Various noise models have been developed in quantum computing study to describe the propagation and effect of the noise that is caused by imperfect implementation of hardware. Identifying parameters such as gate and readout error rates is critical to these models. We use a Bayesian inference approach to identify posterior distributions of these parameters such that they can be characterized more elaborately. By characterizing the device errors in this way, we can further improve the accuracy of quantum error mitigation. Experiments conducted on IBM’s quantum computing devices suggest that our approach provides better error mitigation performance than existing techniques used by the vendor. Also, our approach outperforms the standard Bayesian inference method in some scenarios.

CCS Concepts: • Hardware → Quantum error correction and fault tolerance; • Mathematics of computing → Bayesian computation;

Additional Key Words and Phrases: Error mitigation, gate error, measurement error, Bayesian statistics

ACM Reference format:
Muqing Zheng, Ang Li, Tamás Terlaky, and Xiu Yang. 2023. A Bayesian Approach for Characterizing and Mitigating Gate and Measurement Errors. ACM Trans. Quantum Comput. 4, 2, Article 11 (February 2023), 21 pages.
https://doi.org/10.1145/3563397
advantages of QC algorithms [23]. One of the solutions to this problem is quantum error correction (QEC) [1, 9, 13, 16, 18, 25], which utilizes redundancy to protect the information of a single “logic qubit” from errors. Two representative examples are surface code and color code due to their scalability and high error thresholds [18, 25]. An alternative approach to QEC is bosonic codes. In this coding scheme, the single-qubit information is encoded into a higher-dimensional system, like a harmonic oscillator. One advantage of Bosonic codes is that they provide access to larger Hilbert space with less overhead than traditional QEC codes [9, 13, 16].

However, as described in [23], in the “noisy intermediate-scale quantum (NISQ)” era, the small- or medium-sized but noisy quantum computers cannot afford the cost of QEC codes because they impose a heavy overhead cost in number of qubits and number of gates. As a result, quantum error mitigation (QEM) techniques have become attractive, e.g., [4, 7, 8, 10–12, 17, 19, 30, 31], since their cost is much lower than the QEC codes in terms of the circuit depth and the number of qubits. One important area in the error mitigation study is to filter the measurement errors (or readout errors). These errors are usually modeled by multiplying a stochastic matrix with a probability vector to depict the influence of the noise on the output of QC algorithms. More precisely, the probability vector represents the desired noiseless output of a QC algorithm, the stochastic matrix describes how the noise affects this output, and the resulting vector consists of the probabilities of observing each possible state on the quantum device. Here, the stochastic matrix can be constructed from conditional probabilities if only classical errors are considered, or from results of tomography if non-classical errors are not significant [4, 7, 11, 19]. Similarly, the study in [29] shows the possibility to simulate bit-flip gate error in some quantum circuits in a classical manner.

The goal of QEM from the algorithmic perspective is to recover noise-free information using data from repeated experiments, which is usually achieved via statistical methods. In the existing error models, the parameters, e.g., error rate of measurement or gates, are usually considered as deterministic values (possibly with confidence interval), and the goal is to filter the error in estimating the expectation of an operator. Instead, by considering error mitigation as a stochastic inverse problem, we adopt a new Bayesian algorithm from [6] to construct the distributions of model parameters and use corresponding backward error models to filter errors from the outcomes of a quantum device. Note that our framework does not rely on the specific knowledge of the problems that quantum circuits want to answer, like in [12], or hardware calibration, such as [3, 26]. We aim to estimate the parameters more comprehensively for selected error models as an inverse problem, while error mitigation is achieved by using the error model in a backward direction.

The article is organized as follows. In Section 2, we provide the measurement error model based on independent classical measurement error and expand the gate error model in [29] to a multiple-error scenario. In Section 3, we introduce the use of the Bayesian algorithm in [6] to infer the distributions of parameters of measurement error and gate error models. Then, we demonstrate the creation of our error filter on IBM’s quantum device ibmqx2 (Yorktown) and apply our filter together with other existing error mitigation methods to the measurement outcome from state tomography, an example of Grover’s search [15], an instance of the Quantum Approximate Optimization Algorithm (QAOA) [15], and a 200-NOT-gate circuit in Section 4. The code is available in [32].

2 ERROR MODELS

The goals of our error modeling include estimating the influence of bit-flip gate errors and measurement errors in the outputs of a quantum circuit without accessing any quantum device and recovering the error-free (or error-mitigated) output. Throughout this article, we assume no state-preparation error and only focus on pure state measurements. The three error rates that we care about are as follows:
(1) $\epsilon_g$ = the chance of having a bit-flip error in a gate  
(2) $\epsilon_{m_0}$ = the chance of having a measurement error when measure $|0\rangle$  
(3) $\epsilon_{m_1}$ = the chance of having a measurement error when measure $|1\rangle$

It is reasonable to consider $\epsilon_g \neq 0.5$ and $\epsilon_{m_0} + \epsilon_{m_1} \neq 1$ in the current quantum computer [3, 4]. This assumption is one of the necessary conditions for the existence of the error mitigation solutions in our following models.

### 2.1 Measurement Error

As is demonstrated in [7], classical measurement error is applicable in the device we conduct experiments on, i.e., ibmqx2. We build a measurement error model using conditional probabilities. Consider a single-qubit state $\alpha |0\rangle + \beta |1\rangle$; its distribution of the noisy measurement outcomes is as follows:

$$
\begin{align*}
\text{Pr}(\text{Measure 0 w/noise}) &= |\alpha|^2 \cdot (1 - \epsilon_{m_0}) + |\beta|^2 \cdot \epsilon_{m_1}, \\
\text{Pr}(\text{Measure 1 w/noise}) &= |\alpha|^2 \cdot \epsilon_{m_0} + |\beta|^2 \cdot (1 - \epsilon_{m_1}),
\end{align*}
$$

which is equivalent to

$$
\begin{bmatrix}
1 - \epsilon_{m_0} & \epsilon_{m_1} \\
\epsilon_{m_0} & 1 - \epsilon_{m_1}
\end{bmatrix}
\begin{bmatrix}
\text{Pr}(\text{Measure 0 w/o noise}) \\
\text{Pr}(\text{Measure 1 w/o noise})
\end{bmatrix}
= 
\begin{bmatrix}
\text{Pr}(\text{Measure 0 w/noise}) \\
\text{Pr}(\text{Measure 1 w/noise})
\end{bmatrix},
$$

(1)

where “w/” stands for “with” and “w/o” stands for “without.” Denoting $\epsilon_{m_0}$ and $\epsilon_{m_1}$ for qubit $i$ as $\epsilon_{m_0,i}$ and $\epsilon_{m_1,i}$, respectively, we can extend the matrix form in Equation (1) to an $n$-qubit case:

$$
A r = \bar{r},
$$

(2)

where

$$
A := \bigotimes_{i=1}^{n} 
\begin{bmatrix}
1 - \epsilon_{m_0,i} & \epsilon_{m_1,i} \\
\epsilon_{m_0,i} & 1 - \epsilon_{m_1,i}
\end{bmatrix},
$$

$$
r := 
\begin{bmatrix}
\text{Pr}(\text{Measure 0...00 w/o noise}) \\
\text{Pr}(\text{Measure 0...01 w/o noise}) \\
\vdots \\
\text{Pr}(\text{Measure 1...11 w/o noise})
\end{bmatrix},
$$

$$
\bar{r} := 
\begin{bmatrix}
\text{Pr}(\text{Measure 0...00 w/noise}) \\
\text{Pr}(\text{Measure 0...01 w/noise}) \\
\vdots \\
\text{Pr}(\text{Measure 1...11 w/noise})
\end{bmatrix},
$$

$A_{ij} \in [0, 1], r_i \in [0, 1], \bar{r}_i \in [0, 1]$ by introducing the independence of measurement errors across qubits. We aim to identify $r$, but, in practice, we only have $\bar{r}$, which is the probability vector characterizing the observed results from repeated measurements. Note that $A$ is a non-negative left stochastic matrix (i.e., each column sums to 1), so if $\bar{r} \geq 0$ and its entries sum to 1, $\bar{r} \geq 0$ and its entries also sum to 1.

If $\epsilon_{m_0,i}$ and $\epsilon_{m_1,i}$ for all $i = 1, \ldots, n$ are known, the most straightforward denoising method derived from Equation (2) is $r := A^{-1}\bar{r}$. As $\epsilon_{m_0,i} + \epsilon_{m_1,i} \neq 1$ for all $i = 1, \ldots, n$, each individual 2-by-2 matrix has a non-zero determinant. Thus, $A$ has a non-zero determinant and $A^{-1}$ exists. However, it is not guaranteed that $r^*$ is a valid probability vector. An alternative is to find a constrained
approximation:
\[
    r^* := \arg \min_{\sum_{i=1}^{2^n} r_i = 1, \forall i \in [1, \ldots, 2^n]} \|Ar - \tilde{r}\|_2.
\] (3)

2.2 Bit-flip Gate Error

For the gate error, we focus on the single bit-flip error in this work, and we adopt the error model proposed in [29]. Of note, there is no direct proof in [29] to validate this model. In this section, we complete the proof and also extend it to a multiple-error case. We first consider the case when there is only one gate and qubits could have bit-flip errors (all in the same rate \(\epsilon\)) after this gate, as shown in Figure 1.

2.2.1 Single Bit-flip Error. Let \(p : \{0, 1\}^n \rightarrow [0, 1]\), where \(n\) is the number of qubits, be the Boolean function that represents the noise-free probability distribution of the outcome of a QC algorithm and \(x \in \{0, 1\}^n\) denote the basis used in a QC algorithm. The Fourier expansion of this Boolean function is
\[
    p(x) = \sum_{s \in \{0, 1\}^n} \hat{p}(s)(-1)^{s \cdot x},
\] (4)
where \(\hat{p}(s)\) is the Fourier coefficient of \(p\) and \(s \cdot x = \sum_{i=1}^{n} s_i \cdot x_i\) [22, p. 22]. These Fourier coefficients can be computed from
\[
    \hat{p}(s) = \frac{1}{2^n} \sum_{x \in \{0, 1\}^n} p(x)(-1)^{s \cdot x}.
\]

Let \(y\) be the erroneous version of \(x\) induced by the bit-flip error. In other words, \(y\) is a function of \(x\) that adds bit-flip error into the measurement outcomes. The mathematical expression of \(y\) is
\[
    y_i = \begin{cases} 
    x_i & \text{with probability } 1 - \epsilon \\epsilon_g \\
    \neg x_i & \text{with probability } \epsilon \\epsilon_g 
    \end{cases} \text{ for } i = 1, \ldots, n.
\]
Define \(\tilde{p} : \{0, 1\}^n \rightarrow [0, 1]\) to be the expected distribution function of measurement outcomes under the noise model. Then Equation (4) implies
\[
    \tilde{p}(x) = E_x[p(y)] = \sum_{s \in \{0, 1\}^n} \hat{p}(s)E_x[(-1)^{s \cdot y}].
\]

It is clear that
\[
    E_x[(-1)^{s \cdot y}] = E_y \left[ \prod_{i=1}^{n} (-1)^{s_i \cdot y_i} \right] = \prod_{i=1}^{n} E_y \left[ (-1)^{s_i \cdot y_i} \right] = \prod_{i=1}^{n} \left[ (1 - \epsilon) \cdot (-1)^{s_i \cdot x_i} + \epsilon \epsilon_g \cdot (-1)^{s_i \cdot \neg x_i} \right].
\] (5)
Since $x_i$ and $s_i$ are binary bits, there are four possible cases:

- $s_i = 0, x_i = 0$, then $(1 - \epsilon_g)(-1)^{s_i \cdot x_i} + \epsilon_g(-1)^{s_i \cdot \neg x_i} = 1$
- $s_i = 0, x_i = 1$, then $(1 - \epsilon_g)(-1)^{s_i \cdot x_i} + \epsilon_g(-1)^{s_i \cdot \neg x_i} = 1$
- $s_i = 1, x_i = 0$, then $(1 - \epsilon_g)(-1)^{s_i \cdot x_i} + \epsilon_g(-1)^{s_i \cdot \neg x_i} = 1 - 2\epsilon_g$
- $s_i = 1, x_i = 1$, then $(1 - \epsilon_g)(-1)^{s_i \cdot x_i} + \epsilon_g(-1)^{s_i \cdot \neg x_i} = (1 - 2\epsilon_g) \cdot (-1)$

To summarize,

$$(1 - \epsilon_g)(-1)^{s_i \cdot x_i} + \epsilon_g(-1)^{s_i \cdot \neg x_i} = (1 - 2\epsilon_g)^{s_i}(-1)^{s_i \cdot x_i},$$

for all $s_i \in \{0, 1\}$ and $x_i \in \{0, 1\}$. Consequently, continuing from Equation (5),

$$E_y[(-1)^x] = \prod_{i=1}^n (1 - 2\epsilon_g)^{s_i}(-1)^{s_i \cdot x_i} = (1 - 2\epsilon_g)^{s_1}(-1)^{s_1 \cdot x_1},$$

where $|s| = \sum_{i=1}^n s_i$. Thus, the $\tilde{p}$ with only one bit-flip error is

$$\tilde{p}(x) = E_x[p(y)] = \sum_{s \in \{0, 1\}^n} (1 - 2\epsilon_g)^{|s|} \tilde{p}(s)(-1)^{s \cdot x}.$$  

### 2.2.2 Extension to Multiple Bit-flip Errors

The extension is only applicable on gates that commute with the $X$ gate up to a global phase factor. This commutativity condition allows us to move occurred bit-flip errors to the end of the circuit, like the change from Figures 2(a) to 2(b), where $U_1, \ldots, U_m$ are still noise-free unitary gates. The model is constructed by repeatedly applying the previous proof procedure instead of considering the cancellation of errors, since our interest is on individual gates but not on the accumulated one.

The expected distribution function $\tilde{p}$ of circuit $U_m \cdots U_1 |\phi\rangle$ with up to $m$ layers of bit-flip errors can be recursively defined by

$$\tilde{p}^{(1)}(x) := E_x[p(y^{(1)})],$$

$$\tilde{p}^{(j)}(x) := E_x[p^{(j-1)}(y^{(j-1)})] \quad \text{for} \ j = 2, \ldots, m,$$

$$\tilde{p}(x) = \tilde{p}^{(m)}(x) = E_x[p^{(m-1)}(y^{(m-1)})],$$

where $p$ is the error-free output distribution,

$$y_i^{(1)} = \begin{cases} x_i & \text{with probability } 1 - \epsilon_g, \\ \neg x_i & \text{with probability } \epsilon_g \end{cases}, \quad \text{and} \quad y_i^{(j)} = \begin{cases} y_i^{(j-1)} & \text{with probability } 1 - \epsilon_g, \\ \neg y_i^{(j-1)} & \text{with probability } \epsilon_g \end{cases},$$

for $i = 1, \ldots, n$ and $j = 2, \ldots, m$ (to avoid confusion, the superscripts on $\tilde{p}$ are just indices). Because the expectations are all over $x$ and not $s$, we can repeat the process in Section 2.2.1 $m$ times. Each
repetition provides a \( (1 - 2\epsilon_g)^{|s|} \) term in the multiplication:

\[
\tilde{p}(x) = \sum_{s \in \{0, 1\}^n} \left( \prod_{j=1}^{m} (1 - 2\epsilon_g)^{|s|} \right) \hat{p}(s)(-1)^{s \cdot x} = \sum_{s \in \{0, 1\}^n} (1 - 2\epsilon_g)^{|s|} \hat{p}(s)(-1)^{s \cdot x}.
\]

Equation (6) is also straightforward to be compatible with the case when each layer of bit-flip errors has a different error rate by indexing \( \epsilon_g \) with \( j \)

\[
\tilde{p}(x) = \sum_{s \in \{0, 1\}^n} \left( \prod_{j=1}^{m} (1 - 2\epsilon_{g,j})^{|s|} \right) \hat{p}(s)(-1)^{s \cdot x}.
\]

2.2.3 Bit-flip Error Filter. Let \( j_b \) be the binary representation of a non-negative integer \( j \). Given \( \epsilon_g \) and \( \tilde{p}(x) \) for all \( x \in \{0, 1\}^n \), it is possible to recover the noise-free outcomes of a QC algorithm. The first step is to solve for \( \hat{p}(s) \). With known \( \epsilon_g \), \( \tilde{p}(x) \), and \( x \), a linear system derived from Equation (6) can be built as follows:

\[
G\hat{p} = \tilde{p},
\]

where

\[
G_{ij} := (1 - 2\epsilon_g)^{(j-1)b_i m (i-1)b_j (j-1)} \quad \text{for } i \in \{1, \ldots, 2^n\} \text{ and } j \in \{1, \ldots, 2^n\}
\]

\[
\hat{p} := \begin{pmatrix}
\hat{p}(0 \cdots 00) \\
\hat{p}(0 \cdots 01) \\
\vdots \\
\hat{p}(1 \cdots 11)
\end{pmatrix}, \quad \tilde{p} := \begin{pmatrix}
\tilde{p}(0 \cdots 00) \\
\tilde{p}(0 \cdots 01) \\
\vdots \\
\tilde{p}(1 \cdots 11)
\end{pmatrix},
\]

\( G \in [-1, 1]^{2^n \times 2^n} \), \( \hat{p} \in [-\frac{1}{2^n}, \frac{1}{2^n}]^{2^n} \), and \( \tilde{p} \in [0, 1]^{2^n} \). Using the algorithm to be introduced in Section 3, we can estimate the value of \( \epsilon_g \) to construct matrix \( G \). Using a sufficient number of measurements, we can compute vector \( \hat{p} \). Thus, by solving Equation (7) and substituting the result into Equation (4), we can then re-construct the noise-free distribution function \( p(x) \) for all \( x \in \{0, 1\}^n \). The following lemma implies that the solution of Equation (7) always exists.

**Lemma 2.1.** \( G \) is full-rank for all \( n \geq 1 \).

**Proof.** We decompose \( G \) as

\[
G = G_1^{(n)} \circ G_2^{(n)},
\]

where \( \circ \) is element-wise multiplication, \( G_1^{(n)} \in [0, 1]^{2^n \times 2^n} \), \( G_2^{(n)} \in [-1, 1]^{2^n \times 2^n} \), and

\[
(G_1^{(n)})_{ij} := (1 - 2\epsilon_g)^{(j-1)b_i m (i-1)b_j (j-1)} \quad \text{for } i \in \{1, \ldots, 2^n\} \text{ and } j \in \{1, \ldots, 2^n\}
\]

\[
(G_2^{(n)})_{ij} := (-1)^{(i-1)b_i (j-1)b_j} \quad \text{for } i \in \{1, \ldots, 2^n\} \text{ and } j \in \{1, \ldots, 2^n\}.
\]

We start with \( G_2^{(n)} \) when \( n = 1 \). It is easy to examine that

\[
G_2^{(1)} = \begin{bmatrix}
1 & 1 \\
1 & -1
\end{bmatrix}
\]

is full-rank. Recall that each entry of \( G_2^{(n)} \) is \((-1)^{s \cdot x}\) for binary numbers \( s \) and \( x \), and each row of \( G_2^{(n)} \) shares the same \( x \) while each column shares the same \( s \). Following the little-endian convention, the 16 entries in \( G_2^{(2)} \) can be divided into four divisions equally based on their position:

\[
\begin{array}{c|c|c|c}
\hline
s_2 = 0, x_2 = 0 & s_2 = 0, x_2 = 1 & s_2 = 1, x_2 = 0 & s_2 = 1, x_2 = 1 \\
\hline
\end{array}
\]

ACM Transactions on Quantum Computing, Vol. 4, No. 2, Article 11. Publication date: February 2023.
Namely,

\[
G_2^{(2)} = \begin{bmatrix}
-1 & 0 \\
-1 & 0 \\
-1 & 1 \\
-1 & 1
\end{bmatrix} \begin{bmatrix}
G_2^{(1)} \\
G_2^{(1)} \\
G_2^{(1)} \\
G_2^{(1)}
\end{bmatrix} = \begin{bmatrix}
G_2^{(1)} \\
G_2^{(1)} \\
G_2^{(1)} \\
G_2^{(1)}
\end{bmatrix} \begin{bmatrix}
G_2^{(1)} \\
G_2^{(1)} \\
G_2^{(1)} \\
G_2^{(1)}
\end{bmatrix}.
\]

Similarly, we can have

\[
G_2^{(n)} = \begin{bmatrix}
G_2^{(n-1)} \\
G_2^{(n-1)} \\
G_2^{(n-1)} \\
G_2^{(n-1)}
\end{bmatrix} \begin{bmatrix}
G_2^{(n-1)} \\
G_2^{(n-1)} \\
G_2^{(n-1)} \\
G_2^{(n-1)}
\end{bmatrix}.
\]  (8)

Since \(G_2^{(n)} \in \{-1, 1\}^{2n \times 2n}\), if \(G_2^{(n-1)}\) is full-rank, the structure in Equation (8) implies \(G_2^{(n)}\) is full-rank. As \(G_2^{(1)}\) is also full-rank, by induction, \(G_2^{(n)}\) is full-rank for all \(n \geq 1\).

Note that the \(j\)th column of \(G\) is the \(j\)th column of \(G_2^{(n)}\) multiplied by \((1 - 2\epsilon_j)\)\(^{(j-1)b_m}\) and \(1 - 2\epsilon_j \neq 0\). As all columns of \(G_2^{(n)}\) are linearly independent, their non-zero multiples are linearly independent too. Namely, all columns of \(G\) are linearly independent, so \(G\) is full-rank for all \(n \geq 1\).

However, similar to the problem in Section 2.1, solving Equation (7) cannot guarantee a meaningful \(\hat{\rho}\), that is, a \(\hat{\rho} \in [0, 1]^{2^n}\). Nevertheless, we can consider a optimization problem instead:

\[
\hat{\rho}^* := \arg\min_{\rho} \|G\hat{\rho} - \tilde{\rho}\|_2
\]

s.t.

\[
\sum_{i=1}^{2^n} \sum_{j=1}^{2^n} \hat{\rho}_i (-1)^{(i-1)b_j} (-1)^b_j = 1 \quad (9)
\]

\[
\sum_{i=1}^{2^n} \hat{\rho}_i (-1)^{(i-1)b_j} (-1)^b_j \geq 0 \quad \text{for all } j \in \{1, \ldots, 2^n\}.
\]

As an example, when \(n = 1\), Equation (4) yields

\[
p(0) = \hat{\rho}(0) + \hat{\rho}(1)
\]

\[
p(1) = \hat{\rho}(0) - \hat{\rho}(1),
\]  (10)

so \(\hat{\rho}(0)\) is always \(\frac{1}{2}\) as \(p(0) + p(1) = 1\). Thus, when \(n = 1\), Equation (9) can be simplified as

\[
\hat{\rho}^* := \arg\min_{\tilde{\rho} = \frac{1}{2}, -\frac{1}{2} \leq \tilde{\rho} \leq \frac{1}{2}} \|G\hat{\rho} - \tilde{\rho}\|_2.
\]  (11)

3 ESTIMATING DISTRIBUTIONS OF NOISE PARAMETERS

The bit-flip gate error model in Equation (6) and the measurement error model in Equation (2) together provide us forward models to propagate noise in QC algorithms. Based on these forward models and measurement results from a QC device, we can filter out measurement errors and, in some scenarios, bit-flip gate errors to recover noise-free information. Here, a critical step is to identify model parameters \(\epsilon_g, \epsilon_{m0}\), and \(\epsilon_{m1}\) using repeated measurements of a testing circuit. The Bayesian approach is suited to solving this inverse problem. In this work, we will use both the standard Bayesian inference and a novel Bayesian approach called consistent Bayesian [6] to infer these parameters.
The Bayesian inference considers model parameters conditioned on data $d$ as the posterior distribution $\pi(\lambda|d)$, which is proportional to the product of the prior distribution parameters $\pi(\lambda)$ and the likelihood $\pi(d|\lambda)$, i.e., $\pi(\lambda|d) \propto \pi(\lambda)\pi(d|\lambda)$. It infers the posterior distribution using the stochastic map $d = Q(\lambda) + \epsilon$, where $Q$ is the quantity of interest (QoI) and $\epsilon$ is an assumed error model. In our case, $\lambda$ represents model parameters $\epsilon_g, \epsilon_m$, and $\epsilon_m$; $d$ is the measured data collected from the device; and $\pi(d|\lambda)$ characterizes the difference between forward model output and the data.

Unlike the standard Bayesian inference, the consistent Bayesian directly inverts the observed stochasticity of the data, described as a probability measure or density, using the deterministic map $Q(\lambda)$. This approach also begins with a prior distribution, denoted as $\pi_{\lambda}^{\text{prior}}$, on the model parameters, which is then updated to construct a posterior distribution $\pi_{\lambda}^{\text{post}}(\lambda)$. But its posterior distribution takes a different form:

$$\pi_{\lambda}^{\text{post}}(\lambda) = \pi_{\lambda}^{\text{prior}}(\lambda) \frac{\pi_{\lambda}^{\text{obs}}(Q(\lambda))}{\pi_{\lambda}^{\text{prior}}(Q(\lambda))},$$

where $\lambda \in \Lambda$ and $\mathcal{D}$ is the space of the observed data. Each term in Equation (12) is explained as follows:

- $\pi_{\lambda}^{\text{obs}}$ denotes the push-forward of the prior through the model and represents a forward propagation of uncertainty. It represents how the prior knowledge of likelihoods of parameter values defines a likelihood of model outputs.
- $\pi_{\lambda}^{\text{obs}}$ is the observed probability density of the QoI. It describes the likelihood that the output of the model corresponds to the observed data.

### 3.2 Implementation Details

We take a noisy one-qubit gate $\hat{U}$ (its noise-free version is denoted by $U$) as an example. Suppose we use this gate to build a testing circuit as shown in Figure 3. We set the QoI in our case to be the probability of measuring 0 from the testing circuit. Assume that the measurement operator in the testing circuit is associated with measurement errors $\epsilon_m$ and $\epsilon_m$. Let $\lambda := (\epsilon_g, \epsilon_m, \epsilon_m)$ be the tuple of noise parameters that we want to infer. Note that if $\hat{U}$ is a gate like the Hadamard gate, the bit-flip gate error in theory will not affect the measurement outcome for the testing circuit, which means we only need to infer $\epsilon_m$ and $\epsilon_m$ in this case. In terms of measurement error rate estimation, we provide a choice of testing circuit in Section 4.1 consisting of a single testing circuit for $n$ qubits, which dramatically reduces the number of testing circuits compared with the fully correlated setting. Let $\Lambda := (0, 1) \times (0, 1) \times (0, 1)$ denote the space of noise parameters and $\mathcal{D} := [0, 1]$ denote the space of QoI. Finally, we use $Q : \Lambda \rightarrow \mathcal{D}$ to denote a general function combining Equations (2) and (6) that computes the probability of measuring $|0\rangle$ when the testing circuit has bit-flip gate error and measurement error.

The overall algorithm consists of two parts. In the first part, L number of QoIs, denoted by $q_j$ ($j = 1, \ldots, L$), are generated from $L$ number of prior $\lambda_s$, denoted as $\lambda_j$ for $j = 1, \ldots, L$, with function $Q$. Then, the distribution $\pi_{\lambda}^{\text{prior}}$ is estimated by Gaussian kernel density (KDE) using $q_j$. Next, in the second part, prior $\lambda_s$ are either rejected or accepted based on Equation (12), and those
ALGORITHM 1: Consistent Bayesian inference for error model parameters

Given a set of prior $\lambda_j$ ($j = 1, \ldots, L$), Gaussian KDE $\pi_{Q,D}^{\text{obs}}$ of the observed QoI (i.e., data), model function $Q$ (i.e., combination of Equations (2) and (6), testing circuit, and its input state):

for $j = 1$ to $L$ do
    Use $Q(\lambda_j)$ to compute $q_j$;
end for

Generate Gaussian KDE $\pi_{Q,D}^{(\text{prior})}$ from $q_j$s;

Estimate $\mu := \max_{\lambda \in \Lambda} \frac{\pi_{Q,D}^{\text{obs}}(Q(\lambda))}{\pi_{Q,D}^{(\text{prior})}(Q(\lambda))}$;

for $k = 1$ to $L$ do
    Generate a random number $\zeta_k \in [0, 1]$ from a uniform distribution;
    Compute ratio $\eta_k := \frac{1}{\mu} \cdot \frac{\pi_{Q,D}^{\text{obs}}(Q(\lambda_k))}{\pi_{Q,D}^{(\text{prior})}(Q(\lambda_k))}$;
    if $\eta_k > \zeta_k$ then
        Accept $\lambda_k$;
    else
        Reject $\lambda_k$;
    end if
end for

output Accepted noise parameter $\lambda_k$s.

accepted prior $\lambda_j$s are the posterior noise parameters that we are looking for. The distribution $\pi_{Q,D}^{\text{obs}}$ is the observed probability of measuring $|0\rangle$, i.e., Gaussian KDE of data. The algorithm is summarized in Algorithm 1, which is an implementation of Algorithms 1 and 2 in [6].

In this work, the prior $\lambda_j$s are randomly generated from some relatively flat normal distributions due to the little knowledge of its actual characterization. Thus, for Qubit $i$, suppose we have estimated gate and measurement error rates $(\epsilon_{g,i}^0, \epsilon_{m0,i}^0, \epsilon_{m1,i}^0)$ from past experience and their variances $(\sigma_{\epsilon_{g,i}}, \sigma_{\epsilon_{m0,i}}, \sigma_{\epsilon_{m1,i}})$ that make curves flat; the prior distributions are

$\epsilon_{m0,i} \sim N(0, \sigma_{\epsilon_{m0,i}}^2)$,
$\epsilon_{m1,i} \sim N(0, \sigma_{\epsilon_{m1,i}}^2)$,
$\epsilon_{g,i} \sim N(0, \sigma_{\epsilon_{g,i}}^2)$.

In this setting, the acceptance rates of all experiments in Section 4 range from 10% to 35%. This is high enough to select a sufficient number of posterior parameters in this study.

To demonstrate and compare the difference between the results of consistent and standard Bayesian algorithms, we also use the same priors and observation datasets to infer noise parameters via the standard Bayesian. For a single Qubit $i$, let $(x_j, y_j)$ for $j = 1, \ldots, J$ represent $J$ number of data pairs, where $x_j$ is the theoretical probability of measuring $|0\rangle$ and $y_j$ is the observed probability of measuring $|0\rangle$. As discussed in Equations (1) and (10), we have

$y_j = ((0.5 + (1 - 2\epsilon_{g,i})(x_j - 0.5))(1 - \epsilon_{m0,i}) + (0.5 - (1 - 2\epsilon_{g,i})(x_j - 0.5))\epsilon_{m1,i} + \epsilon_j, \quad (13)$

where $m$ is the number of repetitions of the gate in the testing circuit ($m = 1$ in Figure 3) and $\epsilon_j \sim N(0, \sigma_{\epsilon,j}^2)$ represents noise in general with standard deviation $\sigma_{\epsilon} \geq 0$. We use Cauchy(0, 1) as
Algorithm 2: Standard Bayesian inference for error model parameters

Data.
Number of repetitions of gates in the testing circuit $m$, theoretical probabilities of measuring $|0\rangle$ $x_j$, observed probabilities of measuring $|0\rangle$ $y_j$ ($j = 1, \ldots, f$), prior mean of noise parameters $(\epsilon_{m0,i}, \epsilon_{m1,i}, \epsilon_{g,i})$, and prior variance of noise parameters $(\sigma_{g,i}, \sigma_{m0,i}, \sigma_{m1,i})$.

Model Parameters.
posterior $(\epsilon_{m0,i}, \epsilon_{m1,i}, \epsilon_{g,i}) \in (0, 1)^3$ and $\sigma_e \geq 0$.

Prior Distributions.

- $\sigma_e \sim$ Cauchy(0, 1);
- $\epsilon_{m0,i} \sim N(\epsilon_{m0,i}^0, \sigma_{m0,i}^2)$;
- $\epsilon_{m1,i} \sim N(\epsilon_{m1,i}^0, \sigma_{m1,i}^2)$;
- $\epsilon_{g,i} \sim N(\epsilon_{g,i}^0, \sigma_{g,i}^2)$;

Likelihood Function.

Only measurement errors:

$\forall j, y_j \sim N(x_j(1 - \epsilon_{m0}) + (1 - x_j)\epsilon_{m1}, \sigma_e^2)$.

Gate and measurement errors:

$\forall j, y_j \sim N(((0.5 + (1 - 2\epsilon_{g})^m(x_j - 0.5))(1 - \epsilon_{m0}) + (0.5 - (1 - 2\epsilon_{g})^m(x_j - 0.5))\epsilon_{m1}, \sigma_e^2)$.

Stan parameters.

Default No-U-Turn Sampler, 10,000 iterations, 2,000 warm-up iterations, adapt_delta = 0.99, and other parameters are default.

the prior distribution of $\sigma_e$. Equation (13) yields the following likelihood function:

$$f(y|x, \epsilon_{m0,i}, \epsilon_{m1,i}, \epsilon_{g,i}, \sigma_e) = \prod_{j=1}^{f} f_j(y_j|x_j, \epsilon_{m0,i}, \epsilon_{m1,i}, \epsilon_{g,i}, \sigma_e),$$

where each $f_j$ is the probability density function (PDF)

$$N(((0.5 + (1 - 2\epsilon_{g,i})^m(x_j - 0.5))(1 - \epsilon_{m0,i}) + (0.5 - (1 - 2\epsilon_{g,i})^m(x_j - 0.5))\epsilon_{m1,i}, \sigma_e^2).$$

In this work, we use the RStan package in R [24, 27] to implement the standard Bayesian inference, which is summarized in Algorithm 2.

4 EXPERIMENTS

Because using our bit-flip error model only is not sufficient for analyzing gate errors in a complicated algorithm like Grover’s search or QAOA, the inference for gate errors is performed for a few prototype circuits. For more sophisticated algorithms, we only investigate the measurement error. All experiments are conducted on IBM’s 5-qubit quantum computer ibmqx2. We compare both the consistent Bayesian (Algorithm 1) and the standard Bayesian method (Algorithm 2) with the measurement error filter in Qiskit CompleteMeasFitter [2] and the method in [19] based on quantum detector tomography (QDT) to demonstrate the efficiency of our approaches.

4.1 Measurement Errors Filtering Experiment

4.1.1 Construction of Error Filter. We use the circuit in Figure 4 to infer measurement error parameters in every single qubit, i.e., $\epsilon_{m0,i}, \epsilon_{m1,i}$ for $i \in \{1, 2, 3, 4\}$ on ibmqx2. Here, $H$ is the Hadamard gate for each qubit. Theoretically, the observed results of $H |0\rangle$ are invariant under bit-flip and phase-flip errors. Consequently, in this case, only the measurement error affects the distribution of measurement outputs, and we do not infer gate error rate $\epsilon_g$. The testing circuit
is executed \(1024 \times 128\) times, where the fraction of measuring 0 in each ensemble consisting of 1,024 runs provides estimated probability of measuring 0 from the testing circuit. Thus, we have 128 data points in total, i.e., \(L = 128\) in Algorithm 1 or \(J = 128\) in Algorithm 2. For qubit \(i\), the prior \((\epsilon_{m0,i}, \epsilon_{m1,i}) \subseteq (0,1) \times (0,1)\) are a random number from truncated normal distribution \(N(\epsilon_{m0,i}, 0.1^2)\) and \(N(\epsilon_{m1,i}, 0.1^2)\), respectively, where \(\epsilon_{m0,i}\) and \(\epsilon_{m1,i}\) are corresponding values provided by IBM in Qiskit API IBMQbackend.properties() after the daily calibration. Then, we use Algorithm 1 to generate the posterior distributions. We note that in this test, the results by the consistent Bayesian is very close to the standard Bayesian, so we present the former only.

Figure 5 displays the joint and marginal distribution of posterior distributions of error model parameters for qubits 1–4 using the consistent Bayesian approach. Using these posterior distributions of error model parameters, we can compute the posterior distribution of the QoI by substituting samples of these distributions in the forward model \(Q\). Figure 6 shows that these posteriors of the QoI, denoted as \(\pi_{Q}^{\text{post}}\), approximate the distribution of the observed data \(\pi_{Q}^{\text{obs}}\) very well.

For a more quantitative comparison, we list the posterior mean and the maximum a posteriori probability (MAP) in Table 1. We can see, in general, that \(\epsilon_{m1,i}\) is higher than \(\epsilon_{m0,i}\), which is consistent with the description in [14]. Also, Table 1 presents the Kullback–Leibler (KL) divergence between PDFs of the observed data \(\pi_{Q}^{\text{obs}}\) and the posterior distribution of the QoI \(\pi_{Q}^{\text{post}}\) for each qubit in Figure 6, which illustrates the accuracy of our error model.

In this test, our prior distributions \(N(\epsilon_{m0,i}, 0.1^2)\) and \(N(\epsilon_{m1,i}, 0.1^2)\) are quite flat and not informative. This is because the vendor-provided \(\epsilon_{m0,i}\) and \(\epsilon_{m1,i}\) are not always good estimations. This can be verified by the error mitigation results. When we use relation Equations (2) and (3) to construct measurement error filters using the vendor-provided \((\epsilon_{m0,i}, \epsilon_{m1,i})\) and our posteriors, then apply those filters on the 128 outputs of the circuit in Figure 4, we obtain different results as shown in Figure 7. The theoretical probability of measuring 0 for the circuit in Figure 4 is 0.5, but the provided parameters rarely give this value, and its mean and peak of Gaussian KDE are not even close to 0.5. On the other hand, the filters created by our posteriors can make sure the mean and peak of the denoised probability of measuring \(|0\rangle\) are around the ideal value 0.5. The results in Figure 4 indicate that when applying Equation (3) to mitigate the measurement error, one has a larger chance to obtain a denoised QoI close to the ideal value 0.5 by using the parameters inferred by our method. More importantly, the result by our method is unbiased as the mean value of the denoised QoI is 0.5.

This test indicates that we can use the circuit shown in Figure 4 to estimate the measurement error in multiple qubits at the same time. It only requires preparing the initial state using ground states, and the total number of gates is linearly dependent on the number of qubits.  

\[
\begin{align*}
\left|0\right>^\otimes n
\end{align*}
\]

Fig. 4. Testing circuit for measurement error parameter inference.

| Qubit 1 | Qubit 2 | Qubit 3 | Qubit 4 |
|---------|---------|---------|---------|
| KL-div(\(\pi_{Q}^{\text{post}}\), \(\pi_{Q}^{\text{obs}}\)) | 0.001014 | 0.002243 | 0.000777 | 0.001610 |
| Post. Mean \((1 - \epsilon_{m0,i}, 1 - \epsilon_{m1,i})\) | (0.9354, 0.9009) | (0.9537, 0.8184) | (0.9457, 0.8976) | (0.8272, 0.9492) |
| Post. MAP \((1 - \epsilon_{m0,i}, 1 - \epsilon_{m1,i})\) | (0.9797, 0.9128) | (0.9863, 0.8243) | (0.9858, 0.9180) | (0.8426, 0.9846) |
Fig. 5. Joint and marginal posterior distributions of measurement error parameters in the testing circuit shown in Figure 4. Here, \((1 - \epsilon_{m0,i}, 1 - \epsilon_{m1,i})\) are shown for demonstration purposes.

4.1.2 Application on State Tomography. After obtaining the error model parameters, we can further use this model to mitigate the measurement errors in other circuits. We first apply error filters to the results of state tomography on circuits that make bell basis from \(|00\rangle\) and \(|000\rangle\). Qubits 0 and 1 are used for the 2-qubit state and Qubits 0 to 2 are used for the 3-qubit state in ibmqx2. The fidelity between density matrices from the (corrected) state tomography result and theoretical quantum state is listed in Table 2. For the 2-qubit state tomography, filters constructed from posterior means by the consistent Bayesian and the standard Bayesian provide similar fidelity as that by the Qiskit filter. However, for 3-qubit tomography, filters from both Bayesian methods yield better fidelity, and their performances are similar. We note that the Qiskit filter assumes correlation in the measurements, which requires more model parameters, while our model does not. The fidelity in Table 2 indicates that Bayesian methods enable us to use fewer parameters to obtain better results.
Fig. 6. PDF of the QoI (i.e., the probability of measuring $|0\rangle$) obtained from data ($\pi_{D}^{\text{obs}}$) and from evaluating $Q$ using inferred measurement error parameters ($\pi_{D}^{\text{post}}$).

Table 2. Fidelity of State Tomography Results Filtered by Various Error Filters

| State | Fidelity | Raw Data | By Qiskit Method | By Cons. Mean | By Stand. Mean |
|-------|----------|----------|-----------------|---------------|---------------|
| $\sqrt{2}$($|00\rangle + |11\rangle$) | 0.9051   | 0.9800   | 0.9781          | 0.9783        |
| $\sqrt{2}$($|01\rangle + |10\rangle$) | 0.9157   | 0.9803   | 0.9806          | 0.9808        |
| $\sqrt{2}$($|000\rangle + |111\rangle$) | 0.7389   | 0.9227   | 0.9390          | 0.9391        |
| $\sqrt{2}$($|100\rangle + |011\rangle$) | 0.7006   | 0.9121   | 0.9254          | 0.9207        |
| $\sqrt{2}$($|110\rangle + |001\rangle$) | 0.6974   | 0.8863   | 0.9443          | 0.9446        |

*a“Qiskit Method” means to CompleteMeasFitter in Qiskit [2]. “Cons. Mean” implies the transition matrix is created from posterior mean by Algorithm 1. “Stand. Mean” means the transition matrix is created from posterior mean by standard Bayesian.

4.1.3 Application on Grover’s Search and QAOA. Next, we apply our filter on Grover’s search and QAOA circuits from [15]. We measure Qubits 1 and 2 in ibmqx2 for Grover’s search circuit. The exact solution of this Grover’s search example is $|11\rangle$ and the theoretical probability is 1. Thus, in this case, we compare the probability of measuring $|11\rangle$ by running in the real device ibmqx2 and the denoised probabilities from error filters based on the Qiskit method CompleteMeasFitter, QDT in [19], mean and MAP of posteriors from standard Bayesian, and mean and MAP of posteriors from the consistent Bayesian. All circuits for the Qiskit filter and QDT filters are executed.
Fig. 7. PDFs of the probability of measuring $|0\rangle$ denoised by vendor-provided parameters (priors) and by posteriors.

Table 3. Probability of Measuring $|11\rangle$ in Grover’s Search Example

| Method/Source | Hour 0 | Hour 2 | Hour 4 | Hour 8 | Hour 12 | Hour 16 |
|---------------|--------|--------|--------|--------|---------|---------|
| Raw Data      | 0.6727 | 0.6930 | 0.6724 | 0.6740 | 0.6917  | 0.6841  |
| Qiskit Method | 0.7097 | 0.7335 | 0.7104 | 0.7120 | 0.7323  | 0.7241  |
| QDT           | 0.7107 | 0.7332 | 0.7087 | 0.7108 | 0.7305  | 0.7224  |
| Stand. Mean   | 0.9099 | 0.9324 | 0.9063 | 0.9088 | 0.9290  | 0.9192  |
| Stand. MAP    | 0.8378 | 0.8635 | 0.8372 | 0.8392 | 0.8616  | 0.8522  |
| Cons. Mean    | 0.9128 | 0.9351 | 0.9088 | 0.9114 | 0.9316  | 0.9219  |
| Cons. MAP     | 0.8920 | 0.9158 | 0.8914 | 0.8936 | 0.9128  | 0.9034  |

$^a$“Qiskit Method” means to CompleteMeasFitter in Qiskit [2]. QDT refers to filter in [19], “Stand.” stands for Standard Bayesian, and “Cons.” refers to Algorithm 1. MAP and mean represent the error filters being created from the MAP and mean of posteriors.

$^b$“Hour X” means the experiment is conducted X hours after the data for error filters of all listed methods are collected.

for 8,192 shots, and each probability used in both filters is estimated from 8,192 measurement outcomes.

In addition, as we do not expect that the quantum computer has a stable environment, in order to see the robustness of each method in comparison, after the data for creating error filters are collected, we run our Grover’s search circuit at several different times and then apply the same set of filters. All results are listed in Table 3.

As shown in Table 3, both Bayesian methods yield best performance among all the methods, while the filters constructed from the posterior mean are better than the filters constructed from...
A Bayesian Approach for Characterizing and Mitigating Gate and Measurement Errors

Fig. 8. The graph of QAOA example in [15].

Table 4. Average Size of a Sampled Cut in QAOA Example

| Method/Source | Hour 0 | Hour 2 | Hour 4 | Hour 8 | Hour 12 | Hour 16 |
|---------------|--------|--------|--------|--------|---------|---------|
| Simulator     | 2.8637 | 2.8642 | 2.8651 | 2.8652 | 2.8642  | 2.8626  |
| Raw Data      | 2.3005 | 2.3579 | 2.3197 | 2.2823 | 2.3063  | 2.2871  |
| Qiskit Method | 2.3783 | 2.4623 | 2.4247 | 2.3786 | 2.4063  | 2.3926  |
| QDT           | 2.3812 | 2.4453 | 2.4016 | 2.3589 | 2.3851  | 2.3612  |
| Stand. Mean   | 2.4878 | 2.5483 | 2.5059 | 2.4551 | 2.4860  | 2.4581  |
| Stand. MAP    | 2.4080 | 2.4708 | 2.4222 | 2.3800 | 2.4059  | 2.3829  |
| Cons. Mean    | 2.4911 | 2.5518 | 2.5089 | 2.4578 | 2.4891  | 2.4612  |
| Cons. MAP     | 2.4407 | 2.4996 | 2.4554 | 2.4109 | 2.4382  | 2.4133  |

The graph of the QAOA example in [15] is shown in Figure 8, which has maximum objective value 3 in the Max-Cut problem and 6 bit-string optimal solution $|0010\rangle$, $|0101\rangle$, $|0110\rangle$, $|1001\rangle$, $|1010\rangle$, $|1101\rangle$ (ibmqx2 uses the little-endien convention, so the rightmost bit is Node 1 and the leftmost bit is Node 4). Because the graph in Figure 8 is a subgraph of the coupling map of ibmqx2, we map the nodes to qubits exactly.

The conclusion from Tables 4 and 5 is basically the same as that from Table 3. Namely, Bayesian methods, especially filters from the posterior mean, outperform other methods, and parameters inferred by the consistent Bayesian work slightly better than those by the standard Bayesian in all six time slots. From both Grover’s search and QAOA examples, we can see the accuracy of both Bayesian approaches is better than the existing methods.

4.1.4 Application on Random Clifford Circuits. Finally, we test the measurement error filtering for random 2-Qubit Clifford circuits with one, two, three, and four 2-Qubit Clifford operators (i.e., length 1, 2, 3, 4). For each length, 16 random circuits are generated to draw a boxplot and each circuit is run for 8,192 shots. The results are shown in Figure 9. While the theoretical output of
Table 5. Probability of Measuring an Optimal Solution in QAOA Example

| Method/Source | Hour 0 | Hour 2 | Hour 4 | Hour 8 | Hour 12 | Hour 16 |
|---------------|--------|--------|--------|--------|---------|---------|
| Simulator     | 0.8930 | 0.8937 | 0.8941 | 0.8943 | 0.8940  | 0.8940  |
| Raw Data      | 0.5784 | 0.6038 | 0.5895 | 0.5725 | 0.5748  | 0.5740  |
| Qiskit Method | 0.5968 | 0.6456 | 0.6316 | 0.6074 | 0.6140  | 0.6155  |
| QDT           | 0.6400 | 0.6698 | 0.6525 | 0.6312 | 0.6331  | 0.6325  |
| Stand. Mean   | 0.6952 | 0.7239 | 0.7033 | 0.6766 | 0.6787  | 0.6797  |
| Stand. MAP    | 0.6444 | 0.6695 | 0.6508 | 0.6305 | 0.6309  | 0.6317  |
| Cons. Mean    | 0.6975 | 0.7265 | 0.7058 | 0.6790 | 0.6810  | 0.6822  |
| Cons. MAP     | 0.6610 | 0.6860 | 0.6672 | 0.6431 | 0.6439  | 0.6452  |

Fig. 9. Measurement error filtering for random 2-Qubit Clifford circuits. “Length” represents the number of Clifford operators in the circuit. “Probability” means the probability of measuring $|00\rangle$.

the 2-Qubit Clifford circuit is $|00\rangle$ with probability 1, Figure 9 demonstrates that the filter constructed from the posterior mean estimated by standard Bayesian provides the best performance. The consistent Bayesian results in almost the same results as the standard Bayesian method.

4.2 Gate and Measurement Error Filtering Experiment

We consider the circuit with 200 NOT gates as shown in Figure 10. We still use machine ibmqx2 and run the experiment twice separately on Qubit 1 and Qubit 2. In each trial, the circuit is executed $1024 \times 128$ times where readouts from every 1,024 runs are used to estimate the QoI, i.e., the probability of measuring $|0\rangle$. Namely, we collect 128 samples of the QoI.

Because the aforementioned Qiskit method CompleteMeasFitter and QDT are for measurement errors, in this section, we only compare the results from standard Bayesian and the consistent Bayesian with the same priors and dataset. The priors are truncated normal $N(e_{m0,1}^0, 0.1^2)$, $N(e_{m1,1}^0, 0.1^2)$, and $N(e_{g,1}^0, 0.005^2)$ with range $(0, 1)$. Of note, $e_{m0,1}^0, e_{m1,1}^0, e_{g,1}^0$ are vendor-provided

ACM Transactions on Quantum Computing, Vol. 4, No. 2, Article 11. Publication date: February 2023.
4.2.1 Inference for Noise Parameters. Figure 11 shows the distribution of $\epsilon_g$ in Qubits 1 and 2. Both distributions are right-skewed. Table 6 provides the numerical values for mean and MAP. In Table 6, we can see that both methods give similar measurement error parameters $\epsilon_{m0,i}$ and $\epsilon_{m1,i}$ on Qubits 1 and 2, but the gate error rate $\epsilon_g$ is not always similar. More importantly, as shown in Figure 12, posteriors of the QoI from the consistent Bayesian, i.e., $\pi_D^{Q(post)}$, match the distribution of data, i.e., $\pi_{obs}$, quite well. On the other hand, the posterior distribution of the QoI generated by posterior distributions of model parameters from the standard Bayesian can match the empirical mean of the data only while the shape of the PDF is quite different.

4.2.2 Error Filtering. Using the posterior means from Table 6, we construct gate and measurement error filters and apply them on the 128 samples of the QoI (i.e., probabilities of measuring 0) values from IBM’s daily calibration, where the prior measure error rates $\epsilon_{m0,i}^0$ and $\epsilon_{m1,i}^0$ are often at the scale of $10^{-2}$ to $10^{-1}$, and the prior single-qubit gate error rate $\epsilon_{g,i}^0$ is usually between $10^{-4}$ and $10^{-3}$. Therefore, we adjust the standard deviations to match the scale of prior means. Consequently, the prior distributions are relatively flat due to the lack of knowledge on these parameters.
Table 6. Measurement Error Parameters of the Consistent and Standard Bayesian Inference

| Method                      | Qubit 1                  | Qubit 2                  |
|-----------------------------|--------------------------|--------------------------|
| Consistent Post. Mean       | (0.9255, 0.8922, 0.004934) | (0.9229, 0.8856, 0.003804) |
| Consistent Post. MAP        | (0.9756, 0.8837, 0.004827) | (0.9770, 0.9485, 0.003291) |
| Standard Post. Mean         | (0.9221, 0.8939, 0.004683) | (0.9214, 0.8871, 0.002982) |
| Standard Post. MAP          | (0.9758, 0.8835, 0.006550) | (0.9836, 0.9354, 0.003453) |

Table 7. Probability of Measuring $|11\rangle$ in Grover’s Search Example Denoised by Parameters in Section 4.2

| Method       | Hour 0 | Hour 2 | Hour 4 | Hour 8 | Hour 12 | Hour 16 |
|--------------|--------|--------|--------|--------|---------|---------|
| Stand. Mean  | 0.8398 | 0.8680 | 0.8392 | 0.8414 | 0.8656  | 0.8561  |
| Stand. MAP   | 0.8116 | 0.8367 | 0.8110 | 0.8131 | 0.8348  | 0.8257  |
| Cons. Mean   | 0.8434 | 0.8716 | 0.8428 | 0.8450 | 0.8691  | 0.8595  |
| Cons. MAP    | 0.7992 | 0.8240 | 0.7986 | 0.8005 | 0.8221  | 0.8131  |

The data for error filters in Section 4.1 and experiment in Section 4.2 were collected within 1 hour, so it is reasonable to use posteriors in Section 4.2 to denoise Grover’s search data in Section 4.1. However, comparing the values of measurement error parameters in Table 1 and in Table 6, we can see that there are some noticeable differences. Table 7 provides the results of using parameters in Section 4.2 to filter out errors in data used in Section 4.1. We can see that they are better than the Qiskit method and QDT but worse than values from either Bayesian methods shown in Table 3.

One possible explanation is that with 200 gates, our model is much more sensitive to the gate error than the measurement error. A comparison of the sensitivity is shown in Figure 14, where the results are obtained by using the error models Equations (2) and (6). In Figures 14(a) and 14(b), we can see that when $\epsilon_g$ is fixed, the QoI changes linearly and slowly as $\epsilon_{m0,i}$ or $\epsilon_{m1,i}$ varies. However, as shown in Figure 14(c), when $\epsilon_{m0,i}$ and $\epsilon_{m1,i}$ are fixed, the QoI changes rapidly as $\epsilon_g$ increases. The estimation of measurement error in Section 4.1 uses circuits that have 0.5 chance to measure either $|0\rangle$ or $|1\rangle$ without noise, and this distribution does not change when a Hadamard gate suffers from bit-flip or phase-flip error, so it yields better performance.

Fig. 13. Denoised (both gate and measurement) probability of measuring 0. Parameters used are posterior mean from Table 6.
5 DISCUSSION AND FUTURE WORKS

In this work, we extend a bit-flip error model from a single-gate case to a multiple-gate case and provide theoretical analysis to prove the existence of the error mitigation solution for both cases. In some noise models, such as the depolarizing error model, the rate of bit-flip error is associated with the rates of other types of errors [21, p. 379]. Thus, the inference of bit-flip error rates could provide a connection to a more general noise model. We propose to use Bayesian approaches to infer parameters in the error models to characterize the propagation of the device noise in QC algorithms more effectively. The experiments in Section 4 demonstrate that our methodology outperforms two existing methods on the same error models over a wide range of time, while the number of testing circuits is linear or constant to the number of qubits. The consistent Bayesian approach is, in general, better than the standard Bayesian. These results indicate that our error models can characterize the device noise quite well, and they help to understand the propagation of such noise in QC algorithms.

There are still several limitations in our methodology. One issue that affects the scalability of our method is the exponentially large matrix in the denoising step. The dimension of matrices can be reduced if we can identify the qubits that are independent during the measurement step of an algorithm and filter their measurement outcome separately. A recent work in [20] also indicates a scheme to reduce the dimension of the transition matrix by limiting the range of bases that are put into consideration. Also, a parallel algorithm proposed in [5] can exploit the tensor-product structure of the linear system in the error filtering step to speed up the calculation. On the other hand, because the method of estimating the distribution of model parameters is not limited by the two models we discussed in the article, consideration for a pairwise-correlated measurement error model discussed in [4] will probably be helpful for inferring correlated measurement error rates. As for the gate error model, its applicable gate and error types are limited. A potential extension to the applicable gates is to modify the model to accommodate multi-qubit bit-flip error instead of individual-qubit error. This is because more gates commute with $X \otimes n$ than with elements in $\{X, I\} \otimes n$ for $n \geq 2$. For example, $X \otimes X$ commute with matrix $A \otimes B$ and $e^{-i\delta A \otimes B}$ for $(A, B) \in \{Y, Z\} \otimes 2 \cup \{I, X\} \otimes 2$ and arbitrary $\delta$, where the form $e^{-i\delta A \otimes B}$ is generally utilized in quantum simulation [28].

REFERENCES

[1] Dorit Aharonov and Michael Ben-Or. 2008. Fault-tolerant quantum computation with constant error rate. *SIAM J. Comput.* 38, 4 (Jan. 2008), 1207–1282. https://doi.org/10.1137/s0097539799359385

[2] Gadi Aleksandrowicz, Thomas Alexander, Panagiotis Barkoutsos, Luciano Bello, Yael Ben-Haim, David Bucher, Francisco Jose Cabrera-Hernández, Jorge Carballo-Franquis, Adrian Chen, Chun-Fu Chen, Jerry M. Chow, Antonio D. Córcoles-Gonzales, Abigail J. Cross, Andrew Cross, Juan Cruz-Benito, Chris Culver, Salvador De La Puente González, Enrique De La Torre, Delton Ding, Eugene Dumitrescu, Ivan Duran, Pieter Eendebak, Mark Everitt, Ismael Faro Sertage, Albert Frisch, Andreas Führer, Jay Gambetta, Borja Godoy Gago, Juan Gomez-Mosquera, Donny Greenberg, Ikko Hamamura, Vojtech Havlicek, Joe Hellmers, Lukasz Herok, Hiroshi Horii, Shaohan Hu, Takashi
Arne G. Grimsmo, Joshua Combes, and Ben Q. Baragiola. 2020. Quantum computing with rotation-symmetric bosonic codes. Phys. Rev. X 10, 1 (Mar. 2020), 011058. https://doi.org/10.1103/PhysRevX.10.011058
A Bayesian Approach for Characterizing and Mitigating Gate and Measurement Errors

[14] Rebecca Hicks, Christian W. Bauer, and Benjamin Nachman. 2021. Readout rebalancing for near-term quantum computers. *Phys. Rev. A* 103, 2 (Feb. 2021), 022407. https://doi.org/10.1103/PhysRevA.103.022407

[15] Abhijith J., Adetokunbo Adedoyin, John Ambrosiano, Petr Anisimov, William Casper, Gopinath Chennupati, Carleton Coffrin, Hristo Djidjev, David Gunter, Satish Karra, Nathan Lemons, Shizeng Lin, Alexander Malzychenkov, David Masecarenas, Susan Mniszewski, Balu Nadiga, Daniel O’Malley, Diane Oyen, Scott Pakin, Lakshman Prasad, Randy Roberts, Phillip Romero, Nandakishore Santhi, Nikolai Sinitsyn, Pieter J. Swart, James G. Wendelberger, Boram Yoon, Richard Zamora, Wei Zhu, Stephan Eidenebenz, Andreas Bärtschi, Patrick J. Coles, Marc Vuffray, and Andrey Y. Lokhov. 2022. Quantum algorithm implementations for beginners. *ACM Transactions on Quantum Computing* 3, 4 (Dec. 2022), 1–92. https://doi.org/10.1145/3517340

[16] Atharv Joshi, Kyungjoo Noh, and Yvonne Y. Gao. 2021. Quantum information processing with bosonic qubits in circuit QED. *Quantum Science and Technology* 6, 3 (2021), 033001.

[17] Abhinav Kandala, Kristan Temme, Antonio D. Córcoles, Antonio Mezzacapo, Jerry M. Chow, and Jay M. Gambetta. 2019. Error mitigation extends the computational reach of a noisy quantum processor. *Nature* 567, 7749 (2019), 491–495. https://doi.org/10.1038/s41586-019-1040-7

[18] Aleksander Marek Kubica. 2018. *The ABCs of the Color Code: A Study of Topological Quantum Codes as Toy Models for Fault-tolerant Quantum Computation and Quantum Phases of Matter*. Ph.D. Dissertation. California Institute of Technology. https://doi.org/10.7907/059V-MG69

[19] Filip B. Maciejewski, Zoltán Zimborás, and Michał Oszmaniec. 2020. Mitigation of readout noise in near-term quantum devices by classical post-processing based on detector tomography. *Quantum* 4 (Apr. 2020), 257. https://doi.org/10.22331/q-2020-04-24-257

[20] Paul D. Nation, Hwajung Kang, Neereja Sundaresan, and Jay M. Gambetta. 2021. Scalable mitigation of measurement errors on quantum computers. *PRX Quantum* 2, 4 (Nov. 2021), 040326. https://doi.org/10.1103/PRXQuantum.2.040326

[21] Michael A. Nielsen and Isaac L. Chuang. 2010. *Quantum Computation and Quantum Information*. Cambridge University Press, Cambridge. https://doi.org/10.1017/CBO9780511976667

[22] Ryan O’Donnell. 2014. *Analysis of Boolean Functions*. Cambridge University Press, New York. https://doi.org/10.1017/CBO97811391841782

[23] John Preskill. 2018. Quantum computing in the NISQ era and beyond. *Quantum* 2 (Aug. 2018), 79. https://doi.org/10.22331/q-2018-08-06-79

[24] R Core Team. 2020. RStan: The R Interface to Stan. http://mc-stan.org/. R package version 2.21.1.

[25] Francesco Tacchino, Alessandro Chiesa, Stefano Carretta, and Dario Gerace. 2019. Quantum computers as universal quantum simulators: State-of-the-art and perspectives. *Advanced Quantum Technologies* 3, 3 (Dec. 2019), 1900052. https://doi.org/10.1002/quito.201900052

[26] Yasuhiro Takahashi, Yuki Takeuchi, and Seichiro Tani. 2020. Classically simulating quantum circuits with local depolarizing noise. In *45th International Symposium on Mathematical Foundations of Computer Science (MFCS’20)* (Leibniz International Proceedings in Informatics (LIPIcs), Vol. 170), Javier Esparza and Daniel Král’ (Eds.). Schloss Dagstuhl–Leibniz-Zentrum für Informatik, Dagstuhl, Germany, 83:1–83:13. https://doi.org/10.4230/LIPIcs.MFCS.2020.83

[27] Kristan Temme, Sergey Bravyi, and Jay M. Gambetta. 2017. Error mitigation for short-depth quantum circuits. *Phys. Rev. Lett.* 119, 18 (Nov. 2017), 180509. https://doi.org/10.1103/PhysRevLett.119.180509

[28] Joel J. Wallman and Joseph Emerson. 2016. Noise tailoring for scalable quantum computation via randomized compiling. *Phys. Rev. A* 94, 5 (Nov. 2016), 052325. https://doi.org/10.1103/PhysRevA.94.052325

[29] Muqing Zheng. 2022. QCOL-LU/Bayesian-Error-Characterization-and-Mitigation. Lehigh University Quantum Computing and Optimization Lab. https://github.com/QCOL-LU/Bayesian-Error-Characterization-and-Mitigation.

Received 15 November 2021; revised 31 August 2022; accepted 1 September 2022

ACM Transactions on Quantum Computing, Vol. 4, No. 2, Article 11. Publication date: February 2023.