Percolation in quantum computation and communication

K. Kieling[1][2][3] and J. Eisert[1][2]

[1] QOLS, Blackett Laboratory, Imperial College London, Prince Consort Road, London SW7 2BW, UK

[2] Institute for Mathematical Sciences, Imperial College London, Prince's Gate, London SW7 2PG, UK

[3] konrad.kieling05@imperial.ac.uk

1 Introduction

1.1 Percolation theory

Percolation theory is concerned with the behavior of connected clusters in a random graph. Originally developed to mathematically describe random media, it has over the years matured to a field of research in its own right. At its core is the notion of a percolation threshold: Say, on a cubic two-dimensional lattice $\mathbb{Z}^2$ neighboring vertices are connected by an edge with probability p, the distributions being identical for all edges and independent. We say then that this edge is "open". Clearly, when $p = 0$, all vertices are unconnected to each other. If we now increase the probability p starting from zero, more and more vertices will be connected. For small values of p, this will typically lead to isolated, non-crossing, small clusters of connected vertices. With increasing probability, the size of the clusters will also increase. The key observation now is that there exists a *critical percolation threshold* $p_2^{(\mathrm{c})}$ that marks the arrival of an infinite crossing cluster. For $p < p_2^{(\mathrm{c})}$, all open clusters are finite, whereas for $p > p_2^{(\mathrm{c})}$ there exists almost certainly an open infinite cluster. This behavior of having a connected open crossing cluster in a finite subset of $\mathbb{Z}^2$ can also be impressively documented by simple numerical simulations. Since the early work on fluid flow in a porous stone giving rise to the first simple percolation model [21], percolation theory has found numerous applications in a quite impressive portfolio of diverse fields, besides being a mathematical discipline in its own right [16]: This includes the study of disordered many body systems in classical and quantum physics, of instances of fluid dynamics and fire propagation, of reliability of processes, or of many aspects of statistical mechanics.

When it comes to applying percolation theory to physical systems, this article forms an exception in this book. This chapter is concerned with the *connection between percolation theory on the one hand and quantum information processing on* *the other hand*. We are facing a percolation-type situation somewhat similar to the one encountered in the study of disordered systems.

- However, in the context of quantum information processing, the *randomness* we encounter is not the one of classical probabilistic parameters in a model, but rather originates from the statistical character of quantum theory under measurements as such.
- Also, it is not the aim to model physical systems in their unknown properties, but rather think of *engineered quantum systems* that have been prepared in a very specific state for a very specific purpose. In this sense, the emphasis is less on *a posteriori* description of properties of systems or materials that are already there. But instead, to explore what state preparation and what communication protocols are possible, when limited to probabilistic processes in quantum information applications.

This chapter forms an instance of a "review" on the link between percolation theory and quantum information theory, as a topic that is presently receiving notable attention. We will see that this link is a quite natural one, where percolation in several ramifications enters as a concept to overcome limitations imposed by probabilistic processes that occur in measurement processes. Yet, whereas this link has been mentioned a number of times in the literature, so far, only four research articles flesh out this link in detail [28, 1, 34, 9]. This article can hence be seen as (i) a short review on the material that is known and published at present, as (ii) a presentation of a long and more detailed version of the short proof presented by the same authors in Ref. [28], and, most importantly, as an (iii) invitation to the subject. Readers familiar with elements of quantum information theory may wish to skip to Section 2.

1.2 Quantum computation and communication

Quantum information processing takes the idea seriously that when storing or processing information, it matters whether the underlying physical system follows classical or quantum laws. In classical information theory, one is used to the fact that it hardly makes sense to think of the physical carrier of information, as one can transform the information stored in one form to another carrier in a lossless fashion anyway. This abstraction from the physical carrier is challenged when one thinks of *single quantum systems* forming the elementary processing units. Indeed, the task of transforming the "information stored in a quantum system" into classical information and back is impossible. Quantum information processing is however not so much concerned with limitations due to quantum effects, but rather thinks of new applications in computing and communication when the carriers of information are single quantum systems.

A *quantum computer* [32, 14] is such an envisioned device: One thinks of having an array of quantum systems – spins, say, referred to as *qubits*. This system, associated with a Hilbert space $\mathcal{H} = (\mathbb{C}^2)^{\otimes n}$, is initially prepared in a known, preferably pure quantum state described by a state vector $|\psi\rangle$. Then one manipulates the state by means of unitary time evolution or by means of measurements. Acknowledging that not any unitary dynamics is accessible on a quantum many-body system, in the *circuit model*, this step of computation is broken down to *quantum gates* [32, 14]. One hence implements a sequence of unitary gates that have trivial support on all sites except single sites – giving rise to *single qubit gates* – and pairs of sites – *two-qubit gates*. The state vector after the unitary time evolution is then

$$|\psi\rangle = \prod_j U_j |\psi\rangle. \tag{1}$$

This is followed by local individual measurements on the spins. The measurement outcomes at the end then deliver (typically statistical) data, from which the outcome of the computation can be estimated. The important point is that because the quantum system follow the laws of quantum mechanics and can be prepared in a superposition exploring an exponentially large Hilbert space, some problems could be solved with significantly less effort on this envisioned device than on any classical computer. In fact, they could be solved with polynomial effort in case of some problems like factoring that are believed to be computationally hard classically.

Yet, the circuit model is only one out of several models for quantum computation. In the computational model having center stage in this chapter, one abandons the need for unitary control via quantum gates, and performs only local measurements on an entangled quantum state of many constituents. This is the model of the *one-way computer* [37, 8, 23]. The entanglement present in the initial resource then facilitates the efficient simulation of any other computation that had unitary gates available. Needless to say, in any such approach, the core question is how to actually prepare this initial resource state, an instance of a *cluster* or *graph state* [24, 23]. It is in this preparation – and in the way to overcome the intrinsically probabilistic character of quantum measurements – that percolation ideas will be important in the first part of this chapter. It will turn out that by making use of percolation ideas, some of the key obstacles in some physical architectures to realize quantum computation can be weakened. In particular, in optical approaches, we will see that percolation ideas facilitate the preparation of such states with little dependence of preparation steps on earlier measurement outcomes – an experimentally very difficult prescription in this context. Percolation will hence help us to think of what is known as linear optical quantum computation [29, 47, 28, 9].

The second part of this chapter is dedicated to a related but different question. This is taken from *quantum communication* and *key distribution*: Several applications in quantum information processing, most importantly, quantum key distribution for the secure transmission of classical data, but also instances of the envisioned distributed quantum computing, rely on the availability of almost maximally entangled states shared between spatially separated laboratories. If one has several such stations on the way, each communicating and exchanging quantum particles with a number of neighbors, one arrives at the idea of a *quantum network*. In Section 3 we will review the use of percolation ideas to establish maximally entangled pairs for quantum communication between arbitrary stations, based on the published work Ref. [1, 34]. Here, the main point will be that not only classical percolation ideas can be employed to meet the aim of sharing entanglement between arbitrary vertices of a graph: Instead, appropriate quantum measurements allow to outperform the situation of classical edge percolation. In this sense, one encounters a notion of *entanglement percolation*. In fact, such entanglement percolation can drastically outperform strategies based on standard edge percolation.

2 Percolation and quantum computing

2.1 General concepts

We now turn to the preparation of cluster or more generally graph states as resource state for measurement-based quantum computation. This measurement-based model is a desirable route to quantum computation in a number of architectures. This is not the least due to the very clear distinction between the preparation of entanglement, and the consumption thereof by means of measurements, abandoning the need for unitary gates. Percolation ideas play the crucial role here when thinking of errors, of dealing with probabilistic gates building up the states and, most importantly, of coping with the daunting feed-forwards (dependence of later action on earlier measurement outcomes).

It will be necessary – before we establish the percolation argument – to introduce a number of concepts that will be used later on. A *graph state* [41, 24, 43] is a quantum state described by an undirected graph $G = (V, E)$, with V being the set of vertices, and E the set of edges. The vertices are embodied by physical systems, so single spins or qubits. The edges represent interactions. More specifically, a *graph state* is the simultaneous eigenstate to the set of *stabilizers*

$$K_G^{(a)} = \sigma_x^{(a)} \bigotimes_{b:\mathrm{dist}(a,b)=1} \sigma_z^{(b)} \tag{2}$$

for all $a \in V$ with eigenvalue 1, i.e.,

$$K_G^{(a)} |\psi_G\rangle = |\psi_G\rangle \qquad \forall a \in V. \tag{3}$$

Here, dist(x, y) is the graph-theoretical distance [5] between vertices x and y on G, i.e., dist$(x, y) = 1$ for neighboring vertices. Here and in the following, $\sigma_{x,y,z}^{(b)}$ denotes a Pauli operator with support on the Hilbert space of the physical system labeled b.

Equivalently, this state can be thought of as having each qubit prepared in the state vector $|+\rangle = (|0\rangle + |1\rangle)/\sqrt{2}$ and applying an interaction leading to a *phase gate* or *controlled-Z*-gate

$$U_z = |0,0\rangle\langle 0,0| + |0,1\rangle\langle 0,1| + |1,0\rangle\langle 1,0| - |1,1\rangle\langle 1,1| \tag{4}$$

to neighboring vertices $a, b \in G$, so to vertices that are connected by an edge $(a, b) \in E$. Because these gates are diagonal in the computational basis, they commute, and hence the order in which they are applied does not influence the final state.

Such a graph state may be defined for any graph. Most important for our purposes are lattices where $V \subset \mathbb{Z}^2$, so finite cubic lattices in two dimensions. Such a graph state of a cubic lattice is called a *cluster state* [8]. It has been shown that local measurements on single constituents of such a cluster state are just as powerful as the gate model for quantum computation, and it can hence efficiently simulate any other quantum computer [37, 38]. That is, for a given finite computation there exists a finite graph with $V \subset \mathbb{Z}^2$ such that the computation can be simulated by a sequence of suitable single-qubit local measurements on the cluster state. The usual mapping from a circuit model to a cluster state allows to label one dimension as time which comes in handy because only time slices of a cluster actually have to exist at a given point during the computation. We will also frequently encounter finite subsets of the hexagonal lattice which is a computation resource in the same sense. Note that *a priori* the lattice does not correspond to any spatial lattice – what determines the quantum state are solely the adjacencies on the graph.

When preparing cluster states, there are two main types of errors where percolation comes into play:

- One can think of the above lattice structure emerging from a *physical lattice*. This may be an *optical lattice* generated by standing wave laser light, where atoms are located at individual lattice sites [31].[4] By means of interactions or controlled collisions, implementing (2) or (4), respectively, a cluster state can in principle be prepared in such systems. Yet, Mott hole defects, where sites are left unoccupied in a random fashion, lead to defects. This is a situation where ideas of *site percolation* can overcome the problem of having non-unit filling factors.
- Using other types of quantum systems, cluster states have to be prepared sequentially, by means of explicitly employing gates to pairs of constituents. This should not be confused with the absence of gates when doing the actual computation via measurements – the gates used during state preparation are independent of the actual algorithm to perform. The most resource-efficient and hence most feasible ideas of doing quantum computation with *linear optical systems* or with *atoms in optical cavities* (see, e.g., Ref. [30] and references therein) rely on exactly such a sequential build-up of the cluster state. Here, one starts from elementary building blocks, like entangled photon pairs. From these building blocks the full cluster or graph state can be prepared by means of a sequential application of controlled-Z gates. During the course of this process, in principle gates between any pair of qubits are allowed, which one is actually applied at a given step depends on a chosen prescription which, in turn, depends on outcomes of earlier gate applications. The lattice structure is imposed by the sequence of gates, so every state described by an undirected graph without degenerate edges can be generated.
  This being the most promising approach to think of optical quantum computing, it seems important to address two major challenges that have to be overcome in such an approach:[5].

[4] We leave issues that are under significant consideration in the literature such as the question of locally addressing single sites in measurements aside and focus on the preparation of the cluster state.

[5] Again, we will focus on the intrinsic problems of such an idea. From a physical perspective, the development of heralded sources with high efficiency and mode quality, as well as

1. The applied gates necessarily operate in a probabilistic fashion.[6] Indeed, the probability of success of such gates is typically quite small [13, 40]. Hence, when sequentially preparing a cluster state, needless to say, the very order in which the states are sent through the entangling gates plays a crucial role [26, 20, 27]. Especially the success probability of the fusion gates, $p_{\mathrm{success}} = 1/2$, will be of central interest as this is already the optimal probability for such a gate in linear optics [11, 20]. A naive approach of trying to grow an $n \times n$ cluster state by using such gates will fail. However, this obstacle can in principle be overcome: One can show, using combinatorical methods and ideas of convex optimization, that there exist methods to achieve an optimal scaling of $O(n^2)$ invocations of quantum gates (and number of qubits), despite the gates operating probabilistically [20].
2. Unfortunately, this – and actually any such – procedure leads to a great deal of conditional dynamics: Depending on outcomes of earlier fusion operations, one has to decide which pair of states will be used in the next step. This, however, turns out to be a significant challenge in actual experimental realizations, as it requires active switching including coherent interaction between any pair of modes in the setup, while unused modes are stored in quantum memories, thus clearly rendering it a very difficult prescription.

To lessen this daunting requirement, percolation ideas can come into play: One could think of a static setup, i.e., an underlying structure is imposed onto the resources and only applications of entangling gates between nearest neighbors are allowed. The probabilistic nature of quantum gates is the source of randomness, we now face a problem of *edge percolation*. How exactly this will work and how to find bounds to the scaling of the resource requirements to prepare two-dimensional cluster states even if the probability of an edge being open is smaller than the critical percolation threshold for a two-dimensional cubic lattice $p < p_2^{(\mathrm{c})}$, will be the subject of the next section [28]. The latter property – which is responsible for some technicalities – is important, as the probability of success of the fusion gate happens to be $p_{\mathrm{success}} = 1/2$, which is identical with the critical percolation threshold $p_2^{(\mathrm{c})} = 1/2$. Hence, ways have to be found to achieve percolation even using such gates.

Further developing these ideas, one can ask whether there is a *phase transition* in edge percolation with respect to quantum computing applications: One thinks of a cubic lattice as in 2., with edges being open in case a gate was successful. If $p > p_2^{(\mathrm{c})}$, one can extract a resource almost certainly that allows for universal quantum com-

detectors with high detector efficiency are major obstacles, a significant research effort is dedicated to.

[6] This is a consequence of gates being in turn implemented by means of measurements. All non-linearities in *fully linear optical systems* have to be effectively realized by means of measurements, and this randomness is hence the intrinsic randomness in measurements in quantum mechanics. Gates like the *fusion gates* [35, 10] act in effect as the desired controlled-Z gate, albeit in a probabilistic fashion. In approaches based on *atoms in optical cavities, coupled via light* similar issues occur.

puting. Below the threshold $p < p_2^{(\mathrm{c})}$, one can show that one can almost certainly simulate the evolution of the quantum system on a classical computer. Hence, this regime is not only not useful for quantum computation, but even classically efficiently tractable.[7] In this sense, there is a phase transition in the computational potency of the resource depending on the probability of an edge being open.

In these different flavors we will encounter percolation ideas in the context of quantum computing. To be brief, we concentrate on the latter two aspects, putting an emphasis on bond percolation (so all p will be bond probabilities unless stated otherwise). We note, however, that the problem of site percolation can be treated similarly.

2.2 Resource state preparation in measurement-based computing

Renormalization

The goal of this section will be to show how to generate a cluster state of a given size with probabilistic entangling gates (succeeding with probability p) almost certainly with the help of bond percolation. The lattice will be divided into blocks which will be reduced later to single qubits, thus "renormalizing" the lattice. Whenever a block contains a crossing open cluster that connects the block's four faces (in the first two dimensions) we will refer of this crossing cluster as a renormalized qubit. If the crossing clusters of two neighboring renormalized qubits acually touch each other (i.e., there exists an open path between two vertices of the first and the second renormalized qubits that lies completely within the union of the two respective blocks, and they are connected by open bonds), then the reduction of the blocks to single qubits will yield a "renormalized bond" between these qubits. How this reduction actually works is the scope of the following sections. Now, we are looking for the probability $P_p(\mathfrak{U}(L,k))$ of the event $\mathfrak{U}(L,k)$ to occur. $\mathfrak{U}(L,k)$ denotes the event that the renormalized square lattice of size $L \times L$ with hypercubic blocks of size $k^{\times d}$ is fully occupied and connected. Given a dependence of the block on the lattice size, $k(L)$, we will use the abbreviation $P_p(L) = P_p(\mathfrak{U}(L,k(L)))$.

The result is more precisely phrased as follows:

Theorem 1 (Resource consumption). *Let $p > p_d^{(\mathrm{c})}$, $d \geq 2$ being the dimension of a hypercubic lattice. Then for any $\mu > 0$, the probability $P_p(L)$ of having an $L \times L$ renormalized cubic lattice fulfills*

$$\lim_{L \to \infty} P_p(L) = 1, \tag{5}$$

with an overall resource consumption of $R(L) = O(L^{2+\mu})$.

[7] A similar situation has been observed in Ref. [19], where a *Kitaev's toric code state* is not a universal resource, but can be classically efficiently simulated [7]. If one modifies the state very little by means of local phases, one can not only not keep track of measurements classically in an efficient fashion. But in fact, it can be shown that the state would serve as a universal resource for quantum computation [18, 19].

Here, $p_d^{(\mathrm{c})}$ denotes the critical percolation threshold of the d-dimensional hypercubic lattice. R is used to refer to the number of initial resources, so the constituents that are placed on each lattice site with the ability of "growing" connections to their neighbors in a probabilistic manner. Note, that the dimension d can always be chosen such that the gates at hand ($p > 0$) operate in a regime that allows for percolation ($p > p_d^{(\mathrm{c})}$).

Proof. ($d \geq 3$) In the case of dimensions $d > 2$, so different from $d = 2$, crossing paths in different directions not necessarily intersect. This approach is often nevertheless favorable due to the higher critical percolation threshold in higher dimensions. Let us fix $L \in \mathbb{N}$ and take

$$U = [1, 2kL]^{\times 2} \times [1, 2k]^{\times d-2} \subset \mathbb{Z}^d \tag{6}$$

for some $k \in \mathbb{N}$. This slab can be divided into L^2 disjoint hypercubes with an edge length of $2k$. With $A_y(k)$, $y = (y_1, y_2) \in [2, 2L]^{\times 2}$ we denote the $(2k)^{\times d}$ hypercube starting at $(y_1 k, y_2 k, 1, \ldots, 1)$. For $y = 2x$,

$$x \in M = [1, L]^{\times 2}, \tag{7}$$

these hypercubes $A_y(k)$ are the disjoint blocks, and M plays the role of the renormalized square lattice.

Furthermore, we will use the overlap between adjacent blocks in the first direction,

$$B_y(k) = A_y(k) \cap A_{(y_1+1, y_2)}(k) \tag{8}$$

for $y_1 = 2, \ldots, 2L - 1$, and the union of disjoint neighboring blocks in the second direction, $C_z(k) = A_z(k) \cap A_{(z_1, z_2+2)}(k)$, $z_2 = 2, 4, \ldots, 2(L-1)$.

On these blocks we will define a series of *events* as follows:

- $\mathfrak{A}_y(k)$: There exists an open left-to-right crossing cluster in $A_y(k)$ in the first dimension, so an open path containing open vertices a and b with $a_1 = y_1 k + 1$, $b_1 = (y_1 + 2)k$. For $p > p_d^{(\mathrm{c})}$ there exists a constant $g > 0$, only dependent on p, such that [16]
$$P_p(\mathfrak{A}_y(k)) \geq 1 - \exp(-gk^2). \tag{9}$$
- $\mathfrak{B}_y(k)$: The number of open left-to-right crossing clusters in $B_y(k)$ (see Fig. 1) does not exceed 1. It is shown in Ref. [2] that for $p > p_d^{(\mathrm{c})}$ there exist constants $a, c > 0$, only dependent on p, such that the probability of $\mathfrak{B}_y(k)$ occuring satisfies
$$P_p(\mathfrak{B}_y(k)) \geq 1 - (2k)^{2d} a \exp(-ck). \tag{10}$$
- $\mathfrak{D}_y(k)$: We can now make use of the two events defined before. Let $\mathfrak{D}_z(k)$, $z_1 = 3, 5, \ldots, 2L - 1$, $z_2 = 2, 4, \ldots, 2L$, be the event that there exists an open left-to-right crossing cluster in both blocks, $(z_1 - 1, z_2)$ and $(z_1 + 1, z_2)$, and these two clusters are actually connected (see Fig. 2). That means, there exists an open left-to-right crossing cluster in $A_{(z_1-1,z_2)}(k) \cup A_{(z_1+1,z_2)}$. The events $\mathfrak{A}_y(k)$ and $\mathfrak{B}_y(k)$ are *increasing events*[8]. That allows for application of the

[8] Let $\chi(\mathfrak{A}_p)$ denote the characteristic function of the event $\mathfrak{A}$ for an elementary probability p. An increasing event $\mathfrak{A}$ is one that satisfies $|\chi(\mathfrak{A}_p)| \leq |\chi(\mathfrak{A}_{p'})|$ if $p \leq p'$.

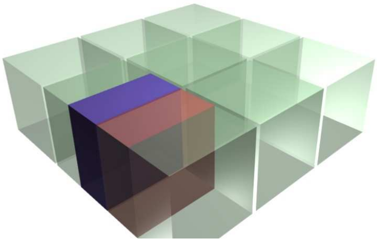

Fig. 1. The blocks A_y for $y = 2x$. The red and blue regions together constitute the block $A_{(3,2)}$, and the blue region is an overlap with a neighboring block, so $B_{(3,2)}$.

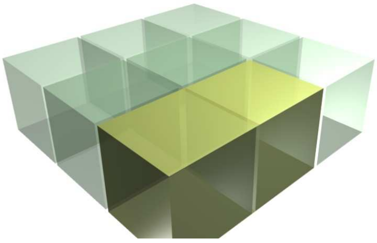

Fig. 2. Again, the blocks A_y for $y = 2x$ are shown and the yellow region is the intersection $C_{(2,2)}$.

Lemma 1 (FKG inequality). *Let $\mathfrak{X}$ and $\mathfrak{Y}$ be increasing events. Then*

$$P_p(\mathfrak{X} \cap \mathfrak{Y}) \geq P_p(\mathfrak{X}) P(\mathfrak{Y}). \tag{11}$$

This results in an upper bound for $P_p(\mathfrak{D}_y(k))$, with

$$\mathfrak{D}_y(k) = \bigcap_{a=0,\pm 1} \mathfrak{A}_{(y_1+a,y_2)}(k) \cap \mathfrak{B}_{(y_1-1,y_2)}(k) \cap \mathfrak{B}_{(y_1,y_2)}. \tag{12}$$

- $\mathfrak{E}_z(k)$: The next event we need is the one that "connects" two blocks in the second dimension. Let $\mathfrak{E}_z(k)$ be the event that there exist at most one open left-to-right crossing cluster in $C_z(k)$. In order to apply the arguments of Ref. [2], we extend the blocks in the last $d-2$ dimensions by another $2k$. The probability of $\mathfrak{E}_z(k)$ occuring is bounded by

$$P_p(\mathfrak{E}_z(k)) \geq 1 - (4k)^{2d} a \exp(-2ck). \tag{13}$$

- The last event, $\mathfrak{F}_z(k)$, we will define here is that of having an open left-to-right crossing clusters in both, $A_{(z_1,z_2)}(k)$ and $A_{(z_1,z_2+2)}(k)$, but at most one in $A_{(z_1,z_2)}(k) \cup A_{(z_1,z_2+2)}(k)$. That means, again, that there is actually one left-to-right crossing cluster in $C_z(k)$, but it connects the left and right faces of $A_{(z_1,z_2)}(k)$ and $A_{(z_1,z_2+2)}(k)$ simultaneously. Again, by using the FKG inequality we can construct an upper bound to the probability of occurrence of $\mathfrak{F}_z(k) = \mathfrak{A}_{(z_1,z_2)}(k) \cap \mathfrak{A}_{(z_1,z_2+2)}(k) \cap \mathfrak{E}_z(k)$.

Having all these events at our disposal, the goal of realizing a fully renormalized lattice can be formulated quite easily: we are looking for a simultaneous occurence of $\mathfrak{D}_z(k)$ and $\mathfrak{F}_z(k)$ for a suitable set of $z = (z_1, z_2)$'s, so

$$\mathfrak{U}(L,k) = \left(\bigcap_{\substack{z_1=3,5,\ldots,2L-1 \\ z_2=2,4,\ldots,2L}} \mathfrak{D}_z(k) \right) \cap \left(\bigcap_{\substack{z_1=2,4,\ldots,2L \\ z_2=2,4,\ldots,2(L-1)}} \mathfrak{F}_z(k) \right). \tag{14}$$

By subsequent application of the FKG inequality we can express an upper bound to the probability of $\mathfrak{U}(L,k)$ occuring in terms of the probabilities of the former events:

$$\begin{aligned}
P_p(\mathfrak{U}(L,k)) \geq & \prod_{\substack{y_1=2,3,\ldots,2L\\ y_2=2,4,\ldots,2L}} P_p(\mathfrak{A}_y(k)) \prod_{\substack{y_1=2,3,\ldots,2L-1\\ y_2=2,4,\ldots,2L}} P_p(\mathfrak{B}_y(k)) \times \\
& \prod_{\substack{y_1=2,4,\ldots,2L\\ y_2=2,4,\ldots,2(L-1)}} P_p(\mathfrak{C}_y(k)) \\
\geq & (1-\exp(-gk^2))^{2L^2-L} \times \qquad (15)\\
& ((1-(2k)^{2d} a\exp(-ck))^2(1-(4k)^{2d} a\exp(-c2k)))^{L(L-1)}.
\end{aligned}$$

Now, we will have to find the block size scaling $k(L)$ such that this probability is approaching unity for large L. Moreover, we are looking for a “good” scaling, in the sense that the overall resource scaling $L^2 k(L)^d$ does not differ too much from the optimal $O(L^2)$. In order to invert (15), so to find the best $k(L)$ consistent with this approach, we, however, still need to relax the problem to some extent. By using the slowest increasing term in (15) we can bound the expression from above: there exists an integer k_0 such that

$$P_p(\mathfrak{U}(L,k)) \geq (1-(2k)^{2d} a\exp(-ck))^{5L^2} \qquad (16)$$

for all $k \geq k_0$. Let us now use the ansatz $k = \lceil L^\varepsilon \rceil$, for some $\varepsilon > 0$.

For any $x \in \mathbb{N}$ there exists a $L_0 \in \mathbb{N}$ such that for all $L \geq L_0$

$$1 - A(2L^\varepsilon)^{2d}\exp(-cL^\varepsilon) \geq 1 - 1/(xL^2). \qquad (17)$$

Further,

$$\lim_{L\to\infty} \left(1 - 1/(xL^2)\right)^{5L^2} = \mathrm{e}^{-5/x} \qquad (18)$$

and for every $\epsilon > 0$ we can find an x such that $1 - \mathrm{e}^{-5/x} < 1 - \epsilon$.

Therefore, the chosen dependence of k on L is sufficient to achieve a success probability within a chosen ϵ around 1, getting arbitrary close in the limit of large L:

$$\lim_{L\to\infty} P_p(\mathfrak{U}(L,k(L))) = 1. \qquad (19)$$

Combining this with the number of blocks used, L^2, this induces a resource scaling of $R(L) = O(L^{2+d\varepsilon})$.

($d = 2$) In the two-dimensional case the connection between paths in the two directions is, of course, not an issue – whenever a block A_y is crossed in both directions, these two crossing paths necessarily intersect. The events needed in this case are the following:

- $\mathfrak{G}_{\mathfrak{i},\mathfrak{r}}$: The rectangle $C_{i,r} = \cup_{y,y_r=i} A_y$ is crossed in the r-th dimension. Here, $C_{i,r}$ is the i-th row or i-th column in case of $r = 1$ or $r = 2$, respectively. The probability for such an event to happen satisfies [16]

$$P_p(\mathfrak{G}_{\mathfrak{i},\mathfrak{r}}) \geq 1 - skL\exp(-tk), \tag{20}$$

with $s, t > 0$.

Again, the $\mathfrak{C}_{i,r}$ are increasing events, so the probability of simultaneous crossing in all rows and columns,

$$\mathfrak{U} = \bigcap_{\substack{i=1,\ldots,L\\ r=1,2}} \mathfrak{G}_{i,r} \tag{21}$$

satisfies

$$P_p(\mathfrak{U})(L,k) \geq (1 - skL\exp(-tk))^{2L}. \tag{22}$$

Now, the choice of $k = \lceil L^{\varepsilon} \rceil$ can be used again, together with the last steps for the case $d > 2$. Thus, in the two-dimensional case a resource scaling of $R(L) = O(L^{2+d\varepsilon})$ holds as well.

Although the proof was explicitly stated in terms of bond percolation, a reasoning along these lines will hold as well for site percolation or mixed site/bond percolation, as long as the probabilities in question are above the respective threshold.

Path identification

For the cluster state to be of any use in quantum computing, the number of resources required to simulate a given quantum circuit have to depend efficiently (i.e., polynomially) on the size of the circuit. As the size L of the cluster state required to implement a given circuit has a polynomial dependence on the circuit's size [37], Theorem 1 already provides a suitable scaling in the number of qubits required.

Still, the amount of time and classical memory required to implement a given computation has to obey a well-tempered scaling as well. The "quantum part" (i.e., the number of subsequent measurements in the preparation- and in the computing stage) only requires $O(1)$ of time for preparation of the initial pieces, a single step for all the simultaneous entangling operations, and the measurements to reduce the cluster to the renormalized one and perform the computation. Many of them can be performed in parallel, but an upper bound is given by $R(L)$.

The classical amount of memory, of course, starts with $R(L)$ to store all gate outcomes (and therefore the percolated graph). In the following, the scaling of classical resources will be analyzed [9].

Crossing clusters

For identification of the crossing clusters within the blocks, cluster finding algorithms like the Hoshen-Kopelman-algorithm [25] can be employed. Out of the box this would require $O(k^{d-1})$ of classical memory and $O(kL^2)$ timesteps. If there exist more than one crossing cluster (which is as of (13) highly improbable), only a single one (e.g., the one with the largest surface) will be chosen for the subsequent procedure.

[9] More details and a MatLab implementation of the relevant parts can be found in the supplementary material of Ref. [28] at http://www.imperial.ac.uk/quantuminformation.

Connecting the blocks

A “mid-qubit” which is a member of a crossing cluster near the center of the block will be chosen ($R(L)$ time-steps) in every block. Let us define an open path on $G = (V, E)$ between $a_1, a_{n+1} \in V$ by $\mathcal{P}(a_1, a_{n+1}) = \{(a_1, a_2), (a_2, a_3), \ldots, (a_n, a_{n+1})\}$ with $(a_i, a_i{+}1) \in E$, $i = 1, \ldots, n$, and its length by $|\mathcal{P}(a_1, a_{n+1})| = n$. Because we are using undirected graphs, $(a, b) = (b, a)$ and there is a corresponding path from B to a for each path from a to b. Open paths between the mid-qubits of all pairs of neighboring blocks are identified using a breadth-first-search (BFS) algorithm [42] ($R(L)$ time and memory complexity).

To prevent loops from being present in the paths in the first place, the following procedure is employed:

1. Using BFS on the crossing clusters starting from the mid-qubits and constrained to the respective block, each site is labeled with the length of the shortest path to the mid-qubit in this block. By going in the direction of decreasing length, the shortest path $\mathcal{P}(s, m(x))$ to a block’s x mid-qubit $m(x)$ can be found within its block, starting from any site s.
2. The facing boundaries of all pairs of neighboring blocks (x_1, x_2) are searched for the pair of sites (s_1, s_2), one in each block, with the least sum of their distances $|\mathcal{P}(s_1, m_1)| + |\mathcal{P}(s_2, m_2)|$ from their respective mid-qubits m_1 and m_2, and the bond (s_1, s_2) being open.
3. With the composite path $\mathcal{P}(m_1, s_1) \cup (s_1, s_2) \cup \mathcal{P}(s_2, m_2)$ a loop-free connection between m_1 and m_2 is found. Although paths from the mid-qubits to different neighbors might have sub-paths in common, there will be no loops inside a block due to starting always with the same site and the same algorithm for all paths to the boundaries in a given block.

Reduction to a renormalized lattice

Instead of the whole square lattice which would require to identify cross-like junctions within the blocks, the procedure will renormalize to a hexagonal lattice where only T-junctions are required. Of course it will be embedded in the square lattice geometry in the obvious way. A square lattice would involve crosses the construction of which is not clear when T-junctions have been found and local measurements are used for reduction. There is, however, an easy way to turn the whole lattice into a square one by using local measurements afterwards [44].

The procedure to cut out parts of the cluster, or yanking paths straight involves single qubit measurements of the Pauli operators σ_y and σ_z. Let us briefly summarize the effective action of these operations when applied to a qubit constituting a vertex $a \in V$ of a graph state $|\psi_G\rangle$ with $G = (V, E)$ [24]. The neighborhood of a vertex a the will be denoted by $N_a = \{b : \mathrm{dist}(a, b) = 1\}$, the subgraph of G induced by $A \subset V$ is $G[A] = (A, \{(a, b) \in E : a, b \in A\})$, and the complement of a graph G with respect to the set of all possible edges by $G^c = (V, \{(a, b) : a, b \in V\} \setminus E)$. The measurement rules now read

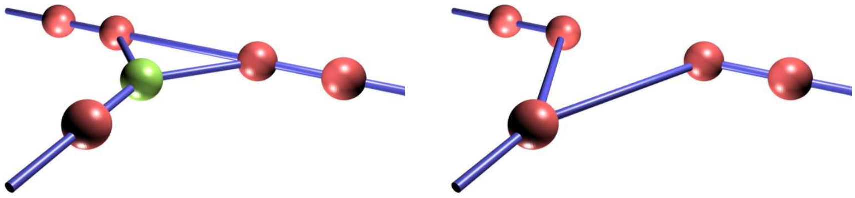

Fig. 3. Effect of a σ_y measurement on a triangular junction of a cluster. If the three arms are, as shown, not immediately connected outside the triangle, a σ_y measurement on the green qubit has the effect of destroying the triangle in favor of a T-shaped junction. Due to symmetry, any of the three qubits in the triangle could have been chosen.

- σ_y measurements perform local complementations: $G' = (G[V \setminus \{a\}] \setminus G[N_a]) \cup G[N_a]^c$. So, it swaps all possible edges in its immediate neighborhood.
- A σ_z "cuts" the vertex a. The graph representing the resulting state is $G' = G[V \setminus \{a\}]$.

The resulting graphs G' actually are only obtained after applying some local unitaries to the neighbors of a, which depend on the measurement outcomes, to the remaining qubits. However, one can also store the effective unitary to apply to each remaining qubit (that is a memory requirement of $O(1)$ per qubit) and adjust the measurement basis of the subsequent measurements. Because each qubit will be measured out by the end of the computation, it is sufficient to use the second approach to the compensation of random measurement results.

The first application is to isolate the paths and eliminate the spare sites and dangling ends towards the mid-qubits. This is achieved by cutting out all unneeded sites by measuring their qubits in the σ_z basis.

Now, one is left with a hexagonal lattice where each edge possibly consists of a long path and each site might consist of a triangular structure in the worst case (depending on the type of lattice used, also the wanted single-qubit sites are possible). The triangles can be destroyed by suitable σ_y-measurements, as shown in Fig. 3. After that, only single-qubit junctions are left, the paths between which can be shrinked by subsequent application of σ_y-measurements to a single edge.

Summing up all these contributions we realize that both – the amount of classical memory and the number of time-steps – are bounded by $R(L)$ as well.

It should be pointed out that one obstactle in one-way quantum computation is to keep the whole state in memory. Having fixed the algorithm in advance, the required state size is known and therefore also the block size for a fixed allowed error rate. Therefore, to grow individual blocks and reduce them to single qubits, only the neighboring blocks have to exist. This especially allows for growing of the

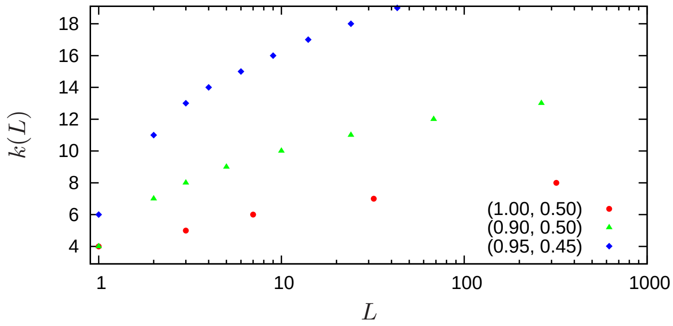

Fig. 4. Results of Monte Carlo simulations to determine the scaling behavior of the renormalization procedure on the diamond lattice. The dependence of the diamond lattice's block size $k \times k \times k$ on the size $L \times L$ of the renormalized square lattice is shown for three different sets of site- and bond probabilities $(p_\mathrm{s}, p_\mathrm{b})$. The threshold for the probability of $\mathfrak{U}$ occuring was chosen to be $1/2$. 10^5 blocks of each size were created and used to randomly populate each lattice size 10^3 times.

reduced lattice in the time direction while the computation moves on, requiring only $O(L^{1+\mu})$ qubits to be kept in memory.

Decreasing the vertex degree

Choosing the appropriate lattice

To actually utilize the protocols based on percolation theory the initial resources (i.e., the stars sitting on the lattice sites) should be as small as possible. As it is more difficult to prepare larger states (this is the problem to be solved in the first place), the lattice with the lowest vertex degree for which $p_\mathrm{success} > p^{(c)}$ is still fulfilled will be favorable.

Again, results known in percolation theory, but also specifics of the physical implementation, can be used to decrease the vertex degree of the initial states. First, using star resources, one would have to look for the lattice with the smallest vertex degree that is still suitable for the bond probabilities at hand. It is not necessary to stay in two dimensions, as the blocking procedure can use high dimensional lattices and renormalize them to two-dimensional square lattices.

For example, in the case mentioned above, $p_\mathrm{success} = 1/2$, the smallest vertex degree compatible with p is four, realized by the diamond lattice with $p_\diamond^{(c)} \approx 0.389$. That translates into five-qubit initial states. To see, that the procedure still works for lattices different from cubic, see the results of numerical simulations on the diamond lattice in Fig. 4.

Covering lattice

If the entangling operations at hand have the property that one of the qubits survives (like the probabilistic parity check "fusion" [10] in linear optics), the following property of bond-percolation can further reduce the size of the initial pieces. So far, the

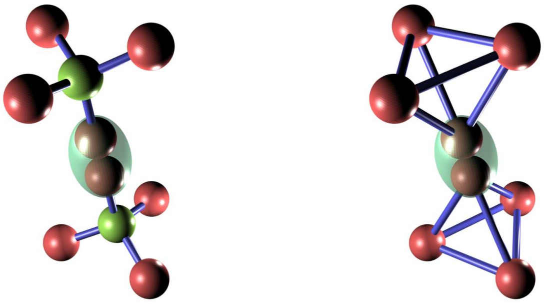

Fig. 5. Pairs of initial resources for the diamond lattice and its covering one, the pyrochlore lattice. A probabilistic entangling gate is applied between pairs of qubits of all neighboring initial states. The central qubit, here shown in green, is measured out in the σ_y-basis, resulting in the transformation from diamond to pyrochlore lattice.

sites of the lattice are occupied by single qubits, the bonds are given by edges in the underlying entanglement graph. If one qubit is left by the entangling operation we will not think of this one as being a site itself, but rather belonging to the bond between its neighboring site.

Having such a graph state (which exact lattice type does not matter), we can measure the central qubit a of the initial stars (the green one in Fig. 5) in the σ_y-basis. Given the specific structure we have at hand, this operation actually performs a transformation from the lattice type we had before to its covering lattice: now think of the qubits that were sitting on the bonds as proper sites. The old sites have disappeared (they have been measured out) and the new ones are connected to all the new sites that were in the same neighborhood of an old site (local complementation).

Intuitively, the covering lattice has the same connectivity properties as the original lattice before. Paths through a star between two arms are existent iff the star was present and the two entangling operations involving these arms were successful. The same holds on the covering lattice. This property is reflected by the equation $p^{(c)}_{G,\mathrm{b}} = p^{(c)}_{G_\mathrm{c},\mathrm{s}}$, so bond percolation on the original lattice induces site percolation on the covering one.

As the local complementation inside the stars commute with the entanglement operation between them, the central qubit might be neglected from the very start (see Fig. 5). So, one further qubit can be saved by starting with the fully connected graph state (locally equivalent to the GHZ state) that has one qubit less than the corresponding star. In case of the diamond lattice, with four-qubit GHZ states (tetrahedral states), the pyrochlore lattice can be built.

Further methods to reduce the amount of conditional dynamics in linear optical quantum computing

Surely, if the scheme requires quite large initial stars, they can be prepared with the same tools probabilistically, starting from smaller stars. For a fair assessment, however, the constraints that led to the percolation scheme in the first place have to be imposed here as well. That is, the restriction to a static setup. Whether composite stars can be used in a static layout now depends on how the entangling gates work in detail, i.e., what the failure outcomes are. That is of interest due to the fact that in general a failure in the star preparation step would require back-up steps that do not allow for subsequent application of further entangling gates without re-routing. A type of gates with suitable properties is the one that acts as a σ_z-measurement on both qubits on failure. For example, parity check based gates in linear optics offer this feature.

That this feature might actually bring some benefit is shown by the following example (see Fig. 6). Two instances of such an entangling gate are applied to a pair of five-qubit stars, one to a pair of arms and one to the central qubits. On success of one of the “arm” fusion gates, the two stars are connected by a two-edge chain. The middle qubit of this chain (green) will be measured out in the σ_x-basis, leaving the two centers merged in a redundantly encoded qubit, being the center of a star with six arms. A second application of such a gate on the two qubits of the new center will always succeed due to the entanglement already existing between them.

If, however, the first gate operation failed, the two arms the gate was acting on will be cut of as a consequence of the σ_z failure outcome. The σ_y measurement now acts on one part of a product state, leaving the other part – the two stars – unchanged. Now, a second try is possible by application of the entangling operation to the center qubits. The success outcome will be the same as above, on failure the centers will be cut off, leaving the six qubits in the product state vector $|+\rangle^{\otimes 6}$. The failure outcome is the result of two consecutive failures of these entangling gates, so $p_{\text{success}} = 1 - (1 - p_{\text{gate}})^2$, which is $p_{\text{success}} = 3/4$ in the case of the linear optics gate mentioned earlier.

All these operations do not need any classical post-processing or re-routing, therefore, this scheme is suitable to be used in the procedures introduced above. Because the center qubits are simply cut off when a failure occurs, this procedure and the bond percolation involving the arms are completely independent. So, both processes together can be modelled as a mixed site/bond percolation with the site probability being $p_{\text{s}} = p_{\text{success}}$.

As long as the bond probability and the initial state preparation (site-) probability are above the percolation threshold of the respective lattice, this scheme might be useful to further reduce the size of the initial star shaped quantum states.

2.3 Phase transitions in the computational power

A related yet complementary motivation to look at percolation theory in the context of quantum computing is the question whether a universal resource state can be prepared in principle. Surely, above the percolation threshold a universal resource for quantum computing can be extracted from a percolating lattice in an efficient manner, as has been described in the previous section. This approach, in turn, will fail almost surely below the threshold. Indeed, it has been shown in Ref. [9] that for $p < p^{(\text{c})}$ the statistics of all possible measurements (so all possible quantum computations) on the state represented by the resulting graph can almost certainly be efficiently simulated

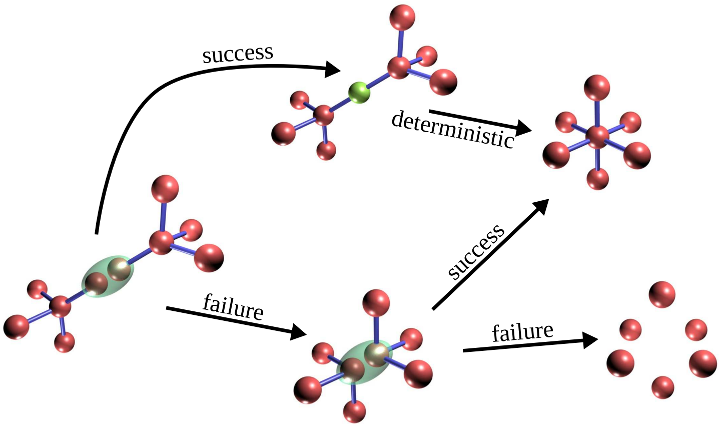

Fig. 6. A pair of five-qubit stars (stars with four arms and one central qubit) can be used to create a single seven-qubit star with a probability of $p_{\rm success} = 1 - (1 - p_{\rm gate})^2$. With $p_{\rm failure} = (1 - p_{\rm gate})^2$ the six arm qubits are separated from each other.

on a classical computer. This effect can indeed be seen as a phase transition in computational power of the resource state described by the lattice (caused by the phase transition of the mean cluster size). Assuming that BQP≠BPP[10], this phase transition distinguishes between states that can be used for universal quantum computing and those which cannot (see also Ref. [19]).

This phase transition can be made precise under the assumption that a certain quantity – the entanglement width – provides a handle on the computational universality of a family of states, by using it as an order parameter and investigating its properties in the sub- and supercritical regime. The entanglement width $E_{{\rm w}d}$ is a measure of entanglement, introduced in Ref. [44]. It can be used to characterize the amount of entanglement – somehow the degree of "non-classicality" – contained in a quantum system. We refer to Ref. [44] for its definition and properties. For $d = 2$ it can be seen that the amount of entanglement contained in the states described by the lattice jumps from a logarithmic to a linear scaling in L at the percolation threshold:

Supercritical

Above the respective percolation threshold of a particular lattice of size $L^{2+\mu}$ with $\mu > 0$, an $L \times L$ square lattice can be extracted efficiently, as shown in Chapter 2.2. In Ref. [9] an alternative algorithm is proposed which achieves $\mu = 0$ for $d = 2$.

Theorem 2 (Supercritical states). *Let $p > p_2^{(\rm c)}$. With probability*

$$\lim_{L\to\infty} P_p(L) = 1, \tag{23}$$

[10] This means that quantum computers are actually more powerful than classical computers. Although this feature is assumed to be highly plausible, the strict separation of the complexity classes remains unproven today, not unlike the situation for P≠NP.

and an overall resource consumption of $O(L^2)$ an $L \times L$ cluster – a resource for universal quantum computing – can be generated from a square lattice.

In this approach, in contrast to the renormalization algorithm described in Section 2.2, the block size of the renormalized lattice is not held fixed [9], resulting in an improved scaling of the resource consumption in case of the two-dimensional square lattice.

Again, this approach can be used to grow the cluster state while computation moves on. With a fixed width new layers can be added on top, allowing for subsequent identification of new horizontal paths (and their connections to the underlying lattice). So it is not required to keep the whole lattice in memory, rather only a set of qubits of size $O(L)$.

A natural substructure of a percolating $n \times n$ lattice is already provided by the fact that there exist at least $m = O(n)$ vertex-disjoint crossing paths (which shall be event $\mathfrak{G}_m(n)$) in either dimension almost certainly in the limit of large n.

Consider the event of having at least one left-to-right crossing path in a two-dimensional $n \times n$ block, which happens with probability

$$P_p(\mathfrak{A}(n)) \geq 1 - \exp(-g(p)n^2), \tag{24}$$

$g(p) > 0$ (Eqn. 9). When this event is still happening when changing the state r arbitrary edges on the lattice, there have to be at least $r + 1$ vertex-disjoint left-to-right crossings in this block. With $r = \beta(p)n$ an inequality from Ref. [16] allows us to write

$$P_p(\mathfrak{G}_{\beta(p)n}(n)) \geq 1 - \left(\frac{p}{p - p_2^{(c)}}\right)^{\beta(p)n} \mathrm{e}^{-ng(p)}. \tag{25}$$

In the limit of $n \to \infty$ this probability approaches unity, iff

$$g(p) - \beta(p) \log\left(\frac{p}{p - p_2^{(c)}}\right) > 0, \tag{26}$$

which can always be achieved by suitable choice of β for a given p. Therefore, in the limit of large m, there are almost certainly m vertex-disjoint open left-to-right crossing paths in a square lattice of side length $n = m/\beta(p)$. In other words, the number of these crossings scales as $O(n)$.

Algorithm

The challenge now is to identify $O(L)$ suitable paths such that the reduction to a fully populated regular lattice with the tools of Chapter 2.2 is possible. In Ref. [9] a very detailed description of an algorithm achieving this goal can be found, some of the steps of which are illustrated in Fig. 7. The important stages are the following:

1. *Path identification*: An algorithm from maze solving, the *right-hand wall follower*, which can identifies the right-most path along a wall (which will be the

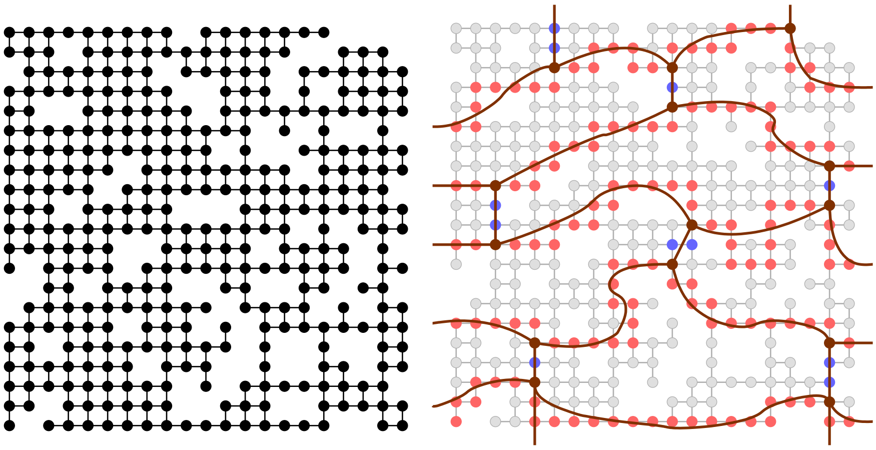

Fig. 7. Illustration of the algorithm to extract an $L \times L$ lattice with only constant overhead per site. The left picture shows the initial percolated lattice. At the right, the identified paths after alternating bridge decomposition and error correction, as well as the final graph after the quantum measurements are shown.

surface of the graph, or the previous path), is used. By applying it consecutively it finds a maximum set of non-intersecting crossing open paths in either direction.

Vertical paths are found by using every third path that result from the right-hand wall follower. For horizontal paths, the *2-local* version of this algorithm is used, which means that a graph distance on the underlying lattice to the previous path of at least 3 is ensured.

These gaps between neighboring paths ensure that the reduction can be performed in the end. A distance of 1 between neighboring horizontal paths is surely not enough when it comes to the shortening of intermediate parts by means of σ_y measurements. Having one qubit in between turns out to be not enough where intersections with vertical paths occur.

2. *Alternating bridge decomposition*: While fixing paths in the horizontal direction, the vertical ones are cut into segments between neighboring horizontal paths (*bridge decomposition*). By only using every other of these segments (alternating bridge decomposition), a hexagonal structure is imposed, simplifying the reduction stage as before, while still giving rise to a universal resource state. Due to the choice of using every third vertical path, the graph has essentially the same topology as a hexagonal lattice, up to local errors at the crossings. For an in-depth analysis we refer to Ref. [9].

   The bridge decomposition further assures a simple topology of the crossing points, because multiple crossings and common parts of horizontal and vertical paths are cut out.
3. *Correction of local errors*: All that is left at this stage is a hexagonal structure where the bonds consist of paths the vertices of which have vertex degree 2,

> and the sites consist of a couple of vertices with degree not necessarily 3. More specifically, they contain vertices of degree ≥ 3 on the horizontal paths (the *abutments*) and one vertex per crossing of degree ≥ 2 at the beginning (or end, respectively) of a vertical path. By appropriately removing the vertices in the middle of the abutments from the graph, the crossings are reduced to single vertices of degree 3 (a T-shaped junction), thus allowing for reduction of the lattice by local Pauli measurements.

After this classical identification process is done, the graph is reduced to a hexagonal lattice using local measurements as in Section 2.2.

Entanglement width

An $L \times L$ cluster state has an entanglement width of $E_{\mathrm{w}d} = O(L)$. Due to the fact that the entanglement width does not increase during one-way computation (which also is what is effectively done to extract the cluster from the percolated lattice in all known proposals, see also Chapter 2.2), the extracted cluster can be used to derive a lower bound to this entanglement width,

$$E_{\mathrm{w}d}(p > p^{(\mathrm{c})}) \geq O(L). \tag{27}$$

This bound holds for any regular lattice with $d \geq 2$ above its respective percolation threshold.

Subcritical

Below the percolation threshold the following behavior is observed:

Theorem 3 (Subcritical states). *For $p < p_2^{(\mathrm{c})}$, the corresponding quantum states can be simulated efficiently on a classical computer.*

The idea behing the proof is that the entanglement width is bounded by the entanglement width of the largest connected component [9]. Below the percolation threshold, this is of size $O(d \log L)$ [3], therefore bounding

$$E_{\mathrm{w}d} \leq O(\log L). \tag{28}$$

Making use of Ref. [3], it is shown that this bound holds for $2 \leq d \leq 6$. Further, the logarithmic scaling of the size of the largest connected component results in the computational space to be of size $\mathrm{poly}(L)$, so efficiently simulatable by classical means.

3 Quantum repeater networks for quantum communication

3.1 General concepts

Another quite natural application of percolation ideas in the context of quantum information processing is to find ways of designing quantum networks for quantum communication. This is now an application no longer related to quantum computation, but to quantum communication and quantum key distribution. Here, the concept of a network emerges quite naturally, and it seems meaningful to ask to what extent percolation ideas can help here.

The setting we will discuss in this section – reviewing the content of Refs. [1, 22, 34] – is slightly different from the previous ones: We will no longer make strict use of classical percolation ideas to identify threshold behavior in the possibility of preparing some quantum states. Instead, but in a closely related fashion, we will ask to what extent one can modify quantum measurement schemes to realize percolation-like settings which outperform standard bond percolation when naive measurement patterns are used. It will hence be not be sufficient to look at classical properties of quantum states as before, but we will have to consider the quantum states as such.

The idea of *quantum communication* or, more specifically, of *quantum key distribution* plays a seminal role in quantum information science: Indeed, whenever two laboratories share quantum systems in highly entangled states, by appropriate measurements and classical post-processing, a *secret key* can be extracted from the measurement data. This key can then be used for the secure transmission of classical data. For a review on this idea, which has led to numerous experimental implementations, and based on which commercial devices are available, see, e.g., Ref. [15]. For "sufficiently entangled" states one can indeed show that the resulting key distribution is unconditionally secure, in the sense that the security proof does not rely on unproven assumptions on the hardness of certain mathematical problems. Instead, it is merely assumed that an eventual eavesdropper could have access to any technological device, but is constraint by the laws of quantum theory as such in his or her possible action.

The functioning of such ideas, however, relies on the availability of sufficiently pure and entangled quantum states. If – in a sense – too much noise is in the system, which is the situation encountered in distribution of entangled state through noisy channels over large distances, then the key will no longer necessarily be secure. In such a situation, ideas of *quantum state distillation* or of *quantum repeaters* offer a solution: This amounts to a fragmentation of the whole transmission line to a sequence of *repeater stations*, or, more generally, to a network of repeater stations. And here, needless to say, is where percolation ideas come in.

We will start by describing the setting of Ref. [1]: Consider some lattice, the vertices of which are identified with repeater stations, or laboratories in which one can perform quantum operations. Between such vertices, non-maximally entangled states can be created. Once two vertices share entangled quantum systems, we will say that they are connected by a *"quantum edge"*. Following this entanglement distribution process, only "local" operations constrained to the qubits of a repeater station are possible, coordinated by classical communication. This is the setting of *local operations with classical communication* (LOCC), which is the standard paradigm when considering distributed quantum systems. Based on such operations, purification protocols [6, 46] can transform a single pair or a chain of two pairs into a maximally entangled pair, the equivalent to an open bond. This process is intrinsically probabilistic, which represents the bond probability. The probability of success of such a protocol (and with it the bond probability) depends not only on the amount of entanglement that is available in the very beginning, but also on the distillation strategy employed. Therefore, we cannot assign unique bond probabilities to a lattice with quantum bonds.

3.2 Classical entanglement percolation

We will now compare two different settings, following Refs. [1, 34]. The first approach is to make use of natural physical measurements where each of the quantum edges is purified separately. This then leads to a familiar edge percolation problem: A quantum edge corresponds to sharing a non-maximally entangled state with a state vector of the form

$$|\varphi\rangle = {\lambda_1}^{1/2}|0,0\rangle + {\lambda_2}^{1/2}|1,1\rangle. \tag{29}$$

λ_1 and λ_2 are the *Schmidt coefficients* satisfying $\lambda_1 + \lambda_2 = 1$, assuming that $\lambda_1 \geq \lambda_2$. For simplicity, all non-maximally entangled states are taken to be pure. It is known that there exist local quantum operations assisted by classical communication (LOCC) that transform $|\varphi\rangle$ into a maximally entangled pair with state vector $2^{-1/2}(|0,0\rangle + |1,1\rangle)$, with a probability of success of

$$p_1 = \min(1, 2(1 - \lambda_1)). \tag{30}$$

This probability of success will be referred to as *singlet conversion probability* (SCP) of a single quantum edge. In general, having a LOCC conversion protocol means that there exist operators $M_A^{(k)}, M_B^{(k)}$ satisfying $\sum_k (M_A^{(k)})^\dagger M_A^{(k)} \leq \mathbb{1}$ and $\sum_k (M_B^{(k)})^\dagger M_B^{(k)} \leq \mathbb{1}$ which can be implemented with LOCC such that

$$(M_A^{(k)} \otimes M_B^{(k)})|\varphi\rangle = (p^{(k)})^{1/2}(|0,0\rangle + |1,1\rangle)2^{-1/2}, \tag{31}$$

$p_1 = \sum_k p^{(k)}$. In this case, the filtering is quite simple: we merely need to consider a single successful outcome, $k = 1$, and take $M_A^{(1)} = (\lambda_2/\lambda_1)^{1/2}|0\rangle\langle 0| + |1\rangle\langle 1|$ and $M_B^{(2)} = \mathbb{1}$. As can be readily verified, this leads to a maximally entangled state with the above probability of success. This filtering process is called *Procrustean method* [4].

Once one has a maximally entangled pair, we encounter the situation as before, and we say the corresponding edge is open, in the sense of usual edge percolation, and as being used throughout this chapter. If the protocol fails, one is led to an unentangled state, and the edge is closed. Hence, λ_1 governs the percolation behavior and defines the edge probability. If one is above the percolation threshold of the respective lattice, a connected path can be identified almost certainly between any two vertices A and B of the lattice: Let $\mathfrak{c}(A, B)$ denote the event that A and B are connected by maximally entangled bonds. Then,

$$\lim_{\text{dist}(A,B)\to\infty} P(\mathfrak{c}(A, B)) > 0 \tag{32}$$

if and only if $p_1 > p^{(c)}$. Physically, this means that A and B share a perfect quantum channel through which the state of a single qubit can be sent in a lossless fashion. This setting which is the natural analogue of edge percolation will be called *classical entanglement percolation (CEP)*.

3.3 Quantum percolation strategies

The quantum character of the involved states ("quantum edges"), however, allows for some improvement. The main observation is that for quantum systems, one does not necessarily have to perform the above measurements, leading individually to open or closed edges, but can resort to appropriate local collective operations. If the task is to achieve a perfect quantum channel between any two vertices A and B in a lattice, then we have achieved this when having a maximally entangled pair of qubits between A and B at our disposal. This process should succeed in the best possible fashion. The aim hence is to maximize the SCP, so the probability to achieve a maximally entangled pair between two vertices A and B, but not necessarily exploiting the above CEP.

To exemplify the mechanism, let us first consider a one-dimensional chain with three vertices, A, B, and C, such that A and B, and B and C share a state with state vector $|\varphi\rangle$ each (they are connected by a quantum edge). This is a quantum repeater situation. Then, clearly, applying the above filtering corresponding to CEP twice, one succeeds with a probability p_1^2.

But, since vertex B holds two quantum systems, one can also do a collective local operation. The SCP asks for the optimal probability $p = \sum_k p^{(k)}$ such that

$$\begin{aligned}(M_A^{(k)} \otimes M_{B_1,B_2}^{(k)} \otimes M_C^{(k)})|\varphi\rangle_{A,B_1}|\varphi\rangle_{B_2,C} = (p^{(k)}/2)^{1/2} \times \\ (|0,0\rangle_{A,C} + |1,1\rangle_{A,C})|0,0\rangle_{B_1,B_2}.\end{aligned} \tag{33}$$

Clearly, $p \geq p_1^2$, as the protocol in Eq. (33) includes the prescription of CEP where one tries to purify the entanglement between A and B_1 and B_2 and C individually.

We can also easily obtain an upper bound to the probability of success: The class of protocols defined in Eq. (33) is included in the one of Eq. (31) in the case when A, B_1, and B_2 are treated as a single system. Doing such collective operations (instead of only in B_1 and B_2) can only improve the probability of success, hence this upper bound.

Somewhat surprisingly, it can be shown that this bound can indeed be achieved [6, 46]. In B_1 and B_2 one performs a joint measurement $k = 1, \ldots, 4$ with

$$M_{B_1,B_2}^{(k)} = |\psi^{(k)}\rangle_{B_1,B_2}\langle\psi^{(k)}|_{B_1,B_2}, \tag{34}$$

where

$$|\psi^{(1)}\rangle = (|0,0\rangle + |1,1\rangle)2^{-1/2},\ |\psi^{(2)}\rangle = (|0,0\rangle - |1,1\rangle)2^{-1/2}, \tag{35}$$

$$|\psi^{(3)}\rangle = (|0,1\rangle + |1,0\rangle)2^{-1/2},\ |\psi^{(4)}\rangle = (|0,1\rangle - |1,0\rangle)2^{-1/2}. \tag{36}$$

Two of the outcome probabilities of this four-outcome measurement are equal to $p_{\text{max}} = (\lambda_1^2 + \lambda_2^2)/2$, whereas the two further outcome probabilities are given by $p_{\text{min}} = \lambda_1\lambda_2$. This local filtering is in fact nothing but a collective *"Bell"-measurement* in the $\sigma_z\sigma_z$-basis. This procedure reaches the above upper bound, and hence

$$p = p_1. \tag{37}$$

This scheme, simple as it is, already shows that with collective operations, one can improve the involved probabilities, from p_1^2 to p_1. Note that here the SCP is taken as a figure of merit, whereas other figures of merit, like the average final entanglement quantified in terms of an entanglement measure, can also be meaningful quantifiers of success of the scheme [34]. Also, the same protocol is optimal in case one has two different pure states in the Schmidt basis at hand.

3.4 Absence of improvement for one-dimensional systems

A first natural instance where this process can be applied is the case of one-dimensional systems. In this case, it turns out, there is no improvement possible [22, 34]. The improvement that is possible for two quantum edges does not persist in the asymptotic limit of a quantum chain, and an exponential decay of the SCP with the distance between two vertices of the chain cannot be avoided [1]. Moreover, the only way of having unit SCP is to have a chain of maximally entangled pairs in the first place. The argument to show this makes use of the entanglement measure of the *concurrence* of a two-qubit state. For a state vector $|\varphi\rangle = \sum_{j,k} T_{j,k}|j,k\rangle$ its concurrence is found to be $C(\varphi) = 2|\text{det}(T)|$, in terms of the 2×2-matrix T. In the situation of having a one-dimensional chain of repeaters involving $N+1$ quantum edges and hence $N+w$ vertices, involving N correlated measurements, one then finds that the maximum average concurrence of the first and the last qubit in the chain is given by

$$C = \sup_{\mathcal{M}} \sum_{r_1,\dots,r_N} 2 \left|\text{det}\left(|\varphi_1\rangle\langle\varphi_1|M_1^{(r_1)}|\varphi_2\rangle\langle\varphi_2|\dots M_N^{(r_N)}|\varphi_{N+1}\rangle\langle\varphi_{N+1}|\right)\right|. \tag{38}$$

In this expression, $\mathcal{M}$ stands for all measurements, whereas the $M_i^{(r_i)}$ are 2×2 matrices specifying the r_i-th outcome at site i, corresponding to the state vector $|\mu_i\rangle = \sum_{j,k}(M_i^{(r_i)})_{j,k}|j,k\rangle$ of this specific measurement outcome. From this expression demonstrating the exponential decay of the concurrence with the distance between two vertices in a one-dimensional chain, one can also derive the exponential decay of the SCP. Hence, while for two repeaters one can find a better probability of success, this does not lead to a different asymptotic performance of this type of quantum percolation strategy.

For other lattices, different from simple one-dimensional chains, one can improve the situation, however, compared to classical entanglement percolation, as we will discuss in the subsequent subsection. In this way, a lattice having some quantum edges can be transformed into a different lattice with maximally entangled pairs being randomly distributed, whereas the distribution will depend on the strategy employed (e.g., CEP or collective operations). This approach can outperform methods relying solely on classical probabilities and equivalences between these lattices (e.g., covering, matching, or dual lattices). This is an effect which is caused by the quantum nature of the bonds.

3.5 Examples of quantum percolation strategies

Hexagonal lattice

An example to highlight the difference to a purely classical approach is the following: One starts from the situation where neighbors on a hexagonal lattice share two specimens of the same state. This situation is very natural in the quantum case: One simply has to distribute a state of the same kind twice. The aim is to establish a connected open path between arbitrary vertices (taken from a triangular sublattice).

Theorem 4 (Singlet conversion probability in the hexagonal lattice). *Consider a hexagonal lattice, where nearest neighbors share* $|\varphi\rangle^{\otimes 2}$ *with* $|\varphi\rangle = {\lambda_1}^{1/2}|0,0\rangle + {\lambda_2}^{1/2}|1,1\rangle$*. Then, if* $\lambda_1 > (1/2 + \sin(\pi/18))^{1/2}$ *but* $1 - \lambda_1 > \sin(\pi/18)$*, CEP does not lead to edge percolation, whereas joint measurements result in an infinite connected cluster.*

We start by describing the situation for CEP: Each pair of neighbors shares $|\varphi\rangle^{\otimes 2}$, so the optimal SCP of transforming these two copies into a maximally entangled state (an open edge) is $p_1 = 2(1 - \lambda_1^2)$, where the λ_1^2 originates from the fact that we have two specimens at hand. It is easy to see that with the above choice for λ_1 one arrives at an edge probability of

$$2(1 - \lambda_1^2) < 1 - 2\sin(\pi/18) = p_{\hexagon}^{(c)} \approx 0.6527. \tag{39}$$

Hence, with this strategy one falls below the percolation threshold of the hexagonal lattice.

We now turn to a strategy exploiting the more potential of the quantum setting. We resolve every second site as shown in Fig. 3.5. Now – as described in the caption of Fig. 3.5, one performs the protocol of Eq. (33), leading to a !indexlattice!triangulartriangular lattice. The edge probability in this new triangular lattice is given by

$$2\lambda_2 = 2(1 - \lambda_1) > 2\sin(\pi/18) = p_{\Delta}^{(c)} \approx 0.3473 \tag{40}$$

with the above choice for λ_1. Hence we are now above the percolation threshold for the triangular lattice, and one can proceed as usual to demonstrate that an infinite cluster emerges. Therefore, bond percolation processes with quantum bonds are related to each other in a different way than those with classical bonds. More specifically, it is possible to find a parameter regime for which the hexagonal lattice does not percolate, but can be transformed into a percolating triangular lattice, if quantum operations are allowed for.

Equally interesting is the proposal to use the optimal singlet conversion strategy to transform a square lattice into two independent square lattices of doubled size, for which the bond probability is larger than in the original lattice.

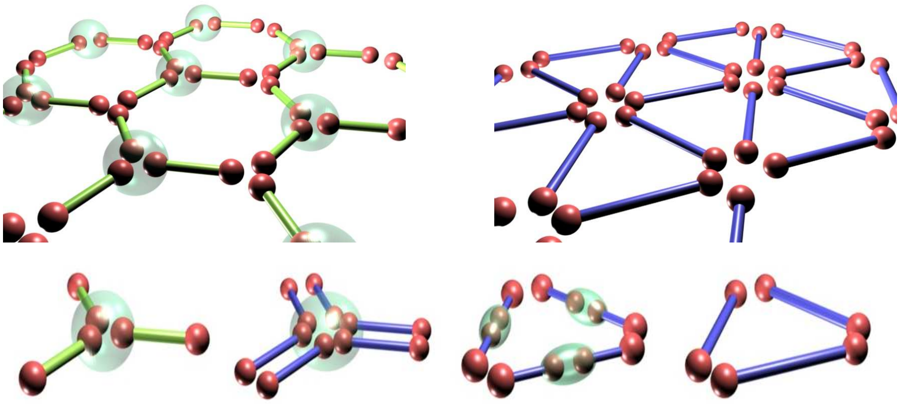

Fig. 8. Top row: A hexagonal and a triangular lattice. Each green bond is constituted by two entangled pairs (here shown as blue bonds). Every second vertex is highlighted. The bottom row shows how the transformation works: Pairs of qubits are taken from the highlighted vertices. Onto the respective pairs of edges, the protocol of Eq. (33) is applied, succeeding with probability p. Using this quantum operation, the lattice is transformed from a hexagonal lattice to a new triangular lattice (for cluster states this actually corresponds to performing a σ_y-measurement on the highlighted sites). On this new triangular lattice, an edge is open if the protocol of Eq. (33) has been successful.

Square lattice

Another striking example of what can happen in quantum percolation is a nested protocol of distillation steps. This has been proposed in Ref. [34], which develops the ideas of Ref. [1] in more detail. Here, a set of non-maximally entangled pairs will be used which cannot be transformed into maximally entangled ones deterministically by employing the process mentioned above, so $p_1 < 1$.

Consider the situation of Fig. 9: Initially, arranged in a square, one has four identical pairs in a state with state vector $|\varphi\rangle$. Then one can perform local operations to the marked subsystems, as Bell-measurements in the $\sigma_x\sigma_z$-basis (Fig. 9a). The original states $|\varphi\rangle$ have Schmidt coefficients $\{\lambda_1, \lambda_2\}$, the new state with state vector $|\varphi'\rangle$ will have Schmidt coefficients, the smaller of which is given by

$$\lambda_2' = (1 - (1 - (4\lambda_1\lambda_2)^2)^{1/2})/2. \tag{41}$$

Given the four original input states, by applying this local filtering twice, two such output states will be left. By using a further local operation (Fig. 9b) – an instance of a distillation protocol – a final state vector $|\varphi''\rangle$ can be achieved (Fig. 9c) that is more entangled than $|\varphi'\rangle$. The largest Schmidt coefficient of this final state is given by $\lambda_1'' = \max\{1/2, (\lambda_1')^2\}$ [34, 33]. This state is maximally entangled iff both Schmidt coefficients are equal, so iff $\lambda_1'' = 1/2$. The range in which that can be achieved is

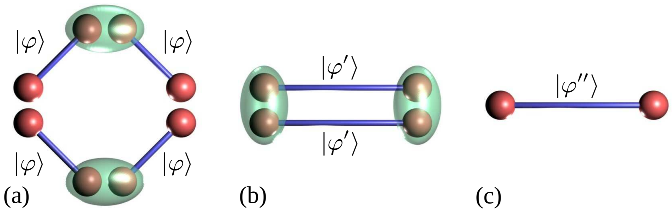

Fig. 9. Transformation of a square lattice to another square lattice by means of a pair of (a) two-partite measurements and (b) an entanglement distillation on the resulting state. Starting with a square lattice where each bond consists of $|\varphi\rangle^{\otimes 2}$ (similar to Fig. 3.5), this process generates a square lattices with $\sqrt{2}$ times the lattice constant.

$$1/2 \leq \lambda_1 \leq \lambda_1^* = \frac{1}{2}\left(1 + (1 - (2(\sqrt{2}-1))^{1/2})^{1/2}\right) \approx 0.65. \tag{42}$$

Hence, certain non-maximally entangled “quantum edges” (with CEP probability smaller than one) can be deterministically transformed into maximally entangled bonds, so edges that are open with unit probability.

Note that each of the above steps involving local operations can be achieved deterministically, resorting to LOCC operations. In each case, the measurement outcome on each vertex has to be available to the other vertices to do the appropriate corrections and to render the scheme deterministic. Such an idea can be used to transform a square lattice, in which vertices share “quantum edges” with nonunit CEP bond probability in form of non-maximally entangled states into a square lattice with unit edge probability.

Theorem 5 (Singlet conversion in the square lattice). *A square lattice where nearest neighbors share $|\varphi\rangle^{\otimes 2}$ can be transformed into a square lattice, the edges of which being fully occupied by maximally entangled pairs, using collective measurements at the individual sites, given that $\lambda_1 < \lambda_1^*$.*

One could say that the protocol transforms a square lattice with

$$1 \geq p_1 \geq 2\lambda_2^* \approx .70 \tag{43}$$

into a square lattice with $p_1 = 1$. Again, and more clearly so, this is a transformation between different bond percolation processes that crosses the critical threshold in a way that is not possible in percolation with classical edges.

Square lattice II

To consider yet another effect, we will have a second look at the square lattice[34]. As mentioned in Section 3.3, a joint measurement can be applied to a two-pair chain such that the SCP of the new bond equals the original bonds’ SCP on average. Let us apply this operation at every other site of a square lattice as indicated in Fig. 10, thereby replacing it with two disjoint square lattices with double the lattice constant.

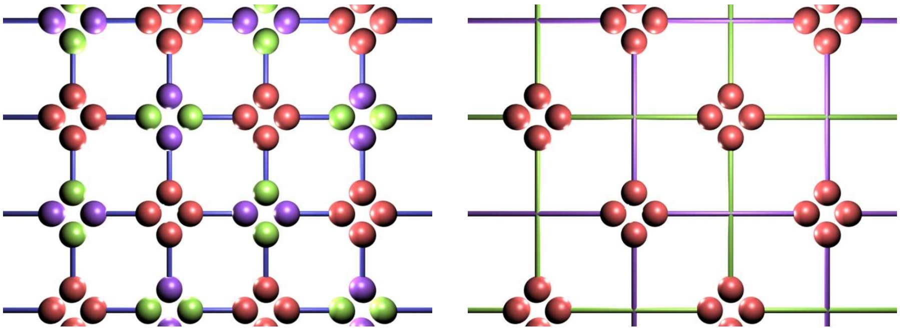

Fig. 10. Transformation of a square lattice into two disjoint ones by means of two qubit measurements. Left: At every other lattice site two operations are carried out, affecting the green and purple qubits, respectively. Right: The resulting lattice consists of two disjoint square lattices.

Neighboring sites in the resulting graph sit on different lattices. We pick two pairs of neighbors, $a = (A, A')$ and $b = (B, B')$ and ask for the probability of finding at least one open path between these two sets of sites, so the event $\mathfrak{H} = a \leftrightarrow b$. On the doubled lattices this accounts to the event of having at most one path out of $A \leftrightarrow B$ and $A' \leftrightarrow B'$. In the limit of large separation of a and b, the probability amounts to

$$P'_p(\mathfrak{H}) = \theta^2(p)(2 - \theta^2(p)), \tag{44}$$

where $\theta(p)$ is the percolation probability, so the probability of a given site to belong to an infinite spanning path.

In contrast, on the original lattice this quantity asymptotically fulfils

$$P_p(\mathfrak{H}) \leq \theta^2(p)(2 - P_p(A \leftrightarrow A'))^2. \tag{45}$$

In Ref. [34] a Monte Carlo simulation to find the quantity $P_p(A \leftrightarrow A')$ for $p \searrow p_2^{(\mathrm{c})}$ is carried out, yielding $2 - P_p(A \leftrightarrow A') \approx 1.313$ in comparison to $(2 - \vartheta^2(p_2^{(\mathrm{c})}))^{1/2} = \sqrt{2}$. So, a doubling of the lattice spacing using joint quantum operations can be advantageous when it comes to the connection of pairs of sites. It is conjectured [34] that this property holds for all p. Further examples along similar lines are also presented in Ref. [34].

4 Summary and open problems

4.1 Open problems

To elaborate the aforementioned invitation to this topic, we list some open problems that arose from the work so far. Concerning issues of “classical percolation for cluster state preparation” this includes the following list of problems:

- In dimensions $d \geq 3$ crossing paths in different directions do not imply a crossing of these paths. Further development could aim at finding an algorithm similar to the one in Section 2.3. These would be able to identify structures that are topologically equivalent to universal computing resource states in a percolated lattice at higher dimension, thus allowing this tool to be used for resource state production for arbitrary gate probabilities with resources amounting to $O(L^2)$.
- While it appears from numerical results that a constant overhead per site cannot be achieved with static renormalization, its optimal scaling is not evident. According to Fig. 4, this is at most logarithmic, rather than polynomial as the bound in Theorem 1 suggests.
- Further, an investigation towards the required initial resource size would help implement these protocols in the future. After all, small highly entangled quantum states have to be produced in the very beginning, the smaller the better. There will be a tradeoff between the possible gate probabilities and the number of initial qubits needed, the exact behaviour of which might be interesting. It includes optimization over possible local linear optical operations which do not require global re-routing.
- All ideas presented in this chapter presume ideal states, optical elements, or quantum gates. This, however, will of course never be achieved in any implementation. The general problem of quantum error correction needs to be revisited in the framework of percolation, as there might arise new ways for error correction of special classes of errors (for first ideas see Ref. [28]).
- As has been shown in Ref. [18], there are a number of resource states very different from the cluster states which form universal resources for quantum computing. Dictated by the underlying physical architecture, there might be states that are more suitable than cluster states, exploiting the specific strengths of the respective architecture. However, if probabilistic processes are involved (like PEPS projections with linear optics), it is not obvious how to generalize the percolation scheme to other states.

  Information flow through these states is in general not ruled as easily as with single qubit Pauli measurements. Suitable applications of SWAP gates might help in using percolated versions of these resources for computing, though not giving a solution for distillation of hole-free states as such (as may be needed as resource for error correction).

Complementing these questions on the tightness of the given bounds, there are a number of open questions relating to the idea of entanglement percolation:

- The aim of Section 3 was to look at the problem of entangled state distribution with the tool of percolation. One – relatively vague – open problem is to find applications of this idea in a context different from entanglement distribution as such.
- One step in this direction could be to combine the idea of entanglement percolation with ones of quantum computation in the first part of this book chapter. Clearly, entanglement percolation could also help to further reduce the required resources in a renormalized lattice, albeit requiring more difficult collective mea-

surements on the way, the probabilistic nature of which might lead to some tradeoff.

- Within restricted classes of percolation settings (such as restrictions to the geometry or the possible local operations), it seems important to identify the optimal entanglement percolation strategy.
- It is still far from clear how to fully incorporate mixed states in this setting, which seems important when considering lossy quantum channels.

4.2 Summary

Even though the first ideas to incorporate quantum information with percolation theory were only proposed very recently [28, 1], it seems clear that open problems in quantum state preparation and quantum communication can indeed benefit from results known in percolation theory. Since randomness is intrinsic in quantum mechanics, one often has to overcome the probabilistic nature of quantum operations. As we have discussed, a context in which this most naturally is in quantum computation using lattice system. Here static lattices with a non-unit probability of the existence for bonds and sites occur naturally in cluster state preparation, where the random nature is due to probabilistic quantum gates, or due to Mott defects in the preparation. Ideas of percolation – specifically questions of when crossing clusters exist – led to methods of renormalization that effectively remove this probabilistic aspect from quantum state preparation. Along similar lines, it seems realistic to expect related instances where percolation ideas help to overcome the intrinsic randomness of quantum mechanical state manipulation.

A key challenge will be, needless to say, to take mixed quantum states fully into account, and to see in quantitative terms to what extent renormalization ideas can be combined with methods of fault tolerance, error correcting codes, or protection strategies in state preparation for the use of quantum computation [12, 45, 39, 36]. A step into this direction has been taken in Ref. [28].

We have also seen that when taking the quantum nature of the states seriously, one can often outperform strategies based on simple measurements and invoking notions of edge percolation. Such an approach seems particularly suitable for realizations in quantum networks for quantum communication and key distribution, but could equally well also be applied to the above context of quantum computation. New phenomena emerge when appropriate collective operations are allowed for, giving rise to an interesting interplay between questions of entanglement theory on the one hand and percolation theory on the other hand.

On a related but different note, ideas of percolation seem also provide powerful tools in a slightly different context, namely to study correlation and entanglement properties of quantum many-body systems from the perspective of quantum information. Notably, the scaling of the entanglement entropy in the Ising model can be assessed by invoking concepts of classical percolation [17]. The question of relating percolation ideas to problems in quantum information science is at its infancy, but one should expect more applications to come.

Acknowledgments

We warmly thank T. Rudolph for numerous fruitful and fun discussions on the subject of the connection between percolation theory and quantum information theory, who is also coauthor of the joint publication Ref. [28]. We also thank T. Acin, D.E. Browne, W. Dür, A. Miyake, and T.J. Osborne for correspondence on percolation ideas in the quantum information context. This work has been supported by Microsoft Research through the European PhD Programme, the EPSRC, the EU (QAP), and the EURYI award scheme.

References

1. A. Acín, J. I. Cirac, and M. Lewenstein. Entanglement percolation in quantum networks. *Nature Physics*, 3:256, 2007.
2. M. Aizenman. On the number of incipient spanning clusters. *Nuclear Physics B*, 485:551, 1997.
3. M. Z. Bazant. Largest cluster in subcritical percolation. *Phys. Rev. E*, 62:1660–1669, 2000.
4. C. Bennett, H. Bernstein, S. Popescu, and B. Schumacher. Concentrating partial entanglement by local operations. *Phys. Rev. A*, 53:2046, 1996.
5. J. A. Bondy and U. S. R. Murty. *Graph theory with applications*. Macmillan, 1976.
6. S. Bose, V. Vedral, and P. L. Knight. Purification via entanglement swapping and conserved entanglement. *Physical Review A*, 60:194, 1999.
7. S. Bravyi and R. Raussendorf. Measurement-based quantum computation with the toric code states. *Physical Review A*, 76:022304, 2007.
8. H. J. Briegel and R. Raussendorf. Persistent entanglement in arrays of interacting particles. *Physical Review Letters*, 86:910, 2001.
9. D. E. Browne, M. B. Elliott, S. T. Flammia, S. T. Merkel, A. Miyake, and A. Short. Phase transition of computational power in the resource states for one-way quantum computation. *arXiv:0709.1729*, 2007.
10. D. E. Browne and T. Rudolph. Resource-efficient linear optical quantum computation. *Phys. Rev. Lett.*, 95:010501, 2005.
11. J. Calsamiglia and N. Lütkenhaus. Maximum efficiency of a linear-optical bell-state analyzer. *Applied Physics B*, 72:67, 2001.
12. C. Dawson, H. Haselgrove, and M. Nielsen. Noise thresholds for optical quantum computers. *Phys. Rev. Lett.*, 96:020501, 2006.
13. J. Eisert. Optimizing linear optics quantum gates. *Physical Review Letters*, 95:040502, 2005.
14. J. Eisert and M. M. Wolf. *Quantum computing, in Handbook of innovative computing*, chapter 8, page 253. Springer, New York, 2004.
15. N. Gisin, G. Ribordy, W. Tittel, and H. Zbinden. Quantum cryptography. *Rev. Mod. Phys.*, 74:145, 2002.
16. G. Grimmett. *Percolation*. Springer, 2nd edition, 1999.
17. G. Grimmett, T. J. Osborne, and P. Scudo. Entanglement in the quantum ising model. 2007.
18. D. Gross and J. Eisert. Novel schemes for measurement-based quantum computing. *Physical Review Letters*, 98:220503, 2007.

19. D. Gross, J. Eisert, N. Schuch, and D. Perez-Garcia. Measurement-based quantum computation beyond the one-way model. *Physical Review A*, 76:052315, 2007.
20. D. Gross, K. Kieling, and J. Eisert. Potential and limits to cluster state quantum computing using probabilistic gates. *Physical Review A*, 74:042343, 2006.
21. J. M. Hammersley and W. Morton. Poor man's monte carlo. *Journal of the Royal Statistical Society (B)*, 16:23, 1954.
22. L. Hardy and D. D. Song. Entanglement swapping chains for general pure states. *Phys. Rev. A*, 62:052315, 2000.
23. M. Hein, W. Dür, J. Eisert, R. Raussendorf, M. V. den Nest, and H.-J. Briegel. Entanglement in graph states and its applications. *quant-ph/0602096*, 2006.
24. M. Hein, J. Eisert, and H. J. Briegel. Multiparty entanglement in graph states. *Physical Review A*, 69:062311, 2004.
25. J. Hoshen and R. Kopelman. Percolation and cluster distribution. I. cluster multiple labeling technique and critical concentration algorithm. *Physical Review B*, 14:3438, 1976.
26. K. Kieling, D. Gross, and J. Eisert. Minimal resources for linear optical one-way computing. *Journal of the Optical Society of America B,* 24(2):184, 2006.
27. K. Kieling, D. Gross, and J. Eisert. Cluster state preparation using gates operating at arbitrary success probabilities. *New Journal of Physics*, 9:200, 2007.
28. K. Kieling, T. Rudolph, and J. Eisert. Percolation, renormalization, and quantum computing with non-deterministic gates. *Phys. Rev. Lett.*, 99:130501, 2007.
29. E. Knill, R. Laflamme, and G. Milburn. A scheme for efficient linear optics quantum computation. *Nature*, 409:46, 2001.
30. Y. L. Lim, S. Barrett, A. Beige, P. Kok, and L. Kwek. Repeat-until-success quantum computing using stationary and flying qubits. *Phys. Rev. A*, 73:012304, 2006.
31. O. Mandel, M. Greiner, A. Widera, T. Rom, T. W. Hänsch, and I. Bloch. Controlled collisions for multi-particle entanglement of optically trapped atoms. *Nature*, 425:937, 2003.
32. M. A. Nielsen and I. Chuang. *Quantum computation and quantum information*. Cambridge University Press, 2000.
33. M. A. Nielsen and G. Vidal. Majorization and the interconversion of bipartite states. *Quantum Information and Computation*, 1:76, 2001.
34. S. Perseguers, J. I. Cirac, A. Acin, M. Lewenstein, and J. Wehr. Entanglement distribution in pure state quantum networks. 2007.
35. T. B. Pittman, B. C. Jacobs, and J. D. Franson. Probabilistic quantum logic operations using polarizing beam splitters. *Physical Review A*, 64:062311, 2001.
36. J. Preskill. Fault-tolerant quantum computation. *quant-ph/9712048*, 1997.
37. R. Raussendorf and H.-J. Briegel. A one-way quantum computer. *Physical Review Letters*, 86:5188, 2001.
38. R. Raussendorf, D. E. Browne, and H. Briegel. Measurement-based quantum computation with cluster states. *Physical Review A*, 68:022312, 2003.
39. R. Raussendorf and J. Harrington. Fault-tolerant quantum computation with high threshold in two dimensions. *Phys. Rev. Lett.*, 98:190504, 2007.
40. S. Scheel and N. Lütkenhaus. Upper bounds on success probabilities in linear optics. *New Journal of Physics*, 6:51, 2004.
41. D. Schlingemann and R. F. Werner. Quantum error-correcting codes associated with graphs. *Physical Review A*, 65:012308, 2002.
42. R. Sedgewick. *Algorithms*. Addison-Wesley, 1983.
43. M. van den Nest, J. Dehaene, and B. D. Moor. Graphical description of the action of local clifford transformations on graph states. *Physical Review A*, 69:022316, 2004.

44. M. van den Nest, A. Miyake, W. Dür, and H. J. Briegel. Universal resources for measurement–based quantum computation. *Phys. Rev. Lett.*, 97:150504, 2006.
45. M. Varnava, D. Browne, and T. Rudolph. Loss tolerance in one-way quantum computation via counterfactual error correction. *Phys. Rev. Lett.*, 97:120501, 2006.
46. G. Vidal. Entanglement of pure states for a single copy. *Physical Review Letters*, 83:1046, 1999.
47. P. Walther, K. Resch, T. Rudolph, E. Schenck, H. Weinfurter, V. Vedral, M. Aspelmeyer, and A. Zeilinger. Experimental one-way quantum computing. *Nature*, 434:169, 2006.

Index