

Cellular automaton decoders of topological quantum memories in the fault tolerant setting

This content has been downloaded from IOPscience. Please scroll down to see the full text.

2017 New J. Phys. 19 063012

(<http://iopscience.iop.org/1367-2630/19/6/063012>)

View [the table of contents for this issue](#), or go to the [journal homepage](#) for more

Download details:

IP Address: 143.167.30.96

This content was downloaded on 13/06/2017 at 16:35

Please note that [terms and conditions apply](#).

You may also be interested in:

[Fast decoders for qudit topological codes](#)

Hussain Anwar, Benjamin J Brown, Earl T Campbell et al.

[Logical error rate scaling of the toric code](#)

Fern H E Watson and Sean D Barrett

[Quantum error correction for beginners](#)

Simon J Devitt, William J Munro and Kae Nemoto

[Thermodynamic stability criteria for a quantum memory based on stabilizer and subsystem codes](#)

Stefano Chesi, Daniel Loss, Sergey Bravyi et al.

[Improved HDRG decoders for qudit and non-Abelian quantum error correction](#)

Adrian Hutter, Daniel Loss and James R Wootton

[Stability of topologically-protected quantum computing proposals as seen through spin glasses](#)

H G Katzgraber and R S Andrist

[Tricolored lattice gauge theory with randomness: fault tolerance in topological color codes](#)

Ruben S Andrist, Helmut G Katzgraber, H Bombin et al.

[Constructions and noise threshold of topological subsystem codes](#)

Martin Suchara, Sergey Bravyi and Barbara Terhal

[Self-correcting quantum computers](#)

H Bombin, R W Chhajlany, M Horodecki et al.



PAPER

Cellular automaton decoders of topological quantum memories in the fault tolerant setting

OPEN ACCESS

RECEIVED

12 January 2017

REVISED

11 April 2017

ACCEPTED FOR PUBLICATION

2 May 2017

PUBLISHED

6 June 2017

Michael Herold¹, Michael J Kastoryano^{2,4}, Earl T Campbell³ and Jens Eisert¹¹ Dahlem Center for Complex Quantum Systems, Freie Universität Berlin, D-14195 Berlin, Germany² NBIA, Niels Bohr Institute, University of Copenhagen, DK-2100 Copenhagen, Denmark³ Department of Physics and Astronomy, University of Sheffield, Sheffield S3 7RH, United Kingdom⁴ Author to whom any correspondence should be addressed.E-mail: mkastoryano@gmail.com**Keywords:** quantum memory, quantum error correction, dissipative dynamics, decoders, topological order, Liouvillians

Original content from this work may be used under the terms of the [Creative Commons Attribution 3.0 licence](https://creativecommons.org/licenses/by/4.0/).

Any further distribution of this work must maintain attribution to the author(s) and the title of the work, journal citation and DOI.

**Abstract**

Active error decoding and correction of topological quantum codes—in particular the toric code—remains one of the most viable routes to large scale quantum information processing. In contrast, passive error correction relies on the natural physical dynamics of a system to protect encoded quantum information. However, the search is ongoing for a completely satisfactory passive scheme applicable to locally interacting two-dimensional systems. Here, we investigate dynamical decoders that provide passive error correction by embedding the decoding process into local dynamics. We propose a specific discrete time cellular-automaton decoder in the fault tolerant setting and provide numerical evidence showing that the logical qubit has a survival time extended by several orders of magnitude over that of a bare unencoded qubit. We stress that (asynchronous) dynamical decoding gives rise to a Markovian dissipative process. We hence equate cellular-automaton decoding to a fully dissipative topological quantum memory, which removes errors continuously. In this sense, uncontrolled and unwanted local noise can be corrected for by a controlled local dissipative process. We analyze the required resources, commenting on additional polylogarithmic factors beyond those incurred by an ideal constant resource dynamical decoder.

1. Introduction

Quantum coherence is capricious, and taming it requires sophisticated methods of control and significant resource overhead. So far the best candidates are based on Kitaev's toric code [1, 2], and variants thereof. While quantum information can be stored in ground states of a toric Hamiltonian, thermal excitations are unconfined in two or three-dimensions, allowing thermal diffusion and corrupting data in constant time [3–7]. Using active decoding, this problem can be overcome. Errors must be regularly monitored by collecting syndrome information, which is analyzed by a decoding algorithm and corrected for. The first steps towards experimental implementations of error-correcting protocols are under way [8–11].

Given reliable snapshots of error syndromes, decoding algorithms can be independently applied to each time slice [12]. Decoders exist using notions of minimum-weight perfect matching [13, 14], renormalization [15, 16], and simulated annealing [17, 18], which is highly problematic for 2D architectures, where on chip wiring becomes cumbersome beyond nine physical qubits. However, parallelized decoding is also possible with cellular-automaton decoders that locally store syndrome information, communicating with only nearest neighbors. Each node in the cellular automaton stores a limited amount of additional data, and our focus here is on so-called ϕ -automata [19] where this data is a single variable, or ϕ -value, per node.

Yet, perfect syndrome measurement is a fantasy. It suffices for developing toy decoders, but is merely the first step to practical decoding. Conventionally, one tackles measurement errors by recording a history of recent syndrome snapshots, then running a decoder suitable for three-dimensional syndrome distributions [13, 14].

In contrast, here we propose the concept of a Markovian dynamical decoder, where each step of the decoder only depends on the present configuration of (faulty) syndromes measurements and the auxiliary ϕ -field. Specifically, we show that a ϕ -automaton decoder can tolerate measurement noise without significant changes to the algorithms, while preserving a number of desirable features. Measurement errors cause ripples in the dynamics, but may be naturally tolerated, provided fresh syndrome information is acquired before the ripple travels too far. No explicit record of the syndrome history is required, rather past syndromes merely leave echoes in the state of the cellular automaton. To the best of our knowledge, there is only one prior study of a Markovian dynamical decoder; due to Harrington [20]. Interest in this early work has recently been revived [21], and appears to be the right setting for certain types of non-Abelian anyon decoding [22, 23].

The Markovian dynamical decoder paradigm is naturally suited to various physical implementations where syndromes can not easily be measured synchronously. In systems where entanglement is generated probabilistically [24–27], including syndrome measurements [28], this situation readily applies. Rather than a well defined series of snapshots, the syndromes are refreshed *asynchronously*; meaning that every unit cell of the cellular-automaton decoder is updated with a certain probability that is independent of the other cells. The dynamical paradigm is equally plausible in other architectures with an inherently modular structure of interacting nodes, as is notably the case in anticipated mesoscopic implementations of the toric code [29, 30], based on Majorana bound states supported by topological semi-conductor nanowires.

The locality, Markovianity and asynchronicity of such decoders makes them conceptually equivalent to certain physical systems undergoing open system dynamics. We show that an asynchronous cellular-automaton decoder can be considered as the discretization of a Markovian master equation which dissipatively protects the topological subspace of the code. This opens up the potential of engineering quantum memories from locally interacting matter systems, providing passive protection against noise. In this sense, one can think of passive dissipative quantum memories protected against unwanted dissipation. Finally, from rudimentary numerical exploration, dynamical cellular-automaton decoders seem promising in the presence of qubit loss or in the presence of spatially and temporally correlated noise, though these settings are all beyond the scope of the present article.

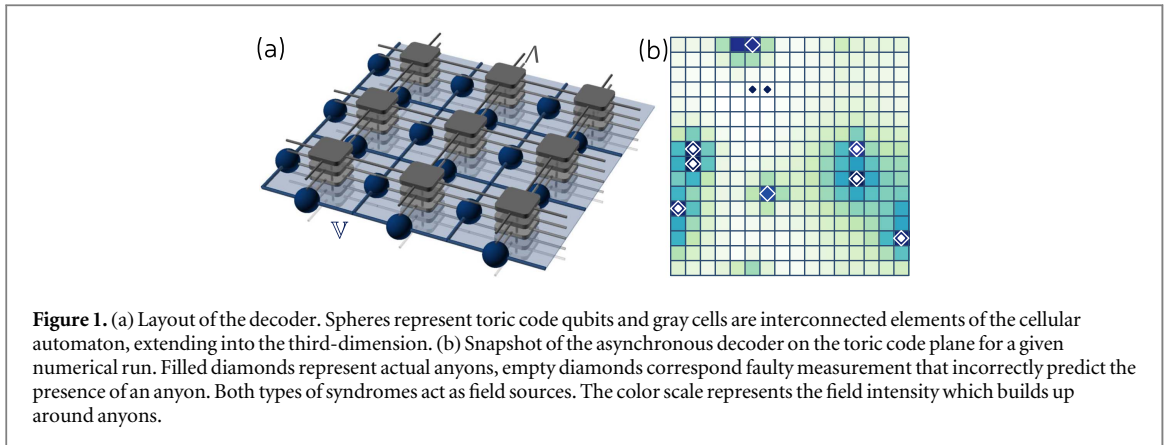
An asynchronous cellular automaton is most naturally built from a synchronous one that does not require a clock or time dependent rules. One such time homogeneous cellular automaton is provided by the 3D ϕ -automaton of [19]. In contrast, the 2D* ϕ -automaton of [19] and an automaton studied by Harrington [20, 31] both keep track of time, and so are less easily adapted to an asynchronous setting. For completeness, we later review the synchronous operation of the Harrington decoder. We show that our dynamical 3D ϕ -automaton decoder provides such an asynchronous dynamical decoder.

2. The decoders

2.1. ϕ -automaton decoders

Decoders in our context are processes that remove errors from a lattice configuration \mathbb{V} and bring the system's state back to the code space. Here we focus on the toric code that uses an L by L square lattice with periodic boundary conditions. For simplicity, we assume uncorrelated X and Z Pauli noise. This allows us to analyze only Pauli X errors, which are measured by *plaquette operators*, with Z noise decoding behaving identically and independently. To restrict ourselves to algorithms that can be highly parallelized and are inherently local, we formulate all algorithms as cellular automata (CA). For every measurable syndrome in the code we also place a cell for the automaton, constituting a parallel lattice Λ with one site per plaquette. The cell has local access to the measured syndrome and can read information from the neighboring cells on the CA lattice Λ of side length L . If a -1 syndrome is measured, we will say that an *anyon* is present. The anyons can be moved locally by applying (or equivalently storing) local bit flip operations. The automaton possesses the same periodic boundary conditions as the code does. As we will see, it is sometimes convenient to allow the CA lattice Λ to extend in the third-dimension (see figure 1).

We specifically focus on and generalize a scheme for CA decoding the 2D toric code, introduced in [19]. The underlying idea is to simulate an attractive interaction between anyons from local update rules. The attraction is mediated via a scalar field $\phi(\mathbf{x})$ on the discrete lattice $\mathbf{x} \in \Lambda$. Sites occupied by anyons act as sources for the field. A field value is stored at each cell of the automaton, and the field builds up according to a local discretized version of Poisson's equation. As it turns out, the field has to be generated via a CA which extends into the third-dimension so as to guarantee that the field falls off as $1/r$ away from anyons [19]. That way, nearby anyons are attracted to each other under the influence of the field, but do not feel the attraction of far away anyons very strongly. Generating such a field locally in two-dimensions seems to be difficult, but it occurs naturally in 3D. We will briefly review the so-called 3D ϕ -automaton depicted in figure 1(a). Other passive memories have been



proposed where an effective interaction emerges in the system Hamiltonian [32, 33], but here the interaction is simulated by the dynamics of a CA rather than being a manifestly physical interaction.

The cellular automaton update rules dynamically evolve a combined quantum–classical system. For ease of presentation, the field and anyon positions are embedded in quantum systems with Hilbert spaces \mathcal{L} and \mathcal{V} , respectively. We use σ to denote states within the composite space $\mathcal{H} \otimes \mathcal{L} \otimes \mathcal{V}$, where \mathcal{H} is the toric code Hilbert space. The update rules are composed of two elementary components: field updates F and anyon updates A . The field update F simultaneously replaces the field value at all sites by the average of all six neighboring field values and adds +1 when an anyon is present. The anyon update A moves anyons to the one of their four neighboring sites in the toric code plane with the highest associated ϕ -value. Unwanted limit cycles are avoided by moving with a probability of 1/2.

In order for two anyons to be attracted to each other, they must ‘feel’ the field from the other anyon. In other words, the influence of an anyon on the field ϕ needs to have propagated to other anyons. To ensure a sufficient convergence of the field between anyon updates, it is required that F is applied $c \sim \log^2 L$ times [19]. This yields the compound update rule $D_c := AF^c$. Applying the update D_c several times guarantees that typical clusters of errors get collapsed back to an anyon free configuration.

2.2. Markovian dynamical decoders

We next introduce the notion of a *Markovian dynamical decoder*, for the moment still in the synchronous setting. A dynamical decoder is to be contrasted with a regular decoder in that a regular decoder minimizes the number of residual errors at each time slice. At a given instance in time, a dynamical decoder only chooses to remove some of the error, according to a suitable prescription. A Markovian dynamical decoder keeps no memory of previous measurement rounds. Dynamical decoders do not need to be local, as in [14], but we will only consider local ones in this work. In fact, dynamical decoding is expected to be particularly advantageous if constraints of locality are to be respected.

We will now turn to making these notions more precise. We denote by E_p the error map that applies X -flips on each qubit of the toric code with probability p . The imperfect error syndrome extraction is reflected by the application of a map M_q , where $q > 0$ denotes the probability that a syndrome is incorrect. For perfect syndrome measurement, $q = 0$, decoding can be achieved with only one round of syndrome extraction and a suitable number τ of updates D_c ,

$$\sigma_\tau = (D_c^\tau M_0 E_p)(\sigma), \quad (1)$$

where σ is a state on the combined quantum–classical Hilbert space. The quantum component is the toric code and the classical component represents both the ϕ -field and the anyon locations reported by measurements. The exponent τ must be chosen such that the decoder terminates. If the error rate p is below a certain critical threshold, a logical error is returned with probability exponentially suppressed in L . However, for active error correction only *partial* decoding is necessary between different rounds of syndrome measurements. In particular, one may consider very short *sequences* in which only partial corrections are calculated before the syndrome is remeasured. As every recovery step only depends on the instantaneous time syndrome information, we refer to this concept as *Markovian dynamical decoding*. Specifically, we define synchronous dynamical decoding as the succession of the maps

$$\sigma_\tau = (D_c M_q E_p)^\tau(\sigma), \quad (2)$$

where τ is the number of sequences. In the setting of dynamical decoding, we are interested in estimating the survival time of the logical subspace; i.e. how long one typically has to wait before a logical X or Z error occurs on

the encoded qubits. In order to determine this survival time we check whether a logical error has occurred by decoding a copy of the state without measurement errors in every sequence.

It is important to note that between every round of measurement, the dynamical decoder will leave many residual errors in the system. This is even the case when the measurement error goes to zero. This is in stark contrast with conventional decoders designed for perfect measurements, which try to minimize the residual errors in the lattice between every measurement. A code is called *single-shot error correcting*, if the residual errors can be minimized at each step in the presence of measurement errors, while still guaranteeing a decoding threshold [34, 35]. Dynamical decoders will not, in general, perform single-shot error correction but the converse is true.

2.3. Decoding in continuous time

Equation (2) defines synchronous Markovian dynamical decoding, where for the ϕ -automaton decoder the field updates F and the anyon updates A are applied simultaneously on all sites at each time step. A more physically relevant scheme would be one that is naturally *asynchronous*. We consider the local operators $f_{\mathbf{x}}$ and $a_{\mathbf{x}}$ performing field and anyon updates but only applied to a single site \mathbf{x} , and the $X_{p,\mathbf{v}}$ -operator applying a Pauli operator to the edge $\mathbf{v} \in \mathbb{V}$ with probability p . The $m_{q,\mathbf{x}}$ -operator reflects a measurement with error probability q at $\mathbf{x} \in \Lambda$. We pick a site \mathbf{x} (or edge \mathbf{v} in the case of $X_{p,\mathbf{v}}$) at random, and apply the local map to that site (or edge). By considering non-zero q , we allow for noisy measurements. Every site keeps a record of its most recent measurement result, and the field updates are calculated on the basis of this most recent result. Therefore, field updates are based on a record of anyon locations that is both noisy and not always up-to-date. Furthermore, the probability to choose $f_{\mathbf{x}}$ is weighted with $c \sim \log^2 L$, so we incur a polylogarithmic increase in the rate of field updates relative to anyonic and measurement processes. To compare the performance of synchronous and asynchronous dynamical decoders, we define one time step as the number of operations after which all anyons on average had the chance to move once.

3. Numerical results

3.1. Results from simulations

We have simulated the proposed synchronous and asynchronous Markovian dynamical decoders for even lattice sizes $L = 12, 14, \dots, 24$ by Monte Carlo sampling. We allowed for measurement errors $q = p$ and considered a range of physical error probabilities. Figure 2(a) presents results for the synchronous dynamical decoder, and figure 2(b) presents the asynchronous case. Each data point reflects the average over 5000 samples. For both decoders, we see clear evidence of survival times extended by several orders of magnitude across the parameter range studied. Typically, one aims to numerically identify a threshold, below which survival times are exponentially prolonged with increasing lattice size. Before discussing possible threshold values, we review the standard techniques for identifying thresholds.

An established methodology is to look for a common crossing between curves for the various lattice sizes. Yet, in practice, even when a threshold exists, the crossings drift to lower error rates with increasing lattice size. At sufficiently large lattice sizes, this drift will become negligible and a common crossing is observed and taken as firm evidence of a threshold. A more sophisticated approach is to fit data to a scaling law [13, 20]

$$\langle T(L, p) \rangle = A + B(p - p_{\text{fit}})L^{1/\nu} + C(p - p_{\text{fit}})^2 L^{2/\nu} + DL^{-1/\mu}, \quad (3)$$

where the term $DL^{-1/\mu}$ accounts for finite lattice size effects. The variable p_{fit} gives a threshold estimate and will be smaller than the lowest observed crossing, having extrapolated to account for finite size effects.

Our numerical investigations show crossings in the range 0.22%–0.36% for the synchronous decoder and for the 0.34%–0.52% asynchronous case. The lowest crossings at $p_{\text{cross}} = 0.22\%$ (synchronous) and $p_{\text{cross}} = 0.30\%$ (asynchronous) correspond to the crossing between the $L = 22$ and $L = 24$ curves and give threshold upper bounds. Next, we select the data with p close to these lowest crossings. Specifically, for the synchronous decoder we select data points with $p = \{0.18\%, 0.223\%, 0.2763\%\}$ and for the asynchronous case we select data points with $p = \{0.223\%, 0.2763\%, 0.3424\%\}$. Fitting this data to the above scaling law, gives threshold estimates of $p_{\text{fit}} = 0.17\%$ (synchronous) and $p_{\text{fit}} = 0.25\%$ (asynchronous). As expected, these p_{fit} values lie below the corresponding lowest crossing p_{cross} . However, they are sufficiently far below the lowest crossing that we can not conclusively claim to have found a threshold at the value p_{fit} .

We conclude that finite size effects are still prominent at $L = 24$. The use of the scaling law fitting is intended to adjust for such effects, but can not be relied upon for large adjustments. Ideally, data at higher L would be available, but we are already at the limits of available computational power. Other decoders have been able to collect reliable data at higher L . While CA can be expected to run quickly on custom hardware, our simulations were executed on a conventional serial processor, leading to an $O(L^3)$ slow down in the runtime, which

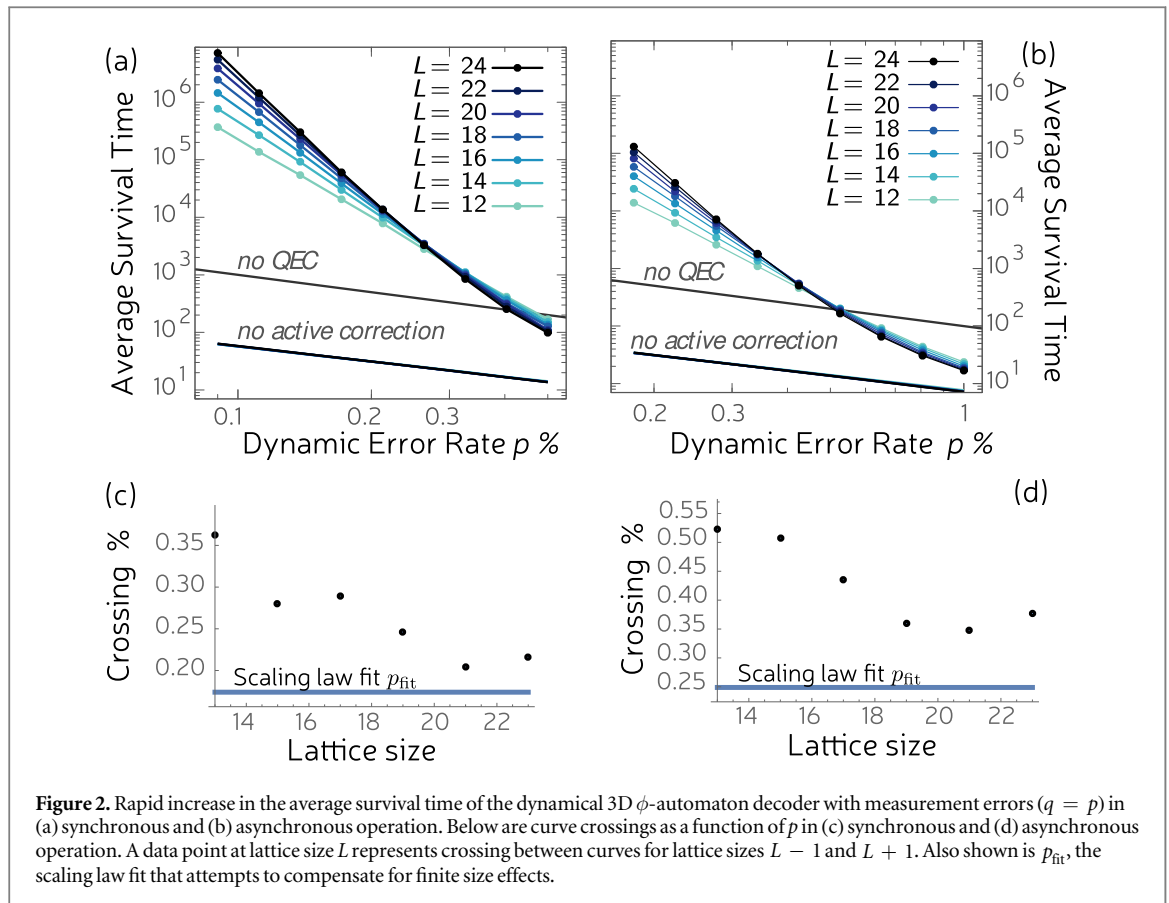


Figure 2. Rapid increase in the average survival time of the dynamical 3D ϕ -automaton decoder with measurement errors ($q = p$) in (a) synchronous and (b) asynchronous operation. Below are curve crossings as a function of p in (c) synchronous and (d) asynchronous operation. A data point at lattice size L represents crossing between curves for lattice sizes $L - 1$ and $L + 1$. Also shown is p_{fit} , the scaling law fit that attempts to compensate for finite size effects.

constrains the range of accessible L values. The need to sample at small probabilities and long survival times, also leads to great computational overhead since more samples are required to obtain good statistics.

It is important to note that ϕ -field decoders suffer very significant finite size effects up to large system sizes. Evidence of this is found in section 3.3 where we investigate a related class of decoders that behave very similarly, but which can reach higher L values on a conventional computer architecture. This opens the possibilities of inferring higher L performance for our CA decoders by bootstrapping our numerical data onto data provided by these related decoders. However, we again found that finite size effects were visibly clear up to $L = 50$, and even up to $L = 80$ did not provide a scaling law fit p_{fit} that was comparable to the lowest observed crossings.

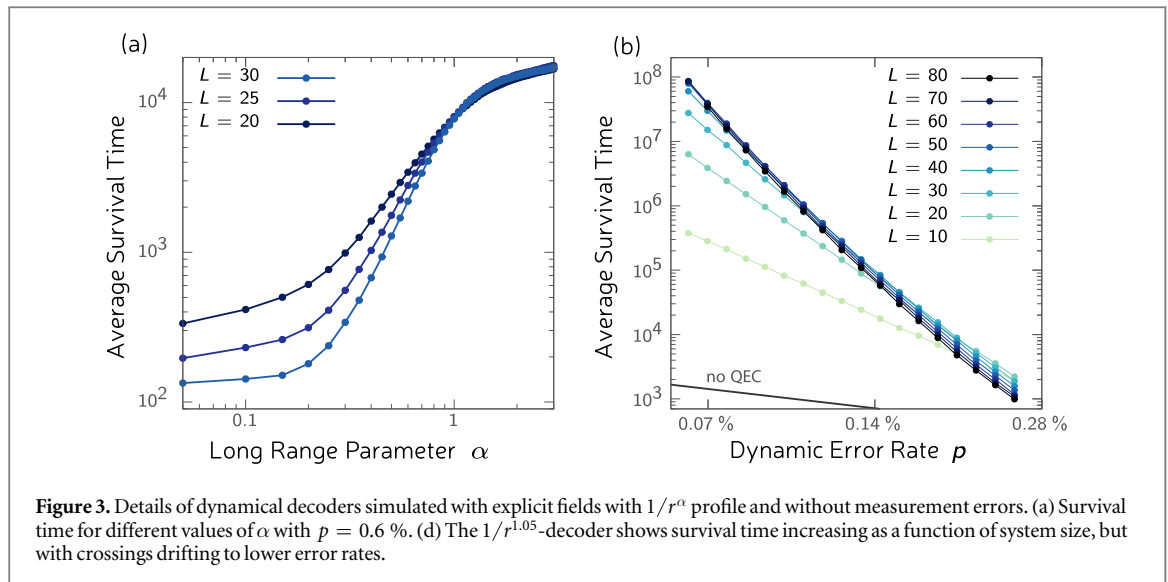
These difficulties seem to be generic for Markovian dynamical decoders so far proposed. For instance, the dynamical decoder identified by Harrington [20] is believed to have a threshold by virtue of an extension of the work of Gács [36–38]. However, all numerical investigations [20, 31] to date have yet to find a common crossing over a wide range of lattice sizes.

3.2. Measurement errors

For many of the existing topological quantum codes, it is necessary to keep a log of the syndrome information whenever there are measurement errors. Only codes allowing for *single shot decoding* [34, 35] can cope with measurement errors in the setting of equation (1). One of the main benefits of both synchronous and asynchronous dynamical decoders is that measurement errors are naturally incorporated without any additional overhead. This feature is inherited from the local structure of the decoding algorithm: if the presence of an anyon is falsely indicated, this may only result in a single additional bit-flip error created by the local anyon move. In the next sequence this bit flip is likely incorporated into the syndrome and is not different from usual errors. Also, the imaginary anyon may attract other anyons, which is not different from the case where a real error occurred.

3.3. Instantly updated ϕ -decoders

The simulation has to run throughout the full survival time, which grows very rapidly with increasing system size and decreasing error rates. Therefore, the simulation of dynamical 3D ϕ -automaton decoders on traditional architecture is numerically expensive. Furthermore, we simulate the 3D CA within a serial centralized computing model, which leads to an additional time cost scaling with L^3 . Parallelization exploiting the cellular nature of the decoder would eliminate this cost, but has not yet been implemented. Accordingly, the simulation is clearly limited in system size to low error rates.



To obtain results for larger system sizes we replace the ϕ -automaton by explicitly calculating steady-state field values. The appropriate field for a 3D ϕ -automaton is a superposition of fields decaying as $1/r$, where r represents the Manhattan, or 1-norm, distance from the anyon position. The anyon-move rules are identical to the synchronous CA decoder. The dimensionality of a ϕ -automaton determines for which number of dimensions the discretized Poisson equation is solved through the iterative algorithm. To construct a ϕ -automaton decoder with a time independent parameter c , it seems necessary to choose a three-dimensional ϕ -automaton [19]. Figure 3(a) gives the results for a dynamical decoder that operates with explicitly calculated field values $1/r^\alpha$ for different choices of the parameter α . We find that the parameter α has a strong impact on the survival time. An increase of survival time with lattice size is only present for $\alpha \geq 1$. The iterative field generation via a CA leads to incomplete convergence of the field leading to fields with slightly shorter range. To incorporate this fact we opted to study an explicit field decaying as $1/r^{1.05}$. For further details on the field profile and convergence properties we refer to [19].

As expected, the explicit field $1/r^{1.05}$ decoder and the 3D CA decoder show similar behavior. Numerics up to system size $L = 80$ can be found in figure 3(b). Fitting equation (3) we obtain threshold values p_{fit} from 0.03 % to 0.05 % with measurement errors $q = p$. The range is obtained by excluding different amounts of small system size data and varying the used error ranges p . However, all our data and observed crossings are for 0.07% and above. Since p_{fit} is again extrapolated to below the regime where we have concrete data, we are again unable to decisively conclude that p_{fit} is an accurate threshold estimate. In other words, figure 3(b) tells us that if there is a threshold, then it must be below 0.07%. Since the crossings for larger lattice sizes happen at average survival times of $\sim 10^8$ time steps, we cannot directly probe the behavior below threshold, and cannot guarantee that there is exponential increase in survival time as a function of system size below a critical p . Figure 3(b) is consistent with other numerical studies of Markovian dynamical decoders in [20, 31].

4. Further conceptual characteristics of Markovian dynamical decoders

4.1. Decoding by open systems dynamics

The concept of dynamical decoders is motivated by the theory of classical robust memories. In this context, CA are studied that redundantly store the information in their local state space. Appropriately tailored update rules can correct local errors and can be used for active error correction. Such error correcting features appear in two fundamental varieties. First, there exist automata which are called *eroders* or *washout automata*. Considering a binary CA, the repeated action of an eroder leads to all cells having the value of the initial states majority occupation. In this way, eroders are the equivalent of a non-dynamical decoder in the absence of measurement errors. Both systems return a partially erroneous memory state to an error free state. It is suggestive to imagine that in this setting the errors only act at the first time slice and afterwards the eroder can act unimpeded. The second variety appears if we let errors appear at every single time slice. In this setting the following question is asked: given the initial state is a fixed point of the automaton, is this fixed point stable under local perturbations? This setting corresponds to dynamical decoding where local update rules permanently counteract to the corruption of memory.

Under certain assumptions for dimensional binary CA Toom's theorem applies and being an eroder is equivalent to the existence of stable fixed points. The most prominent example of such a CA is the North–East–Center (NEC) rule, also referred to as Toom's rule. It updates each cell by majority of the three (NEC) cells. For an initial i.i.d. configuration with probability p , the automaton calculates the majority if $p \neq 1/2$, i.e. the automaton is an eroder. This corresponds to a decoder with threshold $p_{\text{th}} = 1/2$. Considering the symmetry of the update rule and employing Toom's theorem it follows that the automaton must possess two robust fixed points. However, it is implausible, that the system can tolerate an error rate close to $1/2$ at every time step. This intuition can be made rigorous in the proof of phase coexistence. Consequently, the threshold for the fault tolerance of attractive states has to be lower than $1/2$. This lower threshold would correspond to the threshold of a dynamical decoder.

Using these insights into classical CA, we formulate the following rules of thumb for dynamical decoders: first, a dynamical decoder will usually also work as non-dynamical decoder. And second, the dynamical threshold is lower than the non-dynamical threshold. To avoid confusion, it should be stated that survival time simulations can also be performed with non-dynamic decoders. Here, both the decoding without measurement errors and the decoding in $2 + 1$ dimensions with measurement errors classify as non-dynamic. For such experiments, the decoder threshold and the survival time threshold is identical.

4.2. Semigroup representation

An asynchronous cellular automaton can be understood as the evolution induced by a time independent Markovian master equation. At a constant rate, the coherent evolution is interrupted by quantum jumps corresponding to local unitary operations. Such a prescription precisely corresponds to a continuous one parameter semigroup, here acting on both the quantum state, as well as the effectively classical record of the anyon location and the classical auxiliary fields. The dynamical equation is given by

$$\frac{\partial \sigma}{\partial t} = \mathcal{L}(\sigma), \quad (4)$$

where \mathcal{L} is the Liouvillian decomposed as

$$\mathcal{L} := \sum_j \mathcal{L}_j \quad (5)$$

and each local term takes the form

$$\mathcal{L}_j(\sigma) = \gamma_j(L_j \sigma L_j^\dagger - L_j^\dagger L_j \sigma / 2 - \sigma L_j^\dagger L_j / 2). \quad (6)$$

The $\{L_j\}$ are called Lindblad operators and $\{\gamma_j > 0\}$ can be viewed as the rates of the dissipative processes. Specifically, we construct a Liouvillian where the geometrically local jump operators L_j are constructed from $X_{p,\mathbf{v}}$, $m_{q,\mathbf{x}}$, $f_{\mathbf{x}}$, and $a_{\mathbf{x}}$, for all \mathbf{x} and \mathbf{v} , with corresponding weights $\{\gamma_x, \gamma_m, \gamma_f, \gamma_a\}$. The γ_f rates must now scale as $c \sim \log^2 L$ relative to all other rates to emulate the scaling of c . In other words, asynchronous dynamical decoding can be understood as a dissipative process

$$\mathcal{L} = \mathcal{L}_C + \mathcal{L}_E, \quad (7)$$

where the dissipative decoder reflected by \mathcal{L}_C continuously 'corrects' an uncontrollable noise operator \mathcal{L}_E describing the actual noise. In this sense, the promise of 'fighting noise with noise' is most manifestly realized: Unwanted and equally uncontrollable noise processes are here corrected for by a controlled local dissipative process. We hence naturally bridge the gap between decoding and dissipative self-correcting [39, 40]. The Lindblad operators can be directly constructed from the local unitaries or stochastic maps of the update rules as follows: consider a continuous-time classical stochastic process in which the unitary O is applied to a quantum system at random times, uniformly distributed at a constant rate $\lambda > 0$, such that the number of events in time interval $(t, t + \tau]$ follows a Poisson distribution with associated parameter $\lambda\tau$. This classical stochastic process is on average on the level of the quantum system reflected by a quantum Markov process.

Similarly, the purely classical part can be embedded in a quantum system, the involved quantum states being diagonal. In this picture, there is a quantum mechanical equivalent $T_{\mathbf{x}}$ of an update rule $f_{\mathbf{x}}$ at a site \mathbf{x} acting as

$$\rho \mapsto T_{\mathbf{x}} \rho T_{\mathbf{x}}^\dagger \quad (8)$$

on quantum states ρ . Again, if the rule $f_{\mathbf{x}}$ is applied in a continuous asynchronous fashion at rate $\lambda > 0$, this is reflected by the quantum dynamical semi-group of the above form with Lindblad operators $L_{\mathbf{x}} = T_{\mathbf{x}}$. The overall asynchronous cellular automaton can be embedded in a tri-partite quantum system with Hilbert space $\mathcal{H} \otimes \mathcal{L} \otimes \mathcal{V}$, with Lindblad operators $\{L_j\}$ acting locally and at two tensor factors at the time. The actual physical noise process is acting only on the first tensor factor \mathcal{H} associated with the physical quantum degrees of freedom. For a practical local dissipative decoder that does not have a threshold, see the interesting recent proposal in [7].

4.3. System size dependencies

The ϕ -automaton decoder is characterized by the simplicity of its update rules and is particularly amenable to a representation as an asynchronous CA. However, there are some mild system size dependencies hidden in the construction that need to be accounted for. The first obvious one is the necessity to run the field updates for a number of rounds of order L for every anyon move. Translated to the dissipative setting, this corresponds to the rates γ_f scaling as L . In physical terms, this reflects the necessity of the field update operations to be increasingly fast with the size of the system. In practice, this is not much of an issue, as the field updates correspond to purely classical processing.

Secondly, in the rule for anyon displacement, we need the field to have a resolution growing with the system size. If the field is encoded digitally, this corresponds to a local dimension of the CA lattice moderately scaling as $\log L$. Finally, the anyon updates do not depend on the magnitude of the field, but rather move with unit probability, whenever there is a non-zero field gradient. This feature can also be adapted to the dissipative picture with a logarithmic overhead by digitally simulating a step function at the level of the jump operators. Thus, the CA decoder can naturally be represented as a local dissipative process, at the cost of merely logarithmically unbounded rates. These additional polylogarithmic overheads may be acceptable in many reasonable and realistic physical set-ups, though conceptually speaking, ultimately our work does not yet deliver the widely sought after constant overhead dynamical decoder.

4.4. Dynamical decoders using directed signaling

A dynamical CA decoder analyzed by Harrington avoids the required logarithmic speedup of our 3D decoder [20, 41]. The basic working principle is analogous to traditional MWPM heuristics [42]. The subsequent fusion of anyons over larger and larger distances is realized via $O(\log L)$ hardware layers with each layer being responsible for a certain class of anyon distances. Most of the cells implement directed signaling between a few cells that coordinate the correction process. This approach requires inhomogeneous update rules, a certain structure inside the automaton and synchronous operation. The effective time scales are chosen such that the correction processes do not interfere between the layers. [20] provides numerics for three lattice sizes that suggest a threshold between 0.01% and 0.1%. [31] finds similar numbers for the crossings.

It is worth noting that neither our cellular automaton decoders, nor the directed signaling ones in [20, 31] can conclusively claim numerical evidence for a fault-tolerant threshold. However, both have very similar behavior, with crossings shifting to lower values of p as the system size increases, but providing very significant increases in survival time as a function of system size. The directed signaling decoder in [20, 31] are claimed to provably have a threshold for a steep hierarchical structure. We take this as encouraging evidence that our decoder will have a fault-tolerant threshold for low enough p .

4.5. Existence of constant overhead decoders

The fundamental challenge in building topological quantum memories is formulated in the established conjecture that self-correcting quantum memories only exist in four or higher dimensions [43–46]. This leads to the rule of thumb that 1D classical and 2D quantum memories share fundamental features and equally, 2D classical and 4D quantum memories. Classical memories in 2D can exhibit a thermal phase transition, as, e.g., realized by a 2D Ising model (for the connection between thermal phase transitions and quantum error correction, see, e.g., [47]). Again in 2D, dynamical decoders for classical memories with constant overhead exist, as realized by Toom's rule, based on a two-dimensional binary cellular automaton that is capable of calculating the majority in the system locally. The same applies to the 4D toric code [2] for quantum error correction, where the idea of using dynamical decoding for dissipative self correcting has already been studied [39]. While there are no thermal phase transitions in 1D for classical Hamiltonian systems, there do exist ordered phases in one-dimensional interacting particle systems. This is constructively proven by the seminal work of Gács [36–38]. In the light of dynamical decoders, Gács' cellular automaton can be interpreted as an asynchronous dynamical decoder that removes string-like excitations from classical 1D systems using only constant overhead. Harrington has speculated that this ought to generalize to a two-dimensional quantum code [20]. However, even in 1D, the Gács algorithm is tremendously complex with no numerical or experimental realization known to the authors. Analytic work estimates the Gács 1D threshold at $\sim 2^{-1000}$, so presently the outlook is poor for such an approach to yield a quantum decoder of practical relevance.

5. Discussion and outlook

In this work we have introduced a class of dynamical decoders that are suited for operating on a large number of physical and logical qubits. Our proposal of a 3D ϕ -automaton as an asynchronous dynamical decoder suggests that very simple algorithms with modest overhead can exhibit significant error suppression, even in the presence

of measurement errors. Also, we have shown how new decoding algorithms provide a blueprint for engineering dissipative self-correcting memories. The arguments given in [6] are likely to rule out the possibility to achieve robustness in quantum memories by introducing additional terms in the Hamiltonian.

Therefore, it seems imperative to circumvent no-go theorems for Hamiltonian self-correcting memories by either considering codes not covered by such theorems [46]; or to engineer dissipation that renders the code in effect self correcting. This constitutes an exciting perspective: self-correcting quantum memories in 3D may well exist, if one lessens the requirement that the quantum memories have to be of Hamiltonian nature and allows for Liouvillian dynamics. Our 3D ϕ -automaton decoder may not yet quite provide a final answer to those questions, since the corresponding Lindblad operators contain terms having a mild L dependence scaling as $\log^2 L$. In general the design of dynamical decoders seems to inherit a trade-off between the degree of complexity in the local rules and the scaling of the local resources.

It provides a clear perspective, however, how such dissipative decoders can be designed. The features of complete parallelization and asynchronous operation are not only relevant for the implementation also address the problem of wiring for on-chip architectures [30], and naturally incorporates asynchronous syndrome measurements, attractive for photonic chip based processors. Markovian dynamical decoders are therefore particularly interesting playground for fault tolerance while providing intrinsic parallelization.

Acknowledgments

We thank Ben Brown and Barbara Terhal for helpful discussions and for carefully reading a draft. JE is supported by the DFG (CRC 183), for which this work is specifically relevant for project B2, and (EI 519/7-1), the ERC (TAQ), EU (AQuS), and the BMBF (Q.com). ETC is supported by the EPSRC (grant EP/M024261/1). MJK is supported by the Carlsbergfond and the Villum fond.

References

- [1] Kitaev A Y 1997 *Quantum Communication, Computing, and Measurement* (Berlin: Springer) pp 181–8
- [2] Dennis E, Kitaev A, Landahl A and Preskill J 2002 *J. Math. Phys.* **43** 4452
- [3] Nussinov Z and Ortiz G 2008 *Phys. Rev. B* **77** 064302
- [4] Hamma A, Castelnovo C and Chamon C 2009 *Phys. Rev. B* **79** 245122
- [5] Alicki R, Fannes M and Horodecki M 2009 *J. Phys. A: Math. Theor.* **42** 065303
- [6] Landon-Cardinal O, Yoshida B, Poulin D and Preskill J 2015 *Phys. Rev. A* **91** 032303
- [7] Bardyn C-E and Karzig T 2016 *Phys. Rev. B* **94** 094303
- [8] Taminiu T H, Cramer J, van der Sar T, Dobrovitski V V and Hanson R 2014 *Nat. Nanotechnol.* **9** 171
- [9] Waldherr G et al 2014 *Nature* **506** 204
- [10] Sun L et al 2014 *Nature* **511** 444
- [11] Kelly J et al 2015 *Nature* **519** 66
- [12] Terhal B 2015 *Rev. Mod. Phys.* **87** 307
- [13] Wang C, Harrington J and Preskill J 2003 *Ann. Phys.* **303** 31
- [14] Fowler A G, Whiteside A C and Hollenberg L C L 2012 *Phys. Rev. A* **86** 042313
- [15] Duclos-Cianci G and Poulin D 2010 *Phys. Rev. Lett.* **104** 050504
- [16] Bravyi S and Haah J 2013 *Phys. Rev. Lett.* **111** 200501
- [17] Wootton J R and Loss D 2012 *Phys. Rev. Lett.* **109** 160503
- [18] Hutter A, Wootton J R and Loss D 2014 *Phys. Rev. A* **89** 022326
- [19] Herold M, Campbell E T, Eisert J and Kastoryano M J 2015 *NPJ Quantum Inf.* **1** 15010
- [20] Harrington J W 2004 Analysis of quantum error-correcting codes: symplectic lattice codes and toric codes *PhD Thesis* California Institute of Technology
- [21] Terhal B 2016 private communication
- [22] Hutter A, Loss D and Wootton J R 2015 *New J. Phys.* **17** 035017
- [23] Dauphinais G and Poulin D 2016 arXiv: 1607.02159
- [24] Cabrillo C, Cirac J I, Garcia-Fernandez P and Zoller P 1999 *Phys. Rev. A* **59** 1025
- [25] Lim Y L, Beige A and Kwek L C 2005 *Phys. Rev. Lett.* **95** 030505
- [26] Barrett S D and Kok P 2005 *Phys. Rev. A* **71** 060310
- [27] Campbell E T and Benjamin S C 2008 *Phys. Rev. Lett.* **101** 130502
- [28] Nickerson N H, Li Y and Benjamin S C 2013 *Nat. Commun.* **4** 1756
- [29] Terhal B M, Hassler F and DiVincenzo D P 2012 *Phys. Rev. Lett.* **108** 260504
- [30] Landau L A, Plugge S, Sela E, Altland A, Albrecht S M and Egger R 2016 *Phys. Rev. Lett.* **116** 050501
- [31] Breuckmann N P, Michels D, Duivenvoorden K and Terhal B M 2016 Local decoders for the 2D and 4D toric code *Quantum Inf. Comput.* **17** 0181
- [32] Pedrocchi F L, Chesi S and Loss D 2011 *Phys. Rev. B* **83** 115415
- [33] Pedrocchi F, Hutter A, Wootton J and Loss D 2013 *Phys. Rev. A* **88** 062313
- [34] Bombín H 2015 *Phys. Rev. X* **5** 031043
- [35] Brown B J, Nickerson N H and Browne D E 2016 *Nat. Commun.* **7** 12302
- [36] Gács P 1986 *J. Comput. Syst. Sci.* **32** 15
- [37] Gács P 2001 *J. Stat. Phys.* **103** 45
- [38] Gray L F 2001 *J. Stat. Phys.* **103** 1

- [39] Pastawski F, Clemente L and Cirac J I 2011 *Phys. Rev. A* **83** 012304
- [40] Kastoryano M J, Wolf M M and Eisert J 2013 *Phys. Rev. Lett.* **110** 110501
- [41] Michnicki K 2015 Towards self-correcting quantum memories *PhD Thesis* University of Washington
- [42] Avis D 1983 *Networks* **13** 475
- [43] Yoshida B 2011 *Ann. Phys.* **326** 2566
- [44] Pastawski F, Kay A, Schuch N and Cirac I 2010 *Quantum Inf. Comput.* **10** 580
- [45] Haah J 2013 *Commun. Math. Phys.* **324** 351
- [46] Brell C G 2016 *New J. Phys.* **18** 013050
- [47] Lidar D A and Brun T A 2013 *Quantum Error Correction* (Cambridge: Cambridge University Press)