Quantum Acceleration for the American Option Problem

Csatári Jakab

Supervisor: Gilyén András

2023

Abstract

In the first half of the semester, I was studying a financial problem, the American Option. I analyzed the known LSM method for this problem and also a quantum algorithimc approach provided by J. F. Doriguello, A. Luongo, J. Bao, P. Rebentrost and M. Santha. In essence, quantum acceleration depends on the fact that we can estimate expected values quadratically faster with a quantum algorithm. In the rest of the semester, I worked on the first known optimal algorithm for this quadratic acceleration, from last year's article by R. Kothari and R. O'Donnell.

1. American Option Problem

In the financial sector, the American option is one of the most tangible examples of an optimal stopping problem. An option gives its buyer the right to buy or sell a product at a predetermined price, but at some future date. In the case of an American option, the time within it can be exercised is arbitrary within a specified term. In the case of a call right (option to buy), the investor benefits if the market value of the product is higher than it was when the option was purchased.

Let us say that we have a 1-year American-style call option on a portfolio that contains 100 stocks. This means that the option can be exercised for one year any day after purchase, but the value of all shares changes slightly every day. The profit (or as we will refer to it later, *payoff*) will then be the change in the value of the portfolio between the purchase and exercise of the option.

Suppose we model the possible market changes with probability distributions. The problem can then be formulated as an optimal stopping problem:

Problem: Given a stochastic process, with different profits at each point in time, when is it worth stopping so that the expected value of this profit is as high as possible?

Modeling of the problem:

- Let $(X_t)_{t=0}^T$ be a Markov-chain with Ω sample space and $E \subseteq \mathbb{R}^d$ state space. In the example $t = 0, ..., T$ discrete times correspond to days and X_t describes how our shares performed on the t-th day.

- There is also a stochastic process $(Z_t)_{t=0}^T$ which we call *payoff* from now on. The value of the payoff for given t only depends on X_t: $Z_t := z_t(X_t)$, where $z_t \in L_2(E)$.
  $\big(L_2(E) :=$ set of squared integrable functions.$\big)$

- Let U_t be the following:

$$U_t := \begin{cases} Z_T & \text{if } t = T \\ \max\{Z_t, \mathbb{E}(U_{t+1}|X_t)\} & \text{else} \end{cases}$$

  Then U_t describes the expected payoff if we get to time t (we have not yet exercised the option). Given the value this way, it can be seen that it is worth selling in t if the expected payoff is not greater later. If we have a sample $X'_0, ..., X'_T$, then the corresponding $U'_T, U'_{T-1}, ..., U'_0$ values can be defined in this order.

- As $\mathbb{E}(U_{t+1}|X_t)$ is X_t measureable, there is such an $f_t(x)$ that is Borel measureable and satisfies $f_t(X_t) = \mathbb{E}(U_{t+1}|X_t)$, now let u_t be:

$$u_t := \begin{cases} z_T & \text{if } t = T \\ \max\{z_t, f_t\} & \text{else} \end{cases}$$

  Therefore $U_t = u_t(X_t)$.

- Let $\tau_t := \min\{k \geq t \mid U_k = Z_k\}$.

  The values τ_t $(t = 0, ..., T)$ are stopping times. From U_t one can see that for all t there is a k, for which the equality stands - this is precisely the time, when the payoff is expected to be at most the same as later. So if we get to time t, then it is worth to stop at τ_t.

  τ_t can be formalized like this:

$$\tau_t := \begin{cases} T & \text{if } t = T \\ t \cdot \mathbb{1}\{Z_t \geq \mathbb{E}(Z_{\tau_{t+1}}|X_t)\} + \tau_{t+1} \cdot \mathbb{1}\{Z_t < \mathbb{E}(Z_{\tau_{t+1}}|X_t)\} & \text{else} \end{cases}$$

- The task is not only to estimate the stopping time, but also the available profit. So the goal of the task is to calculate the pair (U_0, τ_0), which can be done by estimating u_t functions.

2. Least Squares Monte Carlo algorithm (LSM)

The idea of the algorithm is on one hand to calculate U_t, τ_t values based on their previous definitions as the solution of a dynamic programming. We can do this by taking N independent sample simulations from (X_t):

$$(X_t^{(1)})_{t=0}^T, ..., (X_t^{(N)})_{t=0}^T$$

The values $Z_t^{(i)} = z_t(X_t^{(i)})$ are the corresponding payoffs, and based on the sample, we will calculate in order $t = T, ..., 0$.

Let $\{e_{t,k}\}_{k=1}^m$ be a set of linearly independent "basis functions" from $L_2(E)$ for all t. For short e_t will denote the m dimensional vector: $e_t(\cdot) := \big(e_{t,1}(\cdot), ..., e_{t,m}(\cdot)\big)$. These e_t values $(t = 0, ..., T)$ will be called the approximation scheme.

Notation: if $a \in \mathbb{R}^m$, then $a \cdot e_t := a_1 \cdot e_{t,1} + ... + a_m \cdot e_{t,m}$

So now we want to estimate $\mathbb{E}(U_{t+1}|X_t)$ in this $\{e_{t,k}\}_{k=1}^m$ basis, with $\alpha_t \cdot e_t(X_t)$. Let us take the closest by L_2 norm:

$$\alpha_t = \arg\min_a \mathbb{E}\big((U_{t+1} - a \cdot e_t(X_t))^2\big)$$

Theorem 2.1.: Given $\{e_{t,k}\}_{k=1}^m$ $(t = 0, ..., T)$ approximation scheme, let A $m \times m$ be the covariance matrix where $\big(A_t\big)_{i,j} = \mathbb{E}\big(e_{t,i}(X_t)e_{t,j}(X_t)\big)$ and $b_t = \mathbb{E}\big(U_{t+1}e_t(X_t)\big)$. Then $\alpha_t \approx A_t^{-1} \cdot b_t$.

Proof:

$$\mathbb{E}\big(U_{t+1}\big) \approx \mathbb{E}\big(\alpha_t \cdot e_t(X_t)\big) = \mathbb{E}\big(\alpha_{t,1} \cdot e_{t,1}(X_t) + ... + \alpha_{t,m} \cdot e_{t,m}(X_t)\big)$$

$$\mathbb{E}\big(U_{t+1}\big) \approx \mathbb{E}\big(\alpha_{t,1} \cdot e_{t,1}(X_t)\big) + ... + \mathbb{E}\big(\alpha_{t,m} \cdot e_{t,m}(X_t)\big)$$

Therefore

$$b_t = \mathbb{E}\big(U_{t+1}\big)\mathbb{E}\big(e_t(X_t)\big) \approx \mathbb{E}\big(\alpha_{t,1} \cdot e_{t,1}(X_t)\big)\mathbb{E}\big(e_t(X_t)\big) + ... + \mathbb{E}\big(\alpha_{t,m} \cdot e_{t,m}(X_t)\big)\mathbb{E}\big(e_t(X_t)\big) =$$

$$\mathbb{E}\big(\alpha_{t,1} \cdot e_{t,1}(X_t)\big)\mathbb{E}\big(e_{t,1}(X_t) + ... + e_{t,m}(X_t)\big) + ... + \mathbb{E}\big(\alpha_{t,m} \cdot e_{t,m}(X_t)\big)\mathbb{E}\big(e_{t,1}(X_t) + ... + e_{t,m}(X_t)\big) =$$

$$\alpha_{t_1}\mathbb{E}\big(e_{t,1}(X_t)\sum_{k=1}^m e_{t,1}(X_t)\big) + ... + \alpha_{t_m}\mathbb{E}\big(e_{t,m}(X_t)\sum_{k=1}^m e_{t,1}(X_t)\big) = \alpha_t \cdot A_t$$

During the algorithm we will estimate certain values based on our sample, so for example when we calculate A_t then instead of $\mathbb{E}\big(e_{t,j}(X_t)e_{t,k}(X_t)\big)$ we calculate the mean given by the samples: $\frac{1}{N}\sum_{i=1}^N \big(e_{t,j}(X_t^{(i)})e_{t,k}(X_t^{(i)})\big)$. In such case instead of A_t we use the notation $\widetilde{A}_t$ (and also we will denote similarly the other values when we estimate by the sample).

Classical algoritm:

1. Take sample simulations: $(X_t^{(i)})_{t=0}^T \quad i = 1, ..., N$
2. Calculate values $Z_t^{(i)}$ and $e_{t,k}(X_t^{(i)})$ for all $t = 0, ..., T \ ; \ i = 1, ..., N \ ; \ k = 1, ..., m$
3. Specify $\widetilde{A}_t$ for each $t = 0, ..., T$ and calculate their inverses as well.
4. $\tilde{u}_T = z_T \qquad \forall \ i = 1, ..., N$

   FOR $t = (T-1), ..., 0$:

   $$\tilde{\alpha}_t = \widetilde{A}_t^{-1} \tfrac{1}{N} \sum_{i=1}^{N} \tilde{u}_{t+1}(X_{t+1}^{(i)}) \cdot e_t(X_t^{(i)})$$
   $$\tilde{u}_t := \max\{z_t, \tilde{\alpha}_t \cdot e_t\}$$

5. **RETURN**

$$\widetilde{U}_0 = \frac{1}{N} \sum_{i=1}^{N} \tilde{u}_0(X_0^{(i)})$$

3. Expected value estimation

Let y be a discrete real random variable, we want to estimate the expected value $\mu = \mathbb{E}(y)$ by $\hat{\mu}$, that satisfies for given $0 < \epsilon, \delta < 1$:

$$\mathbb{P}\big(|\hat{\mu} - \mu| > \epsilon\big) \leq \delta$$

Now we can take N samples according to the distribution of X and let $\hat{\mu}$ be their mean (as we did in the LSM too). Then from Chebyshev's inequality we get:

$$\mathbb{P}\big(|\hat{\mu} - \mu| > \epsilon\big) \leq \frac{\sigma^2}{N\epsilon^2}$$

So for constant δ one need to take $N = O(\sigma^2/\epsilon^2)$ samples (and in the classical setup there is no algorithm using less samples [4]).

In contrast in the quantum setup quadratic acceleration can be achieved, meaning there is a quantum algorithm using $O(\sigma/\epsilon)$ samples that satisfies the same requirement.

When article [1] was written only $\widetilde{O}(\sigma/\epsilon)$ was known, the result of A. Montanaro ([3]) is that there is a quantum algorithm which uses $O(\sigma/\epsilon \cdot log(1/\delta) \cdot log^{3/2}(\sigma/\epsilon) \cdot loglog(\sigma/\epsilon))$ samples. However in their last year's article, R. Kothari and R. O'Donnell ([2]) got rid of the logarithmic factors and have shown, that $O(\sigma/\epsilon)$ sample is enough. About this I am writing in section 6., before that let us take a look at the quantum LSM.

4. Quantum LSM

In the quantum LSM we use the expected value estimation as a subrutin to estimate certain values, depending on (X_t). Denote with $\hat{N}$ the required number of samples in the quantum case, so $\hat{N} = O(\sigma/\epsilon)$.

The quantum LSM starts by creating a sample generator oracle $\mathbf{P}$, for which

$$\mathbf{P}|0\rangle = \sum \sqrt{p(x)}|x\rangle \text{ , where } p(x) = \mathbb{P}(X_1 = x_1)\mathbb{P}(X_2 = x_2|X_1 = x_1)...\mathbb{P}(X_T = x_T|X_{T-1} = x_{T-1})$$

So $\mathbf{P}$ gives a superposition of the possible realizations of the Markov chain, which if measured, results in sampling according to the distribution.

The algorithm:

1. Estimate $e_{t,k}(X_t)e_{t,l}(X_t)$ values by $\hat{N}$ queries (i.e. with $\hat{N}$ number of $\mathbf{P}$ calls)
2. By this specify all $\widetilde{A}_t$, and their inverses too classically
3. $\tilde{u}_T := z_T$
4. **FOR** $t = (T-1), ..., 0$:

   Estimate $\tilde{u}_{t+1} \cdot e_{t,k}(X_t)$ values and by them specify $\tilde{b}_t$ (here also with $\hat{N}$ calls)

   $\tilde{\alpha}_t = \widetilde{A}_t^{-1} \cdot \tilde{b}_t$

   $\tilde{u}_t := \max\{z_t, \tilde{\alpha}_t \cdot e_t\}$
5. **RETURN** $\widetilde{U}_0 = \tilde{u}_0(X_0)$

5. Comparing runtimes

Now in classic case we had N samples, let us denote the time it requires by $N \cdot \mathbf{T}_{samp}$, then calculating $Z_t^{(i)}$ and $e_{t,k}(X_t^{(i)})$ takes $O(N \cdot T \cdot m^2)$ time. The inverse of $\widetilde{A}_t$ can be calculated in $O(T \cdot m^\omega)$, finally the calculation of $\tilde{\alpha}_t$ values in $O(T \cdot m^2)$. Therefore the runtime is:

$$O(N \cdot \mathbf{T}_{samp} + N \cdot T \cdot m^2 + T \cdot m^\omega)$$

While in quantum case if $\mathbf{T}_{qsamp}$ is the required time for sampling, then all $e_{t,k}(X_t)e_{t,l}(X_t)$ (hence calculating all $\widetilde{A}_t$) takes $O(\hat{N} \cdot \mathbf{T}_{qsamp} \cdot T \cdot m^2)$ time. The inverse of $\widetilde{A}_t$ can be calculated again in $O(T \cdot m^\omega)$ the same way. Estimating $\tilde{b}_t$ is $O(\hat{N} \cdot \mathbf{T}_{qsamp} \cdot T \cdot m)$. Therefore the runtime is:

$$O(\hat{N} \cdot \mathbf{T}_{qsamp} \cdot T \cdot m^2 + T \cdot m^\omega)$$

Remark: The more detailed runtime analysis that can be found in [1] uses Montanaro's result, so there are additional polylogarithmic factors in their theorems. Also now I examined the runtime under slightly milder conditions as the authors, for example I did not take into account how long it would take to create/invoke the necessary quantum gates.

In any case, it can be seen from this that the quantum LSM achieves an almost quadratic acceleration compared to the classical one. We can assume that typically N, and respectively $\hat{N}$ are the determining terms (because we want ϵ to be small). But it can also be seen that if T is big, we might not accelerate that much.

6. Quantum acceleration for the expected value estimation

Now I would like to summarize the result of R. Kothari and R. O'Donnell. They have shown that the estimation defined in section 3. can be achieved with a quantum algorithm that uses $O(\sigma/\epsilon)$ samples. The algorithm they gave for it is basically a quantum phase estimation, which uses a specific controlled gate $\mathbf{U}$ (which is similar to the one used in Grover's search).

So let y be a real discrete random variable, we want to estimate $\mu = \mathbb{E}(y)$.

Theorem 6.1.: There exists such a quantum algorithm that uses $O(n)$ samples and outputs $\hat{\mu}$ for which:

$$\mathbb{P}\Big(|\hat{\mu} - \mu| \geq \frac{\sigma}{n}\Big) \leq \frac{1}{3}$$

Remark: Then indeed if $n = O(\sigma/\epsilon)$, we get

$$\mathbb{P}\big(|\hat{\mu} - \mu| \geq \epsilon\big) \leq \frac{1}{3}$$

and by $O(log(1/\delta))$ repetitions (then taking the median) any $0 < \delta \leq 1/3$ can be achieved on the right side.

Theorem 6.2.: Given $\epsilon > 0$ suppose that $\mathbb{E}(y^2) \leq 1$, then there exists a quantum algorithm, that uses $O(1/\epsilon)$ samples and with probability $2/3$ distinguishes $a)$ $|\mu| \leq \epsilon/2$, and $b)$ $\epsilon \leq |\mu| \leq 2\epsilon$.

There is a reduction between the two theorems, if **6.2.** stands, then **6.1** is also true. (For this a binary search an the loglog trick can be applied: [2] 4.1. section.)

If y is a discrete random variable on $[D]$ state space, then let $\mathbf{P}$ be a gate, where

$$\mathbf{P}|0\rangle = \sum_{\ell=1}^{D} \sqrt{p(\ell)} \cdot |\ell\rangle |garbage_\ell\rangle$$

Let $\alpha_\ell := -2\ arctan(y_\ell)$ and let $\mathbf{U} := REFL_p \cdot ROT_y$, where

$$REFL_p := \mathbf{P}(2|0\rangle\langle 0| - I)\mathbf{P}^\dagger \qquad \text{and} \qquad ROT_y \text{ such as } \quad ROT_y|\ell\rangle|garbage_\ell\rangle = e^{i\alpha_\ell}|\ell\rangle|garbage_\ell\rangle$$

Notation 6.3.: If $\sum_{j=1}^{D} e^{i\theta_j}|u_j\rangle\langle u_j|$ is an eigendecomposition of $\mathbf{U}$ and $|\sigma\rangle$ is a unitvector, then in this basis $|\sigma\rangle = \sum_{j=1}^{D} \hat{\sigma}_j|u_j\rangle$, where $\hat{\sigma}_j = \langle u_j|\sigma\rangle$. As $|\sigma\rangle$ is a unitvector, $|\hat{\sigma}_1|^2, |\hat{\sigma}_2|^2, ..., |\hat{\sigma}_D|^2$ determines a probability distribution.

In this case $\theta \sim \Theta_{\mathbf{U}}(|\sigma\rangle)$ will denote that $j \in [D]$ index is chosen according to the distribution induced by $|\sigma\rangle$ and we choose θ_j from the eigendecomposition of $\mathbf{U}$ (so $\theta := \theta_j$).

Notation 6.4.: If $z_1, ..., z_D$ are complex numbers, then $([D], p)$ determines a complex $\underline{\mathbf{z}}$ random variable, because at first we can choose an ℓ index according to p, then the appropriate z_ℓ will be the value of the variable. Then denote $|\underline{\mathbf{z}}\rangle := \sum_{\ell=1}^{D} z_\ell\sqrt{p(\ell)}|\ell\rangle$.

If $\underline{\mathbf{1}}$ is the all 1 ($\forall \ell \ : \ z_\ell = 1$) random variable, then $|\underline{\mathbf{1}}\rangle = \mathbf{P}|0\rangle$.

Theorem 6.5.: Denote with $s^2 := \mathbb{E}(y^2)$ and suppose that $s \leq 1/16$. If $\theta \sim \Theta_{\mathbf{U}}(|\underline{\mathbf{1}}\rangle)$, then:

$$\mathbb{P}(4/5 \cdot 2|\mu| \leq |\theta| \leq 5/4 \cdot 2|\mu|) \geq 1 - 2/9$$

Lemma 6.6.: If $\theta \sim \Theta_{\mathbf{U}}(|\underline{\mathbf{1}}\rangle)$, then $\mathbb{E}\big(sin(\theta/2)^{-2}\big) = (1 + s^2)/\mu^2$.

Lemma 6.7.: Let $|\underline{\mathbf{1}+\mathbf{i}\mathbf{y}}\rangle := \sum_{\ell=1}^{D}(1 + iy_\ell)\sqrt{p(\ell)}|\ell\rangle$, the normalized vector is $|\underline{\mathbf{1}+\mathbf{i}\mathbf{y}}\rangle/\sqrt{1+s^2}$. If $\tilde{\theta} \sim \Theta_{\mathbf{U}}(|\underline{\mathbf{1}+\mathbf{i}\mathbf{y}}\rangle/\sqrt{1+s^2})$, then $\mathbb{E}\big(sin(\theta/2)^2\big) = \mu^2/(1 + s^2)$.

Lemma 6.8.: $\big|\langle\underline{\mathbf{1}}|\underline{\mathbf{1}+\mathbf{i}\mathbf{y}}\rangle\big|^2 \geq (1 + \mu^2)/(1 + s^2) \geq 1/(1 + s^2)$

Proof (Theorem 6.5): The proof is provided with worse constants, but these can be refined to those that are in the theorem statement ([2] 3.5. section). Suppose that $s \leq 1/1000$

$$\mathbb{P}(5/100 \cdot 2|\mu| \leq |\theta| \leq 50 \cdot 2|\mu|) \geq 1 - 5/1000$$

Applying Markov inequality for lemma **6.6.**, we get that

$$\mathbb{P}\big(sin(\theta/2)^{-2} \geq 350 \cdot \frac{1 + s^2}{\mu^2}\big) \leq 1/350 < 3/1000$$

Meaning with greater probability than 3/1000:

$$(sin(\theta/2)^{-2} \leq 350 \cdot \frac{1+s^2}{\mu^2} < \frac{400}{\mu^2}$$

As

$$\Big(\frac{|\mu|}{20}\Big)^2 \leq (sin(\theta/2)^2 \leq (\theta/2)^2 \quad \Rightarrow \quad 2/20 \cdot |\mu| \leq |\theta|$$

it means

$$\mathbb{P}(|\theta| < 5/100 \cdot 2|\mu|) < 3/1000 \tag{1}$$

On the other hand applying Markov for lemma **6.7** and $\tilde{\theta} \sim \Theta_{\mathbf{U}}(|\underline{\mathbf{1}+\mathbf{iy}}\rangle/\sqrt{1+s^2})$:

$$\mathbb{P}\big(sin(\tilde{\theta}/2)^2 \geq 1000 \cdot \frac{\mu^2}{1+s^2}\big) \leq 1/1000$$

Meaning with greater probability than 1/1000:

$$(sin(\tilde{\theta}/2)^2 \leq 1000 \cdot \frac{\mu^2}{1+s^2} < 1000 \cdot \mu^2$$

As

$$|sin(\tilde{\theta}/2)| \leq \sqrt{1000}|\mu| \quad \Rightarrow \quad |\tilde{\theta}/2| \leq \pi/2\sqrt{1000}|\mu| < 50 \cdot 2|\mu|$$

it means

$$\mathbb{P}(|\tilde{\theta}| > 50 \cdot 2|\mu|) \leq 1/1000$$

And by lemma **6.8.**:

$$\mathbb{P}(|\theta| > 50 \cdot 2|\mu|) \leq (1+s^2)/1000 < 2/1000 \tag{2}$$

So from inequalities (1) and (2):

$$\mathbb{P}(5/100 \cdot 2|\mu| > |\theta|) + \mathbb{P}(|\theta| > 50 \cdot 2|\mu|) \leq 3/1000 + 2/1000 = 5/1000$$

Quantum phase estimation: Given a gate $\mathbf{U}$ and a $|\psi\rangle$ eigenstate, meaning $\mathbf{U}|\psi\rangle = \lambda|\psi\rangle$, and $\lambda = e^{2\pi i \cdot 0,\varphi_1 \ldots \varphi_n \cdots}$ (φ_i is the ith decimal point). Now if we want to estimate λ up to nth decimal, then the following quantum algorithm is called quantum phase estimation:

1. Starting from $|0\rangle^n|\psi\rangle$ qubits, we apply Hadamard gates on the first n register, so

$$|0\rangle^n|\psi\rangle \to \frac{1}{\sqrt{2^n}} \sum |j\rangle|\psi\rangle$$

2. Then we call n controlled-$\mathbf{U}$ gate, where the control qubits are the $n, n-1, ..., 1$ register in order and the target is always the last register. Now:

$$\frac{1}{\sqrt{2^n}} \sum |j\rangle|\psi\rangle \to \frac{1}{\sqrt{2^n}} \sum |j\rangle \mathbf{U}^j|\psi\rangle = \frac{1}{\sqrt{2^n}} \sum e^{2\pi i \frac{\varphi_1 ... \varphi_n}{2^n} j}|j\rangle|\psi\rangle$$

3. Finally we apply the $\mathbf{QFT}_N^{-1}$ gate, where $N = 2^n$ and $\mathbf{QFT}_N$ is the quantum Fourier transformation, from which we get:

$$\frac{1}{\sqrt{2^n}} \sum e^{2\pi i \frac{\varphi_1 ... \varphi_n}{2^n} j}|j\rangle|\psi\rangle \to |\varphi_1 ... \varphi_n\rangle|\psi\rangle$$

4. Measuring the first n qubits, we get the desired estimate.

Proof (6.2. Theorem): In the end we prove the main theorem, using theorem **6.5**. Recall we want a $O(1/\epsilon)$ runtime quantum algorithm, which distinguishes with at most $1/3$ error between $|\mu| \leq \epsilon/2$, and $|\mu| \geq \epsilon$. If $\epsilon > 0$ is given and $s \leq 1/16$, then let us examine the algorithm:

1. Do a quantum phase estimation on the start state $\mathbf{P}|0\rangle$ where the controlled gates are the previously defined $\mathbf{U} = REFL_p \cdot ROT_y$, let θ' be the output. Let the accuracy of the phase estimation be such as for the estimated θ, it holds that $\mathbb{P}(|\theta - \theta'| > \epsilon/6) \leq 1/9$. This accuracy can be achieved by $O(1/\epsilon)$ controlled-$\mathbf{U}$ gates.
2. If $|\theta'| > 142/100 \cdot \epsilon$, then **RETURN**$(|\mu| \leq \epsilon/2)$, else **RETURN**$(|\mu| \geq \epsilon)$.

This algorithm runs in $O(1/\epsilon)$ time and we are about to see that indeed at least with probability $2/3$ it distinguishes well by the input ϵ, as we want it in theorem **6.2.**. The only additional requirement here is that we want $s \leq 1/16$, whereas in theorem **6.2.** only $s \leq 1$ is required, but we can realize that if y (and also ϵ) is multiplied with $0 < c < 1$ constant (now with $1/16$), then it only affects the runtime by a constant factor.

Now we only need to see that, from theorem **6.5.**:

$$\mathbb{P}(4/5 \cdot 2|\mu| \leq |\theta| \leq 5/4 \cdot 2|\mu|) \geq 1 - 2/9$$

As the accuracy is $\epsilon/6$ for the estimation of θ' with at most $1/9$ error, the overall error is at most $1/9 + 2/9 = 1/3$ for the following:

$$4/5 \cdot |\mu| - \epsilon/12 \leq |\theta'/2| \leq 5/4 \cdot |\mu| + \epsilon/12$$

If $|\mu| \leq \epsilon/2$, then

$$|\theta'/2| \leq 5/8 \cdot \epsilon + \epsilon/12 < 71/100 \cdot \epsilon$$

And if $|\mu| \geq \epsilon$, then

$$|\theta'/2| \geq 4/5 \cdot \epsilon - \epsilon/12 > 71/100 \cdot \epsilon$$

References

[1] Quantum algorithm for stochastic optimal stopping problems with applications in finance - J. F. Doriguello, A. Luongo, J. Bao, P. Rebentrost, M. Santha (2021): https://arxiv.org/abs/2111.15332

[2] Mean estimation when you have the source code; or, quantum Monte Carlo methods - R. Kothari, R. O'Donnell (2022) https://arxiv.org/abs/2208.07544

[3] Quantum speedup of Monte Carlo methods - A. Montanaro (2017) https://arxiv.org/abs/1504.06987

[4] An Optimal Algorithm for Monte Carlo Estimation - P. Dagum, R. Karp, M. Luby, S. Ross (1995) https://ieeexplore.ieee.org/document/492471