

Efficient Bayesian Phase Estimation

Nathan Wiebe¹ and Chris Granade^{2,3}

¹*Quantum Architectures and Computation Group, Microsoft Research, Redmond, WA (USA)*

²*Centre for Engineered Quantum Systems, Sydney, NSW (Australia)*

³*School of Physics, University of Sydney, Sydney, NSW (Australia)*

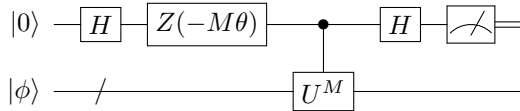
We provide a new method for performing phase estimation using adaptive Bayesian inference and rejection sampling. Our method not only yields comparable performance to iterative phase estimation in absence of decoherence, but also continues to work without modification when decoherence or other imperfections are present. Furthermore, our method is trivial to apply in any experimental system that is already capable of running iterative phase estimation experiments. This work not only shows that decoherence is not a fundamental limitation to our ability to learn eigenvalues and that ideas from computer vision and statistical inference can be leveraged to reveal new classes of efficient phase estimation algorithms that are more broadly applicable than existing methods.

I. INTRODUCTION

Phase estimation is a technique of fundamental importance to quantum information. Phase estimation provides a method to efficiently compute the eigenphase corresponding to an input eigenstate of a unitary U within a user specified precision. The protocol typically forms the core of quantum simulation algorithms, linear systems algorithms or and the amplitude estimation algorithm. As a result, methods that even modestly improve the time and space complexities of phase estimation or increase the algorithm's robustness to experimental errors are of great importance because they can allow sophisticated examples of such algorithms to be executed without the need for a fault tolerant quantum computer.

Iterative phase estimation seeks to learn the eigenvalue of a given eigenvector of U using a classical inference algorithm run in conjunction with a simple quantum circuit to infer the bits of a binary expansion of the phase. Iterative phase estimation is typically used in preference to traditional phase estimation because its simplicity and minimal qubit requirements

The quantum circuit used in iterative phase estimation is [1]



where $U|\phi\rangle = e^{i\phi}|\phi\rangle$ for an unknown eigenphase $\phi \in \mathbb{R}$, and where $Z(M\theta) = e^{iM\theta Z}$ is a rotation used to approximately invert U^M . The most commonly used method for performing iterative phase estimation uses this circuit to learn the bits of ϕ in order from least significant to most significant. The process is near optimal (up to \log^* factors) and affords a classically efficient and deterministic process for inferring these bits.

A drawback to iterative phase estimation is that it is not necessarily robust to experimental errors. This is because the inference process used to find the bits of the unknown phase is specialized for the probability distribution induced by an ideal phase estimation experiment,

which may not accurately model that yielded by an imperfect implementation. This issue becomes particularly significant as more bits of precision are required since typical phase estimation algorithms require that M grow exponentially. This means that initially weak decoherence eventually becomes a major impediment to our ability to learn an unknown eigenphase through phase estimation.

Bayesian methods proposed by Svore *et al* [1] are not as susceptible to these problems, as they do not require that the user chooses a particular sequence of M . Despite this, existing proposals for performing Bayesian phase estimation require an exponential amount of processing time to process the data from the experiments and also may suffer from the use of randomly chosen experiments in cases where a continuum of eigenphases are possible.

Our work addresses these problems by not only providing a method for performing an efficient form of Bayesian phase estimation but also giving an expedient heuristic for adaptively choosing approximately optimal experiments. We illustrate this by showing that our algorithm not only exhibits the same Heisenberg limited scaling of phase estimation in decoherence-free cases but continues to learn even when decoherence would ordinarily prevent traditional phase estimation methods from working.

II. BAYESIAN PHASE ESTIMATION

In order to understand how our algorithm works it is important to discuss how exact Bayesian phase estimation operates. The idea behind Bayesian phase estimation is to use Bayes' rule to estimate the most probable eigenphase given a set of experimental data. It keeps track of the probability that a particular eigenphase, ϕ , is the correct phase using a *prior distribution*. The initial prior distribution is subjective, but it is customarily taken to be flat unless the user has strong reason to suspect that some eigenphases are more probable than others. Importantly, Bayesian inference can recover from bad prior distributions at the cost of additional experiments, provided the true eigenphase is supported by the initial prior.

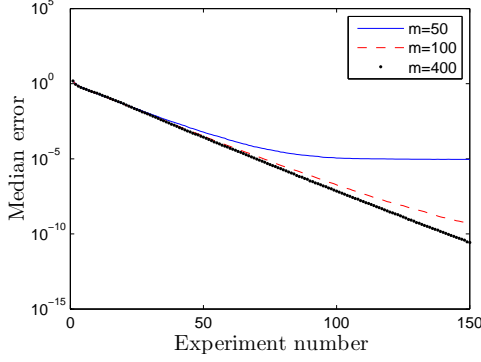


FIG. 1. Median errors in phase estimation for 10 000 random initial choices of the true eigenphase.

The inference process of Svore *et al* [1] involves performing a random set of experiments and then updating the prior distribution using Bayes' rule. For example, if an experiment is performed with using M repetitions of U , $Z(M\theta)$ and a measurement outcome $E \in \{0, 1\}$ is observed then Bayes' rule states that the *posterior probability* distribution for ϕ after observing the datum is

$$P(\phi|E; \theta, M) = \frac{P(E|\phi; \theta, M)P(\phi)}{\int P(E|\phi; \theta, M)P(\phi)d\phi}. \quad (1)$$

The final ingredient that is needed to perform an update is the likelihood function, $P(0|\phi; \theta, M)$, is easy to compute for phase estimation:

$$\begin{aligned} P(0|\phi; \theta, M) &= \frac{1 + \cos(M[\phi + \theta])}{2}, \\ P(1|\phi; \theta, M) &= \frac{1 - \cos(M[\phi + \theta])}{2}. \end{aligned} \quad (2)$$

After using (1) to update the posterior distribution we then set the prior distribution to equal the posterior distribution. This process is then repeated for each of the random experiments in the data set.

Unlike conventional methods, Bayesian inference returns a posterior distribution over the phase. An estimate of and uncertainty in the phase is extracted from the posterior distribution, usually as the mean and variance of the posterior, respectively. More sophisticated estimates of uncertainty, such as credible regions, can also be easily extracted from the posterior distribution [2]. This estimate of the uncertainty is vital for these algorithms because it not only gives the confidence in the result but also provides a stopping criteria. Since iterative phase estimation requires that you learn the bits in reverse order, this feature is significant because it allows the user to prematurely end the inference once an accuracy target has been met.

III. APPROXIMATE BAYESIAN PHASE ESTIMATION

The major drawback of Bayesian PE is that there are an infinite number of hypotheses in the prior. Our method addresses this by updating a model for the posterior distribution, rather than the underlying distribution itself. We do this by assuming a Gaussian model for our initial prior, perform a Bayesian update on samples drawn from the distribution and then refit the updated samples to a Gaussian. This strategy is commonly used in a number of particle filter approximations to the posterior such as the extended Kalman filter, assumed density filtering or sequential Monte Carlo methods.

These methods, although efficient, can require tracking several thousand discrete hypotheses for ϕ and are hard to implement. For this reason, we propose using a much simpler approach that uses rejection sampling to perform Bayesian inference. The procedure is as follows.

1. Perform experiment for given θ , M and observe outcome $E \in \{0, 1\}$.
2. Draw m samples from $\mathcal{N}(\mu, \sigma^2)$.
3. For each sample, ϕ_j , assign ϕ_j to Φ_{accept} with probability $P(E|\phi_j; \theta, M)/\kappa_E$, where $\kappa_E \in (0, 1]$ is a constant s.t. $P(E|\phi_j; \theta, M)/\kappa_E \leq 1$ for all ϕ_j, E .
4. Return $\mu = \mathbb{E}(\Phi_{\text{accept}})$ and $\sigma = \sqrt{\mathbb{V}(\Phi_{\text{accept}})}$.

The samples drawn from this distribution are sampled according to the posterior distribution $P(\phi|E; M, \theta)$. To see this, note that the probability density of a sample being accepted at $\phi = \phi_j$ is the product of the probability of acceptance and the probability density of drawing the sample from $\mathcal{N}(\mu, \sigma^2) = P(\phi)$. Eqn (1) then implies that

$$P(E|\phi; \theta, M)\mathcal{N}(\mu, \sigma^2) \propto P(\phi|E; \theta, M). \quad (3)$$

Thus the distribution of the accepted samples is equivalent to the posterior distribution.

The main issue that remains is how to optimally choose the parameters θ and M to minimize the number of experiments needed to infer the unknown eigenphase ϕ . This issue can be solved by local optimization of the risk function, but the resulting calculation can be too expensive to carry out in online experiments that provide experimental results at a rate of tens of Megahertz or faster. Fortunately, there is an expedient guess heuristic, known as the particle guess heuristic (PGH) that can be used to choose a near optimal experiment for this class of likelihood functions [3]. Specifically we take

$$M = \left\lceil \frac{1.25}{\sigma} \right\rceil, \quad \theta \sim P(\phi). \quad (4)$$

Here the constant factor of 1.25 comes from optimizing the median performance of the guess heuristic to the problem of phase estimation. M need not be an integer

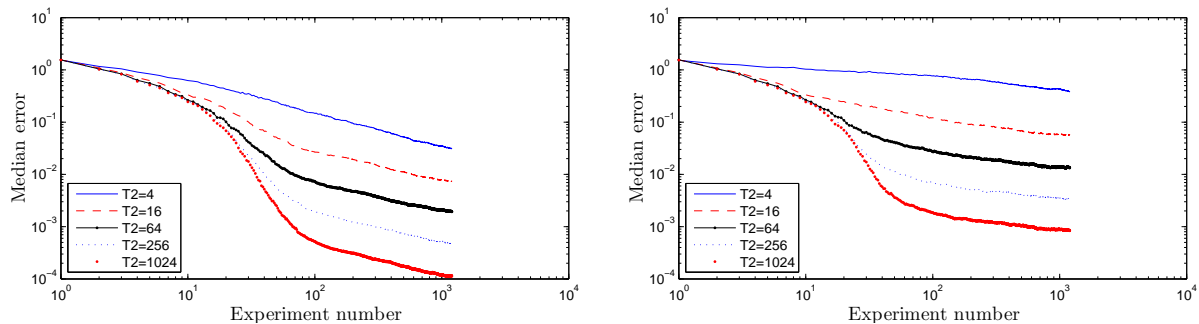


FIG. 2. Median errors for phase estimation in decohering systems for experiments constrained to use $M \leq T_2$ (left) and $M \leq T_2/10$ (right). We take $m = 12\,000$ and use 1 000 random samples to generate the data above. The initial state is taken to be a fixed but unknown eigenstate in all cases.

in cases where fractional applications of U are practical, such as quantum simulation.

We illustrate the performance of this algorithm in [Figure 1](#) wherein we plot the median error in phase estimation using our approach (for non-integer M and using a non-incremental formula for the standard deviation). The most obvious feature is that we observe Heisenberg limited scaling for $m > 100$, meaning that the error shrinks exponentially with the number of experiments (which is proportional to the evolution time under the PGH). Roughly 150 experiments are needed for the method to provide 32 bits of accuracy in the median case. We discuss the scaling in the mean in the appendix.

The number of experiments needed to reach error ϵ scales as $O(\log(1/\epsilon))$ rather than $O(\log(1/\epsilon) \log \log(1/\epsilon))$ as per iterative phase estimation. This illustrates that our adaptive approach can yield substantial performance advantages over existing methods for phase estimation. Although the number of experiments needed is small, the experimental time required to achieve this is not necessarily small. Given that the error shrinks as $\epsilon \in \Theta(e^{-\lambda N})$ where N is the experiment number then $T_{\text{exp}} \in O(\sum_{N=1}^{N_{\text{max}}} e^{\lambda N}) \in O(e^{\lambda N_{\text{max}}}) \in O(1/\epsilon)$. Thus the total experimental time required saturates the Heisenberg limit, up to a multiplicative constant.

A. Phase estimation with depolarizing noise

A criticism that has been levied lately at the phase estimation algorithm is that it can be impractical to execute on non-fault tolerant quantum hardware. This is because phase estimation attains its quadratic advantage over statistical sampling by using exponentially long evolutions. Decoherence causes the resultant phases to become randomized as time progresses, ultimately resulting in $\lim_{M \rightarrow \infty} P(0|\phi; \theta, M) = \frac{1}{2}$. In this limit, the measurements convey no information and inference becomes impossible. It is possible, however, to use our inference method to continue to use phase estimation long after the onset of decoherence. This observation is exactly analogous to the frequency estimation method presented

in [\[2\]](#), which uses particle filter methods to infer an unknown frequency parameter for a small quantum system in the presence of uncharacterized decoherence.

We model the effect of decoherence on the system by assuming the existence of a *decoherence time* T_2 such that

$$\begin{aligned} P(0|\phi) &= e^{-M/T_2} \left(\frac{1 + \cos(M[\phi - \theta])}{2} \right) + \frac{1 - e^{-M/T_2}}{2}, \\ P(1|\phi) &= e^{-M/T_2} \left(\frac{1 - \cos(M[\phi - \theta])}{2} \right) + \frac{1 - e^{-M/T_2}}{2}. \end{aligned} \quad (5)$$

In the limit as $T_2 \rightarrow \infty$ the likelihood function [\(5\)](#) approaches [\(2\)](#). Similarly as $T_2 \rightarrow 0$ the likelihood function approaches the uniform distribution. This model is appropriate when the time required to implement the controlled operation $\Lambda(U)$ is long relative the that required to perform H and an arbitrary Z -rotation, as is appropriate in quantum simulation. For simplicity, in the following we will assume that T_2 is well characterized.

The information provided by each experiment shrinks exponentially with M and hence it is important to choose M differently than one would choose it using the ordinary PGH. The best approach, suggested in [\[2\]](#), is to adaptively design experiments that minimize the Bayes risk given a prior distribution over T_2 . In practice, this can be very expensive given that data is often collected at rates of 1 MHz or greater in existing experiments. This means waiting for a locally optimal experiment means ignoring a prohibitive amount of data even if the optimization requires only a millisecond. For this reason we propose a variation to [\(4\)](#):

$$M = \min \left\{ \left\lceil \frac{1.25}{\sigma} \right\rceil, T_2 \right\}. \quad (6)$$

This heuristic is chosen to be qualitatively similar to optimized solutions for frequency estimation, which are chosen to saturate the Cramér-Rao bound [\[4\]](#). This can be seen through the following argument. The Cramér-Rao bound scales as $O(M^{-2})$ for phase estimation [\[5\]](#) in the absence of decoherence. Eqn. [\(5\)](#) suggests that

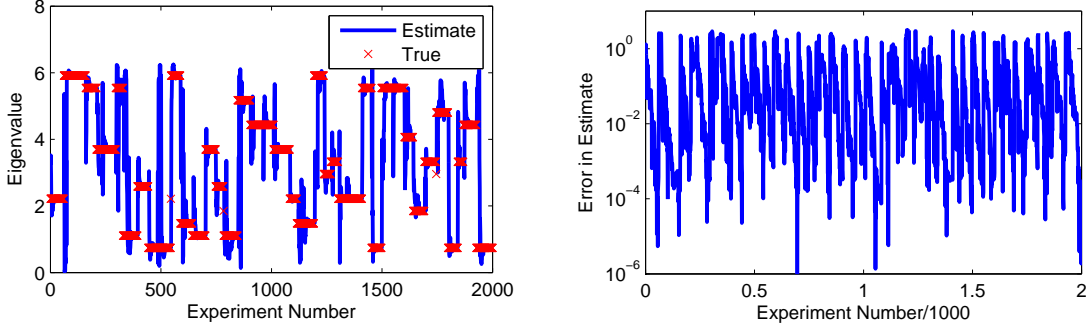


FIG. 3. Tracking the current estimate of an eigenphase during phase estimation for a system with 16 eigenvalues and $T_2 = 10^4$.

depolarizing noise causes the variance to increase as $O(\exp(2M/T_2))$. Calculus reveals that $M = T_2$ yields an optimal trade off between the two tendencies, as also observed in Figure 2.

The same methodology also applies to the case where T_2 is unknown and can easily be seen from the discussion in Granade *et al* [2]. The only modification needed is that the T_2 used in the determination of M should also be drawn from the prior $P(\phi, T_2)$.

There are several issues that need to be considered in order to make this approach practical. The first issue that arises in these cases is that since we cannot increase the evolution time without limit, the likelihood function begins to flatten. This creates an issue because the statistical fluctuations in the calculation in μ and σ can easily be equivalent to the effect of the Bayesian update unless m is large. In particular, the update process becomes asymptotically unstable if $m \in \Omega(1/\delta^2)$ where δ is the maximum size of fluctuations of $P(E|\phi)$. This can be understood by noting that the error in the mean of the accepted samples is $\Theta(1/\sqrt{m})$. The reduction in the uncertainty in the posterior distribution after an experiment is proportional to δ if the prior distribution is Gaussian. Thus if $m \in \Omega(1/\delta^2)$ then the sample noise in μ is greater than the reduction in σ , leading to instability. Either m must be increased or several experiments must be batched together to increase δ to address these issues.

The second issue is that, without modification, the probability that a given hypothesis survives p rounds of rejection sampling step shrinks exponentially with p . This can be combatted by taking

$$\kappa_E = \frac{1 + e^{-M/T_2}}{2}, \quad (7)$$

which makes the probability of accepting a sample 1 if $\theta = \phi_{\text{true}}$ even when $T_2 \gg M$.

Figure 2 shows that our algorithm smoothly interpolates between the Heisenberg limited scaling expected at short times and continues to learn (albeit only at a polynomial rate) when decoherence becomes significant. This illustrates that the presence of decoherence does not impose a fundamental limitation on the accuracy that we can achieve in phase estimation and that our approach.

Additionally, we see that forcing the experiments to stay in a more coherent regime can actually hurt the algorithm's ability to infer the phase as expected.

IV. TRACKING EIGENPHASES

The data in Figure 2 is appropriate when the initial quantum state is an eigenstate and is discarded after each experiment. Performing phase estimation in this way minimizes the number of experiments, but can be prohibitively expensive if preparation of the initial eigenstate is prohibitively costly. Instead, it makes sense to follow the standard prescription for phase estimation by keeping the quantum state until it is clear that the initial eigenstate has been depolarized. This approach can also be used when $T_2 = 0$ to cope with cases where the input state is a superposition of eigenvectors or when the learning algorithm becomes stuck.

The following procedure to addresses this issue. Furthermore, assume that the spectral gap is promised to be at most Δ .

1. After each update with probability e^{-M/T_2} perform an experiment with $\theta = \mu$ and $M = \tau/\sigma$ for $\tau < 1$.
2. If result = 1 then prepare initial state and reset σ .
3. Upon failure, continue updates as normal until $\sigma < \Delta$ then set σ and μ to the previous best values and proceed as normal until depolarization is detected.

The purpose of performing the experiment is to test whether the leading digits of the eigenphase are likely to be correct. If they are found to be incorrect, then the algorithm is restarted the prior distribution is reset to the uniform distribution and the eigenstate is re-learned until the user is confident that the state in the system is the eigenstate that was present in a prior experiment.

This process not only allows the initial state to be reused, but also permits the eigenvalue of an eigenstate in a decohering system to be estimated in real time as seen in Figure 3. There we take $\Delta = 0$, $\tau = 0.1$ and $T_2 = 10^4$ and notice that as the phase estimation algorithm can

rapidly detect a transition away from the instantaneous eigenstate that it had approximately projected the state onto and then project the maximally mixed state onto a new eigenstate.

V. CONCLUSION

To summarize, we have provided a new approach to Bayesian phase estimation that is not only efficient but also is resilient to experimental noise. The simplicity of the algorithm further means that it can be implemented in a few hours but also that it can be executed on a FPGA that is dedicated to controlling a quantum system. Also,

because our algorithm runs well in the presence of noise and imperfections in the system, it may allow much more sophisticated quantum simulation experiments to be run without raising the specter of decoherence ruining the experiment.

Looking forward, our work helps illustrate the fact that the problem of phase estimation has not been solved in its entirety. Rather, the problem of how to best infer an unknown eigenphase in the presence of experimental imperfections is, and will likely remain, an important problem in the foreseeable future. The flexibility of the framework that we present here does suggest that the Bayesian approaches, similar to that presented here, may be instrumental in making such applications practical.

-
- [1] K. M. Svore, M. B. Hastings, and M. Freedman, *Quantum Information & Computation* **14**, 306 (2014).
 - [2] C. E. Granade, C. Ferrie, N. Wiebe, and D. G. Cory, *New Journal of Physics* **14**, 103013 (2012).
 - [3] N. Wiebe, C. Granade, C. Ferrie, and D. Cory, *Physical Review Letters* **112**, 190501 (2014).
 - [4] C. Ferrie, C. E. Granade, and D. G. Cory, *Quantum Information Processing* **12**, 611 (2013).
 - [5] N. Wiebe, C. Granade, and D. G. Cory, *New Journal of Physics* **17**, 022005 (2015).
 - [6] N. Wiebe, A. Kapoor, and K. Svore, *arXiv:1401.2142 [quant-ph]* (2014).
 - [7] N. Wiebe, D. Braun, and S. Lloyd, *Physical Review Letters* **109**, 050505 (2012).

Appendix A: Pseudocode for algorithms

In the main body we sketched the details of our phase estimation algorithm. Here we elaborate on this algorithm and discuss some of the subtle details needed to make the algorithm work. The first such subtlety stems from the fact that eigenphases are equivalent modulo 2π . To see this problem, consider a Gaussian distribution centered at 0. If we take the outputs of the distribution in the branch $[0, 2\pi]$ then we find that the mean of the distribution is π rather than 0. Since the support of the initial Gaussian may be small at $\phi = \pi$, such errors can be catastrophic during the inference procedure. This can be dealt by using the circular mean and by working with a wrapped normal distribution. For expedience, we eschew this approach and instead use a heuristic approach that does not require a large number of trigonometric calls. This is especially important in cases where the algorithm is executed on a microcontroller or an FPGA where native trigonometric functions are not likely to be available.

The heuristic approach, described in [Algorithm 1](#), uses rejection sampling and incremental methods to estimate the mean and standard deviation of the posterior distribution. If $\sigma \ll 2\pi$ then the probability distribution is narrow and does not suffer from significant wrap around unless $\mu \bmod 2\pi \approx 0$. We address this by keeping track of each of the accepted ϕ_j as well as $\phi_j + \pi$. If $\sigma \ll 1$ then it is impossible that both distributions suffer from substantial wrap around. The arithmetic, rather than circular, mean and standard deviation are then computed for both using an incremental formula and the branch with the smallest variance is kept. If the branch that was shifted by π is chosen, then π is subtracted from the reported mean. The standard deviation does not need to be modified because it is invariant with respect to displacements of the mean.

While this approach is correct if $\sigma \ll 1$, it is only approximate if σ is on the order of 1. In such cases, computation of the circular mean is much better justified, however we find that using our heuristic approach continues to provide acceptable results while avoiding trigonometric calls that can be expensive in some contexts. An alternative approach to solving this problem is to begin each phase estimation run with a batch of random experiments, as per [\[1\]](#), before continuing to ensure that the posterior variance is small enough to neglect the wrap around effect.

The choice of the evolution time and the inversion angle strongly impacts the efficiency of the learning algorithm. We provide below code for a modified version of the particle guess heuristic of [\[6\]](#). As discussed in the main body, we expect that choosing $M > T_2$ will typically lead to worse estimates of the eigenphase because the effects of decoherence overwhelm the information that can be gleaned from these long experiments. As a result, we modify the particle guess heuristic in [\[3\]](#) to never choose $M > T_2$. We formally state this procedure in [Algorithm 3](#).

Algorithm 1 Bayes Update for RejS

Input: Prior mean and variance μ, σ , measurement E , settings M, θ , number of attempts m , scale κ_E

```

▷ Initialize accumulators to 0.
 $(\mu_{\text{inc}}, \mu'_{\text{inc}}, V_{\text{inc}}, V'_{\text{inc}}, N_a) \leftarrow 0$ 
▷ Attempt each sample.
for  $i \in 1 \rightarrow m$  do
  ▷ Draw a sample using each “cut” of the prior.
   $x \sim \frac{e^{-(\phi-\mu)^2/2\sigma^2}}{\sigma\sqrt{2\pi}},$ 
   $x \leftarrow x \bmod 2\pi.$ 
   $x' \leftarrow x + \pi \bmod 2\pi.$ 
  ▷ Accept or reject the new sample.
   $u \sim \text{Uniform}(0, 1)$ 
  if  $P(E|x) \geq \kappa_E u$  then
    ▷ Accumulate using the accepted sample w/ each “cut.”
     $\mu_{\text{inc}} \leftarrow \mu_{\text{inc}} + x$ 
     $V_{\text{inc}} \leftarrow V_{\text{inc}} + x^2$ 
     $V'_{\text{inc}} \leftarrow V'_{\text{inc}} + x'^2$ 
     $N_a \leftarrow N_a + 1.$ 
  end if
end for
▷ Return mean, variance of the posterior using accumulators.
 $\mu' \leftarrow \mu_{\text{inc}}/N_a$ 
 $\sigma' \leftarrow \min\left(\sqrt{\frac{1}{N_a-1}}(V_{\text{inc}} - \mu_{\text{inc}}^2), \sqrt{\frac{1}{N_a-1}}(V'_{\text{inc}} - \mu'^2_{\text{inc}})\right)$ 
return  $(\mu', \sigma')$ 

```

Algorithm 2 PGH for decoherent phase estimation using RejS

Input: Prior RejS state μ, Σ . Resampling kernel F .

Output: An experiment (M, θ) .

```

function PGHRejS ( $\mu, \Sigma, T_2$ )
   $M \leftarrow 1.25/\sqrt{\text{Tr}(\Sigma)}$ 
  if  $M \geq T_2$  then
     $M \sim f(x; 1/T_2)$ 
  end if
   $(-\theta/M) \sim F(\mu, \Sigma)$ 
  return  $(M, \theta).$ 
end function

```

▷ Draw M from an exponential distribution with mean T_2 .

Appendix B: Variance reduction strategies

An important drawback of our approach is that in typical applications the tails can be quite fat, meaning that there is significant probability that the error in the inferred eigenphase is orders of magnitude greater than the median. Such a distribution can be seen in [Figure 4](#) where we see that although the median error is roughly 10^{-10} radians after 100 updates for $m > 50$, a non-negligible fraction of the experiments have error on the order of 1.

Fortunately, the need to always repeat the algorithm and use a majority voting scheme to reduce the variance of the estimate is mitigated by the fact that the algorithm outputs σ which estimates the uncertainty in the resultant eigenphase. Nonetheless, alternative strategies exist for reducing the variance of the estimates of the phase after a fixed number of experiments. Standard approaches involve random restarting and the use of multi-modal distributions. Here we focus on heuristics for detecting when the learning algorithm fails to converge.

The central idea behind our restarting strategy is to examine the decay of σ with the number of experiments. In ideal cases, the error decays exponentially which means that it is easy to see when the algorithm fails by plotting σ on a semilog plot. Such intuition can be easily automated. The idea behind our restarting algorithm is to first estimate the derivative of $\log(\sigma)$ with respect to the experiment number and determine whether it is less than a threshold. If

Algorithm 3 Restarting algorithm

Input: Prior ReJS state, records of all previous models found in the phase estimation algorithm $\mu, \vec{\sigma}$, initial standard deviation σ_{init} , a counter CNT, Γ and τ .

Output: CNT, σ

```

function RESTART( $\mu, \vec{\sigma}, T_2$ )
   $D \leftarrow$  derivative of  $\log \sigma$ .
  if CNT < 5 then                                 $\triangleright$  Checks to see if enough points have been considered to accurately estimate gradient.
    CNT  $\leftarrow$  CNT + 1
    return CNT,  $\sigma$ .
  else if  $D \geq \Gamma$  then
    Perform experiment with  $M = \tau/\sigma$  and  $\theta = \mu$ .
    if Outcome is 0 then                             $\triangleright$  Test concludes state estimate is valid
      return CNT,  $\sigma$ .
    else                                             $\triangleright$  Test concludes state estimate is invalid
      CNT  $\leftarrow$  0
       $\sigma \leftarrow \sigma_{\text{init}}$ 
      return CNT,  $\sigma$ 
    end if
  end if
end function

```

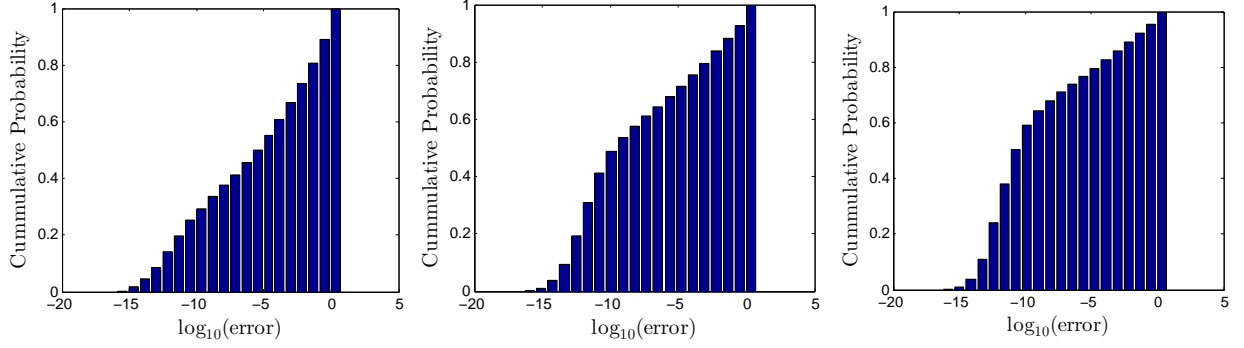


FIG. 4. Cumulative distribution function of probability that PE error is less than x after 150 updates for $m = 50$ (left) $m = 100$ (middle) $m = 200$ (right).

it is less than the threshold, perform an experiment to test to see if the value of μ that has been learned so far is accurate (as per our incoherent phase estimation algorithm). If it is found to be inaccurate then restart the algorithm and abandon the information learned so far.

When a restarting strategy like this is employed, we need to modify the model selection criteria. Previously, we used the value of μ yielded by the most recent experiment as our estimate. Given that a restart late in the algorithm can reset all information learned about a model, it makes sense in this context to use the value of μ corresponding to the smallest σ observed. This corresponds to the model that the inference algorithm has the greatest certainty.

Appendix C: Additional Numerics

Our algorithm is similarly robust against *uncharacterized noise sources* as well. We demonstrate this in [Figure 5](#) wherein we introduce depolarizing noise of strength γ to our experimental system, but do not include such noise in the likelihood function. For example, with $\gamma = 0.4$, the measurement outcome of phase estimation is replaced with a random bit with probability 40%. We find that while the inclusion of such noise causes the error decay exponent to shrink as roughly $0.17e^{-3.1\gamma}$ it does not prevent our algorithm from learning at an exponential rate (until depolarizing noise becomes significant). This illustrates that our method for phase estimation can continue to work even in the presence of noise sources that are both strong and uncharacterized.

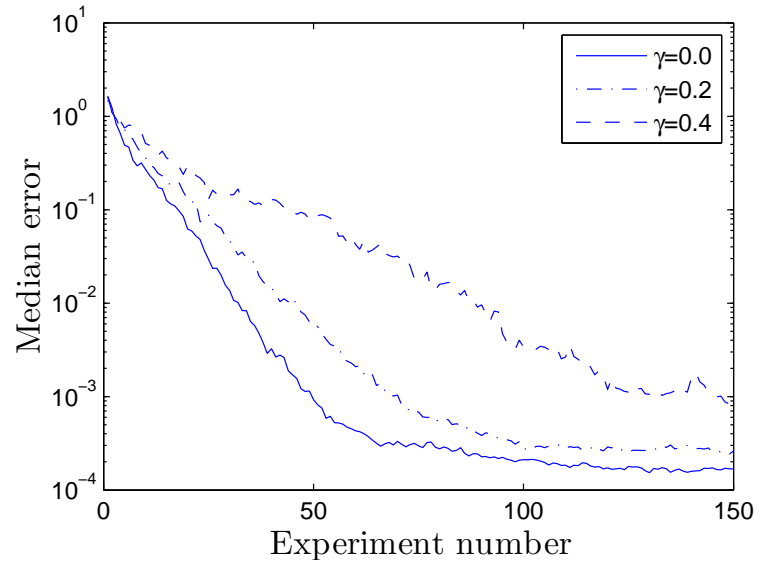


FIG. 5. Errors in inference for phase estimation with different levels of un-modeled noise, γ , for $T_2 = 1000$. In each instance the initial state was taken to be a fixed, but unknown, eigenstate of the Hamiltonian. The median was found for 100 randomly chosen values of the eigenphase of this fixed initial eigenstate.