EPJ Quantum Technology
a SpringerOpen Journal

Open Access

CrossMark

Methodology for bus layout for topological quantum error correcting codes

Martin Wosnitzka, Fabio L Pedrocchi* and David P DiVincenzo

*Correspondence:
fabio.pedrocchi@physik.rwth-aachen.de
JARA Institute for Quantum
Information, RWTH Aachen
University, Aachen, 52056, Germany

Abstract

Most quantum computing architectures can be realized as two-dimensional lattices of qubits that interact with each other. We take transmon qubits and transmission line resonators as promising candidates for qubits and couplers; we use them as basic building elements of a quantum code. We then propose a simple framework to determine the optimal experimental layout to realize quantum codes. We show that this engineering optimization problem can be reduced to the solution of standard binary linear programs. While solving such programs is a NP-hard problem, we propose a way to find scalable optimal architectures that require solving the linear program for a restricted number of qubits and couplers. We apply our methods to two celebrated quantum codes, namely the surface code and the Fibonacci code.

Keywords: topolocial quantum error correcting codes

1 Introduction

Since the theoretical demonstration of fault-tolerant quantum information processing, a holy grail of modern physics has been to realize fault-tolerant quantum computing architectures in the lab. While this still remains a very challenging task, many experimental advances have been achieved. Arguably, one can hope to see the first small-size implementations in a near future.

Among the most promising quantum computing platforms, one finds so called topological quantum codes [1]. The main idea is to encode quantum information (in the form of logical qubits) using a large number of physical qubits. The additional degrees of freedom introduced in the Hilbert space then allow the extraction of some information about the errors induced by the environment (the error syndrome) and to correct them without collapsing the stored logical qubit. Furthermore, topological codes are, by definition, immune to local and static perturbations [1].

Most of the topological quantum codes are realizable as a lattice of qubits (some of them might require qudits instead) that are coupled to each other. Depending on the specifics of the quantum code, one qubit might be coupled to several other qubits in its neighborood. In this work, we present a general framework to determine the optimal architecture to couple the qubits of a quantum code. Here we assume that couplers can be introduced between qubits and we identify the coupling architecture that minimizes the total length of the couplers, rendering the physical implementation more practical. Our analysis is valid for any quantum code and we show that this set of optimization problems are identical to

Springer

well-known binary linear programs. We would like to point out that our choice of metric defining what we call an *optimal* scheme might turn out not to be the one that experimentalists will eventually consider relevant. Experimental and theoretical works with several qubits per transmission lines have just started, [2–5] and the results obtained from experiments will be crucial to determine which optimality criteria should be satisfied in practice. In this context, our work builds a methodology where a huge number of layouts can be studied and from which an optimal layout can be determined; this methodolody applies to a variety of metrics and not only to the precise choice we have made here. We thus believe that it represents a useful tool.

We apply our formalism to two celebrated quantum codes, namely the surface code [6, 7] and the Fibonacci Levin-Wen code [8, 9]. The former one is a planar version of Kitaev's toric code [10] that is among the most promising quantum computing platforms because of its simplicity and its surprisingly high error threshold of about 1%. The latter code is more involved but supports Fibonacci anyons that are universal for topological quantum computation; in other terms every quantum gate can be approximated to any accuracy by braiding Fibonacci anyons. Since the Fibonacci model is universal, it can hardly be simulated on a classical computer. In order to determine the error threshold of the Fibonacci Levin-Wen model, one would probably need to perform a full quantum simulation and thus have a quantum computer at hand. However, what precise algorithm should be run on the hypothetical quantum computer is still open and is a very interesting problem. Recently, some specific limiting cases where the Fibonacci model can be simulated classically have been investigated where the error threshold is approximatively 12.5 % [11].

We think that our work on the surface code is particularly timely since the first set of experiments to build small fragments of surface code (with 9 data qubits) have now started [12]. It is thus interesting to understand what architecture is optimal and could be realized in the lab. Finally we compare our results for the surface code with previously suggested architectures [13, 14].
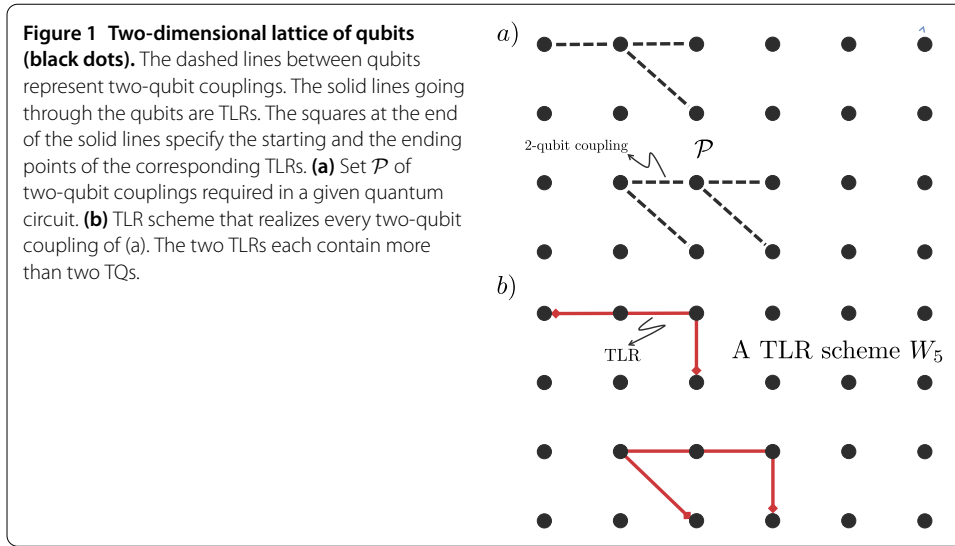
The paper is organized as follows. In Section 2.1 we present the physical model under consideration for a generic quantum code as well as the formalization of the optimization problem. In particular, we show that the optimal architecture is found by solving binary linear programs. In Section 3.1 we apply the formalism developed in Section 2.1 in order to find an optimal architecture for the Fibonacci code. In particular, we present a methodology to find scalable architectures by solving tractable binary linear programs. Section 3.2 finally contains our results for the surface code.

2 Connecting qubits optimally

In this work, we consider Transmon Qubits (TQs) and Transmission Line Resonators (TLRs) as the prototypical examples of physical qubits and moderate distance couplers [15, 16]. However, it is worth pointing out that our approach does not depend on the technological details of the implementation but can be applied to any kinds of qubits and couplers [17–23].

2.1 Model
Consider a set of N TQs $q_1, \ldots, q_N$ that lie on a two-dimensional plane at positions $\mathbf{x}_1, \ldots, \mathbf{x}_N$. Depending on the specific quantum codes that one wants to realize, see Sections 3 for examples, several TQs must interact with each other and thus be coupled

Figure 1 Two-dimensional lattice of qubits (black dots). The dashed lines between qubits represent two-qubit couplings. The solid lines going through the qubits are TLRs. The squares at the end of the solid lines specify the starting and the ending points of the corresponding TLRs. **(a)** Set $\mathcal{P}$ of two-qubit couplings required in a given quantum circuit. **(b)** TLR scheme that realizes every two-qubit coupling of (a). The two TLRs each contain more than two TQs.

through TLRs. As any quantum circuit can be reduced to a succession of single- and two-qubit operations [24], the most straightforward approach is to introduce TLRs containing each exactly two TQs; in this way TLRs realize the set $\mathcal{P}$ of all two-qubit couplings necessary to implement a given quantum circuit, see Figure 1.

However, using a new TLR for each pair of qubits that should be coupled to participate in two-qubit gate operations might not be the most optimal approach to this engineering problem. In fact TLRs are able to couple to more than two TQs and we assume that m individual TQs can reside inside the resonant cavity provided by the TLR. Each of the TQs can be controlled separately and coupled to any of the other $m-1$ TQs through the TLR. Following recent experimental progress [2], we find that $m \leq 5$ is a realistic upper bound. Informative micrographs of TLRs hosting several TQs can be found in [2] and on page 104 of [5] or in Figure 1(a) of [3]. Also, it seems natural to restrict the number p of TLRs that are connected to a single TQ; here we choose $p = 5$ [5]. We would like to mention here that X-mon qubits, a type of transmon qubit, are specifically designed so that they can couple to several TLRs, see Figure 1 in [12] for example where each branch of the X-mon can be coupled to a different TLR.

We call an unordered sequence of sites $i_k \in \{1,\ldots,N\}$ a string $\mathcal{S} = \{i_1, i_2, \ldots, i_m\}$. The length $|\mathcal{S}|$ of a string is defined by the number of sites it contains. To each string $\mathcal{S}$, we associate a number $\kappa_{\mathcal{S}} = 0,1$; if $\kappa_{\mathcal{S}} = 1$, then a TLR is present and hosts the m TQs $q_{i_1}, \ldots, q_{i_m}$, otherwise no single TLR hosts all those specific m qubits. We denote by $\mathfrak{S}_m$ the set of all strings $\mathcal{S}$ with $|\mathcal{S}| \leq m$. We call the vector

$$W_m = (\kappa_{\{1,2\}}, \ldots, \kappa_{\{N-1,N\}}; \kappa_{\{1,2,3\}}, \ldots; \kappa_{\{1,2,\ldots,m\}}, \ldots)^T \tag{1}$$

a TLR scheme. We say that a TLR corresponding to a string $\mathcal{S}$ is *included* in a TLR scheme W_m if $\kappa_{\mathcal{S}}$ is one of the elements of the vector W_m. We say that a TLR scheme W_m *contains* a TLR associated with string $\mathcal{S}$ if it is included and $\kappa_{\mathcal{S}} = 1$.

In order to formalize the concept of *optimal* TLR scheme, we introduce a cost $\mathcal{C}_{\mathcal{S}} \in \mathbb{R}$ associated with each string $\mathcal{S} \in \mathfrak{S}_m$. The cost vector of the scheme W_m is then

$$\mathcal{C}(W_m) = (\mathcal{C}_{\{1,2\}}, \ldots, \mathcal{C}_{\{N-1,N\}}, \mathcal{C}_{\{1,2,3\}}, \ldots, \mathcal{C}_{\{1,2,\ldots,m\}}, \ldots)^T. \tag{2}$$

Using this notation, the total cost of a given TLR scheme W_m is $\mathcal{C}(W_m)^T \cdot W_m$. The goal of this work is to determine one TLR scheme W_m that minimizes the cost and realizes the set $\mathcal{P}$ of two-qubit couplings. In Sections 3 we present concrete examples of cost functions for the surface code and for the Levin-Wen model.

The problem of finding the TLR scheme W_m that has minimal cost is solved by using standard binary linear optimization methods. The problem is formalized as follows: *Given a set of two-qubit connections $\mathcal{P}$, and given two integers m and p, find the TLR scheme W_m that minimizes the cost $\mathcal{C}(W_m)^T \cdot W_m$ such that*

1. *For all $i_j \in \{1, 2, \ldots, N\}$,*

$$\sum_{\mathcal{S} \in \mathfrak{G}_m | i_j \in \mathcal{S}} \kappa_{\mathcal{S}} \leq p. \tag{3}$$

2. *W_m realizes every two-qubit coupling of $\mathcal{P}$.*

It is worth pointing out again that the maximal number m of qubits per TLR, as well as the maximal number p of TLR per qubit, is fixed.

It is now clear why we call this a *binary* linear program; every component of the vector W_m is either 0 or 1. Solving such a binary linear program is generally very difficult and is in fact an NP-hard problem. However, specific instances of such problems can be tractable, and we give explicit examples below. As a side remark, note that when all the numbers in the program are allowed to be real, then the situation is dramatically simplified and the optimization problem can be solved in polynomial time.

In this work, we use the free software lpsolve, available at http://lpsolve.sourceforge.net/5.5/, to find the optimal solution to the binary linear program defined above. In order to simplify the program, we leave out all the *superfluous* TLRs. We call a TLR superfluous if it can be replaced by two (or more) TLRs that host no common qubits such that the same set of required two-qubit couplings is realized; one can thus always replace a superfluous TLR by two TLRs that will have a lower overall cost.

As mentioned in the Introduction, we aim to find the optimal architectures for two important quantum error correcting codes, namely the surface code and the Levin-Wen model. We find interesting that such quantum technological problems can be turned into standard optimization problems.

## 3 Application to Quantum error correcting codes

3.1 Fibonacci Levin-Wen model

Levin-Wen models are a class of spin systems defined on trivalent lattices whose excitations realize any consistent (Abelian or non-Abelian) anyonic theory [25]. Here we focus on a particular Levin-Wen model, namely the Fibonacci Levin-Wen model [8, 9]. Its name takes its origin in the nature of the excitations above the ground states; indeed they are Fibonacci anyons with topological charge τ and fusion rules

$$\tau \times \tau = 1 + \tau. \tag{4}$$

Here 1 represents the vacuum topological charge.

Figure 2 The Fibonacci Levin-Wen model is defined on a trivalent lattice. Each edge hosts a spin-1/2 particle (a so-called data qubit) depicted here by a black dot. **(a)** Vertex *v* where three edges of the lattice meet. The state of the three qubits at the vertex *v* is $|ijk\rangle$. **(b)** Twelve data qubits (black dots) needed to define the plaquette operator B_p on the trivalent lattice. In order to perform non-demolition measurements of vertex and plaquette operators, one introduces ancillary qubits (green squares). Here α_0 is used to measure B_p, while the remaining ancillary qubits α_{1-6} are used to measure the six vertex operators. This number of additional qubits is appropriate for the plaquette reduction method of Ref. [9].
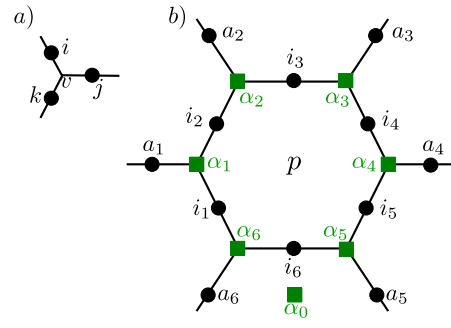
Considering a trivalent lattice with each edge carrying a spin-1/2 particle, we define the Fibonacci Levin-Wen Hamiltonian [8, 9],

$$H = -\sum_v Q_v - \sum_p B_p, \tag{5}$$

where Q_v and B_p are operators that are respectively associated with vertex *v* and plaquette *p* of the lattice, see Figure 2.

The vertex operator Q_v acts on the three qubits residing on the edges that meet at vertex *v*. If the states of the three qubits on theses edges are $|i\rangle$, $|j\rangle$, and $|k\rangle$, then we have

$$Q_v|ijk\rangle = \delta_{ijk}|ijk\rangle, \tag{6}$$

with

$$\delta_{ijk} = \begin{cases} 1 & \text{if } ijk = 000, 011, 101, 110, 111, \\ 0 & \text{otherwise.} \end{cases} \tag{7}$$

The plaquette operators are more complicated and involve 12-qubit interactions. Consider the twelve qubits a_{1-6} and i_{1-6} around a given plaquette *p*, see Figure 2(b). The plaquette operators are then defined through

$$B_p = \frac{1}{1+\phi^2}\left(B_p^1 + \phi B_p^\tau\right), \tag{8}$$

with $\phi = \frac{1+\sqrt{5}}{2}$ the golden ratio and

$$B_p^s|a_1,\ldots,a_6,i_1,\ldots,i_6\rangle = \sum_{i_1',\ldots,i_6'} B_{p,i_1,\ldots,i_6}^{s,i_1',\ldots,i_6'}(a_1,\ldots,a_6)|a_1,\ldots,a_6,i_1',\ldots,i_6'\rangle, \tag{9}$$

where $s = 1, \tau$ and

$$B_{p,i_1,\ldots,i_6}^{s,i_1',\ldots,i_6'}(a_1,\ldots,a_6) = F_{si_1'i_6'}^{a_1i_6i_1}F_{si_2'i_1'}^{a_2i_1i_2}\cdots F_{si_5'i_4'}^{a_5i_4i_5}F_{si_6'i_5'}^{a_6i_5i_6}. \tag{10}$$

For the Fibonacci theory we have [26]

$$F_\tau^{\tau\tau} = \begin{pmatrix} \phi^{-1} & \phi^{-1/2} \\ \phi^{-1/2} & -\phi^{-1} \end{pmatrix},$$

(11)

and all other F's are trivial. One can then show that the Levin-Wen plaquette and star operators satisfy $[B_p, Q_v] = [B_p, B_{p'}] = [Q_v, Q_{v'}] = 0$, for all v, v', p, p' [8, 9].

We define the Fibonacci code [8] $\mathcal{F}$ (an example of a stabilizer code) as the ground-state subspace of Hamiltonian (5), namely

$$\mathcal{F} = \{|\psi\rangle | Q_v|\psi\rangle = B_p|\psi\rangle = |\psi\rangle, \forall p, v\}.$$

(12)

On a surface with nontrivial topology this ground-state subspace of Hamiltonian (5) is degenerate and one uses this set of states to encode logical qubits. A nontrivial operation (a logical error) applied to the logical qubit is implemented by creating pairs of τ-excitations, braiding them, and annihilating them. The logical operation does not depend on the details of the braiding process, but only on its topology; this is in fact the main idea of topological quantum computation [27]. Importantly, Fibonacci anyons are universal for quantum computation and any quantum gate can thus be performed in a topologically protected fashion.

Recently, Ref. [9] has shown how to explicitly construct quantum circuits that measure plaquette and vertex operators of the Fibonacci Levin-Wen model; this is required to measure the error syndrome of $\mathcal{F}$ and to decide how to perform error correction. Here we go one step further and determine the optimal qubit-coupler architecture to realize those quantum circuits. It is not the goal of the present work to review in detail how vertex and plaquette quantum circuits are constructed. But these circuits indicate which qubits must be coupled and this indicates the binary linear program of Section 2 that is to be solved to obtain the optimal architecture. For the sake of completeness in Figure 3 we reproduce the circuit of Ref. [9] for the *plaquette reduction* method.

We note here that ancillary qubits are needed to perform non-demolition measurements of Q_v and B_p. According to the plaquette reduction method of Ref. [9], in Figure 2 the ancillary qubit α_0 is used to measure B_p, while the ancillary qubits α_{1-6} are used to measure the six vertex operators Q_v.
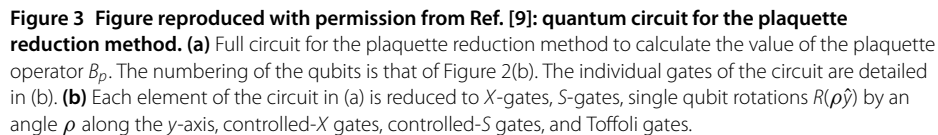
Here we choose the cost function $\mathcal{C}_\mathcal{S}$ that measures the geometric length of the TLR corresponding to $\mathcal{S} = \{i_1, i_2, \ldots, i_m\}$,

$$\mathcal{C}_\mathcal{S} = \min_\sigma \left\{ \sum_{k=2}^m |\mathbf{x}_{i_{\sigma(k)}} - \mathbf{x}_{i_{\sigma(k-1)}}| \right\},$$

(13)

where σ is a permutation of m elements.

Said differently, $\mathcal{C}_\mathcal{S}$ is the geometric length of the shortest path going through all the TQs specified in the string $\mathcal{S}$. In this work we thus look for the TLR scheme that minimizes the total length of the TLR wires.

Following the plaquette reduction method of Ref. [9] and using the notation of Figure 2, we present in Table 1 the set $\mathcal{P}_{\text{reduction}}$ of two-qubit couplings that are necessary to measure the six vertex operators and the single plaquette operator of Figure 2.

Figure 3 Figure reproduced with permission from Ref. [9]: quantum circuit for the plaquette reduction method. (a) Full circuit for the plaquette reduction method to calculate the value of the plaquette operator B_p. The numbering of the qubits is that of Figure 2(b). The individual gates of the circuit are detailed in (b). **(b)** Each element of the circuit in (a) is reduced to *X*-gates, *S*-gates, single qubit rotations $R(\rho \hat{y})$ by an angle ρ along the *y*-axis, controlled-*X* gates, controlled-*S* gates, and Toffoli gates.

Having in hand $\mathcal{P}_{\text{reduction}}$, we can solve the binary linear program of Section 2 and determine the optimal TLR scheme. The result is summarized in Table 2 and a pictorial representation is given in Figure 4.

For completeness, we also investigate the *plaquette swapping* method of Ref. [28] to measure plaquette operators. In this case, more ancillary qubits are required, see Figure 5. For the sake of completeness, in Figure 6 we reproduce the plaquette swapping circuit of Ref. [28]. The set $\mathcal{P}_{\text{swapping}}$ of two-qubit couplings required by the plaquette swapping method is summarized in Table 3. Again, we solve the binary linear program and find the optimal architecture of Table 4; here we have again chosen $m = p = 5$. As the pictorial representation would be too crowded, we refrain from drawing the TLRs corresponding to Table 4.

3.1.1 Scaling

While binary linear programs can be solved rapidly for a small number of qubits, as is the case for the 12 data qubits of Figures 2 and 5, the problem becomes rapidly unsolvable when we increase the number of qubits. This seems to be problematic as one wants to find the optimal architecture for a large Levin-Wen model and not only for a single plaquette. Fortunately, most of the time there is a lot of redundancy in the problem in the sense that a fundamental circuit unit can be identified and translated over the whole lattice. In fact, if one wants for example to measure all the vertex and plaquette operators of a large Fibonacci Levin-Wen model, the circuit will look the same around any plaquette of

Table 1 We list the set $\mathcal{P}_{\text{reduction}}$ of all the two-qubit couplings required to measure the plaquette *p* and the six vertex operators of Figure 2, following the plaquette reduction method of Ref. [9]

Primary qubit	Qubits to which the primary qubit couples
α_0	i_6
a_1	$a_6, i_1, i_2, i_6, \alpha_1$
a_2	$i_1, i_2, i_3, i_6, \alpha_2$
a_3	$i_2, i_3, i_4, i_6, \alpha_3$
a_4	$i_3, i_4, i_5, i_6, \alpha_4$
a_5	i_4, i_5, i_6, α_5
a_6	$a_1, i_1, i_2, i_6, \alpha_6$
i_1	$a_1, a_2, a_6, i_2, i_3, i_6, \alpha_1, \alpha_6$
i_2	$a_1, a_2, a_6, i_1, i_3, i_4, i_6, \alpha_1, \alpha_2$
i_3	$a_2, a_3, a_4, i_1, i_2, i_4, i_5, i_6, \alpha_2, \alpha_3$
i_4	$a_3, a_4, a_5, i_2, i_3, i_5, i_6, \alpha_3, \alpha_4$
i_5	$a_4, a_5, i_3, i_4, i_6, \alpha_4, \alpha_5$
i_6	$\alpha_0, a_1, a_2, a_3, a_4, a_5, a_6, i_1, i_2, i_3, i_4, i_5, \alpha_5, \alpha_6$
α_1	a_1, i_1, i_2
α_2	a_2, i_2, i_3
α_3	a_3, i_3, i_4
α_4	a_4, i_4, i_5
α_5	a_5, i_5, i_6
α_6	a_6, i_1, i_6

The data and ancillary qubits are labeled according to the notation of Figure 2.

Table 2 Information about the optimal TLR scheme obtained by solving the binary linear program for the measurement of the plaquette operator B_p and the six vertex operators Q_v of Figure 2, following the plaquette reduction method of Ref. [9]

Length of the TLR wire	Qubits contained inside the TLR wire
2.52 a.u.	$a_6, \alpha_0, i_6, \alpha_6, i_1$
4.73	i_6, a_6, a_1, i_2, a_2
3.15	$i_1, i_2, a_2, \alpha_2, i_3$
4.73	$i_2, i_3, \alpha_3, i_4, a_3$
3.15	i_3, a_3, a_4, i_5, i_6
3.15	$i_6, \alpha_5, a_5, i_5, i_4$
1.15	i_1, α_1, a_1
1.15	i_4, α_4, a_4
0.58	i_2, α_1
0.58	i_5, α_4

In particular, all the two-qubit couplings of $\mathcal{P}_{\text{reduction}}$ in Table 1 are realized: there are no more than five TQs in each TLR, and each TQ couples to maximally four TLRs. A pictorial representation of this TLR scheme and of the arbitrary unit (a.u.) is given in Figure 4.

the lattice. In such a scenario, it is possible to identify a small number of qubits that we couple optimally and that we translate to cover the whole lattice. The aim of this section is thus to introduce a simple method to optimally solve a given unit cell of the model that can be scaled up by simple translation to build a large two-dimensional lattice, see Figure 7. For the sake of simplicity, we just focus here on the plaquette swapping method of Ref. [28]. In Table 5 we present the set $\mathcal{P}_{\text{swapping}}^0$ of two-qubit couplings that are required between the qubits of unit cell 0, see Figure 7, and the remaining qubits of the lattice. If one would now straightforwardly solve the binary linear program for the unit cell, as we did in Section 3.1, then one would encounter the problem of *equivalent connections*, i.e., connections that are doubled due to the shifting of the unit cell. As an explicit example, let us consider the connection between qubits $i_{0,6}$ and $i_{2,5}$ as well as the connection between

Figure 4 Pictorial representation of the optimal architecture of Table 2. In **(a)** we show the TLRs that contain five TQs. The length of the arbitrary unit (a.u.) is depicted by the dashed arrow. In **(b)** we show the TLRs that contain three and two TQs.
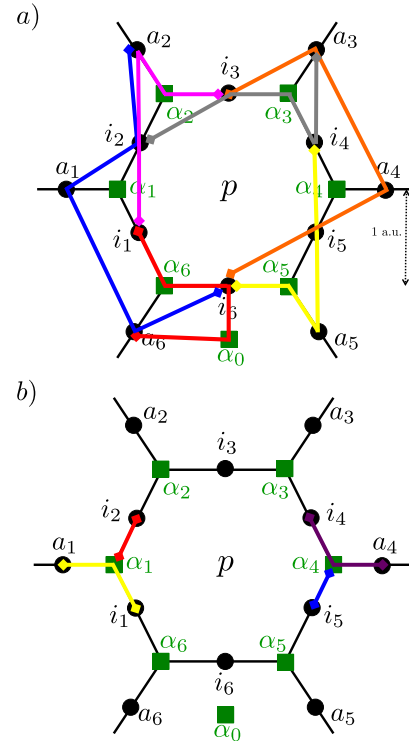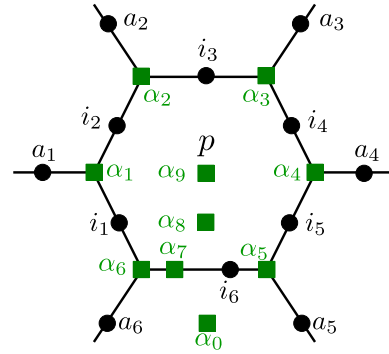

Figure 5 The data qubits of the Fibonacci Levin-Wen model are represented by black dots. This qubit layout is appropriate for the plaquette swapping method of Ref. [28]. The ancillary qubits, necessary to perform non-demolition measurements of plaquette and vertex operators, are represented by green squares.

qubits $i_{0,5}$ and $i_{5,6}$, see Figure 7. It is straightforward to see that after translating unit cell 0 onto unit cell 5, a TLR will be doubled. In order to avoid such doublings, one needs to slightly modify the algorithm as follows.

Consider a given unit cell 0 and two distinct strings $\mathcal{S}_1$ and $\mathcal{S}_2$ that each contains *at least one* site inside unit cell 0. We say that $\mathcal{S}_1 = \{i_1, i_2, \ldots, i_m\}$ and $\mathcal{S}_2 = \{j_1, j_2, \ldots, j_m\}$ are equivalent if $\forall k \in [1, m] \ \exists \ell \in [1, m]$ such that

$$\mathbf{x}_{i_k} = \mathbf{x}_{j_\ell} + \lambda_1 \mathbf{v}_1 + \lambda_2 \mathbf{v}_2, \tag{14}$$

where $\lambda_{1,2} \in \mathbb{Z}$ and $\mathbf{v}_{1,2}$ are basis vectors of the lattice, see Figure 7. If a TLR scheme possesses a TLR hosting the qubits along $\mathcal{S}_1$ and another TLR hosting the qubits along $\mathcal{S}_2$, it is clear that this will not be optimal. Indeed, when we translate unit cell 0 by the vector $\lambda_1 \mathbf{v}_1 + \lambda_2 \mathbf{v}_2$ and the associated TLRs, to cover the whole lattice, then some TLRs will

Table 3 The set $\mathcal{P}_{\text{swapping}}$ of all the two-qubit couplings required to measure the plaquette p and the six vertex operators of Figure 5, following the plaquette swapping method of Ref. [28]

Primary Qubit	Qubits to which the primary qubit couples
α_0	α_9
a_1	$i_1, i_2, \alpha_1, \alpha_7, \alpha_9$
a_2	$i_2, i_3, \alpha_2, \alpha_7, \alpha_9$
a_3	$i_3, i_4, \alpha_3, \alpha_7, \alpha_9$
a_4	$i_4, i_5, \alpha_4, \alpha_7, \alpha_9$
a_5	$i_5, i_6, \alpha_5, \alpha_7, \alpha_9$
a_6	$i_1, i_6, \alpha_6, \alpha_7, \alpha_8, \alpha_9$
i_1	$a_1, a_6, i_2, i_6, \alpha_1, \alpha_6, \alpha_7, \alpha_8, \alpha_9$
i_2	$a_1, a_2, i_1, i_3, \alpha_1, \alpha_2, \alpha_7, \alpha_9$
i_3	$a_2, a_3, i_2, i_4, \alpha_2, \alpha_3, \alpha_7, \alpha_9$
i_4	$a_3, a_4, i_3, i_5, \alpha_3, \alpha_4, \alpha_7, \alpha_9$
i_5	$a_4, a_5, i_4, i_6, \alpha_4, \alpha_5, \alpha_7, \alpha_9$
i_6	$a_5, a_6, i_1, i_5, \alpha_5, \alpha_6, \alpha_7, \alpha_8, \alpha_9$
α_1	a_1, i_1, i_2
α_2	a_2, i_2, i_3
α_3	a_3, i_3, i_4
α_4	a_4, i_4, i_5
α_5	a_5, i_5, i_6
α_6	a_6, i_1, i_6
α_7	$a_1, a_2, a_3, a_4, a_5, a_6, i_1, i_2, i_3, i_4, i_5, i_6, \alpha_8, \alpha_9$
α_8	$a_6, i_1, i_6, \alpha_7, \alpha_9$
α_9	$\alpha_0, a_1, a_2, a_3, a_4, a_5, a_6, i_1, i_2, i_3, i_4, i_5, i_6, \alpha_7, \alpha_8$

The data and ancillary qubits are labeled according to the notation of Figure 5.

Table 4 The optimal TLR scheme obtained by solving the binary linear program for the measurement of the plaquette operator B_p and the six vertex operators Q_v of Figure 5, following the plaquette swapping method of Ref. [28]

Length of the TLR wire	Qubits contained inside the TLR wire
2.52 a.u.	$\alpha_9, \alpha_8, i_6, \alpha_0, a_6$
2.84	$a_1, i_1, \alpha_7, \alpha_8, \alpha_9$
4.04	$\alpha_7, \alpha_9, i_4, i_3, a_3$
4.30	$\alpha_9, \alpha_7, a_5, i_5, a_4$
2.44	$i_1, \alpha_7, \alpha_6, i_6, a_6$
2.15	i_2, α_1, i_1, a_1
2.15	i_3, α_2, i_2, a_2
3.04	$\alpha_7, \alpha_9, i_2, a_2$
2.15	i_5, α_4, i_4, a_4
1.63	i_5, α_5, i_6, a_5
1.15	i_4, α_3, a_3
0.58	i_3, α_3

In particular, all the two-qubit couplings of $\mathcal{P}_{\text{swapping}}$ in Table 3 are realized: there are no more than five TQs in each TLR, and each TQ couples to maximally four TLRs.

be doubled. Having set these definitions, we present the steps that we follow to find the optimal scalable TLR without doubled TLRs.

- Consider the set $\mathcal{P}^0_{\text{swapping}}$ of two-qubit couplings that contains at least one qubit in the unit cell 0.
- Define $\mathfrak{W}$ as the set of TLR schemes that include all TLRs that are not superfluous with respect to $\mathcal{P}^0_{\text{swapping}}$ and all their equivalent TLRs.
- Out of every set of equivalent TLRs, choose one unique representative TLR. For each TLR scheme $W_m \in \mathfrak{W}$, define an associated TLR scheme B_m. This scheme B_m includes the same TLRs as W_m but contains the following TLRs: All TLRs that do not have an

Figure 6 Figure reproduced with permission from Ref. [28]. Quantum circuit for the plaquette swapping method of Ref. [28]. The qubits labeling is the one of Figure 5. The individual elements of the circuits can be found in Figure 3.
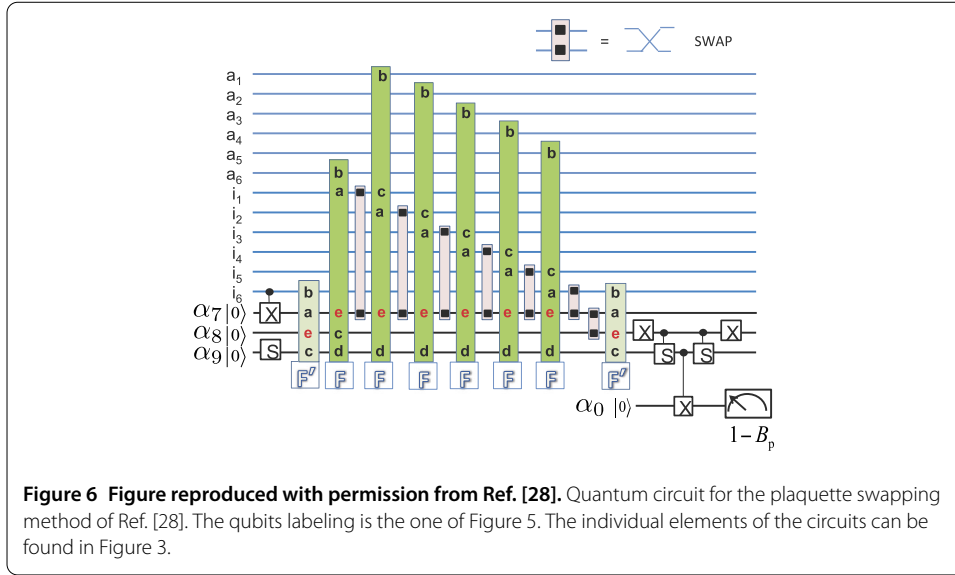

Figure 7 Trivalent lattice on which the Levin-Wen model is defined. The lattice is obtained by translating the unit cell 0 by the unit vectors $\mathbf{v}_1$ and $\mathbf{v}_2$. For example, the unit cell 1 is obtained by translating unit cell 0 by $-\mathbf{v}_2$. For the sake of clarity, we do not represent all the qubits of the lattice. Instead, we draw and label the data (black dots) and ancillary (green squares) qubits of a given unit cell k. This qubit layout is appropriate for the plaquette swapping method [28].
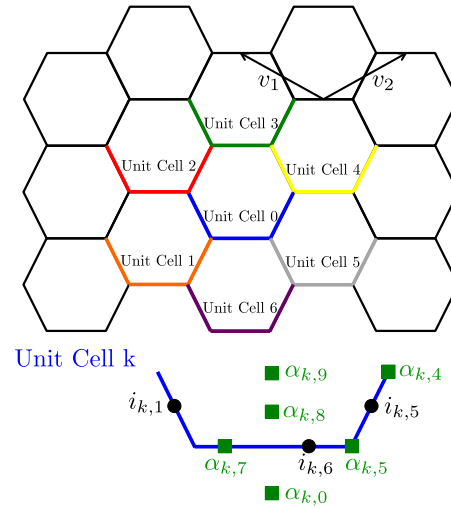
equivalent TLR and are contained in W_m as well as all representatives for which W_m contains at least one equivalent TLR. We call this new set of TLR schemes $\mathfrak{B}$.

- For each TLR scheme $B_m \in \mathfrak{B}$, define a new TLR scheme V_m that includes the same TLRs as B_m and contains all the TLRs that are contained in B_m as well as all equivalent TLRs.

- Perform the linear optimization over $\mathfrak{B}$ to find a TLR scheme B_m that minimizes the cost $C(B_m)^t \cdot B_m$ such that

  1. V_m realizes every two-qubit coupling of $\mathcal{P}^0_{\text{swapping}}$.

  2. For all $i_j \in \{1, 2, \ldots, N\}$,

$$\sum_{\mathcal{S} \in \mathfrak{S}_m | i_j \in \mathcal{S}} \kappa_{\mathcal{S}} \leq p. \tag{15}$$

Table 5 The set $\mathcal{P}^0_{\text{swapping}}$ of two-qubit connections between qubits in unit cell 0 of Figure 7 and the remaining qubits of the lattice, following the plaquette swapping method of Ref. [28]
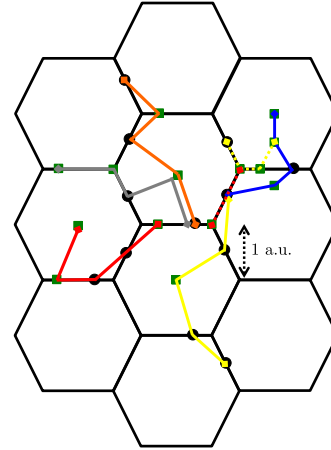
Qubit q in unit cell 0	Qubits to which q couples
$i_{0,1}$	$\alpha_{0,7}, i_{0,6}, \alpha_{0,8}, \alpha_{0,9}, i_{1,5}, \alpha_{1,4}, i_{1,6}, \alpha_{1,9}, \alpha_{2,7}, i_{2,5}, \alpha_{2,5}, i_{2,6}, \alpha_{2,9}, i_{6,6}, \alpha_{6,9}$
$i_{0,5}$	$\alpha_{0,7}, \alpha_{0,5}, \alpha_{0,4}, i_{0,6}, \alpha_{0,9}, \alpha_{4,7}, i_{4,1}, i_{4,6}, \alpha_{4,7}, \alpha_{4,9}, i_{5,1}, i_{5,6}, \alpha_{5,9}, i_{6,6}, \alpha_{6,9}$
$i_{0,6}$	$\alpha_{0,7}, i_{0,1}, i_{0,5}, \alpha_{0,8}, \alpha_{0,9}, i_{1,5}, \alpha_{2,7}, i_{2,5}, \alpha_{3,7}, i_{3,1}, i_{3,5}, \alpha_{4,7}, i_{4,1}, i_{5,1}$
$\alpha_{0,0}$	$\alpha_{0,9}$
$\alpha_{0,4}$	$i_{0,5}, \alpha_{4,7}, i_{4,1}$
$\alpha_{0,5}$	$\alpha_{0,7}, i_{0,5}, i_{5,1}$
$\alpha_{0,7}$	$i_{0,1}, i_{0,5}, \alpha_{0,5}, i_{0,6}, \alpha_{0,8}, \alpha_{0,9}, i_{1,5}, \alpha_{1,4}, i_{1,6}, \alpha_{1,9}, i_{5,1}, i_{5,6}, \alpha_{5,9}, i_{6,6}, \alpha_{6,9}$
$\alpha_{0,8}$	$\alpha_{0,7}, i_{0,1}, i_{0,6}, \alpha_{0,9}, i_{1,5},$
$\alpha_{0,9}$	$\alpha_{0,7}, i_{0,1}, i_{0,5}, \alpha_{0,0}, i_{0,6}, \alpha_{0,8}, i_{1,5}, \alpha_{2,7}, i_{2,5}, \alpha_{3,7}, i_{3,1}, i_{3,5}, \alpha_{4,7}, i_{4,1}, i_{5,1}$
4	$\alpha_{0,7}, i_{0,1}, \alpha_{0,8}, \alpha_{1,4}$
3	$i_{0,5}, \alpha_{0,5}, \alpha_{0,4}$

Table 6 Optimal TLR scheme for the set of couplings $\mathcal{P}^0_{\text{swpapping}}$, i.e., for couplings between qubits in the unit cell 0 and the rest of the lattice

Length of the TLR	Qubits contained inside the TLR wire
3.57 a.u.	$i_{0,6}, \alpha_{0,9}, i_{2,5}, \alpha_{3,7}, i_{3,1}$
3.15	$\alpha_{1,9}, \alpha_{1,7}, i_{1,6}, i_{1,5}, \alpha_{0,7}$
3.48	$\alpha_{2,7}, \alpha_{2,5}, i_{0,1}, \alpha_{0,9}, i_{0,6}$
2.52	$i_{0,5}, \alpha_{4,0}, i_{4,6}, \alpha_{4,8}, \alpha_{4,9}$
3.80	$i_{17,1}, i_{6,6}, \alpha_{6,9}, i_{5,1}, i_{0,5}$
1.44	$\alpha_{0,8}, \alpha_{0,7}, \alpha_{1,4}, i_{0,1}$
1.15	$\alpha_{0,4}, i_{0,5}, \alpha_{0,5}$

By translating this TLR scheme to the remaining unit cells of the lattice, one obtains an optimal TLR scheme for the entire lattice that avoids doubled TLRs.

Figure 8 Pictorial representation of the optimal scalable solution of Table 6. The TLRs couple qubits inside unit cell 0 to the rest of the lattice. This architecture can be translated to cover the whole lattice without generating doubled TLRs. The TLRs are represented by solid and dashed lines for clarity when they traverse the same path. The squares denote the starting and ending points of TLRs. Note that the four qubit TLR, that contains qubits $\alpha_{0,8}, \alpha_{0,7}, \alpha_{1,4}, i_{0,1}$ has not be drawn, but instead we have drawn it translated (yellow, dashed) for the sake of a clear figure.

Following the above algorithm, we find the scalable optimal architectures presented in Table 6 and Figure 8.

3.2 Surface code

The surface code [1] is a planar version of Kitaev's toric code [10] and represents arguably the most promising quantum computing architecture. It is thus justified to determine its optimal architecture using the simple formalism developed in this work, in particular be-
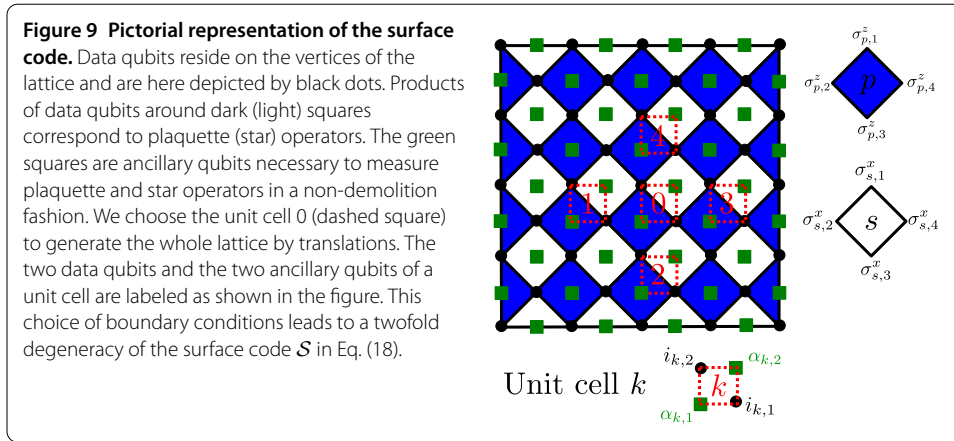
Figure 9 Pictorial representation of the surface code. Data qubits reside on the vertices of the lattice and are here depicted by black dots. Products of data qubits around dark (light) squares correspond to plaquette (star) operators. The green squares are ancillary qubits necessary to measure plaquette and star operators in a non-demolition fashion. We choose the unit cell 0 (dashed square) to generate the whole lattice by translations. The two data qubits and the two ancillary qubits of a unit cell are labeled as shown in the figure. This choice of boundary conditions leads to a twofold degeneracy of the surface code $\mathcal{S}$ in Eq. (18).

Table 7 The set $\mathcal{P}^0_{\text{surface}}$ of couplings between qubits of the unit cell 0 and the rest of the lattice to measure plaquette and star operators of the surface code, see Figure 9

Qubit q in unit cell 0	Qubits to which q couples
$i_{0,1}$	$\alpha_{0,1}, \alpha_{0,2}, \alpha_{3,1}, \alpha_{2,2}$
$\alpha_{0,1}$	$i_{0,1}, i_{0,2}, i_{1,1}, i_{2,2}$
$i_{0,2}$	$\alpha_{0,1}, \alpha_{0,2}, \alpha_{1,2}, \alpha_{4,1}$
$\alpha_{0,2}$	$i_{0,1}, i_{0,2}, i_{4,1}, i_{3,2}$

cause experimental groups are nowadays starting to build small fragments of the surface code.

Consider a square lattice with a spin-1/2 particle on each vertex, see Figure 9. We define the star operators A_s and plaquette operators B_p of the surface code as

$$A_s = \sigma^x_{s,1}\sigma^x_{s,2}\sigma^x_{s,3}\sigma^x_{s,4}, \tag{16}$$

$$B_p = \sigma^x_{p,1}\sigma^x_{p,2}\sigma^x_{p,3}\sigma^x_{p,4}, \tag{17}$$

where s and p label respectively light and dark squares of the lattice, see Figure 9. Note that plaquette and star operators at the boundaries are products of three qubit operators and not four as is the case in the bulk. Similar to the Fibonacci code, we define the surface code $\mathcal{S}$ as

$$\mathcal{S} = \big\{|\psi\rangle \, | \, A_s|\psi\rangle = B_p|\psi\rangle = |\psi\rangle, \forall p, s\big\}. \tag{18}$$

With the boundary conditions represented in Figure 9, the surface code is twofold degenerate and can thus encode a logical qubit [1]. Similar to the Fibonacci code, the surface code is a topological code and it is thus protected against local (static) perturbations. Its most striking property is its surprisingly high error threshold of about 1%, see Ref. [1] for a detailed review on this subject.

Here we do not review the construction of quantum circuits to measure star and plaquette operators of the surface code, rather, using the notation of Figure 9, in Table 7 we show the set $\mathcal{P}^0_{\text{surface}}$ of two-qubit couplings required to measure the eigenvalues of A_s and B_p in a scalable manner [29]. As was the case for the Fibonacci code, we need to introduce ancillary qubits to measure plaquette and star operators non-destructively.

Figure 10 Optimal scalable TLR scheme for the surface code. The problem is first solved for the couplings between qubits of unit cell 0 and the rest of the lattice, and then we have translated the result to cover the whole lattice. This results is equivalent to the one originally proposed in Ref. [13]. The TLRs are represented by solid lines and the squares denote the starting and ending points of TLRs. Note that rotating each individual four-qubit TLR by 90 degrees leads obviously to another optimal solution. The two-qubit TLRs on the edges and the three-qubit TLRs on the corners (purple) have been put by hand, since a full optimization solution of a smaller surface code lead to such a pattern of boundary two- and three-qubit TLRs. Indeed, it seems clear that this solution at the boundaries remains optimal for a larger surface code.
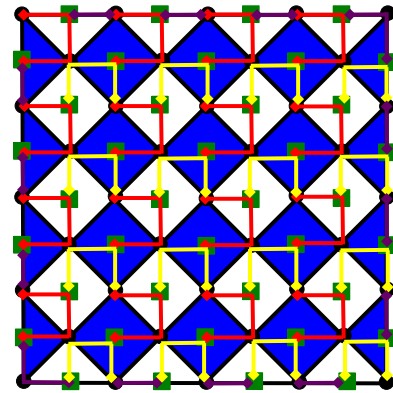
Solving the binary linear optimization problem with $m = 5$, we find the scalable optimal TLR scheme reported in Figure 10. The result is that each bulk TLR hosts four TQs and each bulk TQ is hosted by two TLRs. Interestingly, this result is the one originally proposed in Ref. [13], see also Ref. [14]. We point out that the optimal architecture of Figure 10 is clearly valid for the smallest possible surface code (consisting of 13 data qubits and 12 ancillary qubits) able to detect and correct a single physical qubit error. In fact, we have solved our optimization problem for this small surface code directly and obtained the solution of Figure 10 with two- and three-qubit TLRs at the boundaries.

3.2.1 Distance 5 surface code

As a final relevant explicit example, we consider a surface code that can correct two physical errors. While the surface code depicted in Figure 9 would contain 41 data qubits and 40 ancillary qubits in order to detect and cure two physical errors, there are simple methods to reduce the number of qubits while keeping the distance the same [30]. Such modifications are important for small-scale implementations of surface codes in a near future; indeed there is clearly an intention to realize a quantum code that requires the smallest possible amount of resource. Here we follow the approach of Ref. [30] and consider the rotated surface code of Figure 11. The qubits that are part of the rotated code reside inside the black square and some of the boundary ancillary qubits are also incorporated to measure the boundary stabilizers. As was shown in Ref. [30], such rotated surface code can correct two physical qubit errors although it possesses many fewer than 41 data qubits, in fact it consists only of 25 data qubits and 24 ancillary qubits. Furthermore, one can do slightly better by requiring not each stabilizer to have its individual ancillary qubit but instead by re-using an ancillary qubit to measure several stabilizer operators. We thus remove the 14 ancillary qubits with a (yellow) cross in Figure 11.

We can now solve the linear binary program and find the optimal architecture given in Figure 12.

4 Conclusions

In this work we have developed a methodology to find optimal architectures for quantum codes. Our starting point is to consider a two-dimensional lattice of transmon qubits that interact with each others over moderate distances by coupling them to transmission line resonators. For each layout, we define a cost that allows to designate an optimal scheme.

Figure 11 Pictorial representation of the rotated surface code. The black dots represent data qubits and the green squares ancillary qubits. The rotated surface code [30] contains all the qubits inside the solid square as well as the ancillary qubits surrounded by a black line. In this case, the rotated surface code contains 25 data qubits and 24 ancillary qubits; each ancillary qubit is used to measure exactly one stabilizer. However, one can slightly improve the resource needed by re-using ancillary qubits; we associate many of the ancillary qubits to more than one plaquette and star operator measurements. We thus remove the ancillary qubits with a yellow cross. Here we list which qubits replace the crossed qubits in the syndrome-computation circuit: qubit 2 is replaced by qubit 3, qubit 5 is replaced by qubit 6, qubit 7 is replaced by qubit 8, qubit 10 is replaced by qubit 11, qubit 13 is replaced by qubit 14, qubit 15 is replaced by qubit 16, qubit 17 is replaced by qubit 1, qubit 18 is replaced by qubit 3, qubit 19 is replaced by qubit 4, qubit 20 is replaced by qubit 12, qubit 21 is replaced by qubit 16, qubit 22 is replaced by qubit 14, qubit 23 is replaced by qubit 14, qubit 24 is replaced by qubit 6.
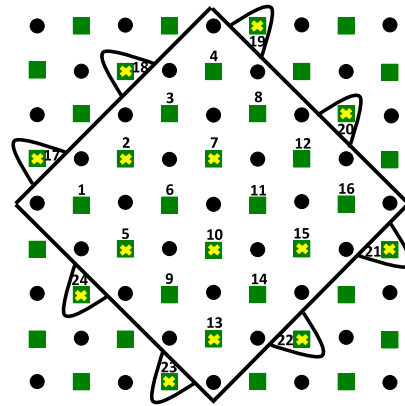
Figure 12 Optimal TLR scheme for the rotated surface code of Figure 11, able to detect and correct two physical qubit errors. The TLRs are represented by solid lines with starting and ending points depicted by squares.
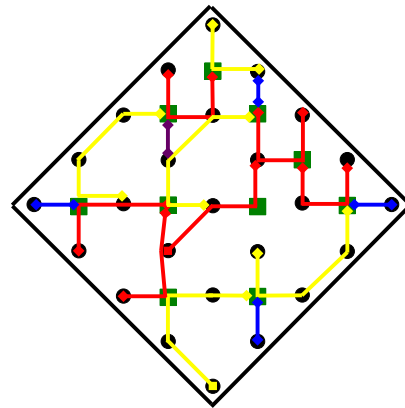
We show that finding such optimal scheme reduces to solve standard binary programs. What optimal means here depends obviously on the choice of a cost function. While we decided to choose to optimize over the total length of transmission line resonators for the Fibonacci and surface codes, our formalism is general enough to be straightforwardly applicable to many other codes and cost functions. While further experimental and theoretical studies will be necessary to determine which cost function will turn out to be most relevant in practice, we believe that our work represents a useful methodology to investigate a large number of layouts and find an optimal one according to some metric. In particular, we show how to apply our method to a restricted set of qubit and couplers that can be scaled up to a large two-dimensional structure.

References
1. Terhal BM. Quantum error correction for quantum memories. Rev Mod Phys. 2015;87:307.
2. Chow JM. Quantum information processing with superconducting qubits. PhD dissertation, Yale University. 2010. http://www.eng.yale.edu/rslab/papers/jmcthesis.pdf.
3. Ristè D, Poletto S, Huang M-Z, Bruno A, Vesterinen V, Saira O-P, DiCarlo L. Detecting bit-flip errors in a logical qubit using stabilizer measurements. Nat Commun. 2015;6:6983.
4. Yang C-P, Su Q-P, Zheng S-B, Nori F. Entangling superconducting qubits in a multi-cavity system. New J Phys. 2016;18:013025.
5. Ristè D. Feedback control of superconducting quantum circuits. PhD dissertation, TU Delft University of Technology. 2014. http://repository.tudelft.nl/view/ir/uuid%3A787bc488-96b2-4e03-abbe-0a9fba2c4d75/.
6. Bravyi SB, Kitaev AY. Quantum codes on a lattice with boundary. arXiv:quant-ph/9811052 (1998).
7. Freedman MH, Meyer DA. Projective plane and planar quantum codes. Found Comput Math. 2001;1:325-32.
8. Koenig R, Kuperberg G, Reichardt BW. Quantum computation with Turaev-Viro codes. Ann Phys. 2010;325:2707-49.
9. Bonesteel NE, DiVincenzo DP. Quantum circuits for measuring Levin-Wen operators. Phys Rev B. 2012;86:165113.
10. Kitaev AY. Fault-tolerant quantum computation by anyons. Ann Phys. 2003;303:2-30.
11. Burton S, Brell CG, Flammia ST. Classical simulation of quantum error correction in a Fibonacci anyon code. arXiv:1506.03815 (2015).
12. Kelly J, et al.. State preservation by repetitive error detection in a superconducting quantum circuit. Nature. 2015;519:66-9.
13. DiVincenzo DP. Fault-tolerant architectures for superconducting qubits. Phys Scr T. 2009;137:014020.
14. Ghosh J, Fowler AG, Geller MR. Surface code with decoherence: an analysis of three superconducting architectures. Phys Rev A. 2012;86:062318.
15. Blais A, Gambetta J, Wallraff A, Schuster DI, Girvin SM, Devoret MH, Schoelkopf RJ. Phys Rev A. 2007;75:032329.
16. DiCarlo L, Chow JM, Gambetta JM, Bishop LS, Johnson BR, Schuster DI, Majer J, Blais A, Frunzio L, Girvin SM, Schoelkopf RJ. Demonstration of two-qubit algorithms with a superconducting quantum processor. Nature. 2009;460:240-4.
17. Childress L, Sørensen AS, Lukin MD. Mesoscopic cavity quantum electrodynamics with quantum dots. Phys Rev A. 2004;69:042302.
18. Burkard G, Imamoglu A. Ultra-long-distance interaction between spin qubits. Phys Rev B. 2006;74:041307(R).
19. Trif M, Golovach VN, Loss D. Spin dynamics in InAs nanowire quantum dots coupled to a transmission line. Phys Rev B. 2008;77:045434.
20. Trifunovic L, Dial O, Trif M, Wootton JR, Abebe R, Yacoby A, Loss D. Long-distance spin-spin coupling via floating gates. Phys Rev X. 2012;2:011006.
21. Shulman MD, Dial OE, Harvey SP, Bluhm H, Umansky V, Yacoby A. Demonstration of entanglement of electrostatically coupled singlet-tripel qubits. Science. 2012;336:202-5.
22. Trifunovic L, Pedrocchi FL, Loss D. Long-distance entanglement of spin qubits via ferromagnet. Phys Rev X. 2013;3:042023.
23. Hassler F, Catelani G, Bluhm H. Exchange-interaction of two spin qubits mediated by a superconductor. arXiv:1509.0638 (2015).
24. Nielsen MA, Chuang IL. Quantum computation and quantum information. Cambridge: Cambridge University Press; 2000.
25. Levin MA, Wen X-G. String-net condensation: a physical mechanism for topological phase. Phys Rev B. 2005;71:045110.
26. Bonderson PH. Non-Abelian anyons and interferometry. PhD dissertation, California Institute of Technology. 2007. http://resolver.caltech.edu/CaltechETD:etd-06042007-101617.
27. Nayak C, Simon SH, Stern A, Freedman M, Das Sarma S. Non-Abelian anyons and topological quantum computation. Rev Mod Phys. 2008;80:1083-159.
28. Feng W, Bonesteel NE, DiVincenzo DP. Non-Abelian errors in the Fibonacci Levin-Wen model. In preparation.
29. Dennis E, Kitaev A, Landahl A, Preskill J. Topological quantum memory. J Math Phys. 2002;43:4452-505.
30. Horsman C, Fowler AG, Devitt S, Van Meter R. Surface code quantum computing by lattice surgery. New J Phys. 2012;14:123011.