Finite-rate quantum sparse codes aplenty

Maxime Tremblay, Guillaume Duclos-Cianci, and Stefanos Kourtis

Institut quantique, Université de Sherbrooke, Sherbrooke, Québec, Canada, J1K 2R1

We present a novel construction of random stabilizer codes from random bipartite graphs using a constraints satisfiability solver. We provide numerical evidences for the existence of regimes where finding codes is much easier than in the general cases. We also observe that sparse codes obtained this way practically achieve the channel capacity for erasure noise.

1 Introduction

Quantum error-correcting codes are essential to achieve reliable quantum computing. While codes like surface codes [10, 17, 29] and color codes [24, 25] have great potential to encode a small constant number of qubits, their performances degrade rapidly when performing computations with more qubits [12, 15]. That being said, quantum error-correcting codes with asymptotically good performances have been known to exist for more than 20 years [7, 14]. However, these codes require the measurement of operators of weight that increases with block size. This is a major limitation to efficient implementations of measurement circuits for these codes.

More recently, Gottesman showed that finite-rate quantum low-density parity-check (LDPC) codes can be used to build constant-overhead fault-tolerant quantum computers [20]. LDPC codes are stabilizer codes with bounded-weight stabilizers. That is, they involve only bounded-weight measurements.

That impressive result together with the pletora of asymptotically optimal classical LDPC code constructions [18, 27] lead to a surge of research toward quantum LDPC codes inspired by their classical counterparts. This resulted in many great quantum LDPC code constructions such as hypergraph product codes [30], homological product codes [11] and fiber bundle codes [23]. Finally, more than 20 years after the introduction of the first quantum error-correcting codes, Panteleev and Kalachev [26] introduced the first family of quantum LDPC codes with optimal asymptotic performances.

However, such constructions are generally based on precise labelling of products and combinations of classical LDPC codes. Their product nature implies that they are more relevant for larger number of qubits and do not allow for fine-grained tuning of code parameters. For example, recent numerical studies of hypergraph product codes [21, 31] involve a few thousands to hundreds of thousands of qubits. Some more randomized constructions include stabilizer codes from random circuits [13, 22], but, to our knowledge, there exists no random procedure to generate the stabilizer operators directly without relying on classical codes.

In this work, we introduce a procedure to generate an arbitrary number of stabilizer operators directly for any given number of qubits. Our construction starts from a random bipartite graph with the first set of nodes corresponding to qubits and the second to stabilizers. We then search for a subset of edges and a coloring of the stabilizer nodes equivalent to a Tanner graph. We garantee that all stabilizers commute by viewing the search as a constraints satisfaction problem.

Finding codes using this procedure is generally hard. However, we find regimes characterized by the density of the initial graph where it is easy to find stabilizer codes. This is akin to random k-SAT and random 2-coloring of k-uniform hypergraphs, two NP-complete problems that display a critical transition between the easy and hard instances [5, 6]. Futhermore, we provide evidence

Stefanos Kourtis: stefanos.kourtis@usherbrooke.ca

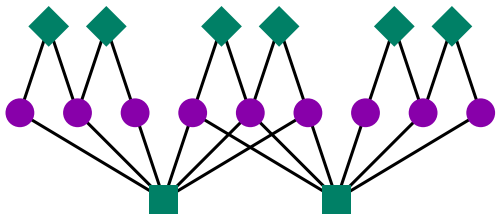

Figure 1: The Tanner graph of the 9 qubits Shor code. Purple circles represent qubit nodes and green squares and diamonds represent respectively X and Z stabilizer generators. The top right stabilizer node corresponds to the Z_7Z_8 operator and the bottom right node corresponds to the $X_0X_1X_2X_3X_4X_5$ operator.

that the resulting codes are much sparser than the initial graphs. We finally observe that codes obtained this way achieve the erasure channel capacity for all practical purposes.

The rest of the paper is divided as follows. In Section 2, we recall notions of quantum coding theory and stabilizer code construction. Then, in Section 3, we introduce our procedure to generate stabilizer codes from random graphs. Finally, we present numerical results in Section 4 before concluding.

2 Stabilizer codes

Given the n-qubit Pauli group $\mathcal{P}_n$, a stabilizer group is a commuting subgroup of $\mathcal{P}_n$ not including the $-I$ operator. A stabilizer group $\mathcal{S}$ defines the stabilizer code [19]

$$\mathcal{C}(\mathcal{S}) = \{|\psi\rangle \mid S|\psi\rangle = |\psi\rangle, \forall S \in \mathcal{S}\}. \quad (1)$$

That is, a stabilizer code is the common $+1$ eigenspace of the operators of a stabilizer group.

We often describe a stabilizer group from a set of generators $g(\mathcal{S}) = \{S_1, S_2, ..., S_m\}$. A family of codes $\{\mathcal{C}(\mathcal{S}_n)\}_n$ is a finite-rate family if there exists a constant $c > 0$ such that $\frac{n-m}{n} > c$ as n goes to infinity. Finite-rate code families are crucial to large-scale fault-tolerant quantum computing since they allow to encode a constant quantity of logical qubits per physical qubits with vanishing error-rate. This is in constrast with zero-rate code families such as surface codes which encode only a small constant number of logical qubits using any given amount of physical qubits without significantly degrading the error-correcting performances [12].

A common class of stabilizer codes are Calderbank-Shor-Steane (CSS) codes [14, 28]. That is, codes for which there exists a set of generators such that each of them is a product of I and X only or I and Z only. Most techniques introduced in this work are applicable to non-CSS stabilzer codes. However, we focus on CSS codes to simplify the presentation since they are by far the most commonly studied codes.

Our construction is based on a graphical representation of the generators of a stabilizer code as shown in Figure 1. A Tanner graph $T = (Q \cup g(\mathcal{S}), E)$ is a bipartite graph with edge $\{q, S\} \in E$ if and only if stabilizer generator S acts non trivially on qubit q. Futhermore, in the case of CSS codes, we add an extra label to stabilizer nodes depending if their non-trivial action is X or Z. In this representation, two stabilizers S_X and S_Z with different actions commute if they share an even number of neighbors.

3 Codes from satisfiability constraints

Our strategy to obtain commuting sets of CSS stabilizer generators is to start with a large bipartite graph $G = (Q \cup g(\mathcal{S}), E_0)$ which we call a support graph. Then, we remove edges and label stabilizer nodes until all stabilizers commute. The result of this procedure is a Tanner graph $T = (Q \cup g(\mathcal{S}), E)$ such that $E \subseteq E_0$ together with a labelling function $l : g(\mathcal{S}) \to \{X, Z\}$.

To ensure commutation while generating the Tanner graph, we define a boolean constraint satisfaction problem from the support graph G. That is, we assign a boolean variable $\boldsymbol{v_a}(e) \in \{0, 1\}$ to each edge $e \in E_0$. We call these variables activators and $e \in E$ if and only if $\boldsymbol{v_a}(e) = 1$. We also define a Pauli variable $\boldsymbol{v_p}(S)$ for each stabilizer $S \in g(\mathcal{S})$ such that $l(S) = X$ if $\boldsymbol{v_p}(S) = 1$ and $l(S) = Z$ otherwise.

In the rest of this section, we define boolean constraints $\boldsymbol{c} : \{0, 1\}^* \to \{0, 1\}$ acting on subsets of activator and Pauli variables. Each constraint represents the commutation relation of a pair of sta-

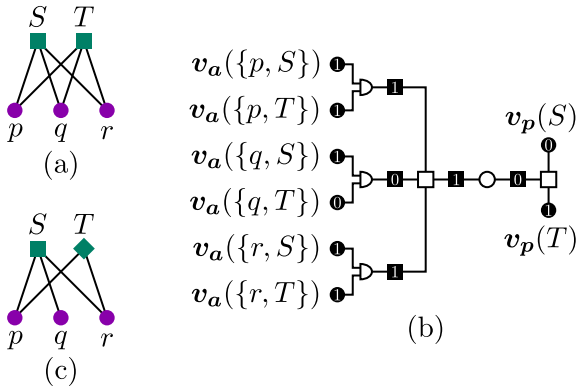

Figure 2: Graphical representation of the commutation constraints for a pair of stabilizers sharing three qubits. (a) The support graph G. The circles and squares respectively represent qubit and stabilizers. (b) The boolean variables and constraints assuring the commutation. Filled nodes correspond to variable. In particular circles are either activator or Pauli variables while the squares are auxiliary variables. The circle constraint correspond to Equation (2), the square constraints to Equations (3) and (4) and the half-circle constraints to Equation (5). The numbers within variable nodes illustrate a valid assignment. (c) The resulting Tanner graph according to the variable assignement.

bilizers and is equal to 1 if they commute and 0 if they anticommute. To build these constraints, we add auxiliary boolean variables to decompose them into smaller constraints. However, the values of these auxiliary variables are fixed by the activators and Pauli variables. Figure 2 illustrates the construction.

Two stabilizers commute if they satisfies at least one of the following conditions: they have the same non trivial action (X or Z) or they they are connected to an even number of common qubits. Thus, the commutation constraint $\boldsymbol{c_c}(S,S')$ between stabilizers S and S' can be decomposed as an OR constraint

$$\boldsymbol{c_c}(S,S') = \boldsymbol{v_s}(S,S') \vee \boldsymbol{v_e}(S,S'). \tag{2}$$

The auxiliary variable $\boldsymbol{v_s}(S,S') = 1$ when the action of S and S' is the same. This is represented by the constraint

$$\boldsymbol{c_s}(S,S') = \boldsymbol{v_s}(S,S') \oplus \boldsymbol{v_p}(S) \oplus \boldsymbol{v_p}(S'). \tag{3}$$

Similary, the variable $\boldsymbol{v_e}(S,S') = 1$ when S and S' have an even overlap. That is, $\boldsymbol{v_e}(S,S')$ needs to satisfy the constraint

$$\boldsymbol{c_e}(S,S') = \boldsymbol{v_e}(S,S') \oplus \left[\bigoplus_{q\in\eta(S,S')} \boldsymbol{v_b}(q,S,S') \right], \tag{4}$$

where $\eta(S,S')$ is the set of common neighbors of S and S' in G. The variable $\boldsymbol{v_b}(q,S,S') = 1$ when both edges $\{q,S\}$ and $\{q,S'\}$ are active. That is, it takes the value of

$$\boldsymbol{v_a}(\{q,S\}) \wedge \boldsymbol{v_a}(\{q,S'\}). \tag{5}$$

Equations (2) to (5) define a set of constraints on activator, Pauli and auxiliary variables. When all satisfied, they ensure that the resulting stabilizer generators all commute. Thus, any variable assignment such that all constraints have value 1 yields a valid CSS codes. In Section 3.2, we present how we find such assignements.

3.1 Extra constraints for good codes

The construction in the previous section, while assuring commutation, also allows for some naive codes. For example, a simple assignment satisfying all constraints is to fix all Pauli variables to the same value. If all Pauli variables have value 0, all stabilizers are of the X type and there is no protection against X errors.

To avoid these simple codes, we impose that every qubit is connected to at least δ_q stabilizers of each kind. To impose such a constraint, we introduce a new variable $\boldsymbol{v_X}(e)$ for each edge $e \in E_0$ taking the value

$$\boldsymbol{v_a}(\{q,S\}) \wedge \boldsymbol{v_p}(S). \tag{6}$$

That is, $\boldsymbol{v_X}(e) = 1$ if the edge is active and the corresponding stabilizer is of the X type. Then, we impose that the number of variables $\boldsymbol{v_X}(\{q,S\})$ with value 1 for $S \in \eta(q)$ is at least δ_q and 0 otherwise. We use similar constraints to impose a minimum number of Z edges per qubit.

These contraints together with the commutation constraints are the first set of constraints we numerically study in the following section. Subsequently, we add futher constraints to search for codes with improved decoding performances.

First, we impose to each stabilizer $S \in g(\mathcal{S})$ that the number of variables $\boldsymbol{v_a}(\{q,S\})$ with value 1 for

$q \in \eta(S)$ is at least δ_s. Then, to keep the codes sparse, we impose that at most Δ_S edges per stabilizer are active.

Finally, we impose that the number of stabilizers of each kind is balanced by enforcing that

$$\sum_{S \in g(\mathcal{S})} \boldsymbol{v_p}(S) = \left\lfloor \frac{|g(\mathcal{S})|}{2} \right\rfloor . \tag{7}$$

3.2 Constraint satisfaction problem solver

To obtain CSS codes from the aforementioned constraints, we use the *OR-Tools* library [3]. The constraints in Equations (4) to (6) are decomposed into XOR and OR constraints which are both native to the library and we use linear constraints to represent the minimum and maximum connectivity and balancing constraints. The implementation is provided online [4].

4 Results

4.1 Phase transition

Our goal is to find combinations of number of qubits and stabilizers and bounds on the degrees for which codes exist. To search for codes with n qubits and m stabilizers, we start by building random support graphs with the corresponding number of nodes by adding each edge with probability γ. We call γ the density of the graph and we use $G_{n,m,\gamma}$ to denote the corresponding random graph.

The property $\mathcal{P}$ that a graph support at least one solution to the constraint satisfaction problem defined above is monotone increasing. That is, if $E \subseteq E'$ and $G = (V, E) \in \mathcal{P}$ then, for any graph $G' = (V, E')$, we have $G' \in \mathcal{P}$. Thus,

$$\Pr[G_{n,m,\gamma} \in \mathcal{P}] \leq \Pr[G_{n,m,\gamma'} \in \mathcal{P}] \tag{8}$$

when $\gamma \leq \gamma'$ and there exists a threshold function $\gamma^*(n)$ such that [9]

$$\lim_{n\to\infty} \Pr[G_{n,m,\gamma} \in \mathcal{P}] = \begin{cases} 0 & \gamma(n)/\gamma^*(n) \to 0, \\ 1 & \gamma(n)/\gamma^*(n) \to \infty. \end{cases} \tag{9}$$

Our numerical results suggest this threshold decreases when the number of qubits increases.

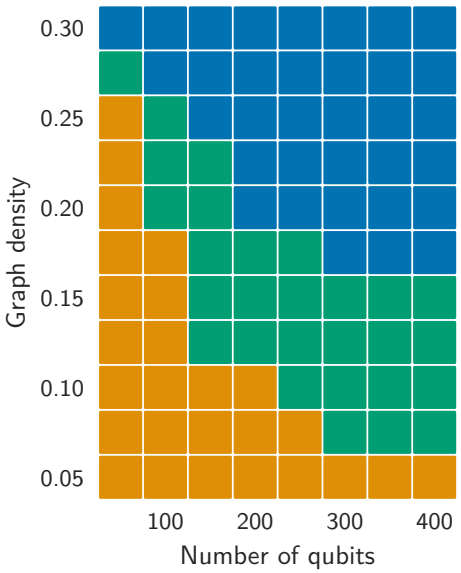

Figure 3: Satisfiability phase diagram for the commutation and minimum qubit degree constraints. The blue region is the set of combinations of number of qubits and initial graph density for which there are more satisfiable then unsatisfiable solvable instances. The orange region is the opposite where more unsatisfiable instances are found. The green region is where less than 10% of the instances are known.

To probe the performance of the solver, we start by imposing only the commutation constraints and a minimum qubit degree. That is, we impose $\delta_q = 3$, $\delta_s = 0$ and $\Delta_s = \infty$ for graphs with fixed ratio $m/n = 9/10$. To obtain data, we run the SAT solver for up to four hours using four CPU cores running at 2.4GHz. If no solution was found and no proof of unsatisfiability was produced, we label the instance as unknown. Note that most instances are solved in a fraction of the maximum time, except close to the threshold.

Figure 3 illustrates clearly that in this regime, the threshold density is decreasing with the number of qubits. Since we see no reason for this threshold to start increasing for larger block sizes, we are confident that in the worst case it plateaus to some value, unresolved by these simulations, of at most 15% or so. The unknown region is where the problem become really hard.

Also, Figure 4 shows that even without enforc-

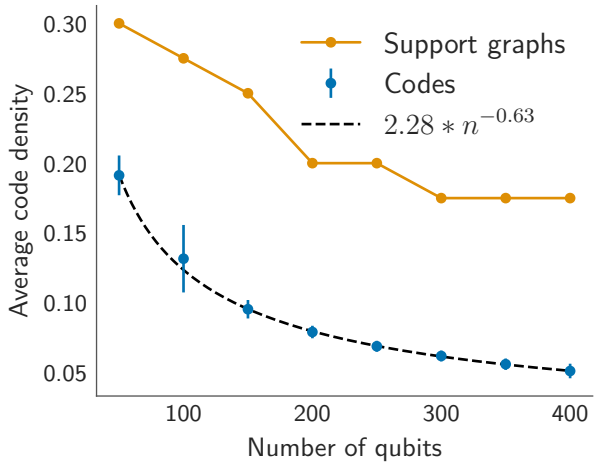

Figure 4: Densities of the resulting codes for commutation and minimum qubit degree constraints. For the support graphs, we plot the minimum density in the satisfiable region. For the codes, we plot the average density over all solutions with the same number of qubits.

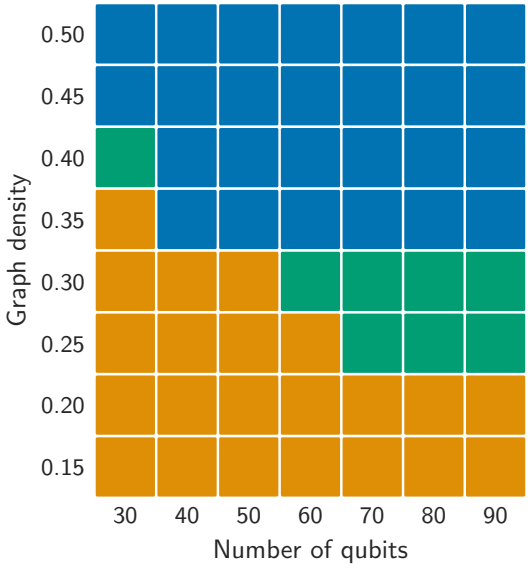

Figure 5: Satisfiability phase diagram with qubit degree, stabilizer degree and balancing constraints. See Figure 3 for color coding.

ing an upper bound on the stabilizer degrees, the densities of the Tanner graphs of the codes goes to zero as the number of qubits increase and are much lower than the densities of the support graphs. We observe similar results for different values of m/n.

To search for codes that could be used in a more realistic scenario, we impose all the extra contraints on the solver. That is, we impose $\delta_q = 3$, $\delta_s = 6$ and $\Delta_S = 20$ together with the stabilizer balancing constraint. For this regime of LDPC codes, Figure 5 also indicates that the threshold function is non-increasing when increasing the number of qubits. In both cases, this is a positive result since increasing thresholds would imply the problems become unsolvable for larger system sizes.

4.2 Decoding experiment

We shown numerical evidences that even if it is generally hard to find commuting sets of random stabilizer operators, it is relatively easy to generate such a set from a random bipartite graph as long as the density of the graph is above some threshold function. However, this result implies nothing about the decoding performance of the code generated this way. We want to know if the extra constraints we impose are enough to get good error correcting codes.

In this section, we probe the performance of the codes using the erasure channel as suggested in [16, 22]. The erasure channel is particularly useful since there exists a maximum likelyhood decoder running in polynomial time. This is in opposition to the depolarizing channel for which we don't have yet a good universal decoder.

The single-qubit erasure channel,

$$\mathcal{E}_p(\rho) = (1-p)\rho \otimes |0\rangle\langle 0| + p\frac{I}{2} \otimes |1\rangle\langle 1|\,, \qquad (10)$$

erases a qubit with probability p by replacing it by a completely mixed state. The second register indicates if the qubit was erased. Since

$$\frac{I}{2} = \frac{1}{4}(I\rho I + X\rho X + Y\rho Y + Z\rho Z), \qquad (11)$$

the multi-qubit version of the channel can be written as

$$\mathcal{E}_p^n(\rho) = \sum_{\mathbf{e}\in\{0,1\}^n} \Pr[\mathbf{e}] \left(\frac{1}{4^{|\mathbf{e}|}} \sum_{E\in P_n(\mathbf{e})} E\rho E \right) \otimes |\mathbf{e}\rangle\langle \mathbf{e}|\,, \qquad (12)$$

where $P_n(\mathbf{e})$ is the set of n-qubit Pauli operators with a trivial action on every qubit q_i for which $e_i = 0$. Then, after a measurement of the second register indentifying erasure positions the channel reduces to a Pauli channel on the erased qubits where each error is equaly likely. Therefore, from a measurement of the syndrome, a maximum likelyhood decoder search for any Pauli operator restricted to the erased location with the appropriate syndrome.

The success probability of this decoder is the inverse of the number of logical operators that can't be moved out of the erased qubits by multiplying a stabilizer. This probability can be computed by gaussian reduction as described in [22].

The capacity, i.e. the maximum rate of information, of the erasure channel [8] is computed from the probability of erasure. That is, the capacity of the channel with erasure probability p is $R_{\max} = 1-2p$. Inverting this relation, we observe that a code family with rate R can be used to protect information as long as the erasure probability is at most $\frac{1-R}{2}$. Figure 6 illustrates that the rate 1/10 codes obtained from the solver achieve this limit. In other words, we have a procedure to construct random stabilizer codes that are essentially capacity achieving for the erasure channel.

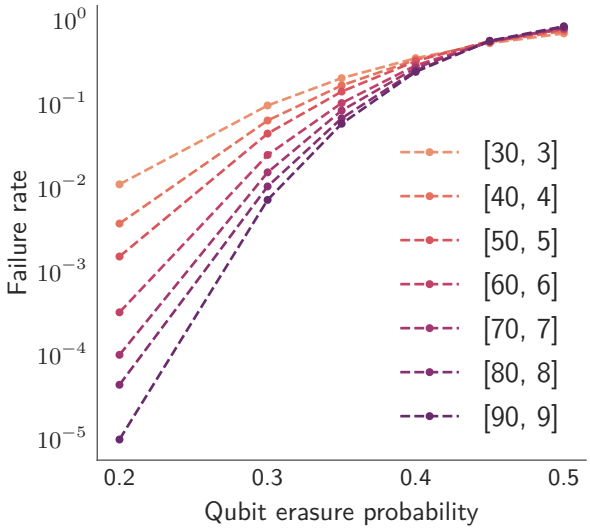

Figure 6: Failure rate for maximum-likelyhood decoding of the erasure channel. For each system size, we show the lowest failure rates amongst all codes found.

5 Discussion and outlook

We provide numerical evidence that we can construct quantum stabilizer codes from random bipartite graphs by leveraging constraint satisfiability solvers. This provides a connection with other random constraint satisfaction problems. We hope the further research can leverage this connection to guide search for new family of random stabilizer codes.

It is also important to note that even if we limit our analysis to CSS codes, the method we introduce is easily adaptable to more general stabilizer codes. For example, we could assign a extra variable to each edge to represent its Pauli value instead of labelling the stabilizer nodes directly. Futhermore, the method is flexible in the sense that is simple to add or remove contraints to fit one's needs.

During our study, we also limit the running time of the solver and the number of cores used to sample more data. However, one could use more computing power to search for larger codes. Also, impressive progress in the performance of solvers in the last decades (as shown by the numerous competions such as [1, 2]) leads us to believe that we will soon be able to search for larger codes without using more ressources.

In that regard, it would be interesting to study different random graph constructions as input to the solvers instead of the uniformly sampled graphs we used. This could lead to sparser initial graphs with structure favorable to the construction of codes. All this could result in improved solver performances.

Finally, we were able to use our approach to construct a family of finite-rate codes achieving optimal threshold for the erasure channel. This result, together with many recent stabilizer code constructions, motivate the search for more generally applicable quantum decoders for more complex noise channels such as the depolarizing channel and correlated Pauli noise. This would allow us to probe the performance of random code constructions more thoroughly.

Acknowledgement

We acknowledge the support of the Natural Sciences and Engineering Research Council of Canada (NSERC). Numerical ressources for this project were provided in part by Compute Canada and its regional partners, Calcul Québec and WestGrid. Finally, we thank all members of the QuiCoPhy research group for valuable discussions and feedbacks.

References

[1] MiniZinc - Challenge, . URL https://www.minizinc.org/challenge.html.

[2] SAT Competitions, . URL http://satcompetition.org/.

[3] OR-Tools - Google Optimization Tools, March 2022. URL https://github.com/google/or-tools. original-date: 2015-02-21T01:25:35Z.

[4] Stabilizer code generation from a CSP solver, June 2022. URL https://github.com/quicophy/csp_code_gen. original-date: 2022-06-10T17:30:48Z.

[5] Dimitris Achlioptas and Cristopher Moore. Random k-SAT: Two Moments Suffice to Cross a Sharp Threshold. *SIAM Journal on Computing*, 36(3):740–762, January 2006. ISSN 0097-5397. DOI: 10.1137/S0097539703434231. URL https://epubs.siam.org/doi/abs/10.1137/S0097539703434231. Publisher: Society for Industrial and Applied Mathematics.

[6] Dimitris Achlioptas, Assaf Naor, and Yuval Peres. Rigorous location of phase transitions in hard optimization problems. *Nature*, 435 (7043):759–764, June 2005. ISSN 1476-4687. DOI: 10.1038/nature03602. URL https://www.nature.com/articles/nature03602. Number: 7043 Publisher: Nature Publishing Group.

[7] Alexei Ashikhmin, Simon Litsyn, and Michael A. Tsfasman. Asymptotically good quantum codes. *Physical Review A*, 63(3): 032311, February 2001. DOI: 10.1103/PhysRevA.63.032311. URL https://link.aps.org/doi/10.1103/PhysRevA.63.032311. Publisher: American Physical Society.

[8] Charles H. Bennett, David P. DiVincenzo, and John A. Smolin. Capacities of Quantum Erasure Channels. *Physical Review Letters*, 78(16):3217–3220, April 1997. DOI: 10.1103/PhysRevLett.78.3217. URL https://link.aps.org/doi/10.1103/PhysRevLett.78.3217. Publisher: American Physical Society.

[9] B. Bollobás and A. G. Thomason. Threshold functions. *Combinatorica*, 7(1):35–38, March 1987. ISSN 1439-6912. DOI: 10.1007/BF02579198. URL https://doi.org/10.1007/BF02579198.

[10] S. B. Bravyi and A. Yu Kitaev. Quantum codes on a lattice with boundary. *arXiv:quant-ph/9811052*, November 1998. URL http://arxiv.org/abs/quant-ph/9811052. arXiv: quant-ph/9811052.

[11] Sergey Bravyi and Matthew B. Hastings. Homological product codes. In *Proceedings of the forty-sixth annual ACM symposium on Theory of computing*, STOC '14, pages 273–282, New York, NY, USA, May 2014. Association for Computing Machinery. ISBN 978-1-4503-2710-7. DOI: 10.1145/2591796.2591870. URL https://doi.org/10.1145/2591796.2591870.

[12] Sergey Bravyi, David Poulin, and Barbara Terhal. Tradeoffs for Reliable Quantum Information Storage in 2D Systems. *Physical Review Letters*, 104(5):050503, February 2010. ISSN 0031-9007, 1079-7114. DOI: 10.1103/PhysRevLett.104.050503. URL https://link.aps.org/doi/10.1103/PhysRevLett.104.050503.

[13] Winton Brown and Omar Fawzi. Short random circuits define good quantum error correcting codes. In *2013 IEEE International Symposium on Information Theory*, pages 346–350, July 2013. DOI: 10.1109/ISIT.2013.6620245. ISSN: 2157-8117.

[14] A. R. Calderbank and Peter W. Shor. Good quantum error-correcting codes exist. *Physical Review A*, 54(2):1098–1105, August 1996. DOI: 10.1103/PhysRevA.54.1098. URL https://link.aps.org/doi/10.1103/

PhysRevA.54.1098. Publisher: American Physical Society.

[15] Nicolas Delfosse. Tradeoffs for reliable quantum information storage in surface codes and color codes. In *2013 IEEE International Symposium on Information Theory*, pages 917–921, July 2013. DOI: 10.1109/ISIT.2013.6620360. ISSN: 2157-8117.

[16] Nicolas Delfosse and Gilles Zémor. Linear-time maximum likelihood decoding of surface codes over the quantum erasure channel. *Physical Review Research*, 2(3):033042, July 2020. DOI: 10.1103/PhysRevResearch.2.033042. URL https://link.aps.org/doi/10.1103/PhysRevResearch.2.033042. Publisher: American Physical Society.

[17] Austin G. Fowler, Matteo Mariantoni, John M. Martinis, and Andrew N. Cleland. Surface codes: Towards practical large-scale quantum computation. *Physical Review A*, 86(3):032324, September 2012. DOI: 10.1103/PhysRevA.86.032324. URL https://link.aps.org/doi/10.1103/PhysRevA.86.032324. Publisher: American Physical Society.

[18] R. Gallager. Low-density parity-check codes. *IRE Transactions on Information Theory*, 8 (1):21–28, January 1962. ISSN 2168-2712. DOI: 10.1109/TIT.1962.1057683. Conference Name: IRE Transactions on Information Theory.

[19] Daniel Gottesman. *Stabilizer Codes and Quantum Error Correction*. PhD Thesis, California Institute of Technology, 2004.

[20] Daniel Gottesman. Fault-Tolerant Quantum Computation with Constant Overhead. *arXiv:1310.2984 [quant-ph]*, October 2013. URL http://arxiv.org/abs/1310.2984.

[21] Antoine Grospellier, Lucien Grouès, Anirudh Krishna, and Anthony Leverrier. Combining hard and soft decoders for hypergraph product codes. *arXiv:2004.11199 [quant-ph]*, April 2020. URL http://arxiv.org/abs/2004.11199.

[22] Michael J. Gullans, Stefan Krastanov, David A. Huse, Liang Jiang, and Steven T. Flammia. Quantum Coding with Low-Depth Random Circuits. *Physical Review X*, 11(3): 031066, September 2021. DOI: 10.1103/PhysRevX.11.031066. URL https://link.aps.org/doi/10.1103/PhysRevX.11.031066. Publisher: American Physical Society.

[23] Matthew B. Hastings, Jeongwan Haah, and Ryan O'Donnell. Fiber Bundle Codes: Breaking the $N^{1/2} \operatorname{polylog}(N)$ Barrier for Quantum LDPC Codes. *arXiv:2009.03921 [quant-ph]*, October 2020. URL http://arxiv.org/abs/2009.03921. arXiv: 2009.03921.

[24] Aleksander Kubica, Michael E. Beverland, Fernando Brandão, John Preskill, and Krysta M. Svore. Three-Dimensional Color Code Thresholds via Statistical-Mechanical Mapping. *Physical Review Letters*, 120(18):180501, May 2018. DOI: 10.1103/PhysRevLett.120.180501. URL https://link.aps.org/doi/10.1103/PhysRevLett.120.180501. Publisher: American Physical Society.

[25] Andrew J. Landahl, Jonas T. Anderson, and Patrick R. Rice. Fault-tolerant quantum computing with color codes. *arXiv:1108.5738 [quant-ph]*, August 2011. URL http://arxiv.org/abs/1108.5738. arXiv: 1108.5738.

[26] Pavel Panteleev and Gleb Kalachev. Asymptotically Good Quantum and Locally Testable Classical LDPC Codes. *arXiv:2111.03654 [quant-ph]*, January 2022. URL http://arxiv.org/abs/2111.03654. arXiv: 2111.03654.

[27] Tom Richardson and Ruediger Urbanke. *Modern Coding Theory*. Cambridge University Press, Cambridge, 2008. ISBN 978-0-511-79133-8. DOI: 10.1017/CBO9780511791338. URL http://ebooks.cambridge.org/ref/id/CBO9780511791338.

[28] A. M. Steane. Simple quantum error-correcting codes. *Physical Review A*, 54(6):4741–4751, December 1996. DOI: 10.1103/PhysRevA.54.4741. URL https://link.aps.org/doi/10.1103/PhysRevA.54.4741. Publisher: American Physical Society.

[29] Ashley M. Stephens. Fault-tolerant thresholds

for quantum error correction with the surface code. *Physical Review A*, 89(2):022321, February 2014. DOI: 10.1103/PhysRevA.89.022321. URL https://link.aps.org/doi/10.1103/PhysRevA.89.022321. Publisher: American Physical Society.

[30] Jean-Pierre Tillich and Gilles Zémor. Quantum LDPC Codes With Positive Rate and Minimum Distance Proportional to the Square Root of the Blocklength. *IEEE Transactions on Information Theory*, 60(2):1193–1202, February 2014. ISSN 1557-9654. DOI: 10.1109/TIT.2013.2292061. Conference Name: IEEE Transactions on Information Theory.

[31] Maxime A. Tremblay, Nicolas Delfosse, and Michael E. Beverland. Constant-overhead quantum error correction with thin planar connectivity. *arXiv:2109.14609 [quant-ph]*, September 2021. URL http://arxiv.org/abs/2109.14609. arXiv: 2109.14609.