



Fault-tolerant connection of error-corrected qubits with noisy links



Joshua Ramette¹ ✉, Josiah Sinclair¹, Nikolas P. Breuckmann² & Vlado Vuletić¹

One of the most promising routes toward scalable quantum computing is a modular approach. We show that distinct surface code patches can be connected in a fault-tolerant manner even in the presence of substantial noise along their connecting interface. We quantify analytically and numerically the combined effect of errors across the interface and bulk. We show that the system can tolerate 14 times higher noise at the interface compared to the bulk, with only a small effect on the code's threshold and subthreshold behavior, reaching threshold with ~1% bulk errors and ~10% interface errors. This implies that fault-tolerant scaling of error-corrected modular devices is within reach using existing technology.

Quantum devices made from noisy components require error correction^{1,2} to scale. Building error-corrected devices involves connecting a large number of qubits with gates of sufficiently high fidelity³. However, due to the general difficulty of controlling ever larger numbers of qubits within a single physical unit, quantum hardware platforms encounter practical system size limits. For example, limits in the range of 10^2 – 10^4 physical qubits are expected for trapped ions (due to spectral crowding of motional modes⁴), superconducting qubits (due to cryostat size and chip fabrication⁵), and Rydberg arrays (due to finite laser power and microscope field of view^{6,7}). Because of these size limits, individual quantum processors may soon support multiple logical qubits^{8–13}, but still not be truly scalable error-corrected devices.

To scale beyond these limits, one can consider architectures of local modules linked together via a physically distinct mechanism which is generally noisier and slower, e.g., trapped-ion chains connected via entangled photons¹⁴. For a quantum computer, such a modular approach reduces the task of achieving true scalability to designing a unit module of fixed qubit number equipped with a fault-tolerant quantum input/output interface, so that scaling simply involves connecting more identical modules. For quantum communication, a similar modular approach is already necessary to move quantum information between processors separated by long distances².

A major challenge for error-corrected modular architectures is transferring quantum information between modules with sufficient speed and fidelity to satisfy the requirements for fault tolerance. Because inter-module communication channels are typically lossy, quantum communication is accomplished by post-selected entanglement distribution. Entanglement shared between two modules in the form of non-local Bell pairs then serves as a resource to enable teleported gates for inter-module operations^{15–17}.

As even the heralded Bell pairs may still be noisy^{17,18}, entanglement distillation has been proposed to convert many low-fidelity pairs into a

smaller number of higher-fidelity pairs¹⁹. However, simple distillation protocols can only reach errors approximately ten times larger than the local gate errors, because they involve many local gates^{20,21}. More sophisticated protocols are necessary to further reduce the noise, but the added complexity lowers the success rate, reducing the achievable code cycle rate and increasing the memory errors per code cycle^{18,22–26}.

In this article, we show that logical qubits encoded in surface code patches in distinct modules can be fault-tolerantly connected despite substantially elevated noise along their shared interface (see Fig. 1a). While previous work pointed out that fault-tolerant quantum communication between code patches with noisy links is possible in the limit where the local noise in the bulk is asymptotically below threshold^{27,28}, we instead study the code in the presence of noise throughout both the bulk and interface, as in Fig. 3. Developing an understanding of how bulk and interface noise contributions combine to form logical failure modes, we provide analytical bounds and numerical simulations to show that the threshold for interface noise is as high as ~10%, even with bulk noise close to the usual surface code threshold of ~1%. This relaxed threshold for communication errors implies that hardware platforms working toward building surface code patches (superconducting^{29,30}, neutral atoms^{13,31,32}, trapped ions^{18,33–35}, solid-state defects^{36–38}, and silicon photonics^{39,40}), many already close to realizing local logical qubits^{8–13}, only require noisy interconnects to immediately scale without distillation, better local gates, or other time or space overheads.

Results

Modeling the logical qubit interface

The surface code^{41–44} is a Calderbank-Shor-Steane (CSS) code with a high circuit-level decoding threshold of $p_{\text{bulk}}^* \approx 1\%$. In Fig. 1 for an $L \times L$ surface code, the 4-body parity check operators $Z^{\otimes 4}$ and $X^{\otimes 4}$, each realized by local CNOT gates between a syndrome qubit and four nearby data qubits, are

¹Department of Physics, MIT-Harvard Center for Ultracold Atoms and Research Laboratory of Electronics, Massachusetts Institute of Technology, Cambridge, MA 02139, USA. ²School of Mathematics, University of Bristol, Fry Building, Woodland Road, Bristol BS8 1UG, UK. ✉e-mail: joshuaramette@gmail.com

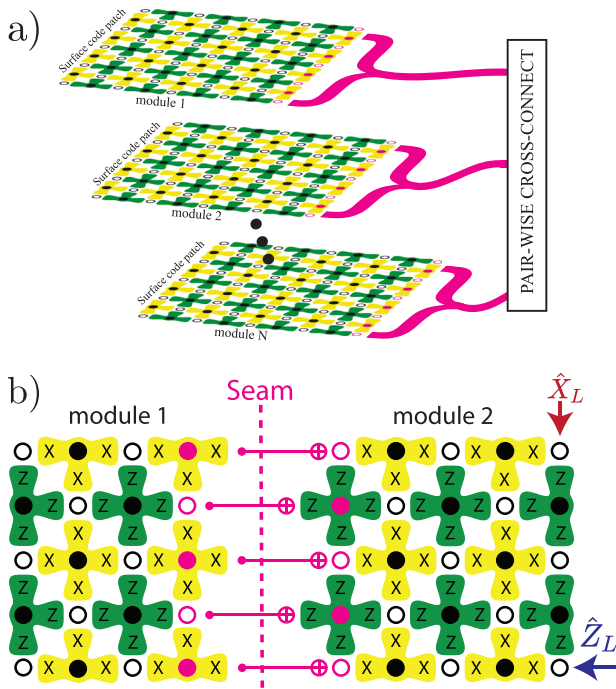


Fig. 1 | Surface code qubits in distinct modules connected with noisy quantum communication links (pink). **a** Communication between surface codes in distinct modules is routed through a cross-connect switch. **b** A logical gate between two surface code “bulk” patches in separate modules connected along a lower dimensional “seam.” Stabilizer checks span the seam with pink teleported gates. Pink data (open circle) and syndrome (filled circle) qubits lying along the interface of the two code patches experience elevated noise levels. \hat{X}_L, \hat{Z}_L indicate logical string operators.

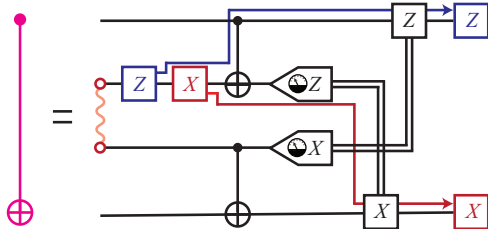


Fig. 2 | X, Z noise on a Bell pair (red squiggle) used in a teleported gate (pink) propagates to the two qubits it operates on. Phase flips only propagate to the control, and bit flips to the target of the teleported CNOT gate.

indicated by the four green or yellow leaves associated with each syndrome qubit. The logical Pauli operators are strings of X and Z Pauli operators along the vertical and horizontal directions of the surface code, respectively.

Various methods exist to perform fault-tolerant two-qubit gates between surface code patches in separate modules. These include lattice surgery⁴⁵, braiding³, or even directly moving logical qubits between modules, but all reduce to maintaining a surface code patch spanning the two modules. For logical computation across modules, it then suffices to connect the edges of two distinct surface code patches and perform parity checks spanning the modules which merge the two patches into a single larger surface code straddling the modules⁴⁵. As shown in Fig. 1b, along the interface or “seam” where the two code patches connect, one leaf from each check stretches between the code patches, indicating a (pink) CNOT gate between a syndrome qubit and a data qubit in separate modules.

Making contact with methods to realize the inter-module gates (highlighted in pink in Fig. 1b) via gate teleportation, Fig. 2 shows how Bell pair bit and phase flip noise propagates onto the target and control qubits of the teleported gate, respectively. Inspection of Figs. 1b and 2 shows that bit

flip noise from the noisy Bell pair propagates to the right of the pink seam, and phase flip noise to the left of the pink seam. As the X and Z errors trigger distinct check operators and because they can be decoded with distinct minimum weight perfect matching (MWPM) decoders⁴⁴, each decoder sees elevated noise on only a single strip of data+syndrome qubits to the right or left of the pink seam.

If the syndrome qubits were noiseless, the connecting interface would be equivalent to a 1D repetition code (bit and phase flip noise rates p_s) embedded in the 2D surface code (bit and phase flip noise rates p_b). To treat syndrome noise, we consider a generic phenomenological noise model where the bulk syndrome noise is set to be equal to the bulk data qubit noise $q_b = p_b$ ⁴⁴, and similarly for the seam syndrome and data qubit noise $q_s = p_s$, since Bell pair noise propagates to both. Noisy syndrome decoding is accomplished through L rounds of syndrome extraction, extending the matching graph along the syndrome qubits into an extra dimension representing time^{3,44}. This results in a $D_s = (1 + 1)$ dimension lattice of size $L \times L$ on which errors occur with rate p_s embedded in a $D_b = (2 + 1)$ dimension lattice of size $L \times L \times L$ on which errors occur with error rate p_b . Using this model of surface code patches, each locally error-corrected in a distinct module with a noisier interface connecting them, we next describe how to analytically model and numerically simulate the logical errors for various choices of L to extract threshold and subthreshold logical error behavior. From the extracted threshold behavior, we can determine the conditions under which two error-corrected logical qubits, housed and operating in distinct modules, can be interfaced using a noisy network. We find negligible degradation of fault tolerance even with interface noise $14 \times$ higher than the circuit-level noise present within each module.

Analytical bounds

We now show how the seam, due to its lower dimension than the bulk, can tolerate elevated levels of noise without severely compromising the integrity of the code spanning the modules, even in the presence of noise near the threshold p_b^* within the bulk of the code. Intuitively, the threshold of a surface code is determined by both the qubit noise level (the probability to extend a chain along a particular edge) and the number of directions available in which to extend the error chain. In a lower dimension, fewer directions are available in which to extend error chains, resulting in a higher noise threshold. Because $D_b > D_s$, the bulk and seam thresholds p_b^*, p_s^* then satisfy $p_s^* > p_b^*$.

Next, we consider how the extra seam noise in Fig. 1b affects the probabilities for \hat{X}_L and \hat{Z}_L logical errors. Since logical \hat{X}_L errors can occur both in the bulk and along the seam, suppressing them requires both seam noise below seam threshold, $p_s < p_s^*$, and bulk noise below bulk threshold, $p_b < p_b^*$. This is the most stringent case and considered in Fig. 4 and the remainder of the paper. In contrast, even for $p_s > p_s^*$, a \hat{Z}_L error still must penetrate the length of the bulk and can be suppressed so long as the bulk is below threshold. For equal data and syndrome qubit noise as we consider ($q_b = p_b$ and $q_s = p_s$), the probability of generating “time-like” error chains stretching through the L rounds of error correction for both the bit and phase flip decoders is identical to that for \hat{X}_L . Thus, Fig. 4 also shows the probability for time-like bit and phase flip error chains spanning L rounds. However, once the threshold criteria from Fig. 4 are reached, these are easily suppressed by extending the code further in the time direction with more rounds of error correction per logical gate.

Now, in order for a logical bit flip error to occur, the combined effect of a round of errors and corrections must generate some nontrivial chain of bit flips $\{\gamma\}$ of length at least L (see Fig. 3). For a single code patch with no seam, using the combinatoric path-counting approach from ref. 44, and also reviewed in refs. 46–48 and Supplementary Note 1, the logical failure probability P_{fail} is bounded by the number of such possible chains $n_{\text{nontrivial}}(L)$ times the probability for each to occur:

$$P_{\text{fail}}/\text{poly}(L) \leq n_{\text{nontrivial}}(L) \times (2^L p^{L/2}) \quad (1)$$

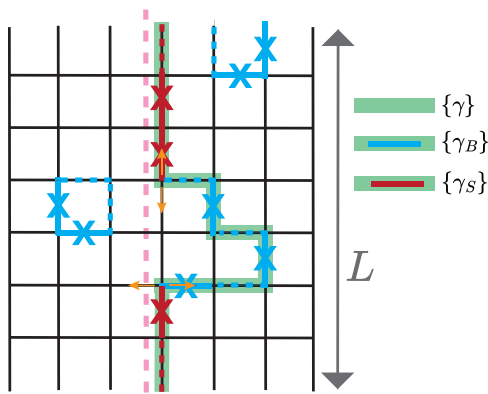


Fig. 3 | Seam (red) and bulk (blue) bit flip errors simultaneously contribute to logical failure. Edges are data qubits and vertices are check operators. Each X is a bit flip error due to the environment, and each dashed segment is a bit flip after decoding to return the logical qubit to the code space. In this example, if only the red seam bit flips occurred (and the bulk was error free $p_b = 0$), we could successfully correct the qubit. However, with $p_b > 0$ blue errors in the bulk can also occur and a logical error path then forms (green highlight) through this “excursion”, $\{\gamma_B\}$, into the bulk. Imagining the logical error path growing from the bottom to the top, the small orange arrows indicate the different directions available to choose when hopping off/on the seam to create an excursion.

Here, the probability bound of $2^L p^{L/2}$ originates from how MWPM fills in the rest of a nontrivial chain once $L/2$ bits flip due to environmental noise, and there being $\leq 2^L$ ways to choose half or more of the L bits to have flipped. To express a bound on $n_{\text{nontrivial}}(L)$, consider how when appending each additional edge to $\{\gamma\}$, one can move in any direction on the lattice other than back, so in $2D - 1$ directions in dimension D , bounding $n_{\text{nontrivial}}(L) \leq (2D - 1)^L \equiv \mu_D^L$. We can then rewrite:

$$P_{\text{fail}}/\text{poly}(L) \leq (4\mu_D^2 \times p)^{L/2} \equiv \left(\frac{p}{p^*}\right)^{L/2} \quad (2)$$

which is exponentially suppressed to zero as $L \rightarrow \infty$ for $p < p^*$. As μ_D is dimension dependent, so is the threshold bound $p^* \equiv 1/(4\mu_D^2)$.

In Supplementary Note 1, we extend the techniques from ref. 44 to derive bounds (Eqs. (3)–(6)) constraining how the threshold changes in the presence of a noisy seam by counting the number of error chains, which hop between the seam and bulk. Here, we present a heuristic argument reaching the same conclusions and elucidating the failure mechanisms. We can interpret our expression for the logical failure (Eq. (2)) as the factor of $\mu_D \times 2\sqrt{p}$ for each additional edge appended to $\{\gamma\}$ (so a probability of $(\mu_D \times 2\sqrt{p})^L$ for appending L edges), with μ_D ways to add edges, and $2\sqrt{p}$ as an effective “probability” per link (with the factor of 2 and the square root appearing because of how the MWPM procedure fills in missing links). If errors occur solely along the seam with $p_b = 0$, as in Fig. 3a, we would have a factor of $\mu_s \times 2\sqrt{p_s}$ per appended edge. But making $p_b > 0$, we suddenly allow additional paths through the bulk before appending the next seam link, as in Fig. 3, which adds terms corresponding to excursions into the bulk before reattaching to the seam. For a given excursion, in addition to flipping an arbitrary possible number of bulk edges ℓ , each of which can be appended in μ_b ways for a factor of $(\mu_b \times 2\sqrt{p_b})^\ell$, an excursion also involves flipping two bulk edges orthogonal to the direction along the seam as well as a final seam link when hopping back on to the seam for another factor of $\mu_c \times 2\sqrt{p_s}(2\sqrt{p_b})^2$. The coefficient $\mu_c \equiv 4D_s(D_b - D_s)$, where D_s is the dimension of the seam and D_b is the dimension of the bulk, counts the number of ways to choose both how to leave the seam ($2(D_s - D_b)$, as this is the number of edges which stick out from the seam into the bulk) and how to reattach to the seam ($2D_s$, as this is the number of directions one could choose within the seam itself when hopping back onto the seam to finish the excursion). See small orange arrows in Fig. 3 to visualize counting the ways to hop off/on the seam when creating an excursion. See Supplementary Note

1 for more details. By analogy with Eq. (2), we add up all of these ways to attach new seam links, giving us a modified factor containing a geometric series for each seam edge we append:

$$\begin{aligned} \mu_s \times 2\sqrt{p_s} + \sum_{\ell=0}^{\infty} (\mu_c \times 2\sqrt{p_s}(2\sqrt{p_b})^2) (\mu_b \times 2\sqrt{p_b})^\ell \\ = \sqrt{\frac{p_s}{p_s^*}} \left(1 + \alpha_c p_b \frac{\sqrt{p_s^*}}{1 - \sqrt{p_b/p_b^*}} \right) \end{aligned} \quad (3)$$

where in the second line, we have summed the geometric series over ℓ and defined $\alpha_c \equiv 8\mu_c$.

Again by analogy, we would then expect the failure probability bound to be Eq. (3) raised to the power of γ_s (from Fig. 3, the number of edges in $\{\gamma\}$ on the seam). We similarly introduce γ_B , the number of edges in $\{\gamma\}$ in the bulk. Additionally, realizing that if $\gamma_s < L$, there must be at least $\gamma_B = L - \gamma_s$ bulk links in $\{\gamma\}$, as $\{\gamma\}$ must have at least L total links in order to fail, we arrive at:

$$\begin{aligned} P_{\text{fail}}/\text{poly}(L) \leq \left(\frac{p_s}{p_s^*}\right)^{\frac{\gamma_s}{2}} + \left(\frac{p_b}{p_b^*}\right)^{\frac{\gamma_B}{2}} \\ + \sum_{\substack{\gamma_s \geq 1 \\ \gamma_B \neq 0}}^L \left[\frac{p_s}{p_s^*} \left(1 + \alpha_c p_b \frac{\sqrt{p_s^*}}{1 - \sqrt{p_b/p_b^*}} \right)^2 \right]^{\frac{\gamma_s}{2}} \left[\frac{p_b}{p_b^*} \right]^{\frac{L - \gamma_s}{2}} \end{aligned} \quad (4)$$

The first two terms correspond to failure chains within purely the seam or bulk, and the additional cross terms apply when considering chains with excursions (with nonzero excursion length $\gamma_B \neq 0$), in which case the threshold criteria mixes conditions on $\frac{p_s}{p_s^*}$ and $\frac{p_b}{p_b^*}$. All these terms are suppressed as $L \rightarrow \infty$ provided that the quantities in brackets are smaller than unity. Re-expressing Eq. (3), we can see that it is equivalent to a small downward “sag” of the seam threshold bound:

$$p_s^* \rightarrow p_{1s}^* \equiv p_s^* \left[1 + \alpha_c p_b \frac{\sqrt{p_s^*}}{1 - \sqrt{p_b/p_b^*}} \right]^2 \quad (5)$$

Fixing $\frac{p_s}{p_s^*} = \frac{p_b}{p_b^*}$, Eq. (4) reduces to:

$$P_{\text{fail}}/\text{poly}(L) \leq \left[\frac{p_b}{p_b^*} \left(1 + \alpha_c p_b \frac{\sqrt{p_s^*}}{1 - \sqrt{p_b/p_b^*}} \right)^2 \right]^{\frac{L}{2}} \equiv f(p_b, \alpha_c, L) \quad (6)$$

plotted in Fig. 4a.

Numerical simulations

In Fig. 4, we compare these analytical bounds to numerical Monte Carlo simulations^{49,50} to quantify the effect of error chains stretching simultaneously across the bulk and the seam. To model two 2D patches of surface code in distinct modules with a noisy interface between them and corrected with repeated rounds of noisy syndrome extraction extending into the time direction^{3,44}, we run the simulation with a $D_s = (1 + 1)$ dimension lattice of size $L \times L$ (the seam) on which errors occur with rate p_s embedded in a $D_b = (2 + 1)$ dimension lattice of size $L \times L \times L$ (the bulk) with error rate p_b . An additional consequence of the seam being a sublattice within the bulk is that per code cycle, while bulk qubits are addressed by four local gates, each seam qubit only interacts with a single Bell pair for communication. Letting p_{bulk} represent the probability of a local gate in bulk to cause a bit flip on a bulk qubit, we model this by directly substituting $4p_{\text{bulk}} = p_b$ (leading to a reasonable $p_{\text{bulk}}^* = 0.75\%$) while maintaining $p_{\text{seam}} = p_s$ since seam qubit noise is dominated by the Bell pair. From Fig. 4, we see that the numerical results display the same qualitative behavior as the analytical formulas, with a slight sag in the threshold as well as subthreshold convergence of the logical failure to the same value as without excursions. By adjusting the parameters μ_s and μ_b to match the known surface code threshold values for $D_s = 2$ and

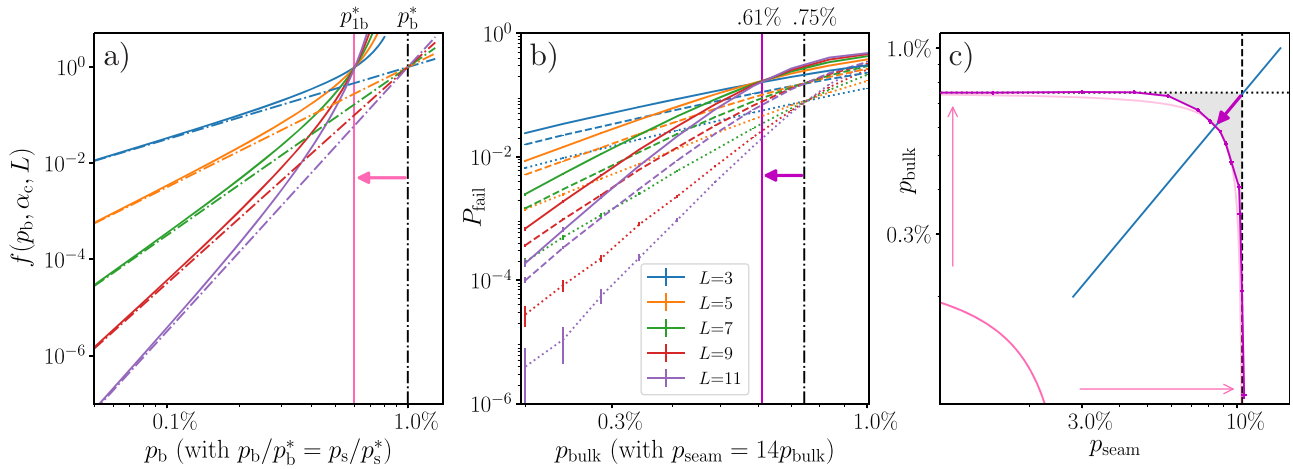


Fig. 4 | Analysis of the case $D_b = 3$ and $D_s = 2$. Error bars represent 95% CI. **a** Analytical bounds (Eq. (6)) for $p_b/p_b^* = p_s/p_s^*$ (solid), in which case the seam and bulk curves (dot-dashed) now overlap when plotted against p_b . Note that p_b^* here corresponds to the phenomenological threshold bound of 1% for 3D as derived in ref. 44 and p_{1b}^* shows the sag in the threshold bound we derive relative to the bound from ref. 44. Seam-bulk interactions reduce the threshold bound slightly to p_{1b}^* as indicated by the pink arrow. The logical failure rate converges to the values for no seam-bulk interactions once the bulk error is a few times below p_b^* , as excursions into the bulk become “frozen out.” **b** Same as (a) but exact numerical simulation with the choice $p_{seam} = 14p_{bulk}$, approximately $p_{bulk}/p_{bulk}^* = p_{seam}/p_{seam}^*$, which aligns the

thresholds for the bulk-only curves (dotted) and seam-only curves (dashed). The full simulation including seam-bulk interaction (solid) similarly sags slightly and then converges toward the seam-only curves as p_{bulk}/p_{bulk}^* becomes small. **c** Numerically extracted threshold plotted in terms of p_{seam}, p_{bulk} (purple), where $p_b = 4p_{bulk}$. Pink shows the threshold bound (Eq. (5)), with arrows pointing to the light pink line showing how the bound formula relaxes when reducing the overcounting of paths by substituting numerical values for p_{seam}^* and p_{bulk}^* along with an effective value of $\alpha_c \rightarrow 1.4$, the minimal value still bounding all the numerical datapoints. The blue diagonal line illustrates the $p_{seam} = 14p_{bulk}$ cut from **b**).

$D_b = 3^{44}$ and relaxing α_c from the rigorous bound to a smaller effective value, the bound formulas also provide a fairly accurate analytical model of the threshold behavior. Notably, up to the small correction represented by the gray region of Fig. 4c, P_{fail} behaves approximately as if bulk and seam were decoupled without cross terms from Eq. (4):

$$P_{fail}(p_{bulk}, p_{seam}) \approx \left(\frac{p_{bulk}}{p_{bulk}^*}\right)^{L/2} + \left(\frac{p_{seam}}{p_{seam}^*}\right)^{L/2} \quad (7)$$

with $p_{bulk}^* \approx 1\%$ and $p_{seam}^* \approx 10\%$. As long as $\frac{p_{seam}}{p_{seam}^*} \approx \frac{p_{bulk}}{p_{bulk}^*}$, so $p_{seam} \approx 10p_{bulk}$, the seam noise has almost no effect.

Discussion

Our formalism can further generalize to connecting surface code patches in a variety of configurations, such as transversal gates or a grid of smaller patches. While in Fig. 1b we aligned the physical edges of two code patches so that the seam extended in both space and time, the results from Fig. 4 can apply to any 2D subspace of a 3D lattice. Consequently, a transversal gate between logical qubits in separate modules mediated by Bell pairs and followed by rounds of local error correction on each logical qubit would be similarly robust to Bell pair noise, as the transversal gate occurs on a single code cycle and introduces noise on a 2D sublattice which now extends in two space directions rather than one space and one time direction as before. By adding multiple seams along space and/or time directions and counting the paths to hop between different seams, our formalism can be extended (see Supplementary Note 1 for details) and used to understand situations including repeated transversal gates⁵¹ or code patches spanning multiple modules, where one could tile large surface code patches together into a single larger logical qubit. This is because from Eq. (3), one can see that the contribution of an excursion is exponentially suppressed in the excursion length ℓ . When modules are large, excursions that hop between distinct seams are long, and thus have negligible contribution. In this case, each seam becomes approximately independent, so that the seam-bulk interaction of each seam alone determines the overall logical failure rate. In Supplementary Note 2, we give a quantitative example of this behavior. Furthermore, since the robustness to noisy links is a consequence of the interface between qubits being lower dimensional, our findings also apply to other families of

topological codes sharing this property, such as color codes⁵², which also have protocols for lattice surgery^{53,54}.

Our main result, that the interface can tolerate a noise level $p_{seam} \approx 10p_{bulk}$ (Eq. (7)), relaxes the communication fidelity required for fault-tolerance in ongoing experimental efforts to build modular architectures. Such efforts include optical interconnects between ions¹⁷, atoms⁵⁵, or superconducting qubits^{56–60}, direct superconducting microwave links^{51–64}, or even shuttling ions⁶⁵ or atoms¹³ between distinct subunits, some having already demonstrated communication errors below $p_{seam}^* \approx 10\%$ ^{17,62,64}. With multiple quantum computing platforms^{13,18,29–40} working toward realizing local surface code qubits and rapidly progressing toward module sizes of thousands of qubits and local gate noise targets of $\sim 0.1\%$ ^{8–13}, interconnects with a corresponding target of $\sim 1\%$ noise will directly enable fault-tolerant modular scalability. For even noisier networks, distillation can still be used, with the simplest and most efficient protocols being sufficient to reach errors of $\sim 10p_{bulk}$ ²¹, enabling scalability.

Methods

Model details for the interface between surface code qubits

Here, we provide additional details regarding the interface between the two logical qubits and how it gives rise to the phenomenological error model we numerically simulate.

When the interface between modules is realized via distributed entanglement, that entanglement serves as a resource for enacting non-local, teleported gates between qubits in distinct modules. Fig. 2 shows how bit and phase flip noise on the distributed Bell pair propagates to the control and target qubits in the distinct modules that the (pink) teleported gate acts on. The propagation is identical for errors occurring on either of the Bell pair qubits, as must be the case since the Bell pair is invariant under the application of XX and ZZ .

Therefore, in Fig. 1b, one can see that bit flip noise on the Bell pair propagates to the target qubits, i.e., the column of qubits to the right of the seam, and phase flip noise to control qubits, i.e., the column of qubits on the left of the seam. As the bit and phase flip errors trigger distinct check operators and can be decoded with distinct MWPM decoders, we can consider correcting for bit and phase flip errors as two separate problems. The seam can be chosen along either the \hat{X}_L or the \hat{Z}_L direction, but as in

Fig. 1b, for concreteness we choose the seam along the \hat{X}_L direction, in which case a logical \hat{X}_L error is possible if too many errors occur along the seam, even with no bulk noise.

For the numerical simulation, we then construct an $L \times L \times L$ lattice representing errors on the 2D surface code with L repeated rounds of syndrome extraction extending into the time direction^{3,44}. To simulate the noisy syndrome bit flip logical error rate, \hat{X}_L , with elevated bit flip noise along the seam, we randomly place bit flips on the edges of the 3D lattice, with elevated probabilities for bit flips occurring along the 2D sublattice constituting the seam, which extends in the \hat{X}_L and time directions. We then decode the syndromes arising from each set of bit flips using the Pymatching MWPM decoder⁵⁰ and apply the corresponding correction, which returns the state of the logical qubit to the code space. The fraction of times that the correction includes a set of remaining bit flips which are topologically equivalent to applying an \hat{X}_L operator is the logical bit flip error rate, plotted in Fig. 4. By symmetry, time-like error strings of length L occur with the same rate as logical \hat{X}_L errors shown in Fig. 4, giving rise to a distinct source of logical error, for example when attempting logical gates with L rounds of syndrome extraction during lattice surgery.

Bit and phase flip logical errors

With the seam oriented as in Fig. 1b, the situation for phase flip noise is different, as the seam does not stretch along the \hat{Z}_L direction. Elevated phase flip noise on the Bell pairs then has virtually no effect on the rate of logical phase flips \hat{Z}_L , as generating a \hat{Z}_L logical error requires phase flips to penetrate the entire length of the bulk, regardless of the level of phase flip noise present along the seam. However, the bulk and seam phase flip noise must still be below the same threshold results from Fig. 4 to control phase flip error strings from stretching long distances in the time direction. Once below this threshold condition, logical failure from long time-like error chains can be suppressed by simply doing more rounds of syndrome extraction per logical gate, which extends the code distance in the time direction with no additional overhead in qubit number. This asymmetry, resulting from the choice of the seam's orientation, allows one to immediately take advantage of any bias in the seam noise by choosing the seam to be along the \hat{X}_L or \hat{Z}_L direction with the lower level of seam noise.

Data availability

All data and code needed to evaluate the conclusions are available from the corresponding author upon reasonable request.

Received: 25 September 2023; Accepted: 28 May 2024;

Published online: 10 June 2024

References

- Shor, P. W. Scheme for reducing decoherence in quantum computer memory. *Phys. Rev. A* **52**, R2493–R2496 (1995).
- Muralidharan, S. et al. Optimal architectures for long distance quantum communication. *Sci. Rep.* **6**, 20463 (2016).
- Fowler, A. G., Mariantoni, M., Martinis, J. M. & Cleland, A. N. Surface codes: towards practical large-scale quantum computation. *Phys. Rev. A* **86**, 032324 (2012).
- Cetina, M. et al. Quantum gates on individually-addressed atomic qubits subject to noisy transverse motion. preprint at: <https://arxiv.org/abs/2007.06768> (2020).
- Ang, J. et al. Architectures for multinode superconducting quantum computers. Preprint at: <https://arxiv.org/abs/2212.06167> (2022).
- Saffman, M. Quantum computing with atomic qubits and Rydberg interactions: progress and challenges. *J. Phys. B* **49**, 202001 (2016).
- Saffman, M. Quantum computing with neutral atoms. *Natl. Sci. Rev.* **6**, 24–25 (2019).
- IonQ. Ionq aria. <https://ionq.com/quantum-systems/aria> (2023).
- IBM. What's next in quantum is quantum-centric supercomputing. <https://research.ibm.com/quantum-computing> (2023).
- IBM. Ibm unveils 400 qubit-plus quantum processor and next-generation ibm quantum system two. <https://newsroom.ibm.com/2022-11-09-IBM-Unveils-400-Qubit-Plus-Quantum-Processor-and-Next-Generation-IBM-Quantum-System-Two> (2023).
- Google. A preview of bristlecone, google's new quantum processor. <https://ai.googleblog.com/2018/03/a-preview-of-bristlecones-googles-new.html> (2023).
- Google. Quantum computer datasheet. <https://quantumai.google/hardware/datasheet/weber.pdf> (2021).
- Bluvstein, D. et al. A quantum processor based on coherent transport of entangled atom arrays. *Nature* **604**, 451–456 (2022).
- Monroe, C. et al. Large-scale modular quantum-computer architecture with atomic memory and photonic interconnects. *Phys. Rev. A* **89**, 022317 (2014).
- Gottesman, D. & Chuang, I. L. Demonstrating the viability of universal quantum computation using teleportation and single-qubit operations. *Nature* **402**, 390–393 (1999).
- Moehring, D. L. et al. Entanglement of single-atom quantum bits at a distance. *Nature* **449**, 68–71 (2007).
- Stephenson, L. J. et al. High-rate, high-fidelity entanglement of qubits across an elementary quantum network. *Phys. Rev. Lett.* **124**, 110501 (2020).
- Nigmatullin, R., Ballance, C. J., de Beaudrap, N. & Benjamin, S. C. Minimally complex ion traps as modules for quantum communication and computing. *New J. Phys.* **18**, 103028 (2016).
- Dür, W. & Briegel, H.-J. Entanglement purification for quantum computation. *Phys. Rev. Lett.* **90**, 067901 (2003).
- Campbell, E. T. Distributed quantum-information processing with minimal local resources. *Phys. Rev. A* **76**, 040302 (2007).
- Krastanov, S., Albert, V. V. & Jiang, L. Optimized entanglement purification. *Quantum* **3**, 123 (2019).
- Jiang, L., Taylor, J. M., Sørensen, A. S. & Lukin, M. D. Distributed quantum computation based on small quantum registers. *Phys. Rev. A* **76**, 062323 (2007).
- Li, Y. & Benjamin, S. C. High threshold distributed quantum computing with three-qubit nodes. *New J. Phys.* **14**, 093008 (2012).
- Fujii, K., Yamamoto, T., Koashi, M. & Imoto, N. A distributed architecture for scalable quantum computation with realistically noisy devices. *arXiv* <https://arxiv.org/abs/1202.6588> (2012).
- Nickerson, N. H., Li, Y. & Benjamin, S. C. Topological quantum computing with a very noisy network and local error rates approaching one percent. *Nat. Commun.* **4**, 1756 (2013).
- Nickerson, N. H., Fitzsimons, J. F. & Benjamin, S. C. Freely scalable quantum technologies using cells of 5-to-50 qubits with very lossy and noisy photonic links. *Phys. Rev. X* **4**, 041041 (2014).
- Fowler, A. G. et al. Surface code quantum communication. *Phys. Rev. Lett.* **104**, 180503 (2010).
- Li, Y. & Benjamin, S. C. Hierarchical surface code for network quantum computing with modules of arbitrary size. *Phys. Rev. A* **94**, 042303 (2016).
- Krinner, S. et al. Realizing repeated quantum error correction in a distance-three surface code. *Nature* **605**, 669–674 (2022).
- Al, G. Q. Suppressing quantum errors by scaling a surface code logical qubit. *Nature* **614**, 676–681 (2023).
- Wu, Y., Kolkowitz, S., Puri, S. & Thompson, J. D. Erasure conversion for fault-tolerant quantum computing in alkaline earth rydberg atom arrays. *Nat. Commun.* **13**, 4657 (2022).
- Auger, J. M., Bergamini, S. & Browne, D. E. Blueprint for fault-tolerant quantum computation with rydberg atoms. *Phys. Rev. A* **96**, 052320 (2017).
- Trout, C. J. et al. Simulating the performance of a distance-3 surface code in a linear ion trap. *New J. Phys.* **20**, 043038 (2018).
- IBM. Kim, J. & Monroe, C. Co-designing a scalable quantum computer with trapped atomic ions. *npj Quantum Inf.* **2**, 16034 (2016).

35. Lekitsch, B. et al. Blueprint for a microwave trapped ion quantum computer. *Sci. Adv.* **3**, e1601540 (2017).
36. Taminiau, T. H., Cramer, J., van der Sar, T., Dobrovitski, V. V. & Hanson, R. Universal control and error correction in multi-qubit spin registers in diamond. *Nat. Nanotechnol.* **9**, 171–176 (2014).
37. Xue, X. et al. Quantum logic with spin qubits crossing the surface code threshold. *Nature* **601**, 343–347 (2022).
38. Abobeih, M. H. et al. Fault-tolerant operation of a logical qubit in a diamond quantum processor. *Nature* **606**, 884–889 (2022).
39. Rudolph, T. Why I am optimistic about the silicon-photonics route to quantum computing. *APL Photonics* **2**, 030901 (2017).
40. Gimeno-Segovia, M., Shadbolt, P., Browne, D. E. & Rudolph, T. From three-photon greenberger-horne-zeilinger states to ballistic universal quantum computation. *Phys. Rev. Lett.* **115**, 020502 (2015).
41. Kitaev, A. Fault-tolerant quantum computation by anyons. *Ann. Phys.* **303**, 2–30 (2003).
42. Bravyi, S. B. & Kitaev, A. Y. Quantum codes on a lattice with boundary. preprint at: <https://doi.org/10.48550/arXiv.quant-ph/9811052> (1998).
43. Freedman, M. H. & Meyer, D. A. Projective plane and planar quantum codes. *Found. Comput. Math.* **1**, 325–332 (2001).
44. Dennis, E., Kitaev, A., Landahl, A. & Preskill, J. Topological quantum memory. *J. Math. Phys.* **43**, 4452–4505 (2002).
45. Horsman, D., Fowler, A. G., Devitt, S. & Meter, R. V. Surface code quantum computing by lattice surgery. *New J. Phys.* **14**, 123011 (2012).
46. Breuckmann, N. P. Phd thesis: homological quantum codes beyond the toric code. <https://arxiv.org/abs/1802.01520> (2018).
47. Dumer, I., Kovalev, A. A. & Pryadko, L. P. Thresholds for correcting errors, erasures, and faulty syndrome measurements in degenerate quantum codes. *Phys. Rev. Lett.* **115**, 050502 (2015).
48. Strikis, A., Benjamin, S. C. & Brown, B. J. Quantum computing is scalable on a planar array of qubits with fabrication defects. *Phys. Rev. Appl.* **19**, 064081 (2023).
49. Tuckett, D. K. Tailoring surface codes: improvements in quantum error correction with biased noise. Ph.D. thesis, University of Sydney (qecsim: <https://github.com/qecsim/qecsim>) (2020).
50. Higgott, O. Pymatching: a python package for decoding quantum codes with minimum-weight perfect matching. *ACM Transactions on Quantum Computing*, <https://dl.acm.org/doi/10.1145/3505637>(2022).
51. Bluvstein, D. et al. Logical quantum processor based on reconfigurable atom arrays. *Nature* **626**, 58–65 (2024).
52. Brown, B. J., Nickerson, N. H. & Browne, D. E. Fault-tolerant error correction with the gauge color code. *Nat. Commun.* **7**, 12302 (2016).
53. Landahl, A. J. & Ryan-Anderson, C. Quantum computing by color-code lattice surgery. preprint at: <https://arxiv.org/abs/1407.5103> (2014).
54. Thomsen, F., Kesselring, M. S., Bartlett, S. D. & Brown, B. J. Low-overhead quantum computing with the color code. preprint at: <https://arxiv.org/abs/2201.07806v2> (2022).
55. Reiserer, A. & Remppe, G. Cavity-based quantum networks with single atoms and optical photons. *Rev. Mod. Phys.* **87**, 1379–1418 (2015).
56. Sahu, R. et al. Quantum-enabled operation of a microwave-optical interface. *Nat. Commun.* **13**, 1276 (2022).
57. Tu, H.-T. et al. High-efficiency coherent microwave-to-optics conversion via off-resonant scattering. *Nat. Photonics* **16**, 291–296 (2022).
58. Delaney, R. D. et al. Superconducting-qubit readout via low-backaction electro-optic transduction. *Nature* **606**, 489–493 (2022).
59. Zhu, N. et al. Waveguide cavity optomagnonics for microwave-to-optics conversion. *Optica* **7**, 1291–1297 (2020).
60. Imany, P. et al. Quantum phase modulation with acoustic cavities and quantum dots. *Optica* **9**, 501–504 (2022).
61. Magnard, P. et al. Microwave quantum link between superconducting circuits housed in spatially separated cryogenic systems. *Phys. Rev. Lett.* **125**, 260502 (2020).
62. Burkhart, L. D. et al. Error-detected state transfer and entanglement in a superconducting quantum network. *PRX Quantum* **2**, 030321 (2021).
63. Yan, H. et al. Entanglement purification and protection in a superconducting quantum network. *Phys. Rev. Lett.* **128**, 080504 (2022).
64. Zhong, Y. et al. Deterministic multi-qubit entanglement in a quantum network. *Nature* **590**, 571–575 (2021).
65. Pino, J. M. et al. Demonstration of the trapped-ion quantum ccd computer architecture. *Nature* **592**, 209–213 (2021).

Acknowledgements

This project was funded in part by DARPA under the ONISQ program (grant # 134371-5113608), the MIT-Harvard Center for Ultracold Atoms, an NSF Physics Frontier Center (NSF grant # PHY-1734011), and AWS. Support is also acknowledged from the U.S. Department of Energy, Office of Science, National Quantum Information Science Research Centers, Quantum Systems Accelerator (contract # 7571809), The authors thank Dolev Bluvstein for helpful discussions.

Author contributions

J.R. originally conceived the work, developed the theoretical formalism, and derived the bound results. J.R. and J.S. conceptually developed the work, performed and interpreted the numerical simulations, and drafted the manuscript. N.B. contributed to formalizing the bound proof and provided helpful discussions. V.V. supervised the project. All authors discussed the results and contributed to the writing of the manuscript.

Competing interests

The authors declare no competing interests: V.V. is a co-founder and shareholder of QuEra.

Additional information

Supplementary information The online version contains supplementary material available at <https://doi.org/10.1038/s41534-024-00855-4>.

Correspondence and requests for materials should be addressed to Joshua Ramette.

Reprints and permissions 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 licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence 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 licence, visit <http://creativecommons.org/licenses/by/4.0/>.

© The Author(s) 2024