

ARTICLE OPEN



Iterative quantum amplitude estimation

Dmitry Grinko^{1,2,3}, Julien Gacon^{1,2}, Christa Zoufal^{1,2} and Stefan Woerner¹✉

We introduce a variant of *Quantum Amplitude Estimation* (QAE), called *Iterative QAE* (IQAE), which does not rely on *Quantum Phase Estimation* (QPE) but is only based on *Grover's Algorithm*, which reduces the required number of qubits and gates. We provide a rigorous analysis of IQAE and prove that it achieves a quadratic speedup up to a double-logarithmic factor compared to classical Monte Carlo simulation with provably small constant overhead. Furthermore, we show with an empirical study that our algorithm outperforms other known QAE variants without QPE, some even by orders of magnitude, i.e., our algorithm requires significantly fewer samples to achieve the same estimation accuracy and confidence level.

npj Quantum Information (2021)7:52; <https://doi.org/10.1038/s41534-021-00379-1>

INTRODUCTION

Quantum Amplitude Estimation (QAE)¹ is a fundamental quantum algorithm with the potential to achieve a quadratic speedup for many applications that are classically solved through Monte Carlo (MC) simulation. It has been shown that we can leverage QAE in the financial service sector, e.g., for risk analysis^{2,3} or option pricing^{4–6}, and also for generic tasks such as numerical integration⁷. While the estimation error bound of classical MC simulation scales as $\mathcal{O}(1/\sqrt{M})$, where M denotes the number of (classical) samples, QAE achieves a scaling of $\mathcal{O}(1/M)$ for M (quantum) samples, indicating the aforementioned quadratic speedup.

The canonical version of QAE is a combination of *Quantum Phase Estimation* (QPE)⁸ and *Grover's Algorithm*. Since other QPE-based algorithms are believed to achieve exponential speedup, most prominently *Shor's Algorithm* for factoring⁹, it has been speculated as to whether QAE can be simplified such that it uses only Grover iterations without a QPE-dependency. Removing the QPE-dependency would help to reduce the resource requirements of QAE in terms of qubits and circuit depth and lower the bar for practical applications of QAE.

Recently, several approaches have been proposed in this direction. In ref. ¹⁰ the authors show how to replace QPE with a set of Grover iterations combined with a Maximum Likelihood Estimation (MLE), in the following called *Maximum Likelihood Amplitude Estimation* (MLAE). In ref. ¹¹, QPE is replaced by the Hadamard test, analog to *Kitaev's Iterative QPE*^{12,13} and similar approaches^{14,15}.

Both in refs. ¹⁰ and ¹¹ propose potential simplifications of QAE but do not provide rigorous proofs of the correctness of the proposed algorithms. In ref. ¹¹, it is not even clear how to control the accuracy of the algorithm other than possibly increasing the number of measurements of the evolving quantum circuits. Thus, the potential quantum advantage is difficult to compare and we will not discuss it in the remainder of this paper.

In ref. ¹⁶, another variant of QAE was proposed. There, for the first time, it was rigorously proven that QAE without QPE can achieve a quadratic speedup over classical MC simulation. Following¹⁶, we call this algorithm *QAE, Simplified* (QAES). Although this algorithm achieves the desired asymptotic complexity exactly (i.e., without logarithmic factors), the involved constants are very large, and likely to render this algorithm impractical unless further optimized—as shown later in this paper.

In the following, we propose a new version of QAE—called *Iterative QAE* (IQAE)—that achieves better results than all other tested algorithms. It provably has the desired asymptotic behavior up to a multiplicative $\log(2/\alpha \log_2(\pi/4\epsilon))$ factor, where $\epsilon > 0$ denotes the target accuracy, and $1 - \alpha$ the resulting confidence level.

Like in ref. ¹⁶, our algorithm requires iterative queries to the quantum computer to achieve the quadratic speedup and cannot be parallelized. Only MLAE allows the parallel execution of the different queries as the estimate is derived via classical MLE applied to the results of all queries. Although parallelization is a nice feature, the potential speedup is limited. Assuming the length of the queries is doubled in each iteration (like for canonical QAE and MLAE) the speedup is at most a factor of two since the computationally most expensive query dominates all the others.

With MLAE, QAES, and IQAE we have three promising variants of QAE that do not require QPE and it is of general interest to empirically compare their performance. Of similar interest is the question of whether the canonical QAE with QPE—while being (quantum) computationally more expensive—might lead to some performance benefits. To be able to better compare the performance of canonical QAE with MLAE, QAES, and IQAE, we extend QAE by a classical MLE postprocessing based on the observed results. This improves the results without additional queries to the quantum computer and allows us to derive proper confidence intervals.

QAE was first introduced in ref. ¹ and assumes the problem of interest is given by an operator \mathcal{A} acting on $n + 1$ qubits such that

$$\mathcal{A}|0\rangle_n|0\rangle = \sqrt{1-a}|\psi_0\rangle_n|0\rangle + \sqrt{a}|\psi_1\rangle_n|1\rangle, \quad (1)$$

where $a \in [0, 1]$ is the unknown, and $|\psi_0\rangle_n$ and $|\psi_1\rangle_n$ are two normalized states, not necessarily orthogonal. QAE allows estimating a with high probability such that the estimation error scales as $\mathcal{O}(1/M)$, where M corresponds to the number of applications of \mathcal{A} . To this extent, an operator $\mathcal{Q} = \mathcal{A}\mathcal{S}_0\mathcal{A}^\dagger\mathcal{S}_{\psi_0}$ is defined where $\mathcal{S}_{\psi_0} = \mathbb{I} - 2|\psi_0\rangle_n\langle\psi_0|_n \otimes |0\rangle\langle 0|$ and $\mathcal{S}_0 = \mathbb{I} - 2|0\rangle_{n+1}\langle 0|_{n+1}$ as introduced in¹. In the following, we denote applications of \mathcal{Q} as *quantum samples* or *oracle queries*.

The canonical QAE follows the form of QPE: it uses m ancilla qubits—initialized in equal superposition—to represent the final result, it defines the number of quantum samples as $M = 2^m$ and applies geometrically increasing powers of \mathcal{Q} controlled by the ancillas. Eventually, it performs an inverse QFT on the ancilla qubits before they are measured, as illustrated in Fig. 1.

¹IBM Quantum, IBM Research - Zurich, Rueschlikon, Switzerland. ²ETH Zurich, Zurich, Switzerland. ³University of Geneva, Geneva, Switzerland. ✉email: wor@zurich.ibm.com

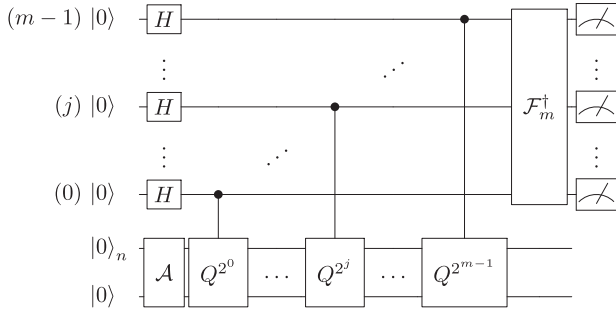


Fig. 1 Canonical QAE. Circuit with m ancilla qubits and $n + 1$ state qubits.

Subsequently, the measured integer $y \in \{0, \dots, M - 1\}$ is mapped to an angle $\theta_a = y\pi/M$. Thereafter, the resulting estimate of a is defined as $\tilde{a} = \sin^2(\theta_a)$. Then, with a probability of at least $8/\pi^2 \approx 81\%$, the estimate \tilde{a} satisfies

$$|a - \tilde{a}| \leq \frac{2\pi\sqrt{a(1-a)}}{M} + \frac{\pi^2}{M^2}, \quad (2)$$

which implies the quadratic speedup over a classical MC simulation, i.e., the estimation error $\epsilon = \mathcal{O}(1/M)$. The success probability can quickly be boosted to close to 100% by repeating this multiple times and using the median estimate². These estimates \tilde{a} are restricted to the grid $\{\sin^2(y\pi/M) : y = 0, \dots, M/2\}$ through the possible measurement outcomes of y .

Alternatively, and similarly to MLAE, it is possible to apply MLE to the observations for y . For a given θ_a , the probability of observing $|y\rangle$ when measuring the ancilla qubits is derived in¹ and given by

$$\mathbb{P}[|y\rangle] = \frac{\sin^2(M\Delta\pi)}{M^2\sin^2(\Delta\pi)}, \quad (3)$$

where Δ is the minimal distance on the unit circle between the angles θ_a and $\pi\tilde{y}/M$, and $\tilde{y} = y$ if $y \leq M/2$ and $\tilde{y} = M/2 - y$ otherwise. Given a set of y -measurements, this can be leveraged in an MLE to get an estimate of θ_a that is not restricted to grid points. Furthermore, it allows using the likelihood ratio to derive confidence intervals¹⁷. This is discussed in more detail in Supplementary Section 1. In our tests, the likelihood ratio confidence intervals were always more reliable than other possible approaches, such as the (observed) Fisher information. Thus, in the following, we will use the term QAE for the canonical QAE with the application of MLE to the y measurements to derive an improved estimate and confidence intervals based on the likelihood ratio.

All variants of QAE without QPE—including ours—are based on the fact that

$$\mathcal{Q}^k A|0\rangle_n|0\rangle = \cos((2k+1)\theta_a)|\psi_0\rangle_n|0\rangle + \sin((2k+1)\theta_a)|\psi_1\rangle_n|1\rangle, \quad (4)$$

where θ_a is defined as $a = \sin^2(\theta_a)$. In other words, the probability of measuring $|1\rangle$ in the last qubit is given by

$$\mathbb{P}[|1\rangle] = \sin^2((2k+1)\theta_a). \quad (5)$$

The algorithms mainly differ in how they derive the different values for the powers k of \mathcal{Q} and how they combine the results into a final estimate of a .

MLAE first approximates $\mathbb{P}[|1\rangle]$ for $k = 2^j$ and $j = 0, 1, 2, \dots, m - 1$, for a given m , using N_{shots} measurements from a quantum computer for each j , i.e., in total, \mathcal{Q} is applied $N_{\text{shots}}(M - 1)$ times, where $M = 2^m$. It has been shown in ref.¹⁰ that the corresponding Fisher information scales as $\mathcal{O}(N_{\text{shots}}M^2)$, which implies a lower bound of the estimation error scaling as $\Omega(1/(\sqrt{N_{\text{shots}}M}))$. Crucially¹⁰, does not provide an upper bound for the estimation error. Confidence intervals can be derived from the measurements

using, e.g., the likelihood ratio approach, see Supplementary Section 1.

In contrast to MLAE, QAES requires the different powers of \mathcal{Q} to be evaluated iteratively and cannot be parallelized. It iteratively adapts the powers of \mathcal{Q} to successively improve the estimate and carefully determines the next power of \mathcal{Q} . However, instead of a lower bound, a rigorous error upper bound is provided. QAES achieves the optimal asymptotic query complexity $\mathcal{O}(\log(1/a)/\epsilon)$, where $a > 0$ denotes the probability of failure. In contrast to the other algorithms considered, QAES provides a bound on the relative estimation error. Although the algorithm achieves the desired asymptotic scaling exactly, the constants involved are very large—likely too large for practical applications unless they can be further reduced.

In the following, we introduce a new variant of QAE without QPE. As for QAES, we provide rigorous performance proof. Although our algorithm only achieves the quadratic speedup up to a multiplicative factor $\log(2/a \log_2(\pi/4\epsilon))$, the constants involved are orders of magnitude smaller than for QAES. Moreover, in practice, this doubly logarithmic factor is small for any reasonable target accuracy ϵ and confidence level $1 - \alpha$, as we will show in the following.

RESULTS

Algorithm description

IQAE leverages similar ideas as refs.^{10,11,16} but combines them in a different way, which results in a more efficient algorithm while still allowing for a rigorous upper bound on the estimation error and computational complexity. As mentioned before, we use the quantum computer to approximate $\mathbb{P}[|1\rangle] = \sin^2((2k+1)\theta_a)$ for the last qubit in $\mathcal{Q}^k A|0\rangle_n|0\rangle$ for different powers k . In the following, we outline the rationale behind IQAE, which is formally given in Algorithm 1. The main sub-routine FINDNEXTK is outlined in Algorithm 2.

Suppose a confidence interval $[\theta_l, \theta_u] \subseteq [0, \pi/2]$ for θ_a and a power k of \mathcal{Q} , as well as an estimate for $\sin^2((2k+1)\theta_a)$. We can translate our estimates for $\sin^2((2k+1)\theta_a)$ into estimates for $\cos((4k+2)\theta_a)$. However, unlike in Kitaev's Iterative QPE, we cannot estimate $\sin((4k+2)\theta_a)$, and the cosine alone is only invertible without ambiguity if we know that the argument is restricted to either $[0, \pi]$ or $[\pi, 2\pi]$, i.e., the upper or lower half-plane. Thus, we want to find the largest k such that the scaled interval $[(4k+2)\theta_l, (4k+2)\theta_u]_{\text{mod } 2\pi}$ is fully contained either in $[0, \pi]$ or $[\pi, 2\pi]$. If this is given, we can invert $\cos((4k+2)\theta_a)$ and improve our estimate for θ_a with high confidence. This implies an upper bound of k , and the heart of the algorithm is the procedure used to find the next k given $[\theta_l, \theta_u]$, which is formally introduced in Algorithm 2 and illustrated in Fig. 2. In the following theorem, we provide convergence results for IQAE that imply the aforementioned quadratic speedup. The respective proof is given in Supplementary Section 2.

Theorem 1

(Correctness of IQAE). Suppose a confidence level $1 - \alpha \in (0, 1)$, a target accuracy $\epsilon > 0$, and a number of shots $N_{\text{shots}} \in \{1, \dots, N_{\text{max}}(\epsilon, a)\}$, where

$$N_{\text{max}}(\epsilon, a) = \frac{32}{(1 - 2\sin(\pi/14))^2} \log\left(\frac{2}{\alpha} \log_2\left(\frac{\pi}{4\epsilon}\right)\right). \quad (6)$$

In this case, IQAE (Algorithm 1) terminates after a maximum number of $\lceil \log_2(\pi/8\epsilon) \rceil$ rounds, where we define one round as a set of iterations with the same k_i , and each round consists of at most $N_{\text{max}}(\epsilon, a)/N_{\text{shots}}$ iterations. IQAE returns $[a_l, a_u]$ with $a_u - a_l \leq 2\epsilon$ and

$$\mathbb{P}[a \notin [a_l, a_u]] \leq \alpha. \quad (7)$$

Thus, $\tilde{a} = (a_l + a_u)/2$ leads to an estimate for a with $|a - \tilde{a}| \leq \epsilon$

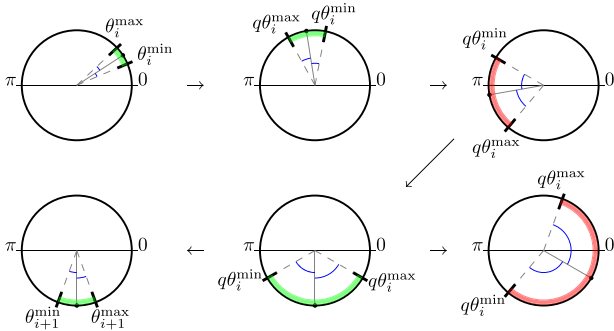


Fig. 2 FindNextK. Given an initial interval $[\theta_l, \theta_u]$, k_i and $K_i = 4k_i + 2$, FINDNEXTK determines the largest feasible k with $K = 4k + 2 \geq 2K_i$ such that the scaled interval $[K\theta_l, K\theta_u]_{\text{mod } 2\pi}$ lies either in the upper or in the lower half-plane, and returns k if it exists and k_i otherwise. The top left circle represents our initial knowledge about $K_i\theta_{a_i}$, while other circles represent extrapolations for different values of $q = K/K_i$. The top middle picture represents a valid q , the top right circle represents an invalid q , and so on. Note that the bottom right circle violates the condition $q \cdot |\theta_i^{\max} - \theta_i^{\min}| \leq \pi$, i.e., the interval is too wide and cannot lie in a single half-plane. The output of FINDNEXTK in the middle bottom circle and the left bottom figure show the improved result in the next iteration after additional measurements.

with a confidence of $1 - a$.

Furthermore, for the total number of \mathcal{Q} -applications, N_{oracle} it holds that

$$N_{\text{oracle}} < \frac{50}{\epsilon} \log \left(\frac{2}{a} \log_2 \left(\frac{\pi}{4\epsilon} \right) \right). \quad (8)$$

Algorithm 1

Iterative Quantum Amplitude Estimation

Function IQAE ($\epsilon, a, N_{\text{shots}}, ci$):

// ci is a chosen confidence interval method, which can be either Clopper-Pearson^{18,19} or Chernoff-Hoeffding²⁰

$i = 0$ // initialize iteration count

$k_i = 0$ // initialize power of \mathcal{Q}

$up_i = \text{True}$ // keeps track of the half-plane

$[\theta_l, \theta_u] = [0, \pi/2]$ // initialize conf. interval

$T = \lceil \log_2(\pi/8\epsilon) \rceil$ // max. number of rounds

calculate L_{max} according to Eqs. (9) and (10) // max. error on every iteration

while $\theta_u - \theta_l > 2\epsilon$ **do**

$i = i + 1$

$k_i, up_i = \text{FindNextK}(k_{i-1}, \theta_l, \theta_u, up_{i-1})$

set $K_i = 4k_i + 2$

if $K_i > \lceil L_{\text{max}}/\epsilon \rceil$ **then**

$N = \lceil N_{\text{shots}} L_{\text{max}}/\epsilon/K_i/10 \rceil$ // no-overshooting condition

else

$N = N_{\text{shots}}$

approximate $a_i = \mathbb{P}[|1\rangle]$ for the last qubit of $\mathcal{Q}^k \mathcal{A}|0\rangle_n|0\rangle$ by measuring N times

if $k_i = k_{i-1}$ **then**

combine the results of all iterations $j \leq i$ with $k_j = k_i$ into a single result, effectively increasing the number of shots

if $ci = \text{"Chernoff-Hoeffding"}$ **then**

$\epsilon_{a_i} = \sqrt{\frac{1}{2N} \log \left(\frac{2T}{a} \right)}$

$a_i^{\max} = \min(1, a_i + \epsilon_{a_i})$

$a_i^{\min} = \max(0, a_i - \epsilon_{a_i})$

if $ci = \text{"Clopper-Pearson"}$ **then**

$a_i^{\max} = I^{-1}(\frac{\alpha}{2T}; Na_i, N(1 - a_i) + 1)$

$a_i^{\min} = I^{-1}(1 - \frac{\alpha}{2T}; Na_i + 1, N(1 - a_i))$ // see Supplementary Eqs. (33)–(36)

calculate the confidence interval $[\theta_i^{\min}, \theta_i^{\max}]$ for $\{K_i\theta_{a_i}\}_{\text{mod } 2\pi}$ from $[a_i^{\min}, a_i^{\max}]$

and boolean flag up_i by inverting $a = (1 - \cos(K_i\theta))/27D2$

$\theta_l = \frac{|K_i\theta_l|_{\text{mod } 2\pi} + \theta_i^{\min}}{K_i}$

$\theta_u = \frac{|K_i\theta_u|_{\text{mod } 2\pi} + \theta_i^{\max}}{K_i}$

$[a_i, a_u] = [\sin^2(\theta_l), \sin^2(\theta_u)]$

return $[a_i, a_u]$

Algorithm 2

Procedure for finding k_{i+1}

Function FindNextK ($k_i, \theta_l, \theta_u, up_i, r = 2$):

$K_i = 4k_i + 2$ // current θ -factor

$\theta_i^{\min} = K_i\theta_l$ // lower bound for scaled θ

$\theta_i^{\max} = K_i\theta_u$ // upper bound for scaled θ

$K_{\text{max}} = \lfloor \frac{\pi}{\theta_u - \theta_l} \rfloor$ // set an upper bound for θ -factor

$K = K_{\text{max}} - (K_{\text{max}} - 2)_{\text{mod } 4}$ // largest potential candidate of the form $4k + 2$

while $K \geq rK_i$ **do**

$q = K/K_i$ // factor to scale $[\theta_i^{\min}, \theta_i^{\max}]$

if $\{q \cdot \theta_i^{\max}\}_{\text{mod } 2\pi} \leq \pi$ **and** $\{q \cdot \theta_i^{\min}\}_{\text{mod } 2\pi} \leq \pi$ **then**

// $[\theta_{i+1}^{\min}, \theta_{i+1}^{\max}]$ is in upper half-plane

$K_{i+1} = K$

$up_{i+1} = \text{True}$

$k_{i+1} = (K_{i+1} - 2)/4$

return (k_{i+1}, up_{i+1})

if $\{q \cdot \theta_i^{\max}\}_{\text{mod } 2\pi} \geq \pi$ **and** $\{q \cdot \theta_i^{\min}\}_{\text{mod } 2\pi} \geq \pi$ **then**

// $[\theta_{i+1}^{\min}, \theta_{i+1}^{\max}]$ is in lower half-plane

$K_{i+1} = K$

$up_{i+1} = \text{False}$

$k_{i+1} = (K_{i+1} - 2)/4$

return (k_{i+1}, up_{i+1})

$K = K - 4$

return (k_i, up_i) // return old value

More intuitively, the heart of our algorithm—the sub-routine FINDNEXTK—allows us to maximize Fisher Information \mathcal{I} on a given iteration in a greedy fashion. The way to see this is to notice, that \mathcal{I} is proportional to $N_{\text{shots}}K^2$, where $K := 4k + 2$ ¹⁰.

Note that the maximum number of applications of \mathcal{Q} given in Theorem 1 is a loose upper bound since the proof uses Chernoff-Hoeffding bound to estimate sufficiently narrow intermediate confidence intervals in Algorithm 1. Using more accurate techniques instead, such as Clopper-Pearson's confidence interval for Bernoulli distributions¹⁸, can lower the constant overhead in N_{oracle} by a factor of 3 (see Supplementary Section 3) but is more complex to analyze analytically.

In Algorithm 2, we require that $K_{i+1}/K_i \geq r = 2$, otherwise we continue with K_i . The choice of the lower bound r is optimal in the proof, i.e., it gives us the lowest coefficient for the upper bound (see Supplementary Section 2). Moreover, the chosen lower bound was working very well in practice.

In Algorithm 1 we imposed the "no-overshooting" condition in order to ensure, that we do not make unnecessary measurement shots at the last iterations of the algorithm. This condition also allows us to keep constants small in the proof (see Supplementary Eq. (11)). It utilizes a quantity L_{max} —the maximum possible error, which could be returned on a given iteration using N_{shots} measurements. It is calculated before the start of the algorithm for chosen ϵ, a and number of shots N_{shots} . It also depends on the type of chosen confidence interval. For Chernoff-Hoeffding one can write a direct analytical expression:

$$L_{\text{max}}(N_{\text{shots}}, \epsilon, a) := \arcsin \left(\frac{2}{N_{\text{shots}}} \log \left(\frac{2T(\epsilon)}{a} \right) \right)^{1/4}, \quad (9)$$

which is derived from Supplementary Eqs. (15) and (21). For Clopper-Pearson, one can only calculate it numerically:

$$L_{\text{max}}(N_{\text{shots}}, \epsilon, a) := \max_{\theta} h_{N_{\text{shots}}, \epsilon, a}(\theta), \quad (10)$$

where function h is defined in Supplementary Section 3. It is derived by analogy with Supplementary Eq. (39), where instead of $N_{\text{max}}(\epsilon, a)$ one should use N_{shots} .

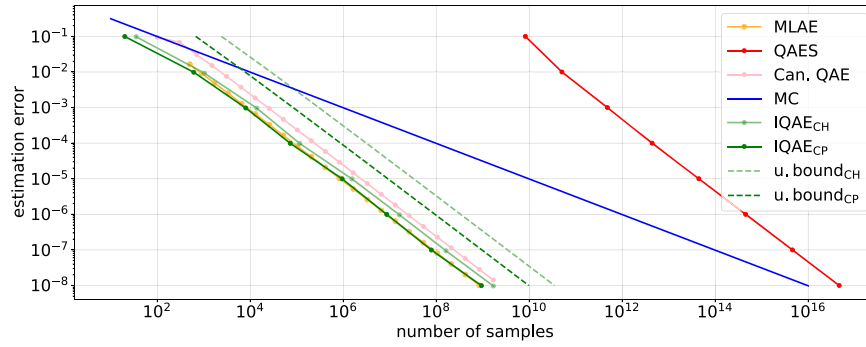


Fig. 3 Comparison of QAE variants. The resulting estimation error for $a = 1/2$ and 95% confidence level with respect to the required total number of oracle queries. We also include theoretical upper bounds for two versions of IQAE (CH = Chernoff-Hoeffding, CP = Clopper-Pearson). Note that QAES provides a relative error estimate, while the other algorithms return an absolute error estimate. For MLAE and canonical QAE we use the likelihood ratio confidence intervals. For MLAE we count the number of oracle calls for the largest power of \mathcal{Q} operator, which corresponds to the parallel execution of the algorithm. For IQAE, MLAE, and canonical QAE we used $N_{\text{shots}} = 100$.

Theorem 1 provides a bound on the query complexity, i.e., the total number of oracle calls with respect to the target accuracy. However, it is important to note that the computational complexity, i.e., the overall number of operations, including classical steps such as all applications of FINDNEXTK and computing the intermediate confidence intervals, scales in exactly the same way.

Numerical experiments

Next, we empirically compare IQAE, MLAE, QAES, QAE, and classical MC with each other and determine the total number of oracle queries necessary to achieve a particular accuracy. We are only interested in measuring the last qubit of $\mathcal{Q}^k \mathcal{A}|0\rangle_n |0\rangle$ for different powers k , and we know that $\mathbb{P}[|1\rangle] = \sin^2((2k+1)\theta_a)$. Thus, for a given θ_a and k , we can consider a Bernoulli distribution with corresponding success probability or a single-qubit R_y -rotation with angle $2(2k+1)\theta_a$ to generate the required samples. All algorithms mentioned in this paper are implemented and tested using Qiskit²¹ in order to be run on simulators or real quantum hardware, e.g., as provided via the *IBM Quantum Experience*.

For IQAE and MC, we compute the (intermediate) confidence intervals based both on Chernoff-Hoeffding²⁰ and on Clopper-Pearson¹⁸. For QAE and MLAE, we use the likelihood ratio¹⁷, see Supplementary Section 1. For QAES, we report the outputted accuracy of the algorithm.

To compare all algorithms we estimate $a = 1/2$ with a $1 - \alpha = 95\%$ confidence interval. For IQAE, MLAE, and QAE, we set $N_{\text{shots}} = 100$. As shown in Fig. 3, IQAE outperforms all other algorithms. QAES, even though achieving the best asymptotical behavior, performs worst in practice. On average, QAES requires about 10^8 times more oracle queries than IQAE which is even more than for classical MC simulation with the tested target accuracies. MLAE performs comparable to IQAE, however, the exact MLE becomes numerically challenging with increasing m . In order to observe the scaling of the quantum part of the algorithm, we collect more data points via the usage of a geometrically smaller search domain around estimated θ with each new round instead of brute force search on the whole initial domain for θ . Lastly, QAE with MLE-postprocessing performs a bit worse than IQAE and MLAE, which answers the question raised at the beginning: Applying QPE in the QAE setting does not lead to any advantage but only increases the complexity, even with an MLE-postprocessing. Thus, using IQAE instead does not only reduce the required number of qubits and gates, but it also improves the performance. Note that the MLE problem resulting from canonical QAE is significantly easier to solve than the problem arising in MLAE since the solution can be efficiently computed with a bisection search, see Supplementary Section 1. However, to evaluate QAE we need to simulate an



Fig. 4 Analysis of constant overhead for Chernoff-Hoeffding IQAE. The average (blue) and worst-case (orange) constant overhead for IQAE runs with different parameter settings.

increasing number of (ancilla) qubits, even for the very simple problem considered here, which makes the simulation of the quantum circuits more costly.

In the remainder of this section, we analyze the performance of IQAE in more detail. In particular, we empirically analyze the total number of oracle queries when using both Chernoff-Hoeffding and the Clopper-Pearson confidence intervals, as well as the resulting k -schedules.

More precisely, we run IQAE for all $a \in \{i/100 | i = 0, \dots, 100\}$ discretizing $[0, 1]$, for all $\epsilon \in \{10^{-j} | j = 3, \dots, 6\}$, and for all $\alpha \in \{1\%, 5\%, 10\%\}$. We choose $N_{\text{shots}} = 100$ for all experiments. For each combination of parameters, we evaluate the resulting number of total oracle calls N_{oracle} and compute

$$\frac{N_{\text{oracle}}}{\log(2/a \log_2(\pi/4\epsilon)) / \epsilon}, \quad (11)$$

i.e., the constant factor of the scaling with respect to ϵ and a . We evaluate the average, as well as the worst-case overall considered values for a . The results are illustrated in Figs. 4 and 5. The empirical complexity analysis of Chernoff-Hoeffding IQAE leads to:

$$N_{\text{oracle}}^{\text{avg}} \leq \frac{2}{\epsilon} \log\left(\frac{2}{\alpha} \log_2\left(\frac{\pi}{4\epsilon}\right)\right), \text{ and} \quad (12)$$

$$N_{\text{oracle}}^{\text{wc}} \leq \frac{6}{\epsilon} \log\left(\frac{2}{\alpha} \log_2\left(\frac{\pi}{4\epsilon}\right)\right), \quad (13)$$

where $N_{\text{oracle}}^{\text{avg}}$ denotes the average and $N_{\text{oracle}}^{\text{wc}}$ the worst-case complexity, respectively. Furthermore, the analysis of Clopper-Pearson

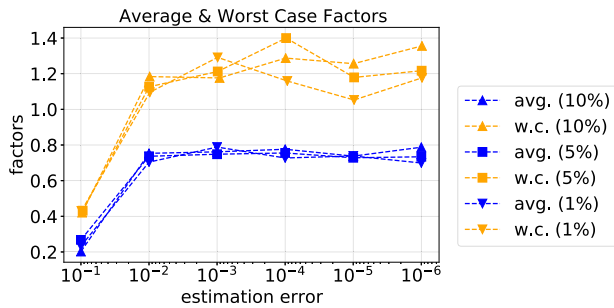


Fig. 5 Analysis of constant overhead for Clopper-Pearson IQAE. The average (blue) and worst-case (orange) constant overhead for IQAE runs with different parameter settings.

IQAE leads to:

$$N_{\text{oracle}}^{\text{avg}} \leq \frac{0.8}{\epsilon} \log \left(\frac{2}{a} \log_2 \left(\frac{\pi}{4\epsilon} \right) \right), \text{ and} \quad (14)$$

$$N_{\text{oracle}}^{\text{wc}} \leq \frac{1.4}{\epsilon} \log \left(\frac{2}{a} \log_2 \left(\frac{\pi}{4\epsilon} \right) \right), \quad (15)$$

To analyze the k -schedule, we set $a = 1/2$, $\epsilon = 10^{-6}$, $\alpha = 5\%$, and again $N_{\text{shots}} = 100$. Figure 6 shows for each iteration the resulting average, standard deviation, minimum, and maximum of K_{i+1}/K_i over 1000 repetitions of the algorithm, for the K_i defined in Algorithm 2. As explained before, we want to achieve as high a possible value of K_{i+1}/K_i for each iteration. Therefore, it can be seen that $N_{\text{shots}} = 100$ seems to be too small for the first round, i.e., another iteration with the same K_i is necessary before approaching an average growth rate slightly larger than four.

DISCUSSION

We introduced *Iterative Quantum Amplitude Estimation*, a new variant of QAE that realizes a quadratic speedup over classical MC simulation. Our algorithm does not require QPE, i.e., it is solely based on Grover iterations, and allows us to prove rigorous error and convergence bounds. We demonstrate empirically that our algorithm outperforms the other existing variants of QAE, some even by several orders of magnitude. This development is an important step towards applying QAE on quantum hardware to practically relevant problems and achieving a quantum advantage.

Our algorithm achieves the quadratic speedup up to a $\log(2/a \log_2(\pi/4\epsilon))$ -factor. In contrast, QAES, the other known variant of QAE without QPE and with a rigorous convergence proof, achieves optimal asymptotic complexity at the cost of very large constants. It is an open question for future research whether there exists a variant of QAE without QPE that is practically competitive while having an asymptotically optimal performance bound. Another difference between IQAE and QAES is the type of error bound: IQAE provides an absolute and QAES a relative bound. Both types are relevant in practice, however, in the context of QAE, where problems often need to be normalized, a relative error bound is sometimes more appropriate. We leave the question of a relative error bound for IQAE open to future research.

Another research direction that seems of interest is the existence of parallel versions of QAE. More precisely, is it possible to realize the powers of the operator Q distributed somehow in parallel over additional qubits, instead of sequential application on the quantum register? However, as shown in ref. ²², this does not seem to be possible.

Another open question for further investigation is the optimal choice of parameters for IQAE. We can set the required minimal growth rate for the oracle calls, as well as the number of classical

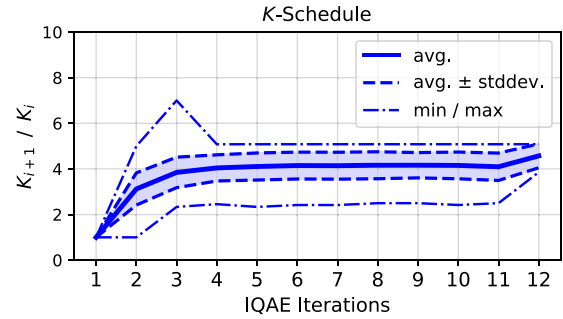


Fig. 6 K -schedule. Average, standard deviation, minimum, and maximum value of K_{i+1}/K_i per iteration over 1000 repetitions of Clopper-Pearson IQAE for $a = 1/2$, $\epsilon = 10^{-6}$, $\alpha = 5\%$ and $N_{\text{shots}} = 100$.

shots per iteration, and both affect the performance of the algorithm. Determining the most efficient setting may further reduce the required number of oracle calls for particular target accuracy.

We also demonstrated that the gap between the bound on the total number of oracle calls provided in Theorem 1 and the actual performance is not too big. The proof technique for the upper bound almost achieves the actual performance. However, one may still ask whether an even tighter analytic bound is possible.

To summarize, we introduced and analyzed a new variant of QAE without QPE that outperforms the other known approaches. Moreover, we provide a rigorous convergence theory. This helps to reduce the requirements on quantum hardware and is an important step towards leveraging quantum computing for real-world applications.

DATA AVAILABILITY

The data that support the findings of this study are available from the corresponding author upon justified request.

CODE AVAILABILITY

The mentioned algorithms are available open source as part of Qiskit and can be found in <https://github.com/Qiskit/qiskit/>. Tutorials explaining the algorithm and its application are located in <https://github.com/Qiskit/qiskit-tutorials>.

Received: 16 December 2019; Accepted: 1 February 2021;

Published online: 19 March 2021

REFERENCES

- Brassard, G., Hoyer, P., Mosca, M. & Tapp, A. Quantum amplitude amplification and estimation. *Contem. Mathemat.* **305**, 53–74 (2002).
- Woerner, S. & Egger, D. J. Quantum risk analysis. *npj Quantum Inf.* **5**, 1–8 (2019).
- Egger, D. J., Gutiérrez, R. G., Mestre, J. C. & Woerner, S. Credit Risk Analysis using Quantum Computers. <http://arxiv.org/abs/1907.03044> (2019).
- Rebentrost, P., Gupt, B. & Bromley, T. R. Quantum computational finance: Monte Carlo pricing of financial derivatives. <http://arxiv.org/abs/1805.00109> (2018).
- Stamatopoulos, N. et al. Option Pricing using Quantum Computers. <http://arxiv.org/abs/1905.02666> (2019).
- Zoufal, C., Lucchi, A. & Woerner, S. Quantum Generative Adversarial Networks for Learning and Loading Random Distributions. <http://arxiv.org/abs/1904.00043> (2019).
- Montanaro, A. Quantum speedup of Monte Carlo methods. <https://arxiv.org/pdf/1504.06987.pdf> (2017).
- Nielsen, M. A. & Chuang, I. L. *Quantum Computation and Quantum Information*. (Cambridge University Press, 2010).
- Shor, P. W. Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. *SIAM J. Comput.* **26**, 1484–1509 (1997).
- Suzuki, Y. et al. Amplitude Estimation without Phase Estimation. <http://arxiv.org/abs/1904.10246> (2019).
- Wie, C. R. Simpler quantum counting. *Quantum Info. Comput.* **19**, 967–983 (2019).

12. Kitaev, A. Y. Quantum measurements and the Abelian Stabilizer Problem. <http://arxiv.org/abs/quant-ph/9511026> (1995).
13. Kitaev, A. Y., Shen, A. & Vyalyi, M. N. *Classical and Quantum Computation*, no. 47 (American Mathematical Soc., 2002).
14. Svore, K. M., Hastings, M. B. & Freedman, M. Faster Phase Estimation. *Quantum Info. Comput.* **14**, 306–328 (2014).
15. Atia, Y. & Aharonov, D. Fast-forwarding of hamiltonians and exponentially precise measurements. *Nat. Commun.* **8**, 1–9 (2017).
16. Aaronson, S. & Rall, P. Quantum Approximate Counting, Simplified. <http://arxiv.org/abs/1908.10846> (2019).
17. Koch, K.-R. *Parameter Estimation and Hypothesis Testing in Linear Models*. (Springer-Verlag, Berlin Heidelberg, 1999).
18. Clopper, C. & Pearson, E. The use of confidence or fiducial limits illustrated in the case of the binomial. *Biometrika* **26**, 404–413 (1934).
19. Scholz, F. Confidence Bounds & Intervals for Parameters Relating to the Binomial, Negative Binomial, Poisson and Hypergeometric Distributions With Applications to Rare Events. <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.295.2790&rep=rep1&type=pdf> (2008).
20. Hoeffding, W. Probability inequalities for sums of bounded random variables. *J. Am. Statist. Assoc.* **58**, 13–30 (1963).
21. Abraham, H. et al. Qiskit: An Open-source Framework for Quantum Computing. <https://doi.org/10.5281/zenodo.2562110> (2019).
22. Burchard, P. Lower bounds for parallel quantum counting. Preprint at arXiv:1910.04555 (2019).

ACKNOWLEDGEMENTS

D.G., J.G., and C.Z. acknowledge the support of the National Centre of Competence in Research *Quantum Science and Technology* (QSIT). D.G. also acknowledges the support from ESOP scholarship from ETH Zurich Foundation. IBM, the IBM logo, and ibm.com are trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. The current list of IBM trademarks is available at <https://www.ibm.com/legal/copytrade>.

AUTHOR CONTRIBUTIONS

S.W. conceived the idea and co-supervised the work together with C.Z. D.G. developed IQAE and performed its theoretical analysis, J.G. worked on the MLE

postprocessing for canonical QAE. D.G. and J.G. jointly worked on the implementation and numerical experiments for all algorithms. D.G. and S.W. wrote the first draft of the manuscript and all authors contributed to its final version.

COMPETING INTERESTS

The authors declare no competing interests.

ADDITIONAL INFORMATION

Supplementary information The online version contains supplementary material available at <https://doi.org/10.1038/s41534-021-00379-1>.

Correspondence and requests for materials should be addressed to S.W.

Reprints and permission information is available at <http://www.nature.com/reprints>

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this license, visit <http://creativecommons.org/licenses/by/4.0/>.

© The Author(s) 2021