=0}^{H-1}, H)$, where $r_h \in [0,1]^{\mathcal{S}\times\mathcal{A}}$ for all $h \in [H]$.

- *Given access to a classical generative oracle G, any algorithm $\mathcal{K}$, which takes $\mathcal{M}$ as an input and outputs ϵ-approximations of $\{Q_h^*\}_{h=0}^{H-1}$ $\{V_h^*\}_{h=0}^{H-1}$ or π^* with probability at least 0.9, must call the classical generative oracle G at least*

$$\Omega\left(\frac{SAH^3}{\epsilon^2 \log^3(\epsilon^{-1})}\right) \tag{33}$$

times on the worst case of input $\mathcal{M}$.

Lower Bounds for time-dependent and finite-horizon MDP

Generative Model Setting

Theorem (Quantum lower bound for finite-horizon MDPs)

Let $\mathcal{S}$ and $\mathcal{A}$ be finite sets of states and actions. Let $H > 0$ be a positive integer and $\epsilon \in (0, 1/2)$ be an error parameter. We consider the following time-dependent and finite-horizon MDP $\mathcal{M} = (\mathcal{S}, \mathcal{A}, \{P_h\}_{h=0}^{H-1}, \{r_h\}_{h=0}^{H-1}, H)$, where $r_h \in [0,1]^{\mathcal{S}\times\mathcal{A}}$ for all $h \in [H]$.

- *Given access to a quantum generative oracle $\mathcal{G}$, any algorithm $\mathcal{K}$, which takes $\mathcal{M}$ as an input and outputs ϵ-approximations of $\{Q_h^*\}_{h=0}^{H-1}$ with probability at least 0.9, must call the quantum generative oracle at least*

$$\Omega\left(\frac{SAH^{1.5}}{\epsilon \log^{1.5}(\epsilon^{-1})}\right) \tag{34}$$

times on the worst case of input $\mathcal{M}$. Besides, any algorithm $\mathcal{K}$, which takes $\mathcal{M}$ as an input and outputs ϵ-approximations of $\{V_h^*\}_{h=0}^{H-1}$ or π^* with probability at least 0.9, must call the quantum generative oracle $\mathcal{G}$ at least

$$\Omega\left(\frac{S\sqrt{A}H^{1.5}}{\epsilon \log^{1.5}(\epsilon^{-1})}\right) \tag{35}$$

times on the worst case of input $\mathcal{M}$.

Summary

Generative Model Setting

Goal: obtain an ϵ-accurate estimate of	Classical sample complexity		Quantum sample complexity	
	Upper bound	Lower bound	Upper bound	Lower bound
$\{Q_h^*\}_{h=0}^{H-1}$	$\frac{SAH^4}{\epsilon^2}$ [Li et al., 2020]	$\frac{SAH^3}{\epsilon^2}$ [Theorem 21]	$\frac{SAH^{2.5}}{\epsilon}$ **[QVI-4]**	$\frac{SAH^{1.5}}{\epsilon}$ [Theorem 21]
$\pi^*, \{V_h^*\}_{h=0}^{H-1}$	$\frac{SAH^4}{\epsilon^2}$ [Li et al., 2020]	$\frac{SAH^3}{\epsilon^2}$ [Theorem 21]	$\frac{SAH^{2.5}}{\epsilon}$ **[QVI-4]** $\frac{S\sqrt{A}H^3}{\epsilon}$ **[QVI-3]**	$\frac{S\sqrt{A}H^{1.5}}{\epsilon}$ [Theorem 21]

Table: Classical and quantum sample complexities for solving time-dependent and finite-horizon MDPs in the generative model setting. The classical lower bound for π^* and $\{V_h^*\}_{h=0}^{H-1}$ was shown in [Sidford et al., 2018].

- **QVI-3** and **QVI-4** are nearly (asymptotically) optimal (up to log terms) in computing near-optimal V/Q value functions and policies, provided the time horizon H is a constant.
- Our quantum lower bounds rule out the possibility of exponential quantum speedups.

Table of Contents

5 Conclusion

- ▶ Introduction
- ▶ Preliminaries
- ▶ Exact Dynamics Setting
- ▶ Generative Model Setting
- ▶ Conclusion
- ▶ Reference

Conclusion and Future Work

Goal:	Query Complexity			
	Classical		Quantum	
	upper bound	lower bound	upper bound	lower bound
optimal π^*, V_0^*	S^2AH	S^2A	$S^2\sqrt{A}H$ **[QVI-1]**	?
ϵ-accurate estimate of π^* and $\{V_h^*\}_{h=0}^{H-1}$	S^2AH	S^2A	$\frac{S^{1.5}\sqrt{A}H^3}{\epsilon}$ **[QVI-2]**	?

Table: Classical and quantum query complexities for different algorithms solving time-dependent and finite-horizon MDPs in the exact dynamics setting. All quantum upper bounds are $\tilde{O}(\cdot)$ assuming a constant failure probability δ. The range of error term ϵ is $(0, H]$. The classical upper bounds are $O(\cdot)$, derived from the classical value iteration algorithm in [Bellman, 1957].

- What are the quantum lower bounds in the exact dynamics setting?
- What are the potential applications of the new quantum subroutines, **QMEBO**, and the quantum value iteration algorithms, **QVI-1** and **QVI-2**?

Conclusion and Future Work

Goal: obtain an ϵ-accurate estimate of	Classical sample complexity		Quantum sample complexity	
	Upper bound	Lower bound	Upper bound	Lower bound
$\{Q_h^*\}_{h=0}^{H-1}$	$\frac{SAH^4}{\epsilon^2}$ [Li et al., 2020]	$\frac{SAH^3}{\epsilon^2}$ [Theorem 21]	$\frac{SAH^{2.5}}{\epsilon}$ **[QVI-4]**	$\frac{SAH^{1.5}}{\epsilon}$ [Theorem 21]
$\pi^*, \{V_h^*\}_{h=0}^{H-1}$	$\frac{SAH^4}{\epsilon^2}$ [Li et al., 2020]	$\frac{SAH^3}{\epsilon^2}$ [Theorem 21]	$\frac{SAH^{2.5}}{\epsilon}$ **[QVI-4]** $\frac{S\sqrt{A}H^3}{\epsilon}$ **[QVI-3]**	$\frac{S\sqrt{A}H^{1.5}}{\epsilon}$ [Theorem 21]

Table: Classical and quantum sample complexities for solving time-dependent and finite-horizon MDPs in the generative model setting. The classical lower bound for π^* and $\{V_h^*\}_{h=0}^{H-1}$ was shown in [Sidford et al., 2018].

- Can we design optimal quantum algorithms whose quantum sample complexities are the same as the quantum lower bounds?
- What are the potential applications of **QVI-3** and **QVI-4**?

Table of Contents

6 Reference

- ▶ Introduction
- ▶ Preliminaries
- ▶ Exact Dynamics Setting
- ▶ Generative Model Setting
- ▶ Conclusion
- ▶ Reference

References

6 Reference

Beals, R., Buhrman, H., Cleve, R., Mosca, M., and de Wolf, R. (2001).
Quantum lower bounds by polynomials.
Journal of the ACM, 48(4):778—-797.

Bellman, R. (1957).
Dynamic programming.
science, 153(3731):34–37.

Cornelissen, A., Hamoudi, Y., and Jerbi, S. (2022).
Near-optimal quantum algorithms for multivariate mean estimation.
In *Proceedings of the 54th Annual ACM SIGACT Symposium on Theory of Computing*, STOC '22. ACM.

Durr, C. and Hoyer, P. (1999).
A quantum algorithm for finding the minimum.

Li, G., Wei, Y., Chi, Y., Gu, Y., and Chen, Y. (2020).
Breaking the sample size barrier in model-based reinforcement learning with a generative model.
Advances in neural information processing systems, 33:12861–12872.

Montanaro, A. (2015).
Quantum speedup of monte carlo methods.
Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences, 471(2181):20150301.

Nayak, A. and Wu, F. (1999).
The quantum query complexity of approximating the median and related statistics.
In *Proceedings of the Thirty-First Annual ACM Symposium on Theory of Computing (STOC 1999)*, page 384–393, Atlanta, GA, United States.

Sidford, A., Wang, M., Wu, X., Yang, L., and Ye, Y. (2018).
Near-optimal time and sample complexities for solving markov decision processes with a generative model.
Advances in Neural Information Processing Systems, 31.

References

6 Reference

Sidford, A., Wang, M., Wu, X., and Ye, Y. (2023).
Variance reduced value iteration and faster algorithms for solving markov decision processes.
Naval Research Logistics (NRL), 70(5):423–442.

Quantum Algorithms for Finite-horizon Markov Decision Processes

Thank you for listening!
Any questions?