

ARTICLE OPEN



Multi-exponential error extrapolation and combining error mitigation techniques for NISQ applications

Zhenyu Cai ^{1,2}✉

Noise in quantum hardware remains the biggest roadblock for the implementation of quantum computers. To fight the noise in the practical application of near-term quantum computers, instead of relying on quantum error correction which requires large qubit overhead, we turn to quantum error mitigation, in which we make use of extra measurements. Error extrapolation is an error mitigation technique that has been successfully implemented experimentally. Numerical simulation and heuristic arguments have indicated that exponential curves are effective for extrapolation in the large circuit limit with an expected circuit error count around unity. In this Article, we extend this to multi-exponential error extrapolation and provide more rigorous proof for its effectiveness under Pauli noise. This is further validated via our numerical simulations, showing orders of magnitude improvements in the estimation accuracy over single-exponential extrapolation. Moreover, we develop methods to combine error extrapolation with two other error mitigation techniques: quasi-probability and symmetry verification, through exploiting features of these individual techniques. As shown in our simulation, our combined method can achieve low estimation bias with a sampling cost multiple times smaller than quasi-probability while without needing to be able to adjust the hardware error rate as required in canonical error extrapolation.

npj Quantum Information (2021)7:80; <https://doi.org/10.1038/s41534-021-00404-3>

INTRODUCTION

While fault-tolerant quantum computers promise huge speed-up over classical computers in areas like chemistry simulations, optimisation and decryption, their implementations remain a long term goal due to the large qubit overhead required for quantum error correction. With the recent rapid advance of quantum computer hardware in terms of both qubit quantity and quality, culminating with the “quantum supremacy” experiment¹, one must wonder is it possible for us to perform classically intractable computations on such Noisy Intermediate-Scale Quantum (NISQ) hardware without quantum error correction². Resource estimation has been performed for one of the most promising applications on NISQ hardware: the Fermi–Hubbard model simulation^{3,4}, realising that even with an optimistic local gate error rate of 10^{-4} , the large number of gates needed for a classically intractable calculation will lead to an expected circuit error count of the order of unity. To obtain any meaningful results under such an expected circuit error count, it is essential to employ error mitigation techniques, which relies on making extra measurements, as opposed to employing extra qubits in the case of quantum error correction, to estimate the noise-free expectation values from the noisy measurement results. Three of the most well-studied error mitigation techniques are symmetry verification^{5,6}, quasi-probability and error extrapolation^{7–9}.

Previously all of these error mitigation techniques have been discussed separately. They make use of different information about the hardware and the computation problems to perform different sets of extra circuit runs for error mitigation. Symmetry verification makes use of the symmetry in the simulated system and performs circuit runs with additional measurements. Quasi-probability makes use of the error models of the circuit components and performs circuit runs with different additional gates in the circuit. Error extrapolation makes use of the

knowledge about tuning the noise strength via physical control of the hardware and performs additional circuit runs at different noise levels. Consequently, the three error mitigation techniques are equipped to combat different types of noise with different additional sampling costs (number of additional circuit runs required). Hence, it is natural to wonder how these techniques might complement each other. For NISQ application, it may be essential to understand and develop ways for these error mitigation techniques to work in unison, to achieve better performance than the individual techniques in terms of lower bias in the noise-free expectation values estimates and/or lower sampling costs. Thus one key focus of our Article is on the methods for combining these error mitigation techniques and trying to gauge their performance under different scenarios through analytical arguments and numerical simulations.

To achieve efficient combinations of these mitigation techniques, we will need to exploit certain features of these constituent techniques. As we will see later, we will show that quasi-probability can be used for error transformation instead of error removal and the circuit runs that fail the symmetry verification can actually be utilised instead of being discarded. It is also essential to understand the mechanism behind error extrapolation, especially in the NISQ limit in which the number of errors in the circuit will follow a Poisson distribution. Heuristic arguments and numerical validations have been made by Endo et al.⁹ on error extrapolation using exponential decay curves in this NISQ limit. However, it cannot be applied to certain situations arising in practice, for example when the data points have an increasing trend. Our Article will take this further and provide a more rigorous argument showing that single-exponential error extrapolation is just a special case of the more general multi-exponential error extrapolation framework, using which we can achieve a much lower estimation bias.

¹Department of Materials, University of Oxford, Oxford, UK. ²Quantum Motion Technologies Ltd., Nexus, Discovery Way, Leeds, UK. ✉email: cai.zhenyu.physics@gmail.com

RESULTS

Symmetry verification

Suppose we want to perform a state preparation and we know that the correct state must follow a certain symmetry S , i.e., we expect our end state to be the eigenstate of S with the correct eigenvalue s (or within a set of eigenvalues $\{s\}$). In such a case, we can perform S measurement on our output state and discard the circuit runs that produce states that violate our symmetry. This was first proposed and studied by McArdle et al.⁵ and Bonet-Monroig et al.⁶. Discarding erroneous circuit runs results in an effective density matrix that is the original density matrix ρ projected into the $S = s$ subspace via the projection operator Π_s

$$\rho_s = \frac{\Pi_s \rho \Pi_s}{\text{Tr}(\Pi_s \rho \Pi_s)} = \frac{\Pi_s \rho \Pi_s}{\text{Tr}(\Pi_s \rho)}.$$

Here, we have used $\Pi_s \Pi_s = \Pi_s$.

Now let us suppose we want to measure an observable O , which commutes with our symmetry S . Thus they can both be measured in the same run and we will discard the measurement results in the runs that failed the symmetry verification. The symmetry-verified expectation value of the observable O is then

$$\langle O_{\text{sym}} \rangle = \text{Tr}(O \rho_s) = \frac{\text{Tr}(O \Pi_s \rho)}{\text{Tr}(\Pi_s \rho)} \equiv \frac{\langle O \Pi_s \rangle}{\langle \Pi_s \rangle} \quad (1)$$

in which we have used $[S, O] = 0 \Rightarrow [\Pi_s, O] = 0$. Note that Π_s measurement takes the value 1 if the symmetry verification is passed and 0 otherwise and hence $\langle \Pi_s \rangle = \text{Tr}(\Pi_s \rho)$ is just the fraction of circuit runs that fulfil the symmetry condition. We will use P_d to denote the fraction of circuit runs that fail the symmetry verification, which gives

$$\langle \Pi_s \rangle = \text{Tr}(\Pi_s \rho) = 1 - P_d.$$

Recall that ρ_s is the effective density matrix of the non-discarded runs, which as mentioned is a $\text{Tr}(\Pi_s \rho)$ fraction of the total number of runs. Therefore to obtain statistics from ρ_s , we need a factor of

$$C_s = \frac{1}{\text{Tr}(\Pi_s \rho)} = \frac{1}{1 - P_d} \quad (2)$$

more circuit runs than obtaining directly from ρ .

In the method discussed above, $O \Pi_s$ is usually obtained through measuring O and S in the same run. However, sometimes this cannot be done due to, for example, the inability to perform non-demolishing measurements. In such a case we need to break $O \Pi_s$ into its Pauli basis⁶ and reconstruct it via post-processing, this is discussed in Supplementary Note 1. In this article, we will mainly be focusing on direct symmetry verification instead of post-processing verification, but most of the arguments are valid for both methods besides discussions about costs.

Now let us move on to see what errors are detectable by symmetry verification. We want to produce the state $|\psi_f\rangle$ which is known to fall within the $S = s$ symmetry subspace

$$S|\psi_f\rangle = s|\psi_f\rangle.$$

To produce the state, we usually start with a state $|\psi_0\rangle$ that follows the same symmetry and uses a circuit U that consists of components that conserve the symmetry

$$|\psi_f\rangle = U|\psi_0\rangle, \quad S|\psi_0\rangle = s|\psi_0\rangle, \quad [U, S] = 0.$$

Suppose that some error E occurs during the circuit in between the symmetry-preserving components and it satisfies

$$\Pi_{s'} E = E \Pi_s \quad (3)$$

in which s, s' are some eigenvalues of the symmetry operator S .

We then have

$$\Pi_s E \Pi_s = E \Pi_{s'} \Pi_s = \begin{cases} E \Pi_s & s = s' \\ 0 & s \neq s' \end{cases} \quad (4)$$

$\Pi_s E \Pi_s = E \Pi_s$ means that E is a transformation within the $S = s$ subspace, hence E is undetectable by the symmetry verification using S . $\Pi_s E \Pi_s = 0$ means that E contains no components that map states in the $S = s$ subspace back into the same subspace, hence E is (completely) detectable by the symmetry verification using S . A general error will be a combination of detectable and undetectable error components.

In this Article, we will be focusing on Pauli errors and Pauli symmetries, for which Eq. (4) is reduced to

$$\begin{aligned} [S, E] &= 0 \Rightarrow s = s' & E \text{ is undetectable} \\ \{S, E\} &= 0 \Rightarrow s = -s' & E \text{ is detectable.} \end{aligned} \quad (5)$$

Quasi-probability

To describe the quasi-probability method, we will make use of the Pauli transfer matrix (PTM) formalism¹⁰. Using \mathbb{G} to denote the set of Pauli operators, any density operators can be written in the vector form by decomposing into the Pauli basis $G \in \mathbb{G}$

$$\begin{aligned} \rho &= \frac{1}{2^N} \sum_{G \in \mathbb{G}} \text{Tr}(G \rho) G \\ \Rightarrow |\rho\rangle\rangle &= \sum_{G \in \mathbb{G}} |G\rangle\rangle \langle\langle G|\rho\rangle\rangle \end{aligned}$$

where we have defined the inner product as

$$\langle\langle G|\rho\rangle\rangle = \frac{1}{\sqrt{2^N}} \text{Tr}(G \rho).$$

Note that we need to add a normalisation factor $\frac{1}{\sqrt{2^N}}$ when we use the Pauli operators as a basis, with N being the number of qubits.

The quasi-probability method was first introduced by Temme et al.⁸ and the implementation details were later studied by Endo et al.⁹. Let us suppose we are trying to perform the operation \mathcal{U} , but in practice, we can only implement its noisy version

$$\mathcal{U}_e = \mathcal{E} \mathcal{U}.$$

In addition to \mathcal{U}_e , we can also implement a set of basis operation $\{\mathcal{B}_n\}$. We can decompose the ideal operation \mathcal{U} that we want to implement into a set of gates $\{\mathcal{B}_n \mathcal{U}_e\}$ that we can implement

$$\mathcal{U} = \sum_n q_n \mathcal{B}_n \mathcal{U}_e \Rightarrow \mathcal{E}^{-1} = \sum_n q_n \mathcal{B}_n.$$

In this way, we are essentially trying to simulate the behaviour of the inverse noise channel \mathcal{E}^{-1} using the set of basis operations $\{\mathcal{B}_n\}$, which can undo the noise \mathcal{E} .

If we have a state $|\rho\rangle\rangle$ passing through the circuit \mathcal{U} and we perform measurement O , then the observable we obtain during the experiment will be:

$$\begin{aligned} \langle O \rangle &= \langle\langle O | \mathcal{U} | \rho \rangle\rangle = \sum_n q_n \langle\langle O | \mathcal{B}_n \mathcal{U}_e | \rho \rangle\rangle \\ &= Q \sum_n s_n \frac{|q_n|}{Q} \langle\langle O | \mathcal{B}_n \mathcal{U}_e | \rho \rangle\rangle. \end{aligned}$$

in which $Q = \sum_n |q_n|$ and $s_n = \text{sgn}(q_n)$. This is implemented by sampling from the set of basis operations $\{\mathcal{B}_n\}$ with the probability distribution $\{\frac{|q_n|}{Q}\}$. We will weight each measurement outcome by the sign factor s_n and rescale the final expectation value by the factor Q .

Now if we break down our computation into many components $\mathcal{U} = \prod_{m=1}^M \mathcal{U}_m$, with noise associated with each component, then

the observable that we want to measure is

$$\langle O \rangle = \left\langle \left\langle O \left| \prod_{m=1}^M \mathcal{U}_m \right| \rho \right\rangle \right\rangle,$$

but in reality, we can only implement the noisy version

$$\langle O_\epsilon \rangle = \left\langle \left\langle O \left| \prod_{m=1}^M \mathcal{E}_m \mathcal{U}_m \right| \rho \right\rangle \right\rangle.$$

Each noise element can be removed by simulating the inverse channels using the set of basis operations $\{\mathcal{B}_n\}$

$$\mathcal{E}_m^{-1} = \sum_n q_{mn} \mathcal{B}_n = R_m \sum_n s_{mn} \frac{|q_{mn}|}{R_m} \mathcal{B}_n$$

with $R_m = \sum_n |q_{mn}|$.

Hence, we can get back the noiseless observable using

$$\begin{aligned} \langle O \rangle &= \left\langle \left\langle O \left| \prod_{m=1}^M \left(\sum_{n_m} q_{mn_m} \mathcal{B}_{n_m} \right) \mathcal{E}_m \mathcal{U}_m \right| \rho \right\rangle \right\rangle \\ &= Q \sum_{\vec{n}} s_{\vec{n}} \frac{|q_{\vec{n}}|}{Q} \left\langle \left\langle O \left| \prod_{m=1}^M \mathcal{B}_{n_m} \mathcal{E}_m \mathcal{U}_m \right| \rho \right\rangle \right\rangle \end{aligned}$$

in which we have used \vec{n} to denote the set of number $\{n_1, n_2, \dots, n_M\}$ and we have defined $q_{\vec{n}} = \prod_{m=1}^M q_{mn_m}$, $s_{\vec{n}} = \prod_{m=1}^M s_{mn_m} = \text{sgn}(q_{\vec{n}})$ and

$$Q = \prod_{m=1}^M R_m = \prod_{m=1}^M \sum_{n_m} |q_{mn_m}|. \quad (6)$$

To implement this, we simply sample the set of basis operations $\{\mathcal{B}_{n_1}, \mathcal{B}_{n_2}, \dots, \mathcal{B}_{n_M}\}$ that we want to implement with the probability $\frac{|q_{\vec{n}}|}{Q}$ and weight each measurement outcome by a sign factor $s_{\vec{n}} = \text{sgn}(q_{\vec{n}})$, so that the outcome we get is an effective Pauli observable O_Q . And the error-free observable expectation value can be obtained via

$$\langle O \rangle = Q \langle O_Q \rangle. \quad (7)$$

Hence, to estimate $\langle O \rangle$ by sampling O_Q , we need C_Q times more samples than sampling O directly, where the sampling cost factor C_Q is

$$C_Q = Q^2 = \left(\prod_{m=1}^M \sum_{n_m} |q_{mn_m}| \right)^2. \quad (8)$$

In this article, we will be mainly focusing on Pauli error channels, which can be inverted using quasi-probability by employing Pauli gates as the basis operations.

Using $\overline{}$ to denote a super-operator

$$\left(\overline{A} + \overline{B} \right) \rho = A \rho A^\dagger + B \rho B^\dagger,$$

any Pauli channel can be written in the form

$$\mathcal{G}_{p_\epsilon} = (1 - p_\epsilon) \mathcal{I} + p_\epsilon \sum_{G \in \mathbb{G}-I} \alpha_G \overline{G} \quad (9)$$

with $\sum_G \alpha_G = 1$. We use p_ϵ to denote the total probability of all the non-identity components. This channel can be approximately inverted using the quasi-probability channel $\mathcal{G}_{-p_\epsilon}$ since

$$\mathcal{G}_{-p_\epsilon} \mathcal{G}_{p_\epsilon} \approx \mathcal{I} + \mathcal{O}(p_\epsilon^2).$$

Hence, to the first-order approximation, the cost of inverting \mathcal{G}_{p_ϵ} will be the cost of implementing $\mathcal{G}_{-p_\epsilon}$, which using Eq. (8) is

$$C_{Q1,0} \approx (1 + 2p_\epsilon)^2 \approx 1 + 4p_\epsilon. \quad (10)$$

Here, we have only discussed approximately inverting a Pauli channel because the exact inverse channel can be hard to express in a compact analytical form. However, it can be obtained

numerically by first obtaining the PTM of the noise channel and then performing matrix inversion.

Instead of removing the error channel completely, quasi-probability can also be used to transform the form of an error channel. In the case of Pauli channels, suppose we want to transform a channel of the form in Eq. (9) to

$$\mathcal{F}_{q_\epsilon} = (1 - q_\epsilon) \mathcal{I} + q_\epsilon \sum_{G \in \mathbb{G}-I} \beta_G \overline{G},$$

we can approximately achieve this transformation up to first order in q_ϵ and p_ϵ by applying the quasi-probability channel

$$\mathcal{R} = (1 + (p_\epsilon - q_\epsilon)) \mathcal{I} - \sum_{G \in \mathbb{G}-I} (p_\epsilon \alpha_G - q_\epsilon \beta_G) \overline{G}.$$

This will incur the implementation cost

$$C_{Q1,q} = \left(|1 + (p_\epsilon - q_\epsilon)| + \sum_{G \in \mathbb{G}-I} |p_\epsilon \alpha_G - q_\epsilon \beta_G| \right)^2.$$

In the limit of small p_ϵ and q_ϵ , we have

$$\begin{aligned} C_{Q1,q} &\approx 1 + 2(p_\epsilon - q_\epsilon) + 2 \sum_{G \in \mathbb{G}-I} |p_\epsilon \alpha_G - q_\epsilon \beta_G| \\ &= 1 + 4 \sum_{\substack{G \in \mathbb{G}-I \\ p_\epsilon \alpha_G > q_\epsilon \beta_G}} (p_\epsilon \alpha_G - q_\epsilon \beta_G). \end{aligned}$$

In the last step we have used $\sum_G (p_\epsilon \alpha_G - q_\epsilon \beta_G) = p_\epsilon - q_\epsilon$ from $\sum_G \alpha_G = \sum_G \beta_G = 1$.

If we are suppressing all error components evenly, or if we are simply removing certain error components, we will have $p_\epsilon \alpha_G \geq q_\epsilon \beta_G \quad \forall G \in \mathbb{G} - I$. In this case, the cost of implementing the transformation using quasi-probability will simply be

$$C_{Q1,q} \approx 1 + 4(p_\epsilon - q_\epsilon). \quad (11)$$

Group errors

Here, we will introduce a special kind of error channel: group error channels, which enable us to make more analytical predictions about the error mitigation techniques that we have already discussed and also will help our understanding about error extrapolation later.

The group error $\mathcal{J}_{p,\mathbb{E}}$ of the group \mathbb{E} is defined to be

$$\mathcal{J}_{p,\mathbb{E}} = (1 - p) \overline{I} + \frac{p}{|\mathbb{E}|} \sum_{E \in \mathbb{E}} \overline{E} \quad (12)$$

By groups we mean the subgroups of the Pauli group with a composition rule that ignores all the irrelevant phase factors. For the case of $p = 1$, we will call $\mathcal{J}_{1,\mathbb{E}}$ the pure group errors.

Many physically interesting noise models like depolarising channels, dephasing channels, Pauli-twirled swap errors and dipole-dipole errors are all group errors.

Now let us consider the effect of applying a set of Pauli symmetry checks \mathbb{S} to the group error in Eq. (12). Using Eq. (5), \mathbb{S} can remove and detect components in $\mathcal{J}_{p,\mathbb{E}}$ that anti-commute with any elements in $S \in \mathbb{S}$. We look at the action of \mathbb{S} on the subset of qubits affected by $\mathcal{J}_{p,\mathbb{E}}$, and denote the set of these operators on the subset of qubits as \mathbb{S}_{sub} . The commutation relationship between \mathbb{S} and \mathbb{E} is equivalent to that of \mathbb{S}_{sub} and \mathbb{E} . We denote their generators as $\tilde{\mathbb{S}}_{\text{sub}}$ and $\tilde{\mathbb{E}}$. Note that here \mathbb{S}_{sub} is not a group, by $\tilde{\mathbb{S}}_{\text{sub}}$ we just mean the set of independent elements in \mathbb{S}_{sub} . For Pauli generators, we can choose $\tilde{\mathbb{E}}$ in such a way that for every $\tilde{S}_{\text{sub}} \in \tilde{\mathbb{S}}_{\text{sub}}$, there will at most be only one element in $\tilde{\mathbb{E}}$ that anti-commutes with it. We will denote the elements in $\tilde{\mathbb{E}}$ that commute with all elements in $\tilde{\mathbb{S}}_{\text{sub}}$ as \tilde{Q}

$$\tilde{Q} = \{ \tilde{E} \in \tilde{\mathbb{E}} \mid [\tilde{E}, \tilde{S}_{\text{sub}}] = 0 \quad \forall \tilde{S}_{\text{sub}} \in \tilde{\mathbb{S}}_{\text{sub}} \}$$

and it will generate the remaining error components in \mathbb{E} that are

not detectable, which we denote as \mathbb{Q} . Hence the detectable error components are just $\mathbb{E} - \mathbb{Q}$

Going back to our error channel in Eq. (12), the probability that the error gets detected is just the total probability of the detectable error components

$$p_d = \frac{|\mathbb{E}| - |\mathbb{Q}|}{|\mathbb{E}|} p. \quad (13)$$

Removing the detected errors in $\mathcal{J}_{p,\mathbb{E}}$ and renormalising the error channel by the factor $1 - p_d$ gives the effective channel after verification, which is just another group error channel

$$\mathcal{J}_{r,\mathbb{Q}} = (1 - r) \bar{I} + \frac{r}{|\mathbb{Q}|} \sum_{Q \in \mathbb{Q}} \bar{Q} \quad (14)$$

with

$$r = \frac{|\mathbb{Q}|p}{|\mathbb{E}|(1 - p_d)} = \frac{|\mathbb{Q}|p}{|\mathbb{E}|(1 - p) + |\mathbb{Q}|p} \approx \frac{|\mathbb{Q}|}{|\mathbb{E}|} p + \mathcal{O}(p^2).$$

An example of removing detectable errors for depolarising channels will be worked out later in Section “Numerical simulation for multi-exponential extrapolation”.

For a given general Pauli channel, we have only discussed its approximate inverse channel in Section “Quasi-probability”. This is because its exact inverse channel can be hard to express in a compact analytical form. However, for any group channel, we can easily write down the explicit form of its inverse channel.

As shown in Supplementary Note 2, it is easy to verify that the inverse of a group channel $\mathcal{J}_{p,\mathbb{E}}$ is just

$$(\mathcal{J}_{p,\mathbb{E}})^{-1} = \mathcal{J}_{-a,\mathbb{E}} = (1 + a)\mathcal{I} - a\mathcal{J}_{1,\mathbb{E}} \quad (15)$$

with $a = \frac{p}{1-p}$.

Using Eq. (15) and (8), the cost of using quasi-probability to invert $\mathcal{J}_{p,\mathbb{E}}$ is thus

$$C_{Q1,0} = \left(1 + 2 \frac{(|\mathbb{E}| - 1)p}{|\mathbb{E}|(1 - p)}\right)^2 \approx 1 + 4 \frac{|\mathbb{E}| - 1}{|\mathbb{E}|} p + \mathcal{O}(p^2), \quad (16)$$

which is the same as Eq. (10) with

$$p_\epsilon = \frac{|\mathbb{E}| - 1}{|\mathbb{E}|} p. \quad (17)$$

As shown in Eq. (14), for a given group error $\mathcal{J}_{p,\mathbb{E}}$, the resultant error channel after symmetry verification is another group channel $\mathcal{J}_{r,\mathbb{Q}}$ where \mathbb{Q} is a subgroup of \mathbb{E} and $r = \frac{|\mathbb{Q}|}{|\mathbb{E}|} p$. The remaining errors can then be completely removed by implementing $\mathcal{J}_{r,\mathbb{Q}}^{-1}$ using quasi-probability.

Similarly, if we implement the quasi-probability inverse channel $\mathcal{J}_{r,\mathbb{Q}}^{-1}$ first and then perform symmetry verification, we can still completely remove the group error $\mathcal{J}_{p,\mathbb{E}}$. The gates we need to implement in the inverse channel $\mathcal{J}_{r,\mathbb{Q}}^{-1}$ will not be detected and thus will not be affected by the symmetry verification. As shown in Supplementary Note 3, the resultant error channel after applying $\mathcal{J}_{r,\mathbb{Q}}^{-1}$ is

$$\mathcal{J}_{r,\mathbb{Q}}^{-1} \mathcal{J}_{p,\mathbb{E}} = (1 - p_d) \bar{I} + \frac{p_d}{|\mathbb{E}| - |\mathbb{Q}|} \sum_{V \in \mathbb{E}, V \notin \mathbb{Q}} \bar{V}. \quad (18)$$

This is a channel that only contains the error components that are detectable by the symmetry verification with the error rate p_d .

As discussed in Section “Quasi-probability” and explicitly shown in Supplementary Note 3, we can implement additional quasi-probability operations to further reduce the error rate of the resultant channel to $q \leq p_d$. The resultant detectable error channel

will be

$$\begin{aligned} \mathcal{V}_q &= (1 - q) \bar{I} + \frac{q}{|\mathbb{E}| - |\mathbb{Q}|} \sum_{V \in \mathbb{E}, V \notin \mathbb{Q}} \bar{V} \\ &= (1 - q) \bar{I} + q \mathcal{V}_1 \quad q \leq p_d. \end{aligned} \quad (19)$$

NISQ limit

The number of possible error locations in the circuit, which is usually the number of gates in the circuit, will be denoted as M . These error locations might experience different noise with different error rates. From here on, instead of building our discussions around local gate error rates, we will see that the more natural quantity to consider in the context of NISQ error mitigation is the expected number of errors occurring in each circuit run, which is called the mean circuit error count and denoted as μ . In order to achieve quantum advantage using NISQ machine, we would generally expect the circuit size to be large enough to be classically intractable while the mean circuit error count should not be too far beyond unity

$$M \gg 1, \quad \mu \sim 1. \quad (20)$$

This is also called the Poisson limit since using Le Cam’s theorem, the number of errors occurring in each circuit run will follow the Poisson distribution—i.e. the probability that l errors occur will be

$$P_l = e^{-\mu} \frac{\mu^l}{l!}. \quad (21)$$

If we assume that every local error channel in the circuit can be approximated as the composition of an undetectable error channel and a detectable error channel, then symmetry verification will have no effects on the undetectable error channels and we can focus only on the detectable error channels. Alternatively, as we have seen in Section “Group errors”, we can use quasi-probability to remove all the local undetectable errors in the circuit, leaving us with only detectable error channels. The expected number of detectable errors occurring in each circuit run is denoted as μ_d . Taking the NISQ limit and using Eq. (20), the probability that l detectable errors occur in the circuit is

$$P_l = e^{-\mu_d} \frac{\mu_d^l}{l!}.$$

Using Eq. (5), an odd number of detectable errors will anti-commute with the symmetry and get detected while an even number of detectable errors will commute with the symmetry and pass the verification. Therefore, the total probability that the errors in the circuit will be detected by the verification of one Pauli symmetry is thus

$$P_d = \sum_{\text{odd } l} P_l = e^{-\mu_d} \sinh(\mu_d) = \frac{1 - e^{-2\mu_d}}{2}. \quad (22)$$

Note that this is upper-bounded by $\frac{1}{2}$, i.e. at most we can catch errors in half of the circuit runs.

Hence, using Eq. (2), the cost of implementing symmetry verification for one Pauli symmetry is

$$C_{S,\mu_d} = \frac{1}{1 - P_d} = \frac{1}{e^{-\mu_d} \cosh(\mu_d)} = \frac{2}{1 + e^{-2\mu_d}} \quad (23)$$

which is upper-bounded by 2.

After symmetry verification, the fraction of circuit runs that still have errors inside is

$$P_{\text{circ}} = 1 - P_d - P_0 = \frac{1}{2} (1 - e^{-\mu_d})^2. \quad (24)$$

In Eq. (11) we have only been focusing on applying quasi-probability to one error channel. Assuming there are M such

channels in the circuit, then using Eq. (11) and taking the NISQ limit (Eq. (20)), the sampling cost factor of transforming all M error channels with error probability p_e into new error channels with error probability $q_e \leq p_e$ using quasi-probability is

$$C_{Q,Mq} = C_{Q1,q}^M \approx \lim_{M \rightarrow \infty} (1 + 4(p_e - q_e))^M = e^{4M(p_e - q_e)}. \quad (25)$$

At $q_e = 0$, we will obtain the sampling cost of removing all the noise using quasi-probability

$$C_{Q,0} \approx e^{4Mp_e}. \quad (26)$$

Remember that we are focusing on Pauli errors and p_e is defined to be the total probability of all non-identity components. This is not equivalent to the error probability p because sometimes there are some identity components in our definition of error probability such as the group errors that we discussed in Section "Group errors". Similar to the definition of p_e , we can denote the expected number of non-identity errors in each circuit run as μ_e . In the above cases, we have $\mu_e = Mp_e$ and similarly we can define $v_e = Mq_e$. In the cases where different noise locations experience different noise, using Le Cam's theorem with negative probabilities and focusing on the zero-occurrence case, we can generalise Eq. (25) to

$$C_{Q,v} = e^{4(\mu_e - v_e)}. \quad (27)$$

i.e. the sampling cost of quasi-probability transformation grows exponentially with the reduction in the error rate $\mu_e - v_e$.

Multi-exponential error extrapolation

The idea of amplifying the hardware error rate and performing extrapolation using the original result and the noise-amplified result was first introduced by Li et al.⁷ and Temme et al.⁸, and was later successfully realised experimentally using superconducting qubits¹¹. Endo et al.⁹ provided heuristic arguments on why the exponential decay curve should be used for error extrapolation in the large circuit limit, whose improved performance over linear extrapolation was validated via numerical simulations in refs. 9,12.

Using \mathbb{L} to denote the set of locations that the errors have occurred, when l errors have occurred in the circuit, our observable O will be transformed to a noisy observable $O_{|\mathbb{L}|=l}$. Recall that in the NISQ limit, the probability that l errors happen (denoted as P_l) will follow the Poisson distribution in Eq. (21). Therefore, the expectation value of the observable O at the mean circuit error count μ is then

$$\langle O_\mu \rangle = \sum_{l=0}^{\infty} P_l \langle O_{|\mathbb{L}|=l} \rangle = e^{-\mu} \sum_{l=0}^{\infty} \frac{\mu^l}{l!} \langle O_{|\mathbb{L}|=l} \rangle. \quad (28)$$

Hence, how $\langle O_\mu \rangle$ changes with μ is entirely determined by how $\langle O_{|\mathbb{L}|=l} \rangle$ changes with l . When we try to fit a n th degree polynomial of μ to $\langle O_\mu \rangle$, for example performing a linear extrapolation^{7,8}, we are essentially assuming that $\langle O_{|\mathbb{L}|=l} \rangle$ is a n th degree polynomial of l using the expressions of the moments of the Poisson distribution.

At $l = 0$, we have the error-free result $\langle O \rangle$

$$\langle O_{|\mathbb{L}|=0} \rangle = \langle O \rangle.$$

At large error number l , in the case of stochastic errors, the circuit will move closer to a random circuit. Hence, for a Pauli observable O we will expect

$$\lim_{l \rightarrow \infty} \langle O_{|\mathbb{L}|=l} \rangle = 0.$$

A generic polynomial of l will not satisfy the above boundary conditions. Hence, to align with the above boundary conditions, we can instead assume an exponential decay of $\langle O_{|\mathbb{L}|=l} \rangle$ as l increase

$$\langle O_{|\mathbb{L}|=l} \rangle = \langle O \rangle (1 - \gamma)^l$$

in which γ is the observable decay rate that satisfies $0 \leq \gamma \leq 1$. This will lead to an exponential function in μ

$$\langle O_\mu \rangle = \langle O \rangle e^{-\mu} \sum_{l=0}^{\infty} \frac{(\mu(1 - \gamma))^l}{l!} = \langle O \rangle e^{-\gamma\mu}, \quad (29)$$

which is just the extrapolation curves employed in ref. 9.

Using Eq. (29), if we probe at the error rates μ and $\lambda\mu$, we can perform two-point exponential extrapolation and obtain the error-mitigated estimate of $\langle O \rangle$, denoted as $\langle O_0 \rangle$, using the following equation:

$$\langle O_0 \rangle = \left(\frac{\langle O_\mu \rangle^\lambda}{\langle O_{\lambda\mu} \rangle} \right)^{\frac{1}{\lambda-1}}.$$

As discussed in Supplementary Note 7B, the sampling cost factor of performing such an extrapolation is

$$C_E \approx 2 \frac{\lambda^2 e^{2\gamma\mu} + e^{2\lambda\gamma\mu}}{(\lambda - 1)^2}. \quad (30)$$

Now let us try to gain a deeper insight about the reason behind the exponential decay of $\langle O_{|\mathbb{L}|=l} \rangle$. If a Pauli error G occurs at the end of a circuit and we are trying to measure a Pauli observable O , then the expectation value is just

$$\text{Tr}(G\rho GO) = \eta(G, O) \text{Tr}(\rho O)$$

where $\eta(G, O)$ is the commutator between G and O

$$GO = \eta(G, O)OG.$$

If a pure group error $\mathcal{J}_{1,\mathbb{E}}$ occurs at the end of the circuit, then the resultant expectation value is

$$\begin{aligned} \text{Tr}(\mathcal{J}_{1,\mathbb{E}}(\rho)O) &= \frac{1}{|\mathbb{E}|} \sum_{E \in \mathbb{E}} \text{Tr}(E\rho EO) \\ &= \left(\frac{1}{|\mathbb{E}|} \sum_{E \in \mathbb{E}} \eta(E, O) \right) \text{Tr}(\rho O) \end{aligned}$$

Using the fact that the composition of the commutators of elements in a Pauli subgroup follows the same structure as the composition of the elements themselves

$$\eta(EE', O) = \eta(E', O)\eta(E, O),$$

we can rewrite the above formula in terms of the generator of \mathbb{E}

$$\text{Tr}(\mathcal{J}_{1,\mathbb{E}}(\rho)O) = \left(\prod_{\tilde{E} \in \mathbb{E}} \frac{1 + \eta(\tilde{E}, O)}{2} \right) \text{Tr}(\rho O) \quad (31)$$

in which

$$\prod_{\tilde{E} \in \mathbb{E}} \frac{1 + \eta(\tilde{E}, O)}{2} = \begin{cases} 1 & \text{if } \eta(\tilde{E}, O) = 1 \quad \forall \tilde{E} \in \mathbb{E} \\ 0 & \text{otherwise.} \end{cases}$$

Hence, if a pure group error $\mathcal{J}_{1,\mathbb{E}}$ occurs right before measuring a Pauli observable O , then the resultant expectation value $\text{Tr}(\mathcal{J}_{1,\mathbb{E}}(\rho)O)$ will only remain unchanged if O commutes with all elements in \mathbb{E} , otherwise the information about O is "erased" by the group error and the expectation value will be 0.

If we decompose the gates in a unitary circuit $U = \prod_{m=M}^1 V_m$ into their Pauli components: $V_m = \sum_{j_m} a_{mj_m} G_{mj_m}$, then we have

$$U = \prod_{m=M}^1 \sum_{j_m} a_{mj_m} G_{mj_m} = \sum_{\vec{j}} \alpha_{\vec{j}} G_{\vec{j}}$$

where

$$\sum_{\vec{j}} = \prod_{m=M}^1 \sum_{j_m} G_{\vec{j}} = \prod_{m=M}^1 G_{mj_m}, \quad \alpha_{\vec{j}} = \prod_{m=M}^1 a_{mj_m}.$$

i.e. the circuit U can be viewed as the superposition of many Pauli circuits $G_{\vec{j}}$.

The expectation value of observing O after applying the circuit U on ρ is

$$\begin{aligned} \langle O \rangle &= \text{Tr}(U\rho U^\dagger O) \\ &= 2 \sum_{\vec{i}, \vec{j}} \text{Re} \left\{ \alpha_i^* \alpha_j \text{Tr}(\rho G_i^\dagger O G_j) \right\} \end{aligned} \quad (32)$$

which is a linear sum of the measurement results for the set of effective Pauli observables $G_i^\dagger O G_j$ for different \vec{i} and \vec{j} . Similar to Eq. (31), the information about $G_i^\dagger O G_j$ will either be “erased” or perfectly preserved if a group error occurs in the circuit. Assuming the average fraction of group error locations in our circuit that can erase the information about $G_i^\dagger O G_j$ is $\gamma_{\vec{i}, \vec{j}}$, then as proven in Supplementary Note 4, in the limit of large M and non-vanishing $1 - \gamma_{\vec{i}, \vec{j}}$, the expectation value given l errors occurred can be approximated to be

$$\langle O_{|\mathbb{L}|=l} \rangle \approx 2 \sum_{\vec{i}, \vec{j}} \text{Re} \left\{ \alpha_i^* \alpha_j \text{Tr}(\rho G_i^\dagger O G_j) (1 - \gamma_{\vec{i}, \vec{j}})^l \right\}, \quad (33)$$

which is just a multi-exponential decay curve.

If we consider the case in which many error locations in the circuit are affected by the same type of noise, and adding onto the fact that in practice there are usually many repetitions of the circuit structures along the circuit and across the qubits, we can expect many $\gamma_{\vec{i}, \vec{j}}$ of different \vec{i} and \vec{j} to be very similar. Hence, by grouping the terms with similar $\gamma_{\vec{i}, \vec{j}}$ together, Eq. (33) becomes

$$\langle O_{|\mathbb{L}|=l} \rangle = \sum_{k=1}^K A_k (1 - \gamma_k)^l \quad (34)$$

where A_k is the sum of $2 \text{Re} \left\{ \alpha_i^* \alpha_j \text{Tr}(\rho G_i^\dagger O G_j) \right\}$ for some subset of \vec{i} and \vec{j} . Note that A_k are independent of l and we have $\sum_{k=1}^K A_k = \langle O \rangle$.

So far we have only been considering group error channels. However, as shown in Supplementary Note 5, by approximating general Pauli channels as the composition of pure group channels, we can prove that decay of the expectation value under general Pauli noise can also be approximated by a sum of exponentials like in Eq. (34).

Equation (34) can be translated into a multi-exponential decay of $\langle O_\mu \rangle$ over the mean circuit error count μ using Eq. (28)

$$\langle O_\mu \rangle = e^{-\mu} \sum_{l=0}^{\infty} \frac{\mu^l}{l!} \sum_{k=1}^K A_k (1 - \gamma_k)^l \quad (35)$$

$$= \sum_{k=1}^K A_k e^{-\gamma_k \mu}. \quad (36)$$

This can be rewritten as

$$\langle O_\mu \rangle = \sum_{k=1}^K A_k (e^{-\gamma_k})^\mu \approx \sum_{k=1}^K A_k (1 - \gamma_k)^\mu.$$

Comparing with Eq. (34), we see that the shape of $\langle O_\mu \rangle$ and $\langle O_{|\mathbb{L}|=l} \rangle$ are the same up to the leading order of γ_k with the mean circuit error count μ in place of the circuit error count l .

The simplest shape that we can fit over $\langle O_\mu \rangle$ is a single exponential decay curve ($K=1$), which is the one we used in exponential extrapolation. We see here that a natural extension of this will be probing $\langle O_\mu \rangle$ at more than two different error rates and trying to fit them using a sum of exponentials ($K > 1$). The

estimate of the error-free observable $\langle O \rangle$ can then be obtained by substituting $\mu=0$ into our fitted curve.

From Eq. (36), we see that the k th exponential component can only survive up to the mean circuit error count $\mu \sim \frac{1}{\gamma_k}$, thus we can only obtain information about this component by probing at the mean circuit error count $\mu \lesssim \frac{1}{\gamma_k}$. Since $0 \leq \gamma_k \leq 1$, we have $\frac{1}{\gamma_k} \geq 1$ for all k , i.e. we should be able to retrieve all exponential components if we can probe enough points within $\mu \lesssim 1$. In practice, there is a minimum mean circuit error count that we can achieve, which we denote as μ^* . To have an accurate multi-exponential fitting, it is essential that for all components with non-negligible A_k , we have $\mu^* \lesssim \frac{1}{\gamma_k}$, i.e. none of the critical exponential components has died away at the minimal error rate that we can probe.

Numerical simulation for multi-exponential extrapolation

In this section, we will try to apply multi-exponential extrapolation to the Fermi–Hubbard model simulation circuit as outlined in ref. ³, which consists of local two-qubit components that correspond to different interaction terms and the fermionic swaps. It can be used for both eigenstate preparation and time evolution simulation. We will assume there are M of these two-qubit components, and they all suffer from two-qubit depolarising noise of error probability p , which is just a group channel of the two-qubit Pauli group (without the phase factors). Using Eq. (17), we have

$$p_e = \frac{|\mathbb{E}| - 1}{|\mathbb{E}|} p = \frac{15}{16} p \Rightarrow \mu_e = \frac{15}{16} \mu. \quad (37)$$

Since we usually know the number of fermions in the system, we can try to verify the fermion number parity symmetry of the output state, which is simply

$$S_\sigma = \prod_i Z_i$$

in the Jordan–Wigner qubit encoding. All the local two-qubit components in the circuit conserve the symmetry S_σ . Hence when we start in a state with the right fermion number, the output state should also have the correct fermion number, enabling us to perform symmetry verification.

By checking $S_\sigma = \prod_i Z_i$, we can detect all error components with one X or Y in the local two-qubit depolarising channels since they anti-commute with S_σ . We will remove the other error components in the local two-qubit depolarising channels using quasi-probability. The removed components are those can be generated from the set $\mathbb{Q} = \{Z_1, Z_2, X_1 X_2\}$ following Section “Group errors”. Thus we have $|\mathbb{Q}| = 2^3 = 8$ and using Eq. (13) we also have

$$p_d = \frac{|\mathbb{E}| - |\mathbb{Q}|}{|\mathbb{E}|} p = \frac{p}{2} \Rightarrow \mu_d = \frac{\mu}{2}. \quad (38)$$

The resultant noise channel after the application of quasi-probability is given by Eq. (19), which is just a uniform distribution of the two-qubit Pauli errors that are detectable by S_σ . We will call it the detectable noise.

In this section and later in Section “Numerical simulation for a combination of error mitigation techniques”, we will perform numerical simulations using the circuit for the 2×2 half-filled Fermi–Hubbard model, which consists of 8 qubits and 144 two-qubit gates. The two-qubit gates in the circuit that correspond to interaction terms are parametrised gates with the parameters indicating the strength of the interaction. In our simulation, we will obtain the results for a set of randomly chosen gate parameters (with additional results for another set of random parameters listed in Supplementary Note 9). We will also look at two different error scenarios: depolarising errors and detectable errors. One of them is a group channel while the other is a more general Pauli channel. The measurements that we perform will be

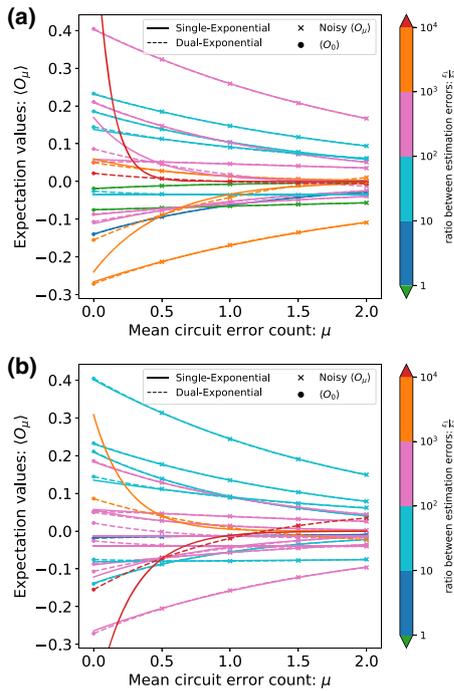


Fig. 1 Comparison between single-exponential extrapolation and dual-exponential extrapolation in an eight-qubit simulation. Plots showing the noisy expectation values of different Pauli observables under **a** depolarising noise and **b** detectable noise obtained at the four mean circuit error counts $\mu = 0.5, 1, 1.5, 2$ (cross markers). The single- and dual-exponential extrapolation curves fitted to the data points are represented by the solid and dashed lines, respectively. The circular markers lie at $\mu = 0$ and denote the true noiseless expectation values. Different colours represent different ratios between the estimation bias of using single- and dual-exponential extrapolation (e.g. orange means that for the given observable, the estimation bias of single-exponential extrapolation is between 10^3 and 10^4 times larger than that of dual-exponential extrapolation).

the Pauli components of the Hamiltonian, from which we can reconstruct the energy of the output state. The simulations are performed using the Mathematica interface¹³ of the high-performance quantum computation simulation package QuEST¹⁴.

In Fig. 1, we have plotted the noisy expectation values $\langle O_\mu \rangle$ for each Pauli observable at the mean circuit error counts $\mu = 0.5, 1, 1.5, 2$. When we perform multi-exponential extrapolation on them, we find that all of the observables can be fitted using a sum of at most two exponentials, even though there should be very few symmetries in our circuits since they are generated from a set of random parameters. We now can proceed to compare the absolute bias in the estimate of the noiseless expectation values ($\mu = 0$) using dual-exponential extrapolation against that using the conventional single-exponential extrapolation for all of the noisy observables in Fig. 1. The absolute biases of single- and dual-exponential extrapolation are denoted as ϵ_1 and ϵ_2 , respectively, and different colours in the plots correspond to different estimation bias ratios $\frac{\epsilon_1}{\epsilon_2}$.

For 32 out of the 34 observables we plotted, dual-exponential extrapolation can achieve a smaller estimation bias than single-exponential extrapolation ($\frac{\epsilon_1}{\epsilon_2} > 1$). Within the two cases that dual-exponential extrapolation is outperformed (the green curves in Fig. 1), one of them is the case in which single- and dual-exponential extrapolation both achieve very small estimation bias of similar order. The other remaining case with larger ϵ_2 relative to ϵ_1 is mainly due to the small magnitude of its true expectation value, which lead to large uncertainties in the fitting parameters. It is also because we are only using the bare minimum of 4 data

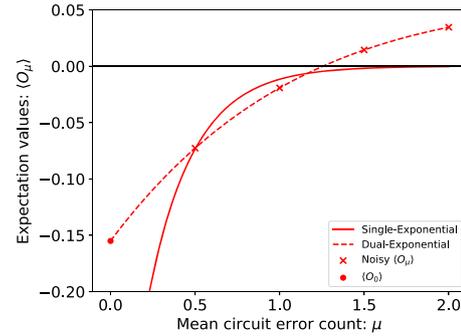


Fig. 2 A noisy Pauli observable from Fig. 1(b) that cannot be fitted well using single exponential decay since it crosses over the x-axis. Here we have plotted the noisy expectation values at the four mean circuit error counts $\mu = 0.5, 1, 1.5, 2$ (cross markers). The single- and dual-exponential extrapolation curves fitted to the data points are represented by the solid and dashed lines, respectively. The circular markers lie at $\mu = 0$ and denote the true noiseless expectation values.

Table 1. The bias in the single- and dual-extrapolation estimates averaged over observables within each plot in Fig. 1 excluding observables with exceptionally large ϵ_1 or ϵ_2 .

$\bar{\epsilon}_1, \bar{\epsilon}_2 / 10^{-4}$	Depolarising	Detectable
Single-exp	150	74
Dual-exp	1.0	1.0

The entries are in the unit of 10^{-4} .

points to fit a dual-exponential curve with 4 free parameters and thus the problem may be alleviated by simply probing at more error rates to obtain more data points. On the other hand, there are also a few cases in which the ϵ_1 are exceptionally large (e.g. certain orange and red curves in Fig. 1a, b). These are usually observables whose decay curves have extrema and/or crossing over the x-axis, thus it is impossible to get a good fit with a single-exponential curve. We have zoomed into one such observable in Fig. 2. For these observables, dual-exponential extrapolation can still perform extremely well and achieve $\epsilon_2 \sim 10^{-5}$, which is up to tens of thousands times lower than ϵ_1 ; $\frac{\epsilon_1}{\epsilon_2} \sim 10^4$.

Now we will exclude the few observables above with exceptionally large ϵ_1 or ϵ_2 and take the average of the remaining ϵ_1 and ϵ_2 to obtain a more representative performance of dual-exponential extrapolation against single-exponential extrapolation. This is shown in Table 1, from which we see that by using dual-exponential extrapolation instead of single-exponential extrapolation, we can achieve a tens or even a hundred times reduction in estimation bias across both noise models. Note that in Fig. 1 it appears to the eye that the true (noiseless) expectation values, marked by filled circles, never deviate from the dual-exponential (dashed) lines. In fact there are minute discrepancies as specified in Table 1, but the extrapolation is remarkably successful.

Combination of error mitigation techniques

We have shown that extrapolating using a multi-exponential curve can be very effective assuming Pauli noise. Besides the shape of the extrapolation curve, the other key component to error extrapolation is the way to tune the noise strength. Previously, noise is boosted by increasing gate pulse duration⁸, applying additional gates that cancel each other¹² or simulating the noise using random gate insertion⁷. There are various practical

challenges associated with these noise-boosting techniques, and furthermore as discussed at the end of Section “Multi-exponential error extrapolation”, data points at boosted error rates may not contain enough information for effective extrapolation. Now as shown in Section “Quasi-probability”, we can actually use quasi-probability to reduce the error rate and obtain a set of data points with reduced noise strength, which can then be used for extrapolation. We should expect a smaller estimation bias by using data points with reduced noise strength rather than boosted noise strength, but the sampling cost will also increase due to the use of quasi-probability. For the special case of two-point extrapolation using a single-exponential curve with the two data points at the unmitigated error rate μ and the quasi-probability-suppressed error rate ν , the total sampling cost as shown in Supplementary Note 7C is

$$C_{QE} \sim 2 \frac{\lambda^2 e^{\lambda(\nu\mu + 2(\lambda-1)\mu_c)} + e^{2\nu\mu}}{(\lambda-1)^2} \quad (39)$$

with $\mu = \lambda\nu$. We will call this special case quasi-probability with exponential extrapolation.

As discussed in Section “Multi-exponential error extrapolation”, the number of exponential components in the multi-exponential extrapolation can be reduced with an increased degree of symmetry in the circuit and/or if the error channels are group channels. Hence, besides using quasi-probability for error suppression, we can also use quasi-probability to transform the error channels into group errors and/or errors of similar form for easier curve fitting in the extrapolation process.

Moving on, we may wish to combine symmetry verification and quasi-probability. We can first apply quasi-probability to transform all the error channels in the circuit with the total mean error count μ into detectable error channels with a total mean error count μ_d . After that, we can apply symmetry verification, but note that the additional quasi-probability operations may contain gates that take us from one symmetry space to another, for which we need to adjust our symmetry verification criterion accordingly. As discussed in Section “NISQ limit”, an even number of occurrence of local detectable errors can still escape the symmetry test and lead to circuit errors. We can further suppress them by applying additional quasi-probability operations as shown in Supplementary Note 6. Alternatively, we can also try to remove these remaining errors by applying error extrapolation as we will see below.

After using quasi-probability to transform all local errors into local detectable errors with the mean circuit error count μ_d , performing symmetry verification will split the circuit runs into two partitions, one has an even number of detectable errors occurring and will pass the symmetry test, the other has an odd number of detectable errors occurring and will fail the symmetry test. Consequently, the noisy observable expectation value (Eq. (35)) can also be split into the weighted sum of these two partitions

$$\langle O_{\mu_d} \rangle = e^{-\mu_d} (\cosh(\mu_d) \langle O_{c,\mu_d} \rangle + \sinh(\mu_d) \langle O_{s,\mu_d} \rangle) \quad (40)$$

in which $e^{-\mu_d} \cosh(\mu_d)$ and $e^{-\mu_d} \sinh(\mu_d)$ are the probability to have an even and an odd number of errors occurring in the circuit, respectively. And $\langle O_{c,\mu_d} \rangle$, $\langle O_{s,\mu_d} \rangle$ are the corresponding expectation values in these cases with

$$\begin{aligned} \langle O_{c,\mu_d} \rangle &= \frac{1}{\cosh(\mu_d)} \sum_{k=1}^K A_k \cosh((1-\gamma_k)\mu_d), \\ \langle O_{s,\mu_d} \rangle &= \frac{1}{\sinh(\mu_d)} \sum_{k=1}^K A_k \sinh((1-\gamma_k)\mu_d). \end{aligned} \quad (41)$$

We will consider the case that the decay of our expectation value $\langle O_{\mu_d} \rangle$ over increased mean circuit error count μ_d follows a

single exponential curve ($K=1$) for simplicity, we then have

$$\begin{aligned} \langle O_{c,\mu_d} \rangle &= \langle O \rangle \frac{\cosh((1-\gamma)\mu_d)}{\cosh(\mu_d)} \\ \langle O_{s,\mu_d} \rangle &= \langle O \rangle \frac{\sinh((1-\gamma)\mu_d)}{\sinh(\mu_d)} \end{aligned} \quad (42)$$

which gives

$$\langle O \rangle = \text{sgn}(\langle O_{c,\mu_d} \rangle) \times \sqrt{\langle O_{c,\mu_d} \rangle^2 \cosh^2(\mu_d) - \langle O_{s,\mu_d} \rangle^2 \sinh^2(\mu_d)}. \quad (43)$$

Note that we have used the fact that $\langle O_{c,\mu_d} \rangle$ and $\langle O \rangle$ have the same sign since $1-\gamma > 0$ and $\mu_d > 0$. Here with the help of symmetry verification and quasi-probability, we can now obtain an estimate of the error-free expectation value $\langle O \rangle$ by combining the expectation value of the passed runs and failed runs at one error rate μ_d instead of combining the expectation value of runs at different error rates in the conventional error extrapolation. Note that here we have assumed that we know the value of the mean detectable circuit error count μ_d , which needs to be known before we can apply the quasi-probability step anyway. The method we employed in Eq. (43) will be called hyperbolic extrapolation.

As derived in Supplementary Note 7D, the sampling cost factor of hyperbolic extrapolation is

$$C_{H,\mu_d} = \cosh(2(1-\gamma)\mu_d) \cosh(\mu_d) e^{\mu_d}. \quad (44)$$

To combine all three error mitigation techniques, we first use quasi-probability to remove the error components that are undetectable by symmetry verifications. Applying symmetry verification will then split the circuit runs into two sets: runs with even number of errors and runs with odd number of errors, obtaining two separate expectation values. Using our understanding about the decay of the expectation value from our study of error extrapolation, we can simply combine these two erroneous expectation values and obtain the error-free expectation value. The full process is called quasi-probability with hyperbolic extrapolation, and the corresponding total sampling cost factor can be obtained using Eqs. (27) and (44)

$$C_{QH}(\gamma) = C_{Q,\mu_d} C_{H,\mu_d} = e^{4\mu_c} \frac{\cosh(\mu_d) \cosh(2(1-\gamma)\mu_d)}{e^{3\mu_d}}. \quad (45)$$

We note that this is always smaller than the cost of pure quasi-probability $C_{Q,0} = e^{4\mu_c}$. Its cost-saving over pure quasi-probability will increase with the increase of γ .

Numerical simulation for a combination of error mitigation techniques

In this section, we will compare the performance of quasi-probability with exponential extrapolation (QE) and quasi-probability with hyperbolic extrapolation (QH) discussed in Section “Combination of error mitigation techniques”. Similar to Section “Numerical simulation for multi-exponential extrapolation”, we will perform Fermi-Hubbard model simulation with local two-qubit depolarising noise with a mean circuit error count μ . The symmetry we will use in QH is the fermionic number parity symmetry, which means that the resultant mean detectable circuit error count after we apply the quasi-probability step in QH will be $\mu_d = \frac{\mu}{2}$ following Eq. (38). For the quasi-probability in QE in this section, we will keep it at the same strength as that in QH, which means that they have the same resultant circuit error rate: $\nu = \frac{\mu}{2}$. Note that even though resultant channels after the partial quasi-probability in both QE and QH give the same mean circuit error count $\nu = \frac{\mu}{2} = \mu_d$, in one case the resultant noise is still depolarising while in the other case the resultant noise is locally detectable. In this section, we will assume the quasi-probability process is performed perfectly. Recall that for simplicity we have only explicitly derived QH under the assumption that the observable follows a single-exponential decay, so for a fair

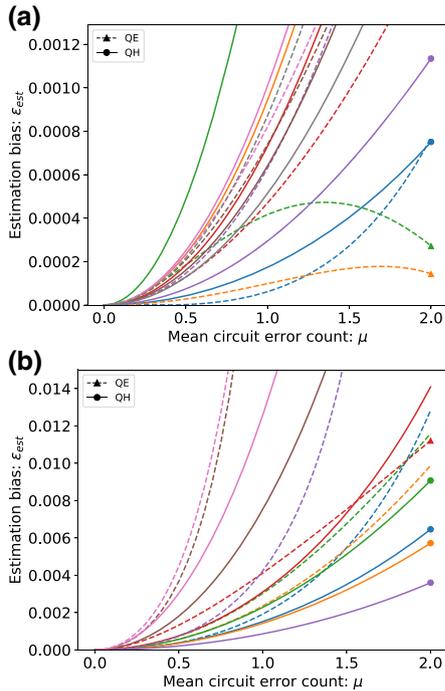


Fig. 3 Comparison of the biases in the error-mitigated expectation values between quasi-probability with exponential extrapolation (QE) and quasi-probability with hyperbolic extrapolation (QH) in an eight-qubit simulation. Plots showing **a** observables following single-exponential decay and **b** observables following dual-exponential decay. Within each plot, different colours represent different observables. The solid lines denote QH, while the dashed lines denote QE. At the mean circuit, error counts $\mu = 2$, for each observable, we use markers to denote the method that has lower estimation bias out of the two. For a given observable, circular markers denote lower estimation bias when using QH, while triangle markers denote lower estimation bias when using QE.

comparison, the QE method in this section will also only employ single-exponential extrapolation. However as we will see later, even when we look at observables that follow a dual-exponential decay, which breaks our assumptions above, QH can still achieve robust performance.

As shown in Fig. 1, for our example circuits, some observables can be fitted well enough using single-exponential decay curves while the other observables can only be fitted well using dual-exponential decay curves. We will call these two types of observables single-exponential observables and dual-exponential observables, respectively. In Fig. 3, we have plotted the absolute estimation bias ϵ_{est} using the two different extrapolation techniques for the single-exponential and dual-exponential observables. First, we can see that the estimation biases for the dual-exponential observables are almost one order of magnitude higher than the biases for the single-exponential observables. This should not come as a surprise since both single-exponential extrapolation and hyperbolic extrapolation are derived under the assumptions of single-exponential observables. At the mean circuit error counts $\mu = 2$, for each observable, we have used markers to label the method that can achieve a lower estimation bias out of the two. We see that the number of single-exponential observables that can achieve a lower estimation bias using QE is comparable to that of QH. On the other hand, almost all dual-exponential observables can achieve a lower estimation bias using QH.

In Table 2, we further calculate the average estimation bias of single-exponential, dual-exponential and all observables

Table 2. The biases in the error-mitigated estimates using QE and QH averaged over single-exponential, dual-exponential and all observables at the mean circuit error count (a) $\mu = 1$ and (b) $\mu = 2$.

a			
$\bar{\epsilon}_{est}/10^{-4}$	1-Exp. Obs.	2-Exp. Obs.	All Obs.
QE	5.1	100	53
QH	7.8	56	32
b			
$\bar{\epsilon}_{est}/10^{-3}$	1-Exp. Obs.	2-Exp. Obs.	All Obs.
QE	1.8	82	39
QH	3.2	20	11

The entries in (a) and (b) are in the units of 10^{-4} and 10^{-3} , respectively.

separately at the circuit error rate $\mu = 1, 2$, which re-confirm all of our observations above. We see that the estimation bias of QE is lower than that of QH for single-exponential observables, and on the other hand, QH can achieve a lower estimation bias for dual-exponential observables. In other words, the performance of QH is more robust against whether the observable is single-exponential or not. When looking at the estimation bias averaged over all observables, we see the estimation bias of QH is always lower than QE and can be four times smaller than QE at $\mu = 2$. The all-observable averages can be more indicative about the practical performance of the mitigation techniques since in experiments we do not know whether a given observable should be fitted with single-exponential or not beforehand.

There is another added layer of robustness when we try to apply QH instead of QE to multi-exponential observables when we look back at the hyperbolic extrapolation equation Eq. (43). We can see that if the shape of the observable is far off from a single-exponential decay, then this might lead to a negative number in the square root of Eq. (43), allowing us to realise that we need to probe at more error rates to perform multi-exponential extrapolation instead, and avoiding performing a bad extrapolation with a very large bias. In the simulation, we indeed identify a few observables that we cannot perform QH on. For these observables, we can still perform QE, but it will lead to huge biases in the estimates. These observables have been excluded in our comparison between QE and QH.

Using Eqs. (45), (37) and (38), the sampling cost factor of performing QH in our example circuit is

$$C_{QH}(\gamma) = e^{3\mu} \cosh\left(\frac{\mu}{2}\right) \cosh((1-\gamma)\mu),$$

where γ is the decay rate of the observable expectation values under noise.

Using Eqs. (39), (37) and $\nu = \mu_d = \frac{\mu}{2} \Rightarrow \lambda = 2$, the sampling cost factor of performing QE is

$$C_{QE} = 2 \left(4e^{(\gamma + \frac{15}{8})\mu} + e^{2\gamma\mu} \right).$$

For comparison purpose, we also write down the sampling cost factor for removing all the errors using quasi-probability given by Eq. (26)

$$C_{Q,0} \approx e^{4\mu_c} = e^{\frac{15}{4}\mu}.$$

The comparison between $C_{Q,0}$, $C_{QE}(\gamma)$ and $C_{QH}(\gamma)$ at different γ is plotted in Fig. 4. We can see that $C_{QH}(\gamma)$ is always lower than $C_{Q,0}$ across all μ and γ , i.e. we can always get a sampling cost saving by applying QH instead of pure quasi-probability, which is also

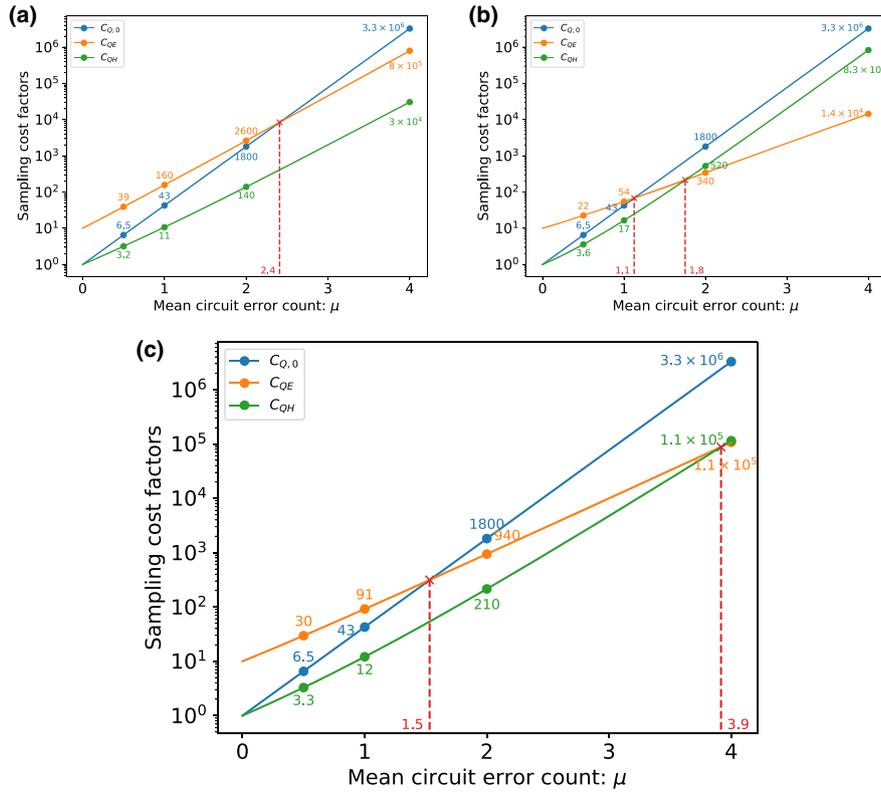


Fig. 4 Comparison of the sampling cost factors between pure quasi-probability (Q), quasi-probability with exponential extrapolation (QE) and quasi-probability with hyperbolic extrapolation (QH). Plots showing different noisy observable decay rate **a** $\gamma = 1$, **b** $\gamma = 0$ and **c** $\gamma = 0.5$. We have labelled the values of the lines at the mean circuit error counts $\mu = 0.5, 1, 2, 4$. We have also labelled the intersects between lines of different methods using red markers.

proven in Section “Combination of error mitigation techniques”. On the other hand, at $\gamma = 1$, C_{QE} is larger than C_{QH} for all μ and larger than $C_{Q,0}$ for $\mu < 2.4$. As γ decreases, $C_{QH}(\gamma)$ will increase while $C_{QE}(\gamma)$ will decrease. Thus they naturally complement each other as QH will be more suitable for large- γ error mitigation while QE will be more suitable for small- γ error mitigation. At $\gamma = 0$, we see that C_{QE} becomes lower than both C_{QH} and $C_{Q,0}$ at $\mu > 1.8$.

The average fitted γ of all the single-exponential observables within each plot in Fig. 1 all lie within the range 0.5–0.6. Hence, we will now focus on the $\gamma = 0.5$ plot in Fig. 4 to get an indication of the practical sampling costs of implementing different mitigation techniques.

At $\mu = 1$, the sampling cost factor of quasi-probability is 43. QE requires a higher sampling cost, thus there is no point performing QE since pure quasi-probability can remove all the noise perfectly in theory with a lower cost. Compared to quasi-probability, QH can reduce the cost by more than 70% while still achieving the small estimation bias $\bar{\epsilon}_{QH} \sim 3 \times 10^{-3}$ shown in Table 2. In order for quasi-probability to have many advantages over QH, we must sample enough times such that the shot noise of pure quasi-probability is smaller than the estimation bias of QH (more rigorous arguments in Supplementary Note 8), which will require $N^* \sim \frac{C_{Q,0}}{\epsilon_{QH}} \approx 4.3 \times 10^6$ samples for each observable. Therefore in practice, QH could be the preferred method over pure quasi-probability as it is challenging to sample more than N^* for each observable within reasonable runtime constraints³.

At $\mu = 2$, now the sampling cost factor of quasi-probability is 1800, which is hardly practical. QE can reduce this sampling cost by half while achieving estimation bias around 4×10^{-2} (Table 2), and QH can reduce this sampling cost by almost 90% while achieving estimation bias around 1×10^{-2} (Table 2), thus they both would be preferred over pure quasi-probability in practice

following similar arguments in the $\mu = 1$ case. We also see that QH outperforms QE in terms of both sampling cost and estimation bias at $\mu = 2$, and thus would be preferred over QE. The cost of QE will only become lower than QH at $\mu = 3.9$, however, at this point, neither of their sampling costs are likely to be practical.

DISCUSSION

In this article, we have recapped and studied the mechanism and performance of three of the most well-known error mitigation techniques: symmetry verification, quasi-probability and error extrapolation under Pauli noise. By introducing the concepts of group errors and NISQ limits, we managed to prove that the change of the expectation value of a Pauli observable with increased Pauli noise strength can be approximated using multi-exponential decay, enabling us to extend exponential error extrapolation to multi-exponential extrapolation. We then performed eight-qubit numerical simulations for Fermi–Hubbard simulation under two different Pauli noise models, finding that the decay of their Pauli expectation values can all be fitted using single- or dual-exponential curves, confirming our earlier proof of multi-exponential decay. Using the same circuits, we performed dual-exponential extrapolation by probing at four different error rates, which is a minimal number of data points required, and managed to obtain a low estimation bias of $\lesssim 10^{-4}$ for almost all 34 observables except for one fringe case. In our simulations, the estimation bias of dual-exponential extrapolation is on average ~ 100 times lower than that of single-exponential extrapolation, with the maximum factor of bias reduction reaching $\sim 10^4$.

We then proceeded to combine different error mitigation techniques in the context of well-characterised local Pauli noise. Instead of using quasi-probability to completely remove all the

noise, we can use it to suppress the noise strength and perform error extrapolation, which is named quasi-probability with exponential extrapolation (QE). Alternatively, we can use quasi-probability to remove the local undetectable noise and perform symmetry verification. Then instead of discarding all the circuit runs that fail the symmetry test, we have developed a way to recombine the expectation values of the “failed” and “passed” runs to obtain an estimate of the noiseless observable. The full combined method is called quasi-probability with hyperbolic extrapolation (QH). Note that both QE and QH are free of the requirement to adjust the hardware error rate despite the name “extrapolation”. By performing eight-qubit Fermi–Hubbard model simulations under local depolarising noise and using the fermionic number parity symmetry, we found that QH outperforms QE in terms of both estimation bias and sampling costs for almost all cases. When compared to pure quasi-probability, QH can achieve factor-of-4 and factor-of-9 sampling cost savings at the mean circuit error count $\mu = 1$ and $\mu = 2$, respectively, while still maintaining a low estimation bias of 10^{-3} – 10^{-2} . Hence, QH would outperform pure quasi-probability in our examples unless we obtain an impractical number of samples (more than millions) per observable.

QH is derived under the assumption that the observables decay along single-exponential curves with increased noise. Our simulation shows that QH can be robust against violation of this assumption when applied to dual-exponential observables. However, such robustness may not persist with a further increase in the number of exponential components. A multi-exponential version of QH can be done through probing at more error rates and fitting Eq. (41) to the data. Alternatively, instead of probing at more error rates, we can also try to verify more symmetries. In such a way, we can obtain a set of expectation values corresponding to different verification syndromes for the multi-exponential version of the hyperbolic fitting. An example can be using the separate fermion number parity symmetries for each spin subspace, which will lead to expectation values corresponding to the four possible verification syndromes. However, how to recombine these expectation values in the case of multiple symmetries and how to use quasi-probability to transform the local error channels into the suitable forms for such a recombination is not a simple extension of the single-symmetry case we considered.

In our derivation, the number of exponential components in the expectation value decay curve in Eq. (34) is expected to scale exponentially with the number of gates. However, in our simulations, we fitted at most two exponential components for each of the observable decay curves. More analysis is needed to bridge the gap between the expected and the actual number of exponential components required, possibly based on the symmetry of the circuit. This will help us understand how the number of exponential components scales with the system size, enabling us to gauge the performance and the costs of scaling up the multi-exponential extrapolation method. It might be useful to draw ideas from non-Clifford randomized benchmarking^{15–17}, in which multi-exponential decay is also employed for the fitting of the fidelity curves. When applying multi-exponential extrapolation in practice, we might want to develop Bayesian methods to determine whether we need to probe at more error rates, which error rates to probe, and whether to change the number of exponential components of our fitted curve based on the existing data. This has been done in the context of randomized benchmarking¹⁸ and it would be interesting to see its performance in the context of multi-exponential extrapolation.

One combination of error mitigation techniques that we have not explored here is pairing symmetry verification with error extrapolation without using quasi-probability. The naive version of such a combination is discussed in ref.³. To make use of the results in this Article, one possible way is to approximate all the local error channels as the compositions of detectable and undetectable error

channels, so that we can deal with them separately using hyperbolic extrapolation and exponential extrapolation. It would be very interesting to see the implementation details of such a method and how it compares to pure error extrapolation.

We have only considered Pauli noise in this Article, thus it will also be interesting to see whether our arguments can be extended to other error channels like amplitude damping or coherent errors. In practice, we can transform any error channels into Pauli channels using Pauli twirling^{19,20} and then apply our methods. Note that we can even perform further twirling like Clifford twirling to transform the error channels into group channels, which can be better mitigated as we have observed. Ways to transform a given error channel into a group channel can be an interesting area of investigation.

DATA AVAILABILITY

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

CODE AVAILABILITY

The code used in the current study is available from the corresponding author upon reasonable request.

Received: 19 October 2020; Accepted: 26 March 2021;

Published online: 25 May 2021

REFERENCES

- Arute, F. et al. Quantum supremacy using a programmable superconducting processor. *Nature* **574**, 505–510 (2019).
- Preskill, J. Quantum Computing in the NISQ era and beyond. *Quantum* **2**, 79 (2018).
- Cai, Z. Resource estimation for quantum variational simulations of the Hubbard model. *Phys. Rev. Appl.* **14**, 014059 (2020).
- Cade, C., Mineh, L., Montanaro, A. & Stanisic, S. Strategies for solving the Fermi–Hubbard model on near-term quantum computers. *Phys. Rev. B* **102**, 235122 (2020).
- McArdle, S., Yuan, X. & Benjamin, S. Error-mitigated digital quantum simulation. *Phys. Rev. Lett.* **122**, 180501 (2019).
- Bonet-Monroig, X., Sagastizabal, R., Singh, M. & O’Brien, T. E. Low-cost error mitigation by symmetry verification. *Phys. Rev. A* **98**, 062339 (2018).
- Li, Y. & Benjamin, S. C. Efficient variational quantum simulator incorporating active error minimization. *Phys. Rev. X* **7**, 021050 (2017).
- Temme, K., Bravyi, S. & Gambetta, J. M. Error mitigation for short-depth quantum circuits. *Phys. Rev. Lett.* **119**, 180509 (2017).
- Endo, S., Benjamin, S. C. & Li, Y. Practical quantum error mitigation for near-future applications. *Phys. Rev. X* **8**, 031027 (2018).
- Greenbaum, D. Introduction to quantum gate set tomography. Preprint at *arXiv* <http://arxiv.org/abs/1509.02921> (2015).
- Kandala, A. et al. Error mitigation extends the computational reach of a noisy quantum processor. *Nature* **567**, 491–495 (2019).
- Giurgica-Tiron, T., Hindy, Y., LaRose, R., Mari, A. & Zeng, W. J. Digital zero noise extrapolation for quantum error mitigation. In *Proc 2020 IEEE International Conference on Quantum Computing and Engineering (QCE)*, 306–316 (2020).
- Jones, T. & Benjamin, S. C. QuESTlink—Mathematica embiggened by a hardware-optimised quantum emulator. *Quantum Sci. Technol.* **5**, 034012 (2020).
- Jones, T., Brown, A., Bush, I. & Benjamin, S. C. QuEST and high performance simulation of quantum computers. *Sci. Rep.* **9**, 1–11 (2019).
- Cross, A. W., Magesan, E., Bishop, L. S., Smolin, J. A. & Gambetta, J. M. Scalable randomised benchmarking of non-Clifford gates. *npj Quantum Inf.* **2**, 1–5 (2016).
- Helsen, J., Xue, X., Vandersypen, L. M. K. & Wehner, S. A new class of efficient randomized benchmarking protocols. *npj Quantum Inf.* **5**, 1–9 (2019).
- Flammia, S. T. & Wallman, J. J. Efficient estimation of Pauli channels. *ACM Trans. Quantum Comput.* **1**, 3:1–3:32 (2020).
- Granade, C. et al. QInfer: statistical inference software for quantum applications. *Quantum* **1**, 5 (2017).
- Wallman, J. J. & Emerson, J. Noise tailoring for scalable quantum computation via randomized compiling. *Phys. Rev. A* **94**, 052325 (2016).
- Cai, Z. & Benjamin, S. C. Constructing smaller Pauli twirling sets for arbitrary error channels. *Sci. Rep.* **9**, 1–11 (2019).

ACKNOWLEDGEMENTS

The author would like to thank Ying Li and Simon Benjamin for reading through the manuscript and providing valuable insights. The author is supported by the Junior Research Fellowship from St John's College, Oxford and acknowledges support from Quantum Motion Technologies Ltd and the QCS Hub (EP/T001062/1).

AUTHOR CONTRIBUTIONS

Z.C. is the sole author of the article.

COMPETING INTERESTS

The author declares no competing interests.

ADDITIONAL INFORMATION

Supplementary information The online version contains supplementary material available at <https://doi.org/10.1038/s41534-021-00404-3>.

Correspondence and requests for materials should be addressed to Z.C.

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

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



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

© The Author(s) 2021