# Fault tolerance for holonomic quantum computation

Ognyan Oreshkov[(1)], Todd A. Brun[(2)], and Daniel A. Lidar[(3)]
[(1)] *Universitat Autònoma de Barcelona, Spain*
*oreshkov@ifae.es*
[(2)] *University of Southern California, USA*
*brun@usc.edu*
[(3)] *University of Southern California, USA*
*lidar@usc.edu*

# Contents

# 1

# Fault tolerance for holonomic quantum computation

In Chapter **??** it was shown how holonomic quantum computation (HQC) can be combined with the method of decoherence-free subspaces (DFSs), leading to passive protection against certain types of correlated errors. However, this is not enough for fault tolerance since other types of errors can accumulate detrimentally unless corrected. Scalability of HQC therefore requires going beyond that scheme, e.g., by combining the holonomic approach with *active* error correction. One way of combining HQC with active quantum error-correcting codes, which is similar to the way HQC is combined with DFSs, was also mention in Chapter **??**. This approach, however, is not scalable since it requires Hamiltonians that commute with the stabilizer elements. As the code grows in size in order to protect against a larger number of errors, the weight of the interactions used for computation would also have to increase.

In this chapter, we will show how HQC can be made fault tolerant by combining it with the techniques for fault-tolerant quantum error correction (FTQEC) on stabilizer codes. The fact that the holonomic method can be mapped directly to the circuit model allows us to construct procedures which resort almost entirely to these techniques. We will discuss two approaches—one which makes use of the encoding already present in a stabilizer code and does not require additional qubits [27, 30], and another one which requires ancillary qubits for implementing transversal operations between qubits in the code [26].

Since protected information is contained in subsystems [18, 38] (see Chapter **??**), the problem of implementing fault-tolerant HQC can be understood as that of manipulating fault-tolerantly the subsystem containing the protected information by holonomic means. We therefore begin by first introducing a generalization of the standard HQC method, which is applicable to encoding in subsystems.

### 1.1 Holonomic quantum computation on subsystems

As pointed out in Chapter **??**, protected quantum information is most generally contained in the subsystems $\mathcal{H}_i^A$ in a decomposition of the Hilbert space of the system of the form

$$\mathcal{H}^{\mathcal{S}} = \bigoplus_{i=1}^{m} \mathcal{H}_i^A \otimes \mathcal{H}_i^B. \tag{1.1}$$

Here the dimensions of the subsystems are related by $\dim\mathcal{H}^S = \sum_{i=1}^{m} \dim\mathcal{H}_i^A \times \dim\mathcal{H}_i^B$. In the formalism of operator quantum error correction [15, 16, 14, 28], the subsystems $\mathcal{H}_i^A$ are used for encoding of the protected information, and $\mathcal{H}_i^B$ are referred to as *gauge* subsystems. In the most general case, all subsystems $\mathcal{H}_i^A$ are used for encoding and computation [3]. In Ref. [26] it was demonstrated that by the use of holonomies it is possible to apply arbitrary computations in the subsystems $\mathcal{H}_i^A$. This result is summarized in the following theorem.

**Theorem 1** . *Consider a non-trivial decomposition into subsystems of the form* (1.1). *Choose an initial Hamiltonian in the form*

$$H(0) = \sum_{i=1}^{m} I_i^A \otimes H_i^B, \tag{1.2}$$

*where $H_i^B$ are non-degenerate operators on $\mathcal{H}_i^B$ with different eigenvalues for $i \neq j$. By varying adiabatically this Hamiltonian along suitable loops in a sufficiently large control manifold, it is possible to generate a unitary of the form*

$$U = \sum_{i=1}^{m} W_i^A \otimes V_i^B, \tag{1.3}$$

*where $\{W_i^A\}$ is an arbitrary set of geometric transformations on $\{\mathcal{H}_i^A\}$.*

To proof of the theorem is based on the following lemma.

**Lemma 1** . *By varying a Hamiltonian adiabatically along suitable loops in a sufficiently large control manifold, it is possible to implement holonomically any combination of unitary transformations in its eigenspaces.*

*Comment.* As discussed in Chapter **??**, if we have sufficient control over the parameters of a Hamiltonian, we can generate holonomically any unitary operation in a given eigenspace. This lemma concerns

the question of whether it is possible to generate any *combination* of holonomies in the different eigenspaces. Although intuitively expected based on considerations concerning the generic irreducibility of the adiabatic connection, the property may not be obvious. For example, in the case of a two-level Hamiltonian, the evolution of one of the eigenspaces completely determines the evolution of the other one, which raises the question of whether it is possible to obtain independent holonomies in the two eigenspaces. We now show that this is possible. Note that even though the proof is constructive and can serve as a general prescription for simultaneous HQC in different eigenspaces, it is primarily meant as a proof of principle. In Sec. 1.3.1 we will see that depending on the task one can find more efficient constructions.

*Proof of Lemma 1.* It is sufficient to show that it is possible to generate an arbitrary operation in any given eigenspace while at the same time generating the identity operation in the rest of the eigenspaces. Without loss of generality, we will assume that there are only two eigenspaces; if there are more, we can operate within the subspace spanned by two of them at a time, by varying only the restriction of the Hamiltonian on that subspace. Then the initial Hamiltonian can be written as

$$H(0) = \varepsilon_1 \Pi_1 + \varepsilon_2 \Pi_2, \tag{1.4}$$

where $\Pi_{1,2}$ are the projectors on the ground and excited eigenspaces, and $\varepsilon_1 < \varepsilon_2$ are their corresponding eigenvalues. Notice that this Hamiltonian is invariant under unitary transformations of the form

$$V = V_1 \oplus V_2, \tag{1.5}$$

where $V_{1,2}$ are unitaries on the subspaces with projectors $\Pi_{1,2}$, respectively. Let the Hamiltonian vary along a loop $H(t)$, $H(0) = H(T)$, which satisfies the adiabatic requirement to some satisfactory precision. To this precision, the resulting unitary transformation can be written as

$$U(T) = \mathcal{T}\exp(-i \int_0^T dt H(t)) = e^{-i\omega_1} U_1 \oplus e^{-i\omega_2} U_2, \tag{1.6}$$

where $U_1$ and $U_2$ are the holonomies resulting in the two eigenspaces, and $\omega_{1,2} = \int_0^T dt \varepsilon_{1,2}(t)$ are dynamical phases. Observe that the Hamiltonian $VH(t)V^\dagger$, where $V = V_1 \oplus V_2$, is a valid loop based on $H(0)$ with the same spectrum as that of $H(t)$, which gives rise to the holonomies $V_1 U_1 V_1^\dagger$ and $V_2 U_2 V_2^\dagger$, respectively. This follows from the fact that the overall unitary transformation generated by $VH(t)V^\dagger$ is equal to $VU(t)V^\dagger$ where $U(t)$ is the unitary generated by $H(t)$.

Imagine that we want to generate holonomically the unitary transformation $W_1$ in the ground space of the Hamiltonian while at the same time obtaining the identity holonomy $I_2$ in the excited space. Choose any loop $H(t)$ which gives rise to the holonomy $W_1^{\frac{1}{d_2}}$ in the ground space, where $d_2$ is the dimension of the excited space (we know that such a loop can be found). Let this loop result in the holonomy $W_2$ in the excited space. The latter can be written as $W_2 = \sum_{j=1}^{d_2} e^{i\alpha_j}|j\rangle\langle j|$, where $\{|j\rangle\}$ is an eigenbasis of $W_2$ and $e^{i\alpha_j}, \alpha_j \in R$, are the corresponding eigenvalues. Consider the unitary transformation $C_2$ which cyclicly permutes the eigenvectors $\{|j\rangle\}$: $C_2|j\rangle = |j+1\rangle$, where we define $|d_2+1\rangle \equiv |1\rangle$. We can implement the desired combination of holonomies in the two eigenspaces as follows. First apply $H(t)$. This results in the holonomies $W_1^{\frac{1}{d_2}}$ and $W_2$ in the ground and excited spaces, respectively. Next, apply $(I_1 \oplus C_2)H(t)(I_1 \oplus C_2)^\dagger$. This generates the holonomies $W_1^{\frac{1}{d_2}}$ and $C_2 W_2 C_2^\dagger = \sum_{j=1}^{d_2} e^{i\alpha_{j-1}}|j\rangle\langle j|$ (we have defined $\alpha_{1-1} \equiv \alpha_{d_2}$). The combined effect of these two operations is $W_1^{\frac{2}{d_2}}$ and $\sum_{j=1}^{d_2} e^{i(\alpha_j+\alpha_{j-1})}|j\rangle\langle j|$. We next apply $(I_1 \oplus C_2^2)H(t)(I_1 \oplus C_2^2)^\dagger$, which generates the holonomies $W_1^{\frac{1}{d_2}}$ and $C_2^2 W_2 C_2^{2\dagger} = \sum_{j=1}^{d_2} e^{i\alpha_{j-2}}|j\rangle\langle j|$. The net result becomes $W_1^{\frac{3}{d_2}}$ and $\sum_{j=1}^{d_2} e^{i(\alpha_j+\alpha_{j-1}+\alpha_{j-2})}|j\rangle\langle j|$. We continue this for a total of $d_2$ rounds, which results in the net holonomic transformations $W_1^{\frac{d_2}{d_2}} = W_1$ and $e^{i(\alpha_1+\alpha_2+\ldots+\alpha_{d_2})}\sum_{j=1}^{d_2}|j\rangle\langle j| \propto I_2$. This completes the proof.

*Proof of Theorem 1.* Let us denote the eigenvalues of $H_i^B$ in Eq. (1.2) by $\omega_{\alpha_i}$ where $\alpha_i = 1, 2, ..., \dim\mathcal{H}_i^B$, and their corresponding eigenvectors by $|\alpha_i\rangle \in \mathcal{H}_i^B$. Then the spectral decomposition of the initial Hamiltonian reads $H(0) = \sum_{i=1}^m \sum_{\alpha_i=1}^{\dim\mathcal{H}_i^B} \omega_{\alpha_i}\Pi_i^A \otimes |\alpha_i\rangle\langle\alpha_i|$, where $\Pi_i^A$ is the projector on $\mathcal{H}_i^A$. According to Lemma 1, we can implement holonomically any combination of unitary transformations in the different eigenspaces of $H(0)$ up to an overall phase. If we want to implement the set of unitary operations $\{W_i^A\}$ in the different subsystems $\mathcal{H}_i^A$, we can do this by implementing the holonomy $W_i^A$ in each of the eigenspaces $\mathcal{H}_i^A \otimes |\alpha_i\rangle$

for $\alpha_i = 1, 2, ..., \dim\mathcal{H}_i^B$. This results in the net unitary

$$U = \bigoplus_i W_i^A \otimes (\sum_{\alpha_i} e^{i\phi_{\alpha_i}} |\alpha_i\rangle\langle\alpha_i|) \equiv \bigoplus_i W_i^A \otimes V_i^B, \qquad (1.7)$$

where $e^{i\phi_{\alpha_i}}$ are the overall phases (dynamical plus geometric) resulting in each eigenspace $\mathcal{H}_i^A \otimes |\alpha_i\rangle$. This completes the proof.

To summarize, HQC on a subsystem can be realized by adiabatically varying a Hamiltonian which acts locally on the corresponding co-subsystem. During the evolution, the information initially encoded in the subsystem transforms to a different subsystem which is related to the initial one via a geometric unitary operation. The dynamical part of the unitary factors out as a transformation on the correspondingly transformed co-subsystem. The problem of FTHQC can be understood as that of finding a fault-tolerant realization of this approach.

## 1.2 FTHQC on stabilizer codes without additional qubits

### *1.2.1 The main idea*

The developed techniques for FTQEC on stabilizer codes provide a prescription for how to encode information and how to transform the subsystem containing the information so that the class of errors for which the code is designed remains correctable. We will try to find realizations of the same transformations that the encoded subsystem follows in a standard fault-tolerant scheme using the generalized method of HQC on subsystems. There are two difficulties in this respect that have to be considered. First, not every unitary evolution of a subsystem can be realized by holonomic means. For example, a general evolution inside a fixed subsystem cannot be implemented holonomically because the HQC method requires that the encoded states leave the original code in order to undergo non-trivial geometric transformations. Second, the holonomic approach unavoidably gives rise to dynamical transformations on the co-subsystem (see below), and these could jeopardize the fault tolerance of the scheme. We will see that neither of these features is a fundamental obstacle to the realization of fault-tolerant HQC.

The standard fault-tolerant techniques are primarily based on the use of transversal operations. In addition, there are non-transversal operations for the preparation and verification of a special ancillary state such as Shor's "cat" state $(|00...0\rangle + |11...1\rangle)/\sqrt{2}$. Since single-qubit unitaries together with the C-NOT gate form a universal set of gates, fault-tolerant computation can be realized entirely in terms of single-qubit

gates and transversal C-NOT gates, assuming that the ancillary state can be prepared reliably. Our goal will thus be to construct holonomic gates on the code space via transformations that in the original basis of the full Hilbert space are equivalent to transversal one- and two-qubit gates or to operations for the preparation of the ancillary state.

It turns out, however, that holonomic transformations on the code subsystem cannot be realized using purely transversal operations without the use of extra qubits, even if the encoded gate has a purely transversal implementation in the dynamical case. This is because the Hamiltonians that leave the code subsystem invariant are linear combinations of elements of the stabilizer or the gauge group of the code and they unavoidably couple qubits in the same block. But transversal operations are not the most general class of operations that do not lead to propagation of errors. A transformation which at every moment is equal to a transversal operation followed by a syndrome-preserving transformation on the co-factor of the subsystem that contains the protected information, is also fault-tolerant. In fact, any transversal operation in a given fault-tolerant protocol can be safely substituted by an operation of the latter type. It is this latter type of transformations by which we can realize fault-tolerant HQC without the use of extra qubits. We will show that by choosing as a starting Hamiltonian a suitable element of the stabilizer or the gauge group of the code, and varying this Hamiltonian along appropriately chosen paths in parameter space, we can generate operations that from the point of view of the full Hilbert space transform both the ground and the excited spaces via the same transversal operation. Such transformations are exactly of the type we described—the relative dynamical phase that accumulates between the ground and excited spaces is equal to a phase on the co-subsystem which does not affect the fault-tolerance of the scheme since it is either projected out when a measurement of the syndrome is performed, or is equivalent to a gauge transformation. By an appropriate sequence of such transformations we can generate a holonomy equal to any desired encoded gate.

For concreteness, we will consider a $[[n, 1, r, 3]]$ stabilizer code. This is a code that encodes 1 qubit into $n$, has $r$ gauge qubits, and can correct arbitrary single-qubit errors. As we saw in the previous section, in order to apply holonomic transformations on the subsystem that contains the logical information, we need a nontrivial starting Hamiltonian which leaves this subsystem invariant. It is easy to verify that the only Hamiltonians that satisfy this property are linear combinations of the elements of the stabilizer, or the gauge group.

Note that the stabilizer and the gauge group transform during the course of the computation under the operations being applied. At any stage when we complete an encoded operation, they return to their initial forms. During the implementation of a standard encoded gate, the Pauli group $G_n$ on a given codeword may spread over other codewords, but it can be verified that this spreading can be limited to at most 4 other codewords counting the "cat" state. This is because the encoded C-NOT gate can be implemented fault-tolerantly on any stabilizer code by a transversal operation on 4 encoded qubits [10], and any encoded Clifford gate can be realized using only the encoded C-NOT provided that we are able to do fault-tolerant measurements (the encoded Clifford group is generated by the encoded Hadamard, Phase and C-NOT gates). Encoded gates outside of the Clifford group, such as the encoded $\pi/8$ or Toffoli gates, can be implemented fault-tolerantly using encoded C-NOT gates conditioned on the qubits in a "cat" state, so they may require transversal operations on a total of 5 blocks. For CSS codes, however, the spreading of the Pauli group of one block during the implementation of a basic encoded operation can be limited to a total of 3 blocks, since the encoded C-NOT gate has a transversal implementation [10].

It also has to be pointed out that fault-tolerant encoded Clifford operations can be implemented using only Clifford gates on the physical qubits [10]. These operations transform the stabilizer and the gauge group into subgroups of the Pauli group, and their elements remain in the form of tensor products of Pauli matrices. The fault-tolerant implementation of encoded gates outside of the Clifford group, however, involves operations that take these groups outside of the Pauli group. We will, therefore, consider separately two cases—encoded operations in the Clifford group, and encoded operations outside of the Clifford group.

### *1.2.2 Encoded operations in the Clifford group*

#### *1.2.2.1 Single-qubit unitary operations*

For applying operations on a given qubit, say, the first one, we will use as a starting Hamiltonian an element of the stabilizer (with a minus sign) or the gauge group of the code, that acts non-trivially on that qubit. Since we are considering codes that can correct arbitrary single-qubit errors, one can always find an element of the initial stabilizer or the initial gauge group that has a factor $\sigma_0 = I$, $\sigma_1 = X$, $\sigma_2 = Y$ or $\sigma_3 = Z$

acting on the first qubit, i.e.,

$$\widehat{G} = \sigma_i \otimes \widetilde{G}, \quad i = 0, 1, 2, 3 \tag{1.8}$$

where $\widetilde{G}$ is a tensor product of Pauli matrices and the identity on the rest $n-1$ qubits. It can be verified that under Clifford gates the stabilizer and the gauge group transform in such a way that this is always the case except that the factor $\widetilde{G}$ may spread on qubits in other blocks. We can assume that the stabilizer spreads on at most 5 blocks including the "cat" state, since this is sufficient to implement any encoded operation. Henceforth, we will use "hat" to denote operators on all qubits on which the stabilizer spreads, and "tilde" to denote operators on all of these qubits excluding the first one.

Without loss of generality we will assume that the chosen stabilizer or gauge-group element for that qubit has the form

$$\widehat{G} = Z \otimes \widetilde{G}. \tag{1.9}$$

As initial Hamiltonian, we will take the operator

$$\widehat{H}(0) = -\widehat{G} = -Z \otimes \widetilde{G}. \tag{1.10}$$

**Proposition 1** *If the initial Hamiltonian* (1.10) *is varied adiabatically so that only the factor acting on the first qubit changes,*

$$\widehat{H}(t) = -H(t) \otimes \widetilde{G}, \tag{1.11}$$

*where*

$$Tr\{H(t)\} = 0, \tag{1.12}$$

*the transformation that each of the eigenspaces of this Hamiltonian undergoes will be equivalent to that driven by a local unitary on the first qubit,* $\widehat{U}(t) \approx U(t) \otimes \widetilde{I}$.

*Proof.* Observe that (1.11) can be written as

$$\widehat{H}(t) = H(t) \otimes \widetilde{P}_0 - H(t) \otimes \widetilde{P}_1, \tag{1.13}$$

where

$$\widetilde{P}_{0,1} = \frac{\widetilde{I} \mp \widetilde{G}}{2} \tag{1.14}$$

are orthogonal complementary projectors. The evolution driven by $\widehat{H}(t)$ is therefore

$$\widehat{U}(t) = U_0(t) \otimes \widetilde{P}_0 + U_1(t) \otimes \widetilde{P}_1, \tag{1.15}$$

where

$$U_{0,1}(t) = \mathcal{T}\exp(-i\int_0^t \pm H(\tau)d\tau). \tag{1.16}$$

Let $|\phi_0(t)\rangle$ and $|\phi_1(t)\rangle$ be the instantaneous ground and excited states of $H(t)$ with eigenvalues $E_{0,1}(t) = \mp E(t)$ ($E(t) > 0$). Then in the adiabatic limit we have

$$U_{0,1}(t) = e^{i\omega(t)}U_{A_{0,1}}(t) \oplus e^{-i\omega(t)}U_{A_{1,0}}(t), \tag{1.17}$$

where $\omega(t) = \int_0^t d\tau E(\tau)$ and

$$U_{A_{0,1}}(t) = e^{\int_0^t d\tau \langle\phi_{0,1}(\tau)|\frac{d}{d\tau}|\phi_{0,1}(\tau)\rangle}|\phi_{0,1}(t)\rangle\langle\phi_{0,1}(0)|. \tag{1.18}$$

The projectors on the ground and excited eigenspaces of $\widehat{H}(0)$ are

$$\widehat{P}_0 = |\phi_0(0)\rangle\langle\phi_0(0)| \otimes \widetilde{P}_0 + |\phi_1(0)\rangle\langle\phi_1(0)| \otimes \widetilde{P}_1 \tag{1.19}$$

and

$$\widehat{P}_1 = |\phi_1(0)\rangle\langle\phi_1(0)| \otimes \widetilde{P}_0 + |\phi_0(0)\rangle\langle\phi_0(0)| \otimes \widetilde{P}_1, \tag{1.20}$$

respectively. Using Eq. (1.17) and Eq. (1.18), one can see that the effect of the unitary (1.15) on each of these projectors is

$$\widehat{U}(t)\widehat{P}_0 = e^{i\omega(t)}(U_{A_0}(t) \oplus U_{A_1}(t)) \otimes \widetilde{I}\,\widehat{P}_0, \tag{1.21}$$

$$\widehat{U}(t)\widehat{P}_1 = e^{-i\omega(t)}(U_{A_0}(t) \oplus U_{A_1}(t)) \otimes \widetilde{I}\,\widehat{P}_1, \tag{1.22}$$

i.e, up to an overall dynamical phase its effect on each of the eigenspaces is the same as that of the unitary

$$\widehat{U}(t) = U(t) \otimes \widetilde{I}, \tag{1.23}$$

where

$$U(t) = U_{A_0}(t) \oplus U_{A_1}(t). \tag{1.24}$$

This completes the proof.

We next show how by suitably choosing $H(t)$ we can implement all necessary single-qubit gates. We will identify a set of points in parameter space, such that by interpolating between these points we can draw various paths resulting in the desired transformations. We remark that if a path does not form a loop, the resulting geometric transformation (1.24) is an open-path holonomy [12].

Consider the single-qubit unitary operator

$$V^{\theta\pm} = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & \mp e^{-i\theta} \\ \pm e^{i\theta} & 1 \end{pmatrix}, \tag{1.25}$$

where $\theta$ is a real parameter. Note that $V^{\theta\mp} = (V^{\theta\pm})^\dagger$. Define the following single-qubit Hamiltonian:

$$H^{\theta\pm} \equiv V^{\theta\pm} Z V^{\theta\mp}. \tag{1.26}$$

Let $H(t)$ in Eq. (1.11) be a Hamiltonian which interpolates between $H(0) = Z$ and $H(T) = H^{\theta\pm}$ (up to a factor) as follows:

$$H(t) = f(t)Z + g(t)H^{\theta\pm} \equiv H^{\theta\pm}_{f,g}(t), \tag{1.27}$$

where $f(0), g(T) > 0$, $f(T) = g(0) = 0$. To simplify our notations, we will drop the indices $f$ and $g$ of the Hamiltonian, since the exact form of these functions is not important for our analysis as long as they are sufficiently smooth (see discussion in Sec. 1.2.2.2). This Hamiltonian has eigenvalues $\pm\sqrt{f(t)^2 + g(t)^2}$ and its energy gap is non-zero unless the entire Hamiltonian vanishes.

It can be shown that in the adiabatic limit, the Hamiltonian (1.11) with $H(t) = H^{\theta\pm}(t)$ gives rise to the geometric transformation

$$\widehat{U}^{\theta\pm}(T) = V^{\theta\pm} \otimes \widetilde{I}. \tag{1.28}$$

Details of the proof can be found in Ref. [29].

We will use this result, to construct a set of standard gates by sequences of operations of the form $V^{\theta\pm}$, which can be generated by interpolations of the type (1.27) run forward or backward. For single-qubit gates in the Clifford group, we will only need three values of the parameter $\theta$: 0, $\pi/2$ and $\pi/4$. For completeness, however, we will also demonstrate how to implement the $\pi/8$ gate, which together with the Hadamard gate is sufficient to generate any single-qubit unitary transformation [34]. For this we will need $\theta = \pi/8$. Note that

$$H^{\theta\pm} = \pm(\cos\theta X + \sin\theta Y), \tag{1.29}$$

so for these values of $\theta$ we have $H^{0\pm} = \pm X$, $H^{\pi/2\pm} = \pm Y$, $H^{\pi/4\pm} = \pm(\frac{1}{\sqrt{2}}X + \frac{1}{\sqrt{2}}Y)$, $H^{\pi/8\pm} = \pm(\cos\frac{\pi}{8}X + \sin\frac{\pi}{8}Y)$.

Consider the adiabatic interpolations between the following Hamiltonians:

$$-Z \otimes \widetilde{G} \to -Y \otimes \widetilde{G} \to Z \otimes \widetilde{G}. \tag{1.30}$$

According to the above result, the first interpolation yields the transformation $V^{\pi/2+}$. The second interpolation can be regarded as the inverse of $Z \otimes \widetilde{G} \to -Y \otimes \widetilde{G}$ which is equivalent to $-Z \otimes \widetilde{G} \to Y \otimes \widetilde{G}$ since $\widehat{H}(t)$ and $-\widehat{H}(t)$ yield the same geometric transformations. Thus the second interpolation results in $(V^{\pi/2-})^\dagger = V^{\pi/2+}$. The net result is therefore $V^{\pi/2+}V^{\pi/2+} = iX$. We see that up to a global phase the above sequence results in a geometric implementation of the $X$ gate.

Similarly, one can verify that the $Z$ gate can be realized via the loop

$$-Z \otimes \widetilde{G} \to -X \otimes \widetilde{G} \to Z \otimes \widetilde{G} \to Y \otimes \widetilde{G} \to -Z \otimes \widetilde{G}. \tag{1.31}$$

The Phase gate $P$ can be realized by applying

$$-Z \otimes \widetilde{G} \to -(\frac{1}{\sqrt{2}}X + \frac{1}{\sqrt{2}}Y) \otimes \widetilde{G} \to Z \otimes \widetilde{G}, \tag{1.32}$$

followed by the $X$ gate.

The Hadamard gate $W$ can be realized by first applying $Z$, followed by

$$-Z \otimes \widetilde{G} \to -X \otimes \widetilde{G}. \tag{1.33}$$

Finally, the $\pi/8$ gate $T$ can be implemented by first applying $XZ$, followed by

$$Z \otimes \widetilde{G} \to -(\cos\frac{\pi}{8}X + \sin\frac{\pi}{8}Y) \otimes \widetilde{G} \to -Z \otimes \widetilde{G}. \tag{1.34}$$

#### *1.2.2.2 A note on the adiabatic condition*

Before we show how to implement the C-NOT gate, let us comment on the conditions under which the adiabatic approximation assumed in the above operations is satisfied. Because of the form (1.15) of the overall unitary, the adiabatic approximation depends on the extent to which each of the unitaries (1.16) approximate the expressions (1.17). The latter depends only on the adiabatic properties of the non-degenerate two-level Hamiltonian $H(t)$. For such a Hamiltonian, the simple version of the adiabatic condition [1] reads

$$\frac{\varepsilon}{\Delta^2} \ll 1, \tag{1.35}$$

where

$$\varepsilon = \max_{0 \le t \le T} |\langle \phi_1(t)| \frac{dH(t)}{dt} |\phi_0(t)\rangle|, \tag{1.36}$$

and

$$\Delta = \min_{0 \le t \le T} (E_1(t) - E_0(t)) = \min_{0 \le t \le T} 2E(t) \tag{1.37}$$

is the minimum energy gap of $H(t)$.

Along the segments of the parameter paths we described, the Hamiltonian is of the form (1.27) and its derivative is

$$\frac{dH^{\theta\pm}(t)}{dt} = \frac{df(t)}{dt} Z + \frac{dg(t)}{dt} H^{\theta\pm}, \qquad 0 < t < T. \tag{1.38}$$

This derivative is well defined as long as $\frac{df(t)}{dt}$ and $\frac{dg(t)}{dt}$ are well defined. The curves we described, however, may not be differentiable at the points connecting two segments. In order for the Hamiltonians (1.27) that interpolate between these points to be differentiable, the functions $f(t)$ and $g(t)$ have to satisfy $\frac{df(T)}{dt} = 0$ and $\frac{dg(0)}{dt} = 0$. This means that the change of the Hamiltonian slows down to zero at the end of each segment (except for a possible change in its strength), and increases again from zero along the next segment. We point out that when the Hamiltonian stops changing, we can turn it off completely by decreasing its strength. This can be done arbitrarily fast and it would not affect a state which belongs to an eigenspace of the Hamiltonian. Similarly, we can turn on another Hamiltonian for the implementation of a different operation.

The above condition guarantees that the adiabatic approximation is satisfied with precision $1 - O((\frac{\varepsilon}{\Delta^2})^2)$. It is known, however, that under certain conditions on the Hamiltonian, we can obtain better results [20, 4]. Let us write the Schrödinger equation as

$$i\frac{d}{dt}|\psi(t)\rangle = H(t)|\psi(t)\rangle \equiv \frac{1}{\epsilon}\bar{H}(t)|\psi(t)\rangle, \tag{1.39}$$

where $\epsilon > 0$ is small. Assume that $\bar{H}(t)$ is smooth and all its derivatives vanish at the end points $t = 0$ and $t = T$ (note that $\bar{H}(t)$ is non-analytic at these points, unless it is constant). Then if we keep $\bar{H}(t)$ fixed and vary $\epsilon$, the adiabatic error would scale super-polynomially with $\epsilon$, i.e., the error will decrease with $\epsilon$ faster than $O(\epsilon^N)$ for any $N$. (Notice that $\frac{\varepsilon}{\Delta^2} \propto \epsilon$, i.e., the standard adiabatic condition guarantees error $O(\epsilon^2)$.)

In our case, the smoothness condition translates directly to the functions $f(t)$ and $g(t)$. For any smooth $f(t)$ and $g(t)$ we can further ensure that the condition at the end points is satisfied by the reparameterization $f(t) \to f(y(t))$, $g(t) \to g(y(t))$ where $y(t)$ is a smooth function of $t$ which satisfies $y(0) = 0$, $y(T) = T$, and has vanishing derivatives at $t = 0$ and $t = T$. Then by slowing down the change of the Hamiltonian by a constant factor $\epsilon$, which amounts to an increase of the total time $T$ by a factor $1/\epsilon$, we can decrease the error super-polynomially in $\epsilon$. We will use this result to obtain a low-error interpolation in Sec. 1.2.5

where we estimate the time needed to implement a holonomic gate with a given precision.

#### 1.2.2.3 The C-NOT gate

The stabilizer or the gauge group on multiple blocks of the code is a direct product of the stabilizers or the gauge groups of the individual blocks. Therefore, from Eq. (1.8) it follows that one can always find an element of the initial stabilizer or gauge group on multiple blocks which has any desired combination of factors $\sigma^i$, $i = 0, 1, 2, 3$ on the first qubits in these blocks. It can be verified that applying transversal Clifford operations on the blocks does not change this property. Therefore, we can find an element of the stabilizer or the gauge group which has the form (1.9) where the factor $Z$ acts on the target qubit and $\widetilde{G}$ acts trivially on the control qubit. We now explain how we can implement the C-NOT gate holonomically starting from such a Hamiltonian.

Notice that a Hamiltonian of the form

$$\widehat{\widehat{H}}(t) = |0\rangle\langle 0|^c \otimes H_0(t) \otimes \widetilde{G} + |1\rangle\langle 1|^c \otimes H_1(t) \otimes \widetilde{G}, \tag{1.40}$$

where the superscript $c$ denotes the control qubit, gives rise to the unitary transformation

$$\widehat{\widehat{U}}(t) = |0\rangle\langle 0|^c \otimes \widehat{U}_0(t) + |1\rangle\langle 1|^c \otimes \widehat{U}_1(t), \tag{1.41}$$

where

$$\widehat{U}_{0,1}(t) = \mathcal{T}\exp(-i\int_0^t d\tau H_{0,1}(\tau) \otimes \widetilde{G}). \tag{1.42}$$

If $H_0(t)$ and $H_1(t)$ have the same non-degenerate instantaneous spectra, and $\mathrm{Tr}\{H_{0,1}(t)\} = 0$, then it follows that in the adiabatic limit each of the eigenspaces of $\widehat{\widehat{H}}(t)$ will undergo the geometric transformation

$$\widehat{\widehat{U}}_g(t) = |0\rangle\langle 0|^c \otimes V_0(t) \otimes \widetilde{I} + |1\rangle\langle 1|^c \otimes V_1(t) \otimes \widetilde{I}, \tag{1.43}$$

where $V_{0,1}(t) \otimes \widetilde{I}$ are the geometric transformations generated by $H_{0,1}(t) \otimes \widetilde{G}$ according to Proposition 1.

Our goal is to find $H_0(t)$ and $H_1(t)$, $H_0(0) = H_1(0) = Z$, such that at the end of the transformation the geometric unitary (1.43) will be equal to the C-NOT gate. In other words, we want $V_0(2T) = I$ and $V_1(2T) = X$ (we choose the total time of evolution to be $2T$ for convenience).

We already saw how to generate holonomically the $X$ gate up to a

phase—Eq. (1.30). We can use the same Hamiltonian in the place of $H_1(t)$:

$$H_1(t) = \begin{cases} H^{\pi/2+}(t), & 0 \le t \le T \\ H^{\pi/2-}(2T-t), & T \le t \le 2T. \end{cases} \tag{1.44}$$

Now we want to find a Hamiltonian $H_0(t)$ with the same spectrum as that of $H_1(t)$, which gives rise to a trivial geometric transformation, $V_0(2T) = I$ (possibly up to a phase, which can be undone later). Since all Hamiltonians of the type $H^{\theta\pm}(t)$ have the same instantaneous spectrum (for fixed $f(t)$ and $g(t)$), we can simply choose

$$H_0(t) = \begin{cases} H^{\pi/2+}(t), & 0 \le t \le T \\ H^{\pi/2+}(2T-t), & T \le t \le 2T, \end{cases} \tag{1.45}$$

which corresponds to applying a given transformation from $t = 0$ to $t = T$ and then undoing it (running it backwards) from $t = T$ to $t = 2T$. This results exactly in $V_0(2T) = I$.

Since, as we saw in Sec. 1.2.2.1, the Hamiltonian $H_1(t) \otimes \widetilde{G}$ gives rise to the geometric transformation $iX \otimes \widetilde{I}$, the above choice for the Hamiltonians (1.45) and (1.44) in Eq. (1.40) will result in the geometric transformation

$$|0\rangle\langle 0|^c \otimes I \otimes \widetilde{I} + i|1\rangle\langle 1|^c \otimes X \otimes \widetilde{I}, \tag{1.46}$$

which is the desired C-NOT gate up to a Phase gate on the control qubit. We can correct the phase by applying the inverse of the Phase gate to the control qubit, either before or after the described transformation.

Notice that from $t = 0$ to $t = T$ the Hamiltonians (1.45) and (1.44) are identical, i.e., during this period the Hamiltonian (1.40) has the form

$$I^c \otimes H^{\pi/2+}(t) \otimes \widetilde{G}, \tag{1.47}$$

so we are simply applying the single-qubit operation $V^{\pi/2+}$ to the target qubit according to the method for single-qubit gates described before. It is easy to verify that during the second period, from $t = T$ to $t = 2T$, the Hamiltonian (1.40) realizes the interpolation

$$-I^c \otimes Y \otimes \widetilde{G} \to -Z^c \otimes Z \otimes \widetilde{G}, \tag{1.48}$$

which has to be understood as in Eq. (1.27).

To summarize, the C-NOT gate can be implemented by first applying the inverse of the Phase gate ($P^\dagger$) on the control qubit, as well as

the transformation $V^{\pi/2+}$ on the target qubit, followed by the transformation (1.48). Due to the form (1.40) of $\widehat{\widehat{H}}(t)$, the extent to which the adiabatic approximation is satisfied during this transformation depends only on the adiabatic properties of the single-qubit Hamiltonians $H^{\pi/2\pm}(t)$ which we discussed in the previous subsection.

### *1.2.3 Encoded operations outside of the Clifford group*

For universal fault-tolerant computation we also need at least one encoded gate outside of the Clifford group. The fault-tolerant implementation of such gates is based on the preparation of a special encoded state [35, 19, 10, 34, 42] which involves a measurement of an encoded operator in the Clifford group. For example, the $\pi/8$ gate requires the preparation of the encoded state $\frac{|0\rangle+\exp(i\pi/4)|1\rangle}{\sqrt{2}}$, which can be realized by measuring the encoded operator $e^{-i\pi/4}PX$ [34]. Equivalently, the state can be obtained by applying the encoded operation $WP^\dagger$ on the encoded state $\frac{\cos(\pi/8)|0\rangle+\sin(\pi/8)|1\rangle}{\sqrt{2}}$ which can be prepared by measuring the encoded Hadamard gate [19]. The Toffoli gate requires the preparation of the three-qubit encoded state $\frac{|000\rangle+|010\rangle+|100\rangle+|111\rangle}{2}$ and involves a similar procedure [42]. In all these instances, the measurement of the encoded Clifford operator is realized by applying transversally the operator conditioned on the qubits in a "cat" state.

We now describe a general method that can be used to implement holonomically any conditional transversal Clifford operation with conditioning on the "cat" state. Let $O$ be a Clifford gate acting on the first qubits from some set of blocks. As we discussed in the previous section, under this unitary the stabilizer and the gauge group transform in such a way that we can always find an element with an arbitrary combination of Pauli matrices on the first qubits. If we write this element in the form

$$\widehat{G} = G_1 \otimes G_{2,\ldots,n}, \tag{1.49}$$

where $G_1$ is a tensor product of Pauli matrices acting on the first qubits from the blocks, and $G_{2,\ldots,n}$ is an operator on the rest of the qubits, then applying $O$ conditioned on the first qubit in a "cat" state transforms this stabilizer or gauge-group element as follows:

$$\begin{aligned} I^c \otimes G_1 \otimes G_{2,\ldots,n} &= |0\rangle\langle 0|^c \otimes G_1 \otimes G_{2,\ldots,n} + |1\rangle\langle 1|^c \otimes G_1 \otimes G_{2,\ldots,n} \\ &\rightarrow |0\rangle\langle 0|^c \otimes G_1 \otimes G_{2,\ldots,n} + |1\rangle\langle 1|^c \otimes OG_1O^\dagger \otimes G_{2,\ldots,n}, \end{aligned} \tag{1.50}$$

where the superscript $c$ denotes the control qubit from the "cat" state.

We can implement this operation by choosing the factor $G_1$ the same as the one we would use if we wanted to implement the operation $O$ according to the previously described procedure. Then we can apply the following Hamiltonian:

$$\widehat{\widehat{H}}_{C(O)}(t) = -|0\rangle\langle 0|^c \otimes G_1 \otimes G_{2,\ldots,n} - \alpha(t)|1\rangle\langle 1|^c \otimes H_O(t) \otimes G_{2,\ldots,n}, \quad (1.51)$$

where $H_O(t) \otimes G_{2,\ldots,n}$ is the Hamiltonian that we would use for the implementation of the operation $O$ and $\alpha(t)$ is a real parameter chosen such that at every moment the operator $\alpha(t)|1\rangle\langle 1|^c \otimes H_O(t) \otimes G_{2,\ldots,n}$ has the same instantaneous spectrum as the operator $|0\rangle\langle 0|^c \otimes G_1 \otimes G_{2,\ldots,n}$. This guarantees that the overall Hamiltonian is degenerate and the geometric transformation in each of its eigenspaces is

$$\widehat{\widehat{U}}_g(t) = |0\rangle\langle 0|^c \otimes I_1 \otimes I_{2,\ldots,n} + |1\rangle\langle 1|^c \otimes U_O(t) \otimes I_{2,\ldots,n}, \quad (1.52)$$

where $U_O(t)$ is the geometric transformation on the first qubits generated by $H_O(t) \otimes G_{2,\ldots,n}$. Since we presented the constructions of the basic Clifford operations up to an overall phase, the operation $U_O(t)$ may differ from the desired operation by a phase. This phase can be corrected by applying a suitable gate on the control qubit from the "cat" state (we explain how this can be done in the next section). We remark that a Hamiltonian of the type (1.51) requires fine tuning of the parameter $\alpha(t)$ and generally can be complicated. In Sec. 1.2.6 we will show that depending on the code one can find more natural implementations of these operations.

If we want to apply a second conditional Clifford operation $Q$ on the first qubits in the block, we can do this via the Hamiltonian

$$\widehat{\widehat{H}}_{C(Q)}(t) = -|0\rangle\langle 0|^c \otimes G_1 \otimes G_{2,\ldots,n} - \beta(t)|1\rangle\langle 1|^c \otimes H_Q(t) \otimes G_{2,\ldots,n}, \quad (1.53)$$

where $H_Q(t) \otimes G_{2,\ldots,n}$ is now the Hamiltonian we would use to implement the operation $Q$, had we implemented the operation $O$ before that. Here again, the factor $\beta(t)$ guarantees that there is no splitting of the energy levels of the Hamiltonian. Subsequent operations are applied analogously. Using this general method, we can implement holonomically any transversal Clifford operation conditioned on the "cat" state.

### *1.2.4 Using the "cat" state*

In addition to transversal operations, a complete fault-tolerant scheme requires the ability to prepare, verify and use a special ancillary state

such as the "cat" state $(|00...0\rangle + |11...1\rangle)/\sqrt{2}$. This can also be done in the spirit of the described holonomic scheme. Since the "cat" state is known and its construction is non-fault-tolerant, we can prepare it by simply treating each initially prepared qubit as a simple code (with $\widetilde{G}$ in Eq. (1.9) being trivial), and updating the stabilizer of the code via the applied geometric transformation as the operation progresses. The stabilizer of the prepared "cat" state is generated by $Z_iZ_j$, $i < j$. Transversal unitary operations between the "cat" state and other codewords are applied as described in the previous sections.

We also have to be able to measure the parity of the state, which requires the ability to apply successively C-NOT operations from two different qubits in the "cat" state to one and the same ancillary qubit initially prepared in the state $|0\rangle$. We can regard the qubit in state $|0\rangle$ as a simple code with stabilizer $\langle Z\rangle$, and we can apply the first C-NOT as described before. Even though after this operation the state of the target qubit is unknown, the second C-NOT gate can be applied via the same interaction, since the transformation in each eigenspace of the Hamiltonian is the same and at the end when we measure the qubit we project on one of the eigenspaces.

### *1.2.5 Fault tolerance of the scheme*

We saw how we can generate any transversal operation on the code space holonomically, assuming that the state has not undergone an error. But what if an error occurs on one of the qubits?

At any moment, we can distinguish two types of errors—those that result in transitions between the ground and the excited spaces of the current Hamiltonian, and those that result in transformations inside the eigenspaces. Due to the discretization of errors in quantum error correction, it suffices to prove correctability for each type separately. The key property of the construction we described is that in each of the eigenspaces the geometric transformation is the same and it is transversal. Because of this, if we are applying a unitary on the first qubit, an error on that qubit will remain localized regardless of whether it causes an excitation or not. If the error occurs on one of the other qubits, at the end of the transformation the result would be the desired single-qubit unitary gate plus the error on the other qubit, which is correctable.

We see that even though the Hamiltonian couples qubits in the same block, single-qubit errors do not propagate. This is because the coupling between the qubits amounts to a change in the relative phase between the

ground and excited spaces, but the latter is irrelevant since it is either equivalent to a gauge transformation, or when we apply a correcting operation we project on one of the eigenspaces. In the case of the C-NOT gate, an error can propagate between the control and the target qubits, but it never results in two errors within the same codeword.

### *1.2.6 FTHQC with the Bacon-Shor code using 3-local Hamiltonians*

The weight of the Hamiltonians needed for the scheme we described depends on the weight of the stabilizer or gauge-group elements. Remarkably, certain codes possess stabilizer or gauge-group elements of low weight covering all qubits in the code, which allows us to perform holonomic computation using low-weight Hamiltonians. Here we will consider as an example a subsystem generalization of the 9-qubit Shor code [36]—the Bacon-Shor code [6, 7]—which has particularly favorable properties for fault-tolerant computation [32, 31]. In the 9-qubit Bacon-Shor code, the gauge group is generated by the weight-two operators $Z_{k,j}Z_{k,j+1}$ and $X_{j,k}X_{j+1,k}$, where the subscripts label the qubits by row and column when they are arranged in a $3 \times 3$ square lattice. Since the Bacon-Shor code is a CSS code, the C-NOT gate has a direct transversal implementation. We now show that the C-NOT gate ca be realized using at most weight-three Hamiltonians.

If we want to apply a C-NOT gate between two qubits each of which is, say, in the first row and column of its block, we can use as a starting Hamiltonian $-Z^t_{1,1} \otimes Z^t_{1,2}$, where the superscript $t$ signifies that these are operators in the target block. We can then apply the C-NOT gate as described in the previous section. After the operation, however, this gauge-group element will transform to $-Z^t_{1,1} \otimes Z^c_{1,1} \otimes Z^t_{1,2}$. If we now want to implement a C-NOT gate between the qubits with index $\{1, 2\}$ using as a starting Hamiltonian the operator $-Z^t_{1,1} \otimes Z^c_{1,1} \otimes Z^t_{1,2}$ according to the same procedure, we will have to use a four-qubit Hamiltonian. Of course, at this point we can use the starting Hamiltonian $-Z^t_{1,2} \otimes Z^t_{1,3}$, but if we had also applied a C-NOT between the qubits labelled $\{1, 3\}$, this operator would not be available—it would have transformed to $-Z^t_{1,2} \otimes Z^t_{1,3} \otimes Z^c_{1,3}$.

What we can do instead, is to use as a starting Hamiltonian the operator $-Z^t_{1,1} \otimes Z^t_{1,2} \otimes Z^c_{1,2}$ which is obtained from the gauge-group element $Z^t_{1,1} \otimes Z^c_{1,1} \otimes Z^t_{1,2} \otimes Z^c_{1,2}$ after the application of the C-NOT between the qubits with index $\{1, 1\}$. Since the C-NOT gate is its own inverse,

we can regard the factor $Z_{1,1}^t$ as $\widetilde{G}$ in Eq. (1.48) and use this starting Hamiltonian to apply the procedure backwards. Thus we can implement any transversal C-NOT gate using at most weight-three Hamiltonians.

Since the encoded $X$, $Y$ and $Z$ operations have a bitwise implementation, we can always apply them according to the described procedure using Hamiltonians of weight 2. For the Bacon-Shor code, the encoded Hadamard gate can be applied via bitwise Hadamard transformations followed by a rotation of the grid to a 90 degree angle [32]. The encoded $P$ gate can be implemented by using the encoded C-NOT and an ancilla.

We point out that the preparation and measurement of the "cat" state can also be done using Hamiltonians of weight 2. To prepare the "cat" state, we prepare first all qubits in the state $(|0\rangle + |1\rangle)/\sqrt{2}$, which can be done by measuring each of them in the $\{|0\rangle, |1\rangle\}$ basis (this ability is assumed for any type of computation) and applying the transformation $-Z \rightarrow -X$ or $Z \rightarrow -X$ depending on the outcome. To complete the preparation of the "cat" state, apply a two-qubit transformation between the first qubit and each of the other qubits ($j > 1$) via the transformation

$$-I_1 \otimes X_j \rightarrow -Z_1 \otimes Z_j. \tag{1.54}$$

Single-qubit transformations on qubits from the "cat" state can be applied according to the method described in the previous section using at most weight-two Hamiltonians.

To measure the parity of the state, we need to apply successively C-NOT operations from two different qubits in the "cat" state to the same ancillary qubit initially prepared in the state $|0\rangle$. This can also be done according to the method described in Sec. 1.2.4 and requires Hamiltonians of weight 2.

For universal computation with the Bacon-Shor code, we also need to be able to apply one encoded transformation outside of the Clifford group. As we mentioned earlier, in order to implement the Toffoli gate or the $\pi/8$ gate, it is sufficient to be able to implement a C-NOT gate conditioned on a "cat" state. For the Bacon-Shor code, the C-NOT gate has a transversal implementation, so the conditioned C-NOT gate can be realized by a series of transversal Toffoli operations between the "cat" state and the two encoded states. We will now show that the latter can be implemented using at most three-qubit Hamiltonians.

Ref. [25] provides a circuit for implementing the Toffoli gate as a sequence of one- and two-qubit gates. We will use the same circuit, except that we flip the control and target qubits in every C-NOT gate

using the identity

$$W_1 W_2 CX_{1,2} W_1 W_2 = CX_{2,1}, \tag{1.55}$$

where $W_i$ denotes a Hadamard gate on the qubit labelled by $i$ and $CX_{i,j}$ denotes a C-NOT gate between qubits $i$ and $j$ with $i$ being the control and $j$ being the target. Let $\text{Toffoli}_{i,j,k}$ denote the Toffoli gate on qubits $i$, $j$ and $k$ with $i$ and $j$ being the two control qubits and $k$ being the target qubit, and let $P_i$ and $T_i$ denote the Phase and $\pi/8$ gates on qubit $i$, respectively. Then the Toffoli gate on three qubits (the first one of which we will assume to belong to the "cat" state), can be written as:

$$\begin{aligned}\text{Toffoli}_{1,2,3} = W_2 CX_{3,2} W_3 T_3^\dagger W_3 W_1 CX_{3,1} W_3 T_3 W_3 CX_{3,2} W_3 T_3^\dagger \\ \times W_3 CX_{3,1} W_3 T_3 W_3 W_2 T_2^\dagger W_2 CX_{2,1} W_2 T_2^\dagger W_2 CX_{2,1} W_2 P_2 W_1 T_1.\end{aligned} \tag{1.56}$$

To show that each of the above gates can be implemented holonomically using Hamiltonians of weight at most 3, we will need an implementation of the C-NOT gate which is suitable for the case when we have a stabilizer or gauge-group element of the form

$$\widehat{G} = X \otimes \widetilde{G}, \tag{1.57}$$

where the factor $X$ acts on the target qubit and $\widetilde{G}$ acts trivially on the control qubit. By a similar argument to the one in Sec. 1.2.2.3, one can verify that in this case the C-NOT gate can be implemented as follows: apply the operation $P^\dagger$ on the control qubit (we describe how to do this for our particular case below) together with the transformation

$$-X \otimes \widetilde{G} \to -Z \otimes \widetilde{G} \to X \otimes \widetilde{G} \tag{1.58}$$

on the target qubit, followed by the transformation

$$I^c \otimes X \otimes \widetilde{G} \to -(|0\rangle\langle 0|^c \otimes Z + |1\rangle\langle 1|^c \otimes Y) \otimes \widetilde{G} \to -I^c \otimes X \otimes \widetilde{G}. \tag{1.59}$$

Since the second and the third qubits belong to blocks encoded with the Bacon-Shor code, there are weight-two elements of the initial gauge group of the form $Z \otimes Z$ covering all qubits. The stabilizer generators on the "cat" state are also of this type. Following the transformation of these operators according to the sequence of operations (1.56), one can see that before every C-NOT gate in this sequence, there is an element of the form (1.57) with $\widetilde{G} = Z$ which can be used to implement the C-NOT gate as described provided that we can implement the gate $P^\dagger$ on the control qubit. We also point out that all single-qubit operations on qubit 1 in this sequence can be implemented according to the procedure describes

in Sec. 1.2.2.1, since at every step we have a weight-two stabilizer element on that qubit with a suitable form. Therefore, all we need to show is how to implement the necessary single-qubit operations on qubits 2 and 3. Due to the complicated transformation of the gauge-group elements during the sequence of operations (1.56), we will introduce a method of applying a single-qubit operation with a starting Hamiltonian that acts trivially on the qubit. For implementing single-qubit operations on qubits 2 and 3 we will use as a starting Hamiltonian the operator

$$\widehat{\widehat{H}}(0) = -I_i \otimes X_1 \otimes \widetilde{Z}, \quad i = 2, 3 \tag{1.60}$$

where the first factor ($I_i$) acts on the qubit on which we want to apply the operation (2 or 3), and $X_1 \otimes \widetilde{Z}$ is the transformed (after the Hadamard gate $R_1$) stabilizer element of the "cat" state that acts non-trivially on qubit 1 (the factor $\widetilde{Z}$ acts on some other qubit in the "cat" state).

To implement a single-qubit gate on qubit 3 for example, we first apply the interpolation

$$-I_3 \otimes X_1 \otimes \widetilde{Z} \rightarrow -Z_3 \otimes Z_1 \otimes \widetilde{Z}. \tag{1.61}$$

This results in a two-qubit geometric transformation $U_{1,3}$ on qubits 1 and 3. We do not have to calculate this transformation exactly since we will undo it later, but the fact that each eigenspace undergoes the same two-qubit geometric transformation can be verified similarly to the C-NOT gate we described in Sec. 1.2.2.3.

At this point, the Hamiltonian is of the form (1.10) with respect to qubit 3, and we can apply any single-qubit unitary gate $V_3$ according to the method described in Sec. 1.2.2.1. This transforms the Hamiltonian to $-V_3 Z_3 V_3^\dagger \otimes Z_1 \otimes \widetilde{Z}$. We can now "undo" the transformation $U_{1,3}$ by the interpolation

$$-V_3 Z_3 V_3^\dagger \otimes Z_1 \otimes \widetilde{Z} \rightarrow -I_3 \otimes X_1 \otimes \widetilde{Z}. \tag{1.62}$$

The latter transformation is the inverse of Eq. (1.61) up to the single-qubit unitary transformation $V_3$, i.e., it results in the transformation $V_3 U_{1,3}^\dagger V_3^\dagger$. Thus the net result is

$$V_3 U_{1,3}^\dagger V_3^\dagger V_3 U_{1,3} = V_3, \tag{1.63}$$

which is the desired single-qubit unitary transformation on qubit 3. We point out that during this transformation, a single-qubit error can propagate between qubits 1 and 3, but this is not a problem since we are implementing a transversal Toffoli operation and such an error would not result in more that one error per block of the code.

We see that with the Bacon-Shor code the above scheme for fault-tolerant HQC can be implemented with at most 3-local Hamiltonians. This is optimal for this approach, because there are no non-trivial codes with stabilizer or gauge-group elements of weight smaller than 2 covering all qubits. We will discuss the possibility of reducing the locality of the Hamiltonian further in Sec. 1.3.

### *1.2.7 Effects on the accuracy threshold*

Since the method we described conforms completely to a given fault-tolerant scheme, it would not affect the error threshold per qubit per operation for that scheme. However, the allowed errors per qubit per operation include both errors due to imperfectly applied transformations and errors due to interaction with the environment. It turns out that certain features of the method we described have effect on the allowed distribution of these errors within the accuracy threshold.

First, observe that when applying the Hamiltonian (1.11), we cannot at the same time apply operations on the other qubits on which the factor $\widetilde{G}$ acts non-trivially. Thus, some operations at the lowest level of concatenation that would otherwise be implemented simultaneously might have to be implemented serially. The effect of this is equivalent to slowing down the circuit by a constant factor. (Note that we could also vary the factor $\widetilde{G}$ simultaneously with $H(t)$, but in order to obtain the same precision as that we would achieve by a serial implementation, we would have to slow down the change of the Hamiltonian by the same factor.) The slowdown factor resulting from this loss of parallelism is usually small since this problem occurs only at the lowest level of concatenation. It can be verified that for the Bacon-Shor code, we can apply operations on up to 6 out of the 9 qubits in a block simultaneously. For example, when applying encoded single-qubit operations, we can address simultaneously any two qubits in a row or column by taking $\widetilde{G}$ to be a single-qubit operator $Z$ or $X$ on the third qubit in the same row or column. The Hamiltonians used for applying operations on the two qubits commute with each other at all times and do not interfere. A similar thing holds for the implementation of the encoded C-NOT gate or the operations involving the "cat" state. Thus for the Bacon-Shor code we have a slowdown due to parallelism by a factor of 1.5.

A more significant slowdown results from the fact that the evolution is adiabatic. In order to obtain a rough estimate of the slowdown due specifically to the adiabatic requirement, we will compare the time $T_h$

needed for the implementation of a holonomic gate with precision $1-\delta$ to the time $T_d$ needed for a dynamical realization of the same gate with the same strength of the Hamiltonian. We will consider a realization of the $X$ gate via the unitary interpolation

$$\widehat{H}(t) = -V_X(\tau(t))ZV_X^\dagger(\tau(t)) \otimes \widetilde{G}, \;\; V_X(\tau(t)) = \exp\left(i\tau(t)\frac{\pi}{2T_h}X\right), \tag{1.64}$$

where $\tau(0) = 0$, $\tau(T_h) = T_h$. The energy gap of this Hamiltonian is constant. The optimal dynamical implementation of the same gate is via the Hamiltonian $-X$ for time $T_d = \frac{\pi}{2}$.

As we argued in Sec. 1.2.2.2, the accuracy with which the adiabatic approximation holds for the Hamiltonian (1.64) is the same as that for the Hamiltonian

$$H(t) = V_X(\tau(t))ZV_X^\dagger(\tau(t)). \tag{1.65}$$

We now present estimates for two different choices of the function $\tau(t)$. The first one is

$$\tau(t) = t. \tag{1.66}$$

In this case the Schrödinger equation can be easily solved in the instantaneous eigenbasis of the Hamiltonian (1.65). For the probability that the initial ground state remains a ground state at the end of the evolution, we obtain

$$p = \frac{1}{1+\varepsilon^2} + \frac{\varepsilon^2}{1+\varepsilon^2}\cos^2\left(\frac{\pi}{4\varepsilon}\sqrt{1+\varepsilon^2}\right) = 1-\delta, \tag{1.67}$$

where

$$\varepsilon = \frac{T_d}{T_h}. \tag{1.68}$$

Expanding in powers of $\varepsilon$ and averaging the square of the cosine whose period is much smaller than $T_h$, we obtain the condition

$$\varepsilon^2 \leq 2\delta. \tag{1.69}$$

Assuming, for example, that $\delta \approx 10^{-4}$ (approximately the threshold for the 9-qubit Bacon-Shor code [32]), we obtain that the time of evolution for the holonomic case must be about 70 times longer than that in the dynamical case.

It is known, however, that if $H(t)$ is smooth and its derivatives vanish at $t = 0$ and $t = T_h$, the adiabatic error decreases super-polynomially

with $T_h$ [20, 4]. To achieve this, we will choose

$$\tau(t) = \frac{1}{a}\int_0^t dt' e^{-1/\sin(\pi t'/T_h)}, \quad a = \int_0^{T_h} dt' e^{-1/\sin(\pi t'/T_h)}. \tag{1.70}$$

For this interpolation, by a numerical solution we obtain that when $T_h/T_d \approx 17$ the error is already of the order of $10^{-6}$, which is well below the threshold values obtained for the Bacon-Shor codes [32]. This is a remarkable improvement in comparison to the previous interpolation which shows that the smoothness of the Hamiltonian plays an important role in the performance of the scheme.

An additional slowdown in comparison to a perfect dynamical scheme may result from the fact that the constructions for some of the standard gates we presented involve long sequences of loops. With more efficient parameter paths, however, it should be possible to reduce this slowdown to minimum. An approach for finding optimal loops presented in Ref. [41] may be useful in this respect.

In comparison to a dynamical implementation, the allowed rate of environment noise for the holonomic case would decrease by a factor similar to the slowdown factor. In practice, however, dynamical gates are not perfect and the holonomic approach may be advantageous if it allows for a better precision.

We finally point out that an error in the factor $H(t)$ in the Hamiltonian (1.11) would result in an error on the first qubit according to Eq. (1.24). Such an error clearly has to be below the accuracy threshold. More dangerous errors, however, are also possible. For example, if the degeneracy of the Hamiltonian is broken, this can result in an unwanted dynamical transformation affecting all qubits on which the Hamiltonian acts non-trivially. Such multi-qubit errors have to be of higher order in the threshold, which imposes more severe restrictions.

## 1.3 FTHQC with 2-local Hamiltonians

It is known that any adiabatic evolution can be approximated using 2-local Hamiltonians with the gadget techniques introduced in Ref. [24] and developed further in Refs. [39, 40]. The perturbative gadgets are 2-local Hamiltonians which simulate $k$-local ones in their lowest energy levels. However, the fault tolerance of the scheme we described depends on the property that in each eigenspace of the Hamiltonian we simultaneously implement the same transversal operations. The gadget techniques introduce additional energy levels where the computation does

not necessarily have this property. In order to reduce the locality of the Hamiltonian, in this section we propose an alternative scheme for which the gadget techniques can be applied without violating fault tolerance. This scheme uses additional gauge qubits.

### 1.3.1 Applying holonomic gates to unencoded qubits by the use of a noisy ancillary qubit

From Theorem 1 one can see that in the case when the Hilbert space factors as $\mathcal{H}^S = \mathcal{H}^A \otimes \mathcal{H}^B$, it is possible to apply holonomic computation on subsystem $\mathcal{H}^A$ without initializing the state of the system in any subspace. In particular, if we are given a system $\mathcal{H}^A$ in an unknown state, we can append to it another ancillary system $\mathcal{H}^B$ also in an unknown state, and apply any desired transformation holonomically on the first system. Since this approach does not require the preparation of pure ancillary states, it can be advantageous in implementations where the latter is difficult, such as nuclear magnetic resonance (NMR) [8].

We now present an explicit scheme for universal computation on qubits based on this principle. The scheme uses a single ancillary gauge qubit. We will show how to implement a universal set of one- and two-qubit gates. As in the previous scheme, we will use interpolations between Hamiltonians with two energy levels of equal degeneracy.

Let us label the two qubits on which we will be applying the gates by 1 and 2, and the gauge qubit by 3. In order to apply a single-qubit gate, say, on qubit 1, we will use the starting Hamiltonian

$$H(0) = I_1 \otimes X_3. \tag{1.71}$$

This is similar to the Hamiltonian (1.60) with the difference that the factor $\widetilde{Z}$ is missing. One can see that the absence of this factor does not change the workings of the construction described at the end of Sec. 1.2.6. Therefore, we can apply operations on qubit 1 in a similar manner. Namely, we first apply

$$I_1 \otimes X_3 \to Z_1 \otimes Z_3, \tag{1.72}$$

which results in the geometric transformation $U_{1,3}$. At this point we can apply holonomically an arbitrary gate on qubit 1 according to the method described in Sec. 1.2.2.1. If $G_1$ is the single-qubit gate which we apply, after the corresponding interpolation the net geometric transformation becomes $G_1 U_{1,3}$ and the Hamiltonian is transformed to $G_1 Z_1 G_1^\dagger \otimes Z_3$. We can now "undo" the unitary $U_{1,3}$ by applying the interpolation

$G_1 Z_1 G_1^\dagger \otimes Z_3 \to I_1 \otimes X_3$. The latter is the inverse of Eq. (1.72) up to the single-qubit unitary transformation $G_1$, i.e., it results in the transformation $G_1 U_{1,3}^\dagger G_1^\dagger$. Thus the net result is $G_1 U_{1,3}^\dagger G_1^\dagger G_1 U_{1,3} = G_1$, which is the desired unitary on qubit 1. Note that the relative dynamic phase between the ground and excited spaces which accumulates during the procedure, at the end is equivalent to a transformation on the gauge qubit 3.

For universal computation, we also need a non-trivial two-qubit gate. We can start again by the interpolation

$$I_1 \otimes I_2 \otimes X_3 \to I_1 \otimes Z_2 \otimes Z_3 \tag{1.73}$$

which results in the geometric transformation $U_{2,3}$. At this point we can apply, for example, the interpolation $I_1 \otimes Z_2 \otimes Z_3 \to I_1 \otimes Y_2 \otimes Z_3 \to Z_1 \otimes Z_2 \otimes Z_3$, which results in the gate $P_1^\dagger CX_{1,2}$ as shown in Sec. 1.2.2.3. To "undo" the operation $U_{2,3}$, we apply the transformation $Z_1 \otimes Z_2 \otimes Z_3 \to I_1 \otimes I_2 \otimes X_3$, which is the inverse of (1.73) up to the transformation $P_1^\dagger CX_{1,2}$. The net result is $P_1^\dagger CX_{1,2} U_{2,3}^\dagger CX_{1,2} P_1 P_1^\dagger CX_{1,2} U_{2,3} = P_1^\dagger CX_{1,2}$.

This method of applying holonomic gates to unencoded cubits suggests a simple way of realizing fault-tolerant computation by holonomic means—we can apply the same operations that we would use in a standard fault-tolerant scheme by coupling every qubit or pair of qubits in the code to an "external" gauge qubit as we just described. Obviously, a single error during the implementation of a transversal operation cannot propagate because qubits in the same block do not interact. By the same method we can also prepare and verify the "cat" state, which does not require transversal operation.

This scheme is conceptually simpler than the scheme described in the previous section, because it does not require Hamiltonians that depend on the stabilizer or the gauge group of the code. At the same time, it requires the use of additional gauge qubits which increases the chance for an error during the implementation of a given operation, and thus decreases the accuracy threshold of the fault-tolerant procedure (see Sec. 1.3.3). Nevertheless, since the scheme does not couple qubits in the same block, it allows for reducing the locality of the Hamiltonian by the use of perturbative gadgets without the risk of loosing fault tolerance.

### 1.3.2 Reducing the locality of the Hamiltonian with the 3-qubit gadget

Perturbative gadgets were introduced in Ref. [24] for the purpose of proving the QMA-completeness of the 2-local Hamiltonian problem and to show that universal adiabatic quantum computation can be implemented using 2-local Hamiltonians. The technique was further developed in Refs. [39] for simulating spatially non-local Hamiltonians with spatially local ones. Most recently, perturbative gadgets were generalized in Ref. [40] to obtain effective $k$-local interaction with 2-local Hamiltonians by a single application of $k^{\text{th}}$ order perturbation theory. For a detailed introduction to perturbative gadgets, we refer the reader to Ref. [24]. Here, we will describe how the 3-local Hamiltonian used in the above holonomic scheme can be reduced to 2-local using the 3-qubit gadget.

In the above scheme, a 3-local Hamiltonian is needed only for the implementation of a 2-qubit gate; single-qubit gates can be realized using 2-local Hamiltonians. Let us take the implementation of the gate $P_1^\dagger CX_{1,2}$ via the interpolation $I_1 \otimes I_2 \otimes X_3 \rightarrow I_1 \otimes Z_2 \otimes Z_3 \rightarrow I_1 \otimes Y_2 \otimes Z_3 \rightarrow Z_1 \otimes Z_2 \otimes Z_3 \rightarrow I_1 \otimes I_2 \otimes X_3$. The first two segments of this interpolation require 2-local Hamiltonians, so only the part $I_1 \otimes Y_2 \otimes Z_3 \rightarrow Z_1 \otimes Z_2 \otimes Z_3 \rightarrow I_1 \otimes I_2 \otimes X_3$ requires reduction. We will explain the use of the 3-qubit gadget for the segment $I_1 \otimes Y_2 \otimes Z_3 \rightarrow Z_1 \otimes Z_2 \otimes Z_3$ which corresponds to the 3-local Hamiltonian

$$H(t) = f(t) I_1 \otimes Y_2 \otimes Z_3 + g(t) Z_1 \otimes Z_2 \otimes Z_3, \tag{1.74}$$

where $f(0), g(T) > 0$, $g(0) = f(T) = 0$. The segment $Z_1 \otimes Z_2 \otimes Z_3 \rightarrow I_1 \otimes I_2 \otimes X_3$ can be obtained analogously.

To simulate the Hamiltonian 1.74, we introduce two ancilla registers, each consisting of three ancilla qubits. One of the registers is for the term $f(t) I_1 \otimes Y_2 \otimes Z_3$ and the other one for the term $g(t) Z_1 \otimes Z_2 \otimes Z_3$. We will label the three ancillas in the first register by the superscript $\{1, j\}$, $j = 1, 2, 3$, and the three ancillas in the second register by the superscript $\{2, j\}$, $j = 1, 2, 3$. The gadget Hamiltonian is [40]

$$H^{\text{gad}}(t) = \sum_{s=1}^{2} H_s^{\text{anc}} + \lambda \sum_{s=1}^{2} V_s(t), \tag{1.75}$$

where

$$H_s^{\text{anc}} = \sum_{1 \le i < j \le 3} \frac{1}{2} (I - Z_{s,i} Z_{s_j}) \tag{1.76}$$

and

$$V_1(t) = \sqrt[3]{f(t)} I_1 \otimes X_{1,1} + \sqrt[3]{f(t)} Y_2 \otimes X_{1,2} + \sqrt[3]{f(t)} Z_3 \otimes X_{1,3}, \quad (1.77)$$

$$V_2(t) = \sqrt[3]{g(t)} Z_1 \otimes X_{2,1} + \sqrt[3]{f(t)} Z_2 \otimes X_{2,2} + \sqrt[3]{f(t)} Z_3 \otimes X_{2,3}. \quad (1.78)$$

Here $\lambda$ is a small parameter in which the perturbation to order $O(\lambda^3)$ is carried out. If each of the ancilla registers is initialized in the cat state $(|000\rangle + |111\rangle)/\sqrt{2}$ (we showed how this can be done holonomically using 2-qubit Hamiltonians), then the restriction of $H^{\rm gad}(t)$ on the space spanned by its $2^3 = 8$ lowest-energy eigenstates is equal, up to an energy shift, to [40]

$$\widetilde{H}_{\rm eff} = \frac{3\lambda^3}{2} H(t) \otimes P_{\rm cat} + O(\lambda^4), \quad (1.79)$$

where $P_{\rm cat}$ is an operator on the space of the ancillas that projects each of the two registers in the "cat" state. In other words, up to a correction of order $O(\lambda^4)$, the three system qubits plus six ancilla qubits will effectively experience the Hamiltonian (1.74) acting locally on the system qubits, rescaled by the factor $\frac{3\lambda^3}{2}$.

### *1.3.3 Effects on the accuracy threshold*

The use of perturbative gadgets involves a larger number of qubits which increases the probability for an error during the implementation of a gate and decreases the accuracy threshold. In order to apply a transversal 2-qubit gate according to the procedure we described, we have to use seven extra qubits—one gauge qubit for the implementation of the 2-qubit gate before the use of the 3-qubit gadget, and six qubits for reducing the locality of the 3-local Hamiltonian with the gadget. An error on any of the total of nine qubits can result in an error on the two qubits on which the gate is applied, and therefore in order to ensure a particular precision of the 2-qubit gate, the error probability per qubit during the gate has to be approximately $9/2 = 4.5$ smaller than what it could be if the gate was implemented without extra qubits. Thus the accuracy threshold for this scheme would be decreased in comparison to the corresponding dynamical scheme or the scheme in Sec. 1.2 by a factor of a similar magnitude. We note that in order to avoid using extra qubits, one can apply all single-qubit gates according to the scheme in Sec. 1.2 and use extra qubits only for the implementation of two qubit gates.

In addition to decreasing the accuracy threshold per qubit per gate,

the perturbative gadgets have a significant effect on the accuracy threshold for environment noise. This is due to the fact that the effective Hamiltonian (1.79) obtained with the perturbative gadgets is a rescaled version of the simulated Hamiltonian $H(t)$ and has a smaller energy gap. Imagine that we are interested in implementing the gate with precision $1 - O(\delta)$, $\delta \ll 1$. If $T$ is the time for which we can achieve such a precision of the adiabatic approximation using the Hamiltonian $H(t)$, then using the Hamiltonian $\frac{3\lambda^3}{2}H(t)$ we would need time $T' = \frac{2}{3\lambda^3}T$ for the same precision. At the same time, the term $O(\lambda^4)$ in Eq.(1.79) during this time would give rise to an error of order $O(\lambda^4 T') = O(\lambda T)$ which also has to be of the order $O(\delta)$. Therefore, we must have

$$\lambda = O(\frac{\delta}{T}), \tag{1.80}$$

i.e.,

$$T' = O(\frac{T^4}{\delta^3}). \tag{1.81}$$

We see that for achieving a given precision $1-O(\delta)$ with the 3-qubit gadget we need time which is $O(\frac{T^3}{\delta^3})$ times longer than the time $T$ needed for this precision with a 3-local Hamiltonian. This is an enormous overhead if we require $\delta \approx 10^{-4}$! Besides the obvious implication of this factor for the computational efficiency of the scheme, it means that the error threshold for environment noise has to be decreased by the same factor in comparison to the scheme using 3-local Hamiltonians.

## 1.4 Conclusion and outlook

In this chapter we saw that HQC can be made fault tolerant by combining it with the techniques for fault-tolerant quantum error correction on stabilizer codes. This means that HQC is, at least in principle, a scalable method of computation. However, further research is needed in order to bring the presented ideas closer to experimental realization.

The first scheme we presented uses Hamiltonians that are elements of the stabilizer or the gauge group of the code. We saw that with the Bacon-Shor code, this scheme can be implemented with 2- and 3-qubit Hamiltonians. Since the scheme conforms completely to a given dynamical fault-tolerant scheme and does not require the use of extra qubits, it has the same error threshold as the dynamical scheme on which it is based. However, due to the fact that adiabatic gates are slower than dynamical gates and that at the lowest concatenation level this scheme

requires certain gates to be implemented serially or more slowly, the allowed error threshold for environment noise is lower in comparison to a dynamical scheme. The factor by which the environment error threshold decreases depends on how smooth the adiabatic interpolations are, but it seems to be at most $\sim 10^2$. Therefore, if the robustness provided by the geometric nature of the gates is sufficiently higher than that achievable by dynamical means, this approach could be advantageous in comparison to dynamical schemes. The main challenge in the implementation of this is approach, however, is that it requires the engineering of 3-local Hamiltonians.

The second scheme we discussed can be implemented with Hamiltonians which are independent of the stabilizer or the gauge group of the code, but it requires extra qubits which decreases the accuracy threshold per qubit per gate. Nevertheless, it allows for reducing the locality of the Hamiltonian using perturbative gadget techniques, showing that 2-local Hamiltonians are universal for fault-tolerant HQC. The disadvantage of using the gadgets is that they decreases the gap of the Hamiltonian by a very large factor, which requires a significant slowdown of the computation and decreases the allowed rate of environment noise. Due to the large value of this slowdown, we regard this scheme as a proof of principle rather than a realistic proposal.

Applying the described strategies to actual physical systems will undoubtedly require modifications in accordance with the available interactions in those systems. A possible way of avoiding the use of multi-local Hamiltonians without the use of perturbative gadgets could be to use higher-dimensional systems (e.g., qutrits) between which two-local interactions are naturally available. It may be possible to encode qubits in subspaces or subsystems of these higher-dimensional systems and use fault-tolerant techniques designed for stabilizer codes based on qudits [11]. Given that simple quantum error-correcting codes and two-qubit geometric transformations have been realized using NMR [9, 22] and ion-trap [23, 13] techniques, these systems seem particularly suitable for hybrid HQC-QEC implementations.

Finally, it is interesting to point out that the adiabatic regime in which the holonomic schemes operate is consistent with the Markovian model of decoherence. In Ref. [37] it was argued that the standard dynamical paradigm of fault tolerance is based on assumptions that are in conflict with the rigorous derivation of the Markovian limit. Although the threshold theorem has been extended to non-Markovian models [2, 33, 5], the Markovian assumption is an accurate approximation for a wide range of

physical scenarios [21] and allows for a much simpler description of the evolution in comparison to non-Markovian models (see Chapter **??**). In Ref. [37] it was shown that the weak-coupling-limit derivation of the Markovian approximation is consistent with computational methods that employ slow transformations, such as adiabatic quantum computation [17] or HQC. A theory of fault-tolerance for the adiabatic model of computation at present is not known, although significant steps in this direction have been undertaken (see Chapter **??**). The hybrid HQC-QEC schemes presented here provide solutions for the case of HQC. However, we point out that it is an open problem whether the Markovian approximation makes sense for a fixed value of the adiabatic slowness parameter when the circuit increases in size. Giving a definitive answer to this question requires a rigorous analysis of the accumulation of non-Markovian errors due to deviation from perfect adiabaticity.

The techniques described in this chapter may prove useful in other areas as well. It is possible that some combination of transversal adiabatic transformations and active correction could provide a solution to the problem of fault tolerance in the adiabatic model of computation.

# Bibliography


[1] A. Messiah. *Quantum Mechanics, Vol. II*. North-Holland Publishing Co., Amsterdam, 1965.

[2] B. M. Terhal and G. Burkard. Fault-Tolerant Quantum Computation For Local Non-Markovian Noise. *Phys. Rev. A*, 71:012336, 2005.

[3] C. Beny, A. Kempf, and D. W. Kribs. Generalization of quantum error correction via the Heisenberg picture . *Phys. Rev. Lett.*, 98:100502, 2007.

[4] D. A. Lidar, A. T. Rezakhani, and A. Hamma. Adiabatic approximation with better than exponential accuracy for many-body systems and quantum computation. e-print arXiv.org:0808.2697.

[5] D. Aharonov, A. Kitaev, and J. Preskill. Fault-tolerant quantum computation with long-range correlated noise. *Phys. Rev. Lett.*, 96:050504, 2006.

[6] D. Bacon. Operator quantum error correcting subsystems for self-correcting quantum memories. *Phys. Rev. A*, 73:012340, 2006.

[7] D. Bacon and A. Cassacino. Quantum error correcting subsystem codes from two classical linear codes. e-print arXiv.org:quant-ph/0610088.

[8] D. G. Cory, A. F. Fahmy, and T. F. Havel. Ensemble quantum computing by NMR spectroscopy. *Proc. Natl. Acad. Sci. USA*, 94:1634, 1997.

[9] D. G. Cory, M. D. Price, W. Maas, E. Knill, R. Laflamme, W. H. Zurek, T. F. Havel, and S. S. Somaroo. Experimental Quantum Error Correction. *Phys. Rev. Lett.*, 81:2152, 1998.

[10] D. Gottesman. Stabilizer Codes and Quantum Error Correction. Ph.D. thesis, Caltech, 1997, e-print arXiv.org:quant-ph/9705052.

[11] D. Gottesman. Fault-tolerant quantum computation with higher-dimensional systems. *Chaos Solitons Fractals*, 10:1749, 1999.

[12] D. Kult, J. Åberg, and E. Sjöqvist. Noncyclic geometric changes of quantum states. *Phys. Rev. A*, 74:022106, 2006.

[13] D. Leibfried, B. DeMarco, V. Meyer, D. Lucas, M. Barrett, J. Britton, W. M. Itano, B. Jelenković, C. Langer, T. Rosenband, and D. J. Wineland. Experimental demonstration of a robust, high-fidelity geometric two ion-qubit phase gate. *Nature*, 422:412, 2003.

[14] D. Poulin. Stabilizer formalism for operator quantum error correction. *Phys. Rev. Lett.*, 95:230504, 2005.

[15] D. W. Kribs, R. Laflamme, and D. Poulin. A unified and generalized approach to quantum error correction . *Phys. Rev. Lett.*, 94:180501, 2005.

[16] D. W. Kribs, R. Laflamme, D. Poulin, and M. Lesosky. Operator quantum

error correction. *Quant. Inf. and Comp.*, 6:383, 2006.

[17] E. Farhi, J. Goldstone, S. Gutmann, and M. Sipser. Quantum Computation by Adiabatic Evolution. e-print arXiv.org:quant-ph/0001106.

[18] E. Knill. On Protected Realizations of Quantum Information . e-print arXiv.org:quant-ph/0603252.

[19] E. Knill, R. Laflamme, and W. Zurek. Resilient quantum computation: error models and thresholds. e-print arXiv.org:quant-ph/9702058.

[20] G. A. Hagedorn and A. Joye. Elementary exponential error estimates for the adiabatic approximation. *J. Math. Anal. and Appl.*, 267:235, 2002.

[21] H. J. Carmichael. *An open system approach to quantum optics*. Springer, Berlin, 1993.

[22] J. A. Jones, V. Vedral, A. Ekert, and G. Castagnoli. Geometric quantum computation using nuclear magnetic resonance. *Nature*, 403:869, 2000.

[23] J. Chiaverini, D. Leibfried, T. Schaetz, M. D. Barrett, R. B. Blakestad, J. Britton, W. M. Itano, J. D. Jost, E. Knill, C. Langer, R. Ozeri, and D. J. Wineland. Realization of quantum error correction. *Nature*, 432:602, 2004.

[24] J. Kempe, A. Kitaev, and O. Regev. The Complexity of the Local Hamiltonian Problem. *SIAM Journal of Computing*, 35:1070, 2006.

[25] M.A. Nielsen and I.L. Chuang. *Quantum Computation and Quantum Information*. Cambridge University Press, Cambridge, UK, 2000.

[26] O. Oreshkov. Holonomic quantum computation on subsystems. in preparation.

[27] O. Oreshkov. Topics in quantum information and the theory of open quantum systems. Ph.D. thesis, University of Southern California, 2008, e-print arXiv:0812:4682.

[28] O. Oreshkov, D. A. Lidar, and T. A. Brun. Operator quantum error correction for continuous dynamics. *Phys. Rev. A*, 78:022333, 2008.

[29] O. Oreshkov, T. A. Brun, and D. A. Lidar. Holonomic computation on stabilizer codes. in preparation.

[30] O. Oreshkov, T. A. Brun, and D. A. Lidar. Fault-tolerant holonomic quantum computation. *Phys. Rev. Lett.*, 102:070502, 2009.

[31] P. Aliferis. Level reduction and the quantum threshold theorem. Ph.D. thesis, Caltech, 2007, e-print arXiv.org:quant-ph/0703230.

[32] P. Aliferis and A. W. Cross. Subsystem fault tolerance with the Bacon-Shor code. *Phys. Rev. Lett.*, 98:220502, 2007.

[33] P. Aliferis, D. Gottesman, and J. Preskill. Quantum accuracy threshold for concatenated distance-3 codes. *Quantum Inf. Comput.*, 6:97, 2006.

[34] P. O. Boykin, T. Mor, M. Pulver, V. Roychowdhury, and F. Vatan. On universal and fault-tolerant quantum computing . e-print arXiv.org:quant-ph/9906054.

[35] P. W. Shor. Fault-tolerant quantum computation. 37th Symposium on Foundations of Computing, 56-65, IEEE Computer Society Press, 1996.

[36] P. W. Shor. Fault-tolerant quantum computation with higher-dimensional systems. *Phys. Rev. A*, 52, 1995.

[37] R. Alicki, D. A. Lidar, and P. Zanardi. Internal consistency of fault-tolerant quantum error correction in light of rigorous derivations of the quantum Markovian limit. *Phys. Rev. A*, 73:052311, 2006.

[38] R. Blume-Kohout, H. K. Ng, D. Poulin, and L. Viola. Constructing qubits in physical systems. *Phys. Rev. Lett.*, 100:030501, 2008.

[39] R. Oliveira and B. Terhal. The complexity of quantum spin systems on a two-dimensional square lattice. e-print arXiv.org:quant-ph/0504050.

[40] S. Jordan and E. Farhi. Perturbative gadgets at arbitrary orders. *Phys. Rev. A*, 77:062329, 2008.

[41] S. Tanimura, M. Nakahara, and D. Hayashi. Exact solutions of the isoholonomic problem and the optimal control problem in holonomic quantum computation. *Jour. Math. Phys.*, 46:022101, 2005.

[42] X. Zhou, D. W. Leung, and I. L. Chuang. Methodology for quantum logic gate constructions. *Phys. Rev. A*, 62:052316, 2000.