Contemporary Mathematics

Pauli Exchange and Quantum Error Correction

Mary Beth Ruskai

Abstract. In many physically realistic models of quantum computation, Pauli exchange interactions cause a special type of two-qubit errors, called exchange errors, to occur as a first order effect of couplings within the computer. We discuss the physical mechanisms behind exchange errors and codes designed to explicitly deal with them.

1. Introduction

Most discussions of quantum error correction assume, at least implicitly, that errors result from interactions with the environment[1] and that single qubit errors are much more likely than two qubit errors. Most discussions also ignore the Pauli exclusion principle and permutational symmetry of the states describing multi-qubit systems. Although this can be justified by consideration of the full wave function, including spatial as well as spin components, an analysis (given in [**27**]) of these more complete wave functions suggests that more attention should be given to the effect of exchange interactions within a quantum computer. Interactions between identical particles can cause an error in two bits simultaneously as a first order effect. Moreover, because they result from interactions within the quantum computer, exchange errors cannot be reduced by better isolating the quantum computer from its environment. The effect of a (single) exchange error is to flip two bits if, and only if, they are different. It is a non-classical type of error in the sense that it arises directly from a physical mechanism which occurs only in the case of identical particles which follow the rules of quantum theory. If classical systems were to exhibit this type of behavior, they would require unusual correlations which do not normally occur from first-order couplings.

Schemes for fault-tolerant computation [**6, 22, 28**] have been developed which treat two-bit, and even multi-bit, errors. (See, e.g., [**8, 9, 11, 29**] and references in [**7, 23**] and at the end of Chapter 10 of [**20**].) However, many of these, such as those

2000 *Mathematics Subject Classification.* Primary 81V70; Secondary 94B60, 81R99, 81Q05.

Supported in part by National Science Foundation Grant DMS-97-06981 and Army Research Office Grant DAAG55-98-1-0374.

[1]We consider the "environment" as external to the quantum computer in the sense that interactions among qubits, whether or not used in the implementation of quantum gates, are *not* regarded as arising from the environment. Many authors follow [**9**] in defining the environment to include all "unwanted interactions".

arising from concatenated codes [**1, 10, 23**] require a large number of physical bits to represent one logical bit. Steane, in particular, has emphasized the problems associated with the size of repeatedly concatenated codes and discussed techniques [**29**] for coding m logical bits in n qubits. Furthermore, threshold estimates [**1, 6, 12, 22, 28**] are generally based on the assumption that two-bit errors are second order effects resulting from uncorrelated interactions with the environment. In those situations where exchange errors are important, shorter codes that explicitly address exchange errors can be effective.

A very different approach to fault tolerant computation is based on the assumption of highly correlated errors at low temperature, allowing the use of "'decoherence free subspaces" (DFS) [**14, 15**]. Shortly after [**27**] was posted, Lidar, et al observed [**16**] that the existing schemes for concatenating a DFS code with a standard 5-qubit code for correcting single-bit errors [**15**] could also correct exchange errors. This is because exchange errors on physical qubits appear as single Pauli errors on the logical bits used in DFS codes, i.e., it appears as if Pauli matrices act on the 4-qubit units which form DFS codes. Subsequently, they [**17**] turned this idea around to show how exchange interactions could be used to construct universal gates within the DFS scheme for quantum computation.

In section 2 we first review some of the basic principles underlying permutational symmetry of multi-particle quantum wave functions and then show how this leads to exchange errors. In Section 3 we discuss the issues associated with correction of exchange errors and present an explicit (non-additive) 9-bit code which can correct both exchange errors and all one-qubit errors. Section 3.4 contains an ambitious proposal for constructing powerful new codes using irreducible representations of the symmetric group.

2. The full wave function

2.1. Permutational symmetry. A (pure) state of a quantum mechanical particle with spin q corresponds to a one-dimensional subspace of the Hilbert space $= \mathbf{C}^{2q+1} \otimes L^2(\mathbf{R}^3)$ and is typically represented by a vector in that subspace. The state of a system of N such particles is then represented by a vector $\Psi(x_1, x_2, \ldots, x_N)$ in $\mathcal{H}^N$. However, when dealing with identical particles Ψ must also satisfy the Pauli principle, i.e., it must be symmetric or anti-symmetric under exchange of the coordinates $x_j \leftrightarrow x_k$ so that, e.g.,

$$\Psi(x_2, x_1, \ldots, x_N) = \pm\Psi(x_1, x_2, \ldots, x_N). \tag{2.1}$$

depending on whether the particles in question are bosons (e.g. photons) or fermions (e.g., electrons). In either case, we can write the full wave function in the form

$$\Psi(x_1, x_2, \ldots, x_N) = \sum_k \chi_k(s_1, s_2, \ldots, s_N)\Phi_k(\mathbf{r}_1, \mathbf{r}_2, \ldots, \mathbf{r}_N) \tag{2.2}$$

where the "space functions" Φ_k are elements of $L^2(\mathbf{R}^{3N})$, the "spin functions" χ_k are[2] in $[\mathbf{C}^{2q+1}]^N$ and $x_k = (\mathbf{r}_k, s_k)$ with $\mathbf{r}$ a vector in $\mathbf{R}^3$ and s_k (called the spin coordinate) an element of $\{0, 1, \ldots 2q\}$ corresponding to spin values going from $-\frac{1}{2}q$ to $+\frac{1}{2}q$ in integer steps. It is not necessary that χ and Φ each satisfy the Pauli principle; indeed, when $q = \frac{1}{2}$ so that $2q + 1 = 2$ and we are dealing with $\mathbf{C}^2$

[2]A spin state χ looks formally like a (possibly entangled) N-qubit state. However, unlike qubits which involve an implicit spatial component, we want only vectors in $[\mathbf{C}^{2q+1}]^N$ itself.

it is *not* possible for χ to be anti-symmetric when $N \geq 3$.[3] Instead, we expect that χ and Φ satisfy certain duality conditions which guarantee that Ψ has the correct permutational symmetry. In the case of anti-symmetric functions there is an extensive literature, e.g., [**18, 19**], about functions in which the χ_k and Φ_k are bases for irreducible representations of S_n with dual Young tableaux.

With this background, we now restrict attention to the important special case in which $q = \frac{1}{2}$ yielding two spin states labeled[4] so that $s = +\frac{1}{2}$ corresponds to $|0\rangle$ and $s = -\frac{1}{2}$ corresponds to $|1\rangle$, and the particles are electrons so that Ψ must be anti-symmetric.

To emphasize the distinction between a pure spin state as an element of $\mathbf{C}^2$ and a spin associated with a particular qubit or spatial wave function, we will replace $|0\rangle$ and $|1\rangle$ by $\uparrow$ and $\downarrow$ respectively. The notation $|01\rangle$ then describes a two-qubit state in which the particle in the first qubit has spin "up" ($\uparrow$) and that in the second has spin "down" ($\downarrow$). What does it mean for a particle to "be" in a qubit? A reasonable answer is that each qubit is identified by the spatial component of its wave function $f_A(\mathbf{r})$ where $A, B, C \ldots$ label the qubits and wave functions for different qubits are orthogonal. Thus, if the qubits did not correspond to identical particles we would have $|01\rangle = f_A(\mathbf{r_1})\uparrow\ f_B(\mathbf{r_2})\downarrow$. In the more realistic situation of identical particles

$$|01\rangle = \frac{1}{\sqrt{2}}\big(f_A(\mathbf{r_1})\uparrow\ f_B(\mathbf{r_2})\downarrow \pm f_B(\mathbf{r_1})\downarrow\ f_A(\mathbf{r_2})\uparrow\big). \tag{2.3}$$

with the plus sign (+) for bosons and the minus sign (−) for fermions. We will henceforth consider the special case of electrons, which are fermions, in which case the antisymmetric function given by (2.3) is called a Slater determinant. Note that a function of the form (2.3) has the important property that the electron whose spatial function is f_A always has spin "up" regardless of whether its coordinates are labeled by 1 or 2. Although (2.3) is not a simple product, but a special type of superposition which is the anti-symmetrization of a product, it behaves in some ways like a product state. It should be contrasted with the a true entangled Bell state such as

$$\begin{aligned}\frac{1}{\sqrt{2}}\big[|01\rangle - |10\rangle\big] = \tfrac{1}{2}\big(&f_A(\mathbf{r_1})\uparrow\ f_B(\mathbf{r_2})\downarrow - f_B(\mathbf{r_1})\downarrow\ f_A(\mathbf{r_2})\uparrow \\ &- f_A(\mathbf{r_1})\downarrow\ f_B(\mathbf{r_2})\uparrow + f_B(\mathbf{r_1})\uparrow\ f_A(\mathbf{r_2})\downarrow\big)\end{aligned} \tag{2.4}$$

which is a superposition of two Slater determinants or four products.

It may be useful to observe that (2.4) has the form of a wave function associated with an entangled state shared by "Alice" and "Bob" when f_A describes a particle localized near Alice and f_B a particle localized near Bob, and discuss its interpretation in that situation. When Alice uses a detector in her lab to measure the spin, she also implicitly makes a measurement of the spatial function, i.e., a measurement which projects onto spatial functions localized in her lab. She may get electron #1 with spin "up" with probability $\frac{1}{4}$ or electron #2 with spin "up" with probability $\frac{1}{4}$. However, there is no physical way to distinguish these two possibilities. The net result is a measurement of spin "up" in Alice's lab with total probability $\frac{1}{2}$. The

[3]This is because an antisymmetric N-particle wave function requires at least N linearly independent one-particle functions.

[4]These labels are the reverse of the usual physicists's convention; in essence, the convention in quantum computation is to label the eigenvectors of σ_z so that the eigenvalue $= e^{i\,\text{label}}$.

other two states in the superposition would correspond to measuring some electron in her lab with spin "down", also with net probability $\frac{1}{2}$. Once Alice has made a measurement, a corresponding measurement by Bob always yields the opposite spin.

Returning to (2.3), we note that it can be rewritten in the form (2.5) as

$$|01\rangle = \tfrac{1}{\sqrt{2}}[\chi^+(s_1,s_2)\phi^-(\mathbf{r_1},\mathbf{r_2}) + \chi^-(s_1,s_2)\phi^+(\mathbf{r_1},\mathbf{r_2})] \tag{2.5}$$

where $\chi^\pm = \frac{1}{\sqrt{2}}\,[\uparrow\downarrow \pm \downarrow\uparrow]$ denote the indicated Bell-like spin states and

$$\phi^\pm = \frac{1}{\sqrt{2}}\,[f_A(\mathbf{r_1})f_B(\mathbf{r_2}) \pm f_B(\mathbf{r_1})f_A(\mathbf{r_2})]\,.$$

We emphasize again that the reduction to a simple expression of the form (2.5), in which each term in the product is either an *antisymmetric* spin function times a *symmetric* spatial functions or vice-versa, is possible only[3] when $N = 2$. For more than two electrons, more complex expressions, of the form (2.2) are needed.

2.2. The origin of Pauli exchange errors. We now describe the origin of Pauli exchange errors by analyzing the two-qubit case in detail, under the additional simplifying assumption that the Hamiltonian is spin-free. This is certainly not realistic; quantum computers based upon spin will involve magnetic fields and hence, not be spin-free. However, the assumption of a spin-free Hamiltonian H, merely implies that the time development of (2.3) is determined by $e^{-iHt}\phi^\pm$, and this suffices to illustrate the principles involved. With a spin-dependent Hamiltonian the time development $e^{-iHt}\chi^\pm$ would also be non-trivial.

We will also assume that the qubits are formed using charged particles, such as electrons or protons, so that H includes a term corresponding to the $\frac{1}{r_{12}} \equiv \frac{1}{|\mathbf{r_1}-\mathbf{r_2}|}$ long-range Coulomb interaction. The Hamiltonian will be symmetric so that the states $\phi^\pm$ retain their permutational symmetry; however, the interaction term implies that they will not retain the simple form of symmetrized (or anti-symmetrized) product states. Hence, after some time the states $\phi^\pm$ evolve into

$$\Phi^-(\mathbf{r}_1,\mathbf{r}_2) = \sum_{m<n} c_{mn}\frac{1}{\sqrt{2}}\,[f_m(\mathbf{r_1})f_n(\mathbf{r_2}) - f_n(\mathbf{r}_1)f_m(\mathbf{r}_2)] \tag{2.6a}$$

$$\Phi^+(\mathbf{r}_1,\mathbf{r}_2) = \sum_{m\le n} d_{mn}\frac{1}{\sqrt{2}}\,[f_m(\mathbf{r_1})f_n(\mathbf{r_2}) + f_n(\mathbf{r}_1)f_m(\mathbf{r}_2)]\,. \tag{2.6b}$$

where f_m denotes any orthonormal basis whose first two elements are f_A and f_B respectively. There is no reason to expect that $c_{mn} = d_{mn}$ in general. On the contrary, only the symmetric sum includes pairs with $m = n$. Hence if one $d_{mm} \neq 0$, then one must have some $c_{mn} \neq d_{mn}$. Inserting (2.6a) in (2.5) yields

$$\begin{aligned} e^{-iHt}|01\rangle &= \frac{c_{AB}+d_{AB}}{2}\big(f_A(\mathbf{r_1})\!\uparrow f_B(\mathbf{r_2})\!\downarrow - f_B(\mathbf{r_1})\!\downarrow f_A(\mathbf{r_2})\!\uparrow\big) \\ &\quad + \frac{c_{AB}-d_{AB}}{2}\big(f_B(\mathbf{r_1})\!\uparrow f_A(\mathbf{r_2})\!\downarrow - f_A(\mathbf{r_1})\!\downarrow f_B(\mathbf{r_2})\!\uparrow\big) + \Psi^{\text{Remain}} \\ &= \frac{c_{AB}+d_{AB}}{2}|01\rangle + \frac{c_{AB}-d_{AB}}{2}|10\rangle + \Psi^{\text{Remain}} \end{aligned} \tag{2.7}$$

where Ψ^{Remain} is orthogonal to $\phi^\pm$.

A measurement of qubit-A corresponds to projecting onto f_A. Hence a measurement of qubit-A on the state (2.5) yields spin "up" with probability $\frac{1}{4}|c_{AB}+d_{AB}|^2$

and spin "down" with probability $\frac{1}{4}|c_{AB} - d_{AB}|^2$. Note that the *full* wave function is *necessarily* an *entangled* state and that the measurement process leaves the system in state $|10\rangle$ or $|01\rangle$ with probabilities $\frac{1}{4}|c_{AB} \pm d_{AB}|^2$ respectively, i.e., a subsequent measurement of qubit-B always yields the opposite spin. With probability $\frac{1}{4}|c_{AB} - d_{AB}|^2$ the initial state $|10\rangle$ has been converted to $|01\rangle$.

Although the probability of this may be small, it is *not* zero. Moreover, it would seem that any implementation which provides a mechanism for two-qubit gates would not allow the qubits to be so isolated as to preclude interactions between particles in different qubits[5]. In general, one would expect qubits to be less isolated from each other than from the external environment so that the interaction between a single pair of qubits would be greater than between a qubit and a particle in the environment. However, the environment consists of a huge number of particles (in theory, the rest of the world) and it may well happen that the number of environmental particles which interact with a given qubit is several orders of magnitude greater than the number of qubits, giving a net qubit-environment interaction which is greater than a typical qubit-qubit interaction. On the other hand, the number of qubit-qubit interactions grows quadratically with the size of the computer. Thus, prototype quantum computers, using only a few qubits, may not undergo exchange errors at the same level as the larger computers needed for real computations.

It is worth emphasizing that when the implementation involves charged particles, whether electrons or nuclei, the interaction *always* includes a contribution from the $\frac{1}{r_{12}}$ Coulomb potential, which is known to have long-range[6] effects. This is true even when the interaction used to implement the gates is entirely different, and does not involve electrostatics. Screening may reduce the effective charge, but it will not, in general, remove the basic long-range behavior of the Coulomb interaction.

Precise estimates of exchange errors require more detailed models of the specific experimental implementations. The role of long-range Coulomb effects (for which exchange errors grow quadratically with the size of the computer) suggests that implementations involving neutral particles may be advantageous for minimizing exchange errors. This would include both computers based on polarized photons (rather than charged particles) and more innovative schemes, such as Briegel, et al's proposal [**3**] using optical lattices. On the other hand, the ease with which exchange errors can be corrected using appropriate 9-qubit codes, suggests that dealing with exchange interactions need not be a serious obstacle.

3. Correcting Exchange Errors

A Pauli exchange error is a special type of "two-qubit" error which has the same effect as "bit flips" if (and *only* if) they are different. Exchange of bits j and

[5]Although the gates themselves require interactions, we expect these to be short-lived and well-controlled, i.e., in a well-designed quantum computer the gates themselves should not be a significant source of error. However, the process of turning gates on and off could induce errors in other qubits. We do not consider this error mechanism.

[6]This is because even when f and g have non-overlapping compact support $[a, b]$ and $[c, d]$ respectively, such expectations as $\int\int |f(\mathbf{r_1})|^2 \frac{1}{|\mathbf{r_1}-\mathbf{r_2}|}|f(\mathbf{r_2})|^2$ will be non-zero because the integrand is non-zero on $[a, b] \times [c, d]$. Non-overlapping initial states will not prevent the system from evolving in time to one whose states are *not* simple products or (in the case of fermions) Slater determinants!

k is equivalent to acting on a state with the operator

$$E_{jk} = \tfrac{1}{2}\Big(I_j \otimes I_k + Z_j \otimes Z_k + X_j \otimes X_k + Y_j \otimes Y_k\Big) \tag{3.1}$$

where X_j, Y_j, Z_j denote the action of the Pauli matrices $\sigma_x, \sigma_y, \sigma_z$ respectively on the bit j.

3.1. Example: the 9-bit Shor code. As an example of potential difficulties with existing codes, consider the simple 9-bit code of Shor [**28**]

$$|c_0\rangle = |\mathbf{000}\rangle + |\mathbf{011}\rangle + |\mathbf{101}\rangle + |\mathbf{110}\rangle \tag{3.2a}$$

$$|c_1\rangle = |\mathbf{111}\rangle + |\mathbf{100}\rangle + |\mathbf{010}\rangle + |\mathbf{001}\rangle \tag{3.2b}$$

where boldface denotes a triplet of 0's or 1's. It is clear that these code words are invariant under exchange of electrons within the 3-qubit triples (1,2,3), (4,5,6), or (7,8,9). To see what happens when electrons in different triplets are exchanged, consider the exchange E_{34} acting on $|c_0\rangle$. This yields $|000\,000\,000\rangle + |001\,011\,111\rangle + |110\,100\,111\rangle + |111\,111\,000\rangle$ so that

$$E_{34}|c_0\rangle = |c_0\rangle + Z_8|c_0\rangle + |001\,011\,111\rangle + |110\,100\,111\rangle \tag{3.3a}$$

$$E_{34}|c_1\rangle = |c_1\rangle - Z_8|c_1\rangle + |110\,100\,000\rangle + |001\,011\,000\rangle \tag{3.3b}$$

If $|\psi\rangle = a|c_0\rangle + b|c_1\rangle$ is a superposition of code words,

$$E_{34}|\psi\rangle = \tfrac{1}{2}\Big(|\psi\rangle + Z_8|\tilde{\psi}\rangle\Big) + \frac{1}{\sqrt{2}}|\gamma\rangle$$

where $|\tilde{\psi}\rangle = a|c_0\rangle - b|c_1\rangle$ differs from ψ by a "phase error" on the code words and $|\gamma\rangle$ is orthogonal to the space of codewords and single bit errors. Thus, this code cannot reliably distinguish between an exchange error E_{34} and a phase error on any of the last 3 bits. This problem occurs because if one tries to write $E_{34}|c_0\rangle = \alpha|c_0\rangle + \beta|d_0\rangle$ with $|d_0\rangle$ orthogonal to $|c_0\rangle$, then one can not also require that $|d_0\rangle$ be orthogonal to $|c_1\rangle$.

3.2. Conditions for error correction. Before discussing specific codes for correcting exchange errors, we first review some of the basic principles of error correction. In order to be able to correct a given class of errors, we identify a set of basic errors $\{e_p\}$ in terms of which all other errors can be written as linear combinations. In the case of unitary transformations on single bit, or one-qubit errors, this set usually consists of X_k, Y_k, Z_k $(k = 1 \ldots n)$ where n is the number of qubits in the code and X_k, Y_k, Z_k now denote $I \otimes I \otimes I \ldots \otimes \sigma_p \otimes \ldots \otimes I$ where σ_p denotes one of the three Pauli matrices acting on qubit-k. If we let $e_0 = I$ denote the identity and $\{C_j\}$ the set of code words, then a sufficient condition for error correction is

$$\langle e_p C_i | e_q C_j\rangle = \delta_{ij}\delta_{pq} \tag{3.4}$$

However, (3.4) can be replaced [**2, 4, 9**] by the weaker condition

$$\langle e_p C_i | e_q C_j\rangle = \delta_{ij} d_{pq}. \tag{3.5}$$

where the matrix D with elements d_{pq} is independent of i, j. When considering Pauli exchange errors, it is natural to seek codes which are invariant under some subset of permutations. This is clearly incompatible with (3.4) since some of the exchange errors will then satisfy $E_{k\ell}|C_i\rangle = |C_i\rangle$. Hence we will need to use (3.5).

The most common code words have the property that $|C_1\rangle$ can be obtained from $|C_0\rangle$ by exchanging all 0's and 1's. For such codes, it is not hard to see that $\langle C_1|Z_kC_1\rangle = -\langle C_0|Z_kC_0\rangle$ which is consistent with (3.5) if and only if it is identically zero. Hence even when using (3.5) rather than (3.4) it is necessary to require

$$\langle C_1|Z_kC_1\rangle = -\langle C_0|Z_kC_0\rangle = 0 \tag{3.6}$$

when the code words have this type of $0 \leftrightarrow 1$ duality.

If the basic error set has size N (i.e., $p = 0, 1 \ldots N-1$), then a two-word code requires codes which lie in a space of dimension at least $2N$. For the familiar case of single-bit errors $N = 3n+1$ and, since an n-bit code word lies in a space of dimension 2^n, any code must satisfy $3n+1 < 2^{n-1}$ or $n \geq 5$. There are $n(n-1)/2$ possible single exchange errors compared to $9n(n-1)/2$ two-bit errors of all types. Thus, similar dimension arguments would imply that codes which can correct all one- and two-bit errors must satisfy $2N = 9n(n-1) + 2(3n+1) \leq 2^n$ or $n \geq 10$. The shortest code known [**4**] which can do this has n = 11. We will see that, not surprisingly, correcting both one-bit and Pauli exchange errors, can be done with shorter codes than required to correct all two-bit errors.

However, the dimensional analysis above need not yield the best bounds when exchange errors are involved. Consider the simple code $|C_0\rangle = |000\rangle, |C_1\rangle = |111\rangle$ which is optimal for single bit flips (but can not correct phase errors). In this case $N = n+1$ and $n = 3$ yields equality in $2(n+1) \leq 2^n$. But, since this code is invariant under permutations, the basic error set can be expanded to include all 6 exchange errors E_{jk} for a total of $N = 10$ without increasing the length of the code words.

3.3. Permutationally invariant codes. We now present a 9-bit code code which can handle both Pauli exchange errors and all one-bit errors. It is based on the realization that codes which are invariant under permutations are impervious to Pauli exchange errors. Let

$$|C_0\rangle = |000\,000\,000\rangle + \frac{1}{\sqrt{28}} \sum |111\,111\,000\rangle \tag{3.7a}$$

$$|C_1\rangle = |111\,111\,111\rangle + \frac{1}{\sqrt{28}} \sum |000\,000\,111\rangle \tag{3.7b}$$

where $\sum$ denotes the sum over all permutations of the indicated sequence of 0's and 1's and it is understood that we count permutations which result in identical vectors only once. This differs from the 9-bit Shor code in that *all* permutations of $|111\,111\,000\rangle$ are included, rather than only three. The normalization of the code words is

$$\langle C_i|C_i\rangle = 1 + \frac{1}{28}\binom{9}{3} = 4\,.$$

The coefficient $1/\sqrt{28}$ is needed to satisfy (3.6). Simple combinatorics implies

$$\langle C_i|Z_kC_i\rangle = (-1)^i \left[1 - \frac{1}{3}\binom{9}{3}\frac{1}{28}\right] = 0.$$

Moreover,

$$\langle Z_kC_i|Z_\ell C_i\rangle = 1 + \delta_{k\ell}\binom{9}{3}\frac{1}{28} = 1 + 3\delta_{k\ell}. \tag{3.8}$$

The second term in (3.8) is zero when $k \neq \ell$ because of the fortuitous fact that there are exactly the same number of positive and negative terms. If, instead, we had used all permutations of κ 1's in n qubits, this term would be $\frac{(n-2\kappa)^2 - n}{n(n-1)}\binom{n}{\kappa}$ when $k \neq \ell$.

Since all components of $|C_0\rangle$ have 0 or 6 bits equal to 1, any single bit flip acting on $|C_0\rangle$, will yield a vector whose components have $1, 5$, or 7 bits equal to 1 and is thus orthogonal to $|C_0\rangle$, to $|C_1\rangle$, to a bit flip acting on $|C_1\rangle$ and to a phase error on either $|C_0\rangle$ or $|C_1\rangle$. Similarly, a single bit flip on $|C_1\rangle$ will yield a vector orthogonal to $|C_0\rangle$, to $|C_1\rangle$, to a bit flip acting on $|C_0\rangle$ and to a phase error on $|C_0\rangle$ or $|C_1\rangle$. This suffices to ensure that (3.4), and hence (3.5), holds if e_p is I or some Z_k and e_q is one of the X_ℓ or Y_ℓ.

However, single bit flips on a given code word need not be mutually orthogonal. To find $\langle X_k C_i | X_\ell C_i\rangle$ when $k \neq \ell$, consider

$$\langle X_k\,(\nu_1\nu_2\ldots\nu_9)\,|\,X_\ell\,(\mu_1\mu_2\ldots\mu_9)\rangle. \tag{3.9}$$

where ν_i, μ_i are in $0, 1$. This will be nonzero only when $\nu_k = \mu_\ell = 0,\ \nu_\ell = \mu_k = 1$ or $\nu_k = \mu_\ell = 1,\ \nu_\ell = \mu_k = 0$ and the other $n-2$ bits are equal. From $\sum$ with κ of n bits equal to 1, there are $2\binom{n-2}{\kappa-1}$ such terms. Thus, for the code (3.7), there are 42 such terms which yields an inner product of $\frac{42}{28} = \frac{3}{2}$ when $k \neq \ell$. We similarly find that

$$\langle Y_k C_i | X_\ell C_i\rangle = -i\langle X_k Z_k C_i | X_\ell C_i\rangle = 0 \ \text{ for all } k \neq \ell$$

because exactly half of the terms analogous to (3.9) will occur with a positive sign and half with a negative sign, yielding a net inner product of zero. We also find

$$\langle Y_k C_i | X_k C_i\rangle = -i\langle X_k Z_k C_i | X_k C_i\rangle = -i\langle Z_k C_i | C_i\rangle = 0$$

so that

$$\langle Y_k C_i | X_\ell C_i\rangle = 0 \ \text{ for all } k, \ell.$$

These results imply that (3.5) holds and that the matrix D is block diagonal with the form

$$D = \begin{pmatrix} D_0 & 0 & 0 & 0 \\ 0 & D_X & 0 & 0 \\ 0 & 0 & D_Y & 0 \\ 0 & 0 & 0 & D_Z \end{pmatrix} \tag{3.10}$$

where D_0 is the 37×37 matrix corresponding to the identity and the 36 exchange errors, and D_X, D_Y, D_Z are 9×9 matrices corresponding respectively to the X_k, Y_k, Z_k single bit errors. One easily finds that $d^0_{pq} = 4$ for all p, q so that D_0 is is a multiple of a one-dimensional projection. The 9×9 matrices D_X, D_Y, D_Z all have $d_{kk} = 4$ while for $k \neq \ell$, $d_{k\ell} = 3/2$ in D_X and D_Y but $d_{k\ell} = 1$ in D_Z. Orthogonalization of this matrix is straightforward. Since D has rank $28 = 3\cdot 9 + 1$, we are using only a $54 < 2^6$ dimensional subspace of our 2^9 dimension space.

The simplicity of codes which are invariant under permutations makes them attractive. However, there are few such codes. All code words must have the form

$$\sum_{\kappa=0}^{n} a_\kappa \sum |\,\underbrace{1\ldots1}_{\kappa}\underbrace{0\ldots0}_{n-\kappa}\rangle. \tag{3.11}$$

Condition (3.5) places some severe restrictions on the coefficient a_κ. For example, in (3.7) only a_0 and a_6 are non-zero in $|C_0\rangle$ and only a_3 and a_9 in $|C_1\rangle$. If we try to change this so that a_0 and a_3 are non-zero in $|C_0\rangle$, i.e.,

$$|C_0\rangle = a_0|000\,000\,000\rangle + a_3 \sum |111\,000\,000\rangle \tag{3.12a}$$

$$|C_1\rangle = a_9|111\,111\,111\rangle + a_6 \sum |000\,111\,111\rangle \tag{3.12b}$$

then it is *not* possible to satisfy (3.6).

The 5-bit error correction code in [**2, 13**] does not have the permutationally invariant form (3.11) because the code words include components of the form $\sum \pm|11000\rangle$, i.e., not all terms in the sum have the same sign. The non-additive 5-bit error *detection* code in [**24**] also requires changes in the $\sum \pm|10000\rangle$ term. Since such sign changes seem needed to satisfy (3.6), one would not expect that 5-bit codes can handle Pauli exchange errors. In fact, Rains [**25**] has shown that the 5-bit error correction code is essentially unique, which implies that no 5-bit code can correct both all 1-bit errors and exchange errors. In [**27**] the possibility of 7-bit codes of the form (3.11) was raised. However, Wallach [**30**] has obtained convincing evidence that no permutationally invariant 7-bit code can correct all one-qubit errors.

3.4. Proposal for a new class of codes. Permutational invariance, which is based on a one-dimensional representation of the symmetric group, is not the only approach to exchange errors. Our analysis of (3.2) suggests a construction which we first describe in over-simplified form. Let $|c_0\rangle, |d_0\rangle, |c_1\rangle, |d_1\rangle$ be four mutually orthogonal n-bit vectors such that $|c_0\rangle, |c_1\rangle$ form a code for one-bit errors and $|c_0\rangle, |d_0\rangle$ and $|c_1\rangle, |d_1\rangle$ are each bases of a two-dimensional representation of the symmetric group S_n. If $|d_0\rangle$ and $|d_1\rangle$ are also orthogonal to one-bit errors on the code words, then the code $|c_0\rangle, |c_1\rangle$ can correct Pauli exchange errors as well as one-bit errors. If, in addition, the vectors $|d_0\rangle, |d_1\rangle$ also form a code isomorphic to $|c_0\rangle, |c_1\rangle$ in the sense that the matrix D in (3.5) is identical for both codes, then the code should also be able to correct products of one-bit and Pauli exchange errors.

However, applying this scheme to an n-bit code requires a non-trivial irreducible representation of S_n of which the smallest has dimension $n-1$. Thus we will seek a set of $2(n-1)$ mutually orthogonal vectors denoted $|C_0^m\rangle, |C_1^m\rangle$ $(m = 1 \ldots n-1)$ such that $|C_0^1\rangle, |C_1^1\rangle$ form a code for one bit errors and $|C_0^m\rangle$ $(m = 1 \ldots n-1)$ and $|C_1^m\rangle$ $(m = 1 \ldots n-1)$ each form basis of the same irreducible representation of S_n. Such code will be able to correct *all* errors which permute qubits; not just single exchanges. If, in addition, (3.5) is extended to

$$\langle e_p C_i^m | e_q C_j^{m'} \rangle = \delta_{ij}\delta_{mm'}d_{pq} \tag{3.13}$$

with the matrix $D = \{D_{pq}\}$ independent of both i and m, then this code will also be able to correct products of one bit errors and permutation errors.

In the construction proposed above, correction of exchange and one-bit errors would require a space of dimension $2(n-1)(3n+1) \leq 2^n$ or $n \geq 9$. If codes satisfying (3.13) exist, they could correct *all* permutation errors as well as products of permutations and one-bit errors (which includes a very special subclass of 3-bit errors and even a few higher ones). Thus exploiting permutational symmetry may yield powerful new codes.

In some sense, the strategy proposed here is the opposite of that of Section 3.3 (despite the fact that both are based on representations of S_n). In Section 3.3 we sought code words with the maximum symmetry of being invariant under all permutatations. Now, we seek instead, a pair of dual code words $|C_0^1\rangle, |C_1^1\rangle$ with "minimal" symmetry in the sense that a set of generators of S_n acting on each of these code words yields an orthogonal basis for a non-trivial irreducible representation of S_n. If the code words $|C_0^m\rangle, |C_1^m\rangle$ $n = 2 \dots n-1$ can be obtained in this way, then each pair should also be a single-bit error correction code, as desired.

3.5. Non-additive codes. Most existing codes used for quantum error correction are obtained by a process [**4, 5, 7**] through which the codes can be described in terms of a subgroup, called the *stabilizer*, of the error group. Such codes are called "stabilizer codes" or "additive codes". In [**24**] an example of a non-additive 5-bit code was given, establishing the existence of non-additive codes. However, this was only an error detection code and, hence, less powerful than the 5-bit error correction code [**2, 13**] obtained using the stabilizer formalism. Subsequently, V. P. Roychowdhury and F. Vatan [**26**] showed that many non-additive codes exist; however, it was not clear how useful such codes might be.

H. Pollatsek [**21**] has pointed out that the 9-qubit code (3.7) is a non-additive code. This establishes that non-additive codes may well have an important role to play in quantum error correction, particularly in situations in which exchange errors and permutational symmetry are important. The non-additivity is immediate if one accepts that for additive codes all non-zero blocks of D consist entirely of $d_{pq} = 1$. In (3.10), this is true only for D_0 (after suitable normalization); but not for D_X, D_Y, D_Z.

Nevertheless, it may be instructive to present an argument is based on the observation that the set of vectors which occur in $\sum$ in (3.7b) spans the vector space of binary 9-tuples $\mathbf{Z}_2^9$. More generally let $\Gamma_{\kappa,n}$ denote the set of all vectors $\mathbf{a} = (a_1, a_2, \dots a_n)$ in $\mathbf{Z}_2^n$ with precisely κ of the a_j taking the value 1 and $n - \kappa$ the value 0 as in (3.11). Then span $\{\Gamma_{\kappa,n}\} = \mathbf{Z}_2^n$ if κ is odd and $\kappa \neq 0, n$. (If $\kappa \neq 0, n$ is even, $\{\Gamma_{\kappa,n}\}$ spans the even subspace of $\mathbf{Z}_2^n$.) By definition, an additive (or stabilizer) code forms an eigenspace for an abelian subgroup S of the error group $E = \{i^\ell X(\mathbf{a})Z(\mathbf{b}) : \mathbf{a}, \mathbf{b} \in \mathbf{Z}_2^9\}$. When the stabilizer S consists of only the scalar multiples of the identity I, then the corresponding eigenspace is all of $\mathbf{C}^{2^9}$. Consequently, to show that (3.7) is *not* a stabilizer code, it suffices to show that no vector of the subspace spanned by the codewords $|C_0\rangle$ and $|C_1\rangle$ can be an eigenvector for an element of E other than I.

It suffices to consider the image of $|C_1\rangle$ under $X(a)Z(b)$, i.e.,

$$X(\mathbf{a})Z(\mathbf{b})|C_1\rangle = (-1)^{\mathbf{b}\cdot\mathbf{e}}|\mathbf{a}+\mathbf{e}\rangle + \frac{1}{\sqrt{28}} \sum_{\mathbf{v}\in\Gamma_{3,9}} (-1)^{\mathbf{b}\cdot\mathbf{v}} |\,\mathbf{v}+\mathbf{a}\,\rangle$$

where $\mathbf{e} = (111\,111\,111)$. If $X(\mathbf{a})Z(\mathbf{b})|C_1\rangle = \lambda|C_1\rangle$, we must have, in particular, $X(\mathbf{a})Z(\mathbf{b})|\mathbf{e}\rangle = \lambda|\mathbf{e}\rangle$ which implies $\mathbf{a} = (000\,000\,000)$ or, equivalently, $X(\mathbf{a}) = I$. Then $\lambda = (-1)^{\mathbf{b}\cdot\mathbf{e}}$ and the eigenvalue equation reduces to

$$Z(\mathbf{b})|C_1\rangle = \lambda \left(|\mathbf{e}\rangle + + \frac{1}{\sqrt{28}} \sum_{\mathbf{v}\in\Gamma_{3,9}} (-1)^{\mathbf{b}\cdot(\mathbf{v}+\mathbf{e})} |\,\mathbf{v}\,\rangle \right)$$

which implies $\mathbf{b}\cdot(\mathbf{v}+\mathbf{e}) = 0$ for every $\mathbf{v} \in \Gamma_{3,9}$. But since (as noted above) $\Gamma_{3,9}$ spans $\mathbf{Z}_2^9$ this implies that $\mathbf{b}$ is orthogonal to all of $\mathbf{Z}_2^9$, which implies $\mathbf{b} = (000\,000\,000)$ so that $Z(\mathbf{b}) = I$ as well. Thus, since any element of E can be written as a multiple of $X(\mathbf{a})Z(\mathbf{b})$, the stabilizer S contains only multiples of the identity.

4. Conclusion

Although codes which can correct Pauli exchange errors will be larger than the minimal 5-qubit codes obtained for single-bit error correction, this may not be a serious drawback. For implementations of quantum computers which have a grid structure (e.g., solid state or optical lattices) it may be natural and advantageous to use 9-qubit codes which can be implemented in 3×3 blocks. (See, e.g., [**3**].) However, codes larger than 9-bits may be impractical for a variety of reasons. Hence it is encouraging that both the code in section 3.3 and the construction proposed in section 3.4 do *not* require $n > 9$.

It may be worth investigating whether or not the codes proposed here can be used advantageously in combination with other schemes, particularly those [**10**] based on hierarchical nesting. Since the code in sections 3.3 and 3.4 can already handle some types of multiple errors, concatenation of one of these 9-bit codes with itself will contain some redundancy and concatenation with a 5-bit code may be worth exploring. Indeed, when exchange correlations are the prime mechanism for multi-bit errors, the need for repeated concatenation may be significantly reduced.

Construction of codes of the type proposed in Section 3.4 remains a significant challenge. However, development of such new methods of may be precisely what is needed to obtain codes powerful enough to correct multi-qubit errors efficiently, without the large size drawback of codes based on repeated concatenation.

Acknowledgment It is a pleasure to thank Professor Eric Carlen for a useful comment which started my interest in exchange interactions, Dr. Daniel Gottesman for helpful information on existing codes and error correction procedures, Professor Chris King for several helpful discussions and comments on earlier drafts, Professor Harriet Pollatsek for additional comments, discussions and permission to include her observations about the non-additivity of the 9-bit code presented here, Professor Nolan Wallach for communications about 7-bit codes, and the five anonymous referees of *Physical Review Letters* for their extensive commentary on [**27**].

Note added in proof: Recent work with H. Pollatsek suggests that condition (3.13) is too strong and the simultaneous orthogonality conditions cannot be satisfied. Other methods for using codes which involve $(n-1)$-dimensional representations of S_n are under investigation.

References

[1] D. Aharanov, M. Ben-Or, "Fault-Tolerant Quantum Computation With Constant Error Rate" lanl preprints quant-ph/9611025 and quant-ph/9906129.

[2] C.H. Bennett, D.P. DiVincenzo, J.A. Smolin and W.K. Wooters, "Mixed State Entanglement and Quantum Error Correction" *Phys. Rev. A* **54**, 3824–3851 (1996) [lanl preprint quant-ph/9604024].

[3] H.J. Briegel, T. Calarco, D. Jaksch, J.I. Cirac, and P. Zoller "Quantum computing with neutral atoms" lanl preprint quant-ph/9904010.

[4] R. Calderbank, E.M. Rains, P.W. Shor and N.J.A. Sloane, "Quantum Error Correction and Orthogonal Geometry" *Phys. Rev. Lett.* **78**, 405–408 (1997) [lanl preprint quant-ph/9605005]; and "Quantum Error Correction via Codes over GF(4)" *IEEE Trans. Info. Theory* **44**, 1369–1387 (1998) [lanl preprint quant-ph/9608006].
[5] D. Gottesman "Stabilizer Codes and Quantum Error Correction" PhD thesis, Caltech (1997). [lanl preprint quant-ph/9705052].
[6] D. Gottesman "A Theory of Fault-Tolerant Quantum Computation" *Phys.Rev. A* **57**, 127–(1998) [lanl preprint quant-ph/9702029].
[7] D. Gottesman, "An Introduction to Quantum Error Correction," in "Quantum Computation: A Grand Mathematical Challenge for the Twenty-First Century and the Millennium," edited by Samuel J. Lomonaco, Jr, Proceedings of the Symposia of Applied Mathematics, Volume 58, American Mathematical Society, Providence, RI, (2002).
[8] A. Y. Kitaev "Fault-tolerant Quantum Computation by Anyons" lanl preprint quant-ph/9707021.
[9] E. Knill and R Laflamme, "A Theory of Quantum Error-Correcting Codes" *Phys. Rev. A* **55**, 900-911 (1997).
[10] E. Knill and R Laflamme, "Concatenated Quantum Codes" lanl preprint quant-ph/9608012.
[11] E. Knill, R Laflamme, and L. Viola "Theory of Quantum Error Correction for General Noise" *Phys. Rev. Lett.* **84**, 25254–28 (2000) [lanl preprint quant-ph/9908066].
[12] E. Knill, R Laflamme, W. H. Zurek "Resilient Quantum Computation: Error Models and Thresholds" *Proc. Roy. Soc. A* **454**, 365–384 (1998). [lanl preprint quant-ph/9702058]
[13] R. Laflamme, C. Miquel, J.P. Paz, W.H. Zurek, "Perfect Quantum Error Correction Code" *Phys. Rev. Lett.* **77**, 198–201 (1996).
[14] D.A. Lidar, I.L. Chuang, and K.B. Whaley "Decoherence Free Subspaces for Quantum Computation" *Phys. Rev. Lett.* **81**, 2594–97 (1998) [lanl preprint quant-ph/9807004].
[15] D.A. Lidar, D. Bacon, and K.B. Whaley "Concatenating Decoherence Free Subspaces with Quantum Error Correcting Codes" *Phys. Rev. Lett.* **82**, 4556–59 (1999) [lanl preprint quant-ph/9809081].
[16] D.A. Lidar, J. Kempe, D. Bacon, and K.B. Whaley "Protecting Quantum Information Encoded in Decoherence Free States Against Exchange Errors" *Phys. Rev. A* **61**, 052307 (2000) [lanl preprint quant-ph/0004064].
[17] D. Bacon, J. Kempe, D.A. Lidar, and K.B. Whaley, "Universal Fault-Tolerant Computation on Decoherence-Free Subspaces" *Phys. Rev. Lett.* **85**, 1758-61 (2000) [lanl preprint quant-ph/9909058]; and J. Kempe, D. Bacon, D.A. Lidar, and K.B. Whaley, " Theory of Decoherence-Free Fault-Tolerant Universal Quantum Computation" lanl preprint quant-ph/0004064.
[18] M. Hamermesh, *Group Theory* (Addison-Wesley Publishing, 1962).
[19] L. Landau and L. Lifshitz, *Quantum Mechanics* (Second edition of English translation, Pergamon Press, 1965).
[20] M.A. Nielsen and I.L. Chuang, *Quantum Computation and Quantum Information* (Cambridge University Press, 2000).
[21] H. Pollatsek, private communication
[22] J. Preskill, "Reliable Quantum Computers" *Proc. Roy. Soc. A* **454**, 385–410 (1998) [lanl preprint quant-ph/9705031], and "Fault-Tolerant Quantum Computation " lanl preprint quant-ph/9712048
[23] J. Preskill, "Battling Decoherence: The Fault Tolerant Quantum Computer" *Physics Today* (6)**52**, 24–30 (June, 1999).
[24] E.M. Rains, R. H. Hardin, P.W. Shor and N.J.A. Sloane, "A nonadditive quantum code" *Phys. Rev. Lett.* **79**, 953–954 (1997).
[25] E.M. Rains, "Quantum Codes of Minimum Distance Two" lanl preprint quant-ph/9704043
[26] V. P. Roychowdhury and F. Vatan, "On the Structure of Additive Quantum Codes and the Existence of Nonadditive Codes" lanl preprint quant-ph/9710031
[27] M.B. Ruskai, "Pauli Exchange Errors in Quantum Computation" *Phys. Rev. Lett.* **85**, 194–197 (2000); [lanl preprint quant-ph/9906114]
[28] P. Shor, "Scheme for Reducing Decoherence in Quantum Computer Memory" *Phys. Rev. A* **52**, 2493-2496 (1995).
[29] A.M. Steane, "Efficient Fault-tolerant Quantum Computing" *Nature* **399**, 124–126 (May 1999).
[30] N. Wallach, private communication.

(Mary Beth Ruskai) Department of Mathematics, University of Massachusetts Lowell, Lowell, MA 01854 USA
E-mail address, Mary Beth Ruskai: bruskai@cs.uml.edu