A. Correctness and Complexity of SolveMdp1 (Algorithm 1)

A.1. Proof of Proposition 1

We first consider the failure probability. As all estimations are carried out with maximum failure probability $f := \delta/4KLSA$ and there are $3KSA + KLSA < 4KLSA$ estimations (Lines 7, 8 and 12), the probability that there exists an incorrect estimate (up to the specified error) is at most δ by the union bound.

We henceforth assume the qEst steps are all correct and proceed to prove Eq. (8) and Eq. (9), which we recall are:

$$v_{k,l} \leq v^{\pi_{k,l}} \leq v^*, \tag{8}$$
$$q_{k,l} \leq q^{\pi_{k,l}} \leq q^*. \tag{9}$$

The second inequalities in Eq. (8) and Eq. (9) are clear from the definitions of v^* and q^*. We therefore only show the first inequalities below and refer to them when referring to Eq. (8) and Eq. (9). The main idea is to use Lemma 1 together with the inequalities

$$x_k \leq P v_{k,0}, \tag{17}$$
$$\Delta_{k,l} \leq P v_{k,l} - P v_{k,0}, \tag{18}$$

that are immediate from the definitions of x_k and $\Delta_{k,l}$ on Lines 8 and 12 respectively because the subtracted terms equal the estimation errors.

To show Eq. (8), it suffices to show

$$v_{k,l} \leq \mathcal{T}^{\pi_{k,l}}(v_{k,l}). \tag{19}$$

Equation (8) then follows from repeatedly applying $\mathcal{T}^{\pi_{k,l}}$ on both sides of Eq. (19), and using the fact that $\mathcal{T}^{\pi_{k,l}}$ is monotone increasing and is a contraction with unique fixed point $v^{\pi_{k,l}}$.

We proceed to show Eq. (19) by induction on $n := (k-1)L + l$. The base case $n = 0$ is true because $v_{1,0} := \mathbf{0} \leq \mathcal{T}^{\pi_{1,0}}(v_{1,0}) = r$. The case $n = 1$ is also true because $v_{1,1} = v(q_{1,0}) = \mathbf{0} \leq \mathcal{T}^{\pi_{1,1}}(v_{1,1}) = r$, where we used $q_{1,0} := \mathbf{0}$. In addition, note that $v_{k,L} \leq \mathcal{T}^{\pi_{k,L}}(v_{k,L})$ is the same as $v_{k+1,0} \leq \mathcal{T}^{\pi_{k+1,0}}(v_{k+1,0})$ by definitions on Line 15. This means that once we have established the truth of Eq. (19) at $k = k', l = L$, we can assume its truth at $k = k' + 1, l = 0$.

Now consider $n > 1$. We prove Eq. (19) element-wise for each $s \in \mathcal{S}$ by considering the following two cases that could happen at the if-clause on Line 10.

1. Case $v(q_{k,l-1})[s] \geq v_{k,l-1}[s]$. Then

$$\begin{aligned} v_{k,l}[s] &:= v(q_{k,l-1})[s] \\ &= q_{k,l-1}[s, \pi_{k,l}[s]] \\ &= \max\{r[s, \pi_{k,l}(s)] + \gamma(x_k[s, \pi_{k,l}[s]] + \Delta_{k,l-1}[s, \pi_{k,l}[s]]), 0\} \\ &\leq r[s, \pi_{k,l}(s)] + \gamma(P v_{k,l-1})[s, \pi_{k,l}(s)] \\ &= \mathcal{T}^{\pi_{k,l}}(v_{k,l-1})[s] \\ &\leq \mathcal{T}^{\pi_{k,l}}(v_{k,l})[s], \end{aligned} \tag{20}$$

   where the second line uses $\pi_{k,l}[s] := \pi(q_{k,l-1})[s]$ in this case, the third line uses definition of $q_{k,l-1}$ (for $n > 1$), the fourth line uses Eq. (17) and Eq. (18) and $0 \leq v_{k,l-1}$ (Lemma 1) to remove the max, and the last line uses $v_{k,l-1} \leq v_{k,l}$ (Lemma 1).

2. Case $v(q_{k,l-1})[s] < v_{k,l-1}[s]$. Then

$$v_{k,l}[s] := v_{k,l-1}[s] \leq \mathcal{T}^{\pi_{k,l-1}}(v_{k,l-1})[s] \leq \mathcal{T}^{\pi_{k,l-1}}(v_{k,l})[s] = \mathcal{T}^{\pi_{k,l}}(v_{k,l})[s], \tag{21}$$

   where the first inequality is by the inductive hypothesis, the second inequality uses $v_{k,l-1} \leq v_{k,l}$ (Lemma 1), and the last equality uses $\pi_{k,l}[s] := \pi_{k,l-1}[s]$ in this case.

Therefore, we have established Eq. (19), and so Eq. (8).

Equation (9) then follows from

$$q_{k,l} \leq r + \gamma P v_{k,l} \leq r + \gamma P v^{\pi_{k,l}} = q^{\pi_{k,l}}, \tag{22}$$

where the first inequality again uses Eq. (17) and Eq. (18) and $\mathbf{0} \leq v_{k,l}$ (Lemma 1), and the second inequality uses Eq. (8) which we have just established. □

A.2. Proof of Proposition 2

By reusing the first paragraph in the proof of Proposition 1, we can readily set aside consideration of the failure probability.

We henceforth again assume the qEst steps are all correct and proceed to prove Eq. (10) and Eq. (11), which we recall are:

$$v^* - \epsilon_k \leq v_{k,L}, \tag{10}$$
$$q^* - \epsilon_k \leq q_{k,L}. \tag{11}$$

We proceed by induction on $k \geq 0$ with the inductive hypothesis comprising both inequalities above for all indices strictly less than k. The base case $k = 0$ can be established by defining $\epsilon_0 \coloneqq \Gamma$, $v_{0,L} \coloneqq \mathbf{0}$, and $q_{0,L} \coloneqq \mathbf{0}$. Note that these definitions will be consistent with the induction steps below.

Now consider $k > 0$. The main idea is to use Theorem 1 and the inequalities

$$x_k \geq P v_{k,0} - 2c(1-\gamma)^{1.5}\epsilon\sqrt{y_k + b}, \tag{23}$$
$$\Delta_{k,l} \geq P v_{k,l} - P v_{k,0} - 2c(1-\gamma)\epsilon_k, \tag{24}$$

that are immediate from the definitions of x_k and $\Delta_{k,l}$ on Lines 8 and 12 respectively.

We first show Eq. (11). Define vector $\xi_k \in \mathbb{R}^{SA}$ by

$$\xi_k \coloneqq 2c(1-\gamma)^{1.5}\epsilon\sqrt{y_k + b} + 2c(1-\gamma)\epsilon_k, \tag{25}$$

then we have

$$\begin{aligned}
q^* - q_{k,l} &= r + \gamma P^{\pi^*} q^* - \max\{r + \gamma(x_k + \Delta_{k,l}), \mathbf{0}\} \\
&\leq \gamma P^{\pi^*} q^* - \gamma(x_k + \Delta_{k,l}) \\
&\leq \gamma P^{\pi^*} q^* - \gamma(\cancel{P v_{k,0}} + P v_{k,l} - \cancel{P v_{k,0}} - 2c(1-\gamma)^{1.5}\epsilon\sqrt{y_k + b} - 2c(1-\gamma)\epsilon_k) \\
&\leq \gamma P^{\pi^*} q^* - \gamma P v_{k,l} + 2c(1-\gamma)^{1.5}\epsilon\sqrt{y_k + b} + 2c(1-\gamma)\epsilon_k \\
&= \gamma P^{\pi^*} q^* - \gamma P v_{k,l} + \xi_k \\
&\leq \gamma P^{\pi^*} q^* - \gamma P v(q_{k,l-1}) + \xi_k \\
&\leq \gamma P^{\pi^*}(q^* - q_{k,l-1}) + \xi_k,
\end{aligned} \tag{26}$$

where the fourth line uses $\gamma \leq 1$, the sixth line uses $v(q_{k,l-1}) \leq v_{k,l}$ (Lemma 1), and the last line uses $P^{\pi^*} q_{k,l-1} \leq P v(q_{k,l-1})$ which follows from definitions.

Recursing Eq. (26) with respect to $l \geq 1$ gives

$$\begin{aligned}
q^* - q_{k,l} &\leq \gamma^l (P^{\pi^*})^l (q^* - q_{k,0}) + \sum_{i=0}^{l-1} \gamma^i (P^{\pi^*})^i \xi_k \\
&\leq \gamma^l \Gamma + (I - \gamma P^{\pi^*})^{-1} \xi_k,
\end{aligned} \tag{27}$$

where the last line uses $q^* - q_{k,0} \leq q^* \leq \Gamma$ as $q_{k,0} \geq \mathbf{0}$ by definitions on Line 4 and Line 13. The first term, $\gamma^l \Gamma$, can be bounded when $l = L - 1, L$:

$$\gamma^L \Gamma \leq \gamma^{L-1} \Gamma \leq \exp(-(L-1)(1-\gamma))\Gamma \leq \epsilon/4 \leq \epsilon_k/2, \tag{28}$$

where the second inequality uses $x \leq \exp(-(1-x))$ for all $x \in \mathbb{R}$, the third inequality uses the definition $L \coloneqq \Gamma\lceil\log(4\Gamma/\epsilon)\rceil + 1$, and the last inequality uses $\epsilon \leq 2\epsilon_K \leq 2\epsilon_k$ for all $k \in [K]$ which follows from $K \leq \log_2(\Gamma/\epsilon) + 1$.

We now bound the second term, $(I - \gamma P^{\pi^*})^{-1}\xi_k$. To this end, we first bound the term $\sqrt{y_k + b}$ appearing in ξ_k. From the definition of y_k, there exists a b' with $|b'| \leq b$ such that

$$\begin{aligned}
\sqrt{y_k + b} &\leq \max\{(Pv_{k,0}^2 + b - (Pv_{k,0} + (1-\gamma)b')^2)^{1/2}, \sqrt{b}\} \\
&\leq (\sigma^2(v_{k,0}) + b + 2(1-\gamma)|b'|Pv_{k,0})^{1/2} \\
&\leq \sqrt{\sigma^2(v_{k,0}) + 3b} \\
&\leq \sigma(v_{k,0}) + \sqrt{3b} \\
&\leq \sigma(v^*) + \sigma(v^* - v_{k,0}) + \sqrt{3b},
\end{aligned} \tag{29}$$

where the second line uses $\mathbf{0} \leq v_{k,0}$ (Lemma 1) to remove the max, the third line uses $v_{k,0} \leq \Gamma$ (Proposition 1), and the last line uses the fact that, for any random variables X and Y, we have $\mathrm{Var}[X+Y] = \mathrm{Var}[X] + \mathrm{Var}[Y] + 2\,\mathrm{Cov}[X,Y] \leq (\sqrt{\mathrm{Var}[X]} + \sqrt{\mathrm{Var}[Y]})^2$.

But we have $v_{k,0} - v^* \leq 0$ from Eq. (8) of Proposition 1 and $v^* - v_{k,0} = v^* - v_{k-1,L} \leq \epsilon_{k-1}$ by the inductive hypothesis. Therefore, $\sigma(v_{k,0} - v^*) \leq \|v_{k,0} - v^*\| \leq \epsilon_{k-1} = 2\epsilon_k$, and therefore

$$\sqrt{y_k + b} \leq \sigma(v^*) + 2\epsilon_k + \sqrt{3b}. \tag{30}$$

Therefore, recalling $\xi_k \coloneqq 2c(1-\gamma)^{1.5}\epsilon\sqrt{y_k + b} + 2c(1-\gamma)\epsilon_k$ from Eq. (25), we have

$$\begin{aligned}
(I - \gamma P^{\pi^*})^{-1}\xi_k &= 2c(1-\gamma)^{1.5}\epsilon(I - \gamma P^{\pi^*})^{-1}\sqrt{y_k + b} + 2c(1-\gamma)\epsilon_k(I - \gamma P^{\pi^*})^{-1}\,\mathbf{1} \\
&\leq 2c(1-\gamma)^{1.5}\epsilon(I - \gamma P^{\pi^*})^{-1}(\sigma(v^*) + 2\epsilon_k + \sqrt{3b}) + 2c(1-\gamma)\epsilon_k(I - \gamma P^{\pi^*})^{-1}\,\mathbf{1} \\
&\leq 2c(1-\gamma)^{1.5}\epsilon(I - \gamma P^{\pi^*})^{-1}\sigma(v^*) + 2c\sqrt{1-\gamma}\,\epsilon\,2\epsilon_k + 2c\sqrt{1-\gamma}\,\epsilon\,\sqrt{3b} + 2c\,\epsilon_k \\
&\leq 2c\,\sqrt{2}\,\epsilon + 2c\,\sqrt{1-\gamma}\,\epsilon\,2\epsilon_k + 2c\,\epsilon\sqrt{3b} + 2c\,\epsilon_k \\
&\leq 2c\,(2\sqrt{2} + 2 + 2\sqrt{3b} + 1)\epsilon_k \\
&< \epsilon_k/2,
\end{aligned} \tag{31}$$

where the third line uses $(I - \gamma P^{\pi^*})^{-1}\,\mathbf{1} \leq (1-\gamma)^{-1}$, the fourth line *crucially* uses Theorem 1 with π set to π^*, the fifth line uses $\epsilon \leq 2\epsilon_k$ for all $k \in [K]$ and the input assumption $\sqrt{1-\gamma}\,\epsilon \leq 1$, i.e., $\epsilon \leq \sqrt{1-\gamma}$, and the last line uses definitions $b \coloneqq 1$ and $c \coloneqq 0.01$.

Using Eq. (28) and Eq. (31) to bound the first and second terms in Eq. (27) respectively, we find

$$q^* - q_{k,L} \leq \epsilon_k, \tag{32}$$

$$q^* - q_{k,L-1} \leq \epsilon_k. \tag{33}$$

The top equation is one inequality we wish to show in our induction. The bottom equation can be used to establish the other inequality as follows. For all $s \in \mathcal{S}$, we have

$$v_{k,L}[s] \geq v(q_{k,L-1})[s] = \max_a\{q_{k,L-1}[s,a]\} \geq \max_a\{q^*[s,a] - \epsilon_k\} = v^*[s] - \epsilon_k, \tag{34}$$

where the first inequality is by Lemma 1. Hence $v_{k,L} \geq v^* - \epsilon_k$, as desired. □

A.3. Proof of Theorem 5 (Complexity of SolveMdp1)

As in the correctness analysis, we assume that all estimations are correct, up to the specified error, because the probability that this does not hold is at most δ. This means we can assume all results obtained during the correctness analysis. In the following, we will use $K = O(\log(\Gamma/\epsilon))$ and $L = O(\Gamma\log(\Gamma/\epsilon))$ without further remarks.

Let C be the complexity of SolveMdp1 as if all estimations were carried out with maximum failure probabilities set to constant. Then, since the actual maximum failure probabilities are set to $f \coloneqq \delta/4KLSA$, the actual complexity of SolveMdp1 is

$$O(C\log(KLSA/\delta)) = O(C\log(SA\Gamma\log(\Gamma/\epsilon)/\delta)). \tag{35}$$

Now we bound C by examining each line involving qEst in turn and using Theorem 2.

On Line 7, we can bound $\mathbf{0} \leq v_{k,0} \leq v^* \leq \Gamma$. Therefore, we can use quantum mean estimation algorithm qEst1 in Theorem 2, which results in an overall query cost of order

$$SAK(\Gamma^2 b^{-1} + \sqrt{\Gamma^2 b^{-1}} + \Gamma(1-\gamma)^{-1}b^{-1} + \sqrt{\Gamma(1-\gamma)^{-1}b^{-1}}) = O(SA\Gamma^2 \log(\Gamma/\epsilon)). \tag{36}$$

On Line 8, we see that $\sigma^2(v_{k,0})[s,a] \leq \sqrt{y_k[s,a]+b}$. We also note that $0 < (1-\gamma)^{1.5}\epsilon\sqrt{y_k[s,a]+b} < 4\sqrt{y_k[s,a]+b}$. Therefore, we can use quantum mean estimation algorithm qEst2 in Theorem 2, with error set to $(1-\gamma)^{1.5}\epsilon\sqrt{y_k[s,a]+b}$ and variance upper bound set to $y_k[s,a]+b$, which results in an overall query cost of order

$$K \sum_{(s,a)\in\mathcal{S}\times\mathcal{A}} w[s,a]\log^2(w[s,a]) = O(SA\Gamma^{1.5}\epsilon^{-1}\log^3(\Gamma/\epsilon)), \tag{37}$$

where, importantly, $w[s,a] \coloneqq \cancel{\left(\sqrt{y_k[s,a]+b}\right)}\left((1-\gamma)^{1.5}\epsilon\cancel{\sqrt{y_k[s,a]+b}}\right)^{-1} = \Gamma^{1.5}/\epsilon$.

On Line 12, we can bound $\mathbf{0} \leq v_{k,l} - v_{k,0} \leq v^* - v_{k,0} \leq \epsilon_{k-1} = 2\epsilon_k$. Therefore, we can use quantum mean estimation algorithm qEst1 in Theorem 2, which results in an overall cost of order

$$LSA\left(\frac{\cancel{2\epsilon_k}}{c(1-\gamma)\cancel{\epsilon_k}} + \sqrt{\frac{\cancel{2\epsilon_k}}{c(1-\gamma)\cancel{\epsilon_k}}}\right) = O(SA\Gamma^2\log(\Gamma/\epsilon)). \tag{38}$$

Adding together Eq. (36), Eq. (37), and Eq. (38), and noting that all logarithmic terms are at most $\log^3(\Gamma/\epsilon)$, shows that

$$C = O(SA(\Gamma^{1.5}\epsilon^{-1} + \Gamma^2)\log^3(\Gamma/\epsilon)). \tag{39}$$

Combining the above equation with Eq. (35) shows that the overall quantum query complexity of SolveMdp1 is

$$O(SA(\Gamma^{1.5}\epsilon^{-1} + \Gamma^2)\log^3(\Gamma/\epsilon)\log(SA\Gamma\log(\Gamma/\epsilon)/\delta)) = O(SA(\Gamma^{1.5}\epsilon^{-1} + \Gamma^2)\log^4(\Gamma/\epsilon)\log(SA\Gamma/\delta)), \tag{40}$$

as desired. □

B. Correctness and Complexity of SolveMdp2 (Algorithm 2)

Failure probability aside, our strategy for proving the correctness of SolveMdp2 (Theorem 6) is to observe the similarity between SolveMdp2 and SolveMdp1 and then reuse the arguments used to prove the correctness of SolveMdp1.

As mentioned in the main text, SolveMdp2 is similar to SolveMdp1 with k set to 1. In particular, the vectors $z_l,\ q_l \in \mathbb{R}^{SA}$, defined entry-wise by

$$z_l[s,a] \coloneqq z_{l,s}[a], \tag{41}$$
$$q_l[s,a] \coloneqq q_{l,s}[a], \tag{42}$$

are analogous to the vectors $x_1 + \Delta_{1,l}$ and $q_{1,l}$ appearing in SolveMdp1 respectively. Moreover, the $\tilde{v}_l[s] \coloneqq \max_a\{q_{l-1,s}[a]\} = \max_a\{q_{l-1}[s,a]\}$ appearing in SolveMdp2 corresponds exactly to the $v(q_{1,l-1})[s] \coloneqq \max_a\{q_{1,l-1}[s,a]\}$ appearing in SolveMdp1.

Having observed the similarity between SolveMdp2 and SolveMdp1, the following analogue of Lemma 1 due to the if-then-else statement is clear.

Lemma 2. *For all $l \in [L]$, the v_ls are monotone increasing, that is $v_{l-1} \leq v_l$, and moreover we have $v_l \geq v(q_{l-1})$.*

B.1. Proof of Theorem 6 (Correctness of SolveMdp2)

We first consider the failure probability. The analysis is similar to the previous one except that we now need to analyze quantum oracles that may fail. To do this, we appeal to basic facts about unitary matrices, in particular, a quantum version of the union bound stating that the failure probabilities of quantum operators, i.e., unitary matrices, add linearly.

On Line 10, because $U_{z_{l,s}}$ is created using qEst with failure probability f, it is $2Af$-close to its "ideal version". More precisely, we mean that there exists a quantum oracle $U^{\text{ideal}}_{z_{l,s}}$ encoding $\widehat{(Pv_l)}[s,a] - (1-\gamma)\epsilon/4$, where $\widehat{(Pv_l)}[s,a]$ satisfies $|\widehat{(Pv_l)}[s,a] - (Pv_l)[s,a]| \le (1-\gamma)\epsilon/4$, such that $\|U^{\text{ideal}}_{z_{l,s}} - U_{z_{l,s}}\|_{\text{op}} \le 2Af$. Since $U_{q_{l,s}}$ can be created using one call to $U_{z_{l,s}}$ and one call to $U^{-1}_{z_{l,s}}$, it is $4Af$-close to its ideal version (defined similarly). Then, on Line 6, qArgmax uses the oracle $U_{q_{l,s}}$ at most $c_{\max}\sqrt{A}\log(1/\delta)$ times. By the quantum union bound and substituting in the definition of f, this means the quantum operation implemented by qArgmax is $(c_{\max}\sqrt{A}\log(1/\delta)\cdot 4Af = \delta/LS)$-close to its ideal version. This means that the output of qArgmax is incorrect with probability at most δ/LS. Since qArgmax is invoked a total of LS times, we see that the overall probability of failure is at most δ by the (usual) union bound.

We henceforth assume the qEst and qArgmax steps are all correct and proceed to prove Eq. (15), which we recall is:

$$v^* - \epsilon \le \hat{v} \le v^{\hat{\pi}} \le v^*. \tag{15}$$

The last inequality, $v^{\hat{\pi}} \le v^*$, is clear.

To prove the middle inequality, $\hat{v} \le v^{\hat{\pi}}$, we can directly reuse the proof of Proposition 1 provided we have $z_l \le Pv_l$. But this is clear because x_l is equal to an estimate of Pv_l with the estimation error subtracted off.

To prove the first inequality, $v^* - \epsilon \le \hat{v}$, we can reuse the proof of Proposition 2, provided we have $z_l \ge Pv_l - (1-\gamma)\epsilon/2$, which is true. Defining $\xi = (1-\gamma)\epsilon/2 \cdot \mathbf{1} \in \mathbb{R}^{SA}$, we see from the proof of Proposition 2 that

$$q^* - q_{L-1} \le \gamma^{L-1}\Gamma + (1 - \gamma P^{\pi^*})^{-1}\xi \le \epsilon, \tag{43}$$

since $L \coloneqq \Gamma\lceil \log(4\Gamma/\epsilon)\rceil + 1$. Therefore, for all $s \in \mathcal{S}$, we have

$$v_L[s] \ge v(q_{L-1})[s] = \max_a\{q_{L-1}[s,a] \ge \max_a\{q^*[s,a] - \epsilon\} = v^*[s] - \epsilon. \tag{44}$$

□

B.2. Proof of Theorem 7 (Complexity of SolveMdp2)

Like the proof of Theorem 5, we can assume all results obtained from the correctness analysis.

We again let C be the complexity of SolveMdp2 as if all estimations and maximum finding were carried out with maximum failure probabilities set to constant. Then the actual complexity of our algorithm is

$$O(C\log(LSA/\delta)) = O(C\log(SA\Gamma\log(\Gamma/\epsilon)/\delta)), \tag{45}$$

since the actual maximum failure probabilities are set to $f \coloneqq \delta/4c_{\max}LSA^{1.5}\log(1/\delta)$ and $L = O(\Gamma\log(\Gamma/\epsilon))$.

Now we bound C. Note that, for all $l \in [L]$, we have

$$\mathbf{0} \le v_l \le v^* \le \Gamma. \tag{46}$$

By using qEst1 of Theorem 2 to do the qEst on Line 10, the query complexity of $U_{z_{l,s}}$ is

$$\frac{\Gamma}{(1-\gamma)\epsilon/4} + \sqrt{\frac{\Gamma}{(1-\gamma)\epsilon/4}} = O(\Gamma^2/\epsilon), \tag{47}$$

provided $\epsilon = O(\Gamma^2)$. But we have (trivially) assumed $\epsilon \le \Gamma$ on the input ϵ, so this holds.

As $U_{q_{l,s}}$ uses one call to $U_{z_{l,s}}$ and one call to its inverse $U^{-1}_{z_{l,s}}$, the query complexity of $U_{q_{l,s}}$ is twice that of $U_{z_{l,s}}$.

By means of the quantum maximum finding algorithm (Theorem 3) we only incur a multiplicative factor of $O(\sqrt{A})$ when we invoke qArgmax over an action space of size A. That is, for each $l \in [L]$ and $s \in \mathcal{S}$, qArgmax makes $O(\sqrt{A})$ queries to $U_{q_{l,s}}$ to find $\text{argmax}_a\{q_{l-1,s}[a]\}$. There are also L iterations, so

$$C = O(L\,S\sqrt{A}\,\Gamma^2\epsilon^{-1}) = O(S\sqrt{A}\,\Gamma^3\epsilon^{-1}\,\log(\Gamma/\epsilon)), \tag{48}$$

because $L = O(\Gamma \log(\Gamma/\epsilon))$. Combining the above equation with Eq. (45) shows that the overall quantum query complexity of SolveMdp2 is

$$O(S\sqrt{A}\,\Gamma^3\epsilon^{-1}\,\log(\Gamma/\epsilon)\log(SA\Gamma\log(\Gamma/\epsilon)/\delta)) = O(S\sqrt{A}\,\Gamma^3\epsilon^{-1}\,\log^2(\Gamma/\epsilon)\log(SA\Gamma/\delta)), \tag{49}$$

as desired. □

C. Lower Bounds

We first establish the lower bound for an MDP with $S = 2$ and $A = 1$. Note that when $A = 1$, there is only one action per state, so it is trivial to compute the optimal policy. So we can only show hardness for computing q^* or v^*, which will be the same because there is only one action.

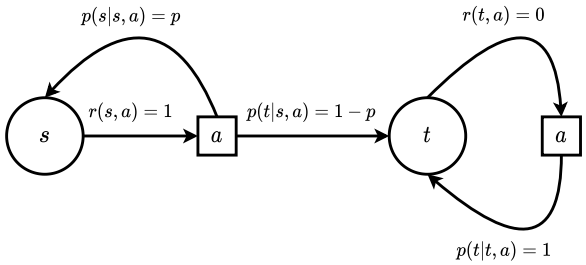

Figure 1. The MDP we use for the lower bound with $S = 2$ and $A = 1$. Distinguishing between $p \le p_0$ and $p \ge p_0 + \alpha$ is hard.

Lemma 3. *Fix any $\gamma \in [0.9, 1)$. Let $\Gamma := (1-\gamma)^{-1} \ge 10$ and fix any $\epsilon \in (0, \Gamma/4)$. There exists an MDP shown in Figure 1 with 2 states and 1 action, for which computing v^* (or equivalently, q^*) to error ϵ requires $\Omega(\Gamma^3/\epsilon^2)$ queries to a classical generative oracle or $\Omega(\Gamma^{1.5}/\epsilon)$ queries to a quantum generative oracle.*

Proof. The MDP shown in Figure 1 has two states we call s and t. State t is a sink and the only transition from there is back to t with no reward. Hence $v^*(t) = 0$. State s is a source, and on taking action a, there is a reward $r(s,a) = 1$. The transition is probabilistic and controlled by an unknown probability $p \in (0,1)$. With probability p we come back to s, and with probability $1-p$ we move to t. We can compute $v^*(s)$ using the equation $v^*(s) = 1 + \gamma(pv^*(s) + (1-p)v^*(t))$, which yields

$$v^*(s) = \frac{1}{1-\gamma p}. \tag{50}$$

Now further assume that we are promised that $p \le p_0$ or $p \ge p_0 + \alpha$, where

$$p_0 = 1 - \frac{1}{\Gamma} \quad \text{and} \quad \alpha = \frac{3\epsilon}{\Gamma^2}. \tag{51}$$

Note that $p_0 + \alpha < 1$ because of the way we have chosen the range of ϵ.

We claim that computing $v^*(s)$ to additive error ϵ will allow us to distinguish these two cases. To see this, note that the difference between the two values of $v^*(s)$ is at least

$$\begin{aligned} &\frac{1}{1-\gamma(p_0+\alpha)} - \frac{1}{1-\gamma p_0} \\ =&\frac{\gamma\alpha}{(1-\gamma(p_0+\alpha))(1-\gamma p_0)} \\ >&\frac{\gamma\alpha}{(1-\gamma p_0)^2} \ge \frac{0.9\alpha}{(1.1/\Gamma - 1/10\Gamma^2)^2} \\ \ge&0.9\alpha\Gamma^2/1.21 \ge \alpha\Gamma^2/1.35 \ge 2\epsilon. \end{aligned} \tag{52}$$

Thus computing v^* to additive error ϵ will allow us to distinguish these two possibilities.

Now we just have to show that distinguishing a coin with probability of heads at most p_0 or at least $p_0 + \alpha$ given samples from this coin is as hard as claimed in the lower bound. We prove this via query complexity.

Suppose that instead of having sample access to a coin, we have query access to an n-bit string x with the promise that either at most p_0 fraction of its bits is equal to 1 or at least $p_0 + \alpha$ fraction of its bits is equal to 1. Both quantumly and classically, we can query any bit x_i of x using 1 query. It is easy to see that we can generate a sample from our coin with probability of heads equal to $|x|/n$ (the fraction of 1s in x) with only 1 query to x. This works both classically and quantumly.

So we have shown a reduction from the problem of computing v^* to error ϵ to the problem of deciding whether $|x|/n \leq p_0$ or $|x|/n \geq p_0 + \alpha$ given query access to an n-bit string x. This is the approximate counting problem. If we count the number of 0s, we want to distinguish $1/\Gamma$ 0s from $(1/\Gamma - 3\epsilon/\Gamma^2)$ 0s. We need to approximate the count to multiplicative precision $O(\epsilon/\Gamma)$. Finally, we can invoke the known lower bounds for approximate counting summarized in Lemma 4. These give a classical lower bound of $\Omega(\Gamma^3/\epsilon^2)$ and a quantum lower bound of $\Omega(\Gamma^{1.5}/\epsilon)$ as claimed. □

We formally state the approximate counting lemma used in the previous proof. The quantum bounds are due to (Nayak & Wu, 1999) and (Brassard et al., 2000).

Lemma 4 (Approximate counting)**.** *Let $x \in \{0,1\}^n$ be a string to which we have standard classical or quantum query access (i.e., we can query the ith bit and receive x_i). Then deciding whether $|x| \leq k$ or $|x| \geq k(1+\epsilon)$ for $k < n/2$, requires $\Theta(\frac{n}{\epsilon^2 k})$ classical queries or $\Theta(\frac{1}{\epsilon}\sqrt{\frac{n}{k}})$ quantum queries.*

We can now extend the lower bound to larger S and A. Before doing so, we will need some structural theorems about quantum query complexity and randomized query complexity. For a function f, let $R(f)$ and $Q(f)$ denote their randomized and quantum query complexities. The first result shows that computing the logical OR of k copies of a problem scales with k. The classical result is due to (Göös et al., 2017) and the quantum result follows from a general composition theorem for quantum query complexity in (Reichardt, 2011). The second result, known as a direct sum result, can also be found in (Reichardt, 2011).

Lemma 5. *Let OR_k be the logical OR function on k bits and f be an arbitrary Boolean function. Then the complexity of the composed function $\mathsf{OR}_k \circ f$, which is defined as the logical OR of the k outputs of k independent instances of f is related to the complexity of f as follows: $Q(\mathsf{OR}_k \circ f) = \Omega(\sqrt{k}\,Q(f))$ and $R(\mathsf{OR}_k \circ f) = \Omega(kR(f))$. In addition, computing all k outputs of k independent instances of f requires $\Omega(k\,R(f))$ queries classically and $\Omega(k\,Q(f))$ queries quantumly.*

Note that the "in addition" result can be viewed as a result about the query complexity of f composed with the function $\mathsf{Identity}_k : \{0,1\}^k \to \{0,1\}^k;\ x \mapsto x$.

We are now ready to prove the main lower bound theorem.

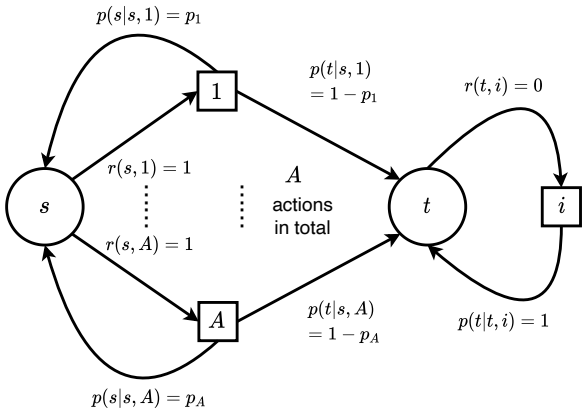

Figure 2. The MDP we use for the lower bound with $S = 2$ and arbitrary A. For each i, p_i is promised to be either $\leq p_0$ or $\geq p_0 + \alpha$. Any action $i \in \mathcal{A}$ taken from state t always returns to t with zero reward.

C.1. Proof of Theorem 8

We start by keeping $S = 2$ and allowing arbitrarily large $A \geq 2$. For notational convenience, we identify $\mathcal{A}$ with $\{1, \ldots, A\}$.

We will use essentially the same instance as in Fig. 1 but now with A outgoing actions from state s, each with transition probability p_a for $a \in \mathcal{A}$. The modified instance is illustrated in Fig. 2. We again consider the case where all the p_a satisfy the promise that they are either small ($\leq p_0$) or large ($\geq p_0 + \alpha$). As argued in the previous proof, deciding if a given p_a is small or large has a classical lower bound of $\Omega(\Gamma^3/\epsilon^2)$ and a quantum lower bound of $\Omega(\Gamma^{1.5}/\epsilon)$.

Now consider the problem of deciding whether any of the p_a is small or large. This is the logical OR of A independent problems, each of which we have already shown a lower bound for. If we could compute v^* to error ϵ, then we would be able to solve this problem. Hence using Lemma 5, we get a classical lower bound of $\Omega(A\Gamma^3/\epsilon^2)$ and a quantum lower bound of $\Omega(\sqrt{A}\Gamma^{1.5}/\epsilon)$ for the problem of computing v^*.

Similarly, consider the problem of deciding which of the p_a is large, promised that exactly one of them is large and the rest are small. This is similar to logical OR, except the goal is to identify the location of a 1 promised that it exists. This problem is as hard as logical OR, and we get the same lower bounds. For such an instance, computing π^* to error ϵ will allow us to distinguish the two cases, since $\pi^*(s)$ should equal the unique action for which p_a is large. This gives us the claimed lower bounds for π^*.

Similarly, consider the problem of learning which p_as are large and which are small for all a (without any promise on the number of each type). This is the problem of solving A independent instances of a problem for which we have already proved a lower bound. For quantum and classical algorithms, this increases the complexity by a factor of A as stated in the second part of Lemma 5. Thus we get a classical lower bound of $\Omega(A\Gamma^3/\epsilon^2)$ and a quantum lower bound of $\Omega(A\Gamma^{1.5}/\epsilon)$ for this problem. But if we could compute q^* to error ϵ, then we would be able to solve this problem since such an estimate encodes whether each p_a is large or small. This gives us the claimed lower bounds for q^*.

Thus we have established all the lower bounds for $S = 2$ and arbitrary A. Finally, to extend the lower bounds to arbitrarily large S, we can just use $S/2$ copies of the MDP in Fig. 2. Computing any one of the quantities q^*, v^*, or π^* on this MDP instance means solving $S/2$ independent copies of the problems discussed above. As stated in the second part of Lemma 5, for both classical and quantum algorithms, this increases the complexity by a factor of $\Omega(S)$. This yields the claimed lower bounds for general S and A. □