QUANTUM ERROR CORRECTION WITH IMPERFECT GATES

A. Yu. Kitaev

L.D.Landau Institute for Theoretical Physics
117940, Kosygina St. 2
Moscow 117940, Russia
e-mail: kitaevitp.ac.ru

Quantum error correction can be performed fault-tolerantly. This allows to store a quantum state intact (with arbitrary small error probability) for arbitrary long time at a constant decoherence rate.

A quantum computer is capable to solve some computational problems (e.g. factoring of integers and the discrete logarithm [1]) which are exponentially hard for an ordinary computer. Howevere, physical implementation of a quantum computer still remains a big problem. Besides physical and technological difficulties, an essential theoretical issue is whether quantum computation can be performed fault-tolerantly. Error-correcting quantum codes [2–8] give only partial solution to this problem. Generally, use of error correcting codes requires some ideal device to recover a codeword (or its quantum analogue) from error. In this report I show how to recover from error despite new errors that may come during the process. This fault-tolerant recovery procedure allows to store a quantum state intact for an arbitrary long time under constant (but low enough) error rate. Note: the error rate is the decoherence probability per a codebit per unit time whereas the stored quantum state belongs to some low-dimensional *information subspace* of the 2^n-dimensional state space of n qubits.

This work was done before I learned about Shor's paper [9] where fault-tolerant quantum computing was suggested. This latter result goes far beyond simple storage of quantum states. However, Shor's method requires the error rate to be as low as $(\log t)^{-c}$ where t is the storage/computation time. Thus, combining both methods makes it possible to perform arbitrary long quantum computation at constant error rate [11].

1. ERROR-CORRECTING CODES

Codes can be most naturally introduced in connection with the information transmission problem. (We will use codes for another purpose, however). A classical communication channel receives binary words of length n which may be distored while being transmitted. This process is described by transition probabilities $P(x \mapsto y)$ between an input word x and an output word y. In the simplest model, $P(x \mapsto y) = p^{d(x,y)}(1-p)^{n-d(x,y)}$, where $d(x,y)$ is the Hamming distance between x and y (p is the error probability per bit). To simplify this description, we may divide all transitions

Quantum Communication, Computing, and Measurement
Edited by Hirota *et al.*, Plenum Press, New York, 1997

into likely and unlikely ones, by saying that a transition is unlikely if its probability is smaller than a given number. Equivalently, a transition $x \mapsto y$ is unlikely if $d(x,y) > k$, where k is a given number. Denote by $N = \mathbf{B}^n = \{0,1\}^n$ the set of all binary words, $E \subseteq N \times N$ the set of likely transitions. Error correction is possible if input words belong to a subset $M \subseteq N$ such that any two distinct words $x_1, x_2 \in M$ are not glued by E (i.e. there is no such y that $(x_1,y) \in E$ and $(x_2,y) \in E$). The subset $M \subseteq \mathbf{B}^n$ is called a *classical code*. To be more exact, a classical code is an injection $C : M \to \mathbf{B}^n$, where M is a given set of messages to be encoded.

An input of a quantum channel is a quantum state of n qubits, $|\xi\rangle \in \mathcal{N} = \mathcal{B}^{\otimes n}$ (where $\mathcal{B} = \mathbf{C}^2$ is the state space of a single qubit). A quantum channel is described by a superoperator (i.e. operator acting on density operators). In this report I will stay with a naive approach, assuming that errors occur to each qubit with probability p. See [11] for rigorous consideration. * We can also define a (likely) error space $\mathcal{E} \subseteq \mathbf{L}(\mathcal{N})$, where $\mathbf{L}(\mathcal{N})$ is the space of linear operators $\mathcal{N} \to \mathcal{N}$. More specifically, $\mathcal{E}$ is the linear span of all operators acting on arbitrary k qubits. A quantum code is a linear subspace $\mathcal{M} \subseteq \mathcal{B}^{\otimes n}$, or an isometric injection $\mathcal{M} \to \mathcal{B}^{\otimes n}$. Errors can be recovered if any two orthogonal vectors $|\xi\rangle$, $|\eta\rangle \in \mathcal{M}$ remain orthogonal, meaning that $\langle \xi Y^\dagger | X\eta \rangle = 0$ for any $X, Y \in \mathcal{E}$.

The simplest classical code is based on repetition: 0 is represented by $(0,\ldots,0)$, 1 by $(1,\ldots,1)$. Does this construction extend to the quantum case? Of course, one can define a mapping $\mathcal{B} \to \mathcal{B}^{\otimes n}$ by the formulas $|0\rangle \mapsto |0,\ldots,0\rangle$, $|1\rangle \mapsto |1,\ldots,1\rangle$. However, such a code does not protect even from one error. Indeed, the corresponding error space $\mathcal{E}$ is the linear span of the identity operator (no errors) and the Pauli operators $\sigma_\gamma[j]$, $(\gamma = x,y,z,\ j = 1,\ldots,n)$. (This notation means the the Pauli operator σ_γ acts on the j-th qubit). Let us take two orthogonal vectors from the information subspace,

$$|\xi\rangle \ = \ 2^{-1/2}\big(|0,\ldots,0\rangle + |1,\ldots,1\rangle\big) \qquad\qquad |\eta\rangle \ = \ 2^{-1/2}\big(|0,\ldots,0\rangle - |1,\ldots,1\rangle\big)$$

and check the above error correction criterion. Consider two operators from the error space, $X = 1$ and $Y = \sigma_z[j]$. The operator Y preserves the vector $|0,\ldots,0\rangle$ whereas the vector $|1,\ldots,1\rangle$ is multiplied by factor -1 (such operators are called *phase errors*). Clearly, $Y|\xi\rangle = X|\eta\rangle = |\eta\rangle$, so orthogonal vectors do not remain orthogonal! Thus, repetition is not sutable for quantum coding. The simplest quantum error-correcting code maps 1 qubit into 5, see below.

More general and widely used classical codes are linear codes [10]. A linear code can be described by a collection of linear forms (check sums) which vanish on the code-words. A quantum analogue of a check sum is an operator of the form $\sigma(\gamma_1,\ldots,\gamma_n) = \sigma_{\gamma_1}[1]\ldots\sigma_{\gamma_n}[n]$ $(\gamma_j \in \{0,x,y,z\})$, where $\sigma_0 = 1$. Consider several operators of this form, $X_k = \sigma(f_k)$ $(k = 1,\ldots,s)$ which commute with each other. (This condition can be repesented in terms of f_k, see below). Let us define the information subspace as follows

$$\mathcal{M} \ = \ \Big\{ |\xi\rangle \in \mathcal{B}^{\otimes n} \ : \ X_j|\xi\rangle = |\xi\rangle \ \ (j = 1,\ldots,s) \Big\}. \qquad (1)$$

To check whether a given quatum state belongs to $\mathcal{M}$, one should measure the eigenvalues of $X_1,\ldots,X_s$ — all the eigenvalues must be equal to 1. (Hence the operators $X_1,\ldots,X_s$ are called *check operators*). This wide class of quantum codes was first described in ref. [8]. We will call such codes *symplectic* because of an intrinsic symplectic structure which underlie their properties.

* The naive approach may be misleading in estimating the probability of a specific event because quantum mechanics involves addition of amplitudes rather than probabilities. As a heuristic rule, one may take the square root of the naive probability to be an upper bound for the true one.

In study of symplectic codes commutation relations between the Pauli matrices play an important role. These relations become mathematically clear if we change the notations: $\sigma_0 = \sigma_{00} = 1$, $\sigma_x = \sigma_{10}$, $\sigma_y = \sigma_{11}$, $\sigma_z = \sigma_{01}$. The index set $G = \{0, x, y, z\} = \{00, 10, 11, 11\}$ can be identified with the group $\mathbf{Z}_2 \times \mathbf{Z}_2$. Then

$$\sigma_{\alpha\beta}\sigma_{\alpha'\beta'} = (-\mathrm{i})^{\alpha\beta'-\beta\alpha'}\sigma_{\alpha+\alpha',\beta+\beta'} = (-1)^{\alpha\beta'-\beta\alpha'}\sigma_{\alpha',\beta'}\sigma_{\alpha,\beta}$$

$$\sigma(f)\sigma(g) = (-1)^{\omega(f,g)}\sigma(g)\sigma(f) \tag{2}$$

where ω is the canonical skew-symmetric bilinear form on the group G^n

$$\omega\Big((\alpha_1\beta_1, \ldots, \alpha_n\beta_n), (\alpha'_1\beta'_1, \ldots, \alpha'_n\beta'_n)\Big) = \sum_{j=1}^{n} \alpha_j\beta'_j - \beta_j\alpha'_j \pmod 2. \tag{3}$$

Hence the operators $X_j = \sigma(f_j)$ commute with each other iff $\omega(f_j, f_m) = 0$ for every j and m. It follows that the form ω vanishes on the linear subspace (subgroup) $F \subseteq G^n$ generated by $f_1, \ldots, f_s$. It is called a *characteristic subspace* of a symplectic code. (Beware that it is a subspace over the residui field modulo 2, not over complex numbers!) The information subspace $\mathcal{M}$ depends on F rather than the check vectors $f_1, \ldots, f_s$. From now on, s stands for the dimensionality of F, i.e. linearly dependent check vectors are excluded. The dimensionality of $\mathcal{M}$ equals 2^{n-s}, which is sufficient to encode a state of $n - s$ qubits.

Let us turn to the error-correcting properties of symplectic codes. The error space $\mathcal{E} = \mathcal{E}(n, k)$ is generated by the operators $\sigma(g) : g \in E(n, k)$, where

$$E(n, k) = \Big\{g \in G^n : |\operatorname{Supp}(g)| \le k\Big\},$$
$$\operatorname{Supp}(\alpha_1\beta_1, \ldots, \alpha_n\beta_n) = \Big\{j : \alpha_j \neq 0 \text{ or } \beta_j \neq 0\Big\}.$$

By abuse of language, vectors $g \in E(n, k)$ will be called called "errors". (Thus, "correcting errors from $E(n, k)$" and "correcting k errors" is the same). Two errors, g' and g'' are called *equivalent* if the corresponding operators $\sigma(g')$ and $\sigma(g'')$ coincide on the information subspace $\mathcal{M}$. This condition can be written as $g' - g'' \in F$. Errors equivalent to 0 may be neglected because they do not affect quantum states from the information subspace.

Suppose that a quantum state $|\xi\rangle \in \mathcal{M}$ undergoes an error g. The resulting state $\sigma(g)|\xi\rangle$ is an eigenvector of all the check operators X_j. The corresponding eigenvalues are equal to $(-1)^{\mu_j(g)}$, where $\mu_j(g) = \omega(f_j, g)$. The binary vector $\mu(g) = (\mu_1(g), \ldots, \mu_s(g))$ is called the error *syndrom*. Obviously, equivalent errors have the same syndrom but the coinverse is not always true.

Theorem 1 *A symplectic code corrects k errors iff any two errors $g', g'' \in E(n, k)$ with equal syndroms are equivalent.*

Proof.
Let $|\xi\rangle$, $|\eta\rangle \in \mathcal{M}$ be arbitrary orthogonal vectors. The condition $\langle\xi\sigma(g')^\dagger|\sigma(g'')\eta\rangle = 0$ is guaranteed, for every $|\xi\rangle$ and $|\eta\rangle$, in the following two cases:

1. The errors g' and g'' are equivalent. (Make use of the fact that the operator $\sigma(g')$ is unitary).

2. The errors g' and g'' have different syndroms. Then $\sigma(g')|\xi\rangle$ and $\sigma(g'')|\eta\rangle$ belong to different eigenspaces of some check operator.

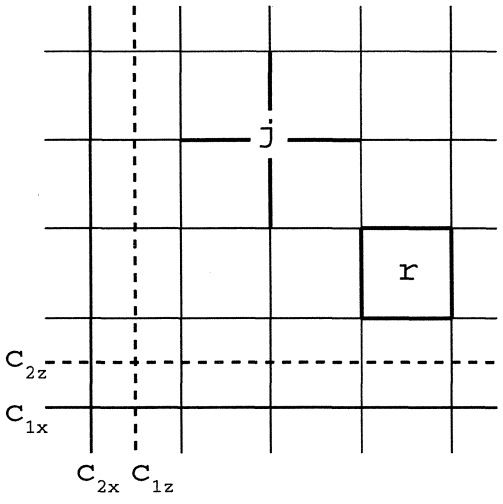

Figure 1. The toric code TOR(5).

Q.E.D.

This proof is based on the general error correction criterion. It is also possible to describe a correction procedure explicitly. First, the eigenvalues of the check operators are measured, yielding the error syndrom $\mu(g)$. The corresponding error g can be determined uniquely, up to equivalence. (This step may be computationally difficult). Finally, the operator $\sigma(g)$ is applied, which cancels the error.

Let $F_+ = \{g \in G^k : \mu(g) = 0\}$. Obviously, $F_+ \supseteq F$. Theorem 1 can be formulated as follows: *A symplectic code corrects k errors iff* $E(n, 2k) \cap F_+ \subseteq F$.

Example of a symplectic code which maps 1 qubit into 5 and corrects 1 error (see [6, 7]). The check vectors $f_1, \ldots, f_4$ are given by the rows of the following table:

1	0	1	0	0	1	0	1	0	0
1	0	0	1	1	0	0	0	0	1
0	1	0	0	0	1	1	0	0	1
0	1	0	1	0	0	0	1	1	0

Any two pairs of columns are linearly independent, hence $E(4, 2) \cap F_+ = \{0\}$.

2. TORIC CODES

Now I am going to describe an infinite sequence of symplectic codes TOR(k) : $k = 1, 2, \ldots$ which map 2 qubits into $n = 2k^2$. These codes are not optimal in any sense but they have the following nice properties:

1. Each check operators involves bounded number of qubits (at most 4).
2. Each qubit is involved in a bounded number of check operators (at most 4).
3. The number of corrected errors is unlimited. (More specifically, the code TOR(k) corrects $\left\lfloor \frac{k-1}{2} \right\rfloor$ errors).

Such codes are called *local check codes.*

Consider a $k \times k$ square lattice on the torus (see fig. 1). Let A_0 be the set of vertices of this lattice, A_1 the set of edges, A_2 the set of faces (i.e. square cells). We associate a qubit with each edge. (So there are $n = 2k^2$ qubits). Check operators are associated with each face $r \in A_2$ and each vertex $j \in A_0$,

$$X_r = \prod_{l \in \text{border}(r)} \sigma_x[l], \qquad X_j = \prod_{l \in \text{star}(j)} \sigma_z[l]. \tag{4}$$

It is easy to verify that these operators commute with each other. The key observation is that a border and a star either have 2 common edges or do not overlap at all.

There are two relations between the check operators: $\prod_{r \in A_2} X_r = 1$ and $\prod_{j \in A_0} X_j = 1$. It follows that $s = 2k^2 - 2$, hence the dimensionality of the information subspace equals $2^{n-s} = 4$.

Let us define *information operators* $Y_{x1}, Y_{x2}, Y_{z1}, Y_{z2}$ to be the products of $\sigma_x[l]$ or $\sigma_z[l]$ over all edges along the cycles c_{x1}, c_{x2} or cocycles (cuts) c_{z1}, c_{z2} (see fig. 1). The information operators commute with the check ones but do not commute with each other. The commutation relations between $Y_{x1}, Y_{x2}, Y_{z1}, Y_{z2}$ are as follows

$$Y_{xk}Y_{xl} = Y_{xl}Y_{xk}, \qquad Y_{zk}Y_{zl} = Y_{zl}Y_{zk}, \qquad Y_{xk}Y_{zl} = (-1)^{\delta_{kl}} Y_{zl}Y_{xk}, \qquad Y_{\gamma k}^2 = 1.$$

One can show that $Y_{x1}, Y_{x2}, Y_{z1}, Y_{z2}$ act on the information subspace $\mathcal{M}$ exactly the same way as the Pauli matrices $\sigma_x[1], \sigma_x[2], \sigma_z[1], \sigma_z[2]$ on two qubits. Hence $\mathcal{M}$ can be canonically identified with $\mathcal{B}^{\otimes 2}$. For practical applications, it is enough to use 1 of the 2 information qubits corresponding to $\mathcal{M}$.

Error-correcting properties of the toric codes are related to homology of the torus. Let us decompose each vector $g = (\alpha_1\beta_1, \ldots, \alpha_n\beta_n) \in G^n$ into an *x-component* $g_x = (\alpha_1 0, \ldots, \alpha_n 0)$ and a *z-component* $g_z = (0\beta_1, \ldots, 0\beta_n)$. (The corresponding operators $\sigma(g_x)$ and $\sigma(g_z)$ consist of $\sigma_x[j]$ and $\sigma_z[j]$, respectively). Each x-error corresponds to a 1-chain, the syndrom being represented by its boundary. An x-error g is equivalent to 0 iff the operator $\sigma(g)$ can be constructed from the check operators. This is the case iff g is a boundary of some 2-chain. Hence, two nonequivalent errors with equal syndroms must differ by a nontrivial cycle. Such a cycle includes $\geq k$ edges. It follows that the code TOR(k) corrects $\left\lfloor \frac{k-1}{2} \right\rfloor$ x-errors. (z-errors are treated similarly, but cycles are replaced with cocycles).

3. FAULT-TOLERANT PROCEDURES FOR ERROR CORRECTION

The usual error correction procedure was described after the proof of Theorem 1. It can be realize by a quantum circuit. Unfortunately, absolutely reliable quantum gates do not exist, so new errors may occur during error correction. Real quantum gates (to be constructed one day) should suffer from noise and decoherence. We will assume that the error rate (error probability per gate) p is arbitrarily small but constant. The problem is how to decrease an effective error probability for encoded qubits. To begin with, let us make the correction procedure stable to a fixed number (say, 2) errors during its execution.

Recall that the correction procedure consists of 3 steps. The second step is a classical computation, so it is safe. (However, the computation should not last too long, otherwise the qubits will decohere because of interaction with their environment). The third step is also rather safe because one error spoils one qubit. The most dangerous is the first step, the measurement. First of all, a single error in the syndrom can make impossible to determine the error in the qubits. What is even worse, one error during a

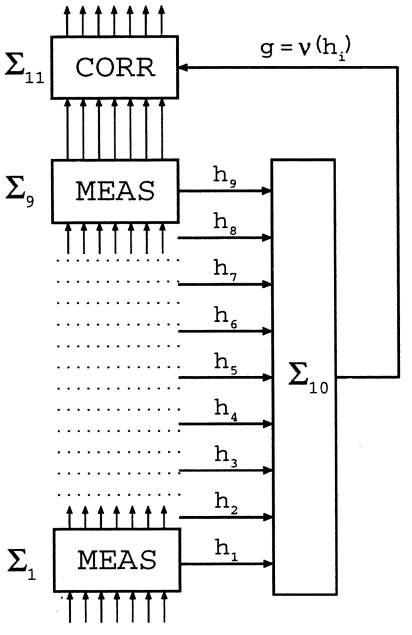

Figure 2. An error correction procedure which is stable to 2 new errors.

measurement can spoil *all the qubits involved in this measurement.* † That is why local check codes are especially useful for fault-tolerant error correction. For a local check code, all the measurements can be organized into a quantum circuit of bounded depth. Therefore, one error will spoil a bounded number of qubits. We just have to satisfy the inequality

The number of new errors during error correction	×	The maximum number of qubits spoiled by one error	≤	The number of errors permitted by the code

The right hand side is not limited, so "the number of new errors" can be arbitrary large. For this, the code parameter k should be large enough.

In the above consideration we didn't worry about errors in the syndrom. To ensure that the syndrom is correct, all the measurements must be repeated several time. For example, assume that up to 2 errors may happen during error correction. Then we have to measure the whole syndrom 9 times. The corresponding circuit is schematically drawn in fig. 2. It consists of 11 blocks (subcircuits). The blocks $\Sigma_1, \ldots, \Sigma_9$ perform syndrom measurements. The block Σ_{10} selects 3 identical measurement results $h_i = h_{i+1} = h_{i+2}$, coming one by one, and compute the corresponding error $g : \mu(g) = h_i$. Such results must exist (because at least 7 of 9 measurements are correct) but possibly are not unique. In any case, at least 1 of 3 selected results was correct for the time of measurement (no matter which, because they are identical). Subsequent wrong measurements may disturb some of the qubits, but not too many. Therefore, the computed value of g is nearly correct after all the measurements are done. Actual corrections are made to the qubits by the block Σ_{11}.

†P. Shor [9] suggested a special procedure which prevents errors from being so destructive. I do not use this result.

Note that this procedure works with a *constant* number of qubits n and takes a constant time t. [‡] It allows to reduce the effective error probability to $O(p^3)$, because 3 errors can still spoil the encoded quantum state. However, we must remember that quantum probability is a subtle thing, see footnote on page 182. A reliable upper bound for the effective error probability is $p_1 = O(p^{3/2})$, see ref. [11]. It is not also too bad.

The above result can be improved by the use of *cascade codes*. We have already encoded 1 qubit into n. Let us encode each of these n qubits again. Thus we reduce the effective error probability down to $p_2 = O\left((p^{3/2})^{3/2}\right)$. Now we need n^2 code qubits. Proceeding this way, we find that effective error probability ϵ is attained with $\sim (log(1/\epsilon))^\alpha$ qubits, where $\alpha = \log_{3/2} n$.

One problem is still remaining: how to perform error correction for a cascade code? For simplicity, consider the 2-level code, TOR(k) substituted into TOR(k). The top level correcting circuit must work with qubits represented in the code TOR(k). So, we must be able to manipulate with encoded qubits. Moreover, such manipulation must be fault-tolerant. Thus we have arrived to a harder problem than the original one: we must not only store a quantum state but also make some quantum computations fault-tolerantly. Fortunately, these are rather simple computations, just syndrom measurements. Such a measurement can be performed by a circuit with gates of two types:

$$S = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}, \qquad \widehat{\oplus} : |a, b\rangle \mapsto |a, a \oplus b\rangle, \tag{5}$$

where $\oplus$ stands for addition modulo 2. So, we should realize these gates on qubits represented in the code TOR(k). Such a realization must be a quantum circuit constructed from the same gates. It must be stable to 2 errors.

I will describe these realizations very briefly. To perform the operation S on an encoded information qubit, one should apply the operator S to each code qubit separately. After that, the code qubits should be permuted as follows. The square lattice on the torus (fig. 1) is flipped around the diagonal and shifted by 1/2 of its period in both directions. This operation transforms the vertex operators X_j into the face operators X_r and conversely. Clearly, the information subspace is invariant under this transformation. The information operators Y_{xk} and Y_{zk} are exchanged. Now recall that Y_{xk} and Y_{zk} represent the action of σ_x and σ_z on the information qubits. Therefore, each of the 2 information qubits undergoes the transformation $\sigma_x \leftrightarrow \sigma_z$. It is exactly what we need, because $S\sigma_x S^\dagger = \sigma_z$, $S\sigma_z S^\dagger = \sigma_x$.

The operator $\widehat{\oplus}$ is realized by bitwise action of $\widehat{\oplus}$ on the code qubits.

ACKNOWLEDGMENTS

This work was supported by the Russian Foundation for Fundamental Research, grant No 96-01-01113.

REFERENCES

[1] P. W. Shor, "Algorithms for quantum computation: discrete log and factoring", *Proceedings of the 35th Annual Symposium on the Foundations of Computer Science* (IEEE Computer Society Press, Los Alamitos, CA, 1994), p. 124.

[‡] I didn't try to calculate the actual value of the code parameter k which is necessary for the procedure to work. Probably, it is about 20 which corresponds to $n \sim 1000$ qubits. It is somewhat expensive. However, my purpose is to propose a method rather than a practically usable algorithm; optimization is a separate task.

[2] P. W. Shor, "Scheme for reducing decoherence in quantum memory," *Phys. Rev.* **A 52**, 2493–2496 (1995).

[3] A. M. Steane, "Multiple particle interference and quantum error correction", LANL e-print quant-ph/9601029, http://xxx.lanl.gov (submitted to *Proc. Roy. Soc. London A*).

[4] C. H. Bennett, G. Brassard, S. Popescu, B. Schumacher, J. A. Smolin, and W. K. Wootters, "Purification of noisy entanglement and faithful teleportation via noisy channels," *Phys. Rev. Lett.* **76**, 722 (1996).

[5] A. R. Calderbank and P. W. Shor, "Good quantum error-correcting codes exist", LANL e-print quant-ph/9512032, http://xxx.lanl.gov (to appear in *Phys. Rev. A*).

[6] R. Laflamme, C. Miquel, J. P. Paz,and W. H. Zurek, "Perfect quantum error correction code," LANL e-print quant-ph/9602019, http://xxx.lanl.gov

[7] C. H. Bennett, D. P. DiVincenzo, J. A. Smolin, W. K. Wootters, "Mixed state entanglement and quantum error-correcting codes", LANL e-print quant-ph/9604024, http://xxx.lanl.gov

[8] A. R. Calderbank, E. M. Rains, P. W. Shor, and N. J. A. Sloane, "Quantum error correction and orthogonal geometry", LANL e-print quant-ph/9605005, http://xxx.lanl.gov

[9] P. W. Shor, "Fault-tolerant quantum computation", LANL e-print quant-ph/9605011, http://xxx.lanl.gov

[10] F. J. MacWilliams and N. J. A. Sloane, "The Theory of Error-Correcting Codes", North-Holland, Amsterdam (1977).

[11] A. Yu. Kitaev, "Quantum computing: algorithms and error correction", *Russian Mathematical Surveys*, to appear.