

Limits of fault tolerance on resource-constrained quantum circuits for classical problems

Uthirakalyani G. ^{1,*}, Anuj K. Nayak ^{2,*}, Avhishek Chatterjee ¹ and Lav R. Varshney ²

¹*Department of Electrical Engineering, Indian Institute of Technology, Madras, Chennai 600036, India*

²*Coordinated Science Laboratory, University of Illinois at Urbana-Champaign, Urbana, Illinois 61801, USA*



(Received 24 March 2023; accepted 31 October 2023; published 22 November 2023)

Existing lower bounds on redundancy in fault-tolerant quantum circuits are applicable when both the input and the intended output are quantum states. These bounds may not necessarily hold, however, when the intended outputs are classical bits, as in Shor's or Grover's algorithms. Here we show that, indeed, noise thresholds obtained from existing bounds do not apply to a simple fault-tolerant implementation of the quantum phase estimation algorithm where the output quantum state suffers from noise before the measurement. Then we obtain a nonasymptotic lower bound on the minimum required redundancy for fault-tolerant quantum circuits with classical inputs and outputs. Further, recent results show that, due to physical resource constraints in quantum circuits, increasing redundancy can increase noise, which in turn may render many fault-tolerance schemes useless. So it is of both practical and theoretical interest to characterize the effect of resource constraints on the fundamental limits of fault-tolerant quantum circuits. As an application of our lower bound, we characterize the fundamental limit of fault-tolerant quantum circuits with classical inputs and outputs under resource-constraint-induced noise models.

DOI: [10.1103/PhysRevA.108.052425](https://doi.org/10.1103/PhysRevA.108.052425)

I. INTRODUCTION

Advantages of quantum computing over classical computing [1,2], especially when demonstrated mathematically [3,4], have spurred considerable interest. However, noise in quantum circuits heavily restricts the class of problems that can be solved using quantum hardware. Indeed, the formal term *noisy intermediate scale quantum* (NISQ) has been introduced to describe the current era where quantum processors are noise limited [5].

To limit the corruption of quantum states due to noise, pursuing fault-tolerant quantum circuits has led to a large literature in quantum error correction. Early papers demonstrated one can achieve arbitrary computational accuracy when physical noise is below a certain threshold. Achievability of any desired fault tolerance required a polylogarithmic redundancy with respect to the size of the quantum circuit in these initial works [6–9]. More recent works extend such *threshold theorems* to require only a constant overhead [10,11], reminiscent of work in classical fault-tolerant computing [12,13].

In this direction, there are fundamental lower bounds on redundancy for arbitrarily accurate computation [14–18]. However, all of these lower bounds are for quantum input and output, rather than classical input and output which is common for a large class of algorithms, such as those due to Deutsch-Jozsa [4], Shor [19], and Grover [20]. The bounds in [15,16] can be extended to classical input and output, but under restrictive assumptions on the final measurement. Here, we demonstrate by example that such lower bounds in quantum fault tolerance are not applicable for shallow quantum

circuits with classical input and output, and prove a general alternate nonasymptotic bound.

The effects of noise on computational accuracy of quantum circuits are typically studied assuming the noise per physical qubit is constant with respect to the size of the circuit. Unfortunately, this is not true in many quantum devices today. Often due to limited physical resources such as energy [21], volume [22], or available bandwidth [23], they have physical noise levels that grow as the quantum computer grows [24]. Fellous-Asiani *et al.* [24] introduce physical models of such scale-dependent noise and also aim to extend threshold theorems to this setting. However, the characterization of computational error (per logical qubit error) is restricted to concatenated codes and does not apply to more general fault-tolerant schemes [10,11]. As our lower bound is nonasymptotic, it is also applicable to scale-dependent noise. Using our lower bound and tools from optimization theory, we characterize the limits of scale dependence on fault-tolerant quantum circuits with classical input and output, applicable to any fault-tolerance scheme.

The two motivations for the present paper are therefore to obtain lower bounds on the required redundancy of a quantum circuit for computation with classical input and output, and to investigate the effect of resource constraints (like energy or volume) on this bound.

The experimental finding that noise increases with more redundancy under resource constraints implies that simple per (logical) qubit redundancy cannot achieve arbitrary computational accuracy even if noise per physical qubit is below the fault-tolerance threshold, in contrast to conventional threshold theorems [24]. This is due to two opposing forces: improvement in accuracy due to increased redundancy and worse overall noise with redundancy due to scale

*These authors contributed equally to this work.

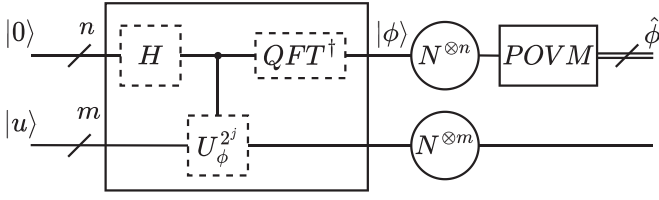


FIG. 1. Phase estimation algorithm in the presence of noise.

dependence. In this regard, we find the sweet spot on redundancy for a desired computational accuracy using techniques from resource-limited (finite blocklength) quantum information theory.

The remainder of the paper is organized as follows. Section II gives a counterexample to illustrate the need for a new redundancy lower bound. Section III gives mathematical models of computation, noise, and resource constraints that form the basis of our analysis. Then, the primary contributions follow. Section IV proves a nonasymptotic converse bound on redundancy required for classical computation on quantum circuits, drawing on one-shot capacity of classical-quantum channels (Theorem 2). Section V analyzes the limits of scale dependence for fault-tolerant computation, including closed-form and numerical solutions for some canonical noise models. Finally, Sec. VII concludes.

II. WHY IS A NEW BOUND NEEDED?

In this section, we shall demonstrate the need for a new redundancy lower bound for quantum circuits with classical inputs and outputs with the help of a simple noisy computational model of a quantum phase estimation circuit. The quantum phase estimation algorithm is crucial to many important problems like discrete log and factoring. It is used to estimate the n -bit phase ϕ (with $\phi_1\phi_2\ldots\phi_n$ as the binary representation) of the eigenvalue $e^{j2\pi\phi}$ of a unitary operator U_ϕ , given the corresponding eigenvector $|u\rangle$ and a controlled U_ϕ [25].

Consider a simple noisy computation model (in Fig. 1), where the qubits coming from a phase estimation circuit are corrupted independently by depolarizing (or erasure) noise before the final measurement (please refer to Appendix A for the definition of depolarizing and erasure channels).

Indeed, the noise model considered is a simplistic one, since it does not capture the noise in individual gates involved in quantum phase estimation. However, the existing bounds on quantum noise thresholds (e.g., [14]) for general purpose quantum computation are applicable to this model. In fact, the noisy computation model in Fig. 1 conforms to the model in [14] Sec. 1.1 with the entire quantum phase estimation circuit as a single layer quantum circuit followed by a layer of noise acting on each qubit independently. The sole purpose of this simplistic noise model and also of this section is to exhibit that the bounds for general purpose quantum computation are vacuous in the special but important case considered in this paper, i.e., classical inputs and outputs. We serve this purpose by showing that for a range of noise in this simplistic model where the existing bounds rule out any possibility of reasonably accurate computation, a simple fault-tolerant

scheme can achieve arbitrarily high accuracy. Note that the fault-tolerant scheme proposed here for this simplistic noise model may not be useful in other settings.

When the output of noiseless quantum phase estimation circuit $|\phi\rangle$ undergoes depolarizing noise, each qubit is replaced by a maximally mixed state with probability p , which results in each classical bit being flipped with probability $\frac{p}{2}$ independently upon measurement. In the case of erasure noise, each qubit (before measurement) flips to an orthogonal state $|e\rangle$ with probability p . Now, consider the following simple modification.

(1) Run quantum phase estimation algorithm, shown in Fig. 1, T times.

(2) Perform T independent measurements on each of the n noisy output qubits $\mathcal{N}^{\otimes n}(|\phi\rangle\langle\phi|)$. For erasure noise, declare the i th bit as 0 (or 1) if the measurement outcomes for the corresponding qubit were $\hat{\phi}_i = 0$ (or 1) at least once in T runs, and for the depolarizing channel declare the i th bit as 0 (or 1) if the measurement outcomes for the corresponding qubit were $\hat{\phi}_i = 0$ (or 1) in more than $T/2$ runs.

The above scheme can be seen as a fault-tolerant scheme against the noise in Fig. 1 with redundancy of size T , i.e., using a larger circuit with the algorithm repeated at most T times. We have the following guarantee on the performance of this simple fault-tolerant scheme for any $p < 1$.

Theorem 1. For a quantum phase estimation circuit that is corrupted by noise before the final measurement, (1) for depolarizing noise with probability p , if we choose $T \geq \frac{2\ln(n/\epsilon)}{(1-p)^2}$, and (2) for erasure noise with probability p , if we choose $T \geq \lceil \frac{\ln(n/\epsilon)}{\ln p} \rceil$, then the modified circuit discussed above correctly outputs ϕ with probability at least $1 - \epsilon$.

Proof. (1) *Depolarizing noise:* When a qubit $|\phi_i\rangle \in \{|0\rangle, |1\rangle\}$ is subject to a depolarizing noise with parameter p , then probability of error in detecting ϕ_i (in a single run) is

$$\mathbb{P}\{\phi_i \neq \hat{\phi}_i\} = \text{Tr}(\mathcal{N}(|\phi_i\rangle\langle\phi_i|)|\phi_i\rangle\langle\phi_i|) = \frac{p}{2},$$

where $\hat{\phi}_i$ is the outcome of measurement in the computational basis, and $\text{Tr}(\cdot)$ is the trace operation. Now, the probability of error in estimating the n -bit phase (after T runs), denoted by P_e , is given by

$$P_e = \mathbb{P}\{\cup_{i=1}^n \text{Error in } i\text{th bit}\},$$

$$\leq n \mathbb{P}\{\text{Error in one bit}\}.$$

The last inequality is due to the symmetry of noise across qubits and union bound. Since we run the algorithm T times and assign the majority to be the estimate,

$$\mathbb{P}\{\text{Error in one bit}\} = \mathbb{P}\{\phi_i \neq \hat{\phi}_i \text{ in } \geq T/2 \text{ runs}\},$$

which is the tail of the binomial distribution with probability $\frac{p}{2}$. Applying Hoeffding's inequality the probability of error is bounded as

$$\mathbb{P}\{\text{Error in one bit}\} \leq \exp\left(-2\left(\frac{1-p}{2}\right)^2 T\right),$$

$$P_e \leq n \exp\left(-2\left(\frac{1-p}{2}\right)^2 T\right) \leq \epsilon.$$

Choosing the number of runs to be at least

$$T \geq \frac{2 \ln \frac{n}{\epsilon}}{(1-p)^2}$$

yields $P_e \leq \epsilon$ for any $p \in [0, 1)$.

(2) *Erasure noise*: For erasure noise with probability of erasure p and input $|\phi_i\rangle \in \{|0\rangle, |1\rangle\}$, the probability of error in detecting ϕ_i is

$$\mathbb{P}\{\phi_i \neq \hat{\phi}_i\} = \text{Tr}(\mathcal{N}(|\phi_i\rangle\langle\phi_i|)|\phi_i\rangle\langle\phi_i|) = p.$$

The probability of error in estimating the n -bit phase (after T runs) is bounded above as

$$P_e = \mathbb{P}\left\{\bigcup_{i=1}^n \text{Error in } i\text{th bit}\right\} \leq n\mathbb{P}\{\text{Error in one bit}\}.$$

For erasure noise, error in estimating a bit occurs when the corresponding qubit is erased in all T runs. Therefore,

$$\begin{aligned} \mathbb{P}\{\text{Error in one bit}\} &= \mathbb{P}\{|\phi_i\rangle \text{ flips to } |e\rangle \text{ in all } T \text{ runs}, |\phi_i\rangle = |1\rangle\} \\ &\quad + \mathbb{P}\{|\phi_i\rangle \text{ flips to } |e\rangle \text{ in all } T \text{ runs}, |\phi_i\rangle = |0\rangle\} \\ &= \mathbb{P}\{|\phi_i\rangle \text{ flips to } |e\rangle \text{ in all } T \text{ runs} \mid |\phi_i\rangle = |1\rangle\} = p^T. \end{aligned}$$

Therefore, the probability of error is bounded as

$$P_e \leq np^T.$$

Choosing the number of runs

$$T \geq \left\lceil \frac{\ln \frac{\epsilon}{n}}{\ln p} \right\rceil,$$

we can achieve $P_e \leq \epsilon$ for any $p \in [0, 1)$. \blacksquare

The well-known threshold theorems [14–18] imply that when the noise strength, p , is above a threshold, no fault-tolerant scheme with finite redundancy can compute a quantum state within reasonable accuracy. This is because the lower bound on redundancy is given by $n/Q(\mathcal{N})$, where n is the number of physical qubits and $Q(\mathcal{N})$ is the quantum capacity of channel \mathcal{N} [14]. For the shallow noisy quantum computational model for phase estimation discussed above, the best-known threshold for depolarizing [14,16,18] and erasure [14,18] noise are $\frac{1}{3}$ and $\frac{1}{2}$, respectively (since quantum capacity vanishes above this threshold). However, through a noisy quantum phase estimation example, we have shown that when the inputs or outputs are classical with the performance criterion as the probability of error, it is indeed possible to achieve an arbitrarily small probability of error. Thus, Theorem 1 shows that the known redundancy lower bounds do not hold for quantum computation with classical input and output. This highlights the need for a bound which holds for classical inputs or outputs. A similar argument can be developed for the Deutsch-Jozsa algorithm and other well-known algorithms like discrete logarithms in the presence of noise.

Note that these do not imply that prior bounds on redundancy are incorrect; the apparent contradiction is due to differences in the definition of accuracy. Prior works use a notion of distance (or similarity) between the output quantum states of noiseless and noisy circuits to quantify accuracy. This requirement is too stringent when input and output are

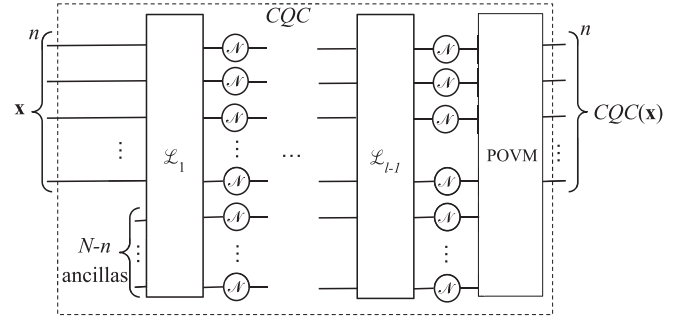


FIG. 2. CQC model of computation: classical input, quantum computation, and classical output.

classical bits and error probability is a more suitable performance criterion [26,27]. As such, we obtain a lower bound on the redundancy under the error probability criterion and then study the effect of resource constraints.

III. MODEL

In this section, we discuss the model of computation, the relevant accuracy criteria, and the noise model.

A. Model of computation

Consider the quantum circuit with classical inputs and classical outputs model in Fig. 2, which is a standard model for gate-based quantum computation. This is denoted by $CQC : \{0, 1\}^n \rightarrow \{0, 1\}^n$ or equivalently $CQC(\mathbf{x})$ for $\mathbf{x} \in \{0, 1\}^n$, where n is the input size. The goal of the circuit is to realize a function $f : \{0, 1\}^n \rightarrow \{0, 1\}^n$.

The circuit consists of l layers. The first layer takes n classical inputs (\mathbf{x}) as orthogonal quantum states $|0\rangle$ and $|1\rangle$ along with $N - n$ ancillas. It maps the input to a density operator of dimension 2^N . Any subsequent layer i , for $2 \leq i \leq l - 1$, takes the output of the previous layer (layer $i - 1$) as input. The output of any layer i , $1 \leq i \leq l - 1$, is a density operator of dimension 2^N . The final layer (layer l) performs a positive operator-valued measurement (POVM) and obtains classical output $CQC(\mathbf{x})$.

Each layer i , with $i \in \{1, 2, \dots, l - 1\}$ is a noisy quantum operation. This is modeled as a noiseless quantum operation \mathcal{L}_i on density operators of dimensions 2^N followed by N independent and identically distributed (i.i.d.) quantum channels \mathcal{N} (Fig. 2). Finally, the last layer, layer l , performs a measurement (POVM), which yields a classical output. Thus the quantum circuit can be represented as a composition of quantum operations as $CQC(\mathbf{x}) = \mathcal{L}_l \circ \mathcal{N}^{\otimes N} \circ \mathcal{L}_{l-1} \circ \dots \circ \mathcal{L}_2 \circ \mathcal{N}^{\otimes N} \circ \mathcal{L}_1(\mathbf{x})$, where \circ has the usual meaning of function composition. Next, we present the ϵ -accuracy criteria for computations with classical input and output, taken from the seminal work by von Neumann [26].

Definition 1 (ϵ -accuracy criteria). Suppose $f(\cdot)$ is a classical function realized by a quantum circuit $CQC(\cdot)$ as defined in Sec. III. Then the ϵ accuracy is

$$\mathbb{P}\{CQC(\mathbf{x}) \neq f(\mathbf{x})\} < \epsilon, \text{ for all } \mathbf{x} \in \{0, 1\}^n.$$

In oracle-based algorithms like phase estimation and Deutsch-Jozsa, there is no explicit input to the first layer,

rather the classical parameters of the oracle are implicit classical inputs to the circuit. The results presented in this paper are directly applicable to that scenario as well.

B. Noise model

Here, we consider only Holevo-additive channels characterized by a single parameter $p \in [0, 1]$ and whose Holevo capacity is monotonically decreasing in p . The channels we specifically study are (i) p -erasure, (ii) p -depolarizing, and (iii) symmetric generalized amplitude damping channels (GADCs), i.e., $\text{GADC}(p, \frac{1}{2})$.

In [24], it was shown that resource constraints can lead to an increase in noise with increase in redundancy, *scale-dependent noise*. A few models of scale-dependent noise, such as linear, polynomial, and exponential models, have been studied in [24]. Let $k \triangleq N/n \geq 1$ be the redundancy and $p(k)$ be the noise strength when the redundancy is k . In the polynomial model, $p(k) = \min(p_0[1 + \alpha(k-1)]^\gamma, 1)$, and in the exponential model $p(k) = \min(p_0 \exp(\alpha(k-1)^\gamma), 1)$ [24]. Here, $p_0 \in [0, 1]$ is the noise strength in the absence of any redundancy, i.e., $k = 1$, and α and γ are positive parameters.

Intuitively, p_0 is the original noise strength of the particular quantum technology. As redundancy increases, more and more physical qubits have to share the same resource, which leads to increased interactions with the environment and among each other. These undesired interactions result in increased noise, which is captured by the models proposed in [24]. In the interest of potentially wider applicability, we consider the following generic scale-dependent noise model, which includes the special cases discussed above.

Definition 2. Noise \mathcal{N}_p is parametrized by a single parameter $p \in [0, 1]$ and the Holevo information $\chi(\mathcal{N}_p)$ is nonincreasing in p . The parameter p is a function of redundancy k , given by $p(k) \triangleq \min(p(k; p_0, \theta), 1)$, where θ is a tuple of non-negative parameters, and (i) $p_0 = p(1; p_0, \theta)$ for all θ and, (ii) for any $k \geq 1$, $p(k; p_0, \theta)$ is nondecreasing in any component of θ and in p_0 , given the other parameters are fixed.

Here, p_0 represents the noise without redundancy, i.e., the initial noise without any resource constraint arising due to redundancy. Clearly, the polynomial and exponential models are special cases with $\theta = (\alpha, \gamma)$.

The threshold for p_0 , i.e., the minimum p_0 beyond which reliable quantum computation is not possible, was studied in [24] assuming concatenated codes for error correction. Here, we obtain a universal threshold for all fault-tolerance schemes.

IV. NONASYMPTOTIC LOWER BOUND ON REDUNDANCY

We obtain the lower bound by forming a mathematical relationship between the problem of ϵ -accurate classical computation on a noisy quantum circuit and the problem of classical communication over a finite number of i.i.d. uses of a quantum channel. Our approach builds on the following two simple observations.

First, a lower bound on redundancy obtained on a computational and noise model that have more capability would

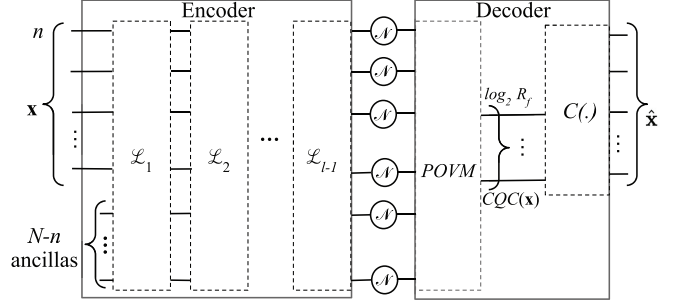


FIG. 3. Reduction of the noisy computation model in Fig. 2 to a noisy communication problem.

also be a lower bound for the original model. Second, a lower bound obtained under more relaxed accuracy criteria also applies to the original accuracy criteria.

Following the first observation, we obtain a redundancy bound for a model where there is no noise in layers $1, 2, \dots, l-1$, and noise only at the last layer. Following the second observation, we use the following relaxation of the ϵ -accuracy criteria.

Let the cardinality of the range of f be R_f , and $\mathbf{x}^{(1)}, \mathbf{x}^{(2)}, \dots, \mathbf{x}^{(R_f)} \in \{0, 1\}^n$ be such that $|\{f(\mathbf{x}^{(i)}) : 1 \leq i \leq R_f\}| = R_f$. Then, we use the following accuracy criterion: for all $i = 1, 2, \dots, R_f$

$$\mathbb{P}\{CQC(\mathbf{x}^{(i)}) \neq f(\mathbf{x}^{(i)})\} < \epsilon. \quad (1)$$

Clearly, this accuracy criterion is strictly weaker than the ϵ -accuracy criteria when f is not bijective. Further, suppose we have access to a hypothetically perfect (quantum or classical) circuit $C(\cdot)$ that can invert f on the set $\{f(\mathbf{x}^{(i)}) : 1 \leq i \leq R_f\}$, where f is bijective. Then, the relaxed accuracy criterion in (1) becomes equivalent to

$$\mathbb{P}\{C[CQC(\mathbf{x}^{(i)})] \neq \mathbf{x}^{(i)}\} < \epsilon, \text{ for all } i = 1, 2, \dots, R_f$$

for the circuit shown in Fig. 3. This relaxed criterion can then be seen as the criterion for classical communication of R_f messages with the maximum probability of error ϵ over a finite number of independent uses of a channel.

Using results from one-shot classical communication over a quantum channel ([28] Sec. 7.1.2) and using Holevo additivity, we obtain the following lower bound on redundancy.

Theorem 2. For ϵ -accurate computation of any n -bit Boolean function f using a quantum circuit in the presence of noise in Definition 2, the required redundancy is

$$k > \frac{c(\epsilon, R_f, n)}{\chi(\mathcal{N}_{p(k)})}, \quad (2)$$

where $c(\epsilon, R_f, n) \triangleq (1 - \epsilon) \frac{\log_2 R_f}{n} + \frac{\epsilon \log_2 \epsilon + (1 - \epsilon) \log_2 (1 - \epsilon)}{n}$, and $p(k) = \min(p(k; p_0, \theta), 1)$.

Proof. For any additive quantum channel \mathcal{N} , an upper bound for classical communication over a quantum channel using an $(\mathcal{M}, N, \epsilon)$ code is ([28] Sec. 7.1.2)

$$\log_2(|\mathcal{M}|) \leq \frac{\chi(\mathcal{N}^{\otimes N}) + h_2(\epsilon)}{1 - \epsilon},$$

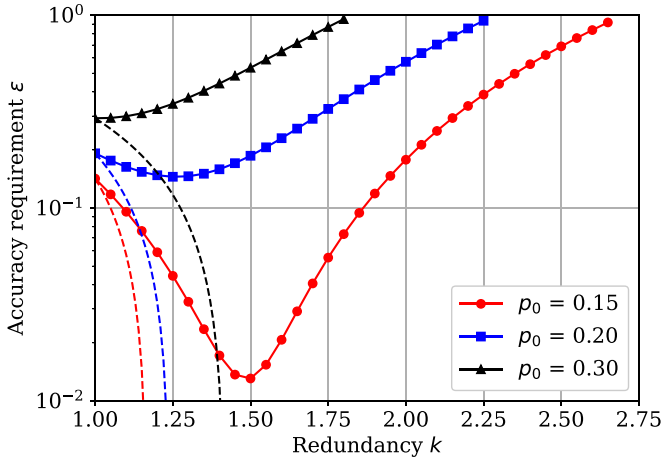


FIG. 4. Solid and dashed lines are for erasure noise with $p(k) = p_0$ and $p(k) = \min(p_0[1 + \alpha(k-1)]^\gamma, 1)$, respectively.

where \mathcal{M} is the message alphabet and $h_2(\cdot)$ is the binary entropy function. Assigning $|\mathcal{M}| = R_f$ yields

$$\begin{aligned} \epsilon > P_e &\geq 1 - \frac{\chi(\mathcal{N}^{\otimes N}) + h_2(P_e)}{\log_2 R_f} \\ &\geq 1 - \frac{\chi(\mathcal{N}^{\otimes N}) + h_2(\epsilon)}{\log_2 R_f}. \end{aligned}$$

The last inequality holds, since $h_2(\cdot)$ is increasing in $[0, \frac{1}{2}]$.

Rearranging, we obtain

$$\chi(\mathcal{N}^{\otimes N}) > (1 - \epsilon) \log_2 R_f - h_2(\epsilon).$$

Noting that Holevo information is additive,

$$\begin{aligned} N\chi(\mathcal{N}) &> (1 - \epsilon) \log_2 R_f - h_2(\epsilon), \\ N &> \frac{(1 - \epsilon) \log_2 R_f - h_2(\epsilon)}{\chi(\mathcal{N})}. \end{aligned}$$

Dividing both sides of the inequality by n , we obtain

$$k \triangleq \frac{N}{n} > \frac{(1 - \epsilon) \frac{\log_2 R_f}{n} - \frac{h_2(\epsilon)}{n}}{\chi(\mathcal{N}_{p(k)})}.$$

Unlike existing lower bounds on redundancy, this is a nonasymptotic bound and hence is applicable to quantum computers of any size (including the NISQ regime) and to any fault-tolerance scheme. A potential limitation is that our bound may be loose since we have reduced our noisy model to consist of only one layer of noise in our derivation.

Note that the above bound is applicable for any Holevo-additive noise with a single parameter p , even when p is a function of k . Thus, this bound can be used for understanding the limits of scale-dependent noise, as in the next section.

V. SCALE-DEPENDENT NOISE: NEW THRESHOLDS

The terms in (2) can be rearranged to obtain ϵ , a lower bound on the probability of error, over all fault-tolerant schemes, for different redundancy k . In Fig. 4, the thin dashed lines plot the same for erasure noise with $p(k) = p_0$. This shows that there may exist a fault-tolerant scheme that can

take the probability of error arbitrarily close to zero by increasing k .

In contrast, the curves are significantly different for scale-dependent noise, i.e., when p changes with k . In Fig. 4, we plot this for erasure $p(k) = \min(p_0[1 + \alpha(k-1)]^\gamma, 1)$. First, observe that there is a certain value of p_0 ($p_0 = 0.3$ in this case) beyond which an increase in redundancy actually hurts performance. Second, observe that for each p_0 there is a minimum ϵ below which no fault-tolerant scheme can reach.

These two observations lead to two interesting thresholds for initial noise p_0 . The first threshold is the value of p_0 beyond which redundancy only hurts accuracy. The second threshold of interest is p_0 beyond which the ϵ -accuracy criteria cannot be achieved using any fault-tolerance scheme for a given $\epsilon > 0$. Clearly, both these thresholds depend on the model of the scale-dependent noise and the parameters, e.g., α and γ therein. However, while the second threshold depends on ϵ , the first threshold does not.

The first threshold is similar to the threshold in [24], but notably, their threshold calculation is limited to concatenated codes, whereas the threshold obtained here applies to any fault-tolerance scheme. The threshold is obtained using a derivative-based approach similar to [24]. For brevity, we omit details to focus on the second threshold. Please note that since both the noise thresholds are obtained using a genie-aided redundancy lower bound in Theorem 2, they are only (upper) bounds of the actual noise threshold, and the bounds could potentially be loose.

For a given θ and $p(k; p_0, \theta)$, the second threshold $p_{\text{th}}(\theta, \epsilon)$ is the minimum p_0 beyond which ϵ -accurate computation is not possible. Thus, using Theorem 2, in particular, using (2), $p_{\text{th}}(\theta, \epsilon)$ is the minimum p_0 beyond which the criterion in Definition 1 is not satisfied by any $k \geq 1$. Thus, $p_{\text{th}}(\theta, \epsilon)$ is obtained by solving the following optimization problem:

$$\text{minimize } p_0 \quad \text{s.t.} \quad \min_{k \geq 1, 0 \leq p(k) \leq 1} g_\theta(k, p_0) \geq 0, \quad (3)$$

where $g_\theta(k, p_0) = \frac{c(\epsilon, R_f, n)}{k} - \chi(\mathcal{N}_{p(k)})$.

Consider the following optimization problem:

$$P_L : \quad \min_{k \geq 1, 0 \leq p(k) \leq 1} g_\theta(k, p_0).$$

Clearly, (3) has the optimization problem P_L , which we refer to as the lower-level optimization problem, as a constraint. Thus, (3) is a bilevel optimization problem. For a given set of θ the solution to P_L is a function of p_0 , which we denote as $g_\theta^*(p_0)$. Thus, the bilevel optimization problem in (3) can also be written as

$$\min p_0 \quad \text{s.t.} \quad g_\theta^*(p_0) \geq 0. \quad (4)$$

In general, to compute the threshold $p_{\text{th}}(\theta, \epsilon)$ one needs to solve (3). However, for erasure noise and some special classes of $p(k; p_0, \theta)$, $g_\theta(k, p_0)$ is convex in k for all p_0 . Hence, the minimum of the constraint in (3) for a given p_0 can be derived in closed form. Then, solving for the smallest p_0 for which that minimum is non-negative gives the following theorem.

Theorem 3. For erasure noise, thresholds for a fixed accuracy requirement $\epsilon > 0$ are as follows.

(1) If $p(k; p_0, \alpha) = p_0[1 + \alpha(k - 1)]$, then

$$p_{\text{th}}(\theta, \epsilon) = \begin{cases} 1 - c, & \text{if } \alpha \geq \frac{c}{1-c}, \text{ and} \\ \frac{(\sqrt{c\alpha} - \sqrt{c\alpha - \alpha + 1})^2}{(\alpha - 1)^2}, & \text{otherwise.} \end{cases}$$

(2) If $p(k; p_0, \gamma) = p_0 k^\gamma$, then

$$p_{\text{th}}(\theta, \epsilon) = \begin{cases} 1 - c, & \text{if } \gamma \geq \frac{c}{1-c}, \text{ and} \\ \left(\frac{\gamma}{\gamma+1}\right)^{\gamma+1}, & \text{otherwise.} \end{cases}$$

Here $c = c(\epsilon, R_f, n)$ as defined in Theorem 2.

Proof. Consider the following procedure to find a closed-form expression for $p_{\text{th}}(\theta, \epsilon)$.

(1) Minimize $g_\theta(k, p_0)$ over k . Since $p(k; p_0, \theta)$ is nondecreasing in k , it is enough to minimize $g_\theta(k, p_0)$ over $[1, k_{\max}]$, where $k_{\max} = \max\{k \mid p(k; p_0, \theta) \leq 1\}$. The minimum occurs at either $k = 1$, $k = k_{\max}$, or a stationary point of $g_\theta(k, p_0)$ in $(1, k_{\max})$.

(2) Substitute the minimizer k into $g_\theta(k, p_0) \geq 0$, which yields an equation in p_0, θ .

(3) Solving the equation for p_0 yields a closed-form expression for $p_{\text{th}}(\theta, \epsilon)$.

The derivation of $p_{\text{th}}(\theta, \epsilon)$ for corresponding $p(k; p_0, \theta)$ is given in Appendix B. ■

Depending on the scale-dependent noise model, the optimization problem in the constraint of (3) may or may not be convex. In both cases, we develop algorithms that can solve the problem for all noise models in Definition 2.

For a general $p(k; p_0, \theta)$, a closed-form expression for $p_{\text{th}}(\theta, \epsilon)$ in terms of θ cannot be obtained, and therefore, $p_{\text{th}}(\theta, \epsilon)$ must be computed numerically.

We develop Algorithm 1 to obtain $p_{\text{th}}(\theta, \epsilon)$ by solving bilevel optimization problem (3). In Algorithm 1, we solve the alternate formulation (4) using the bisection method, while assuming access to an oracle that computes $g_\theta^*(p_0)$ for any p_0 . Later, we also develop efficient algorithms that solve P_L and obtain $g_\theta^*(p_0)$ for any p_0 . The proof of convergence of Algorithm 1 is given in Appendix C.

Algorithm 1 computes the threshold $p_{\text{th}}(\theta, \epsilon)$ (up to an error of δ_{p_0}), for a predetermined set of θ (denoted by Θ of finite cardinality). Lines 4–17 describe the bisection method to compute $p_{\text{th}}(\theta, \epsilon)$. Depending on whether P_L is convex or nonconvex, Algorithm 1 or Algorithm 3 is used to compute $g_\theta^*(p_0)$, respectively.

Obtaining $g_\theta^*(\cdot)$ requires solving P_L . Next, we present efficient algorithms for solving P_L for erasure, depolarizing, and symmetric generalized amplitude damping channels (GADCs), and numerically obtain the converse surface for those noise models.

A. Threshold for erasure

In this section, we derive necessary conditions for ϵ -accurate computation when the source of corruption of quantum states is erasure. Substituting for the classical capacity of quantum erasure channel (QEC) from (A1) in (2) yields

$$g_\theta(k, p_0) = \frac{c(\epsilon, \eta, n)}{k} + p(k) - 1 < 0. \quad (5)$$

Algorithm 1: Algorithm to obtain $p_{\text{th}}(\theta, \epsilon)$ or $\bar{\Theta}_s$ numerically.

```

1 Initialize the set  $\Theta$ ,  $\max\_iters$ ,  $\delta_{p_0}$ ,  $\delta$ ,  $\bar{\Theta}_s \leftarrow \{\}$ ,
   $k \leftarrow 1$ ;
2 for each  $\theta \in \Theta$  do
3   Initialize  $i \leftarrow 0$ ,  $\Delta p_0 \leftarrow 1$ ,  $p_0 \leftarrow 0.5$ ,  $p_0^- \leftarrow 0$ ,
     $p_0^+ \leftarrow 1$ ;
    // Bisection Method
4   while  $\Delta p_0 > \delta_{p_0}$  and  $i < \max\_iters$  do
5      $p_{-1} \leftarrow p_0$ ;
6      $p_0 \leftarrow (p_0^- + p_0^+)/2$ ;
7      $k_{\max} \leftarrow 1 + \alpha^{-1}((p_0)^{-\frac{1}{\gamma}} - 1)$ ;
8     Solve  $P_L$ :  $g_\theta^*(p_0) = \min_{k \geq 1} g_\theta(k, p_0)$ ;
9     if  $g_\theta^*(p_0) > 0$  then
10       $p_0^+ \leftarrow p_0$ ;
11    end
12  else
13     $p_0^- \leftarrow p_0$ ;
14  end
15   $\Delta p_0 \leftarrow |p_0 - p_{-1}|$ ;
16   $i \leftarrow i + 1$ ;
17 end
18  $p_{th}(\theta, \epsilon) \leftarrow p_0$ .
19  $\bar{\Theta}_s \leftarrow \bar{\Theta}_s \cup \{(p_0, \alpha, \gamma)\}$ ;
20 end

```

Remark 1. One can equivalently solve P_L by restricting the range of k to $[1, k_{\max}]$, where $k_{\max} = \max\{k \mid p(k; p_0, \theta) \leq 1\}$. Also, k_{\max} is finite and hence $[1, k_{\max}]$ is compact, which makes it convenient to solve (3). Therefore, one can replace line 8 with $g_\theta^*(p_0) = \min_{k \in [1, k_{\max}]} g_\theta(k, p_0)$ to obtain the same value of threshold $p_{\text{th}}(\theta, \epsilon)$. Please refer to Appendix F1 for more details.

1. Physical noise $p(k; p_0, \theta)$ convex in redundancy k

For the erasure channel, if $p(k; p_0, \theta)$ is convex, then $g_\theta(k, p_0)$ is convex in $[1, k_{\max}]$, since the Holevo information $\chi(\mathcal{N}_p)$ is affine in p . Therefore, the problem P_L in (3) is convex, and from Remark 1 the feasible set is compact.

Algorithm 2: Projected gradient descent routine.

```

1 Function PROJGD( $g_\theta, p_0, k_{in}, k_{\max}, L, \zeta$ ):
2   Initialize  $j \leftarrow 0$ ,  $k_j \leftarrow k_{in}$ ,  $\xi = \frac{1}{L}$ ,  $\Delta g = 2\zeta$ ;
3   while  $\Delta g \geq \zeta$  do
4      $d_j \leftarrow |g'_\theta(k_j, p_0)|$ ;
5      $k_{j+1} \leftarrow \min\{k_{\max}, \max\{1, k_j + \xi d_j\}\}$ ;
6      $\tilde{g} \leftarrow g_\theta(k_{j+1}, p_0)$ ;
7      $\Delta g \leftarrow |g_\theta(k_j, p_0) - \tilde{g}|$ ;
8      $j \leftarrow j + 1$ ;
9   end
10  return  $\tilde{g}, k_j$ ;

```

Algorithm 3: Line search algorithm to find $\min_{k \geq 1} g_\theta(k, p_0)$, when $g_\theta(k, p_0)$ is nonconvex w.r.t. k .

```

1 Function LINESEARCH( $g_\theta, p_0, k_{\max}, L, \delta$ ):
2   Initialize  $i \leftarrow 0, k_i \leftarrow 1$ ;
3    $\tilde{g} \leftarrow \min_{k \in \{1, k_{\max}\}} g_\theta(k, p_0)$ ;
4   while  $i < \text{max\_iters}$  and  $k_i < k_{\max}$  do
5      $g_i, k_i^- \leftarrow \text{ProjGD}(g_\theta, p_0, k_i, k_{\max}, L, \delta)$ ;
6      $\hat{g}_i, \hat{k}_i, k_{i+1} \leftarrow \text{Perturb}(g_\theta, p_0, k_i^-, k_{\max}, \delta)$ ;
7      $\tilde{g} \leftarrow \min\{\tilde{g}, \hat{g}_i\}$ ;
8      $i \leftarrow i + 1$ ;
9   return  $\tilde{g}$ ;
10 Function PERTURB( $g_\theta, p_0, k, k_{\max}, L, \delta$ ):
11    $\Delta k \leftarrow \sqrt{\frac{2\delta}{L}}, \xi \leftarrow \frac{1}{L}, \hat{g} \leftarrow g_\theta(k, p_0), \hat{k} \leftarrow k$ ;
12    $k' \leftarrow \min\{k_{\max}, k + \xi |g'_\theta(k, p_0)|\}$ ;
13   while  $|g_\theta(k, p_0) - g_\theta(k', p_0)| < \delta$  and
14      $k < k_{\max}$  do
15      $k \leftarrow \min\{k_{\max}, k + \Delta k\}$ ;
16      $k' \leftarrow \min\{k_{\max}, k + \xi |g'_\theta(k, p_0)|\}$ ;
17      $\hat{g} \leftarrow \min\{\hat{g}, g_\theta(k, p_0)\}$ ;
18      $\hat{k} \leftarrow \arg\min_{k \in \{\hat{k}, k\}} \{g_\theta(k, p_0)\}$ ;
19   return  $\hat{g}, \hat{k}, k$ ;

```

A convex function over a compact set can be optimized using a gradient projection method given in [29]. There are many algorithms to solve general gradient projection problems such as sequential quadratic programming and augmented Lagrangian methods that can be directly applied to solve P_L . Since, our problem is a one-dimensional convex problem (with only a Lipschitz gradient constraint) over a finite range $[1, k_{\max}]$, we provide a simple constant step-size gradient projection algorithm (Algorithm 1). The algorithm starts from $k_{\text{in}} = 1$ when $g_\theta(\cdot, p_0)$ is convex [without loss of generality (w.l.o.g.)].

Algorithm 1 solves P_L optimally if step size (ξ) and stopping criterion (ζ) are chosen appropriately. Sufficient conditions for convergence are (1) $\xi \in (0, \frac{1}{L}]$, if $g'_\theta(k, p_0) \triangleq \frac{\partial}{\partial k} g_\theta(k, p_0)$ is L Lipschitz over $[1, k_{\max}]$, and (2) the stopping criterion provided in Definition 3. In all our computations, we choose $\xi = \frac{1}{L}$ as the step size for fast convergence.

Definition 3. Stopping criterion 1: Let $\{k_j\}$ be the iterates generated by the projected gradient descent algorithm (Algorithm 1); we use the following *stopping criterion* for the projected gradient descent algorithm:

$$|g_\theta(k_j, p_0) - g_\theta(k_{j+1}, p_0)| < \frac{\delta^2}{2Lk_{\max}^2} =: \zeta. \quad (6)$$

Then, it follows from the convexity and L -Lipschitz property of $g_\theta(\cdot, p_0)$ that stopping criterion (6) is a sufficient condition for convergence, which is $g_\theta(k_{j+1}, p_0) - g_\theta^*(p_0) \leq \delta$. The proof of convergence of Algorithm 1 and derivation of the Lipschitz constant for erasure noise are given in Appendix D.

2. Physical noise $p(k; p_0, \theta)$ nonconvex in redundancy k

Suppose $p(k)$ is nonconvex; then $g_\theta(k; \theta, \epsilon)$ is also nonconvex. Hence, the lower-level problem P_L cannot be solved using Algorithm 1 (PROJGD). Therefore, we provide a line-search algorithm (Algorithm 3) to compute solution for a nonconvex problem P_L .

In Algorithm 3, the compact set $[1, k_{\max}]$ is traversed by successive gradient descent (or ascent) and perturbation over a one-dimensional nonconvex function using an iterate starting from $k = 1$ (w.l.o.g.) and moving in the positive k direction. Lines 3–7 include one iteration of Algorithm 3, which contains calls to PROJGD and PERTURB as subroutines. The variable \tilde{g} keeps track of the minimum value of $g_\theta(\cdot, p_0)$ encountered thus far with an error of $\delta > 0$.

In Algorithm 3 we reuse the PROJGD routine for gradient ascent or descent but with a different (more relaxed) stopping criterion than in Definition 3.

Definition 4. Stopping criterion 2: Let $\{k_j\}$ be the iterates generated by the projected gradient descent algorithm (Algorithm 1). We use the following *stopping criterion* for the projected gradient descent algorithm:

$$|g_\theta(k_j, p_0) - g_\theta(k_{j+1}, p_0)| < \delta =: \zeta.$$

Definition 5. Stopping criterion for PERTURB: Let $\{k_j\}$ be a sequence generated by the PERTURB routine:

$$|g_\theta(k_j, p_0) - g_\theta(k'_j, p_0)| \geq \delta.$$

where $k'_j = \min\{k_{\max}, k + \xi |g'_\theta(k_j, p_0)|\}$ in line 14 of Algorithm 3, and $g'_\theta(z, p_0) = \frac{\partial g_\theta(k, p_0)}{\partial k} \big|_{k=z}$.

Note that the stopping criterion for PERTURB is similar to Definition 4, but with the inequality reversed. Since the stopping criteria of PROJGD and PERTURB are complementary, only one of the routines will be active during the execution of Algorithm 3. The proof of convergence of Algorithm 3 is given in Appendix E.

B. Threshold for the symmetric GADC and the depolarizing channel

1. Symmetric GADC

Let us compute converse regions when quantum states are corrupted by GADCs. We only consider the symmetric GADC (with $\mu = 1/2$), since its classical capacity is additive; for $\mu \neq 1/2$, the additivity of classical capacity is not known. Substituting the classical capacity of the symmetric GADC from (A3) in the necessary condition for ϵ accuracy in (2) yields

$$g_\theta(k, p_0) = \frac{c(\epsilon, \eta, n)}{k} - 1 + h_2\left(\frac{1 - \sqrt{1 - p(k)}}{2}\right) \leq 0. \quad (7)$$

In (7), the last term is monotonic (increasing) in p , and $p(k; p_0, \theta)$ is monotonic (increasing) in θ . Therefore, Corollary 1 also holds for the symmetric GADC. Therefore, for a given θ , the threshold $p_{\text{th}}(\theta, \epsilon)$ can be computed by solving bilevel optimization problem (3). However, we cannot obtain closed-form expressions like for the erasure channel due to the challenge from the binary entropy term in (7); therefore,

the threshold $p_{\text{th}}(\theta, \epsilon)$ must be computed numerically. Since the symmetric GADC is additive, and scale-dependent noise $p(k; p_0, \theta)$ is monotonic in θ componentwise, the threshold $p_{\text{th}}(\theta, \epsilon)$ can be computed using Algorithm 1 (Theorem 4 holds).

However, since Holevo information of the symmetric GADC is concave in p , even if $p(k; p_0, \theta)$ is convex in k , unlike the erasure case, $g_\theta(k, p_0)$ is not convex in k . Therefore, the lower-level problem P_L must be solved numerically using Algorithm 3 to obtain the threshold $p_{\text{th}}(\theta, \epsilon)$ for a given θ . For a polynomial noise model described in Sec. V, we can compute Lipschitz constant L in closed form for a given θ (refer to Appendix F2a for the derivation).

2. Depolarizing channel

In this section, we compute the converse region when computational states are corrupted by depolarizing noise. Substituting for the classical capacity of the depolarizing channel from (A3) in (2), we obtain

$$g_\theta(k, p_0) = h_2\left(\frac{p(k)}{2}\right) - 1 + \frac{c(\epsilon, \eta, n)}{k} \leq 0. \quad (8)$$

Similar to the symmetric GADC, the first term is increasing in p , and $p(k; p_0, \theta)$ is nondecreasing in θ . Therefore, Corollary 1 and computation of threshold $p_{\text{th}}(\theta, \epsilon)$ by solving bilevel optimization problem (3) also hold. Also, similar to the symmetric GADC, since obtaining closed-form expressions for $p_{\text{th}}(\theta, \epsilon)$ is not possible, it can be computed using Algorithm 1. Since $g_\theta(k, p_0)$ is nonconvex [due to $h_2(\cdot)$ in (8) being concave], the threshold $p_{\text{th}}(\theta, \epsilon)$ can be computed using line search (Algorithm 3). Again, similar to the symmetric GADC, Lipschitz constant L can be computed in closed form for a given θ (refer to Appendix F2b for the derivation).

VI. CONVERSE REGION

Beyond the notion of a scalar threshold $p_{\text{th}}(\theta, \epsilon)$, there exists a more general notion of the converse region, which extends the concept of threshold to multiple parameters simultaneously. We specifically aim to characterize the set of (p_0, θ) for which ϵ -accurate computation is not possible. The following corollary to Theorem 2 provides a converse in terms of (p_0, θ) .

Corollary 1. Suppose we have

$$\bar{\Theta} \triangleq \{(p_0, \theta) | \min_{k \geq 1} g_\theta(k, p_0) \geq 0\},$$

where

$$g_\theta(k, p_0) \triangleq \frac{c(\epsilon, R_f, n)}{k} - \chi(\mathcal{N}_{p(k)}).$$

Then ϵ -accurate computation is not possible for $(p_0, \theta) \in \bar{\Theta}$. Also, if $(p_0, \theta) \in \bar{\Theta}$ then $(p'_0, \theta') \in \bar{\Theta}$ if $(p'_0, \theta') \geq (p_0, \theta)$ in a componentwise sense.

Proof. From Definition 1, we must prove that if $P_e < \epsilon$, then $(p_0, \theta) \notin \bar{\Theta}$. From Theorem 2, we have if $P_e < \epsilon$, then

$$k > \frac{c(\epsilon, R_f, n)}{\chi(\mathcal{N}_{p(k)})},$$

$$g_\theta(k, p_0) = \frac{c(\epsilon, R_f, n)}{k} - \chi(\mathcal{N}_{p(k)}) < 0. \quad (9)$$

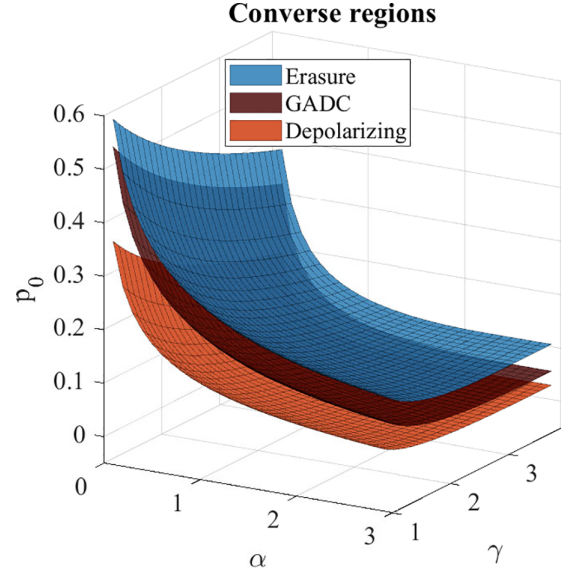


FIG. 5. Comparison of converse regions (surfaces) for erasure, depolarizing, and symmetric GADC with $\epsilon = 0.1$, $\log_2 R_f = n = 128$. The probability of error per physical qubit is assumed to scale with redundancy k as $p(k; p_0, \theta) = p_0[1 + \alpha(k - 1)]^\gamma$.

For any θ , (9) is satisfied only if

$$\min_{k \geq 1} g_\theta(k, p_0) < 0.$$

In other words, $(p_0, \theta) \notin \bar{\Theta}$.

As $\chi(\mathcal{N}_p)$ is nonincreasing in p and $p(k; p_0, \theta)$ is nondecreasing in each component, $(p'_0, \theta') \geq (p_0, \theta)$ in a componentwise sense implies $(p'_0, \theta') \in \bar{\Theta}$ whenever $(p_0, \theta) \in \bar{\Theta}$. ■

We refer to $\bar{\Theta}$ as the *converse region* since ϵ -accurate classical computation on quantum circuits is not possible if the parameters of the scale-dependent noise are in $\bar{\Theta}$. As any fault-tolerant implementation has to avoid this region, characterizing $\bar{\Theta}$ is of particular interest. By Corollary 1, for characterizing $\bar{\Theta}$, it is enough to find the minimum p_0 for each θ such that $(p_0, \theta) \in \bar{\Theta}$.

Figure 5 shows the converse regions when quantum computation is affected by erasure, depolarizing, and generalized amplitude damping noise. These are obtained by solving (3) for $\epsilon = 0.1$ using the aforementioned provably accurate algorithms. For a given $\theta = (\alpha, \gamma)$ the thresholds are related as $p_{\text{th}}(\theta, \epsilon)^{(e)} \geq p_{\text{th}}(\theta, \epsilon)^{(g)} \geq p_{\text{th}}(\theta, \epsilon)^{(d)}$ (pointwise), where the superscripts stand for the erasure channel, symmetric GADC, and depolarizing channel, respectively. This relation is expected since Holevo information of the channels is related for a given $p \in (0, 1)$ as $\chi^{(e)}(\mathcal{N}_p) \geq \chi^{(g)}(\mathcal{N}_p) \geq \chi^{(d)}(\mathcal{N}_p)$ (pointwise).

The converse region only indicates that achieving ϵ -accurate computation is not possible for the scaling parameters within that specific region. However, it does not imply that ϵ -accurate computation is automatically feasible outside this region. It is important to note that using the bound only shrinks the size of the converse region compared to the actual impossibility region. The actual impossibility region could potentially be larger and also subsume the converse region.

VII. CONCLUSION

We considered a model of quantum circuits where inputs and outputs are classical, which includes a large class of algorithms due to Deutsch-Jozsa, Grover, and Shor. Using the example of the phase estimation circuit, we demonstrated that the currently best-known redundancy lower bounds for quantum computation are not applicable for quantum circuits with classical input and output. Then, we considered the scenario where quantum states are corrupted by i.i.d. (Holevo) additive quantum noise. We established a connection between the problem of noisy computation and noisy classical communication over a quantum channel and obtained a nonasymptotic lower bound on redundancy.

Using this bound we studied fault-tolerant quantum computation under scale-dependent noise, where noise increases with added redundancy due to sharing of the limited physical resource. This led to two interesting thresholds on the original noise p_0 , beyond which (i) redundancy impacts only adversely and (ii) no fault-tolerant scheme can achieve ϵ accuracy for a given ϵ , respectively. The first threshold is similar to the one studied in [24] for concatenated codes; however, it is applicable to all fault-tolerant schemes. For the second threshold, we derived closed-form expressions whenever possible, and for other cases, we used optimization techniques for numerical characterizations.

In our derivation of the redundancy lower bound, certain relaxations were introduced to the noisy computation model. As a result, a potential limitation of our approach may be that the bound is not tight. In this paper, we have not explored the tightness of the bound, leaving it as a potential area for future research extensions. Future work could focus on either tightening the redundancy lower bound or establishing an achievability bound (redundancy upper bound) and characterizing the gap between the two, and its implications on the noise threshold.

ACKNOWLEDGMENTS

The authors are thankful to the two anonymous reviewers and the AE, Bill Munro, for many useful comments. A.K.N. and L.R.V. were supported in part by National Science Foundation Grant No. PHY-2112890. A.C. gratefully acknowledges support through Grants No. SERB/SRG/2019/001809, No. DST/INSPIRE/04/2016/001171 and NFIG IIT Madras. A.C. thanks Arul Lakshminarayan for useful feedback on the paper.

APPENDIX A: HOLEVO CAPACITIES OF THE ERASURE CHANNEL, DEPOLARIZING CHANNEL, AND GADC

Here, we consider only Holevo-additive channels characterized by a single parameter $p \in [0, 1]$ and whose Holevo capacity is monotonically decreasing in p . The candidate channels that we specifically study are (i) p -erasure, (ii) p -depolarizing, and (iii) symmetric generalized amplitude damping channels, i.e., GADC($p, \frac{1}{2}$).

1. Erasure channel

In a QEC, each qubit flips to $|e\rangle\langle e|$, which is orthogonal to every $\rho \in L(\mathbb{C}^d)$, with probability p . Therefore, whenever a

qubit gets corrupted, the location of corruption is known:

$$\mathcal{N}_p(\rho) = (1 - p)\rho + p\text{Tr}[\rho]|e\rangle\langle e|.$$

The classical capacity is [28]

$$\chi(\mathcal{N}_p) = 1 - p. \quad (\text{A1})$$

2. Depolarizing channel

When a qubit undergoes depolarizing noise, it is replaced by a maximally mixed state $I/2$ with probability p [28]:

$$\mathcal{N}_p(\rho) = (1 - p)\rho + \frac{p}{2}I.$$

In contrast to the erasure channel, the receiver (or the decoder) is not aware of the location of the error. The Holevo information of the depolarizing channel is

$$\chi(\mathcal{N}_p) = 1 - h_2\left(\frac{p}{2}\right), \quad (\text{A2})$$

where $h_2(\cdot)$ is the binary entropy function. Note that the Holevo information is similar to the capacity of a binary symmetric channel with crossover probability $p/2$.

3. GADC

Amplitude damping channels model the transformation of an excited atom to ground state by spontaneous emission of photons. The changes are expressed using $|0\rangle$ for the ground (no photon) state and $|1\rangle$ for the excited state. If the initial state of the environment $|0\rangle\langle 0|$ is replaced by the state $\theta_\mu \triangleq (1 - \mu)|0\rangle\langle 0| + \mu|1\rangle\langle 1|$, $\mu \in [0, 1]$ where μ is thermal noise, we get the generalized amplitude damping channel described using the following four Kraus operators [28]:

$$A_1 = \sqrt{1 - \mu} \begin{bmatrix} 1 & 0 \\ 0 & \sqrt{1 - p} \end{bmatrix}, \quad A_2 = \sqrt{1 - \mu} \begin{bmatrix} 0 & \sqrt{p} \\ 0 & 0 \end{bmatrix},$$

$$A_3 = \sqrt{\mu} \begin{bmatrix} \sqrt{1 - p} & 0 \\ 0 & 1 \end{bmatrix}, \quad A_4 = \sqrt{\mu} \begin{bmatrix} 0 & 0 \\ \sqrt{p} & 0 \end{bmatrix}.$$

The GADC is not additive in general (for arbitrary μ). However, in the special case of symmetric generalized amplitude damping, i.e., generalized amplitude damping with $\mu = 1/2$, it is a Holevo additive channel. The classical capacity of the symmetric GADC ($\mu = 1/2$) is [30]

$$\chi(\mathcal{N}_p) = 1 - h_2\left(\frac{1 - \sqrt{1 - p}}{2}\right), \quad (\text{A3})$$

where p is the probability that an atom decays from excited to ground state.

Remark 2. Note that we have used p to describe different impairments in different channels, so p must be interpreted appropriately based on context.

APPENDIX B: PROOF OF THEOREM 3

The Holevo capacity of the erasure channel is $\chi(\mathcal{N}_{p(k)}) = 1 - p(k)$. Therefore,

$$g_\theta(k, p_0) = \frac{c(\epsilon, R_f, n)}{k} + p(k) - 1. \quad (\text{B1})$$

If $\theta \in \bar{\Theta}$ (converse region), then from Theorem 1 the following holds:

$$g_\theta(k, p_0) = \frac{c(\epsilon, R_f, n)}{k} + p(k) - 1 \geq 0, \quad \forall k \geq 1. \quad (\text{B2})$$

Differentiating with respect to k and equating to zero (to find the stationary point),

$$g'_\theta(k, p_0) = -\frac{c(\epsilon, R_f, n)}{k^2} + p'(k; p_0, \theta) = 0. \quad (\text{B3})$$

Henceforth, we shall use $c = c(\epsilon, R_f, n)$ for brevity. For a fixed $\theta = \alpha$ and γ (respectively), the thresholds $p_{\text{th}}(\theta, \epsilon)$ are derived for some well-behaved $p(k; p_0, \theta)$ as follows.

(1) $p(k; p_0, \theta) = p_0[1 + \alpha(k - 1)]$: Suppose (B2) holds for some p_0 , then

$$\begin{aligned} g_\theta(k, p_0) &\geq 0, \quad \text{for } k = 1, \\ c + p_0 - 1 &\geq 0, \\ p_0 &\geq 1 - c. \end{aligned} \quad (\text{B4})$$

Note that $k_{\text{max}} = 1 + \alpha^{-1}[(p_0)^{-1} - 1]$, obtained by solving for k in $p(k; p_0, \theta) = 1$. Also since $p(\cdot; p_0, \theta)$ is linear, $g_\theta(\cdot, p_0)$ is convex in $[1, k_{\text{max}}]$. Therefore, $g_\theta(k, p_0)$ is minimized at any one of $k = 1$, $k = k_{\text{max}}$, or a stationary point in $(1, k_{\text{max}})$. Substituting $p'(k; p_0, \theta) = p_0\alpha$ in (B3), the stationary point is

$$k = \sqrt{\frac{c}{p_0\alpha}}. \quad (\text{B5})$$

(a) Note that for $k \in (1, k_{\text{max}})$ to be the minimum, $g'(k, p_0, \theta)|_{k=1} < 0$. Also, noting that $p_0 \geq 1 - c$ [from (B4)], we obtain

$$(1 - c)\alpha \leq p_0\alpha < c.$$

Therefore, $\alpha < \frac{c}{1-c}$. Substituting (B5) in (B2),

$$\begin{aligned} p_0 &\geq \frac{(\sqrt{c\alpha} - \sqrt{c\alpha - \alpha + 1})^2}{(\alpha - 1)^2}, \\ p_{\text{th}}(\theta, \epsilon) &= \frac{(\sqrt{c\alpha} - \sqrt{c\alpha - \alpha + 1})^2}{(\alpha - 1)^2}. \end{aligned}$$

Note that since $\alpha < \frac{c}{1-c}$, the second term in the numerator, $c\alpha - \alpha + 1 = 1 - c \geq 0$. Therefore, the threshold $p_{\text{th}}(\theta, \epsilon)$ exists.

(b) If $\alpha \geq \frac{c}{1-c}$, then

$$\begin{aligned} k &= \sqrt{\frac{c}{p_0\alpha}} \leq \sqrt{\frac{1-c}{p_0}}; k \geq 1, \\ p_0 &\leq 1 - c. \end{aligned}$$

However, $p_0 \geq 1 - c$ from (B4). Therefore, $p_{\text{th}}(\theta, \epsilon) = 1 - c$.

(2) $p(k; p_0, \theta) = p_0k^\gamma$: Here, $\theta = \gamma$. The value of k ranges from $1 \leq k \leq (\frac{1}{p_0})^{\frac{1}{\gamma}}$. For the choice of $p_{\text{th}}(\theta, \epsilon)$, (B2) must hold for all k in this range. Similar to the linear case, for this choice of $p(k)$ and range of k , (B2) is convex. Hence for

$$p_{\text{th}}(\theta, \epsilon),$$

$$\begin{aligned} g_\theta(k, p_0)|_{k=1} &\geq 0, \\ c + p_0 - 1 &\geq 0, \\ p_0 &\geq 1 - c. \end{aligned} \quad (\text{B6})$$

Substituting $p'(k; p_0, \theta) = p_0\gamma k^{\gamma-1}$ in (B3), we obtain the stationary point as

$$k = \left(\frac{c}{p_0\gamma}\right)^{\frac{1}{\gamma+1}}. \quad (\text{B7})$$

Similar to linear $p(k; p_0, \theta)$, there are two cases.

(a) If $\gamma < \frac{c}{1-c}$, substituting the stationary point computed in (B7) in (B2), the threshold $p_{\text{th}}(\theta, \epsilon)$ can be computed as

$$\begin{aligned} p_0 &\geq \frac{\left(\frac{\gamma}{c}\right)^\gamma}{(\gamma + 1)^{\gamma+1}}, \\ p_{\text{th}}(\theta, \epsilon) &= \frac{\left(\frac{\gamma}{c}\right)^\gamma}{(\gamma + 1)^{\gamma+1}}. \end{aligned}$$

(b) If $\gamma \geq \frac{c}{1-c}$, then

$$\begin{aligned} k &= \left(\frac{c}{p_0\gamma}\right)^{\frac{1}{\gamma+1}} \leq \left(\frac{1-c}{p_0}\right)^{\frac{1}{\gamma+1}}; k \geq 1, \\ p_0 &\leq 1 - c. \end{aligned}$$

However, $p_0 \geq 1 - c$ from (B6). Therefore, $p_{\text{th}}(\theta, \epsilon) = 1 - c$.

APPENDIX C: PROOF OF CONVERGENCE OF ALGORITHM 1

The following theorem provides a proof of global convergence of Algorithm 1, with only a monotonicity assumption in θ (note that continuity in θ is not needed).

Theorem 4. Suppose a quantum circuit is corrupted by a scale-dependent noise per physical qubit, $p(k; \theta)$, that is monotonic in θ . Then for any given θ , the sequence $\{p_{0_i}\}$ generated using Algorithm 1 converges to the threshold $p_{\text{th}}(\theta, \epsilon)$.

Proof. Algorithm 1 generates a nonincreasing sequence $\{p_{0_i}^+\}$ and a nondecreasing sequence $\{p_{0_i}^-\}$, which at every iteration yields $g^*(p_{0_i}^+) \geq 0$ and $g^*(p_{0_i}^-) < 0$, with $p_{0_i} = \frac{p_{0_i}^+ + p_{0_i}^-}{2}$. Since the bisection method halves the difference between $p_{0_i}^+$

and $p_{0_i}^-$ at every iteration (i.e., $p_{0_{i+1}}^+ - p_{0_{i+1}}^- = \frac{p_{0_i}^+ - p_{0_i}^-}{2}$), we have that for all $\epsilon > 0$, there exists an i_0 such that for all $i \geq i_0$, we get $p_{0_i}^+ - p_{0_i}^- < \epsilon$. Also, since both $\{p_{0_i}^+\}$ and $\{p_{0_i}^-\}$ are bounded, they converge, and since for all $i \geq i_0$, $p_{0_i}^+ - p_{0_i}^- < \epsilon$, they converge to a common limit point (say p^*). Due to the monotonicity of $g_\theta^*(p_0)$ (nondecreasing with p_0), the following inequality holds: $g_\theta^*(p_{0_i}^-) \leq g_\theta^*(p_0^*) \leq g_\theta^*(p_{0_i}^+)$. Therefore, $g_\theta^*(p_0) < 0$, for all $p_0 < p^*$, and $g_\theta^*(p_0) \geq 0$, for all $p_0 > p^*$, which is by definition $p^* = p_{\text{th}}(\theta, \epsilon)$. ■

APPENDIX D: PROOF OF CONVERGENCE OF ALGORITHM 1

The following theorem provides proof of convergence of Algorithm 1. For better readability, the associated lemmas used in the proof are included in Appendix F3.

Theorem 5. Convergence of Algorithm 1: Suppose $g_\theta^*(p_0) = \min_{k \geq 1} g_\theta(k, p_0)$, which is convex in k . Then Algorithm 1 yields \tilde{g} arbitrarily close to $g_\theta^*(p_0)$, i.e., for any predetermined $\delta > 0$, $|\tilde{g} - g_\theta^*(p_0)| \leq \delta$.

Proof. Let $\{1, \dots, k_l\}$ be a sequence generated by projected gradient descent, PROJGD, where k_l satisfies the stopping criterion. Note that PROJGD does not cross any stationary point if the step size $\xi \leq \frac{1}{L}$ (from Lemma 6). So, $k_l = 1$ if and only if $\tilde{g} = g_\theta^*(p_0) = g_\theta(1, p_0)$, and similarly $k_l = k_{\max}$ if and only if $\tilde{g} = g_\theta^*(p_0) = g_\theta(k_{\max}, p_0)$. Otherwise $k_l \in (1, k_{\max})$ and $g_\theta(1, p_0) < 0$, which implies from Lemma 6 that $g_\theta(k_l, p_0) \leq 0$. From Lemmas 4 and 5, k_l satisfying the stopping criterion in Definition 3 is sufficient for convergence, i.e., $\tilde{g} = g_\theta(k_l, p_0)$ and $|\tilde{g} - g_\theta^*(p_0)| \leq \delta$. ■

The following lemma shows $g'_\theta(k, p_0)$ is indeed L Lipschitz over $[1, k_{\max}]$ for a general polynomial noise model and gives a closed-form expression for L .

Lemma 1. Computing Lipschitz constant L : $g'_\theta(k, p_0)$ is L Lipschitz over $[1, k_{\max}]$ for scale-dependent erasure noise $p(k; p_0, \theta) = p_0[1 + \alpha(k-1)]^\gamma$ with $\gamma \geq 0$ where, for $c = c(\epsilon, \eta, n)$,

$$L = 2c + \alpha^2 \gamma |\gamma - 1| p_0^{\min\{1, \frac{2}{\gamma}\}}.$$

Proof. Let g'_θ and g''_θ denote the partial derivatives $\frac{\partial}{\partial k} g_\theta(k, p_0)$ and $\frac{\partial^2}{\partial k^2} g_\theta(k, p_0)$, respectively. The magnitude of the second-order partial derivative is bounded above as

$$|g''_\theta| \leq \sup_k \left| \frac{2c}{k^3} \right| + \sup_k |p''(k; p_0, \theta)|,$$

where the inequality follows from triangle inequality and by maximizing each summand. Observe that the first summand is maximized when $k = 1$, and the second term is bounded above as

$$p''(k; p_0, \theta) \leq \begin{cases} \alpha^2 \gamma |\gamma - 1| p_0, & 0 \leq \gamma < 2, k = 1, \text{ and} \\ \alpha^2 \gamma |\gamma - 1| p_0^{2/\gamma}, & \gamma \geq 2, k = k_{\max}, \end{cases}$$

$$\leq \alpha^2 \gamma |\gamma - 1| p_0^{\min\{1, \frac{2}{\gamma}\}}, \quad \gamma \geq 0,$$

where $k_{\max} = 1 + \alpha^{-1}(p_0^{-(1/\gamma)} - 1)$. Therefore,

$$g''_\theta \leq 2c + \alpha^2 \gamma |\gamma - 1| p_0^{\min\{1, \frac{2}{\gamma}\}} =: L.$$

■

APPENDIX E: PROOF OF CONVERGENCE OF ALGORITHM 3

Theorem 6 proves convergence of Algorithm 3. Required lemmas are in Sec. F4.

Theorem 6. Proof of convergence of Algorithm 3: Algorithm 3 yields \tilde{g} , which is arbitrarily close to $g^* = \min_{k \in [1, k_{\max}]} g_\theta(k, p_0)$, i.e., $|\tilde{g} - g^*| \leq \delta$, for a pre-determined $\delta > 0$.

Proof. Suppose $\{\dots, k_i, k_i^-, k_{i+1}^-, k_{i+1}, \dots\}$ is the sequence generated by Algorithm 3. From Lemma 6 there are no stationary points in (k_i, k_i^-) . Then, the PERTURB routine keeps track of the minimum value of $g_\theta(\cdot, p_0)$ in $[k_i^-, k_{i+1}]$ at discrete increments: $\hat{g}_i = \min_{k \in \{k_i^-, k_i^- + \Delta k, \dots, k_{i+1}\}} g_\theta(k, p_0)$. This is

followed by executing PROJGD again from k_{i+1} to k_{i+1}^- , and so on. In every call to the PERTURB routine, \tilde{g} tracks the minimum of \hat{g}_i until the i th iteration. From Lemma 8, \hat{g}_i differs from $\min_{k \in [k_i^-, k_{i+1}]} g_\theta(k, p_0)$ by at most δ . In line 2 of Algorithm 3, \tilde{g} is initialized with the minimum at boundary points $k = \{1, k_{\max}\}$. Therefore, $\tilde{g} - g^* \leq \delta$. Finally, from Corollary 3, Lemma 8, and Lemma 9, Algorithm 3 terminates in finite steps when $k_j = k_{\max}$ or $k_j^- = k_{\max}$ for some $j \geq i + 1$. ■

APPENDIX F: LEMMAS FOR CONVERGENCE OF ALGORITHMS

1. Restriction of the feasible set of P_L to $[1, k_{\max}]$

Let $k_{\max} = \max\{k \mid p(k; p_0, \theta) \leq 1\}$. If $k_{\max} = \infty$, then solving (3) yields $p_{\text{th}}(\theta, \epsilon) = 1$. Therefore, (3) is nontrivial only if k_{\max} is finite. Let $g_1(p_0) = \min_{k \geq 1} g_\theta(k, p_0)$ and $g_2(p_0) = \min_{k \in [1, k_{\max}]} g_\theta(k, p_0)$. From (5), it can be observed that $g_1(p_0) = 0$ whenever $g_2(p_0) > 0$, and $g_1(p_0) = g_2(p_0)$ whenever $g_2(p_0) \leq 0$. Hence, the threshold $p_{\text{th}}(\theta, \epsilon)$ obtained using $g_1(\cdot)$ and that obtained using $g_2(\cdot)$ as a solution to P_L in (3) are identical. Therefore, (3) can be equivalently solved by restricting the domain of $g_\theta(\cdot, p_0)$ in P_L to $[1, k_{\max}]$. In other words, one can replace line 8 with $g_\theta^*(p_0) = \min_{k \in [1, k_{\max}]} g_\theta(k, p_0)$ to obtain the same value of threshold $p_{\text{th}}(\theta, \epsilon)$. Additionally, this restriction makes the feasible set compact. Moreover, notice that the restriction and equivalence hold for all channels (not just erasure) as long as $\chi(\mathcal{N}_{p(k)}) = 0$ whenever $p(k) = 1$.

2. Derivation of Lipschitz constants for the symmetric GADC and depolarizing channel

a. Symmetric GADC

Lemma 2. Computing Lipschitz constant L : $g'_\theta(k, p_0)$ is L Lipschitz over $[1, k_{\max}]$ for a polynomial scale-dependent symmetric generalized amplitude damping noise $p(k; p_0, \theta) = p_0[1 + \alpha(k-1)]^\gamma$, where

$$L := 2c + \frac{\alpha^2 \gamma}{2 \ln 2} \left(|\gamma - 1| + \frac{\gamma}{3} \right).$$

Proof. Denote $q(p) = q[p(k)] = \frac{1 - \sqrt{1 - p(k)}}{2}$, and $p = p(k; p_0, \theta)$ (for brevity); the magnitude of the second-order derivative of $g_\theta(k, p_0)$ is bounded above as

$$|g''_\theta| \leq \sup_k \left| \frac{2c}{k^3} \right| + \sup_k |h''_2[q(k)]|$$

$$\leq 2c + \sup_k \{ |h''_2(q)q'(p)^2 p^2 + h'_2(q)q''(p)p^2| \} + \sup_k \{ |h'_2(q)q'(p)p''| \}.$$

The last two terms on the right-hand side of the inequality are maximized when $k = k_{\max}$. The second term and the third term are bounded above as

$$|h''_2(q)q'(p)^2 p^2 + h'_2(q)q''(p)p^2|$$

$$\leq \alpha^2 \gamma^2 \left| \frac{p/(1-p)}{4 \ln 2} \ln \left[\frac{1}{e} \left(\frac{1-q(p)}{q(p)} \right)^{\frac{p}{2\sqrt{1-p}}} \right] \right| \leq \frac{\alpha^2 \gamma^2}{6 \ln 2}$$

and

$$\begin{aligned} h'_2(q)q'(p)p'' &\leq \frac{\alpha^2\gamma|\gamma-1|}{4\sqrt{1-p}} \log_2 \left(\frac{1-q(p)}{q(p)} \right) \\ &\leq \frac{\alpha^2\gamma|\gamma-1|}{2\ln 2}, \end{aligned}$$

respectively. The last inequalities in both the terms are obtained by allowing $p \rightarrow 1$ (i.e., $k \rightarrow k_{\max}$). Therefore,

$$|g''_\theta| \leq 2c + \frac{\alpha^2\gamma}{2\ln 2} \left(|\gamma-1| + \frac{\gamma}{3} \right) =: L.$$

b. Depolarizing channel

Lemma 3. Computing Lipschitz constant L : $g'_\theta(k, p_0)$ is L Lipschitz over $[1, k_{\max}]$ for a polynomial scale-dependent depolarizing noise $p(k; p_0, \theta) = p_0[1 + \alpha(k-1)]^\gamma$, where

$$L = 2c + \frac{\alpha^2\gamma}{2\ln 2} \left[\frac{2\gamma}{p_0(2-p_0)} + |\gamma-1| \ln \left(\frac{2-p_0}{p_0} \right) \right].$$

Proof. The second-order derivative of $g_\theta(k, p_0)$ is bounded above as

$$\begin{aligned} |g''_\theta| &\leq \sup_k \left| \frac{2c}{k^3} \right| + \sup_k |h''_2(p(k)/2)|, \\ |g''_\theta| &\leq 2c + \sup_k \left\{ \frac{1}{4} |h''_2(z)| p'(k; p_0, \theta)^2 + \frac{1}{2} h'(z) p''(k; p_0, \theta) \right\}, \end{aligned}$$

where $z = p(k; p_0, \theta)/2$. Noting that the $|h''_2(z)|$ and $h'(z)$ are maximized when $p(k; p_0, \theta) = p_0$, we obtain

$$|g''_\theta| \leq 2c + \frac{\alpha^2\gamma}{2\ln 2} \left[\frac{2\gamma}{p_0(2-p_0)} + |\gamma-1| \ln \left(\frac{2-p_0}{p_0} \right) \right] =: L.$$

3. Lemmas: Projected gradient descent

Definition 6. In Appendices F3 and F4, we consider $g(\cdot)$ to be of the following form: $g: [1, k_{\max}] \rightarrow \mathbb{R} : k \mapsto g(k)$, where $g'(\cdot)$ is L Lipschitz.

Lemma 4. Stopping criterion and bounded gradient: Suppose a pair of iterates (k_j, k_{j+1}) , which lie in the interior $(1, k_{\max})$, generated by PROJGD satisfies the stopping criterion $|g(k_j) - g(k_{j+1})| < \frac{\delta^2}{2Lk_{\max}^2}$, then the first-order derivative is bounded above as $|g'(k_j)| < \frac{\delta}{k_{\max}}$.

Proof. Applying the descent lemma to k_j, k_{j+1} , we get

$$\begin{aligned} g(k_{j+1}, p_0) &\leq g(k_j, p_0) + g'(k_j, p_0)(k_{j+1} - k_j) \\ &\quad + \frac{1}{2}L|k_{j+1} - k_j|^2. \end{aligned} \quad (\text{F1})$$

Substituting $k_{j+1} - k_j = -\xi g'_\theta(k_j)$ in (F1),

$$\xi \left(1 - \frac{\xi L}{2} \right) |g'(k_j)|^2 \leq g(k_j) - g(k_{j+1}).$$

Choosing $\xi = \frac{1}{L}$, we obtain

$$\frac{1}{2L} |g'(k_j)|^2 \leq g(k_j) - g(k_{j+1}) < \frac{\delta^2}{2Lk_{\max}^2}.$$

Therefore,

$$|g'(k_j)| < \frac{\delta}{k_{\max}}. \quad \blacksquare$$

Lemma 5. Suppose $g(\cdot)$ is convex, and $g^* = \min_{k \in [1, k_{\max}]} g(k)$. If $|g'(k_j)| < \delta/k_{\max}$, then $|g(k_j) - g^*| \leq \delta$, for any $k_j \in [1, k_{\max}]$.

Proof. From the convexity of $g(\cdot)$, we have $g(k) \geq g'(k_j)(k - k_j) + g(k_j)$, for any $k, k_j \in [1, k_{\max}]$. If $g'(k_j) < 0$, then

$$g(k) - g(k_j) \geq g'(k_j)(k_{\max} - k_j) \geq g'(k_j)(k_{\max} - 1), \forall k.$$

Therefore,

$$g(k_j) - g^* \leq |g'(k_j)|(k_{\max} - 1) \leq |g'(k_j)|k_{\max} \leq \delta.$$

On the other hand, if $g'(k_j) \geq 0$, then

$$\begin{aligned} g(k) - g(k_j) &\geq g'(k_j)(k - k_j) \geq g'(k_j)(1 - k_j), \forall k, \\ g(k_j) - g^* &\leq g'(k_j)(k_{\max} - 1) \leq g'(k_j)k_{\max} \leq \delta. \end{aligned}$$

Therefore, combining both cases, if $|g'(k_j)| \leq \frac{\delta}{k_{\max}}$, then $|g(k_j) - g^*| \leq \delta$. \blacksquare

Lemma 6. Projected gradient descent (PROJGD) does not cross any stationary point: Let k_j and k_{j+1} be the successive iterates generated by the PROJGD routine for $g(\cdot)$. Suppose the step size $\xi \in (0, \frac{1}{L}]$, then $g'(k_j)g'(k_{j+1}) \geq 0$.

Proof. From the definition of the Lipschitz gradient, we have $|g'(k_j) - g'(k_{j+1})| \leq L|k_j - k_{j+1}| = L\xi|g'(k_j)|$, where the last equality holds, since k_{j+1} is generated from the PROJGD routine. Suppose, $g'(k_j) \geq 0$, then the following inequalities hold:

$$\begin{aligned} -L\xi g'(k_j) &\leq g'(k_j) - g'(k_{j+1}) \leq L\xi g'(k_j), \\ -L\xi g'(k_j) &\leq g'(k_{j+1}) - g'(k_j) \leq L\xi g'(k_j), \\ (1 - L\xi)g'(k_j) &\leq g'(k_{j+1}) \leq (1 + L\xi)g'(k_j). \end{aligned}$$

For $\xi \leq \frac{1}{L}$, we obtain

$$g'(k_{j+1}) \geq g'(k_j)(1 - L\xi) \geq 0.$$

Symmetrically, if $g'(k_j) \leq 0$, then $g'(k_{j+1}) \leq 0$. Combining both cases, we obtain $g'(k_j)g'(k_{j+1}) \geq 0$. \blacksquare

Lemma 7. Least difference between the successive iterates of PROJGD: Let k_j and k_{j+1} be the successive iterates generated by the PROJGD routine for $g(\cdot)$, with a step size $\xi = \frac{1}{L}$. If $|g(k_j) - g(k_{j+1})| > \zeta$, then $|k_j - k_{j+1}| > \sqrt{\frac{2\zeta}{5L}}$.

Proof. From the definition of the Lipschitz gradient, we have

$$\begin{aligned} |g'(k_{j+1}) - g'(k_j)| &\leq L|k_{j+1} - k_j|, \\ |g'(k_{j+1})| &\leq |g'(k_j)| + L|k_{j+1} - k_j| \\ &= \frac{|k_{j+1} - k_j|}{\xi} + L|k_{j+1} - k_j|. \end{aligned}$$

Substituting $\xi = \frac{1}{L}$ we obtain

$$|g'(k_{j+1})| \leq 2L|k_j - k_{j+1}|. \quad (\text{F2})$$

Suppose $g(k_j) \geq g(k_{j+1})$, then using descent lemma [31] on $g(\cdot)$ at k_j and k_{j+1} yields

$$\begin{aligned} g(k_j) - g(k_{j+1}) &\leq g'(k_{j+1})(k_j - k_{j+1}) + \frac{L}{2}|k_j - k_{j+1}|^2 \\ &\leq |g'(k_{j+1})|(k_j - k_{j+1}) + \frac{L}{2}|k_j - k_{j+1}|^2. \end{aligned} \quad (\text{F3})$$

Substituting (F2) in (F3) and using $|g(k_j) - g(k_{j+1})| > \zeta$ we obtain

$$\zeta < |g(k_j) - g(k_{j+1})| \leq \frac{5L}{2}|k_j - k_{j+1}|^2. \quad (\text{F4})$$

It can be also verified that (F4) holds when $g(k_j) < g(k_{j+1})$. Therefore,

$$|k_{j+1} - k_j| > \begin{cases} \sqrt{\frac{2\zeta}{5L}}, & \text{if } k_{j+1} < k_{\max}, \\ \sqrt{\frac{2}{5L}(g(k_j) - g(k_{\max}))}, & \text{if } k_{j+1} = k_{\max}. \end{cases}$$

The second case is mentioned separately since (F4) may not hold when $k_{j+1} = k_{\max}$. ■

Corollary 2. Suppose PROJGD is used with stopping criterion 1 (Definition 3). If $|k_j - k_{j+1}| \leq \frac{\delta}{\sqrt{5Lk_{\max}}}$, then $|g(k_j) - g(k_{j+1})| \leq \frac{\delta^2}{2Lk_{\max}^2}$, when $k_{j+1} < k_{\max}$.

Proof. The result follows by substituting $\zeta = \frac{\delta^2}{2Lk_{\max}^2}$ in Lemma 7. ■

Corollary 3. Suppose PROJGD is used with stopping criterion 2 (Definition 4). If $|k_j - k_{j+1}| \leq \sqrt{\frac{2\delta}{5L}}$, then $|g(k_j) - g(k_{j+1})| \leq \delta$, when $k_{j+1} < k_{\max}$.

Proof. The result follows by substituting $\zeta = \delta$ in Lemma 7. ■

Remark 3. Corollaries 2 and 3 imply that PROJGD terminates in a finite number of iterations depending on the stopping criteria.

4. Lemmas: Perturbation

Lemma 8. The PERTURB routine does not miss stationary points: Suppose k_j meets the stopping criterion 2 (Definition 4). If the perturbation $\Delta k \leq \sqrt{\frac{2\delta}{L}}$, then the PERTURB routine does not miss any stationary points with an error greater than δ .

Proof. Let k_j, k_{j+1} be any two points in $[1, k_{\max}]$. Since $g(\cdot)$ is L Lipschitz, using descent lemma [31] on $g(\cdot)$ at k_j and k_{j+1} , we obtain

$$g(k_j) - g(k_{j+1}) \leq g'(k_{j+1})(k_j - k_{j+1}) + \frac{L}{2}|k_j - k_{j+1}|^2.$$

Let k_{j+1} be the closest stationary point to k_j , then

$$|g(k_j) - g(k_{j+1})| \leq \frac{L}{2}|k_j - k_{j+1}|^2.$$

Therefore, the following condition is necessary for the stopping criterion 2, i.e., $|g(k_j) - g(k_{j+1})| \geq \delta$ (Definition 4), to hold:

$$\Delta k = |k_j - k_{j+1}| \geq \sqrt{\frac{2\delta}{L}}.$$

■

Lemma 9. Upper bound on the number of stationary points: Consider a set of stationary points $\{k_s\}$ of $g(\cdot)$ in $[1, k_{\max}]$ such that for every k_s , the adjacent stationary point k_{s+1} , $|g(k_s) - g(k_{s+1})| \geq \delta$. The number of such stationary points is finite and bounded above as $\lceil k_{\max}L/\delta \rceil$.

Proof. From the proof of Lemma 8, it follows that if $|g(k_s) - g(k_{s+1})| \geq \delta$, then the stationary points are separated by at least $|k_s - k_{s+1}| \geq \sqrt{\frac{2\delta}{L}}$. Therefore, the number of stationary points in $[1, k_{\max}]$ is at most $\lceil k_{\max} \sqrt{\frac{L}{2\delta}} \rceil$. ■

-
- [1] Y. I. Manin, *Vychislimoe i nevychislimoe* (Sov. Radiophys., 1980).
 - [2] R. P. Feynman, Simulating physics with computers, *Int. J. Theor. Phys.* **21**, 467 (1982).
 - [3] D. Deutsch, Quantum theory, the Church-Turing principle and the universal quantum computer, *Proc. R. Soc. A* **400**, 97 (1985).
 - [4] D. Deutsch and R. Jozsa, Rapid solution of problems by quantum computation, *Proc. R. Soc. A* **439**, 553 (1992).
 - [5] J. Preskill, Quantum computing in the NISQ era and beyond, *Quantum* **2**, 79 (2018).
 - [6] P. W. Shor, Fault-tolerant quantum computation, in *Proceedings of the 37th Annual Conference on Foundations of Computer Science* (IEEE, Burlington, Vermont, USA, 1996), pp. 56–65.
 - [7] A. M. Steane, Error correcting codes in quantum theory, *Phys. Rev. Lett.* **77**, 793 (1996).
 - [8] D. Aharonov and M. Ben-Or, Fault-tolerant quantum computation with constant error, in *Proceedings of the 29th Annual ACM Symposium on Theory of Computing* (ACM, El Paso, Texas, USA, 1997), pp. 176–188.
 - [9] A. Y. Kitaev, Quantum computations: Algorithms and error correction, *Russian Math. Surveys* **52**, 1191 (1997).
 - [10] D. Gottesman, Fault-tolerant quantum computation with constant overhead, *Quantum Inf. Comput.* **14**, 1338 (2014).
 - [11] O. Fawzi, A. Grospellier, and A. Leverrier, Constant overhead quantum fault tolerance with quantum expander codes, *Commun. ACM* **64**, 106 (2021).
 - [12] M. G. Taylor, Reliable information storage in memories designed from unreliable components, *Bell Syst. Tech. J.* **47**, 2299 (1968).
 - [13] L. R. Varshney, Performance of LDPC codes under faulty iterative decoding, *IEEE Trans. Inf. Theory* **57**, 4427 (2011).
 - [14] O. Fawzi, A. Müller-Hermes, and A. Shayeghi, A lower bound on the space overhead of fault-tolerant quantum computation, in *Proceedings of the 13th Innovations in Theoretical Computer Science Conference* (Schloss Dagstuhl, Berkeley, California, USA, 2022).
 - [15] A. A. Razborov, An upper bound on the threshold quantum decoherence rate, *Quantum Inf. Comput.* **4**, 222 (2004).

- [16] J. Kempe, O. Regev, F. Unger, and R. d. Wolf, Upper bounds on the noise threshold for fault-tolerant quantum computing, in *International Colloquium on Automata, Languages, and Programming* (Springer, New York, 2008), pp. 845–856.
- [17] A. W. Harrow and M. A. Nielsen, Robustness of quantum gates in the presence of noise, *Phys. Rev. A* **68**, 012308 (2003).
- [18] Uthirakalyani G, A. K. Nayak, and A. Chatterjee, A converse for fault-tolerant quantum computation, *Quantum* **7**, 1087 (2023).
- [19] P. W. Shor, Algorithms for quantum computation: discrete logarithms and factoring, in *Proceedings of the 35th Annual Symposium on Foundations of Computer Science* (IEEE, Santa Fe, New Mexico, USA, 1994), pp. 124–134.
- [20] L. K. Grover, A fast quantum mechanical algorithm for database search, in *Proceedings of the 28th Annual ACM Symposium on Theory of Computing* (ACM, Philadelphia, Pennsylvania, USA, 1996), pp. 212–219.
- [21] J. Ikonen, J. Salmilehto, and M. Möttönen, Energy-efficient quantum computing, *npj Quantum Inf.* **3**, 17 (2017).
- [22] C. Monroe and J. Kim, Scaling the ion trap quantum processor, *Science* **339**, 1164 (2013).
- [23] F. Arute, K. Arya, R. Babbush, D. Bacon, J. C. Bardin, R. Barends, R. Biswas, S. Boixo, F. G. Brandao, D. A. Buell, B. Burkett, Y. Chen, Z. Chen, B. Chiaro, R. Collins, W. Courtney, A. Dunsworth, E. Farhi *et al.*, Quantum supremacy using a programmable superconducting processor, *Nature (London)* **574**, 505 (2019).
- [24] M. Fellous-Asiani, J. H. Chai, R. S. Whitney, A. Auffèves, and H. K. Ng, Limitations in quantum computing from resource constraints, *PRX Quantum* **2**, 040335 (2021).
- [25] M. A. Nielsen and I. Chuang, *Quantum computation and quantum information* (Cambridge University Press, Cambridge, 2002).
- [26] J. von Neumann, Probabilistic logics and the synthesis of reliable organisms from unreliable components, in *Automata Studies*, edited by C. E. Shannon and J. McCarthy (Princeton University Press, Princeton, NJ, USA, 1956), pp. 43–98.
- [27] N. Pippenger, Reliable computation by formulas in the presence of noise, *IEEE Trans. Inf. Theory* **34**, 194 (1988).
- [28] S. Khatri and M. M. Wilde, Principles of quantum communication theory: A modern approach, [arXiv:2011.04672](https://arxiv.org/abs/2011.04672).
- [29] S. Wright *et al.*, Numerical optimization, *Springer Science* **35**, 7 (1999).
- [30] S. Khatri, K. Sharma, and M. M. Wilde, Information-theoretic aspects of the generalized amplitude-damping channel, *Phys. Rev. A* **102**, 012401 (2020).
- [31] D. P. Bertsekas, Nonlinear programming, *J. Operat. Res. Society* **48**, 334 (1997).