

The Accuracy vs. Sampling Overhead Trade-off in Quantum Error Mitigation Using Monte Carlo-Based Channel Inversion

Yifeng Xiong, *Student Member, IEEE*, Soon Xin Ng, *Senior Member, IEEE*, and Lajos Hanzo, *Fellow, IEEE*

Abstract—Quantum error mitigation (QEM) is a class of promising techniques for reducing the computational error of variational quantum algorithms. In general, the computational error reduction comes at the cost of a sampling overhead due to the variance-boosting effect caused by the channel inversion operation, which ultimately limits the applicability of QEM. Existing sampling overhead analysis of QEM typically assumes exact channel inversion, which is unrealistic in practical scenarios. In this treatise, we consider a practical channel inversion strategy based on Monte Carlo sampling, which introduces additional computational error that in turn may be eliminated at the cost of an extra sampling overhead. In particular, we show that when the computational error is small compared to the dynamic range of the error-free results, it scales with the square root of the number of gates. By contrast, the error exhibits a linear scaling with the number of gates in the absence of QEM under the same assumptions. Hence, the error scaling of QEM remains to be preferable even without the extra sampling overhead. Our analytical results are accompanied by numerical examples.

NOTATIONS

- Deterministic scalars, vectors and matrices are represented by x , \mathbf{x} , and \mathbf{X} , respectively, whereas their random counterparts are denoted as \mathbf{x} , \mathbf{x} , and \mathbf{X} , respectively. Deterministic sets, random sets, and operators are denoted as \mathcal{X} , \mathbf{X} , and \mathcal{X} , respectively.
- The notations $\mathbf{1}_n$, $\mathbf{0}_n$, $\mathbf{0}_{m \times n}$, and \mathbf{I}_k , represent the n -dimensional all-one vector, the n -dimensional all-zero vector, the $m \times n$ dimensional all-zero matrix, and the $k \times k$ identity matrix, respectively. The subscripts may be omitted if they are clear from the context.
- The notation $\|\mathbf{x}\|_p$ represents the ℓ_p -norm of vector \mathbf{x} , and the subscript may be omitted when $p = 2$. For matrices, $\|\mathbf{A}\|_p$ denotes the matrix norm induced by the corresponding ℓ_p vector norm. The notation $1/\mathbf{x}$ represents the element-wise reciprocal of vector \mathbf{x} .
- The notation $[\mathbf{A}]_{i,j}$ denotes the (i,j) -th entry of matrix \mathbf{A} . For a vector \mathbf{x} , $[\mathbf{x}]_i$ denotes its i -th element. The submatrix obtained by extracting the i_1 -th to i_2 -th rows and the j_1 -th to j_2 -th columns from \mathbf{A} is denoted as $[\mathbf{A}]_{i_1:i_2, j_1:j_2}$. The notation $[\mathbf{A}]_{:,i}$ represents the i -th column of \mathbf{A} , and $[\mathbf{A}]_{i,:}$ denotes the i -th row, respectively.
- The notation $\mathbf{A} \otimes \mathbf{B}$ represents the Kronecker product between matrices \mathbf{A} and \mathbf{B} . The notation $\mathbf{A} \odot \mathbf{B}$ denotes the Hadamard product between matrices \mathbf{A} and \mathbf{B} .

Authors are with School of Electronics and Computer Science, University of Southampton, SO17 1BJ, Southampton (UK).

L. Hanzo would like to acknowledge the financial support of the Engineering and Physical Sciences Research Council projects EP/P034284/1 and EP/P003990/1 (COALESCE) as well as of the European Research Council's Advanced Fellow Grant QuantCom (Grant No. 789028). This work is also supported in part by China Scholarship Council (CSC).

I. INTRODUCTION

NOISY intermediate scale quantum computers [1] has been one of the most impressive recent advances in the area of quantum computing. In particular, a quantum computer consisting of 53 quantum bits (qubits) has been built in 2019, and has been shown being capable of performing computational tasks that are challenging to be carried out by state-of-the-art classical supercomputers [2].

Due to their limited number of qubits, noisy intermediate scale quantum computers may not be capable of supporting fully fault-tolerant quantum operations relying on quantum error correction codes [3]–[6], which are widely believed to be necessary for implementing complex algorithms that require relatively long coherence time [7]–[9], such as Shor's factorization algorithm [10] and Grover's search algorithm [11]. Alternatively, a class of algorithms tailored for these computers, namely that of the variational quantum algorithms (VQAs) [12]–[15], is receiving much attention. Briefly, VQAs aim for sharing their computational tasks between relatively simple quantum circuits and classical computers. A little more specifically, quantum circuits are employed in VQA for computing a cost function or its gradient [16], which is then fed into an optimization algorithm run on classical computers. The objective of this design paradigm is to assist near-term quantum devices in outperforming classical computers in the context of practical problems, such as solving combinatorial optimization problems using the quantum approximate optimization algorithm [14], [17], [18] and quantum chemistry problems using the variational quantum eigensolver [12].

Although the performance of VQAs has been characterized using some illustrative examples [12], [24]–[26], it is not known whether these examples could be scaled up to problems of larger size. In fact, recent analytical results in [19], [27], [28] support the opposite statement. More explicitly, [19] proves that the magnitude of the cost function (or its gradient) computed by VQAs vanishes exponentially as the number of qubits n increases. Fortunately, the follow-up investigations [29], [30] found that this so-called “barren plateau” phenomenon may be mitigated to a certain extent by techniques borrowed from the literature of classical machine learning, such as pre-training and layer-by-layer training. However, the authors of [20] show that when decoherence is taken into account, the dynamic range of the computational results also vanishes exponentially upon increasing the circuit depth N_L , even if these techniques are applied. To summarize, these results imply that when the quantum circuit is long in depth or large in the number of qubits, the computational error become excessive in practical applications.

TABLE I

COMPARISON BETWEEN THE CONTRIBUTIONS OF THIS TREATISE AND EXISTING LITERATURE EVALUATING THE PERFORMANCE OF VQAs AND QEM.

	Circuit condition		Subject of Analysis	Method of performance evaluation
	Noisy?	QEM implementation		
J. R. McClean <i>et al.</i> [13]	×	No QEM	Only accuracy	Only numerical
J. R. McClean <i>et al.</i> [19]	×	No QEM		Analytical and numerical
S. Wang <i>et al.</i> [20]	✓	No QEM		Analytical and numerical
S. Endo <i>et al.</i> [21]	✓	Exact channel inversion	Sampling overhead vs. accuracy trade-off	Only numerical
Y. Xiong <i>et al.</i> [22]	✓	Exact channel inversion		Analytical and numerical
R. Takagi [23]	✓	Exact channel inversion		Analytical and numerical
Our contributions	✓	Monte Carlo-based channel inversion		Analytical and numerical

To improve the error scaling of VQAs with respect to the depth of the circuits, a body of literature has been devoted to searching for methods that efficiently mitigate the effect of decoherence-induced impairment, without using quantum error correction codes [31]–[34]. Among these research efforts, one of the most promising methods is quantum error mitigation (QEM) [31], which aims for applying an “inverse channel” right after each quantum channel modelling the impact of decoherence. Both the numerical and experimental results of [21], [35] show that QEM is indeed capable of reducing the computational error in VQAs in the context of quantum chemistry problems.

The error reduction capability of QEM comes at the price of a computational overhead. To elaborate, QEM is implemented by sampling from a “quasi-probability representation” of the inverse channel, which would increase the variance of the computational results, hence some computational overhead (termed as “sampling overhead” [21], [22]) is required for ensuring that a satisfactory accuracy can be achieved. By appropriately choosing the total number of samples, one may strike a beneficial computational accuracy vs. overhead trade-off. Therefore, it is important to quantify the sampling overhead, before we can conclude whether QEM can play a significant role in making VQAs practical.

The literature of QEM sampling overhead analysis typically assumes that the channel inversion procedure is implemented exactly [21]–[23].¹ Under this assumption, the sampling overhead can be characterized by the sampling overhead factor [22], which is determined by the quality of the channel as well as the by basis operations implementing the channel inversion. However, exact channel inversion may be unrealistic in practical scenarios, since it requires a pre-processing stage that is computationally excessive. Moreover, the computational cost of this pre-processing stage increases rapidly with the number of gates, which may negate the benefit of QEM.

Against this background, in this treatise, we consider a practical channel inversion method based on Monte Carlo sampling, which only increases the pre-processing complexity linearly with the number of gates. The drawback of this method is that it cannot invert the channel exactly, hence there would be some residual error that accumulates during

computation. Compared to the ideal QEM, this method has a less beneficial accuracy vs. overhead trade-off, because additional samples would be necessary to compensate for the residual error. To characterize this trade-off, we investigate the relationship between the residual error and the number of gates N_G , the number of samples N_s , and the gate error probability ϵ . We boldly and explicitly contrast our contributions to the related recent research on VQAs and QEM in Table I, which are further detailed as follows.

- We analyse the error scaling in the absence of QEM, by providing both upper and lower bounds of the magnitude of the computation error. We show that the error magnitude scales linearly with the number of gates N_G , as well as with the gate error probability ϵ , when we have $\epsilon N_G \ll 1$.
- We propose an upper bound on the root-mean-square error (RMSE) of the computational error in the presence of Monte Carlo-based QEM. Specifically, we show that the RMSE is upper bounded by the square root of N_G as well as ϵ , when $\epsilon N_G \ll 1$. This implies that when we use the same number of samples as the ideal QEM, Monte Carlo-based QEM can still provide a quadratic error reduction versus N_G , compared to the case of no QEM.
- We provide an intuitive interpretation of the proposed error scaling laws, by visualizing the decoherence-induced impairments on the Bloch sphere as the quantum circuit executes.
- We demonstrate the analytical results using various numerical examples. Specifically, we consider a practical application of carrying out multiuser detection in wireless communication systems using the quantum approximate optimization algorithm and show that our analytical results do apply.

The rest of this treatise is organized as follows. In Section II, we present the formulation of VQAs and the channels modelling the decoherence. Then, in Section III, we discuss a pair of QEM implementation strategies, namely the Monte Carlo-based QEM and the exact channel inversion. Based on this discussion, in Section IV, we analyse the error scaling behaviours of these two QEM implementations, respectively, under the assumption that they use the same number of circuit executions. We provide further intuitions concerning these analytical results in Section V, with an emphasis on the

¹We will refer to QEM based on exact channel inversion as “ideal QEM” in the rest of this treatise.

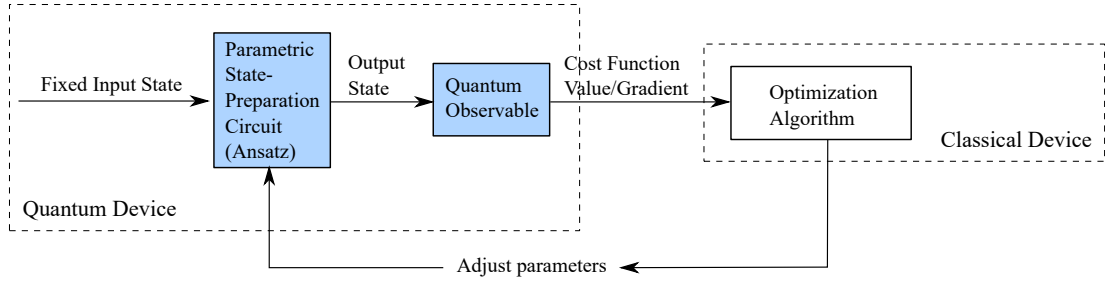


Fig. 1. The structure of a typical implementation for variational quantum algorithms.

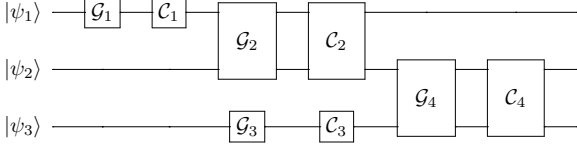


Fig. 2. Simple example of the noisy parametric state-preparation circuit seen in Fig. 1.

accuracy vs. sampling overhead trade-off, complemented by numerical examples in Section VI. Finally, we conclude in Section VII.

II. FORMULATION OF VARIATIONAL QUANTUM ALGORITHMS

A typical VQA iterates between classical and quantum devices, as portrayed in Fig. 1. The parametric state-preparation circuit (also known as the “ansatz” [17]) transforms a fixed input state to an output state, according to the parameters chosen by a classical optimizer. The output state is then measured and fed into a quantum observable, which maps the measurement outcomes to the desired computational results. The results correspond to the value of a cost function or its gradient, which in turn serve as the input of the associated classical optimization algorithm. The iterations continue until certain stopping criterion is met, for example, the computed gradient becomes almost zero.

In this treatise, we focus on the error induced by the sampling procedure in QEM, hence we consider the computational result of a single iteration, meaning that the parameters used for state preparation are fixed. We model the decoherence-induced impairment in the parametric state-preparation circuit as quantum channels acting upon the associated quantum states at the output of perfect quantum gates, as exemplified by the simple circuit shown in Fig. 2. In this figure, C_k , $k = 1 \dots 4$ represents the channel modelling the decoherence in the k -th quantum gate, while G_k represents the k -th ideal decoherence-free quantum gate.

In the subsequent subsections, we present the mathematical formulations of the system models shown in Fig. 1 and 2.

A. Operator-sum Representation

Without loss of generality, we assume that the input state of the circuit is the all-zero state $|0\rangle^{\otimes n}$, where n is the number of qubits. In general, when the circuit is decoherence-free, the

computational result of a variational quantum circuit may be expressed as

$$\tilde{r} = \langle 0 |^{\otimes n} \left(\prod_{k=1}^{N_G} G_k^\dagger \right) \mathcal{M}_{\text{ob}} \left(\prod_{k=1}^{N_G} G_{N_G-k+1} \right) | 0 \rangle^{\otimes n}, \quad (1)$$

where N_G is the number of gates in the circuit, G_k denotes the k -th quantum gate, and the operator \mathcal{M}_{ob} represents the quantum observable, which describes the computational task as a linear function of the final state.

If we consider a more practical scenario, where the quantum state evolves owing to quantum decoherence as the circuit operates, the state can no longer be fully characterized using the state vector formalism. Instead, we may use the density matrix formalism. In particular, the input state may be described as

$$\rho_0 = |0\rangle \langle 0|^{\otimes n}. \quad (2)$$

Correspondingly, the output state of the k -th imperfect quantum gate may be represented in an operator-sum form [36, Sec. 8.2.4], relying on following recursive relationship

$$\rho_k = C_k \left(G_k \rho_{k-1} G_k^\dagger \right), \quad (3)$$

where the operator C_k is characterized by

$$C_k(\rho) = \sum_{i=1}^{n_k} \left(E_{k,i} \rho E_{k,i}^\dagger \right), \quad (4)$$

representing the channel modelling the imperfection of the k -th gate. The matrices $E_{k,i}$ represent the operation elements [36, Sec. 8.2.4] of the channel C_k satisfying the completeness condition of $\sum_{i=1}^{n_k} E_{k,i}^\dagger E_{k,i} = \mathbf{I}$. Finally, when all gates completed their tasks and the measurement results have been obtained, the computational result may be expressed as

$$r = \text{Tr} \{ \mathcal{M}_{\text{ob}} \rho_{N_G} \}. \quad (5)$$

B. Pauli Transfer Matrix Representation

In the standard operator-sum form [36, Sec. 8.2.4], the quantum states are represented by matrices. However, in many applications, such as the error analysis considered in this treatise, it would be more convenient to treat them as vectors. Correspondingly, the quantum channels and gates would then be represented by matrices. To this end, the Pauli transfer matrix (PTM) representation of quantum operators was proposed in [37], which allows a quantum operator \mathcal{O} to be expressed as

$$[\mathcal{O}]_{i,j} = \frac{1}{2^n} \text{Tr} \{ S_i \mathcal{O} (S_j) \}, \quad (6)$$

where \mathcal{S}_i denotes the i -th Pauli operator in the n -qubit Pauli group. Similarly, a quantum state ρ can be expressed as

$$[\rho]_i = \frac{1}{\sqrt{2^n}} \text{Tr} \{ \mathcal{S}_i \rho \}. \quad (7)$$

Under the PTM representation, the computational result may be rewritten as

$$r = \mathbf{v}_{\text{ob}}^T \left(\prod_{k=1}^{N_G} (C_{N_G-k+1} \mathbf{G}_{N_G-k+1}) \right) \mathbf{v}_0, \quad (8)$$

where \mathbf{G}_k represents the k -th perfect gate, and C_k represents the channel modelling the imperfection of the k -th gate. The vector \mathbf{v}_0 denotes the initial state, whereas \mathbf{v}_{ob} is the vector representation of the quantum observable \mathcal{M}_{ob} .

To simplify the notation, we define

$$\mathbf{R}_k := \prod_{i=1}^k (C_{k-i+1} \mathbf{G}_{k-i+1}), \quad \tilde{\mathbf{R}}_k := \prod_{i=1}^k \mathbf{G}_{k-i+1}. \quad (9)$$

Especially, for $k = 0$, we define $\mathbf{R}_0 = \tilde{\mathbf{R}}_0 = \mathbf{I}$. The output state of the k -th quantum gate can then be expressed as

$$\mathbf{v}_k := \mathbf{R}_k \mathbf{v}_0 = C_k \mathbf{G}_k \mathbf{v}_{k-1}. \quad (10)$$

Hence we have

$$r = \mathbf{v}_{\text{ob}}^T \mathbf{v}_{N_G} = \mathbf{v}_{\text{ob}}^T \mathbf{R}_{N_G} \mathbf{v}_0. \quad (11)$$

C. Channel Model

In this treatise, we consider Pauli channels [38], for which the Pauli transfer matrices take the following form

$$C_k = \text{diag} \{ \mathbf{c}_k \}, \quad (12)$$

where

$$\mathbf{c}_k = \tilde{\mathbf{H}} \mathbf{p}_k, \quad (13)$$

with $\tilde{\mathbf{H}}$ denoting the Hadamard transform, whereas \mathbf{p}_k represents a probability distribution satisfying $\mathbf{1}^T \mathbf{p}_k = 1$, $\mathbf{p}_k \geq 0$.

III. QEM AND ITS IMPLEMENTATION STRATEGIES

Ideally, for a channel C_k , QEM would apply its inverse based on a linear combination of predefined quantum operations, taking the following form

$$C_k^{-1} = \sum_{l=1}^L \alpha_l^{(k)} \mathbf{O}_l, \quad (14)$$

where \mathbf{O}_l is the l -th quantum operation, while $\alpha_k := [\alpha_1 \dots \alpha_L]^T$ is the quasi-probability representation vector satisfying $\mathbf{1}^T \alpha_k = 1$. This linear combination may be rewritten as a probabilistic mixture of the quantum operations as follows:

$$C_k^{-1} = \|\alpha_k\|_1 \sum_{l=1}^L s_l^{(k)} p_l^{(k)} \mathbf{O}_l, \quad (15)$$

where $s_l^{(k)}$ and $p_l^{(k)}$ are the l -th entries of \mathbf{s}_k and \mathbf{p}_k , respectively, given by

$$p_i^{(k)} = \frac{|\alpha_i^{(k)}|}{\|\alpha_k\|_1}, \quad \mathbf{s}_k = \text{sgn} \{ \alpha_k \}. \quad (16)$$

Note that the vector \mathbf{p}_k describes a probability distribution.

Typically, the probabilistic mixture in (15) is implemented by generating a set of candidate circuits and performing post-processing on the output of these circuits. In the following subsections, we will discuss different candidate selection strategies and their characteristics.

A. Exact Implementation and Sampling Overhead

The inverse channel C_k^{-1} in (15) is assumed to be implemented exactly in the seminal paper [31] that proposed QEM for the first time, as well as in many other existing contributions [21]–[23]. Exact implementation implies that, each quantum operation \mathbf{O}_l should appear in exactly $N p_l^{(k)}$ candidate circuits in every N samples of the computational result.

The assumption of exact implementation significantly simplifies the performance analysis of QEM. In particular, it leads to a clear and concise formula of sampling overhead, which describes the computational overhead imposed by the variance-boosting effect of QEM. To elaborate, assume that the variance of the computational result is σ^2 based on N_0 samples. According to (15), if we implement the inverse channel C_k^{-1} , the variance would become $\|\alpha_k\|_1^2 \sigma^2$. Therefore, in order to achieve the same accuracy as the case without QEM, we should acquire $N_0 (\|\alpha_k\|_1^2 - 1)$ additional samples. If we further assume that all gates are protected by QEM, we have the following formula for the total sampling overhead

$$N_{\text{exact}} = N_0 \left(\prod_{k=1}^{N_G} \|\alpha_k\|_1^2 - 1 \right). \quad (17)$$

The simplicity of (17) is largely due to the assumption of exact implementation.

Despite its theoretical convenience, the practicality of exact implementation is doubtful. Specifically, the number of the l -th candidate circuit, $N p_l^{(k)}$ has to be an integer, which might be unrealistic for an arbitrary $p_l^{(k)}$. Furthermore, the number of the probability parameters $p_l^{(k)}$ would increase exponentially as the number of QEM-protected gates increases, which may render the candidate circuit selection procedure computationally prohibitive when N_G is large. Motivated by these drawbacks, we propose to use a Monte Carlo implementation of QEM, detailed in the next subsection.

B. Monte Carlo Implementation

In the Monte Carlo implementation, we first sample from the probability distribution \mathbf{p}_k for each gate, and obtain N samples constituting a set $\mathbf{L} = \{l_1, \dots, l_N\}$, where for all k we have $l_k = 1, 2, \dots, L$. Thus we may approximate the inverse channel as

$$\mathbf{\Gamma}_k = \frac{\|\alpha_k\|_1}{N} \sum_{i=1}^N s_{l_i}^{(k)} \mathbf{O}_{l_i} = \|\alpha_k\|_1 \sum_{l=1}^L s_l^{(k)} \tilde{p}_l^{(k)} \mathbf{O}_l, \quad (18)$$

where

$$\tilde{p}_m^{(k)} = \frac{1}{N} \sum_{i=1}^N \mathbb{I}\{l_i = m\}.$$

The advantage of the Monte Carlo approach is that it may result in a much lower complexity for candidate circuit generation, compared to the exact implementation. To elaborate further, as a “toy” example, for a circuit consisting of two gates we have

$$\mathbf{r}_2 \tilde{\mathbf{G}}_2 \mathbf{r}_1 \tilde{\mathbf{G}}_1 = \frac{\|\alpha_1\|_1 \|\alpha_2\|_1}{N} \sum_{i=1}^N s_{l_{i,1}}^{(1)} s_{l_{i,2}}^{(2)} \mathbf{O}_{l_{i,2}} \tilde{\mathbf{G}}_2 \mathbf{O}_{l_{i,1}} \tilde{\mathbf{G}}_1, \quad (19)$$

where $\tilde{\mathbf{G}}_k = \mathbf{C}_k \mathbf{G}_k$, and $l_{i,k}$ denotes the i -th sample drawn from the distribution \mathbf{p}_k . This implies that in order to obtain a sample for the entire circuit, we may simply generate one sample for each gate, and concatenate them as shown in the right hand side of (19). Compared to the exact implementation, the Monte Carlo implementation can generate an arbitrary number of circuit samples N , at a relatively low computational cost of $O(NN_G)$.

The reduced complexity of the Monte Carlo implementation comes with a cost of inaccurate channel inversion, since \mathbf{G}_k is only an approximation of \mathbf{C}_k^{-1} . Hence there would be a residual channel for each gate, which is given by

$$\tilde{\mathbf{C}}_k = \mathbf{r}_k \mathbf{C}_k. \quad (20)$$

A natural question that arises is, whether the additional computational error caused by these residual channels would erode the error reduction capability of Monte Carlo-based QEM. In the rest of this treatise, we will discuss the impact of these residual channels on the accuracy vs. sampling overhead trade-off.

IV. ERROR SCALING ANALYSIS OF MONTE CARLO-BASED QEM

In this section, we discuss the error scaling behaviour of quantum circuits protected by Monte Carlo-based QEM, and contrast the results to that of circuits without QEM protection. In order to make a fair comparison, we consider the following assumptions.

A. Assumptions

Assumption 1 (Bounded gate error rate): The error probability of each quantum gate is upper bounded by ϵ_u .

Since we consider Pauli channels in this treatise, the gate error probability corresponding to a quantum channel \mathbf{C}_k (under its PTM representation) may be computed as

$$\epsilon(\mathbf{C}_k) = 1 - \frac{1}{4^n} \text{Tr} \{ \mathbf{C}_k \}. \quad (21)$$

Assumption 2 (Bounded observable): The eigenvalues of the quantum observable \mathcal{M}_{ob} are bounded in the interval $[-1, 1]$.

Assumption 2 ensures the boundedness of the computation result r . In this treatise we assume that the upper and lower bounds are 1 and -1 , respectively, but they may be replaced with any other constant real numbers without affecting our analytical results. The assumption may also be rewritten as

$$\max_{\mathbf{v} \in \mathbb{S}^n} \mathbf{v}_{\text{ob}}^T \mathbf{v} \mathbf{v}_{\text{ob}}^T \leq 1, \quad (22)$$

where \mathbb{S}^n denotes the space of all density matrices over n qubits. Furthermore, the assumption also implies that

$$\|\mathbf{v}_{\text{ob}}\|_2 \leq \sqrt{2^n}. \quad (23)$$

This follows from the fact that $\|\mathbf{v}_{\text{ob}}\|_2 = \|\mathcal{M}_{\text{ob}}\|_F$, and that

$$\|\mathcal{M}_{\text{ob}}\|_F = \sum_{i=1}^{2^n} \lambda_i(\mathcal{M}_{\text{ob}}) \leq \sqrt{2^n},$$

where $\lambda_i(\cdot)$ denotes the i -th largest eigenvalue of its argument.

Assumption 3 (Zero bias term): We assume that

$$\text{Tr} \{ \mathcal{M}_{\text{ob}} \} = \sqrt{2^n} [\mathbf{v}_{\text{ob}}]_1 = 0. \quad (24)$$

Note that $[\mathbf{v}_{\text{ob}}]_1$ is the coefficient of the identity operator, which serves as a bias term in the computation result being constant with respect to the quantum state. Thus this assumption does not restrict the generality of our results.

B. Benchmark: Error Scaling in the Absence of QEM

In this subsections, we characterize the error scaling of quantum circuits that are not protected by QEM. The results will serve as important benchmarks in the following discussions. Let us start with a bound of the dynamic range of computational results, which will lead to a lower bound of the computational error.

Proposition 1: Assume that each qubit would be processed by at least N_L gates, and that for each of these gates, the probability of each type of Pauli error (i.e., X error, Y error or Z error) on each qubit is lower bounded by ϵ_l . The computational result r exhibits the following convergence behaviour:

$$|r| \leq \exp(-4\epsilon_l N_L). \quad (25)$$

Proof: Please refer to Appendix I. \square

Proposition 1 implies that decoherence would force the computation result to be almost independent of the quantum observable \mathbf{v}_{ob} in an asymptotic sense. Indeed, as indicated by (25), when N_L is large, r is only determined by the first entry of \mathbf{v}_{ob} . Moreover, consider the case where $|\tilde{r}| \geq 1 - c$ holds for all N_G , the computational error is lower bounded as

$$|r - \tilde{r}| \geq 1 - c - \exp(-4\epsilon_l N_L). \quad (26)$$

From the Taylor expansion

$$\exp(-4\epsilon_l N_L) = 1 - 4\epsilon_l N_L + \frac{(4\epsilon_l N_L)^2}{2} - \dots,$$

we see that when $\epsilon_l N_L \ll 1$, the lower bound is approximately

$$|r - \tilde{r}| \gtrsim 4\epsilon_l N_L - c, \quad (27)$$

which increases linearly with respect to $\epsilon_l N_L$.

We may also provide an upper bound for the computational error as follows.

Proposition 2: The computational error can be upper bounded as

$$|r - \tilde{r}| \leq 2\epsilon_u N_G. \quad (28)$$

Proof: Please refer to Appendix II. \square

Combining Propositions 1 and 2, we see that the computational error grows linearly with N_G , when the number of gates in each “layer” is constant (hence N_L is a constant multiple of N_G). This is typically true for VQAs.

C. The Statistics of the Residual Channels

Before diving into details about the error scaling, in this subsection, we first investigate the characteristics of the residual channels of gates protected by Monte Carlo-based QEM.

According to the sampling overhead analysis in [22] based on the assumption of exact channel inversion, if we wish to execute the decoherence-free circuit N_s times, we should sample from the probabilistic mixture of candidate circuits for as many as

$$N = N_s \|\alpha_k\|_1^2 \quad (29)$$

times, in order to keep the variance of the computational result unchanged by the channel inversion procedure. Here we consider the Monte Carlo-based channel inversion using the same number of samples, hence we have

$$\tilde{p}_m^{(k)} = \frac{1}{N_s \|\alpha_k\|_1^2} \sum_{i=1}^{N_s \|\alpha_k\|_1^2} \mathbb{I}\{l_i = m\}. \quad (30)$$

Of course, the Monte Carlo-based channel inversion has lower accuracy compared to the exact channel inversion, when they use the same number of samples. The accuracy could be improved by using additional samples, which will be discussed in more detail in Section V-B.

Using (12), the quasi-probability representation vector may be expressed as

$$\alpha_k = \widetilde{H}^{-1}(1/c_k), \quad (31)$$

where \widetilde{H} is the Hadamard transform over n qubits, and \widetilde{H}^{-1} is the corresponding inverse transform given by $\widetilde{H}^{-1} = \frac{1}{4^n} \widetilde{H}$. After the sampling procedure, α_k is approximated by $\tilde{\alpha}_k$ taking the following form

$$\begin{aligned} \tilde{\alpha}_k &= \|\alpha_k\|_1 \cdot s_k \odot \tilde{p}_k \\ &= \|\alpha_k\|_1 \cdot s_k \odot (p_k + \mathbf{n}) \\ &= \alpha_k + \|\alpha_k\|_1 \cdot s_k \odot \mathbf{n}, \end{aligned} \quad (32)$$

where \mathbf{n} denotes the sampling error. The approximated inverse channel can now be expressed as

$$\Gamma_k = \text{diag}^{-1}\{\widetilde{H}\tilde{\alpha}_k\}, \quad (33)$$

and thus the residual channel takes the following form

$$\tilde{C}_k = \text{diag}^{-1}\{\widetilde{H}\tilde{\alpha}_k \odot c_k\}. \quad (34)$$

To simplify our further analysis, we introduce $\tilde{c}_k := \widetilde{H}\tilde{\alpha}_k \odot c_k$, where \tilde{c}_k may be further expressed as

$$\begin{aligned} \tilde{c}_k &= \mathbf{1} + \|\alpha_k\|_1 \cdot \widetilde{H}(s_k \odot \mathbf{n}) \odot c_k \\ &= \mathbf{1} + \|\alpha_k\|_1 \cdot c_k \odot \tilde{\mathbf{n}}, \end{aligned} \quad (35)$$

and $\tilde{\mathbf{n}} := \widetilde{H}(s_k \odot \mathbf{n})$. Note that the vector \tilde{p}_k is a multinomial distributed random vector, satisfying

$$\mathbb{E}\{\tilde{p}_k\} = p_k, \quad \text{Cov}\{\tilde{p}_k\} = \frac{1}{N_s \|\alpha_k\|_1^2} (P_k - p_k p_k^T), \quad (36)$$

where $P_k = \text{diag}\{p_k\}$. Therefore, the vector $\tilde{\mathbf{n}}$ satisfies

$$\mathbb{E}\{\tilde{\mathbf{n}}\} = \mathbf{0}, \quad \text{Cov}\{\tilde{\mathbf{n}}\} = \widetilde{H} \text{Cov}\{\tilde{p}_k\} \widetilde{H}, \quad (37)$$

since the sign vector s_k does not have an effect on the covariance matrix. Thus we have the following results for \tilde{c}_k :

$$\begin{aligned} \mathbb{E}\{\tilde{c}_k\} &= \mathbf{1}, \\ \text{Cov}\{\tilde{c}_k\} &= \|\alpha_k\|_1^2 \cdot \text{Cov}\{c_k \odot \tilde{\mathbf{n}}\} \\ &= \frac{1}{N_s} \widetilde{H} (P_k - p_k p_k^T) \widetilde{H} \odot c_k c_k^T. \end{aligned} \quad (38)$$

For simplicity of further derivation, we use the notation of $\Xi_k := \text{Cov}\{\tilde{c}_k\}$.

D. Error Scaling in the Presence of Monte Carlo-based QEM

In this subsection, we investigate the scaling law of computational error when the quantum circuit is protected by Monte Carlo-based QEM, based on the above discussions concerning the residual channels in the previous subsection.

We note that for QEM-protected circuits, the computational result is a random variable due to the randomness in the sampling procedure, given by

$$r = v_{\text{ob}}^T v_{N_G}, \quad (39)$$

where

$$v_k = R_k v_0 = \tilde{C}_k G_k v_{k-1}. \quad (40)$$

Using these definitions, we have the following bound on the RMSE of the computational result r .

Proposition 3 (Square-root Increase of QEM Inaccuracy): For a quantum circuit consisting of N_G gates which is protected by QEM, the RMSE of the computational result is upper bounded by

$$\sqrt{\mathbb{E}\{(r - \tilde{r})^2\}} \leq 2^{n/2} \sqrt{\exp(2N_G N_s^{-1}) - 1}. \quad (41)$$

Proof: Please refer to Appendix III. \square

Note that by applying the Taylor expansion to $\exp(2N_G N_s)$, we have

$$\exp(2N_G N_s) - 1 = \frac{2}{N_s} N_G + \frac{1}{2} \left(\frac{2}{N_s} N_G \right)^2 + \dots,$$

which is approximately $2N_G N_s^{-1}$, when $N_G \ll N_s$. This means that when the RMSE is far less than 1, its scaling law is given by $O(\sqrt{N_G}/\sqrt{N_s})$. This is particularly useful, since in typical applications (e.g., variational quantum algorithms), having an RMSE close to 1 would be excessive.

In Proposition 3, the dependence of the RMSE on the error probability of quantum gates is not demonstrated. According to (71) of the Appendix, this dependence mainly relies on the term $\|\Xi_k\|_{\max}$. Next we expound a little further on this issue based on Assumption 1.

Proposition 4: Under Assumption 1, we have the following refined upper-bound for the RMSE of the computational result:

$$\begin{aligned} \sqrt{\mathbb{E}\{(r - \tilde{r})^2\}} &\leq 2^{n/2} \sqrt{\exp(\tilde{\epsilon} N_G N_s^{-1}) - 1} \\ &\approx 2^{n/2} \sqrt{\exp(10\epsilon_u N_G N_s^{-1}) - 1} \end{aligned} \quad (42)$$

where $\tilde{\epsilon}$ is given by

$$\tilde{\epsilon} := \frac{5}{2} \sigma_u + \frac{1}{4} \sigma_u^2, \quad (43)$$

and

$$\sigma_u := 4\epsilon_u \cdot \frac{1 - \epsilon_u}{(1 - 2\epsilon_u)^2}. \quad (44)$$

The approximation is valid when $\epsilon_u \ll 1$.

Proof: Please refer to Appendix IV. \square

Proposition 4 implies that, when $\epsilon_u N_G \ll N_s$, the RMSE is on the order of $O(\sqrt{\epsilon_u N_G}/\sqrt{N_s})$.

Remark 1: We note that the factor $2^{n/2}$ in (41) and (42) is an artifact of our specific proof technique. According to the numerical results which will be presented in Section VI, we conjecture this scaling factor is essentially unnecessary, implying that

$$\sqrt{\mathbb{E}\{(r - \tilde{r})^2\}} \leq \sqrt{\exp(\epsilon N_G N_s^{-1}) - 1}. \quad (45)$$

However, it seems to be technically challenging to remove this factor from the bounds. Further investigations into this issue will be left for our future research.

V. DISCUSSIONS

In this section, we provide further intuitions concerning the analytical results of Section IV.

A. Intuitions about the Error Scaling with the Circuit Size

The results in Section IV indicate that, with respect to N_G , we observe an $O(\sqrt{N_G})$ scaling of the computational error of circuits protected by Monte Carlo-based QEM, when the number of samples is the same as that of QEM based on exact channel inversion. By contrast, when QEM is not applied, the computational error scales as $O(N_G)$, as discussed in Section IV. Thus we may conclude that, although there are residual channels due to the inexact channel inversion, Monte Carlo-based QEM can still slow down the accumulation of computational error.

We may understand these error scaling behaviours more intuitively by considering the low-complexity example of a single-qubit circuit, where the entire space of all legitimate quantum states can be described by the celebrated Bloch sphere [36, Sec. 1.2]. As demonstrated in Fig. 3, the Bloch sphere would shrink as N_G increases when no QEM is applied, since the Pauli channels are contractive. This is in stark contrast with the case where Monte Carlo-based QEM is applied, when the Bloch sphere becomes “blurred” as N_G increases, since it is not determined whether the sphere will expand or shrink after each gate. Consequently, the sphere may expand after one gate and then shrink after another, hence the corresponding errors will cancel each other to certain extent. In light of the aforementioned intuition, we may interpret the error scaling of Monte Carlo-based QEM in following informal way.

Assume that every gate k would transform the Bloch sphere in a way that its radius becomes $(1 + \lambda_k)$ times that of its original value, where λ_k is a zero-mean random variable with variance σ_k^2 . If additionally all λ_k values are mutually independent, we can see that

$$\frac{1}{N_G} \sum_{k=1}^{N_G} \ln(1 + \lambda_k) \sim \mathcal{N}\left(-\frac{1}{2}\sigma^2, \sigma^2\right)$$

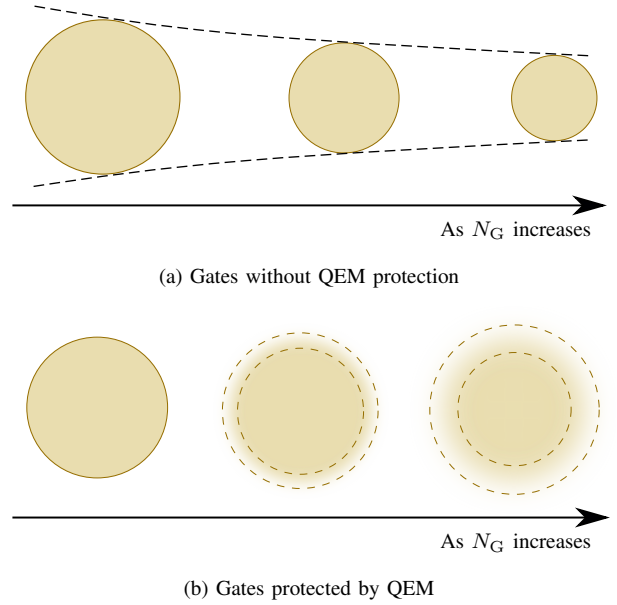


Fig. 3. Graphical illustration of the Bloch sphere undergoing a sequence of N_G imperfect gates. The Bloch sphere shrinks when QEM is not applied, whereas it becomes “blurred” when QEM is applied.

holds asymptotically as $N_G \rightarrow \infty$ by applying the central limit theorem, where we have:

$$\sigma^2 = \frac{1}{N_G} \sum_{k=1}^{N_G} \sigma_k^2.$$

Hence the radius of the Bloch sphere after k gates, denoted by a_k , tends to be a log-normally distributed random variable characterized by

$$\mathbb{E}\{a_k\} = 1, \quad \text{Var}\{a_k\} = \exp(N_G \sigma^2) - 1.$$

Therefore, the standard deviation of the Bloch sphere’s radius tends to be $\sqrt{\exp(N_G \sigma^2) - 1}$, which is on the order of $O(\sqrt{N_G \sigma^2})$ when $N_G \sigma^2 \ll 1$. This agrees with our formal analytical results.

The linear error scaling experienced in the case where no QEM is applied may be interpreted by considering the graphical illustration in Fig. 4. Since the computational result r converges exponentially fast to zero as indicated by Proposition 1, it deviates from \tilde{r} linearly when N_G is relatively small, which may be viewed as a lower bound of the computational error. Additionally, the actual evolution of r is also bounded by the tangent line of it at $N_G = 0$, which gives rise to the error upper-bound in Proposition 2.

B. The Accuracy vs. Sampling Overhead Trade-off

If we denote by ϵ the average error probability of each gate, from the discussion in Section IV we see that the computational error roughly scales as $\Theta(\epsilon)$ when QEM is not applied, whereas it scales as $O(\sqrt{\epsilon})$ when Monte Carlo-based QEM is applied, according to Section IV-D. This may be understood by considering the variance of the samples, which is proportional to ϵ . Hence the RMSE is proportional to $\sqrt{\epsilon}$.

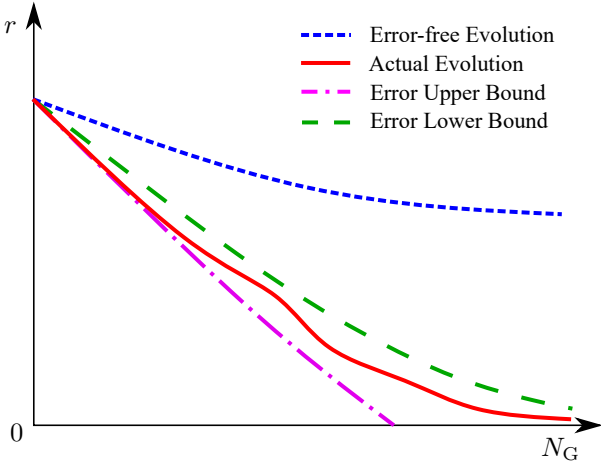


Fig. 4. Demonstration of the evolution of computational result r in the absence of QEM, as a function of N_G . The upper and lower bound of computational error correspond to the results in Proposition 2 and Proposition 1, respectively.

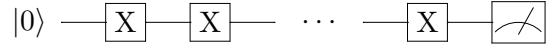
Since ϵ is typically far less than 1, it seems that Monte Carlo-based QEM has a less preferable performance. Nevertheless, it is noteworthy that the error scaling in the QEM-protected case is actually $O(\sqrt{\epsilon N_s^{-1}})$, where the number N_s of effective circuit executions is a configurable parameter. The $O(\sqrt{N_s^{-1}})$ dependency on N_s originates from the fact that the sampling variance scales as $O(N_s^{-1})$. Therefore, our results should not be viewed as indicating the superiority of non-QEM-based solutions. Rather, they should be viewed as a suggestion on the specific selection of N_s , in the sense that it should be on the order of ϵ to ensure the error scaling is as beneficial as that of the family of non-QEM-based solutions.

Similarly, by increasing N_s as a function of N_G , one could also improve the error scaling of Monte Carlo-based QEM with the circuit size. Indeed, since the error of Monte Carlo-based QEM scales as $O(\sqrt{N_G N_s^{-1}})$, we can choose an N_s that is proportional to N_G in order to attain a constant error with respect to N_G . Note that the exact channel inversion also has a constant error with respect to N_G in the asymptotic limit of $N_G \rightarrow \infty$. Therefore, using Monte Carlo-based QEM, we could use N_G times the number of samples to attain the same error scaling as that of QEM based on exact channel inversion. In practical scenarios, however, this may be an excessive sampling overhead. Fortunately, even if we use the same number of samples as that of the exact channel inversion, Monte Carlo-based QEM still exhibits a quadratic error scaling improvement compared to the no-QEM-based case.

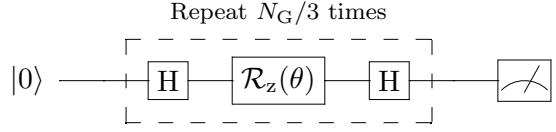
VI. NUMERICAL RESULTS

In this section, we evaluate the analytical results presented in the previous sections via numerical examples. If not otherwise stated, the following parameters and assumptions will be used throughout the section.

- The number of effective circuit executions is $N_s = 5000$;
- For Monte Carlo-based QEM, we use the same number of samples (i.e., actual circuit executions) as that of QEM based on exact channel inversion;



(a) Repeated Pauli X gates.



(b) Repeated θ -rotations around the X-axis.

Fig. 5. Circuits implementing rotations around the X-axis of the Bloch sphere.

- The quantum channels modelling the gate imperfections are single-qubit depolarising channels having gate error probability 10^{-3} .

A. Rotations Around the Bloch Sphere

We first consider the simplest scenario, where the quantum circuits are constituted of single-qubit gates, because these simple circuits allow us to clearly observe the error scaling described in the previous sections. In particular, we consider the circuits shown in Fig. 5. The quantum observable \mathcal{M}_{ob} in this example is the Pauli Z operator \mathcal{Z} on the qubit, which satisfies

$$\mathcal{Z} |0\rangle = |0\rangle, \quad \mathcal{Z} |1\rangle = -|1\rangle.$$

The corresponding PTM representation is given by $\mathbf{v}_{\text{ob}} = [0 \ 0 \ 0 \ \sqrt{2}]^T$.

For the circuit consisting of repeated Pauli X gates shown in Fig. 5a, the RMSE of the computational results both with and without QEM protection is demonstrated in Fig. 6a, as a function of N_G . As it can be observed from the figure, when N_G is relatively small, the RMSE of circuits operating without QEM protection grows linearly with N_G , while the RMSE of circuits protected by Monte Carlo-based QEM scales as $O(\sqrt{N_G})$. The RMSE of QEM based on exact channel inversion scales as $O(\sqrt{N_G})$ for small N_G , but converges to a constant ($\approx \sqrt{N_s^{-1}}$) when N_G is large. Furthermore, when N_G is large, the RMSE of circuits operating without QEM protection converges to a constant. This agrees with Proposition 1, which indicates that their computational results converge to zero regardless of the quantum observable.

The RMSE scalings with respect to the gate error probability ϵ are shown in Fig. 6b, where we choose $N_G = 10$, while the number of effective circuit executions, namely $N_s = 5000$, does not vary as the gate error probability increases. It is noteworthy that when ϵ is small, the RMSE of circuits operating without QEM protection is lower than that of their counterparts protected by QEM. This phenomenon may be understood from our discussion in Section V-B, where we have indicated that the error scaling of QEM-protected circuits is $O(\sqrt{\epsilon N_s^{-1}})$. Compared to the $O(\epsilon)$ scaling of non-QEM-protected circuits, the RMSE may be higher when ϵ is much smaller than N_s . Interestingly, as seen from the figure, the square root scaling with respect to ϵ becomes preferable to the linear scaling when ϵ is relatively large.

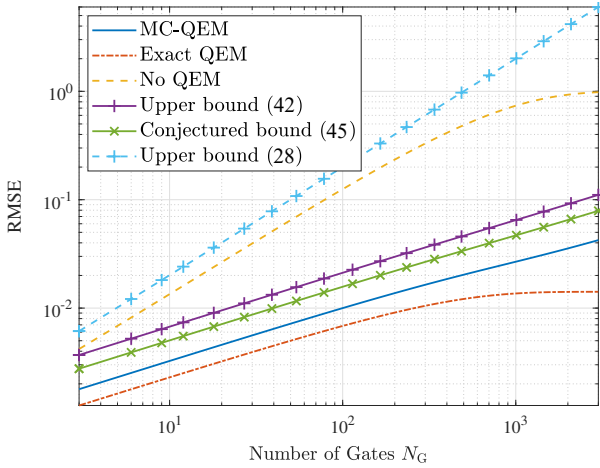
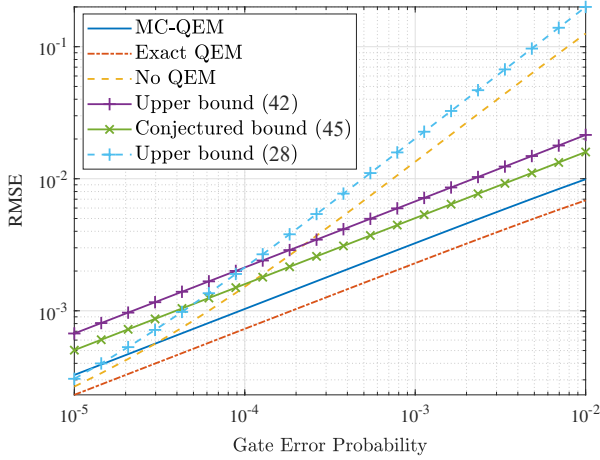
(a) RMSE versus N_G .(b) RMSE versus gate error probability ($N_G = 10$).

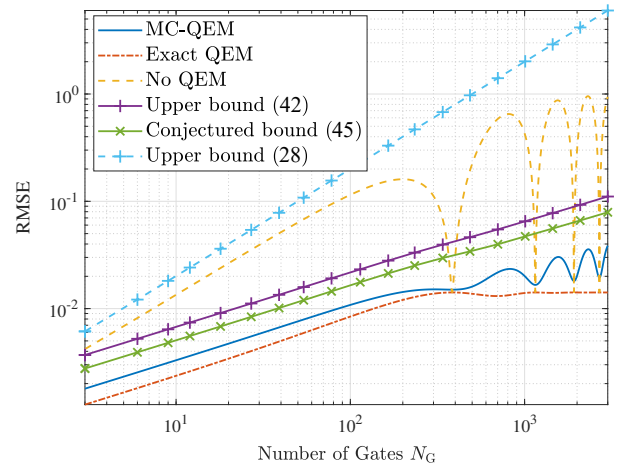
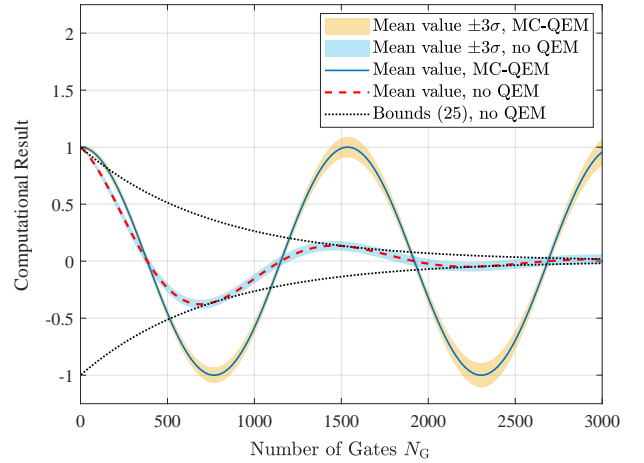
Fig. 6. The RMSE of the results computed by quantum circuits consisting of repeated Pauli X gates (as demonstrated in Fig. 5a).

For the circuit comprising repeated rotations around the X -axis, as illustrated in Fig. 5, we set $\theta = \pi/256$, and the results are plotted in Fig. 7. Observe that the envelope of the RMSE curves exhibit similar scaling behaviours as those in Fig. 6b, but there are some oscillations. To understand the RMSE oscillations of circuits protected by Monte Carlo-based QEM, from (65) we may express the covariance matrix of \mathbf{v}_k as follows

$$\Sigma_k = (\mathbf{1}\mathbf{1}^T + \Xi_k) \odot G_k \Sigma_{k-1} G_k^\dagger + \Xi_k \odot \mu_k \mu_k^T. \quad (46)$$

Note that the term $\Xi_k \odot \mu_k \mu_k^T$ varies with k under the observable $\mathcal{M}_{\text{ob}} = \mathcal{Z}$, and hence the RMSE is oscillatory.

The RMSE oscillation of non-QEM-protected circuits may be better understood by investigating the evolution of the computational result r as N_G increases, which is portrayed in Fig. 8. It can be seen that the mean values of the non-QEM-protected circuits fit nicely within the bounds given by Proposition 1. Furthermore, the RMSE of QEM-protected circuits is mainly contributed by the variance of the computation results, while the RMSE of circuits not protected by QEM is mainly determined by the mean value, since in the latter case the bias is far larger than the standard deviation. As the

Fig. 7. The RMSE of the results computed by quantum circuits carrying out repeated rotations around the X -axis of the Bloch sphere (as shown in Fig. 5b), as functions of N_G , when $\theta = \pi/256$.Fig. 8. The computational results of QEM-protected and non-QEM-protected circuits configured for carrying out repeated rotations ($\theta = \pi/256$) around the X -axis of the Bloch sphere (shown in Fig. 5b), as functions of N_G .

dynamic range of the mean values is reduced, by coincidence, there are multiple intersections of the ground truth and the mean values, and thus the computational error of non-QEM-protected circuits oscillates as N_G increases.

Finally, we demonstrate that some non-Pauli channels may also exhibit the $O(\sqrt{N_G})$ error scaling. In particular, we consider amplitude damping channels [39] having the following PTM representation

$$C_{\text{damp}} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & \sqrt{1-\gamma} & 0 & 0 \\ 0 & 0 & \sqrt{1-\gamma} & 0 \\ \gamma & 0 & 0 & 1-\gamma \end{pmatrix}, \quad (47)$$

where γ is the amplitude damping probability. Here, we set the amplitude damping probability to $\gamma = 1 \times 10^{-3}$. The RMSE scalings with respect to the number of gates N_G are shown in Fig. 9a for the circuit comprising repeated Pauli X gates, and in Fig. 9b for the circuit consisting of repeated $(\pi/256)$ rotations around the X -axis. We observe

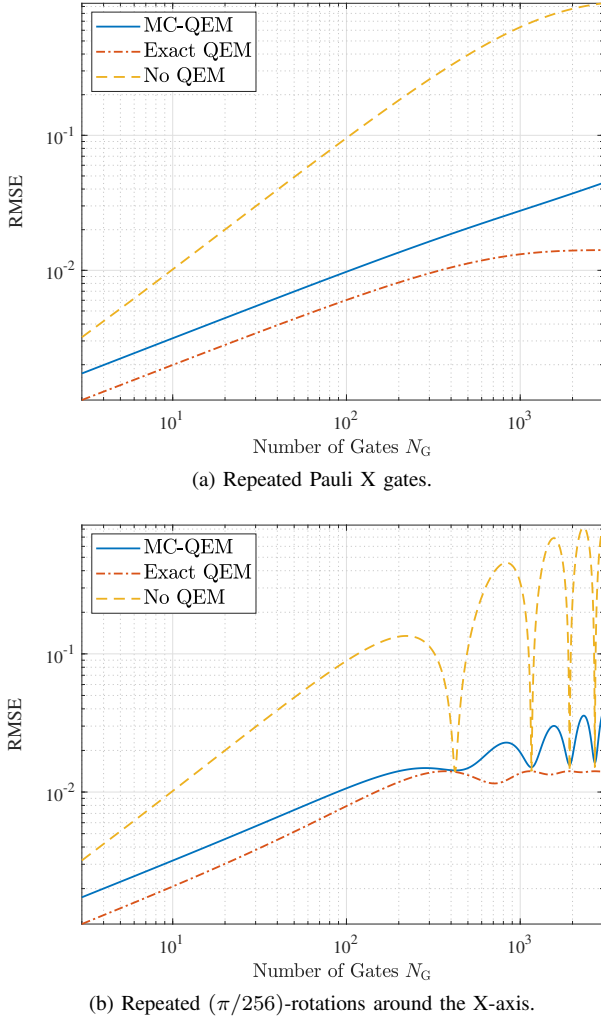


Fig. 9. The RMSE of the results computed by quantum circuits demonstrated in Fig. 5, which are contaminated by amplitude damping channels.

that the curves corresponding to QEM based on exact channel inversion and those corresponding to Monte Carlo-based QEM exhibit the $O(\sqrt{N_G})$ scaling behavior, while the non-QEM-protected curves scale as $O(N_G)$, which is similar to the error scaling under Pauli channels as portrayed in Fig. 6 and Fig. 7.

B. The Quantum Approximate Optimization Algorithm Aided Multi-User Detection

In this subsection, we apply our analytical results to a practical variational quantum algorithm, the quantum approximate optimization algorithm [14], which aims for solving combinatorial optimization problems assuming the following form

$$\max_{\mathbf{z} \in \{-1, +1\}^n} \sum_{k=1}^K w_k \prod_{i=1}^{n_k} z_{l_{k,i}}, \quad (48)$$

where $\mathbf{z} = [z_1 \dots z_n]^T$, and $l_{k,i} \in \{1, 2, \dots, n\}$. In the formulation of the quantum approximate optimization algo-

rithm, the problem (48) is transformed into the maximization of $\langle \psi | \mathcal{H} | \psi \rangle$, where the quantum observable \mathcal{H} is given by

$$\mathcal{H} = \sum_{k=1}^K w_k \prod_{i=1}^{n_k} \mathcal{Z}_{l_{k,i}}. \quad (49)$$

The trial state $|\psi\rangle$ is prepared using a parametric circuit having an alternating structure, so that

$$|\psi\rangle = e^{-i\beta_P \mathcal{B}} e^{-i\gamma_P \mathcal{H}} \dots e^{-i\beta_1 \mathcal{B}} e^{-i\gamma_1 \mathcal{H}} |+\rangle^{\otimes n},$$

where P is the number of stages in the alternating circuit, and \mathcal{B} is the “mixing Hamiltonian” [17] given by $\mathcal{B} = \sum_{i=1}^n \mathcal{X}_i$. The parameters $\beta = [\beta_1 \dots \beta_P]^T$ and $\gamma = [\gamma_1 \dots \gamma_P]^T$ are typically obtained using via an optimization procedure implemented on classical computers [12]. For the purpose of this treatise, here we do not optimize the parameters, but use the following (suboptimal) adiabatic configuration [40] instead

$$\gamma_k = kP^{-1}, \quad \beta_k = 1 - kP^{-1}.$$

We consider the multiuser detection problem of wireless communications [41]. In particular, assuming that the modulation scheme is BPSK, in a spatial division multiple access system, the signal received at a base station equipped with m antennas from n single-antenna uplink transmitters may be expressed as

$$\mathbf{y} = \mathbf{H}\mathbf{x} + \boldsymbol{\omega},$$

where $\mathbf{H} \in \mathbb{R}^{m \times n}$ denotes the channel, $\mathbf{x} \in \{-1, +1\}^n$ represents the transmitted signal, and $\boldsymbol{\omega} \in \mathbb{R}^m$ is the noise. We assume here that the noise is i.i.d. Gaussian. Hence the maximum likelihood estimate of \mathbf{x} is given by

$$\hat{\mathbf{x}}_{\text{ML}} = \arg \max_{\mathbf{x} \in \{-1, +1\}^n} 2(\mathbf{H}\mathbf{y})^T \mathbf{x} - \mathbf{x}^T \mathbf{H}^T \mathbf{H} \mathbf{x}.$$

This may be further reformulated as the maximization of the quadratic form $\langle \psi | \mathcal{H} | \psi \rangle$, where

$$\mathcal{H} = \frac{1}{Z} \left(\sum_{i=1}^n [\mathbf{H}^T \mathbf{y}]_i \mathcal{Z}_i - \sum_{i=1}^{n-1} \sum_{j=i+1}^n [\mathbf{H}^T \mathbf{H}]_{i,j} \mathcal{Z}_i \mathcal{Z}_j \right), \quad (50)$$

and Z is a normalizing coefficient ensuring that the quantum observable \mathcal{H} satisfies our Assumption 2.

In this illustrative example, we consider the case where $m = n = 4$, and $[\boldsymbol{\omega}]_i \sim \mathcal{N}(0, 0.0631)$, $\forall i$, such that the signal-to-noise ratio is 12dB. We assume furthermore that the channels between each pair of antennas are uncorrelated non-dispersive Rayleigh channels, hence the entries of the channel \mathbf{H} are i.i.d. Gaussian variables with zero mean and variance m^{-1} [42]. For the quantum circuits, we choose gate error probability $\epsilon = 3 \times 10^{-4}$. Under these assumptions, the RMSE scalings with respect to P of non-QEM-protected circuits and that of circuits protected by Monte Carlo-based QEM are portrayed in Fig. 10. It can be observed that the non-QEM-protected circuits exhibit an $O(P)$ scaling, while the QEM-protected circuits exhibit an $O(\sqrt{P})$ scaling, as indicated by Propositions 2 and 4, respectively.

To illustrate the evolution of the computational results during the execution of circuits, we plot the objective function

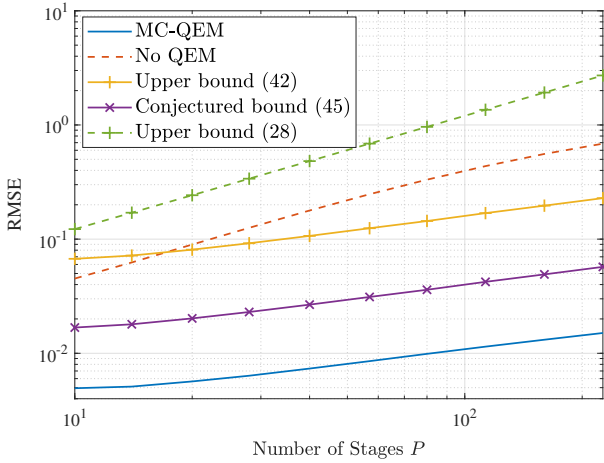


Fig. 10. The RMSE of the results computed by QEM-protected and non-QEM-protected circuits implementing the quantum approximate optimization algorithm based on (50), as functions of the number of stages P .

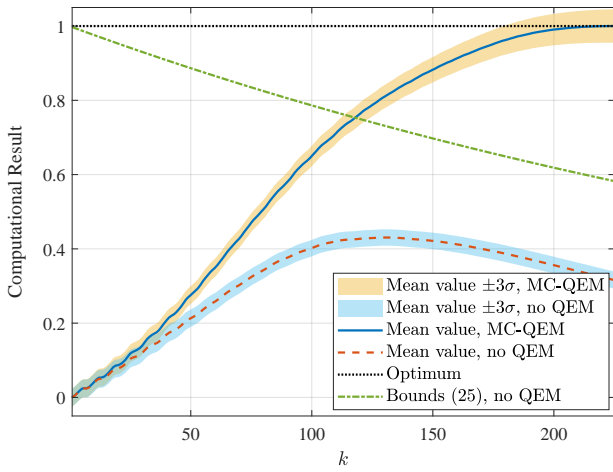


Fig. 11. The objective function values computed at the k -th stage of the quantum approximate optimization algorithm (for which $P = 225$) implemented based on (50).

values (i.e., $\langle \psi | \mathcal{H} | \psi \rangle$) computed at each stage k of the circuits in Fig. 11, for the case where $P = 225$. Note that the results computed by the QEM-protected circuits converge monotonically towards the optimum, for which the main source of error is the variance. By contrast, for the non-QEM-protected circuits, the results were on the right track for $k < 100$, but soon they deviate from their QEM-protected counterparts, and start to converge to zero. In this example, the bound (25) is not as tight as it was in Section VI-A, but it still indicates that the dynamic range of the results computed by non-QEM-protected circuits decays exponentially as k increases.

VII. CONCLUSIONS

The trade-off between the computational overhead and the error scaling behaviour of both quantum circuits protected by Monte Carlo-based QEM and their non-QEM-protected counterparts was investigated. As for the non-QEM-protected circuits, we have shown that the dynamic range of the noisy computational results shrinks exponentially as the number

of gates N_G increases, implying a linear error scaling with N_G . By contrast, the error scales as the square root of N_G in the presence of Monte Carlo-based QEM, at the same computational cost as that of QEM based on exact channel inversion. Moreover, the error scaling of Monte Carlo-based QEM can be further improved with increased computational cost.

We have also demonstrated the analytical results both for low-complexity examples and for a more practical example of the quantum approximate optimization algorithm employed for multi-user detection in wireless communications. It may be an interesting future research direction to apply the results to other practical examples, or verify them using experimental approaches.

APPENDIX I PROOF OF PROPOSITION 1

Proof: First observe that the matrix representation of a perfect gate G_i as well as that of a channel C_i take the following block-diagonal form

$$G_i = \begin{pmatrix} 1 & \mathbf{0}^T \\ \mathbf{0} & U_i \end{pmatrix}, \quad C_i = \begin{pmatrix} 1 & \mathbf{0}^T \\ \mathbf{0} & D_i \end{pmatrix}, \quad (51)$$

where U_i is a unitary matrix, whereas D_i is a diagonal matrix having diagonal entries taking values in the interval $[0, 1]$. Since the matrix R_{N_G} is the product of several G_i and C_i , it becomes clear that its largest singular values satisfies $\sigma_1(R_{N_G}) = 1$, and its second largest singular value satisfies

$$\sigma_2(R_{N_G}) \leq \prod_{i=1}^{N_G} \|D_i\|_2. \quad (52)$$

Furthermore, we have

$$\left| r - \frac{1}{2^n} \text{Tr} \{ \mathcal{M}_{\text{ob}} \} \right| \leq \sigma_2(R_{N_G}) \quad (53)$$

due to the “bounded observable” Assumption 2.

Note that the quantity N_L defined in this proposition is related to the depth of the circuit. To elaborate, if we say that “a layer of gates” is executed if each qubit has been processed by at least one gate, then the entire circuit consists of at least N_L layers. For each single-qubit channel C in these layers, due to the assumption that each single-qubit Pauli error occurs at probability of at least ϵ_1 , the following bound holds:

$$C = I - 2 \text{diag} \{ [p_X + p_Z \ p_Y + p_Z \ p_X + p_Y] \} \preceq (1 - 4\epsilon_1)I, \quad (54)$$

where p_X , p_Y and p_Z are error probabilities corresponding to the Pauli-X, Y and Z errors, respectively. Thus we obtain

$$\begin{aligned} \sigma_2(R_{N_G}) &\leq (1 - 4\epsilon_1)^{N_L} \\ &= \exp\{N_L \ln(1 - 4\epsilon_1)\} \\ &\leq \exp(-4\epsilon_1 N_L). \end{aligned} \quad (55)$$

Hence the proof is completed. \square

APPENDIX II PROOF OF PROPOSITION 2

Proof: In this proof, we will work under the operator-sum representation of quantum channels. Since we consider Pauli channels, the recursion (3) may be rewritten as

$$\begin{aligned}\rho_k &= \sum_{i=1}^{4^n} [\mathbf{p}_k]_i \mathcal{S}_i \mathcal{G}_k \rho_{k-1} \mathcal{G}_k^\dagger \mathcal{S}_i \\ &= [\mathbf{p}_k]_1 \mathcal{G}_k \rho_{k-1} \mathcal{G}_k^\dagger + \sum_{i=2}^{4^n} [\mathbf{p}_k]_i \mathcal{S}_i \mathcal{G}_k \rho_{k-1} \mathcal{G}_k^\dagger \mathcal{S}_i.\end{aligned}\quad (56)$$

Assumption 2 implies that $\|\mathcal{M}_{\text{ob}}\|_2 \leq 1$, meaning that

$$\text{Tr}\{\mathcal{M}_{\text{ob}}\rho\} \leq 1$$

holds for any legitimate density matrix ρ . Note that terms such as $\mathcal{S}_i \mathcal{G}_k \rho_{k-1} \mathcal{G}_k^\dagger \mathcal{S}_i$ in (56) are indeed legitimate density matrices. Thus the computational result satisfies

$$\begin{aligned}|\tilde{r} - \text{Tr}\{\mathcal{M}_{\text{ob}}\rho_{N_G}\}| &\leq (|\tilde{r}| + \|\mathcal{M}_{\text{ob}}\|_2) \left(1 - \prod_{k=1}^{N_G} [\mathbf{p}_k]_1\right) \\ &\leq 2 \left(1 - \prod_{k=1}^{N_G} [\mathbf{p}_k]_1\right).\end{aligned}\quad (57)$$

According to Assumption 1, for any k , the vector \mathbf{p}_k satisfies

$$[\mathbf{p}_k]_1 \geq 1 - \epsilon_u, \quad \sum_{i=2}^{4^n} [\mathbf{p}_k]_i \leq \epsilon_u. \quad (58)$$

Therefore, we have

$$\begin{aligned}|\tilde{r} - \text{Tr}\{\mathcal{M}_{\text{ob}}\rho_{N_G}\}| &\leq 2(1 - (1 - \epsilon_u)^{N_G}) \\ &\leq 2\epsilon_u N_G.\end{aligned}\quad (59)$$

Hence the proof is completed. \square

APPENDIX III PROOF OF PROPOSITION 3

Proof: We first expand the expression of MSE as follows

$$\begin{aligned}\mathbb{E}\{(r - \tilde{r})^2\} &= \mathbb{E}\left\{\left(\mathbf{v}_{\text{ob}}^T \mathbf{v}_{N_G} - \tilde{r}\right)^2\right\} \\ &= \mathbf{v}_{\text{ob}}^T \mathbb{E}\{\mathbf{v}_{N_G} \mathbf{v}_{N_G}^T\} \mathbf{v}_{\text{ob}} + \tilde{r}^2 - 2\tilde{r} \mathbf{v}_{\text{ob}}^T \mathbb{E}\{\mathbf{v}_{N_G}\}.\end{aligned}\quad (60)$$

Hence the RMSE is given by

$$\sqrt{\mathbb{E}\{(r - \tilde{r})^2\}} = \sqrt{\mathbf{v}_{\text{ob}}^T \mathbf{A}_k \mathbf{v}_{\text{ob}} + \tilde{r}^2 - 2\tilde{r} \mathbf{v}_{\text{ob}}^T \boldsymbol{\mu}_k}, \quad (61)$$

where $\mathbf{A}_k := \mathbb{E}\{\mathbf{v}_k \mathbf{v}_k^T\}$ and $\boldsymbol{\mu}_k := \mathbb{E}\{\mathbf{v}_k\}$.

Using (40), we have

$$\mathbf{v}_k = \tilde{\mathbf{c}}_k \odot (\mathbf{G}_k \mathbf{v}_{k-1}). \quad (62)$$

This implies the following recursive relationships:

$$\mathbf{A}_k = \mathbb{E}\{\tilde{\mathbf{c}}_k \tilde{\mathbf{c}}_k^T\} \odot \mathbf{G}_k \mathbf{A}_{k-1} \mathbf{G}_k^T, \quad \boldsymbol{\mu}_k = \mathbf{G}_k \boldsymbol{\mu}_{k-1}. \quad (63)$$

In fact, we have

$$\mathbb{E}\{\tilde{\mathbf{c}}_k \tilde{\mathbf{c}}_k^T\} = \mathbf{1}\mathbf{1}^T + \boldsymbol{\Xi}_k, \quad (64)$$

which follows from (38). Substituting (64) into (63), we obtain

$$\mathbf{A}_k = \mathbf{G}_k \mathbf{A}_{k-1} \mathbf{G}_k^T + \boldsymbol{\Xi}_k \odot \mathbf{G}_k \mathbf{A}_{k-1} \mathbf{G}_k^T. \quad (65)$$

Let us now consider the case of $k = 1$. In this case, \mathbf{v}_0 of (8) is a deterministic vector, thus we have

$$\mathbf{A}_0 = \mathbf{v}_0 \mathbf{v}_0^T, \quad \boldsymbol{\mu}_0 = \mathbf{v}_0. \quad (66)$$

Using the recursive relationship of $\boldsymbol{\mu}_k = \mathbf{G}_k \boldsymbol{\mu}_{k-1}$, we now see that $\mathbf{v}_{\text{ob}}^T \boldsymbol{\mu}_{N_G} = \tilde{r}$. Hence we may simplify (60) as

$$\begin{aligned}\mathbb{E}\{(r - \tilde{r})^2\} &= \mathbf{v}_{\text{ob}}^T \mathbf{A}_{N_G} \mathbf{v}_{\text{ob}} + \tilde{r}^2 - 2\tilde{r} \mathbf{v}_{\text{ob}}^T \boldsymbol{\mu}_{N_G} \\ &= \mathbf{v}_{\text{ob}}^T \mathbf{A}_{N_G} \mathbf{v}_{\text{ob}} - (\mathbf{v}_{\text{ob}}^T \boldsymbol{\mu}_{N_G})^2 \\ &= \mathbf{v}_{\text{ob}}^T (\mathbf{A}_{N_G} - \boldsymbol{\mu}_{N_G} \boldsymbol{\mu}_{N_G}^T) \mathbf{v}_{\text{ob}}.\end{aligned}\quad (67)$$

Observe that the term $\mathbf{A}_{N_G} - \boldsymbol{\mu}_{N_G} \boldsymbol{\mu}_{N_G}^T$ is in fact the covariance matrix of \mathbf{v}_{N_G} , upon defining

$$\boldsymbol{\Sigma}_k := \mathbf{A}_k - \boldsymbol{\mu}_k \boldsymbol{\mu}_k^T, \quad (68)$$

and substituting into (67) we arrive at

$$\mathbb{E}\{(r - \tilde{r})^2\} = \mathbf{v}_{\text{ob}}^T \boldsymbol{\Sigma}_{N_G} \mathbf{v}_{\text{ob}}. \quad (69)$$

The covariance matrix can be further formulated as

$$\begin{aligned}\boldsymbol{\Sigma}_k &= \mathbf{A}_k - \boldsymbol{\mu}_k \boldsymbol{\mu}_k^T \\ &= \mathbf{A}_k - \tilde{\mathbf{R}}_k \mathbf{v}_0 \mathbf{v}_0^T \tilde{\mathbf{R}}_k^T.\end{aligned}\quad (70)$$

It now suffices to compute \mathbf{A}_k . Taking trace from both sides of (65), we have

$$\begin{aligned}\text{Tr}\{\mathbf{A}_k\} &= \text{Tr}\{\mathbf{G}_k \mathbf{A}_{k-1} \mathbf{G}_k^T + \boldsymbol{\Xi}_k \odot \mathbf{G}_k \mathbf{A}_{k-1} \mathbf{G}_k^T\} \\ &= \text{Tr}\{\mathbf{A}_{k-1}\} + \text{Tr}\{\boldsymbol{\Xi}_k \odot \mathbf{G}_k \mathbf{A}_{k-1} \mathbf{G}_k^T\} \\ &\leq \text{Tr}\{\mathbf{A}_{k-1}\} (1 + \|\boldsymbol{\Xi}_k\|_{\max}),\end{aligned}\quad (71)$$

where the second line follows from the fact that unitary transformations preserve the trace, and $\|\cdot\|_{\max}$ is defined as

$$\|\mathbf{A}\|_{\max} := \max_{i,j} |[\mathbf{A}]_{i,j}|.$$

From (38) we obtain

$$\begin{aligned}\|\boldsymbol{\Xi}_k\|_{\max} &= \frac{1}{N_s} \|\widetilde{\mathbf{H}} (\mathbf{P}_k - \mathbf{p}_k \mathbf{p}_k^T) \widetilde{\mathbf{H}} \odot \mathbf{c}_k \mathbf{c}_k^T\|_{\max} \\ &\leq \frac{1}{N_s} \|\widetilde{\mathbf{H}} (\mathbf{P}_k - \mathbf{p}_k \mathbf{p}_k^T) \widetilde{\mathbf{H}}\|_{\max} \|\mathbf{c}_k \mathbf{c}_k^T\|_{\max} \\ &\leq \frac{1}{N_s} \left(\|\widetilde{\mathbf{H}} \mathbf{P}_k \widetilde{\mathbf{H}}\|_{\max} + \|\widetilde{\mathbf{H}} \mathbf{p}_k \mathbf{p}_k^T \widetilde{\mathbf{H}}\|_{\max} \right),\end{aligned}\quad (72)$$

where the third line follows from the fact that \mathbf{c}_k represents a contractive transformation, so that $\mathbf{c}_k \preceq \mathbf{1}$. Note that every entry in $\widetilde{\mathbf{H}}$ has an absolute value of 1, and hence

$$\|\widetilde{\mathbf{H}} \mathbf{P}_k \widetilde{\mathbf{H}}\|_{\max} \leq \|\text{vec}\{\mathbf{P}_k\}\|_1 = 1, \quad (73)$$

and

$$\|\widetilde{\mathbf{H}} \mathbf{p}_k \mathbf{p}_k^T \widetilde{\mathbf{H}}\|_{\max} \leq \|\text{vec}\{\mathbf{p}_k \mathbf{p}_k^T\}\|_1 = 1. \quad (74)$$

In light of this, the upper bound of $\text{Tr}\{\mathbf{A}_k\}$ can now be simplified as follows:

$$\text{Tr}\{\mathbf{A}_k\} \leq \text{Tr}\{\mathbf{A}_{k-1}\} \left(1 + \frac{2}{N_s}\right). \quad (75)$$

From (66) we have $\text{Tr}\{\mathbf{A}_0\} = 1$ since \mathbf{v}_0 is a unit vector, hence we obtain

$$\begin{aligned} \text{Tr}\{\mathbf{A}_{N_G}\} &\leq \prod_{k=1}^{N_G} \left(1 + \frac{2}{N_s}\right) \\ &\leq \exp(2N_G N_s^{-1}). \end{aligned} \quad (76)$$

Using (70), we have

$$\begin{aligned} \text{Tr}\{\mathbf{\Sigma}_{N_G}\} &= \text{Tr}\{\mathbf{A}_{N_G}\} - \text{Tr}\{\mathbf{v}_0 \mathbf{v}_0^T\} \\ &\leq \exp(2N_G N_s^{-1}) - 1. \end{aligned} \quad (77)$$

Note that $\mathbf{\Sigma}_{N_G}$ is a positive semidefinite matrix, hence we have

$$\text{Tr}\{\mathbf{\Sigma}_{N_G}\} \geq \lambda_{\max}(\mathbf{\Sigma}_{N_G}), \quad (78)$$

where $\lambda_{\max}(\cdot)$ denotes the maximum eigenvalue of a matrix. This implies that

$$\begin{aligned} \sqrt{\mathbb{E}\{(r - \tilde{r})^2\}} &= \sqrt{\mathbf{v}_{\text{ob}}^T \mathbf{\Sigma}_{N_G} \mathbf{v}_{\text{ob}}} \\ &\leq \sqrt{\text{Tr}\{\mathbf{\Sigma}_{N_G}\} \cdot \|\mathbf{v}_{\text{ob}}\|} \\ &\leq \sqrt{\exp(2N_G N_s^{-1}) - 1} \cdot \|\mathbf{v}_{\text{ob}}\|. \end{aligned} \quad (79)$$

Hence the proof is completed by applying (23). \square

APPENDIX IV PROOF OF PROPOSITION 4

Proof: We start the proof from revisiting the inequality in (72), and arrive at:

$$\begin{aligned} \|\mathbf{\Xi}_k\|_{\max} &\leq \frac{1}{N_s} \|\widetilde{\mathbf{H}}(\mathbf{P}_k - \mathbf{p}_k \mathbf{p}_k^T) \widetilde{\mathbf{H}}\|_{\max} \\ &\leq \frac{1}{N_s} \|\text{vec}\{\mathbf{P}_k - \mathbf{p}_k \mathbf{p}_k^T\}\|_1. \end{aligned} \quad (80)$$

Next we construct an upper bound for the term $\|\text{vec}\{\mathbf{P}_k - \mathbf{p}_k \mathbf{p}_k^T\}\|_1$. According to the sampling overhead of QEM in [22], Assumption 1 implies that

$$\|\alpha\|_1 \leq \sqrt{1 + \sigma_u}. \quad (81)$$

Since $\alpha_k^{(1)} \geq 1$, we have

$$\sum_{i \neq 1} |\alpha_k^{(i)}| \leq \sqrt{1 + \sigma_u} - 1. \quad (82)$$

This further implies that

$$p_k^{(1)} \geq \frac{1}{\sqrt{1 + \sigma_u}}, \quad \sum_{i \neq 1} p_k^{(i)} \leq \sqrt{1 + \sigma_u} - 1. \quad (83)$$

Therefore, upon taking the entry-wise absolute value, we obtain

$$|\mathbf{P}_k - \mathbf{p}_k \mathbf{p}_k^T| \leq \begin{pmatrix} \frac{\sigma_u}{1 + \sigma_u} & p_k^{(2)} & \cdots & p_k^{(4^n)} \\ p_k^{(2)} & p_k^{(2)} & p_k^{(2)} p_k^{(3)} & \cdots \\ \vdots & p_k^{(3)} p_k^{(2)} & \ddots & \vdots \\ p_k^{(4^n)} & \vdots & \cdots & p_k^{(4^n)} \end{pmatrix}. \quad (84)$$

Here, the symbol “ \leq ” stands for entry-wise “not larger than”. Observe that summing up the first row, the first column and the main diagonal, by applying (83), we see that

$$\begin{aligned} \|\text{vec}\{\mathbf{P}_k - \mathbf{p}_k \mathbf{p}_k^T\}\|_1 &\leq 3(\sqrt{1 + \sigma_u} - 1) \\ &\quad + \frac{\sigma_u}{1 + \sigma_u} + \|\text{vec}\{\mathbf{q}_k \mathbf{q}_k^T\}\|_1, \end{aligned} \quad (85)$$

where $\mathbf{q}_k := [p_k^{(2)} \dots p_k^{(4^n)}]^T \in \mathbb{R}^{4^n - 1}$. Note that

$$\begin{aligned} \|\text{vec}\{\mathbf{q}_k \mathbf{q}_k^T\}\|_1 &= \mathbf{1}^T \mathbf{q}_k \mathbf{q}_k^T \mathbf{1} \\ &\leq (\sqrt{1 + \sigma_u} - 1)^2, \end{aligned} \quad (86)$$

implying that

$$\|\text{vec}\{\mathbf{P}_k - \mathbf{p}_k \mathbf{p}_k^T\}\|_1 \leq \frac{5}{2} \sigma_u + \frac{1}{4} \sigma_u^2, \quad (87)$$

which follows from that fact that

$$\sqrt{1 + x} - 1 \leq \frac{x}{2}$$

holds for all $x \geq 0$. Hence we have

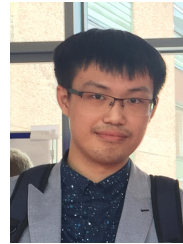
$$\|\mathbf{\Xi}_k\|_{\max} \leq \frac{1}{N_s} \left(\frac{5}{2} \sigma_u + \frac{1}{4} \sigma_u^2 \right), \quad (88)$$

which proves (42). Thus the proof is completed. \square

REFERENCES

- [1] J. Preskill, “Quantum Computing in the NISQ era and beyond,” *Quantum*, vol. 2, pp. 1–21, Aug. 2018.
- [2] F. Arute, K. Arya, R. Babbush *et al.*, “Quantum supremacy using a programmable superconducting processor,” *Nature*, vol. 574, no. 7779, pp. 505–510, Oct. 2019.
- [3] A. R. Calderbank, E. M. Rains, P. M. Shor, and N. J. A. Sloane, “Quantum error correction via codes over $\text{GF}(4)$,” *IEEE Trans. Inf. Theory*, vol. 44, no. 4, pp. 1369–1387, Jul. 1998.
- [4] C. H. Bennett and P. W. Shor, “Quantum information theory,” *IEEE Trans. Inf. Theory*, vol. 44, no. 6, pp. 2724–2742, Oct. 1998.
- [5] Z. Babar, D. Chandra, H. V. Nguyen, P. Botsinis, D. Alanis, S. X. Ng, and L. Hanzo, “Duality of quantum and classical error correction codes: Design principles and examples,” *IEEE Commun. Surv. Tuts.*, vol. 21, no. 1, pp. 970–1010, 1st quart. 2019.
- [6] D. Chandra, Z. Babar, H. V. Nguyen, D. Alanis, P. Botsinis, S. X. Ng, and L. Hanzo, “Quantum topological error correction codes: The classical-to-quantum isomorphism perspective,” *IEEE Access*, vol. 6, pp. 13 729–13 757, 2018.
- [7] E. Knill, R. Laflamme, and W. H. Zurek, “Resilient quantum computation: error models and thresholds,” *Proc. Roy. Soc. London A, Math. Phys. Eng. Sci.*, vol. 454, no. 1969, p. 365–384, Jan. 1998.
- [8] D. Aharonov and M. Ben-Or, “Fault-tolerant quantum computation with constant error rate,” *SIAM J. Comput.*, vol. 38, no. 4, p. 1207–1282, Jul. 2008.
- [9] D. Gottesman, “Theory of fault-tolerant quantum computation,” *Phys. Rev. A*, vol. 57, no. 1, p. 127, 1998.
- [10] P. W. Shor, “Algorithms for quantum computation: Discrete logarithms and factoring,” in *Proc. 35th Annual Symp. Foundations of Computer Science*, Santa Fe, New Mexico, USA, Nov. 1994, pp. 124–134.
- [11] L. K. Grover, “A fast quantum mechanical algorithm for database search,” in *Proc. 28th Annual ACM Symp. Theory of Computing*, Philadelphia, Pennsylvania, USA, May 1996, pp. 212–219.
- [12] P. J. Love, J. L. O’Brien, A. Aspuru-Guzik, A. Peruzzo, M.-h. Yung, X.-Q. Zhou, P. Shadbolt, and J. McClean, “A variational eigenvalue solver on a photonic quantum processor,” *Nature Commun.*, vol. 5, no. 1, pp. 1–7, Jul. 2014.
- [13] J. R. McClean, J. Romero, R. Babbush, and A. Aspuru-Guzik, “The theory of variational hybrid quantum-classical algorithms,” *New Journal of Physics*, vol. 18, no. 2, pp. 1–22, Feb. 2016.
- [14] E. Farhi, J. Goldstone, and S. Gutmann, “A quantum approximate optimization algorithm,” *arXiv preprint*, 2014. [Online]. Available: <https://arxiv.org/abs/arXiv:1411.4028>

- [15] X. Xu, J. Sun, S. Endo, Y. Li, S. C. Benjamin, and X. Yuan, "Variational algorithms for linear algebra," *arXiv preprint*, 2019. [Online]. Available: <https://arxiv.org/abs/1909.03898>
- [16] R. Sweke, F. Wilde, J. J. Meyer, M. Schuld, P. K. Fährmann, B. Meynard-Piganeau, and J. Eisert, "Stochastic gradient descent for hybrid quantum-classical optimization," *Quantum*, vol. 4, p. 314, 2020.
- [17] S. Hadfield, Z. Wang, B. O'Gorman, E. Rieffel, D. Venturelli, and R. Biswas, "From the quantum approximate optimization algorithm to a quantum alternating operator ansatz," *Algorithms*, vol. 12, no. 2, pp. 1–45, Feb. 2019.
- [18] G. E. Crooks, "Performance of the quantum approximate optimization algorithm on the maximum cut problem," *arXiv preprint*, 2018. [Online]. Available: <https://arxiv.org/abs/1811.08419>
- [19] J. R. McClean, S. Boixo, V. N. Smelyanskiy, R. Babbush, and H. Neven, "Barren plateaus in quantum neural network training landscapes," *Nat. Commun.*, vol. 9, no. 1, pp. 1–6, Nov. 2018.
- [20] S. Wang, E. Fontana, M. Cerezo, K. Sharma, A. Sone, L. Cincio, and P. J. Coles, "Noise-induced barren plateaus in variational quantum algorithms," *arXiv preprint*, 2020. [Online]. Available: <https://arxiv.org/abs/2007.14384>
- [21] S. Endo, S. C. Benjamin, and Y. Li, "Practical quantum error mitigation for near-future applications," *Phys. Rev. X*, vol. 8, no. 3, pp. 1–21, Jul. 2018.
- [22] Y. Xiong, D. Chandra, S. X. Ng, and L. Hanzo, "Sampling overhead analysis of quantum error mitigation: Uncoded vs. coded systems," *IEEE Access*, vol. 8, pp. 228 967–228 991, Dec. 2020.
- [23] R. Takagi, "Optimal resource cost for error mitigation," *Phys. Rev. Research*, vol. 3, p. 033178, Aug. 2021. [Online]. Available: <https://link.aps.org/doi/10.1103/PhysRevResearch.3.033178>
- [24] N. Moll, P. Barkoutsos, L. S. Bishop, J. M. Chow, A. Cross, D. J. Egger, S. Filipp, A. Fuhrer, J. M. Gambetta, M. Ganzhorn *et al.*, "Quantum optimization using variational algorithms on near-term quantum devices," *Quantum Science and Technology*, vol. 3, no. 3, 2018.
- [25] T. Matsumine, T. Koike-Akino, and Y. Wang, "Channel decoding with quantum approximate optimization algorithm," in *Proc. IEEE Int. Symp. Inf. Theory (ISIT)*, Paris, France, Jul. 2019, pp. 2574–2578.
- [26] P. J. O'Malley, R. Babbush, I. D. Kivlichan, J. Romero, J. R. McClean, R. Barends, J. Kelly, P. Roushan, A. Tranter, N. Ding *et al.*, "Scalable quantum simulation of molecular energies," *Physical Review X*, vol. 6, no. 3, 2016.
- [27] K. Sharma, M. Cerezo, L. Cincio, and P. J. Coles, "Trainability of dissipative perceptron-based quantum neural networks," *arXiv preprint*, 2020. [Online]. Available: <https://arxiv.org/abs/2005.12458>
- [28] M. Cerezo, A. Sone, T. Volkoff, L. Cincio, and P. J. Coles, "Cost-function-dependent barren plateaus in shallow quantum neural networks," *arXiv preprint*, 2020. [Online]. Available: <https://arxiv.org/abs/2001.00550>
- [29] G. Verdon, M. Broughton, J. R. McClean, K. J. Sung, R. Babbush, Z. Jiang, H. Neven, and M. Mohseni, "Learning to learn with quantum neural networks via classical neural networks," *arXiv preprint*, 2019. [Online]. Available: <https://arxiv.org/abs/1907.05415>
- [30] A. Skolik, J. R. McClean, M. Mohseni, P. van der Smagt, and M. Leib, "Layerwise learning for quantum neural networks," *arXiv preprint*, 2020. [Online]. Available: <https://arxiv.org/abs/2006.14904>
- [31] K. Temme, S. Bravyi, and J. M. Gambetta, "Error mitigation for short-depth quantum circuits," *Phys. Rev. Lett.*, vol. 119, no. 18, pp. 1–5, Nov. 2017.
- [32] J. I. Colless, V. V. Ramasesh, D. Dahlen, M. S. Blok, M. E. Kimchi-Schwartz, J. R. McClean, J. Carter, W. A. de Jong, and I. Siddiqi, "Computation of molecular spectra on a quantum processor with an error-resilient algorithm," *Phys. Rev. X*, vol. 8, Feb. 2018.
- [33] J. R. McClean, Z. Jiang, N. C. Rubin, R. Babbush, and H. Neven, "Decoding quantum errors with subspace expansions," *Nat. Commun.*, vol. 11, no. 1, pp. 1–9, 2020.
- [34] X. Bonet-Monroig, R. Sagastizabal, M. Singh, and T. E. O'Brien, "Low-cost error mitigation by symmetry verification," *Phys. Rev. A*, vol. 98, Dec. 2018.
- [35] C. Song, J. Cui, H. Wang, J. Hao, H. Feng, and Y. Li, "Quantum computation with universal error mitigation on a superconducting quantum processor," *Science Advances*, vol. 5, no. 9, 2019.
- [36] M. A. Nielsen and I. L. Chuang, *Quantum Computation and Quantum Information*, 2nd ed. New York, NY, USA: Cambridge University Press, 2011.
- [37] J. M. Chow, J. M. Gambetta, A. D. Córcoles, S. T. Merkel, J. A. Smolin, C. Rigetti, S. Poletto, G. A. Keefe, M. B. Rothwell, J. R. Rozen, M. B. Ketchen, and M. Steffen, "Universal quantum gate set approaching fault-tolerant thresholds with superconducting qubits," *Phys. Rev. Lett.*, vol. 109, no. 6, pp. 1–5, Aug. 2012.
- [38] M. A. Cirone, A. Delgado, D. G. Fischer, M. Freyberger, H. Mack, and M. Mussinger, "Estimation of quantum channels with finite resources," *Quantum Information Processing*, vol. 1, no. 5, pp. 303–326, Oct. 2002.
- [39] D. Greenbaum, "Introduction to quantum gate set tomography," *arXiv preprint*, 2015. [Online]. Available: <https://arxiv.org/abs/1509.02921>
- [40] E. Farhi, J. Goldstone, S. Gutmann, J. Lapan, A. Lundgren, and D. Preda, "A quantum adiabatic evolution algorithm applied to random instances of an NP-complete problem," *Science*, vol. 292, no. 5516, pp. 472–475, 2001.
- [41] S. Verdu *et al.*, *Multiuser detection*. Cambridge university press, 1998.
- [42] P. Botsinis, S. X. Ng, and L. Hanzo, "Fixed-complexity quantum-assisted multi-user detection for CDMA and SDMA," *IEEE Trans. Commun.*, vol. 62, no. 3, pp. 990–1000, Mar. 2014.



Yifeng Xiong received his B.S. degree in information engineering, and the M.S. degree in information and communication engineering from Beijing Institute of Technology (BIT), Beijing, China, in 2015 and 2018, respectively. He is currently pursuing the PhD degree with Next-Generation Wireless within the School of Electronics and Computer Science, University of Southampton. His research interests include quantum computation, quantum information theory, graph signal processing, and statistical inference over networks.



Soon Xin Ng (S'99-M'03-SM'08) received the B.Eng. degree (First class) in electronic engineering and the Ph.D. degree in telecommunications from the University of Southampton, Southampton, U.K., in 1999 and 2002, respectively. From 2003 to 2006, he was a postdoctoral research fellow working on collaborative European research projects known as SCOUT, NEWCOM and PHOENIX. Since August 2006, he has been a member of academic staff in the School of Electronics and Computer Science, University of Southampton. He is involved in the

OPTIMIX and CONCERTO European projects as well as the IU-ATC and UC4G projects. He is currently an Associate Professor in telecommunications at the University of Southampton.

His research interests include adaptive coded modulation, coded modulation, channel coding, space-time coding, joint source and channel coding, iterative detection, OFDM, MIMO, cooperative communications, distributed coding, quantum error correction codes and joint wireless-and-optical-fibre communications. He has published over 200 papers and co-authored two John Wiley/IEEE Press books in this field. He is a Senior Member of the IEEE, a Chartered Engineer and a Fellow of the Higher Education Academy in the UK.



Lajos Hanzo (<http://www-mobile.ecs.soton.ac.uk>, https://en.wikipedia.org/wiki/Lajos_Hanzo)

(FIEEE'04) received his Master degree and Doctorate in 1976 and 1983, respectively from the Technical University (TU) of Budapest. He was also awarded the Doctor of Sciences (DSc) degree by the University of Southampton (2004) and Honorary Doctorates by the TU of Budapest (2009) and by the University of Edinburgh (2015). He is a Foreign Member of the Hungarian Academy of Sciences and a former Editor-in-Chief of the IEEE Press. He has served several terms as Governor of both IEEE ComSoc and of VTS. He has published 2000+ contributions at IEEE Xplore, 19 Wiley-IEEE Press books and has helped the fast-track career of 123 PhD students. Over 40 of them are Professors at various stages of their careers in academia and many of them are leading scientists in the wireless industry. He is also a Fellow of the Royal Academy of Engineering (FREng), of the IET and of EURASIP. He is the recipient of the 2022 Eric Sumner Field Award.