Quantum error-correcting codes associated with graphs

D. Schlingemann and R. F. Werner
Institut für Mathematische Physik, TU Braunschweig, Mendelssohnstrasse 3, 38106 Braunschweig, Germany
(Received 21 December 2000; published 11 December 2001)

We present a construction for quantum error correcting codes. The basic ingredients are a graph and a finite Abelian group, from which the code can explicitly be obtained. We prove necessary and sufficient conditions for the graph such that the resulting code corrects a certain number of errors. This allows a simple verification of the one-error correcting property of codes of length 5 in any dimension. As examples, we construct a large class of maximum distance separable codes, i.e. codes saturating the Singleton bound, as well as a code of length 10 detecting three errors.

DOI: 10.1103/PhysRevA.65.012308 PACS number(s): 03.67.−a

I. INTRODUCTION

From the beginning of quantum-information theory it was recognized that error-correcting codes play a crucial role. On the one hand it was clear that without error correction, decoherence effects could easily annihilate the gain in computing time promised by the new fast quantum algorithms. On the other hand, the no-cloning theorem [1] seemed to forbid at least the most naive approach to classical error correction for noisy channels, e.g., sending each bit three times and taking a majority vote at the output of the channel. Clearly, this simple scheme reduces classical errors with small probability of order ε to order ε^2. The cloning required for sending ''the same bit'' three times rules out direct quantum analogs of this scheme. It was therefore an important step to realize [2] that quantum mechanics had other, more subtle, ways of ''distributing'' quantum information over several channels to stabilize against errors. One problem with the known schemes of quantum error correction (e.g., [3,4]), however, is that they tend to be subtle indeed, and the verification of their error-correcting capabilities often requires a lengthy computation. It is therefore desirable to find new, perhaps simpler ways of constructing error-correcting codes, on which more direct intuitions might be built.

In this paper we propose a scheme for constructing quantum error-correcting codes, which has some of these advantages. The ingredients of our construction are a graph and a finite Abelian group. The order of the group determines the type of systems for which errors are corrected so that, e.g., the two-element group corresponds to the qubit case (compare [5–8] for other constructions of nonbinary codes). Concerning the graph, there are two different kinds of vertices. The input vertices, labeling the logical systems that we wish to encode, and the output vertices, labeling the physical systems in which the information, carried by the logical systems, is encoded. From the edges of the graph one can then read off an explicit expression for the code. However, not every graph corresponds to a good code, and we will discuss the condition for the code to correct a certain number of errors. In the simplest case, the fivefold code [9–11] (for qubits as well as higher-dimensional systems), it can be verified in a few lines that any two errors are detected. We also give an example of a more complex tenfold code detecting three errors.

As we are going to discuss in a following paper in more detail, it turns out that the codes that can be achieved by our method are *stabilizer codes*. There are various efficient methods for constructing stabilizer codes [3,4,12–16]. However, we think that, compared to previous stabilizer constructions, our technique has some interesting features.

(1) Often the condition for error correction can be proved for many groups simultaneously, so that one gets *code families* for systems of variable sizes.

(2) The geometric intuitions about graphs may become helpful for finding new constructions.

(3) Our codes have the property that in their natural basis all matrix elements of the coding operator have the same modulus (Hadamard form). This is helpful to the usual goal of getting a compact expression for the code. Contrary to most of the existing examples, our codes have only nonvanishing matrix elements. However, by applying a discrete Fourier transform to an appropriate set of outputs, one obtains zero matrix elements for the resulting (equivalent) code. Moreover, the Hadamard form appears to be an interesting normal form for the codes.

(4) For some codes it is possible to exchange some input vertices with some output vertices while retaining the error-correction property. This kind of symmetry is much harder to see in the usual stabilizer constructions, and may prove to be helpful in coding problems with additional inputs and outputs, such as the internal state of the coding device in convolutional codes.

The paper is organized as follows. We begin by describing the general construction of the coding operator in Sec. II. In Sec. III we recapitulate the Knill-Laflamme condition for error correction and adapt it to our particular type of codes, resulting in a necessary and sufficient condition for a graph to generate a quantum error-detecting code. The remaining sections contain examples of codes constructed in this way. In Sec. IV we show that it becomes simple indeed to verify the fivefold quantum codes. In Sec. V we demonstrate that for a given number e of errors and number k of inputs, there is a graph generating an infinite code family using $4e+k$ output systems, i.e., a family of codes saturating the singleton bound. Finally, in Sec. VI we construct a code with one input and ten outputs, detecting three errors for arbitrary system size.

II. BASIC CONSTRUCTION

Every code we construct is completely determined by the follow ingredients.

(1) An undirected graph Γ with two kinds of vertices: We distinguish the set X of input vertices and the set Y of output vertices. The links of the graph are given by the *coincidence matrix* of the graph, which we will denote by Γ for short. Its matrix element $\Gamma(z_1,z_2)$ is 1 iff the vertices $z_1,z_2 \in (X\cup Y)$ are linked, and 0 otherwise. More generally, we allow *weighted graphs*, whose incidence matrices have arbitrary integer entries, apart from the constraints $\Gamma(z_1,z_2)=\Gamma(z_2,z_1)$ and $\Gamma(z,z)=0$.

(2) A finite Abelian group G with a nondegenerate symmetric *bicharacter*.

By definition, a bicharacter is a function $\chi: G\times G\to\mathbb{C}$ such that $\chi(g+h,g')=\chi(g,g')\chi(h,g')$ and a similar condition holds for the second argument, which is also implied by the assumed symmetry $\chi(g,g')=\chi(g',g)$. We also assume nondegeneracy in the sense that

$$\sum_g \chi(g,g')=|G|\,\delta(g')\equiv\begin{cases}|G| & \text{for } g'=0\\ 0 & \text{for } g'\neq 0.\end{cases} \tag{1}$$

Note that since every $g\in G$ has finite order, $\chi(g,g')$ is always a root of unity, and $\overline{\chi(g,g')}=\chi(-g,g')$. In particular, a nondegenerate bicharacter χ corresponds to an isomorphism $\phi: G\to G^\wedge$ from G onto the group $G^\wedge$ of characters of G, where χ satisfies $\chi(g,g')=\phi(g)(g')$. For $G=\mathbb{Z}_p$, the cyclic group of order p, the standard bicharacter is given by

$$\chi(g,h)=\exp\left(\frac{2\pi i}{p}gh\right), \tag{2}$$

where g,h are integers representing their class modulo p. Since every finite Abelian group is a direct product of cyclic groups, this also shows the existence of bicharacters for any such group.

The input and output systems of the code are labeled by X and Y. They are all of the same type, i.e., they are described by the same Hilbert space $\mathcal{H}=L^2(G)$. This is the space of all functions $\psi: G\to\mathbb{C}$ with scalar product $\langle\phi,\psi\rangle=|G|^{-1}\Sigma_g\overline{\phi(g)}\psi(g)$. For compactness of notation we write such normalized sums as integrals. Hence the scalar product becomes $\int dg\,\overline{\phi(g)}\psi(g)$. The combined input system is thus described in the $|X|$-fold tensor product $\mathcal{H}^{\otimes X}=L^2(G^X)$. Vectors in this space are functions of $|X|$ variables, one variable g_z for every $z\in X$. The entire collection of variables will be denoted by g^X. The error-correcting code will be an isometry

$$\mathbf{v}_\Gamma: L^2(G^X)\to L^2(G^Y), \tag{3}$$

$$(\mathbf{v}_\Gamma\psi)(g^Y)=\int dg^X \mathbf{v}_\Gamma[g^{X\cup Y}]\psi(g^X), \tag{4}$$

where $\mathbf{v}_\Gamma$ under the integral denotes the integral kernel of the operator $\mathbf{v}_\Gamma$. This kernel depends on both input and output variables, which are combined into a collection of variables g_z, one for each vertex $z\in X\cup Y$ of the graph. The core of our construction is an explicit expression for this integral kernel: apart from an overall normalization factor, it will simply be a product of phases with each factor corresponding to a link of the graph,

$$\mathbf{v}_\Gamma[g^{X\cup Y}]=|G|^{|X|/2}\prod_{\{z,z'\}}\chi(g_z,g_{z'})^{\Gamma(z,z')}, \tag{5}$$

where the product is taken all over two elementary subsets $\{z,z'\}\subset(X\cup Y)$. Allowing direct loops within the graph, i.e., vertices connected with themselves, the incidence matrix $\Gamma=D+\Gamma'$ can be split into a diagonal part D and a part Γ' corresponding to a graph with no direct loops. However, the diagonal part can be eliminated by a local unitary transformation to the code and we may assume without loss of generality that there are no direct loops within the graph. Furthermore, the links between input vertices do not affect the error-correcting capabilities of the code. They can be eliminated by a unitary transformation leaving the protected subspace invariant.

Thus for an ordinary graph [$\Gamma(z,z')=0,1$], the right-hand side of Eq. (5) is the product of all $\chi(g_z,g_{z'})$ for which z and z' are linked. The remarkable property of such codes is that apart from the normalization factor the kernel is everywhere of modulus 1. When $G=\mathbb{Z}_p$ is cyclic and χ is given by Eq. (2), we can write the phase in a more compact form as

$$\mathbf{v}_\Gamma[g^{X\cup Y}]=|G|^{|X|/2}\exp\left(\frac{\pi i}{p}g^{X\cup Y}\cdot\Gamma\cdot g^{X\cup Y}\right), \tag{6}$$

where the centered dot denotes the product of integer-valued matrices and vectors. Note that every term in the sum $g^{X\cup Y}\cdot\Gamma\cdot g^{X\cup Y}$ occurs twice, which we compensated by a factor 1/2.

This completes the construction of the operator $\mathbf{v}_\Gamma$ from the defining ingredients listed at the beginning of this section. Of course, in general, this will not be an error-correcting code nor even an isometry. The conditions for this will be studied in the following section.

III. THE CONDITION FOR ERROR CORRECTION

A general characterization of quantum error-correcting codes has first been worked out by Knill and Laflamme [17]. We briefly review here the main aspects and adapt the condition in the particular case of codes constructed as in the previous section. In this theory a quantum code is an isometry $\mathbf{v}: \mathcal{H}\to\mathcal{K}$ from the ''input Hilbert space'' $\mathcal{H}$ to the ''output Hilbert space'' $\mathcal{K}$. Thus an input density operator is transformed by coding into $\mathbf{v}\rho\mathbf{v}^*$, which is a density operator on $\mathcal{K}$. The output of the coding is then passed through a noisy channel. The noise is described by a certain class of errors, which are represented by a linear subspace $\mathcal{E}$ of operators on $\mathcal{K}$. The channel is thus represented by a completely positive linear map of the form

$$\mathbf{T}(\rho)=\sum_\alpha F_\alpha\rho F_\alpha^*, \tag{7}$$

where $F_\alpha \in \mathcal{E}$ and are chosen such that the output is always normalized. The isometry $\mathbf{v}$ is said to be an error-correcting code for $\mathcal{E}$ if there is a completely positive ''recovery operator'' $\mathbf{R}$ such that

$$\mathbf{R}(\mathbf{T}(\mathbf{v}\rho\mathbf{v}^*))=\rho \tag{8}$$

for all density operators on $\mathcal{H}$. By the theory of Knill and Laflamme [17] this is equivalent to the factorization condition

$$\langle \mathbf{v}\psi_1, F_\alpha^* F_\beta \mathbf{v}\psi_2\rangle = \omega(F_\alpha^* F_\beta)\langle \psi_1,\psi_2\rangle, \tag{9}$$

where $\omega(F_\alpha^*,F_\beta)$ is a factor independent of the arbitrary vectors ψ_1,ψ_2. As in much of the literature on codes we will consider here a specific type of errors, namely, errors happening only on a small number of outputs of the code. Thus the tensor product structure $\mathcal{H}^{\otimes Y}=L^2(G^Y)$ of the output space becomes important. Let $\mathfrak{A}(E)$ denote the set of all operators on $L^2(G^Y)$, which are *localized* in $E\subset Y$, i.e., which are the tensor product of an arbitrary operator on $\mathcal{H}^{\otimes E}$ with the identity on $\mathcal{H}^{\otimes Y\setminus E}$. We say that a code *corrects e errors*, if F_α, F_β in Eq. (9) may be chosen arbitrarily in the linear span of $\cup_{|E|\leqslant e}\mathfrak{A}(E)$. Note that the operators $F_\alpha^* F_\beta$ appearing in the scalar product (9) can then be localized on arbitrary sets of $2e$ elements and any operator with such localization may be written as a linear combination of such $F_\alpha^* F_\beta$. It is therefore convenient to introduce the following terminology: we say that the code $\mathbf{v}$ *detects the error configuration* $E\subset Y$, if

$$\langle \mathbf{v}\psi_1, F\mathbf{v}\psi_2\rangle = \omega(F)\langle \psi_1,\psi_2\rangle, \tag{10}$$

for all $F\in\mathfrak{A}(E)$. Then a code corrects e errors, iff it detects all error configurations $E\subset Y$ with $|E|\leqslant 2e$.

We will now adapt these conditions to operators $\mathbf{v}_\Gamma$ of the special form (3). Consider a fixed error configuration $E\subset Y$ and let $I=Y\setminus E$. Then if F is an operator F on $\mathcal{H}^{\otimes E}$ with integral kernel $F[g^E,h^E]$, the integral kernel of $\mathbf{v}^* F\mathbf{v}$ is

$$\mathbf{v}_\Gamma^* F \mathbf{v}_\Gamma[g^X,h^X] = \int dg^E dg^I dh^E \overline{\mathbf{v}_\Gamma[g^X,g^E,g^I]} \times F[g^E,h^E]\mathbf{v}_\Gamma[h^X,h^E,g^I]. \tag{11}$$

This must be a multiple of the identity for every choice of F. Choosing, in particular, a rank-one operator $F=|g^E\rangle\langle h^E|$ we find that error detection for the configuration E is equivalent to the property that the correlation function

$$\mathbf{w}_{[\Gamma,E]}[g^{X\cup E},h^{X\cup E}] := \int dg^I \overline{\mathbf{v}_\Gamma[g^X,g^E,g^I]}\mathbf{v}_\Gamma[h^X,h^E,g^I] \tag{12}$$

factorizes in the following manner:

$$\mathbf{w}_{[\Gamma,E]}[g^{X\cup E},h^{X\cup E}] = C(g^E,h^E)\,\delta(g^X-h^X), \tag{13}$$

where $\delta(g^X)$ is defined to be 1 if $g_x=0$ for all $x\in X$ and zero otherwise, and $C(g^E,h^E)$ is a factor independent of the input variables g^X,h^X.

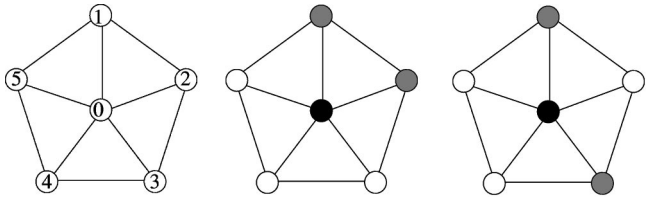

FIG. 1. The left figure shows a graph for a code of length 5, where the central vertex is the input vertex. The two figures on the right side are the relevant two-error configurations.

For two subsets K,L of $X\cup Y$ we denote by Γ_L^K the group homomorphism from G^L to G^K that can be derived from the corresponding submatrix of the incidence matrix Γ by the prescription

$$\Gamma_L^K g^L := \left[\sum_{l\in L}\Gamma(k,l)g_l\right]_{k\in K} \tag{14}$$

and the following condition is necessary and sufficient for quantum error detection.

Theorem III.1. Given a finite Abelian group G and a weighted graph Γ as in the basic construction. Then an error configuration $E\subset Y$ is detected by the quantum code $\mathbf{v}_\Gamma$ iff the system of equations

$$\Gamma_{X\cup E}^I d^{X\cup E}=0 \tag{15}$$

with $I=Y\setminus E$ implying that

$$d^X=0 \quad\text{and}\quad \Gamma_E^X d^E=0. \tag{16}$$

The proof of Theorem III.1 is given in the Appendix and in view of this result we discuss some examples. Note that the condition for $\mathbf{v}_\Gamma$ being an isometry is equivalent to the detection of zero errors, which can be seen from Eq. (10) by choosing for the error operator $F=\mathbf{1}$. This means, expressed in terms of the graph, that $\Gamma_X^Y d^X=0$ implies $d^X=0$.

IV. EXAMPLE: A CODE OF LENGTH 5

The first example of an optimal quantum error-correcting code correcting *all* one-bit errors was the famous five-qubit code by Ref. [10]. The original code is not easy to verify, so it is gratifying to see that our construction produces such a code that can be verified in a few lines. Moreover, our construction works simultaneously for all groups G and is hence not restricted to qubits. Codes of length 5 for higher-dimensional systems have been constructed before [11], and if we believe a recent result by Rains [18], the qubit code is essentially unique anyway. Hence this section has a mainly illustrative character.

Consider the graph in Fig. 1 where the central vertex ''0'' is the input vertex and the remaining five are the output vertices.

We will verify the condition of Theorem III in a particularly strong form. Namely, we will show that, for every two-element error configuration E,

$$\Gamma^I_{X\cup E} d^{X\cup E} = 0 \quad \Rightarrow \quad d^{X\cup E} = 0. \tag{17}$$

It turns out that, in terms of the Knill-Laflamme condition, Eq. (17) corresponds to Eq. (10), where ω is replaced by the normalized trace and hence a code fulfilling Eq. (17) is nondegenerate.

The error configuration is a two-element subset of the output vertices $1,\ldots,4$, and for the purpose of verifying criterion (17) the input vertex 0 plays exactly the same role as an error. It is clear by symmetry that only the two configurations for $X\cup E$ shown by filled dots in Fig. 1 need to be considered. Now the condition $\Gamma^I_{X\cup E} d^{X\cup E}=0$ is a set of equations, one for each "integration vertex" $y\in I$: For each vertex y we have to sum the d_x for all vertices of $x\in X\cup E$ connected to y, and equate it to zero. (In a weighted graph, we would have to sum with coefficients given by the matrix Γ). The following is a table of equations arising in this way for the first error configuration, $X\cup E=\{0,1,2\}$:

Vertex y	Equation
3	$d_0+d_2=0$
4	$d_0=0$
5	$d_0+d_1=0$

Clearly, this implies $d_0=d_1=d_2=0$ in any Abelian group. Similarly, for the second error configuration $X\cup E=\{0,1,3\}$ we get the equations

Vertex y	Equation
2	$d_0+d_1+d_3=0$
4	$d_0+d_3=0$
5	$d_0+d_1=0$

which once again implies $d_0=d_1=d_2=0$. This concludes the verification that the code associated with the graph in Fig. 1, and an arbitrary finite Abelian group G, detects any two errors, and hence corrects one error.

In fact, we proved a little bit more than that. The essential part of the proof was to look at certain 3×3 submatrices of the 6×6 matrix Γ, namely, those corresponding to an off-diagonal block in the partition of the vertices $\{0,1,\ldots,5\}$ into two disjoint subsets $X\cup E$ and I, and to show that each such submatrix is nonsingular. Regarded in this way, it becomes irrelevant to which of the two sets in the partition the input vertex "0" belongs, so we showed that *any* vertex, even a peripheral one, may be taken as an input vertex and we still get a one-error correcting code.

This may seem like a rather strong property of the graph we chose. However, there is (exactly) one other graph with six vertices, which produces in the same way a one-error-correcting code for arbitrary choice of the input vertex and Abelian group G. This is shown in Fig. 2.

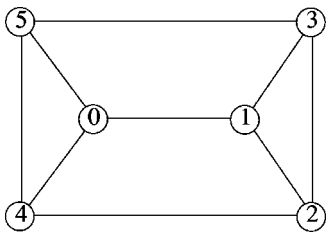

FIG. 2. An alternative graph for a code of length 5 that corrects one error.

V. CODES SATURATING THE SINGLETON BOUND

In this section we briefly discuss one natural generalization of the idea emerging in the previous section. For definiteness, let us fix G as a cyclic group of prime order d, so $\mathbb{Z}_d$ is a field, and choose an integer m. We then ask for symmetric $2m\times 2m$ matrices Γ with integer entries (or, equivalently, entries in $\mathbb{Z}_d$) with the following property: for any m-element subset $I\subset\{1,\ldots,2m\}$ the $m\times m$ submatrix Γ_{ij} with $i\in I, j\notin I$ is invertible in the field $\mathbb{Z}_d$. For the purpose of this section, let us call such a matrix *strongly error correcting* for the prime number d.

What codes can we get from such matrices? Just as in the previous section, let us specify any set X of $k<m$ vertices as input vertices, and call the remaining $n=(2m-k)$ ones output vertices. Then for any configuration of $e=m-k$ errors, the set $X\cup E$ and its complement I will have exactly m elements. By assumption, the strong form of the error-correcting condition (17) is satisfied, hence the code detects e errors. These parameters satisfy

$$n=2e+k, \tag{18}$$

i.e., the general inequality $n\leqslant 2e+k$, known as the *singleton bound* [17], is satisfied with equality. Within the present literature, the term MDS code (for maximum distance separable) is used.

How can one get strongly error-correcting matrices Γ in a practical way? Here is a procedure, we found it easy to work with small m, using a symbolic algebra program. First, we introduce variables for each matrix element Γ_{ij} with $i>j$ and compute the determinants of all off-diagonal $m\times m$ submatrices as symbolic expressions in these variables. As we go along fixing integer values for these Γ_{ij}, the determinant expressions become simpler, and in some cases factorize. Each of these factors has to be kept nonzero by the next choice of a Γ_{ij} value. Finally, we end up with an integer matrix, whose off-diagonal $m\times m$ submatrices all have nonzero integer determinants. Then, for any prime d, which does not divide any of these integers, we have solved the problem.

It is natural to begin this process by setting as many weights as possible equal to zero. It is easy to see that Theorem III does not allow too many $\Gamma(k,l)=0$ because an entire row of zeros in the matrix $\Gamma^I_{X\cup E}$ leaves one of the difference variables unconstrained. Similarly, in the condition for strong error correction it is clear that no off-diagonal submatrix should have a row of zeros, i.e., each one of the $2m$ vertices must be connected to at least m other vertices. The graph in Fig. 3 is as sparse as possible under these constraints ($m=4$), and was the starting point for a search for nonzero weights, as described above, resulting in the matrix

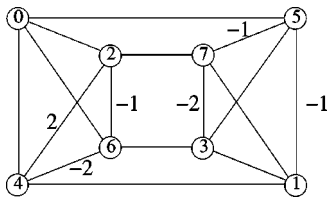

FIG. 3. Graphical representation of the weighted graph in Eq. (19). Edges without label have weight 1.

$$\Gamma=\begin{pmatrix} 0 & 0 & 1 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 & 1 \\ 1 & 0 & 0 & 0 & 2 & 0 & -1 & 1 \\ 0 & 1 & 0 & 0 & 0 & 1 & 2 & -2 \\ 1 & 1 & 2 & 0 & 0 & 0 & -2 & 0 \\ 1 & 1 & 0 & 1 & 0 & 0 & 0 & -1 \\ 1 & 0 & -1 & 2 & -2 & 0 & 0 & 0 \\ 0 & 1 & 1 & -2 & 0 & -1 & 0 & 0 \end{pmatrix}. \tag{19}$$

This matrix can either be used to get codes detecting three errors on an arbitrarily chosen single input vertex, or as a code detecting two errors (or correcting one) on two arbitrarily chosen inputs. The set of determinants is $\{-11,-8,-5,-4,-2,-1,1,2,4,5,8,9\}$, so this will work for any prime d not in the set $\{2,3,5,11\}$. By fixing the choice of the input vertices, we may restrict to a smaller set of partitions (the input vertices always belong to the same set), hence we get fewer constraints. For example, the code with input vertices $\{1,2\}$ has no relevant subdeterminant containing a factor 3, so the resulting code corrects one error on arbitrary pairs of three-level systems.

Within the above example, the number of matrix elements set to zero is maximal, namely, $\nu_i=4$ for each row $i=1,\ldots,8$. Looking at the corresponding graph, $\nu_i=4$ is just the number of lines meeting a particular vertex as one can see from Fig. 3.

Strongly error-correcting matrices exist in any dimension, so the code parameters saturating the singleton bound (18) can be chosen arbitrarily, if the dimension d of the one-site system is taken to avoid a certain finite set of primes (compare also [5]). The argument is quite simple: consider the $m\times m$ subdeterminants of symmetric $2m\times 2m$ matrices as a family of polynomials f_α, $\alpha=1,\ldots,\binom{2n}{n}$ in $\binom{2n}{2}$ variables. None of these vanishes identically and since $\mathbb{Z}$ is an integral domain, which in contrast is wrong for finite fields, the product polynomial $\Pi_\alpha f_\alpha\neq 0$ is nonzero ([19], p. 106). This implies that there exists an integer tuple Γ of arguments such that $\Pi_\alpha f_\alpha(\Gamma)\neq 0$. Thus we have the following statement.

Proposition V.1 For each number e of errors, there exists a prime d and a weighted graph Γ such that the quantum code, associated with the weighted graph Γ, is a MDS quantum error-correcting code, which encodes k d-level systems into $4e+k$ p-level systems, and which corrects e errors.

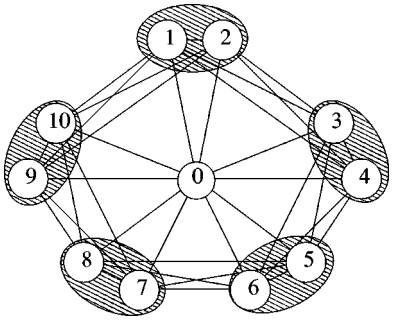

FIG. 4. The graph for a code of length 10.

VI. A QUANTUM ERROR-DETECTING CODE OF LENGTH 10

In this section we present a more complex example for a graph, which yields for every finite Abelian group G, a ten-bit code detecting three errors as given by Fig. 4. At the first look, this graph looks rather complicated, but it can be described in a simple fashion by looking at the graph for the code of length 5 in Fig. 1.

Namely, the graph, given by Fig. 4 can be obtained as follows: The output vertices $\{1,2,3,4,5\}$ of the graph in Fig. 1 are replaced by pairs $1\mapsto\{1,2\}$, $2\mapsto\{3,4\}$, $\ldots$, $5\mapsto\{9,10\}$. Each output vertex is connected with the following vertices: The central input vertex 0, the vertex belonging the same pair, and all output vertices belonging to neighbored pairs.

The symmetry of this graph can efficiently be used to check that each error configuration with three errors can be detected. As is depicted by Fig. 5, there are only four three-error configurations to distinguish.

(1) All errors occur within different pairs and all these pairs are neighbored (first graph in Fig. 5).

(2) All errors occur within different pairs and only two of these pairs are neighbored (second graph in Fig. 5).

(3) One pair is totally affected by errors and the remaining error occurs within a pair that is not neighbored (third graph in Fig. 5).

(4) One pair is totally affected by errors and the remaining error occurs within a neighbored pair (fourth graph in Fig. 5).

Proposition VI.1 For each finite Abelian group G, the quantum code, which is associated with the graph given by Fig. 5, is a quantum error-detecting code, encoding one input system into ten output systems and detecting three errors.

Proof. Suppose that each error occurs in different pairs and all these pairs are neighbored (first graph in Fig. 5), e.g., the error configuration $\{1,3,5\}$. Then we proceed as in the example for the code of length 5 to obtain the system of equations (15),

Vertices y	Equation
2	$d_0+d_1+d_3=0$
4	$d_0+d_1+d_3+d_5=0$
6	$d_0+d_3+d_5=0$
7 and 8	$d_0+d_5=0$
9 and 10	$d_0+d_1=0$

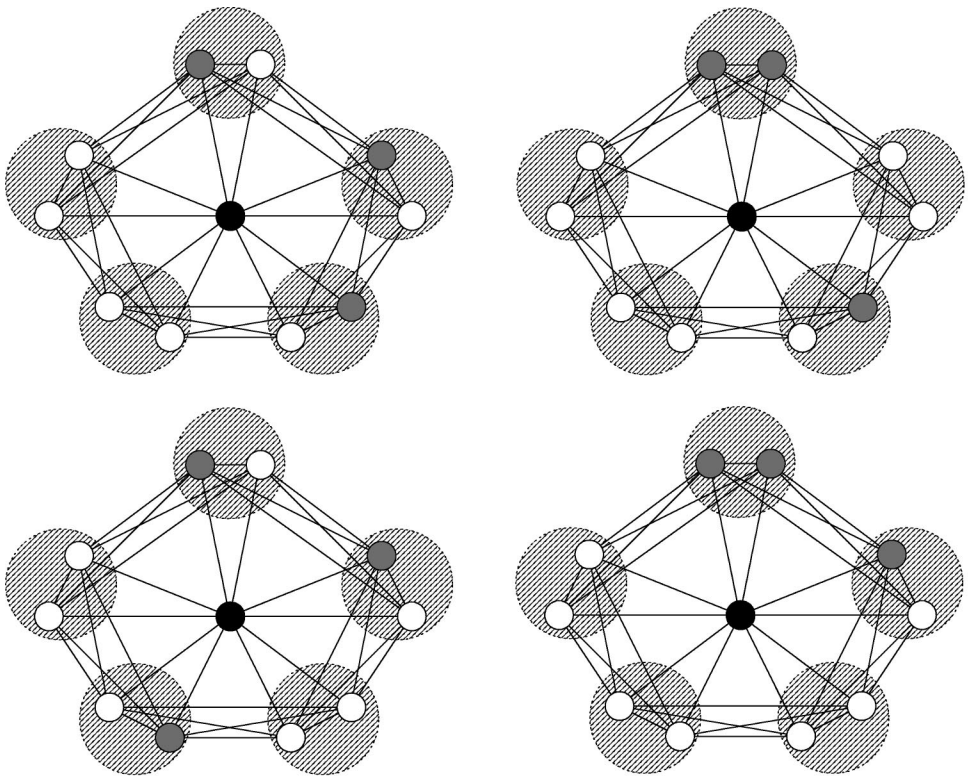

FIG. 5. A complete family of inequivalent error configurations for the graph in Fig. 3.

and we conclude that $d_0=0$ and $d_1+d_3+d_5=0$ are true. Thus Eqs. (16) are fulfilled and the corresponding error configuration is detected.

Analogously, one checks that each error configuration, where each error occurs in a different pair and only two of these pairs are neighbored (second graph in Fig. 5) is also detected.

We now consider an error configuration, where one pair $\{1,2\}$ is totally corrupted and the remaining error vertex, e.g., 5 is contained within a non-neighbored pair (third graph in Fig. 5). We obtain for the system of equations (15),

Vertices y	Equation
3 and 4	$d_0+d_1+d_2+d_5=0$
6, 7 and 8	$d_0+d_5=0$
9 and 10	$d_0+d_1+d_2=0$
7 and 8	$d_0+d_5=0$

and we conclude again that $d_0=0$ and $d_1+d_2+d_5=0$ is true. Equations (16) are fulfilled, which implies that the corresponding error configuration is also detected.

Finally, it is a bit more straightforward as in the previous case to show that an error configuration, where two errors occur within one pair and the remaining error occurs within a neighbored pair (fourth graph in Fig. 5), is detected. ■

VII. CONCLUSION AND OUTLOOK

In this paper we had to limit the exploration of our construction scheme to a few examples. A more systematic investigation is, of course, under way. Some of the issues in this investigation are the following.

(1) We have already mentioned that all codes constructed from graphs are stabilizer codes. This is verified by explicitly constructing a group of unitaries, composed out of shifts and multiplication by characters, leaving all vectors in the code space invariant. The converse of this statement is not so clear, i.e., how to embed the usual stabilizer code constructions into our scheme, and to characterize the subset of codes for which this is possible.

(2) We have seen in Sec. IV that different graphs generate five-qubit codes, although such codes are presumably unique up to local transformations. It would be helpful to characterize the local unitary transformations taking one graph code into another, and to study the relationships between the resulting graphs. The Rains invariants [20–22] for graph codes can be computed relatively easily, and should help to decide such isomorphism issues.

(3) From Sec. V it is clear that the singleton bound becomes easier to satisfy as the dimension d of the single-system Hilbert space increases. This suggests the search for bounds describing the resource limitations in coding more adequately, perhaps by taking into account more detailed features of errors than just considering arbitrary errors that occur at a single output system with a fixed probability. Our

construction could be helpful for developing and testing such bounds.

(4) Nonstabilizer quantum codes can be constructed from families of stabilizer codes by taking their *union* [23], where one has to require that the protected subspaces, corresponding to the codes within the family, are mutually orthogonal and that this property remains valid after error operations. Examples of such nonstabilizer codes are given in [24,25]. In view of our construction scheme, it would be desirable to find sufficient conditions for a family of graphs such that the union of their corresponding graph codes yields a (possibly more efficient) nonstabilizer code.

(5) After the first submission, we realized that there is a direct relation between our graph codes and the *cluster states*, which has been introduced by Briegel and Raussendorf [26]. In view of their considerations, cluster states can be used for performing a quantum computational process by a sequence of local von Neumann measurements. For storing quantum information within a cluster state, graph codes can directly be implemented for protection against decoherence. It would be desirable to implement graph codes within the structure of a cluster state in such a manner that the conditions for fault-tolerant computation are satisfied.

ACKNOWLEDGMENTS

We would like to thank Mary Beth Ruskai for helpful discussions and for supporting this investigation with many ideas. Funding by the European Union project EQUIP (Contract No. IST-1999-11053) is gratefully acknowledged.

APPENDIX

Proof of Theorem III.1. We first compute the function $\mathbf{w}_{[\Gamma,E]}$, defined by Eq. (12). It is convenient to introduce for two subsets K,K' of $X\cup Y$ the expression

$$\chi^{\Gamma}(g^K,g^{K'}):=\prod_{\{k,k'\}:k\in K,k'\in K'}\chi(g_k,g_{k'})^{\Gamma(k,k')}, \quad \text{(A1)}$$

where the product is taken over the complete two elementary sets with one element taken from K and the other taken from K'. Hence the factor for $\{k,k'\}=\{k',k\}$ only occurs once within the product. Now we write the integrand in Eq. (12) as a product of two terms

$$\overline{\mathbf{v}_{\Gamma}[g^X,g^E,g^I]}\mathbf{v}_{\Gamma}[h^X,h^E,g^I] = |G|^{|X|}\frac{\chi^{\Gamma}(h^{X\cup E},h^{X\cup E})}{\chi^{\Gamma}(g^{X\cup E},g^{X\cup E})}\frac{\chi^{\Gamma}(h^{X\cup E},g^I)}{\chi^{\Gamma}(g^{X\cup E},g^I)}. \quad \text{(A2)}$$

Here only the last factor on the right-hand side depends on the integration variables g^I associated with the set $I=Y\backslash E$. In order to carry out the integral over one variable g_i, $i\in I$, we select the g_i-dependent part out of $\chi^{\Gamma}(h^{X\cup E},g^I)$, which is

$$\prod_{\{z,i\}:z\in X\cup E}\chi(h_z,g_i)^{\Gamma(z,i)}. \quad \text{(A3)}$$

Analogously the g_i-dependent part of $\chi^{\Gamma}(g^{X\cup E},g^I)$ is the same expression with h replaced by g. Thus the g_i-dependent part of Eq. (A2) is

$$\prod_{\{z,i\}:z\in X\cup E}\chi(g_z,g_i)^{-\Gamma(z,i)}\chi(h_z,g_i)^{\Gamma(z,i)}. \quad \text{(A4)}$$

Using the character property of $\chi(\cdot,g_i)$ we can simplify this to a single factor of the form $\chi(k,g_i)$. Explicitly,

$$k=\sum_{j\in X\cup E}\Gamma_{ij}(h_j-g_j) \quad \text{(A5)}$$

and this sum contains none of the variables associated with I because $I\cap(X\cup E)=\varnothing$. The integral over g_i then gives $\delta(k)$, and we find

$$\mathbf{w}_{[\Gamma,E]}[g^{X\cup E},h^{X\cup E}]=|G|^{|X|}\frac{\chi^{\Gamma}(h^{X\cup E},h^{X\cup E})}{\chi^{\Gamma}(g^{X\cup E},g^{X\cup E})}\times\delta(\Gamma^I_{X\cup E}(h^{X\cup E}-g^{X\cup E})). \quad \text{(A6)}$$

Our task is to establish the necessary and sufficient conditions for this to be of the form

$$C(g^E,h^E)\,\delta(h^X-g^X), \quad \text{(A7)}$$

required by Eq. (13).

Now the expression (A6) has the required property of vanishing except for $g^X=h^X$ if and only if this is already implied by the vanishing of the δ function in Eq. (A6), i.e., if and only if $d^X\neq 0$ implies $\Gamma^I_{X\cup E}(d^{X\cup E})\neq 0$. This is the first part of the condition in Theorem III.1.

From now on we assume, as we may, that $\Gamma^I_{X\cup E}(d^{X\cup E})=0$ implies $d^X=0$. Then the dependence of Eq. (A6) on the input variables g^X and h^X can be simplified. The δ function can be written as

$$\delta(\Gamma^I_{X\cup E}(h^{X\cup E}-g^{X\cup E}))=\delta(\Gamma^I_E(h^E-g^E))\delta(h^X-g^X), \quad \text{(A8)}$$

because the two expressions are equal for $h^X=g^X$, and for $h^X\neq g^X$ they both vanish by assumption.

To simplify the bicharacter quotient in Eq. (A6), we use Eq. (A1) to write

$$\chi^{\Gamma}(h^{X\cup E},h^{X\cup E})=\chi^{\Gamma}(h^X,h^X)\chi^{\Gamma}(h^E,h^X)\chi^{\Gamma}(h^E,h^E). \quad \text{(A9)}$$

With a similar decomposition of $\chi^{\Gamma}(g^{X\cup E},g^{X\cup E})$ we use the condition that wherever the δ function in Eq. (A6) is nonzero, we have $g^X=h^X$. Hence the $X-X$ factors cancel, and we can write the quotient of the $X-E$ terms as

$$\frac{\chi^{\Gamma}(h^E,h^X)}{\chi^{\Gamma}(g^E,h^X)}=\prod_{j\in X}\chi\left(\sum_{i\in E}\Gamma_{ji}(h_i-g_i),h_j\right). \quad \text{(A10)}$$

For Eq. (A6) to be of the desired form (A7) with $C(g^E,h^E)$ independent of the X variables, this expression must be independent of all $h_j, j\in X$, whenever $\delta(\Gamma_E^I(h^E-g^E))\neq 0$. But Eq. (A10) is independent of h_j if and only if $\Sigma_i\Gamma_{ji}(h_i-g_i)=0$. Hence we must have that $\Gamma_E^I(h^E-g^E)=0$ implies $\Gamma_E^X(h^E-g^E)=0$. This is the second condition from Theorem III.1, which we have thus shown to be necessary. Conversely, it is sufficient to ensure that Eq. (A10) is equal to 1, and Eq. (A6) has the desired form (A7) with

$$C(g^E,h^E):=|G|^{|X|}\frac{\chi^{\Gamma}(h^E,h^E)}{\chi^{\Gamma}(g^E,g^E)}\delta(\Gamma_E^I(h^E-g^E)). \quad \text{(A11)}$$

■

[1] W.K. Wootters and W.H. Zurek, Nature (London) **299**, 802 (1982).
[2] A.R. Calderbank and P.W. Shor, Phys. Rev. A **54**, 1098 (1996).
[3] A.R. Calderbank, E.M. Rains, P.W. Shor, and N.J.A. Sloane, Phys. Rev. Lett. **78**, 405 (1997).
[4] A.R. Calderbank, E.M. Rains, P.W. Shor, and N.J.A. Sloane, e-print quant-ph/9608006.
[5] E. Knill, Los Alamos National Laboratory Report No. LAUR-96-2717, 1996 (unpublished).
[6] E.M. Rains, e-print quant-ph/9703048.
[7] R. Matsumoto and T. Uyematsu, IEICE Trans. Fundamentals **E83-A**, 10 (2000).
[8] A. Ashikhmin and E. Knill, e-print quant-ph/0005008.
[9] C.H. Bennet D.P. DiVincenzo J.A. Smolin, and W.K. Wootters Phys. Rev. A **54**, 3824 (1996).
[10] R. Laflamme, C. Miquil, J.-P. Paz, and W.H. Zurek, Phys. Rev. Lett. **77**, 198 (1996).
[11] H.F. Chau, e-print quant-ph/9702033.
[12] D. Gottesman, Phys. Rev. A **54**, 1862 (1996).
[13] D. Gottesman, Ph.D. thesis, California Institute of Technology, 1997.
[14] M. Grassl, and T. Beth, in *Proceedings of the International Symposium on Theoretical Electrical Engineering, Magdeburg, 1999*, edited by W. Mathis and T. Schindler (Universität Magdeburg, Magdeburg, 1999).
[15] M. Grassl, and T. Beth, e-print quant-ph/9910061.
[16] M. Grassl, W. Geiselmann, and T. Beth, in *Proceedings of the AAECC-13, Honolulu, 1999*, edited by M. Fossorier, H. Imai, S. Lin, and A. Poli (Springer-Verlag, Heidelberg, 1999).
[17] E. Knill and R. Laflamme, Phys. Rev. A **55**, 900 (1997).
[18] E.M. Rains, e-print quant-ph/9704043.
[19] N. Jacobson, *Basic Concepts*, Lectures in Abstract Algebra Vol I (Springer, New York, 1964).
[20] E.M. Rains, e-print quant-ph/9612015.
[21] E.M. Rains, e-print quant-ph/9611001.
[22] E.M. Rains, e-print quant-ph/9704042.
[23] M. Grassl and T. Beth, e-print quant-ph/9703016.
[24] E.M. Rains, R.H. Hardin, P.W. Shor, and N.J.A. Sloane, e-print quant-ph/9703002.
[25] M.B. Ruskai, e-print quant-ph/0006008.
[26] H.J. Briegel and R. Raussendorf, e-print quant-ph/0010033.