

Sparse Quantum Codes from Quantum Circuits

Dave Bacon
University of Washington,
Seattle, WA, USA
Current affiliation: Google Inc.,
Mountain View, CA, USA
dabacon@gmail.com

Steven T. Flammia
University of Sydney, Sydney,
Australia
sflammia@physics.usyd.edu.au

Aram W. Harrow
MIT, Cambridge, MA, USA
aram@mit.edu

Jonathan Shi
Cornell, Ithaca, NY, USA
jshi@cs.cornell.edu

ABSTRACT

Sparse quantum codes are analogous to LDPC codes in that their check operators require examining only a constant number of qubits. In contrast to LDPC codes, good sparse quantum codes are not known, and even to encode a single qubit, the best known distance is $O(\sqrt{n \log(n)})$, due to Freedman, Meyer and Luo.

We construct a new family of sparse quantum subsystem codes with minimum distance $n^{1-\epsilon}$ for $\epsilon = O(1/\sqrt{\log n})$. A variant of these codes exists in D spatial dimensions and has $d = n^{1-\epsilon-1/D}$, nearly saturating a bound due to Bravyi and Terhal.

Our construction is based on a new general method for turning quantum circuits into sparse quantum subsystem codes. Using this prescription, we can map an arbitrary stabilizer code into a new subsystem code with the same distance and number of encoded qubits but where all the generators have constant weight, at the cost of adding some ancilla qubits. With an additional overhead of ancilla qubits, the new code can also be made spatially local.

1. INTRODUCTION

Sparse quantum error-correcting codes obey the simple constraint that only a constant number of qubits need to be measured at a time to extract syndrome bits. Considerable effort has been devoted to studying sparse quantum codes, most notably in the context of topological quantum error correction [43]. This effort is driven by the fact that the sparsity constraint is quite natural physically, and existing fault-tolerant thresholds [38] and overheads [24] are optimized when the underlying code is sparse. Despite this effort, finding families of *good* sparse quantum codes – i.e. codes with asymptotically constant rate and relative distance – remains an open problem, in stark contrast to the situation for classical codes (see e.g. [33]).

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.
STOC'15, June 14–17, 2015, Portland, Oregon, USA.

Copyright is held by the owner/author(s). Publication rights licensed to ACM.
ACM 978-1-4503-3536-2/15/06 ...\$15.00.
<http://dx.doi.org/10.1145/2746539.2746608>.

Quantum subsystem codes [37] form a broad generalization of standard stabilizer codes where a subset of logical qubits is sacrificed to allow for extra gauge degrees of freedom. The two principle advantages of subsystem codes are that the measurements needed to extract syndrome information are in general sparser and the errors only need to be corrected modulo gauge freedom, which often improves fault-tolerance thresholds [1] (though not always [24]).

In this paper, we consider a general recipe that constructs a sparse quantum subsystem code for every Clifford quantum circuit. The new code resembles the circuit in that the layout of the circuit is replaced with new qubits in place of the inputs and outputs of each of the circuit elements. The gauge generators are localized to the region around the erstwhile circuit elements, and thus the sparsity s of the new code is constant when the circuit is composed of few-qubit gates.

When the circuit is a special form of a fault-tolerant syndrome measurement circuit for a “base” quantum stabilizer code encoding k qubits with distance d , then the new sparse subsystem code *inherits* the k and d parameters of the base code. We construct general fault-tolerant circuits of the requisite special form, and from this we show how *every* stabilizer code can be mapped into a sparse subsystem code with the same k and d as the original base code. The number of physical qubits n required for the new code is roughly the circuit size, and this can be chosen to be proportional to the sum of the weights of the original stabilizer generators if we do not insist on spatial locality of the circuit elements. Formally we have:

THEOREM 1. *Given any $[n_0, k_0, d_0]$ quantum stabilizer code with stabilizer generators of weight $w_1, \dots, w_{n_0-k_0}$, there is an associated $[n, k, d]$ quantum subsystem code whose gauge generators have weight $O(1)$ and where $k = k_0$, $d = d_0$, and $n = O(n_0 + \sum_i w_i)$. This mapping is constructive given the stabilizer generators of the base code.*

The proof is in Section 5 of the appended long version. It involves applying our circuit-to-code construction, as well as a new fault-tolerant measurement gadget that uses expander graphs. While expander graphs have played an important role in classical error correction, to our knowledge this is their first use in quantum error correction.

We then demonstrate the power of Theorem 1 by applying it to two natural scenarios: first to concatenated codes and

then to spatially local codes. By applying our construction to concatenated stabilizer codes, we obtain families of sparse subsystem codes with by far the best distance to date. The previous best distance for a sparse quantum code was due to Freedman, Meyer, and Luo [23], who constructed a family of stabilizer codes encoding a single logical qubit having minimum distance $d = O(\sqrt{n \log n})$. Our construction provides for the following improvement in parameters.

THEOREM 2. *Quantum error correcting subsystem codes exist with gauge generators of weight $O(1)$ and minimum distance $d = n^{1-O(1/\sqrt{\log n})}$.*

It is natural to ask if our construction can also be made spatially local. By spatially local we mean that all of the qubits can be arranged on the vertices of a square lattice in D dimensions with each gauge generator having support in a region of size $O(1)$. Incorporating spatial locality is indeed also possible, though it will in general increase the size of the circuit we use, and hence the total number of qubits in the subsystem code.

THEOREM 3. *Spatially local subsystem codes exist in $D \geq 2$ dimensions with gauge generators of weight $O(1)$ and minimum distance $d = n^{1-O(1/\sqrt{\log n})-1/D}$.*

Although the spatial locality constraint comes at the cost of decreased performance in the rate and relative distance, this scaling of the distance is nearly optimal. Several upper bounds have been proven about the parameters of spatially local subsystem codes in D dimensions. For this case, Bravyi and Terhal [11] have shown that $d \leq O(n^{1-1/D})$. Our codes nearly saturate this bound and have the virtue that they are in general constructive. In particular, our codes in $D = 3$ dimensions already improve on the previous best results (by Ref. [23] again) for *arbitrary* sparse codes and achieve $d = n^{2/3-O(1/\sqrt{\log n})}$.

Furthermore, for the class of local commuting projector codes in D dimensions (a class that generalizes stabilizer codes, but does *not* contain general subsystem codes), Bravyi, Poulin, and Terhal [10] have shown the inequality

$$kd^{2/(D-1)} \leq O(n). \quad (1)$$

It is open whether a similar upper bound holds for subsystem codes, but a corollary of our main results is that there are spatially local subsystem codes for every $D \geq 2$ that achieve

$$k \geq \Omega(n^{1-1/D}) \quad \text{and} \quad d \geq \Omega(n^{(1-1/D)/2}), \quad (2)$$

implying that $kd^{2/(D-1)} \geq \Omega(n)$.

The remainder of the paper is organized as follows. In Section 2 we review the theory of subsystem codes and the prior art. We define the construction for our codes in Section 3 and review the relevant properties of the construction in Section 4. Those sections provide a proof of Theorem 1 conditional on the existence of certain fault-tolerant circuits for measuring stabilizer-code syndromes, which we subsequently show exist in Section 5, thus completing the proof. Sections 6 and 7 are devoted to the proofs of Theorems 2 and 3 respectively, and we conclude with a discussion of open problems in Section 8.

2. BACKGROUND AND RELATED WORK

2.1 Quantum Subsystem Codes

For a system of n qubits, we can consider the group \mathcal{P}^n of all n -fold tensor products of single-qubit real-valued Pauli operators $\{I, X, iY, Z\}$ and including the phases $\{\pm 1\}$. A *stabilizer code* (see e.g. [35]) is the joint $+1$ eigenspace of a group of commuting Pauli operators $\mathcal{S} = \langle S_1, \dots, S_l \rangle$, where the S_i label a generating set for the group. (To avoid trivial codes, we require that $-I \notin \mathcal{S}$.) If each of the l generators are independent, then the code space is 2^k -dimensional where $k = n - l$, and there exist k pairs of *logical operators* which generate a group $\mathcal{L} = \langle X_1, Z_1, \dots, X_k, Z_k \rangle$. In general, the logical group is isomorphic to $N(\mathcal{S})/\mathcal{S}$ where $N(\mathcal{S})$ is the normalizer of \mathcal{S} in \mathcal{P}^n , meaning the set of all Paulis that commute with \mathcal{S} as a set. The logical group is isomorphic to \mathcal{P}^k , meaning that for each logical operator in \mathcal{L} we have that $[L_i, L_j] = 0$ for all $i \neq j$, and $X_i Z_i = -Z_i X_i$ for all i . The fact that $\mathcal{L} \subseteq N(\mathcal{S})$ means that $[L_i, S_j] = 0$ for all $S_j \in \mathcal{S}$. The *weight* of a Pauli operator is the number of non-identity tensor factors, and the *distance* of a code is the weight of the minimum weight element among all possible non-trivial logical operators (i.e. those which are not pure stabilizers).

A *subsystem code* [37, 30] is a generalization of a stabilizer code where we ignore some of the logical qubits and treat them as “gauge” degrees of freedom. More precisely, in a subsystem code the stabilized subspace \mathcal{H}_S further decomposes into a tensor product $\mathcal{H}_S = \mathcal{H}_L \otimes \mathcal{H}_G$, where by convention we still require that \mathcal{H}_L is a 2^k -dimensional space, and the space \mathcal{H}_G contains the unused logical qubits called gauge qubits. The gauge qubits give rise to a *gauge group* \mathcal{G} generated by former logical operators G_i (which obey the Pauli algebra commutation relations for a set of qubits) together with the stabilizer operators. We note that $-I$ is always in the gauge group, assuming that there is at least one gauge qubit. The logical operators in a subsystem code are given by $\mathcal{L} = N(\mathcal{G})$ and still preserve the code space. The center of the gauge group $Z(\mathcal{G})$ is defined to be the subgroup of all elements in \mathcal{G} that commute with everything in \mathcal{G} . Since $Z(\mathcal{G})$ contains $-I$, it cannot be the set of stabilizers for any nontrivial subspace. Instead we define the stabilizer subgroup \mathcal{S} to be isomorphic to $Z(\mathcal{G})/\{\pm I\}$. Concretely, if $Z(\mathcal{G})$ has generators $(-I, S_1, \dots, S_l)$ then we define the stabilizer group to be $\langle \epsilon_1 S_1, \dots, \epsilon_l S_l \rangle$ for some arbitrary choice of $\epsilon_1, \dots, \epsilon_l \in \{\pm 1\}$.

A classic example of a subsystem code is the Bacon-Shor code [4] having physical qubits on the vertices of an $L \times L$ lattice (so $n = L^2$). The gauge group is generated by neighboring pairs of XX and ZZ operators across the horizontal and vertical links respectively. The logical quantum information is encoded by a string of X operators along a horizontal line and a string of Z operators along a vertical line, and the code distance is $L = \sqrt{n}$.

We differentiate between two types of logical operators in a subsystem code: *bare* logical operators are those that act trivially on the gauge qubits, while *dressed* logical operators may in general act nontrivially on both the logical and gauge qubits. In other words, the bare logical group is $N(\mathcal{G})/\mathcal{S}$ while the dressed logical group is $N(\mathcal{S})/\mathcal{S}$. The distance of a subsystem code is the minimum weight among all nontrivial dressed logical operators, i.e. $\min\{|g| : g \in N(\mathcal{S}) - \mathcal{S}\}$. We

say that a code is a $[n, k, d]$ code if it uses n physical qubits to encode k logical qubits and has distance d .

2.2 Sparse Quantum Codes and Related Work

The sparsity of a code is defined with respect to a given set of gauge generators. If each generator has weight at most s_g and each qubit partakes in at most s_q generators, then we define $s = \max\{s_g, s_q\}$ and say the code is s -sparse. We call a code family simply *sparse* if $s = O(1)$. The most important examples of sparse codes are topological stabilizer codes, also called homology codes because of their relation to homology theory. The archetype for this code family is Kitaev's toric code [28], which encodes $k = O(1)$ qubits and has minimum distance $d = O(\sqrt{n})$ (although it can correct a constant fraction of random errors). It is known that $2D$ homological codes obey $d \leq O(\sqrt{n})$ [22]. Many other important examples of such codes are known; see Ref. [43] for a survey.

The discovery of subsystem codes [37, 30] led to the study of sparse subsystem codes, first in the context of topological subsystem codes, of which there are now many examples [4, 5, 19, 13, 42, 39, 8, 6, 12]. However, these codes are all concerned with the case $k = O(1)$. Work on codes with large k initially focused on random codes, where it was shown that random stabilizers have $k, d \propto n$ [16, 15, 3], and more recently that short random circuits generate good codes [14]. There are also known constructive examples of good stabilizer codes such as those constructed by Ashikhmin, Litsyn, and Tsfasman [2] and others [17, 18, 34, 31]. All of these codes have stabilizer generators with weight $\propto n$, however.

A growing body of work has made simultaneous improvement on increasing k and d while keeping the code sparse. The best distance achievable with a sparse code is due to Freedman, Meyer and Luo [23], encoding a single qubit with distance $O(\sqrt{n \log n})$. A different construction called hypergraph product codes by Tillich and Zémor [45] achieves a distance of $O(\sqrt{n})$ but with constant rate. These codes, like the toric code, can still correct a constant fraction of random errors [29] but they abandon spatial locality in general.

Some notion of spatial locality can be recovered by working with more exotic geometries than a simple cubic lattice in Euclidean space. Zémor constructed a family of hyperbolic surface codes with constant rate and logarithmic distance [46]; see also [26]. Guth and Lubotzky [25] have improved this by constructing sparse codes with constant rate and $d = O(n^{3/10})$. These codes and those of Ref. [23] live most naturally on cellulations of Riemannian manifolds with non-Euclidean metrics and unfortunately cannot be embedded into a cubic lattice in $D \leq 3$ without high distortion.

The Bacon-Shor codes [4] mentioned in the previous section were generalized by Bravyi [7] to yield a family of sparse subsystem codes encoding $k \propto \sqrt{n}$ qubits while still respecting the geometric locality of the gauge generators in $D = 2$ dimensions and maintaining the distance $d = \sqrt{n}$. This is an example of how subsystem codes can outperform stabilizer codes under spatial locality constraints, since two-dimensional stabilizer codes were proven in [7] to satisfy $kd^2 \leq O(n)$ (which generalizes [10] in D dimensions to $kd^{2/(D-1)} \leq O(n)$). Bravyi [7] has also shown that all spatially local subsystem codes in $D = 2$ dimensions obey the bound $kd \leq O(n)$ for $D = 2$ and so this scaling is optimal for two dimensions.

A family of $O(\sqrt{n})$ -sparse codes called homological product codes, due to Bravyi and Hastings [9], leverage random codes with added structure to create good stabilizer codes with a nontrivial amount of sparsity, but no spatial locality of the generators.

By way of comparison, classical sparse codes exist that are able to achieve linear rate and distance, and can be encoded and decoded from a constant fraction of errors in linear time [41].

3. CONSTRUCTING THE CODES

Our codes are built from existing stabilizer codes, and indeed our construction can be thought of as a recipe for sparsifying stabilizer codes.

3.1 The Base Code and Error-Detecting Circuits

Our code begins with an initial code called C_0 which is a stabilizer code with stabilizer group S_0 . By a slight abuse of notation, we use C_0 to also refer to the actual code space. It uses n_0 qubits to encode k_0 logical qubits with distance d_0 . Assume that there exists an error-detecting circuit consisting of the following elements: (i) A total of n_a ancilla qubits initialized in the $|0\rangle$ state; (ii) A total of n_0 data qubits; (iii) A Clifford unitary U_{ED} applied to the data qubits and ancillas; (iv) Single-qubit postselections onto the $|0\rangle$ state. Denote the resulting operator V_{ED} . By ordering the qubits appropriately we have

$$V_{ED} = (I^{\otimes n_0} \otimes \langle 0|^{\otimes n_a}) U_{ED} (I^{\otimes n_0} \otimes |0\rangle^{\otimes n_a}). \quad (3)$$

This satisfies $V_{ED}^\dagger V_{ED} \leq I$ automatically.

DEFINITION 4. A circuit V_{ED} is a good error-detecting circuit for C_0 if $V_{ED}^\dagger V_{ED}$ is the projector onto C_0 .

This means that it always accepts states in C_0 and always rejects states orthogonal to C_0 , assuming no errors occur while running the circuit. In other words,

$$\begin{aligned} V_{ED}^\dagger V_{ED} &= \\ & (I^{\otimes n_0} \otimes \langle 0|^{\otimes n_a}) U_{ED}^\dagger (I^{\otimes n_0} \otimes |0\rangle\langle 0|^{\otimes n_a}) U_{ED} (I^{\otimes n_0} \otimes |0\rangle^{\otimes n_a}) \\ &= \frac{1}{|S_0|} \sum_{s \in S_0} s. \end{aligned} \quad (4)$$

We allow the initializations and postselections to occur at staggered times across the circuit, so that the circuit is *not* simply a rectangular block in general. Describing this in sufficient detail for our purposes necessitates introducing somewhat cumbersome notation. The i th qubit is input or initialized at time T_i^{in} and output or measured at time T_i^{out} . All initializations, postselections, and elementary gates take place at *half-integer time steps*. Thus, a single-qubit gate acting at time, say, $t = 2.5$, can be thought of as mapping the state from time $t = 2$ to $t = 3$. The total depth of the circuit is then $\max_i T_i^{\text{out}} - \min_i T_i^{\text{in}} + 1$.

We defer a discussion of fault-tolerance in our circuits until Sec. 5.

3.2 Localized codes

To construct our code, we place a physical qubit at each integer spacetime location in the circuit. Thus, each wire of the circuit now supports up to T physical qubits, and in general the i th wire will hold $T_i^{\text{out}} - T_i^{\text{in}} + 1$ physical qubits.

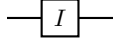
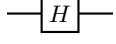
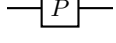
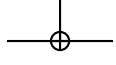
Circuit element	Gauge generators
	XX, ZZ
	ZX, XZ
	YX, ZZ
	XX, II, ZZ, ZI XI, XX, II, ZZ
$\langle 0 $ —	Z
— $ 0\rangle$	Z

Table 1: Dictionary for transforming circuit elements into generators of the gauge group. For every input and output of a circuit element in the left column, we add the corresponding generators from the right column, placed on the appropriate physical qubits. (This is the purpose of the η_t^i map in the main text.) We only list the gauge generators for the standard generators of the Clifford group, but the circuit identities of any Clifford circuit can be used instead. Pre- and postselections are special and have only one gauge generator associated to them.

Assume each $T_i^{\text{in}} \geq 0$ and let $T = \max_i T_i^{\text{out}}$. Then each qubit is active for some subset of times $\{0, \dots, T\}$. In some of our analysis it will be convenient to pretend that each qubit is present for the entire time $\{0, \dots, T\}$, and that all initializations and measurements happen at times 0 and T respectively. During the “dummy” time steps the qubits are acted upon with identity gates. It is straightforward to see that the code properties (except for total number of physical qubits) are identical with or without these dummy time steps. Thus, we will present our proofs as though dummy qubits are present, but will perform our resource accounting without them.

We introduce the function $\eta_t^i(P)$ to denote placing a Pauli P at spacetime position (i, t) . If P is a multi-qubit Pauli then we let $\eta^i(P)$ or $\eta_t(P)$ denote placing it either on row i coming from circuit qubit i or on column t corresponding to circuit time slice t . For a two-qubit gate U , we write $\eta_t^{i,j}(U)$ to mean that we place U at locations (i, t) and (j, t) . When describing a block of qubits without this spacetime structure, we also use the more traditional notation of P_i to denote Pauli P acting on position i : that is, $P_i := I^{\otimes i-1} \otimes P \otimes I^{\otimes n-i}$, where n is usually understood from context.

With this notation in hand, we define the gauge group of our codes. This is summarized in Table 1, and defined more precisely below. The gauge group will have $2k$ generators per k -qubit gate and one for each measurement or initialization. Let U be a single qubit gate that acts on qubit i as it transitions from time t to time $t+1$. Corresponding to this gate, we add the gauge generators

$$\eta_{t+1}^i(UXU^\dagger)\eta_t^i(X) \quad \text{and} \quad \eta_{t+1}^i(UZU^\dagger)\eta_t^i(Z). \quad (5a)$$

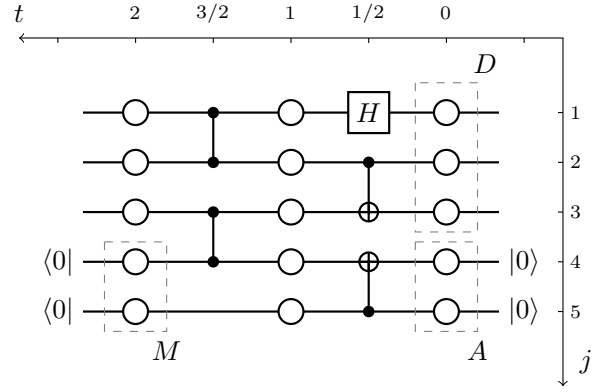


Figure 1: Illustration of the circuit-to-code mapping. Using integral spacetime coordinates (j, t) , the open circles at integer time steps (j, t) are physical qubits of the subsystem code, while gates of the circuit are “syncopated” and live at half-integer time steps (j, t) . The three sets of qubits in the dashed boxes labelled D , A , and M correspond to the input qubits for the base code, the ancillas, and the measurements (postselections) respectively. For this circuit, for example, we have among others the gauge generators $\eta_0^1(X)\eta_1^1(Z)$ and $\eta_0^1(Z)\eta_1^1(X)$ because these are the circuit identities for the Hadamard gate at spacetime location $(1, 1/2)$. Note also that we pad each line with identity gates to ensure that there are always an even number of gates on each line, which is important to maintain our code properties (see long version for details). We draw our circuit diagram with time moving from right to left to match the way that operators are composed; e.g. if we apply U_1 then U_2 then U_3 the resulting composite operator is $U_3U_2U_1$.

Similarly for a two-qubit gate U acting on qubits i, j at time $t+1/2$, we add the generators

$$\{\eta_{t+1}^{i,j}(UPU^\dagger)\eta_t^{i,j}(P) : P \in \{X \otimes I, Z \otimes I, I \otimes X, I \otimes Z\}_{i,j}\}. \quad (5b)$$

More generally, a k -qubit gate U acting on qubits i_1, \dots, i_k at time $t+1/2$ has generators

$$\{\eta_{t+1}^{i_1, \dots, i_k}(UPU^\dagger)\eta_t^{i_1, \dots, i_k}(P) : P = I^{\otimes j-1} \otimes Q \otimes I^{\otimes k-j}, j \in [k], Q \in \{X, Z\}\}. \quad (5c)$$

For measurements or initializations of qubit i we add generators $\eta_{T_i^{\text{out}}}^i(Z)$ or $\eta_{T_i^{\text{in}}}^i(Z)$ respectively.

An illustration of the mapping from the circuit to the code is given in Fig. 1.

4. CODE PROPERTIES

In this section we prove that our codes match—in the sense of Theorem 1—the performance of the base codes with respect to k and d . It constitutes the main technical part of our result. Specifically we prove (in the long version):

THEOREM 5. *If V is a fault-tolerant error-detection circuit (i.e. satisfying Definitions 4 and 7) for a code with k*

encoded qubits and distance d then the corresponding localized code has also has k encoded qubits and distance d .

This theorem relies on the following definitions.

DEFINITION 6. Given a collection of errors $E = (E_t)_t$, define the weight $|E_t|$ to be the number of nonidentity terms in E_t and define $|E| = \sum_t |E_t|$.

DEFINITION 7. A subcircuit V is fault-tolerant if for any error pattern E either $V_E = 0$ or there exists a Pauli operator E' on the input qubits such that $V_E = VE'$ and $|E'| \leq |E|$.

This is related to the conventional notion of fault-tolerance as both notions entail non-propagation of errors to equivalent errors of larger weight. Here, we demand that any non-detected error must be equivalent to an error on the input of no greater weight. Thus any circuit that handles up to d input errors will also be able to handle up to d total errors on the input and the circuit combined, if it is fault-tolerant.

A crucial feature of this definition is its composability (proved in the full version.)

LEMMA 8. If $V^{(1)}, \dots, V^{(T)}$ are fault-tolerant subcircuits that partition a circuit $U = V^{(T)} \dots V^{(1)}$, then U is fault tolerant as well.

The proof of Theorem 5 consists of carefully showing that our circuit-to-code mapping preserves the properties of several key structures. First, ‘‘circuit identities’’ (e.g. an X on qubit 4 at time 7 is equivalent to a Z on qubit 6 at time 8 together with a Y on qubit 5 at time 2) are shown to be equivalent to the gauge group. Next, the original stabilizer and logical group are mapped to the new stabilizer and logical group. Here the key challenge is showing that the wrong elements do not appear in the gauge group. Finally we show that the new code inherits the distance properties of the original code if we build it using a fault-tolerant error-detecting circuit. Proving distance is usually the nontrivial part of any code construction, and ours is no exception. The proof is based on constructing, for any error pattern, an equivalent set of errors acting only on the inputs of the circuit.

Although fault-tolerance is a difficult property to establish, our approach allows us to import results from the significant literature on fault-tolerant quantum computing and derive distance bounds from them. Nevertheless, in order to make our result self-contained, Section 5 describes a general construction of fault-tolerant measurement gadgets (based on the DiVincenzo-Shor syndrome measurement scheme [40, 20]) with the performance we need.

5. FAULT-TOLERANT GADGETS

The final piece of our construction is a fault-tolerant gadget for measuring a single stabilizer generator. With this we can construct fault-tolerant circuits for any stabilizer code, and therefore sparse subsystem codes from any stabilizer code, completing the proof of Theorem 1.

The requirements for fault-tolerance here are somewhat different from those in existing fault-tolerant measurement strategies. Our circuits are restricted to stabilizer circuits, and cannot make use of classical feedback or post-processing. On the other hand, the circuits here only need to detect errors rather than correct them. The gadgets we use are hence a variation on the DiVincenzo-Shor cat-state method [40,

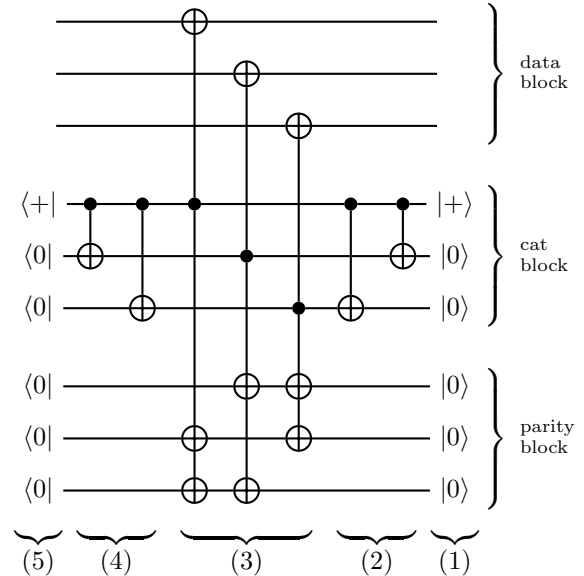


Figure 2: Example configuration for the fault-tolerant postselection gadget, for $w = 3$. The data block consists of input wires which are postselected to a $+1$ eigenstate of $X^{\otimes w}$. The cat block is prepared to contain a cat state, and the parity block is used for parity checks on that cat state. Time goes from right to left.

20], modified to detect instead of correct errors, and to do so with only Clifford gates (e.g. without majority voting).

The idea behind the gadgets is to prepare a w -qubit cat state $\frac{1}{\sqrt{2}}(|0\rangle^{\otimes w} + |1\rangle^{\otimes w})$ and perform a CNOT from each qubit in the cat state to each qubit that we want to measure. If we then postselect on the cat state remaining unchanged, this postselects the measured qubits onto the $+1$ eigenspace of $X^{\otimes w}$. The initial problem with this strategy is that it is not fault-tolerant: even though each measured qubit is involved in only one interaction, it is still possible for errors to propagate from the cat state into the measured qubits. To prevent this, we add additional CNOTs from the cat state into a block of check qubits, which are each initialized and post-selected onto the $|0\rangle$ state. We will see that by building this pattern of CNOTs from a sufficiently strong expander graph, the gadget can be made fault-tolerant.

A sketch of this argument is as follows. There are three blocks of qubits: w data qubits, w cat-state qubits and $w' = O(w)$ check qubits. Let (V, E) be a graph with $|V| = w$ and $|E| = w'$. We identify each data qubit and each cat-state qubit with vertices in V , and each check qubit with an edge in E . Our circuit will perform one parallel CNOT (i.e. several controlled NOTs all with the same control but different targets) for each $v \in V$. The control of each parallel CNOT will be the cat-state qubit corresponding to v , and the targets will be the corresponding data qubit and all the check qubits that correspond to edges incident on v . An example configuration using the complete graph on 3 vertices is given in Figure 2, with cat state initialization performed in steps 1-2, parallel CNOTs performed in step 3, and cat state postselection performed in steps 4-5.

If there are no errors then each check qubit will be the target of two CNOTs and their post-selections will all pass. If there are errors on the cat-state qubits that propagate to the data qubits, then they will also propagate to the check qubits. Now, if (V, E) is an expander with edge expansion ≥ 1 (i.e. any $S \subset V$ with $|S| \leq w/2$ has $\geq |S|$ edges from S to $V \setminus S$) then this process must result in more errors on the check qubits than the cat-state qubits. These in turn will either cancel a larger number of other errors there or cause a post-selection to fail. This argument rules out the only possible way for a small number of errors to be magnified by the circuit and affect a larger number of qubits; in other words, the only way for the fault-tolerant condition to fail. Finally we can satisfy the edge-expansion condition with a constant-degree explicit graph. Putting these ingredients together yields the desired fault-tolerant gadget. (Again, the full details are in the long version of the paper.)

6. SPARSE QUANTUM CODES WITH IMPROVED DISTANCE AND RATE

Our Theorem 1 implies that substantially better distance can be achieved with sparse subsystem codes than has previously been achieved. The following argument (suggested to us by Sergey Bravyi) is based on applying Theorem 1 to concatenated stabilizer codes with good distance.

To apply this argument, we must first have that codes with good distance exist. This is guaranteed by the quantum Gilbert-Varshamov bound, one version of which states that if $\sum_{j=0}^{d-1} \binom{n}{j} 3^j \leq 2^{n-k}$ then an $[n, k, d]$ quantum stabilizer code exists [16]. We need only the fact that there exist $[n_0, 1, d_0]$ codes with $d_0 = \Omega(n_0)$. In general the generators will have high weight, but of course this weight cannot be higher than n_0 .

Next we concatenate this code with itself m times. This starts with one qubit, encodes it into n_0 qubits, encodes each of *those* into n_0 qubits, and so on a total of m times, ending with n_0^m qubits. The resulting distance is easily seen to be d_0^m . The stabilizer generators come from each level of concatenation; at the lowest level there are n_0^m generators with weight $\leq n_0$; at the next, there are n_0^{m-1} generators with weight $\leq n_0^2$; and so on. The total weight is $\leq mn_0^{m+1}$. Choosing m optimally and applying Theorem 1 then yields Theorem 2. Details are in the long version.

7. MAKING SPARSE CODES LOCAL

We can use SWAP gates, identity gates, and some rearrangement to embed the circuits from Theorem 1 into D space-time dimensions so that all gates become spatially local. The codes constructed in this way are not just sparse, but also geometrically local. This results in nearly optimal distances of $\Omega(n^{1-1/D-o(1)})$ [11], as well as spatially local codes that achieve $kd^{2/(D-1)} \geq \Omega(n)$ in $D = 4$ dimensions.

The key ingredient here is to perform the fault-tolerant error-detection in a way that optimizes not only the number of gates but also the depth, while respecting spatial locality. This is achieved by making use of the recursive structure of concatenated codes. We still need to use a small number of permutation circuits, for which efficient constructions have been known for several decades [44]. The basic principle here is the same one used to embed a generic computation into D spatial dimensions. Again the details are in the long version.

8. DISCUSSION

The construction presented here leaves numerous open questions.

We have not addressed the important issue of efficient decoders for these codes. It seems likely that the subsystem code can be decoded efficiently if the base code can, but we have not yet checked this in detail and leave this to future work. One potential stumbling block is that the subsystem code requires measuring gauge generators which must be multiplied together to extract syndrome bits. Since the stabilizers for our subsystem codes are in general highly nonlocal and products of many gauge generators, this might lead to difficulties in achieving a fault-tolerant decoder in the realistic case of noisy measurements.

Another open question is whether the distance scaling of Theorems 2 and 3 can be extended to apply also to k to some degree. Improving the fault-tolerant gadgets or using specially designed base codes seem like obvious avenues to try to improve on our codes. Conversely, extending the existing upper bounds by Bravyi [7] to $D > 2$, as well as extending the bound from Bravyi, Poulin, and Terhal [10] to subsystem codes would be also be interesting. We conjecture that Eq. (1) extends to subsystem codes, and that the scaling in (2) is tight.

It would be interesting to see if the base codes for our construction could be extended to include subsystem codes. This would open up the possibility to bootstrap this construction into multiple layers of concatenation.

It is still an open question whether any distance greater than $O(\sqrt{n \log n})$ can be achieved for stabilizer codes with constant-weight generators. If an upper bound on the distance for such stabilizer codes were known, then it could imply an asymptotic separation between the best distance possible with stabilizer and subsystem codes with constant-weight generators, like the separation for spatially local codes in $D = 2$ dimensions [7].

Another open question is whether the recent methods of Gottesman [24] for using sparse codes in fault-tolerant quantum computing (FTQC) schemes can be modified to work with subsystem codes. If they could, then improving our scaling with k would imply that FTQC is possible against adversarial noise at rate $R = \exp(-c\sqrt{\log n})$.

We conjecture pessimistic answers to both questions, i.e., sparse stabilizer codes cannot achieve distance above $O(\sqrt{n \log n})$ and FTQC is impossible against rate- R adversarial noise. Nevertheless, subsystem codes have in the past proven useful for FTQC [1] and we are hopeful that our codes might assist in further developments of FTQC techniques.

Finally, we cannot resist the temptation to speculate on the ramifications of these codes for self-correcting quantum memories. The local versions of our codes in 3D have no string-like logical operators. To take advantage of this for self-correction, we need a local Hamiltonian that has the code space as the (at least quasi-) degenerate ground space and a favorable spectrum. The underlying code should also have a threshold against random errors [36]. The obvious choice of Hamiltonian is minus the sum of the gauge generators, but this will not be gapped in general. Indeed, the simplest example of a Clifford circuit – a wire of identity gates – maps directly onto the quantum XY model, which is gapless when the coupling strengths are equal [32], but somewhat encouragingly is otherwise gapped and maps onto Kitaev’s proposal for a quantum wire [27]. Other models of

subsystem code Hamiltonians exist; some are gapped [13, 8, 12] and some are not [4, 21]. Addressing the lack of a satisfying general theory of gauge Hamiltonians is perhaps a natural first step in trying to understand the power of our construction in the quest for a self-correcting memory.

Acknowledgments

We thank Sergey Bravyi for suggesting the argument in Theorem 2, David Poulin for discussions and Larry Guth for explanations about [25]. DB was supported by the NSF under Grants No. 0803478, 0829937, and 0916400 and by the DARPA-MTO QuEST program through a grant from AFOSR. STF was supported by the IARPA MQCO program, by the ARC via EQuS project number CE11001013, by the US Army Research Office grant numbers W911NF-14-1-0098 and W911NF-14-1-0103, and by an ARC Future Fellowship FT130101744. AWH was funded by NSF grant CCF-1111382 and ARO contract W911NF-12-1-0486. JS was supported by the Mary Gates Endowment, Cornell University Fellowship, and David Steurer’s NSF CAREER award 1350196.

9. REFERENCES

- [1] P. Aliferis and A. W. Cross. Subsystem fault tolerance with the Bacon-Shor code. *Phys. Rev. Lett.*, 98:220502, 2007, [arXiv:quant-ph/0610063](#).
- [2] A. Ashikhmin, S. Litsyn, and M. A. Tsfasman. Asymptotically good quantum codes. *Phys. Rev. A*, 63(3):032311, Feb 2001, [arXiv:quant-ph/0006061](#).
- [3] A. E. Ashikhmin, A. M. Barg, E. Knill, and S. N. Litsyn. Quantum error detection II: Bounds. *IEEE Trans. Info. Theory*, 46(3):789–800, 2000, [arXiv:quant-ph/9906131](#).
- [4] D. Bacon. Operator quantum error-correcting subsystems for self-correcting quantum memories. *Phys. Rev. A*, 73(1):012340, 2006, [arXiv:quant-ph/0506023](#).
- [5] H. Bombin. Topological subsystem codes. *Phys. Rev. A*, 81(3):032301, Mar 2010, [arXiv:0908.4246](#).
- [6] H. Bombin. Gauge Color Codes. 2013, [arXiv:1311.0879](#).
- [7] S. Bravyi. Subsystem codes with spatially local generators. *Phys. Rev. A*, 83:012320, Jan 2011, [arXiv:1008.1029](#).
- [8] S. Bravyi, G. Duclos-Cianci, D. Poulin, and M. Suchara. Subsystem surface codes with three-qubit check operators. *Quant. Inf. Comp.*, 13(11&12):0963–0985, 07 2013, [arXiv:1207.1443](#).
- [9] S. Bravyi and M. B. Hastings. Homological product codes, 2013, [arXiv:1311.0885](#).
- [10] S. Bravyi, D. Poulin, and B. Terhal. Tradeoffs for reliable quantum information storage in 2d systems. *Phys. Rev. Lett.*, 104:050503, Feb 2010, [arXiv:0909.5200](#).
- [11] S. Bravyi and B. Terhal. A no-go theorem for a two-dimensional self-correcting quantum memory based on stabilizer codes. *New J. Phys.*, 11(4):043029, 2009, [arXiv:0810.1983](#).
- [12] C. G. Brell, S. D. Bartlett, and A. C. Doherty. Perturbative 2-body Parent Hamiltonians for Projected Entangled Pair States. 2014, [arXiv:1407.4829](#).
- [13] C. G. Brell, S. T. Flammia, S. D. Bartlett, and A. C. Doherty. Toric codes and quantum doubles from two-body Hamiltonians. *New J. Phys.*, 13(5):053039, 2011, [arXiv:1011.1942](#).
- [14] W. Brown and O. Fawzi. Short random circuits define good quantum error correcting codes. *IEEE International Symposium on Inf. Theory*, pages 346–350, 2013, [arXiv:1312.7646](#).
- [15] A. R. Calderbank, E. M. Rains, P. W. Shor, and N. J. A. Sloane. Quantum error correction via codes over $GF(4)$. *IEEE Trans. Info. Theory*, 44:1369–1387, 1998, [arXiv:quant-ph/9608006](#).
- [16] A. R. Calderbank and P. W. Shor. Good quantum error-correcting codes exist. *Phys. Rev. A*, 54(2):1098–1105, Aug 1996, [arXiv:quant-ph/9512032](#).
- [17] H. Chen. Some good quantum error-correcting codes from algebraic-geometric codes. *IEEE Trans. Info. Theory*, 47(5):2059–2061, Jul 2001, [arXiv:quant-ph/0107102](#).
- [18] H. Chen, S. Ling, and C. Xing. Asymptotically good quantum codes exceeding the Ashikhmin-Litsyn-Tsfasman bound. *IEEE Trans. Info. Theory*, 47:2055, 2001.
- [19] G. M. Crosswhite and D. Bacon. Automated searching for quantum subsystem codes. *Phys. Rev. A*, 83:022307, Feb 2011, [arXiv:1009.2203](#).
- [20] D. P. DiVincenzo and P. W. Shor. Fault-tolerant error correction with efficient quantum codes. *Phys. Rev. Lett.*, 77:3260–3263, Oct 1996, [arXiv:quant-ph/9605031](#).
- [21] J. Dorier, F. Becca, and F. Mila. Quantum compass model on the square lattice. *Phys. Rev. B*, 72(2):024448, Jul 2005, [arXiv:cond-mat/0501708](#).
- [22] E. Fetaya. Bounding the distance of quantum surface codes. *J. Math. Phys.*, 53(6):062202, 2012.
- [23] M. Freedman, D. Meyer, and F. Luo. \mathbb{Z}_2 -systolic freedom and quantum codes. In R. K. Brylinski and G. Chen, editors, *Math. of Quantum Computation*, pages 287–320. Chapman & Hall/CRC, 2002.
- [24] D. Gottesman. Fault-tolerant quantum computation with constant overhead, 2013, [arXiv:1310.2984](#).
- [25] L. Guth and A. Lubotzky. Quantum error-correcting codes and 4-dimensional arithmetic hyperbolic manifolds, 2013, [arXiv:1310.5555](#).
- [26] I. H. Kim. Quantum codes on Hurwitz surfaces. Master’s thesis, Massachusetts Institute of Technology, 2007.
- [27] A. Y. Kitaev. Unpaired majorana fermions in quantum wires. *Phys.-Usp.*, 44(10S):131–136, oct 2001, [arXiv:cond-mat/0010440](#).
- [28] A. Y. Kitaev. Fault-tolerant quantum computation by anyons. *Ann. Phys.*, 303(1):2–30, 2003, [arXiv:quant-ph/9707021](#).
- [29] A. A. Kovalev and L. P. Pryadko. Fault tolerance of quantum low-density parity check codes with sublinear distance scaling. *Phys. Rev. A*, 87:020304, 2013, [arXiv:1208.2317](#).
- [30] D. W. Kribs, R. Laflamme, D. Poulin, and M. Lesosky. Operator quantum error correction. *Quant. Inf. Comp.*, 6:383–399, 2006, [arXiv:quant-ph/0504189](#).

- [31] Z. Li, L. Xing, and X. Wang. A family of asymptotically good quantum codes based on code concatenation. *IEEE Trans. Info. Theory*, 55(8):3821, Aug 2009, [arXiv:0901.0042](#).
- [32] E. Lieb, T. Schultz, and D. Mattis. Two soluble models of an antiferromagnetic chain. *Ann. Phys.*, 16(3):407–466, 1961.
- [33] D. J. C. MacKay. *Information Theory, Inference, and Learning Algorithms*. Cambridge University Press, 2003.
- [34] R. Matsumoto. Improvement of Ashikhmin-Litsyn-Tsfasman bound for quantum codes. *IEEE Trans. Info. Theory*, 48(7):2122–2124, Jul 2002.
- [35] M. A. Nielsen and I. L. Chuang. *Quantum Computation and Quantum Information*. Cambridge University Press, Cambridge, 2000.
- [36] F. Pastawski, A. Kay, N. Schuch, and I. Cirac. Limitations of passive protection of quantum information. *Quant. Info. Comp.*, 10(7 & 8):0580–0618, 2009, [arXiv:0911.3843](#).
- [37] D. Poulin. Stabilizer formalism for operator quantum error correction. *Phys. Rev. Lett.*, 95(23):230504, Dec 2005, [arXiv:quant-ph/0508131](#).
- [38] R. Raussendorf and J. Harrington. Fault-tolerant quantum computation with high threshold in two dimensions. *Phys. Rev. Lett.*, 98:190504, 2007, [arXiv:quant-ph/0610082](#).
- [39] P. Sarvepalli and K. R. Brown. Topological subsystem codes from graphs and hypergraphs. *Phys. Rev. A*, 86:042336, Oct 2012, [arXiv:1207.0479](#).
- [40] P. W. Shor. Fault-tolerant quantum computation. In *FOCS*, pages 56–65, Oct 1996, [arXiv:quant-ph/9605011](#).
- [41] D. A. Spielman. Linear-time encodable and decodable error-correcting codes. *IEEE Trans. Info. Theory*, 42(6):1723–1731, 1996.
- [42] M. Suchara, S. Bravyi, and B. Terhal. Constructions and noise threshold of topological subsystem codes. *J. Phys. A: Math. Theor.*, 44(15):155301, 2011, [arXiv:1012.0425](#).
- [43] B. M. Terhal. Quantum error correction for quantum memories. 02 2014, [arXiv:1302.3428](#).
- [44] C. D. Thompson and H. T. Kung. Sorting on a mesh-connected parallel computer. *Commun. ACM*, 20(4):263–271, 1977.
- [45] J.-P. Tillich and G. Zémor. Quantum LDPC codes with positive rate and minimum distance proportional to $n^{1/2}$. In *Proceedings of the 2009 IEEE international conference on Symposium on Information Theory - Volume 2*, ISIT'09, pages 799–803, 2009, [arXiv:0903.0566](#).
- [46] G. Zémor. On Cayley graphs, surface codes, and the limits of homological coding for quantum error correction. In Y. Chee, C. Li, S. Ling, H. Wang, and C. Xing, editors, *Coding and Cryptology*, volume 5557 of *Lecture Notes in Computer Science*, pages 259–273. Springer Berlin Heidelberg, 2009.