Benchmarks for approximate CNOTs based on a 17-qubit surface code

Andreas Peter,* Daniel Loss, and James R. Wootton

Department of Physics, University of Basel, Klingelbergstrasse 82, CH-4056 Basel, Switzerland

Scalable and fault-tolerant quantum computation will require error correction. This will demand constant measurement of many-qubit observables, implemented using a vast number of CNOT gates. Indeed, practically all operations performed by a fault-tolerant device will be these CNOTs, or equivalent two-qubit controlled operations. It is therefore important to devise benchmarks for these gates that explicitly quantify their effectiveness at this task. Here we develop such benchmarks, and demonstrate their use by applying them to a range of differently implemented controlled gates and a particular quantum error correcting code. Specifically, we consider spin qubits confined to quantum dots that are coupled either directly or via floating gates to implement the minimal 17-qubit instance of the surface code. Our results show that small differences in the gate fidelity can lead to large differences in the performance of the surface code. This shows that gate fidelity is not, in general, a good predictor of code performance.

INTRODUCTION

Two-qubit controlled operations are the workhorses of quantum algorithms. In combination with single qubit rotations they can be used to implement a complete gate set. The most widely considered controlled operations are the controlled-NOT (or CNOT), and gates that are equivalent to it by local unitaries [1].

In practice, the CNOT needs to be implemented using the physical effects provided by a particular system. The interactions that can be directly implemented by a system are compiled into a sequence of gates, designed to create an effective CNOT. These gate sequences will only approximately implement a CNOT, both due to *practical* limitations of experimental control and *fundamental* limitations of approximating a gate by a limited set of other gates. Therefore it is important to assess how the choice of a gate sequence to approximate a CNOT influences the performance of a quantum algorithm.

To determine the quality of a CNOT, the standard means is to calculate the *gate fidelity*. While this provides some insight into how good an approximation a given gate is, there is no information on the effects of the imperfections. This has led to the fidelity being recognized as a not entirely trustworthy means to compare quantum gates [2].

Since one of the main tasks of the CNOT gates is the implementation of quantum error correction, its performance in this context is especially important. Ideally we would assess the performance in error correcting codes that are large enough for practical applications, implementing a set of universal gates that could be applied fault-tolerantly. However, the system sizes that need to be achieved for such a setup are far beyond the abilities of current numerical techniques or experimental setups.

We will therefore consider a minimal working example: The 17-qubit surface code which can both detect and correct quantum errors [3]. The effects of both coherent and incoherent noise can be simulated for this code using a tensor network decomposition. The results provide a

direct insight into the performance of an approximate CNOT in an error correcting code.

IMPLEMENTING CNOTS WITH SPIN QUBITS

In order to provide specific and realistic examples for which calculations can be made, we will restrict ourselves to CNOTs implemented on qubits formed by electrons confined to quantum dots. At the most basic level, a quantum dot is a structure that *confines* electrons to volumes whose lengths are comparable to the wavelengths of the electrons.

In this work we consider *lateral* quantum dots [4], where an electron gas is strongly confined in one dimension, becoming an effectively two-dimensional electron gas (2DEG). This can be achieved at the boundary of two semiconductors with different band gaps and suitable doping [5].

Placing metallic gates on the 2DEG and applying a negative potential on these gates, the electrons can further be confined within the 2DEG to specific islands: the quantum dots. The quantum dots can be loaded with single electron precision and the *electron's spin* can be used to encode a *qubit*.

We consider two ways to perform CNOT gates on these qubits.

Loss-DiVincenzo CNOT

A straightforward way to couple the electrons, and therefore the qubits, in two neighboring quantum dots is *transient Heisenberg coupling* [6]. By lowering the potential between the two quantum dots, the wavefunctions of the qubits overlap which leads to a coupling of the form

$$H_s(t) = J(t)\vec{\sigma}^{(1)} \cdot \vec{\sigma}^{(2)}, \tag{1}$$

where $J(t)$ is a time-dependent exchange constant and $\vec{\sigma}$ are vectors with the Pauli operators as elements where

the superscripts show on which qubit they act.

Applying the gate for a duration τ_s such that $\int_0^{\tau_s} J(t)dt = \frac{\pi}{2}$ results in the application of H_s being equivalent to a $\sqrt{\text{SWAP}}$ gate, i.e. a gate which, when applied twice, interchanges the state of the two qubits.

The $\sqrt{\text{SWAP}}$ gate can be used to construct a CNOT using the sequence

$$\text{CNOT}_{\text{L-DiV}} = \sqrt{\sigma_z}^{(1)}\sqrt{-\sigma_z}^{(2)}\sqrt{\text{SWAP}}\,\sigma_z^{(1)}\sqrt{\text{SWAP}}. \quad (2)$$

The required single qubit rotations can be achieved in the quantum dot by applying appropriate magnetic fields and using electron spin resonance [7].

As written in Equation 2, the CNOT is exact. For a more realistic analysis, we replace the unitary H_s in Equation 2 with a *quantum channel* that takes interactions with the environment during the application of H_s into account.

Using Markov and Born approximations, the channel

$$\mathcal{V}(t) = \exp[-(t - \tau_s)\mathcal{K}_3]\mathcal{U}_s(\tau_s)(\mathbb{1} - \mathcal{K}_2), \quad (3)$$

is derived in [6]. In Equation 3, τ_s is the time the Hamiltonian Equation 1 is applied, $t > \tau_s$ is the total time considered and $\mathcal{U}_s$, $\mathcal{K}_3$, and $\mathcal{K}_2$ are the channels for the unitary application of H_s, effect of the environment during the application of H_s and the correction to the initial state due to the environment, respectively.

Cast into the form of matrices, the channel in Equation 3 can be expressed as a function of three parameters: t, Γ and Δ, with Γ being the spin relaxation rate and Δ a phase shift. Both are functions of the coupling strength between spin qubit and environment. For the derivation and more details see [6].

This implementation of the CNOT requires short-range coupling, and a technically challenging close packing of the qubits.

Floating Gate CNOT

These challenges can be avoided using other methods to implement the CNOT, such as the capacitive coupling of quantum dots via floating gates [8].

This implementation takes advantage of electron spins coupling electrostatically, since the electrons feel each others charges. Specifically, the charge degree of freedom is coupled to the orbital motion, and the orbital motion is coupled to the spin via the *spin-orbit interaction* (SOI) [9].

The charges are screened by gates and the 2DEG between quantum dots, and so electrostatic coupling cannot be used directly for long-distance coupling. Instead it can be mediated by floating gates: two flat metallic disks connected by a thin metallic wire. Placing connected disks near a quantum dot each, the mirror charge induced in either disk by one electron charges the other

disk with opposite sign and acts on the other electron. The resulting coupling decays only weakly with the wire distance, enabling long-distance coupling and loosening the requirements on the quantum dot arrangements.

A complete analysis of the electrostatic forces, see [8] for details, yields an effective Hamiltonian of the form

$$H = E_z(\sigma_z^{(1)} + \sigma_z^{(2)}) + J_{12}(\vec{\sigma}^{(1)} \cdot \vec{\gamma})(\vec{\sigma}^{(2)} \cdot \vec{\gamma}), \quad (4)$$

where J_{12} is a coupling constant that depends on both magnetic field and the details of the metallic gate and

$$\vec{\gamma} = (\alpha_D \cos(2\gamma), -\alpha_R - \alpha_D \sin(2\gamma), 0), \quad (5)$$

with α_R and α_D being the *Rashba* and *Dresselhaus* SOI-strength, respectively, and γ is the angle between the crystallographic axis along the [100] direction and the axis along the wire.

The Hamiltonian in Equation 4 can be approximated by

$$H' = \frac{J_{12}}{2}|\gamma_x|^2\left(\sigma_x^{(1)}\sigma_x^{(2)} + \sigma_y^{(1)}\sigma_y^{(2)}\right) + E_z(\sigma_z^{(1)} + \sigma_z^{(2)}), \quad (6)$$

if $E_z \gg J_{12}|\gamma_x|^2$ and assuming that the magnetic field is perpendicular to the 2DEG.

We can use H' to implement a $\sqrt{\sigma_x\sigma_x}$ gate in two ways, with two and four applications of H', respectively:

$$\sqrt{\sigma_x\sigma_x}_{v1} = \exp[\imath(\sigma_z^{(1)} + \sigma_z^{(2)})E_zt]\exp[-\imath H't]\sigma_x^{(1)} \quad (7)$$
$$\exp[\imath(\sigma_z^{(1)} + \sigma_z^{(2)})E_zt]\exp[-\imath H't]\sigma_x^{(1)}$$
$$\sqrt{\sigma_x\sigma_x}_{v2} = \sigma_x^{(2)}\exp[-\imath H'\tfrac{t}{2}]\sigma_x^{(1)}\sigma_x^{(2)}\exp[-\imath H'\tfrac{t}{2}] \quad (8)$$
$$\sigma_x^{(2)}\exp[-\imath H'\tfrac{t}{2}]\sigma_x^{(1)}\sigma_x^{(2)}\exp[-\imath H'\tfrac{t}{2}],$$

where the application time t is given by

$$t = \frac{\pi}{4J_{12}\left(\gamma_x^2 + \gamma_y^2\right)}. \quad (9)$$

Given an implementation for $\sqrt{\sigma_x\sigma_x}$ it is straightforward to implement a CNOT as

$$\text{CNOT} = \sqrt{\sigma_z}^{(1)}\sqrt{\sigma_x}^{(2)}\text{H}^{(1)}\sqrt{\sigma_x\sigma_x}\,\text{H}^{(1)}, \quad (10)$$

where H is the *Hadamard* gate.

This implementation is only exact for H'. For the real Hamiltonian of Equation 4, the gate, while unitary, will not exactly be a CNOT but instead an approximate channel CNOT$_{v1}$ or CNOT$_{v2}$ for two and four Hamiltonian applications, respectively. To define the nature of these approximate CNOTs it is only necessary to choose the parameters of the floating gate Hamiltonian.

17-QUBIT SURFACE CODE

To protect quantum information against physical noise induced by an environment, *quantum error correcting*

codes can be used [10]. For our purposes we use nine *physical data* qubits in whose Hilbert space we encode one *logical* qubit. The nine data qubits are arranged on a 3×3 lattice as the points labeled 9 to 17 in Figure 1. These qubits span a 2^9-dimensional Hilbert space.
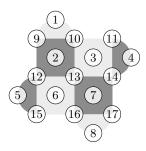

FIG. 1: A surface code defined on a 3×3 array of physical data qubits. The numbering used for the qubits in this work is shown.

The 17 qubit surface code, $\mathcal{S}_{17}$, is defined by the set of eight mutually commuting *stabilizer* operators

$$\mathcal{S}_{17} = \{X_2 = \sigma_x^{(9)}\sigma_z^{(10)}\sigma_z^{(12)}\sigma_x^{(13)},$$
$$X_7 = \sigma_x^{(13)}\sigma_z^{(14)}\sigma_z^{(16)}\sigma_x^{(17)}, X_4 = \sigma_x^{(11)}\sigma_z^{(14)},$$
$$X_5 = \sigma_z^{(12)}\sigma_x^{(15)}, Z_3 = \sigma_x^{(10)}\sigma_z^{(11)}\sigma_z^{(13)}\sigma_x^{(14)},$$
$$Z_6 = \sigma_x^{(12)}\sigma_z^{(13)}\sigma_z^{(15)}\sigma_x^{(16)}, Z_1 = \sigma_z^{(9)}\sigma_x^{(10)},$$
$$Z_8 = \sigma_x^{(16)}\sigma_z^{(17)}\}, \quad (11)$$

where the operators are grouped into X and Z operators. We can restrict data qubits to be in the mutual $+1$ eigenstate of all operators in $\mathcal{S}_{17}$. The resulting Hilbert space — the *code space* — is $2^{9-8} = 2$-dimensional and can encode *one* logical qubit.

We define an operator $Z_L = \sigma_x^{(10)}\sigma_z^{(13)}\sigma_x^{(16)}$ that commutes with all stabilizers of $\mathcal{S}_{17}$. Identifying it with the logical Z-operator lets us define the logical $|0\rangle$ and $|1\rangle$ states as the $+1$ and -1 eigenstates of Z_L in the code space, respectively. The logical X-operator is $X_L = \sigma_z^{(12)}\sigma_x^{(13)}\sigma_z^{(14)}$ which commutes with all elements of $\mathcal{S}_{17}$ and anticommutes with Z_L. Since stabilizers do not change logical states, the logical operators are only unique up to multiplication with stabilizers.

$\mathcal{S}_{17}$ is an example of a *surface code* which belongs to the class of *stabilizer codes*. For detailed information on both topics see references [10] and [?], but for our purpose it is enough to note that $\mathcal{S}_{17}$ is the smallest useful realization of a surface code considering both number of qubits and number of operations.

Having defined the logical state and the code space with $\mathcal{S}_{17}$, it is easy to see that any single-qubit error on one of the data qubits will be projected to a Pauli operator upon measurement of a stabilizer and any Pauli operator on one of the qubits anticommutes with at least

one stabilizer. Thus by measuring the stabilizers, single qubit errors can be *detected*.

To enable measurement without genuine 4-qubit interactions, we use 8 *ancilla* qubits — one associated with each stabilizer. The ancillas and their corresponding stabilizers are shown in Figure 1 as the qubits 1 to 8 and the plaquettes they are on, respectively. The ancillas are entangled to each qubit their respective stabilizers act on non-trivially such that their measurement is equivalent to the measurement of the stabilizer.

As an example, consider the indirect measurement of X_7 shown in Figure 2 via an entangled ancilla qubit requiring only two-qubit gates — CNOTs — and one-qubit measurement. The sequence is chosen such that the concurrent measurement of all stabilizers with the same sequence does not move errors around but note that with concurrent measurements, Hadamards would need to be applied directly before and after the CNOTs.
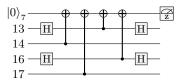

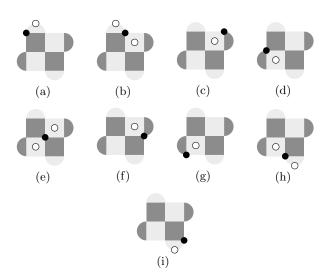FIG. 2: Measurement of $X_7 \in \mathcal{S}_{17}$ via ancilla 7.

Decoding

FIG. 3: The syndrome for all one-qubit errors that $\mathcal{S}_{17}$ can detect by Z-stabilizer measurement. Errors are shown as black dots, -1 syndromes as white dots with black boundaries. The graphic for X-stabilizers is the same with syndrome and errors rotated by 90 degrees.

From the previous section it is clear that $\mathcal{S}_{17}$ can *detect* any one-qubit error on one of the 9 data qubits. The

measurement of the stabilizers gives us 8 bits of information which, adding the measurement result of a logical operator, we bundle together as the *syndrome*. Taking a syndrome and extracting the most likely logical state is the process of *decoding*, which enables us to *correct* errors.

In Fig. 3 we can see the syndrome resulting from each one-qubit error detectable by Z-stabilizers. The error, which is projected to a Pauli operator, is detected by the part of the stabilizer which anticommutes with its effect. While some errors lead to a unique syndrome and can be undone straight-forwardly, others, such as 3c and 3f, cause the same syndrome. It is, however, easy to see that it does not matter which one of the possible errors are undone since the corrections of both errors are equivalent up to stabilizers.

For more than one Pauli error, decoding might fail to reveal the correct logical state. This is clear since logical operators themselves can be mimicked by three Pauli errors, yet are invisible to stabilizer measurements.

The above description relies on the measurements being perfect which implies both perfect entangling gates — CNOTs — and noiseless ancillas. In a physical implementation these assumptions are clearly not realistic.

To enable error correction under these conditions, multiple measurement rounds in quick succession are necessary to construct a three-dimensional syndrome which contains information about both data and ancilla qubit errors — the latter showing themselves as successive flips of the syndrome.

The final syndrome measurement round is done at the time of readout, using the results from direct measurement of the data qubits. Any imperfections in the measurement for this can be modeled as additional noise preceding a perfect measurement. This final, effectively perfect, syndrome measurement round ensures that it is possible that single qubit measurement noise is detected and corrected.

For large codes — in time, space or both — the decoding requires sophisticated algorithms [11]. However, the system considered is compact enough that a simple and optimal procedure based on lookup-tables is possible [12].

The lookup-table results from many runs of a surface code for multiple possible encoded states, obtained either experimentally or from simulations. The final result $\boldsymbol{s}$, which is comprised of the syndrome and the result of a final logical measurement, is recorded for each encoded state $|\phi\rangle$. This data is then used to calculate the conditional probabilities, $p_{|\phi\rangle}(\boldsymbol{s})$, which form the lookup-table.

If we are interested to measure in the computational basis, we take $|\phi\rangle = |0\rangle$ and $|\phi\rangle = |1\rangle$. A state can then be corrected by taking the syndrome and choosing whichever logical state is more likely given that syndrome according to the conditional probabilities calculated beforehand.

The associated computational basis state of the prob-

ability chosen is then the *best possible guess* as to which state the system should be in without the effect of noise, e.g. after a computation. Correction can then be applied to match the measured logical state with the probable logical state [13].

The lookup-table not only provides us the best guess for the *true* state of a system but also its probability and thus the probability of that guess being wrong.

Note that the $\mathcal{S}_{17}$ code is able to detect and correct all forms of single qubit error that can occur during its implementation, whether they be from interaction with the environment, imperfection in measurements or faulty implementation of gates. However, it is the last of these that we will primarily focus on in this work. Our main benchmarks will be measures of how well a code performs when the main noise present is that associated with the implementation of a CNOT.

CALCULATIONS

Fidelity

To assess the quality of an approximate CNOT we can use the *gate fidelity* of approximate and exact CNOT defined as

$$F(\mathcal{E},\mathcal{C}) = \min_{|\psi\rangle} F\big(\mathcal{E}(|\psi\rangle\langle\psi|),\mathcal{C}(|\psi\rangle\langle\psi|)\big), \qquad (12)$$

where $\mathcal{E}$ and $\mathcal{C}$ are the channels for the approximate and exact CNOT, respectively, and the minimum is taken over all pure states $|\psi\rangle$. The fidelity F for two states ρ and σ is defined as

$$F(\rho,\sigma) = \mathrm{Tr}\left(\sqrt{\sqrt{\rho}\,\sigma\sqrt{\rho}}\right). \qquad (13)$$

F is symmetric in its inputs, invariant under unitary transformations, and obeys $0 \leq F(\rho,\sigma) \leq 1$, where equality with 0 implies orthogonal support while equality with 1 implies equality of states. Additionally, Equation 13 obeys *strong concavity* which implies that the minimization need only be performed over pure states.

To limit computational cost we restrict the calculation further and only consider the tensor products of eigenvectors of the Pauli operators. Thus our result is strictly speaking only an upper bound on the gate fidelity but more elaborate searches showed little difference with our calculated fidelities.

Performance in a Surface Code

To assess the performance of an approximate CNOT in a surface code, we came up with two scenarios which we can handle numerically and deem realistic for experimental implementation in the near future.

For both scenarios we choose the probability of failed decoding in $\mathcal{S}_{17}$ as the measure of its usefulness which can be calculated using a lookup-table as explained above. To arrive at the lookup-table we calculate the probabilities of all possible syndromes conditional on the initial logical state.

From the lookup-table it is straightforward to extract the probability of a logical error by summing over the probabilities of all wrong decodings. The result is the probability of wrong decoding or failed error correction.

For noise we consider, test calculations showed that there was not a significant difference between results for the initial logical state being in either of the computational basis states. We therefore consider only initial logical state $|0\rangle$.

Noise Model

In the context of quantum computation, any effects from the environment outside our direct control must be interpreted as *noise*; the environment changes our carefully crafted quantum states in unwanted ways.

Simulation of noise would ideally be done by determining the full time evolution of both system and environment. The environment could then be traced out to obtain a final noise channel $\mathcal{E}$ which maps the state at an initial time t_0 to that at the later time t,

$$\rho_{\mathrm{s}}(t) = \mathcal{E}(\rho_{\mathrm{s}}(t_0)).$$

Such a full simulation is not feasible for realistic noise, and so determining $\mathcal{E}$ usually requires strong assumptions about the nature of the environment and its entanglement with the system.

We will consider the representation of noise channels in terms of Kraus operators. We therefore consider representations in terms of operators K_j such that

$$\begin{aligned}
\rho_{\mathrm{s}}(t) &= \mathcal{E}(\rho_{\mathrm{s}}(t_0)) \\
&= \sum_j K_j \, \rho_{\mathrm{s}}(t_0) \, K_j^\dagger.
\end{aligned} \tag{14}$$

For our calculations we consider bit and phase flip and depolarizing noise. While the former are primarily toy models that are easy to parametrize, the latter provides a more realistic picture with parameters that depend on physical considerations.

In terms of Kraus operators, bit and phase flip are represented as

$$K_1^{\mathrm{bit}} = \sqrt{1-p}\,\mathbb{1}, \qquad K_2^{\mathrm{bit}} = \sqrt{p}\,\sigma_{\mathrm{x}} \tag{15}$$

and

$$K_1^{\mathrm{phase}} = \sqrt{1-p}\,\mathbb{1}, \qquad K_2^{\mathrm{phase}} = \sqrt{p}\,\sigma_{\mathrm{z}}, \tag{16}$$

respectively. As expressed in Equation 15 and Equation 16, the two errors flip the state or the phase with a probability p and are related by a simple base-change via a Hadamard.

Depolarization describes a type of error that erases *all* information about a state with a probability p, leaving us with the completely mixed state, or leave the state undisturbed. It can be represented by a channel

$$\mathcal{E}(\rho) = (1 - p_{\mathrm{depol}})\rho + \frac{p_{\mathrm{depol}}}{2}\mathbb{1}. \tag{17}$$

Equation 17 can be expressed in terms of Kraus operators as

$$K_1 = \sqrt{1 - \tfrac{3}{4}p_{\mathrm{depol}}}\,\mathbb{1}, \qquad K_i = \sqrt{\tfrac{p_{\mathrm{depol}}}{4}}\,\sigma_{\mathrm{i}}, \tag{18}$$

where i goes from 2 to 4 and over the Pauli operators, respectively.

The channels used for the CNOTs will also be brought into the form of Kraus operators. The effects of these are then simulated using tensor network techniques [13].

Our main aim in this study is to investigate how well the gate fidelity captures the performance of a CNOT in an actual surface code. Parameters for CNOTs have therefore been chosen such that the *gate fidelity* of the CNOTs have a nontrivial dependence on the parameters.

For the *floating gate* CNOTs, we choose as parameters the ratio between Zeeman energy and coupling constant $R = E_z / \left(J_{12}\gamma_x^2 \right)$ and $\gamma = \gamma_y/\gamma_x$. We look at $R \in (30, 35)$ and $\gamma \in (0, 1)$ since for these parameters the gate fidelity of both CNOT$_{\mathrm{v1}}$ and CNOT$_{\mathrm{v2}}$ shows interesting dependence on the parameters.

For the Loss-DiVincenzo CNOT we use as parameters the time since the beginning of the interaction $t > 1$ in units of the interaction time τ_{s} and the decoherence parameter Γ. We look at $t \in (1, 1.1)$ and $\Gamma \in (0.007, 0.027)$ to stick close to the values used in [6] and keep Δ fixed at -0.0145.

Scenarios

Scenario I

The surface code $\mathcal{S}_{17}$ is initialized as an eigenstate of the Z stabilizers (see Equation 11), which can be achieved by starting in a product state of $|0\rangle$ and subsequently applying Hadamards on qubits 10, 12, 14, and 16. An initial round of noise is applied to *all* qubits as *depolarizing noise* with strength $p_{\mathrm{depol}} = p_{\mathrm{init}}$ to mimic the preparation noise of the experiment.

Using the CNOT chosen, one round of syndrome measurement is applied where Hadamards are applied to data qubits before and after measurement if they are to be measured in the σ_{x} basis. The ancillas of the Z-stabilizers are measured and for a final readout each data
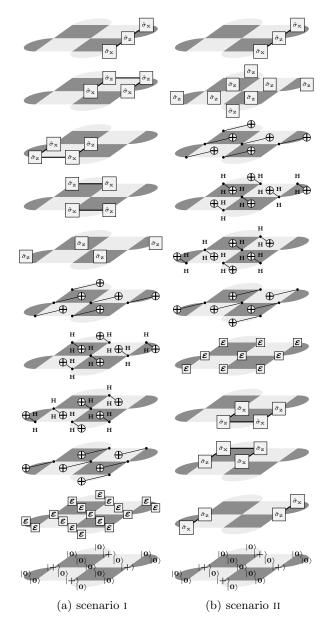
(a) scenario I   (b) scenario II

FIG. 4: A depiction of the steps of the two scenarios used to assess the effectiveness of CNOTs. The boxes with Paulis are projectors such that the projectors after (above) the CNOTs correspond to measurements with the outcome determined by the projections. The corresponding probability of the measurement can then be extracted by the norm of the state.

qubit is measured in the basis that commutes with the Z-stabilizers acting on it.

The results of these measurements are then used to deduce the values of the Z-stabilizers, effectively forming a final round of syndrome measurement, as well as the final value of the Z logical operator. All other information is discarded.

The results are therefore made up of eight bits of syndrome information from two rounds, and one bit of logical information. From this syndrome we can calculate the conditional probabilities we need to get the code performance for scenario I.

This is the same setup we considered in [12], where we considered exact CNOTs affected by different single-qubit noise channels.

Scenario II

The first scenario gives us information about just one type of stabilizers, measured once indirectly and once directly. To extract information about the effect of using both sets of stabilizers and within the constraints of our computational resources, scenario II includes a full measurement of all stabilizers via the ancillas but no final direct stabilizer measurement as in scenario I. $\mathcal{S}_{17}$ is initialized as a logical state, i.e. as a mutual $+1$ eigenstate of all operators in $\mathcal{S}_{17}$. This is achieved starting in the same eigenstate of all Z operators as in scenario I and subsequent application of projectors into X-stabilizer eigenstates.

Preparation noise is applied to the data qubits as depolarizing noise as before with strength p_{init}. After that we apply one round of syndrome measurement using the chosen CNOTs. This round ends with the measurement of *all* ancillas to get an eight bit syndrome.

The final readout consists simply of the logical Z measurement. All other information is discarded. The results are therefore again made up of eight bits of syndrome information, and one bit of logical information.

Since we only measure one round of ancillas, we can not correct ancilla errors. This is the reason preparation noise is only applied to the data qubits. All ancilla errors will then be due to the imperfect channels that act as noisy CNOTs.

RESULTS

Tensor network techniques were used to calculate the probabilities of each possible syndrome in the two scenarios, inspired by the approach in [13]. From these simulations, the probability of a logical error, p_{code}, was calculated for both scenarios and all CNOTs described above. This allowed us to determine how p_{code} depends on the imperfections of the CNOT gates used, and how this relates to the fidelity.

Results for Scenario I

We will now look at the code performance as a function of the gate infidelities, i.e. $(1 - \text{fidelity})$ of the CNOTs, focusing first on scenario I. To do this we construct CNOTs

using multiple different coupling parameters. For each we calculate both p_{code} and the gate infidelities of the CNOTs, and plot these against each other to determine the dependence of the code performance on gate infidelity.

The dependence of p_{code} on the infidelity of a CNOT is shown in Fig. 5, Fig. 6, and Fig. 7 for CNOT_{v1}, CNOT_{v2}, and $\text{CNOT}_{\text{L-DiV}}$, respectively.
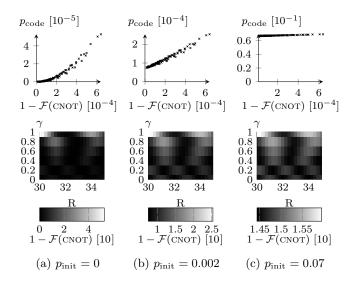
(a) $p_{\text{init}} = 0$ (b) $p_{\text{init}} = 0.002$ (c) $p_{\text{init}} = 0.07$

FIG. 5: Plots of p_{code} for scenario I versus the infidelity for CNOT_{v1}. Different curves correspond to different values of p_{init}.

(a) $p_{\text{init}} = 0$ (b) $p_{\text{init}} = 0.007$ (c) $p_{\text{init}} = 0.014$
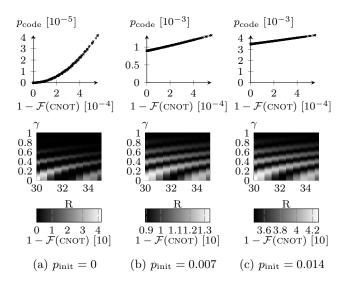
FIG. 6: Plots of p_{code} for scenario I versus the infidelity for CNOT_{v2}. Different curves correspond to different values of p_{init}.

For CNOT_{v1}, the plots of code performance against gate infidelity show a quadratic dependence for low p_{init}. In this case, imperfections in the CNOTs provide the primary source of noise. The quadratic dependence is due to the
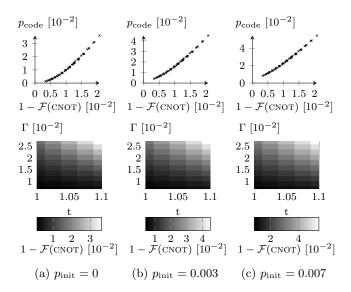

(a) $p_{\text{init}} = 0$ (b) $p_{\text{init}} = 0.003$ (c) $p_{\text{init}} = 0.007$

FIG. 7: Plots of p_{code} for scenario I versus the infidelity for $\text{CNOT}_{\text{L-DiV}}$. Different curves correspond to different values of p_{init}.

fact that at least two errors need to occur for a logical error.

The dependence becomes linear by at least $p_{\text{init}} = 0.002$, with a value of $p_{\text{code}} \sim 10^{-4}$ at zero gate infidelity. The latter feature is due to logical errors occurring caused entirely by initialization noise. The former is due to logical errors composed of a single initialization error and a single error from a CNOT. There is no longer any hint of the quadratic dependence, showing that logical errors due to only to pairs of CNOT errors have become rare in comparison.

By $p_{\text{init}} = 0.07$, any dependence of p_{code} on the gate infidelity has essentially disappeared. The initialization noise is therefore the dominant source of logical errors in this case.

For each value of the infidelity, there is a range of different p_{code} that can result. This is most easily seen for $p_{\text{init}} = 0$. For low infidelity, the corresponding values of p_{code} can differ by as much as a factor of two. This factor decreases for larger infidelity, but still remains sizable.

This shows that, if we know only the gate infidelity for a CNOT_{v1} gate, we cannot form a highly accurate prediction of the p_{code} it will realize, since the specific structure of the infidelity will also play a role. Instead, the infidelity can only provide a lower-bound estimator for the code performance.

The results for CNOT_{v2} show similar qualitative features. The main difference is that the range of possible p_{code} for any given infidelity is much smaller in this case. For this type of gate, the fidelity is therefore a better indicator of code performance.

For the coupling parameters we consider, the results for $\text{CNOT}_{\text{L-DiV}}$ cover a range of much higher gate infidelities.

Much higher values are therefore also seen for p_{code}. The loss of the quadratic scaling therefore occurs much earlier. The spread of the curve is found to be quite small in this case, and so the fidelity is found to be a good indicator of the code performance for such a CNOT.

Results for Scenario II

We will now look at the code performance as a function of the gate infidelities for scenario II. The dependence of p_{code} for this case on the infidelity of a CNOT is shown in Fig. 8, Fig. 9 and Fig. 10 for CNOT_{v1}, CNOT_{v2} and $\text{CNOT}_{\text{L-DiV}}$, respectively.

(a) $p_{\text{init}} = 0$ (b) $p_{\text{init}} = 0.002$ (c) $p_{\text{init}} = 0.07$
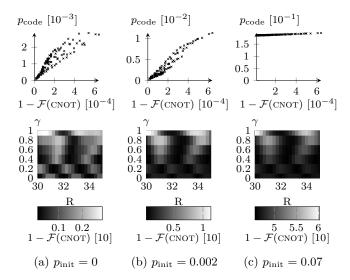
FIG. 8: Plots of p_{code} for scenario II versus the infidelity for CNOT_{v1}. Different curves correspond to different values of p_{init}.

For this scenario, a logical error can already occur with only single qubit errors. It is therefore a linear dependence of p_{code} on gate infidelity that can be expected when this noise is dominant.

As with scenario I, the dependence of the code performance on the coupling parameters still generally mirrors the dependence of the gate infidelity on the parameters, but it is not a highly accurate predictor. In fact, we find a much greater spread of different values of code performance for a given gate fidelities for scenario II.

For CNOT_{v1}, this effect is most pronounced for low p_{init}. As the initial noise increases, the relative importance of the CNOT diminishes yet again and we recover a weak linear dependence on the gate fidelity for high initial noise.

For CNOT_{v2} we find that the greatest spread in code performance is not for vanishing initial noise, as with CNOT_{v1}. Instead it reaches a maximum around $p_{\text{init}} \approx 0.007$.

Especially interesting in both cases is that the worst code performance is not always achieved for the highest
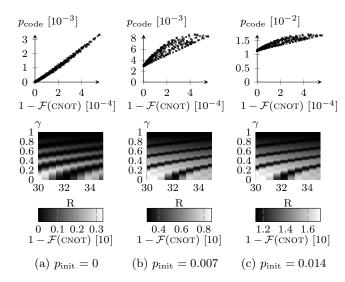
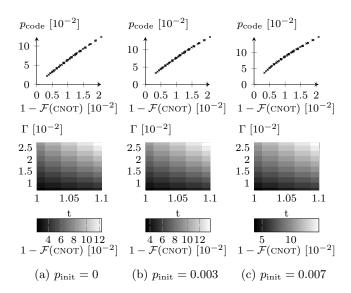(a) $p_{\text{init}} = 0$ (b) $p_{\text{init}} = 0.007$ (c) $p_{\text{init}} = 0.014$

FIG. 9: Plots of p_{code} for scenario II versus the infidelity for CNOT_{v2}. Different curves correspond to different values of p_{init}.

(a) $p_{\text{init}} = 0$ (b) $p_{\text{init}} = 0.003$ (c) $p_{\text{init}} = 0.007$
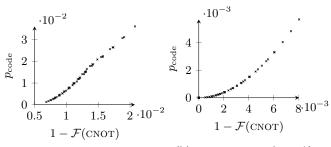
FIG. 10: Plots of p_{code} for scenario II versus the infidelity for $\text{CNOT}_{\text{L-DiV}}$. Different curves correspond to different values of p_{init}.

gate infidelities. These results therefore serve as a powerful demonstration of cases for which gate fidelity serves as a poor indicator of code performance.

The results for $\text{CNOT}_{\text{L-DiV}}$ do not show such features. The value of p_{code} simply increases with the linear scaling expected for scenario II, and with little spread. The dependence on p_{init} is minimal, since the gate infidelity is the dominant source of noise.

Effects of the $\mathcal{K}_2$ Term

For the noisy implementation of the Heisenberg coupling, such as that in CNOT$_\text{L-DiV}$, the effects of the $\mathcal{K}_2$ term are often ignored for simplicity [14]. To assess the impact of this approximation, we consider the effects of the noise on the code both with and without this term. The results are shown in Fig. 11.

(a) CNOT$_\text{L-DiV}$ with $\mathcal{K}_2$ term.

(b) CNOT$_\text{L-DiV}$ without $\mathcal{K}_2$ term.

FIG. 11: p_code for scenario I as a function of the all channel infidelities found within the parameter regime studied for CNOT$_\text{L-DiV}$.

From these results it is evident that the $\mathcal{K}_2$ term has a significant effect. Within the parameter regime studied, infidelities are found as high as 2×10^{-2} with the $\mathcal{K}_2$ term, but only as high as 8×10^{-3} without it. Though these values are of the same order, the difference is nevertheless significant.

Only a few values for the results with $\mathcal{K}_2$ fall within the same range as those without it. In each case we find that, for a given fidelity, CNOT$_\text{L-DiV}$ without $\mathcal{K}_2$ has a larger p_code than that with.

We therefore find that neglecting the $\mathcal{K}_2$ term leads to a significant overestimate of the fidelity of an instance of CNOT$_\text{L-DiV}$. However, we also note that CNOTs of a given fidelity obtained without the $\mathcal{K}_2$ term have noise that is more damaging to the surface code than one with it.

CONCLUSIONS

In this work we proposed benchmarks for approximate CNOT gates. These directly assess their capability to implement the surface code, and do so using a system small enough to allow straightforward calculation.

The two measures of CNOT performance proposed were then calculated and compared to the standard gate infidelity for a range of different implementations of the CNOT with spin qubits.

Our results showed that the infidelity cannot be regarded as a good predictor of code performance, beyond a simple rule of thumb. We also found very different behaviour for both of the surface code tasks, showing that our metrics give different perspectives on the effectiveness of a CNOT within the code. This shows that, even within the limited realm of tasks involving the surface code, the quality of a CNOT cannot be easily reduced to a single number.

Our benchmarks are defined by assessing the response of the surface code to specific sources of noise. This is primarily the noise that arises during the implementations of the CNOTs. Since these conditions are not easily replicable experimentally, our benchmarks are primarily designed to be calculated using either a theoretical description of the imperfections within a CNOT, or data from a quantum process tomography. This makes our approach applicable for a wide range of theoretically described or experimentally implemented CNOTs, regardless of whether they could be used in a 17-qubit surface code in the near term.

* andreas.peter@strath.ac.uk

[1] M. Nielsen and I. Chuang, *Quantum Computation and Quantum Information*, Cambridge Series on Information and the Natural Sciences (Cambridge University Press, 2000), ISBN 978-0-521-63503-5.

[2] S. Bravyi, M. Englbrecht, R. Koenig, and N. Peard, arXiv:1710.02270 [quant-ph] (2017), arXiv: 1710.02270, URL http://arxiv.org/abs/1710.02270.

[3] Y. Tomita and K. M. Svore, Phys. Rev. A **90**, 062320 (2014), URL https://link.aps.org/doi/10.1103/PhysRevA.90.062320.

[4] L. P. Kouwenhoven, C. M. Marcus, P. L. McEuen, S. Tarucha, R. M. Westervelt, and N. S. Wingreen, in *Mesoscopic electron transport* (Springer, 1997), pp. 105–214.

[5] B. Etienne and E. Paris, Journal de Physique **48**, 2049 (1987).

[6] D. Loss and D. P. DiVincenzo, Phys. Rev. A **57**, 120 (1998), URL https://link.aps.org/doi/10.1103/PhysRevA.57.120.

[7] C. Kloeffel and D. Loss, Annual Review of Condensed Matter Physics **4**, 51 (2013), URL https://doi.org/10.1146/annurev-conmatphys-030212-184248.

[8] L. Trifunovic, O. Dial, M. Trif, J. R. Wootton, R. Abebe, A. Yacoby, and D. Loss, Phys. Rev. X **2**, 011006 (2012), URL https://link.aps.org/doi/10.1103/PhysRevX.2.011006.

[9] M. Trif, V. N. Golovach, and D. Loss, Phys. Rev. B **75**, 085307 (2007), URL https://link.aps.org/doi/10.1103/PhysRevB.75.085307.

[10] D. A. Lidar, T. A. Brun, and T. Brun, eds., *Quantum Error Correction* (Cambridge University Press, Cambridge,

2013), ISBN 978-1-139-03480-7, URL `http://ebooks.cambridge.org/ref/id/CBO9781139034807`.

[11] A. G. Fowler, Quantum Info. Comput. **15**, 145 (2015), ISSN 1533-7146, URL `http://dl.acm.org/citation.cfm?id=2685188.2685197`.

[12] J. R. Wootton, A. Peter, J. R. Winkler, and D. Loss, Phys. Rev. A **96**, 032338 (2017), URL `https://link.aps.org/doi/10.1103/PhysRevA.96.032338`.

[13] A. S. Darmawan and D. Poulin, Physical Review Letters **119** (2017), ISSN 0031-9007, 1079-7114, URL `http://link.aps.org/doi/10.1103/PhysRevLett.119.040502`.

[14] T. F. Watson, S. G. J. Philips, E. Kawakami, D. R. Ward, P. Scarlino, M. Veldhorst, D. E. Savage, M. G. Lagally, M. Friesen, S. N. Coppersmith, et al., Nature **555**, 633 (2018), ISSN 0028-0836, 1476-4687, URL `http://www.nature.com/doifinder/10.1038/nature25766`.