PROCEEDINGS OF THE ROYAL SOCIETY A

Proc. R. Soc. A (2009) **465**, 1645–1672
doi:10.1098/rspa.2008.0439
Published online 4 March 2009

Asymmetric quantum codes: constructions, bounds and performance

By Pradeep Kiran Sarvepalli[1,*], Andreas Klappenecker[1] and Martin Rötteler[2]

[1]*Texas A&M University, College Station, TX 77843, USA*
[2]*NEC Laboratories America, Inc., 4 Independence Way, Suite 200, Princeton, NJ 08540, USA*

Recently, quantum error-correcting codes have been proposed that capitalize on the fact that many physical error models lead to a significant asymmetry between the probabilities for bit- and phase-flip errors. An example for a channel that exhibits such asymmetry is the combined amplitude damping and dephasing channel, where the probabilities of bit and phase flips can be related to relaxation and dephasing time, respectively. We study asymmetric quantum codes that are obtained from the Calderbank–Shor–Steane (CSS) construction. For such codes, we derive upper bounds on the code parameters using linear programming. A central result of this paper is the explicit construction of some new families of asymmetric quantum stabilizer codes from pairs of nested classical codes. For instance, we derive asymmetric codes using a combination of Bose–Chaudhuri–Hocquenghem (BCH) and finite geometry low-density parity-check (LDPC) codes. We show that the asymmetric quantum codes offer two advantages, namely to allow a higher rate without sacrificing performance when compared with symmetric codes and vice versa to allow a higher performance when compared with symmetric codes of comparable rates. Our approach is based on a CSS construction that combines BCH and finite geometry LDPC codes.

Keywords: quantum codes; asymmetric quantum channels; stabilizer codes; Bose–Chaudhuri–Hocquenghem codes; low-density parity-check codes; finite geometry codes

1. Introduction

In many quantum mechanical systems, the mechanisms for the occurrence of bit- and phase-flip errors are quite different. In a recent work, Ioffe & Mézard (2007) postulated that quantum error correction should take into account this asymmetry. The main argument given by Ioffe & Mézard (2007) is that most of the known quantum computing devices have relaxation times (T_1) that are approximately one to two orders of magnitude larger than the corresponding dephasing times (T_2). In general, relaxation leads to both bit- and phase-flip errors, whereas dephasing only leads to phase-flip errors. This large asymmetry between T_1 and T_2 suggests that bit-flip errors occur less frequently than phase-flip errors

* Author for correspondence (pradeep@cs.tamu.edu).

Received 29 October 2008
Accepted 30 January 2009

and a well-designed quantum code would exploit this asymmetry of errors to provide better performance. In fact, this observation and its consequences for quantum error correction, especially quantum fault tolerance, has been studied by several authors (Evans *et al.* 2007; Stephens *et al.* 2007; Aliferis & Preskill 2008).

Our goal is to construct quantum codes that exploit asymmetry. The focus of the present paper is on quantum memory and communication; at present, we do not consider the issue of fault tolerance. As a concrete illustration of this, we consider the amplitude damping and dephasing channel. For this channel, we can compute the probabilities of bit and phase flips in closed form. In particular, by giving explicit expressions for the ratio of these probabilities in terms of the ratio T_1/T_2, we show how the channel asymmetry arises.

(*a*) *Related work*

Several recent papers discuss the situation of quantum error correction in the presence of an asymmetric error model that gives a strong bias towards certain errors.

Aliferis & Preskill (2008) gave a construction based on the concatenation of a repetition code with any other quantum code to get asymmetric quantum codes for biased noise. While this has the advantage that universal fault-tolerant quantum computation is possible, we expect the codes constructed in this way to have a lower rate than the ones constructed in the present paper. On the other hand, it is not known whether the codes proposed in the present paper admit a set of universal fault-tolerant gates that preserve the channel asymmetry. Evans *et al.* (2007) used symmetric Calderbank–Shor–Steane (CSS) codes with asymmetric error-correction strategy to obtain an advantage for fault-tolerant quantum error correction over a symmetric strategy. The asymmetry in this case comes from higher frequency of syndrome measurements for the X-only generators as compared with the Z-only generators. Stephens *et al.* (2007) used a combination of a symmetric code along with an asymmetric code to achieve fault-tolerant computation. In this approach, one has to use fault-tolerant circuits to switch between the symmetric and asymmetric encodings. This idea may be applicable in the present context also, though further study is needed. But we mention that our constructions include as special cases symmetric stabilizer codes for which fault-tolerant universal computation is possible. Some of them based on low-density parity-check (LDPC) codes are of independent interest and can be useful even in the absence of an asymmetric channel.

The paper by Ioffe & Mézard (2007) is closest to our work regarding the methods used to construct asymmetric quantum codes as both employ a CSS construction having a classical LDPC code for the Z-errors and a classical Bose–Chaudhuri–Hocquenghem (BCH) code for the X-errors. However, as we show in §4*b*, we employ a different approach to construct the LDPC code that allows us to have more control on the degree profile of the LDPC code. Arguably, this gives an advantage regarding not only the structure of these codes but also their performance, since similar to the classical case, our codes do not show an error floor for very small probabilities of channel errors.

It should also be noted that the performance of symmetric quantum codes for some specific quantum codes, such as a $[[5, 1, 3]]$ or $[[7, 1, 3]]$ code, over arbitrary (not necessarily symmetric) Pauli channels has been studied by Rahn *et al.* (2002).

However, contrary to the present paper in which the goal is to exploit channel asymmetries for the code design, in this paper, the goal is to characterize the performance of a given code under variation of the error weights and to find the channel under which the given code has optimal performance. Finally, in a recent development, Fletcher *et al.* (2008) studied adaptive quantum error-correction strategies in which the optimal error-correction strategy is found for a given channel using semi-definite programming. These techniques have been applied to the case of asymmetric channels that arise from amplitude damping channels. The amplitude damping model was first studied from the point of view of quantum error correction by Leung *et al.* (1997), where an approximate quantum code that encodes one qubit into four qubits was given. This code can also be seen as an *exact* quantum error-correcting code for a certain choice of errors, namely amplitude damping errors $X+\mathrm{i}Y$ and bit-flip/phase damping errors Z (B. Zeng 2008, personal communication). There is a physically arising asymmetry between the error probabilities for these two types of errors; see also appendix A for a derivation.

(*b*) *Organization of this paper*

In §2, we provide the necessary background on quantum channels and give a motivation for asymmetric quantum channels, i.e. noise models that show a significant bias towards specific types of errors. We consider the concrete example of the noise model given by amplitude damping and dephasing. This serves as motivation of asymmetric channels and has the advantage that the amount of asymmetry in the bit flip in phase-flip probabilities can be quantified easily and can be related to physical quantities such as T_1 and T_2.

Next, in §3, we explain why our focus in this paper will be asymmetric codes that are obtained from the CSS construction. After briefly sketching the ideas of asymmetric codes, we derive some bounds on the parameters of these codes. Similar to the case of standard quantum codes, which we will refer to as symmetric quantum codes as they are not designed to exploit any bias towards one specific error, it is possible to derive good upper bounds using linear programming.

We then address the question of how to construct asymmetric quantum codes. In §4, we show that, in general, a family of nested classical codes is well suited to constructing such asymmetric quantum codes. We illustrate this for a few well-known families of codes, namely Reed–Muller (RM) codes and BCH codes. Then, we propose an alternative approach to Ioffe & Mézard (2007) and use a finite geometry LDPC code and BCH code to construct an asymmetric stabilizer code.

The performance of asymmetric codes is the subject of §5. We simulate the performance of several examples of asymmetric codes constructed by the methods described in this paper in terms of the resulting block error rate versus the probability of channel errors. Simulations are carried out for various choices of channel asymmetries and codes. We explain why a modification of the standard iterative decoding algorithms known from classical LDPC codes is required as, in the quantum case, no (soft) channel information is available. We present a suitable modification of the iterative decoding algorithm that starts with information in the check nodes and then proceeds in a similar fashion to the classical hard-decision bit-flipping algorithm.

The present paper is an expanded version of Sarvepalli *et al.* (2008) and contains some of the results presented therein.

2. Background on quantum error models

Recall that a quantum channel is a completely positive trace-preserving map (Nielsen & Chuang 2000). Such maps can be written in Kraus operator form, where the action of the channel on a given input state ρ is described as follows: $\rho \mapsto \sum_i A_i \rho A_i^\dagger$, where the completeness relation $\sum_i A_i^\dagger A_i = \mathbf{1}$ holds. The operators A_i are the Kraus operators of the channel. A special case of a channel on one qubit arises if the Kraus operators are simply given by the Pauli matrices, i.e. a state ρ is mapped to

$$(1 - p_x - p_y - p_z)\rho + p_x X\rho X + p_y Y\rho Y + p_z Z\rho Z, \tag{2.1}$$

with

$$\mathbf{1} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}; \qquad X = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}; \qquad Y = \begin{bmatrix} 0 & -\mathrm{i} \\ \mathrm{i} & 0 \end{bmatrix}; \qquad Z = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}. \tag{2.2}$$

Such a channel is called a *Pauli channel*. In a Pauli channel, one has independent probabilities p_x, p_y, p_z (subject to $p_x + p_y + p_z \leq 1$) that an input qubit in state ρ is subjected to a Pauli X, Y or Z error, respectively.

As an example of a channel that arises in the study decoherence in concrete physical systems, we consider the combined *amplitude damping and dephasing channel* $\mathcal{E}$. Important parameters for the noise process underlying this channel are the relaxation time T_1 and dephasing time T_2. Suppose the channel $\mathcal{E}$ acts on a single-qubit state $\rho = (\rho_{ij})_{i,j \in \{0,\,1\}}$ for a time t. This yields the resulting density matrix

$$\mathcal{E}(\rho) = \begin{bmatrix} 1 - \rho_{11}\mathrm{e}^{-t/T_1} & \rho_{01}\mathrm{e}^{-t/T_2} \\ \rho_{10}\mathrm{e}^{-t/T_2} & \rho_{11}\mathrm{e}^{-t/T_1} \end{bmatrix}.$$

We would like to determine the probability p_x, p_y and p_z, such that an X-, Y- or Z-error occurs in a combined amplitude damping and dephasing channel. However, it turns out that this question is not well posed, since $\mathcal{E}$ is not a Pauli channel, i.e. it cannot be written in the form (2.1). However, we can obtain a Pauli channel $\mathcal{E}_T$ by conjugating the channel $\mathcal{E}$ by Pauli matrices and averaging over the results. The channel $\mathcal{E}_T$ is called the Pauli twirl of $\mathcal{E}$ and is explicitly given by

$$\mathcal{E}_T(\rho) = \frac{1}{4} \sum_{A \in \{\mathbf{1}, X, Y, Z\}} A^\dagger \mathcal{E}(A\rho A^\dagger) A.$$

Twirling (DiVincenzo *et al.* 2002; Emerson *et al.* 2005; Dankert *et al.* 2006) is used in quantum information theory as a tool to map quantum channels to simpler ones, while preserving many interesting features of the initial channel. Twirling has been used, for instance, for the task of estimating the average fidelity of a quantum gate and for the task of determining more general features

of channels (Emerson *et al.* 2007), as well as for the task of identifying codes for general quantum channels (Silva *et al.* 2007). In our situation, we apply the Pauli twirl to map the channel $\mathcal{E}$ to a Pauli channel.

Theorem 2.1. *Given a combined amplitude damping and dephasing channel $\mathcal{E}$ as above, the associated Pauli-twirled channel is of the form*

$$\mathcal{E}_T(\rho) = (1-p_x-p_y-p_z)\rho + p_x X\rho X + p_y Y\rho Y + p_z Z\rho Z,$$

where $p_x = p_y = (1-\mathrm{e}^{-t/T_1})/4$ *and* $p_z = 1/2 - p_x - (1/2)\mathrm{e}^{-t/T_2}$. *In particular,*

$$\frac{p_z}{p_x} = 1 + 2\frac{1-\exp(t/T_1(1-T_1/T_2))}{\mathrm{e}^{t/T_1}-1}.$$

If $t \ll T_1$, *then we can approximate this ratio as* $2T_1/T_2 - 1$.

The proof of this theorem is straightforward but technical and is given in appendix A. From theorem 2.1, it follows that an asymmetry in the T_1 and T_2 times does translate into an asymmetry in the occurrence of bit- and phase-flip errors. Note that $p_x = p_y$, indicating that the Y-errors are as unlikely as the X-errors. We shall refer to the ratio p_z/p_x as the channel asymmetry and throughout the paper shall denote it by A. Please note in some papers the asymmetry is quantified in terms of the ratio $(p_z+p_y)/(p_x+p_y)$.

3. Asymmetric quantum codes: basics and bounds

Asymmetric codes use the fact that the phase errors are much more likely than the bit-flip errors or the combined bit–phase-flip errors. Therefore, the code has different error-correcting capabilities for handling different types of errors. We require the code to correct many phase errors but it is not required to handle the same number of bit-flip errors. If we assume a CSS code, then we can meaningfully speak of X- and Z-distances. A CSS stabilizer code that can detect all X-errors up to weight $d_x - 1$ is said to have an X-distance of d_x. Similarly, if it can detect all Z-errors up to weight $d_z - 1$, then it is said to have a Z-distance of d_z. We shall denote such a code by $[[n, k, d_x/d_z]]_q$ to indicate it as an asymmetric code. We could also view this code as an $[[n, k, \min\{d_x, d_z\}]]_q$ stabilizer code. Further extension of these metrics to an additive non-CSS code is an interesting problem, but we will not go into the details here. We would like to point out that Steane (1996) had earlier used a similar notation to distinguish the two distances. He also suggested that efficient codes can be designed should we be in possession of more information about the noise processes. Though the notion of asymmetric non-binary quantum codes needs further treatment, the idea of non-binary CSS is well understood. Some of the results are given for a non-binary alphabet.

We will exclusively use the CSS construction (Steane 1996; Calderbank *et al.* 1998) to construct asymmetric quantum codes. Recall that in the CSS construction a pair of codes are used, one for correcting the bit-flip errors and the other for correcting the phase-flip errors. Our choice of these codes will be such that the code for correcting the phase-flip errors has a larger distance than the code for correcting the bit-flip errors. We restate the CSS construction in a form convenient for asymmetric stabilizer codes.

Lemma 3.1 (CSS construction, Steane (1996) and Calderbank *et al.* (1998)). *Let C_x, C_z be linear codes over $\mathbb{F}_q^n$ with the parameters $[n, k_x]_q$ and $[n, k_z]_q$, respectively. Let $C_x^\perp \subseteq C_z$. Then, there exists an $[[n, k_x+k_z-n, d_x/d_z]]_q$ asymmetric quantum code, where $d_x = \mathrm{wt}(C_x \setminus C_z^\perp)$ and $d_z = \mathrm{wt}(C_z \setminus C_x^\perp)$.*

If in the above construction $d_x = \mathrm{wt}(C_x)$ and $d_z = \mathrm{wt}(C_z)$, then we say that the code is pure. In this context, we can give a bound for CSS-type pure asymmetric stabilizer codes similar to the quantum Singleton bound.

Lemma 3.2. *A pure asymmetric $[[n, k, d_x/d_z]]_q$ CSS code satisfies*

$$k \leq n - d_x - d_z + 2.$$

Proof. Assume that the asymmetric code is constructed using lemma 3.1, then $\mathrm{wt}(C_x) = d_x$ and $\mathrm{wt}(C_z) = d_z$. By the classical Singleton bound, $|C_x| \leq q^{n-d_x+1}$ and $|C_z| \leq q^{n-d_z+1}$. Then, $|C_x| \cdot |C_z| = q^{n+k} \leq q^{2n-d_x-d_z+2}$. It follows $k \leq n - d_x - d_z + 2$. ∎

The bound in lemma 3.2 can be extended for all $\mathbb{F}_q$-linear asymmetric CSS-type codes.

Lemma 3.3. *Any CSS-type $\mathbb{F}_q$-linear $[[n, k, d_x/d_z]]_q$ CSS-type code satisfies*

$$k \leq n - d_x - d_z + 2. \tag{3.1}$$

Proof. Let us assume that the asymmetric CSS code was constructed using lemma 3.1. Let C_x^c be the complement of $C_z^\perp$ in C_x, i.e. a subcode of C_x, such that the span of C_x^c and $C_z^\perp$ is C_x. Similarly, let C_z^c be the complement of $C_x^\perp$ in C_z. Let $\dim C_x^\perp = k_a$ and $\dim C_z^\perp = k_b$. We have $|C_x^\perp| \cdot |C_z^\perp| = q^{n-k} = q^{k_a+k_b}$. Now, the $\mathbb{F}_q$ linearity of the stabilizer code implies that we can choose C_x^c to be all zeros in k_b columns, because we can perform Gaussian elimination using $C_z^\perp$ to get rid of the non-zero elements in k_b columns. In effect, C_x^c is a code of length $n-k_b$. Now the classical Singleton bound implies that $|C_x^c| \leq q^{(n-k_b)-d_x+1}$. The minimum distance of C_x^c must be at least d_x because it is a subcode of $C_x \setminus C_z^\perp$ and we know that $d_x = \mathrm{wt}(C_x \setminus C_z^\perp)$. Similarly, we can show that $|C_z^c| \leq q^{(n-k_a)-d_z+1}$, where $d_z = \mathrm{wt}(C_z \setminus C_x^\perp)$. It follows that

$$|C_x^c| \cdot |C_z^c| \leq q^{n-k_b-d_x+1} q^{n-k_a-d_z+1} = q^{2n-k_a-k_b-d_x-d_z+2}.$$

But $|C_x^\perp| \cdot |C_z^\perp| = q^{n-k} = q^{k_a+k_b}$, therefore

$$|C_x^c| \cdot |C_z^c| \leq q^{2n-k_a-k_b-d_x-d_z+2} = q^{n+k-d_x-d_z+2}.$$

Now using the fact that $|C_x^c| \cdot |C_z^c| = q^{2k}$, we have $q^k \leq q^{n-d_x-d_z+2}$, i.e. $k \leq n - d_x - d_z + 2$. ∎

This bound seems to imply that if there was an asymmetry in the channel, then using asymmetric quantum codes we could potentially gain in rate. It would be interesting to extend this bound to all asymmetric stabilizer codes linear or otherwise. The (quantum) Singleton bound, in general, is not very tight, especially for large lengths. In this context, the linear programming bounds turn out to be more useful. We shall derive some linear programming bounds for CSS-type asymmetric stabilizer codes.

Theorem 3.4 (Linear programming bounds). *If an* $[[n, k, d_x/d_z]]_2$ *asymmetric CSS stabilizer code with* $k>0$ *exists, then there exists a solution to the optimization problem: maximize* $\sum_{j=1}^{d_z-1} A_j$ *subject to the constraints*

(i) $A_0 = A_0^{\perp} = B_0 = B_0^{\perp} = 1$ *and* $A_j, A_j^{\perp}, B_j, B_j^{\perp} \geq 0$, *for all* $1 \leq j \leq n$,

(ii) $0 < k' < n-k$,

(iii) $$\sum_{j=0}^{n} A_j = 2^{k'},$$

(iv) $$\sum_{j=0}^{n} B_j = 2^{n-k-k'},$$

(v) $$A_j^{\perp} = \frac{1}{2^{k'}} \sum_{r=0}^{n} K_j(r) A_r, \quad \text{for all } j \text{ in the range } 0 \leq j \leq n,$$

(vi) $$B_j^{\perp} = \frac{1}{2^{n-k-k'}} \sum_{r=0}^{n} K_j(r) B_r, \quad \text{for all } j \text{ in the range } 0 \leq j \leq n,$$

(vii) $A_j = B_j^{\perp}$, *for all* j *in* $0 \leq j < d_x$ *and* $A_j \leq B_j^{\perp}$, *for all* $d_x \leq j \leq n$, *and*

(viii) $B_j = A_j^{\perp}$, *for all* j *in* $0 \leq j < d_z$ *and* $B_j \leq A_j^{\perp}$, *for all* $d_z \leq j \leq n$,

where the coefficients A_j, $A_j^{\perp}$, B_j *and* $B_j^{\perp}$ *are integrals and* $K_j(r)$ *denotes the Krawtchouk polynomial*

$$K_j(r) = \sum_{s=0}^{j} (-1)^s \binom{r}{s} \binom{n-r}{j-s}. \tag{3.2}$$

Proof. If an $[[n,k,d_x/d_z]]_2$ asymmetric stabilizer code exists, then by lemma 3.1, there exists classical codes $C_x^{\perp} \subseteq C_z \subseteq \mathbb{F}_2^n$. Let the weight distributions of $C_z^{\perp}$ and $C_x^{\perp}$ be given by A_i and B_i, respectively, where $0 \leq i \leq n$. If we let $|C_z^{\perp}| = 2^{k'}$, then this means $\sum A_i = 2^{k'}$ and since $|C_x^{\perp}| \cdot |C_z^{\perp}| = 2^{n-k}$, it follows that $\sum B_i = 2^{n-k-k'}$. We restrict the range of $0 < k' < n-k$ to ensure that d_x, $d_z > 1$. The weight distributions of C_x and C_z are given by the MacWilliams duality relations (MacWilliams & Sloane 1977). These give us

$$A_j^{\perp} = \frac{1}{2^{k'}} \sum_{r=0}^{n} K_j(r) A_r \tag{3.3}$$

and

$$B_j^{\perp} = \frac{1}{2^{n+k-k'}} \sum_{r=0}^{n} K_j(r) B_r. \tag{3.4}$$

Since $C_z^{\perp} \subseteq C_x$ and $C_x^{\perp} \subseteq C_z$, we must have $A_j \leq B_j^{\perp}$ and $B_j \leq A_j^{\perp}$, for $0 \leq j \leq n$. As the quantum code has X-distance d_x, all vectors of weight less than d_x in C_x must be in $C_z^{\perp}$, giving us $A_j = B_j^{\perp}$, for $1 \leq j \leq d_x - 1$. Similarly, all vectors of weight less than d_z in C_z must be in $C_x^{\perp}$ and we get $B_j = A_j^{\perp}$, for $1 \leq j \leq d_z - 1$. ■

In order to implement the above constraints as a linear programming problem, we must fix the value of k'. Then, the above constraints reduce to $n-k-1$ instances of a linear programming problem. An $[[n, k, d_x/d_z]]$ code will not exist if none of the instances have a solution. It is possible that the code might not exist even if some instance has a solution.

Using the linear programming bounds, we were able to show that there can exist a $[[15, 1, 3/7]]$ code. This code can be constructed using two BCH codes. Further details about this code are given in §5. Note that there does not exist a $[[15, 1, 7]]$ stabilizer code.

One of the referees pointed out that a $[[13, 1, 3/5]]$ exists. We were, however, unable to construct a smaller code, even though the linear programming bounds indicate that a $[[12, 1, 3/5]]$ may exist. Our interest in these small codes is due to the fact that small codes are easier to analyse and study, and might perhaps provide insight into asymmetric quantum error correction. It would be an interesting problem to show its existence or non-existence.

4. Asymmetric quantum codes: constructions

(a) *Construction from families of nested codes*

Our first construction makes use of RM codes (for an introduction see Huffman & Pless 2003, pp. 33–36). Recall that a RM code of order r and length 2^m has the parameters

$$\left[2^m, \sum_{j=0}^{r} \binom{m}{j}, 2^{m-r}\right].$$

Let us denote an rth order RM code as $\mathcal{R}(r, m)$. RM codes have the following interesting properties of relevance for us:

(i) $\mathcal{R}(r, m)^{\perp} = \mathcal{R}(m-1-r, m)$ and
(ii) $\mathcal{R}(r_1, m) \subseteq \mathcal{R}(r_2, m)$ if $r_1 \leq r_2$.

We shall call a family of codes that satisfy a property as (ii) with respect to some code parameter *nested codes*. The BCH codes are also a family of nested codes. As a first example, we construct asymmetric quantum codes based on the nested family of RM codes. The resulting codes are not new, they have been already constructed by Steane (1999*b*); in fact, the use of RM codes for quantum error correction was suggested earlier by Knill *et al.* (1996).

Lemma 4.1 (Asymmetric RM stabilizer codes). *Let $0 \leq r_1 < r_2 < m$. Then, there exists an*

$$\left[\left[2^m, \sum_{j=r_1+1}^{r_2} \binom{m}{j}, 2^{m-r_2}/2^{r_1+1}\right]\right]_2$$

asymmetric RM stabilizer code.

Proof. Let $C_x = \mathcal{R}(r_2, m)$ and $C_z = \mathcal{R}(m-1-r_1, m) = \mathcal{R}(r_1, m)^{\perp}$. Then, the code follows from the application of lemma 3.1. Note that the dimension of the quantum code is

$$k_x + k_z - n = k_x - \dim C_z^{\perp} = \sum_{j=0}^{r_2} \binom{m}{j} - \sum_{j=0}^{r_1} \binom{m}{j} = \sum_{j=r_1+1}^{r_2} \binom{m}{j},$$

where the last equality follows from the fact $r_1 < r_2$. With respect to the distance, we have $d_x = \mathrm{wt}(C_x \setminus C_z^{\perp}) = \mathrm{wt}(\mathcal{R}(r_2, m) \setminus \mathcal{R}(r_1, m)) = 2^{m-r_2}$, because the $\mathrm{wt}(\mathcal{R}(r_1, m)) = 2^{m-r_1} > 2^{m-r_2} = \mathrm{wt}(\mathcal{R}(r_2, m))$. Similarly, $d_z = \mathrm{wt}(C_z \setminus C_x^{\perp}) = \mathrm{wt}(\mathcal{R}(m-1-r_1, m) \setminus \mathcal{R}(m-1-r_2, m)) = 2^{r_1+1}$. ∎

These codes illustrate the trade-offs involved in the design of asymmetric stabilizer codes. In order to have large asymmetry in the distance, we need d_x/d_z to be very small. In this case, we need $2^{r_1+r_2+1-m}$ large. In order to get a large rate, we will require that the difference between $r_2 - r_1$ is also large.

Lemma 4.2 (Rate gain for asymmetric RM stabilizer codes). *Let* $0 \leq \Delta r \leq r \leq \lfloor (m-1)/2 \rfloor$, *then there exists an*

$$\left[\left[2^m, \sum_{j=r+1}^{m-1-r} \binom{m}{j}, 2^{r+1}\right]\right]_2$$

code which can be turned into an

$$\left[\left[2^m, \sum_{j=r+1}^{m-1-r} \binom{m}{j} + \Delta k, 2^{r+1-\Delta r}/2^{r+1}\right]\right]_2$$

asymmetric code, where

$$\Delta k = \sum_{m-r}^{m-1-r+\Delta r} \binom{m}{j}.$$

Proof. Under the conditions on r, we can choose $C_x = C_z = \mathcal{R}(m-1-r, m)$ and apply lemma 3.1, then the stabilizer code follows. If we choose $C_x = \mathcal{R}(m-1-r+\Delta r, m)$, then we can convert the stabilizer code into the asymmetric stabilizer code with the given parameters. ∎

The preceding result indicates that asymmetry can be exploited to get higher rates compared with a symmetric stabilizer code.

Example 4.3. In table 1, we show the gains in rate as one allows for more and more asymmetry.

We denote a q-ary narrow-sense BCH code of length n and design distance δ by $\mathcal{BCH}_q(\delta, n)$. We drop n in the notation if the code is primitive, i.e. $n = q^m - 1$, where $m = \mathrm{ord}_n(q)$ is the multiplicative order of q modulo n. For binary codes, we suppress q also for convenience. If $\delta \leq 2^{\lceil m/2 \rceil} - 1$, we can compute the dimension of the BCH code exactly as $2^m - 1 - mt$ (see MacWilliams & Sloane 1977, p. 263).[1] In the following, we assume that the design distances of binary BCH codes are odd.

[1] Actually, this relation holds for $\delta \leq 2^{\lceil m/2 \rceil} + 3$. But this suffices for our purposes.

Table 1. Asymmetric quantum codes constructed from the nested codes construction applied to RM stabilizer codes.

r	symmetric code $[[n, k, d]]_2$	asymmetric code $[[n, k, d_x/d_z]]_2$		
Δr	0	1	2	3
4	$[[1024, 252, 32]]_2$	$[[1024, 462, 16/32]]_2$	$[[1024, 582, 8/32]]_2$	$[[1024, 627, 4/32]]_2$
3	$[[1024, 672, 16]]_2$	$[[1024, 792, 8/16]]_2$	$[[1024, 837, 4/16]]_2$	$[[1024, 847, 2/16]]_2$
2	$[[1024, 912, 8]]_2$	$[[1024, 957, 4/8]]_2$	$[[1024, 967, 2/8]]_2$	
1	$[[1024, 1002, 4]]_2$	$[[1024, 1012, 2/4]]_2$		

Lemma 4.4 (Asymmetric BCH stabilizer codes). *Let* $m \geq 2$ *and* $2 \leq \delta_1 < \delta_2 < \delta_{\max}$, *where* $\delta_{\max} = 2^{\lceil m/2 \rceil} - 1$ *and* $\delta_i \equiv 1 \bmod 2$. *Then, there exists an*

$$[[2^m - 1, m(\delta_2 - \delta_1)/2, d_x/d_z]]_2$$

asymmetric BCH stabilizer code, where $d_x \geq \delta_1$ *and* $d_z \geq \delta_{\max} + 1$.

Proof. In this case, we choose $C_x = \mathcal{BCH}(\delta_1)$ and $C_z = \mathcal{BCH}(\delta_2)^\perp$. Under the restrictions on δ_1 and δ_2, we can compute the dimension of the BCH codes explicitly. The dimension of $(\delta)^\perp$ is given by $m(\delta-1)/2$, assuming odd δ. By lemma 3.1, we can then compute the dimension of the quantum code as $m(\delta_2-1)/2 - m(\delta_1-1)/2 = m(\delta_2-\delta_1)/2$.

We have $d_x \geq \mathrm{wt}(\mathcal{BCH}(\delta_1) \setminus \mathcal{BCH}(\delta_2)) \geq \delta_1$ and $d_z \geq \mathrm{wt}(\mathcal{BCH}(\delta_2)^\perp \setminus \mathcal{BCH}(\delta_1)^\perp) \geq \delta_{\max}$. The last inequality follows from the fact that the distance of $(\delta_2)^\perp$ code is at least $\delta_{\max}+1$, as shown by lemma 10 of Aly *et al.* (2007). ■

Example 4.5. Suppose we let $m=10$ and vary δ_1 and δ_2. Some of the codes that can be constructed are given in table 2.

Alternatively, we could consider dual containing BCH codes to construct stabilizer codes. In this case, it is easier to see that the gain in rate is proportional to the loss in the distance for the bit-flip channel.

Lemma 4.6. *Let* $n = 2^m - 1$, $\delta_{\max} = 2^{\lceil m/2 \rceil} - 1$ *and* $\delta = 2t+1 \leq \delta_{\max}$. *Then, there exists an* $[[n, n - m(\delta-1), \geq \delta]]_2$ *stabilizer code that can be converted to an* $[[n, n - m(\delta-1) + m\Delta/2, d_x/d_z]]_2$ *asymmetric stabilizer code, where* $0 \leq \Delta = 2l \leq \delta - 2$ *and* $d_x \geq \delta - \Delta$ *and* $d_z \geq \delta$.

Proof. Under the restrictions on δ, we have $\mathcal{BCH}(\delta)^\perp \subseteq \mathcal{BCH}(\delta)$ (see lemma 1 of Steane 1999*a*). Then, the stabilizer code is obtained by choosing $C_x = C_z = \mathcal{BCH}(\delta)$ in lemma 3.1. If C_x is chosen to be $\mathcal{BCH}(\delta - \Delta)$, then we get the asymmetric stabilizer code. ■

These results seem to indicate, once again, that asymmetry, in general, can lead to a rate gain. In the case of the BCH stabilizer codes constructed above, it appears that the rate gain is linearly proportional to the reduction in the distance of the code used for correcting bit-flip errors.

In general, any family of nested codes can lead to a class of asymmetric stabilizer codes. The main obstacle to such a method is usually the knowledge of the dual distances, a knowledge of which is required to determine the error-correcting capability of the quantum code. However, algebraic codes have this

Table 2. Asymmetric quantum codes constructed from the nested codes construction applied to BCH stabilizer codes.

δ_1	δ_2	code $[[n, k, d_x/d_z]]_2$	rate k/n	asymmetry d_z/d_x
29	31	$[[1023, 10, 29/32]]_2$	0.00978	≈ 1
17	31	$[[1023, 70, 17/32]]_2$	0.06843	
15	31	$[[1023, 80, 15/32]]_2$	0.07820	≈ 2
13	31	$[[1023, 90, 13/32]]_2$	0.08798	
11	31	$[[1023, 100, 11/32]]_2$	0.09775	≈ 3
9	31	$[[1023, 110, 9/32]]_2$	0.10753	
7	31	$[[1023, 120, 7/32]]_2$	0.11730	≈ 4
5	31	$[[1023, 130, 5/32]]_2$	0.12708	≈ 6
3	31	$[[1023, 140, 3/32]]_2$	0.13685	≈ 10
27	29	$[[1023, 10, 27/32]]_2$	0.00978	≈ 1
15	29	$[[1023, 60, 15/32]]_2$	0.05865	≈ 2
9	29	$[[1023, 100, 9/32]]_2$	0.09775	≈ 3
7	29	$[[1023, 110, 7/32]]_2$	0.10753	≈ 4
5	29	$[[1023, 120, 5/32]]_2$	0.11730	≈ 6
3	29	$[[1023, 130, 3/32]]_2$	0.12708	≈ 10

nice structure of being nested, and within certain ranges it is also possible to lower bound the distances including the dual distances. So, while we cannot claim a truly asymmetric code because of our lack of knowledge of these distances, we can certainly form a class of codes quite suitable for the asymmetric channels.

(*b*) *Construction from dual pairs of LDPC and BCH codes*

Nested codes are not the only method to construct asymmetric codes. Ioffe & Mézard (2007) used a combination of BCH and LDPC codes to construct asymmetric codes. The intuition being that the stronger LDPC code should be used for correcting the phase errors and the BCH code can be used for the infrequent bit flips. This essentially reduces to finding a good LDPC code such that the dual of the LDPC code is contained in the BCH code. They solve this problem by randomly choosing code words in the BCH code which are of low weight (so that they can be used for the parity-check matrix of the LDPC code). However, their method is a little ad hoc and it is not clear how good is the resulting LDPC code. For instance, the degree profiles of the resulting code are irregular and there is little control over the final degree profiles of the code. It was claimed by Ioffe & Mézard (2007) that their codes can be analysed using the traditional methods of analysis for the LDPC codes. But there are many questions that must be answered before it can be fully justified. For instance, it is not apparent what ensemble or degree profiles one will use for performing this analysis.

We consider using LDPC codes to construct asymmetric stabilizer codes. Part of the reason being that these codes are among the best classical codes with efficient decoding algorithms. It should be noted that several classes of quantum LDPC codes have been proposed (MacKay *et al.* 2004; Camara *et al.* 2007; Hagiwara & Imai 2007). So far, however, the structure of quantum LDPC codes

in general, including their decoding algorithms, is not well understood. For some time it was even doubtful whether quantum LDPC codes exist at all, since a simple observation shows, for instance, that quantum LDPC codes constructed as stabilizer codes must necessarily have four cycles in their Tanner graph, but see the work of Hagiwara & Imai (2007) for an interesting work around. To circumvent this problem, we are interested in families of quantum LDPC codes which have the property to tolerate large numbers of four cycles without affecting the performance of the decoding algorithm too adversely.

More precisely, we propose two families of quantum codes based on LDPC codes. In the first case, we use LDPC codes for both the X- and Z-channels. We will need the following facts about generalized RM codes and finite geometry LDPC codes.

(i) *Some facts about finite geometry LDPC codes*

In this section, we will briefly review finite geometry LDPC codes (Kou *et al.* 2001; Tang *et al.* 2005). Let us denote by $\mathrm{EG}(m, p^s)$ the Euclidean finite geometry over $\mathbb{F}_{p^s}$ consisting of p^{ms} points. For our purposes, it only suffices to use the fact that this geometry is equivalent to the vector space $\mathbb{F}_{p^s}^m$. A μ-dimensional subspace of $\mathbb{F}_{p^s}^m$ or its coset is called a *μ-flat*. We can describe a μ-flat by the following equation:

$$a_0 + \lambda_1 a_1 + \lambda_2 a_2 + \cdots + \lambda_\mu a_\mu; \quad \lambda_i \in \mathbb{F}_{p^s}, \tag{4.1}$$

where the $a_i \in \mathbb{F}_{p^s}^m$ are linearly independent. The 0-flats and 1-flats are the familiar points and lines. Assume that $0 \le \mu_1 < \mu_2 \le m$. Then, we denote by $N_{\mathrm{EG}}(\mu_2, \mu_1, s, p)$ the number of μ_1-flats in a μ_2-flat and by $A_{\mathrm{EG}}(m, \mu_2, \mu_1, s, p)$, the number of μ_2-flats that contain a given μ_1-flat. These are given by (see Tang *et al.* 2005)

$$N_{\mathrm{EG}}(\mu_2, \mu_1, s, p) = q^{(\mu_2 - \mu_1)} \prod_{i=1}^{\mu_1} \frac{q^{\mu_2 - i + 1} - 1}{q^{\mu_1 - i + 1} - 1} \tag{4.2}$$

and

$$A_{\mathrm{EG}}(m, \mu_2, \mu_1, s, p) = \prod_{i=\mu_1+1}^{\mu_2} \frac{q^{m-i+1} - 1}{q^{\mu_2 - i + 1} - 1}, \tag{4.3}$$

where $q = p^s$. Index all the μ_1-flats from $i = 1$ to $n = N_{\mathrm{EG}}(m, \mu_1, s, p)$ as F_i. Let $\mathfrak{F}$ be a μ_2-flat in $\mathrm{EG}(m, p^s)$. Then, we can associate an incidence vector to $\mathfrak{F}$ with respect to the μ_1 flats as follows:

$$I_{\mathfrak{F}} = \left\{ i_j \,\middle|\, \begin{array}{ll} i_j = 1, & \text{if } F_j \text{ is contained in } \mathfrak{F}, \\ i_j = 0, & \text{otherwise.} \end{array} \right\}$$

Index the μ_2-flats from $j = 1$ to $J = N_{\mathrm{EG}}(m, \mu_2, s, p)$. Construct the $J \times n$ matrix $H_{\mathrm{EG}}^{(1)}(m, \mu_2, \mu_1, s, p)$, whose rows are the incidence vectors of all the μ_2-flats with respect to the μ_1-flats. This matrix is also referred to as the incidence matrix. Then, the type-I Euclidean geometry code from μ_2- and μ_1-flats is defined to be the null space (i.e. the Euclidean dual) of the $\mathbb{F}_p$-linear span

of $H^{(1)}_{\mathrm{EG}}(m,\mu_2,\mu_1,s,p)$. This is denoted as $C^{(1)}_{\mathrm{EG}}(m,\mu_2,\mu_1,s,p)$. Let $H^{(2)}_{\mathrm{EG}}(m,\mu_2,\mu_1,s,p)=H^{(1)}_{\mathrm{EG}}(m,\mu_2,\mu_1,s,p)^t$. Then, the type-II Euclidean geometry code $C^{(2)}_{\mathrm{EG}}(m,\mu_2,\mu_1,s,p)$ is defined to be the null space of $H^{(2)}_{\mathrm{EG}}(m,\mu_2,\mu_1,s,p)$. Let us now consider the μ_2- and μ_1-flats that do not contain the origin of $\mathrm{EG}(m,p^s)$. Now form the incidence matrix of the μ_2-flats with respect to the μ_1-flats not containing the origin. The null space of this incidence matrix gives us a quasi-cyclic code in general, which we denote by $C^{(1)}_{\mathrm{EG},c}(m,\mu_2,\mu_1,s,p)$.

(ii) *Some facts about the generalized Reed–Muller codes*

Before we proceed further, we recall some salient facts about the generalized RM codes (Kasami *et al.* 1968*a*). Let α be a primitive element in $\mathbb{F}_{q^m}$. The cyclic generalized RM code of length q^m-1 and order ν is defined as the cyclic code with the generator polynomial, whose roots α^j satisfy $0<j\leq m(q-1)-\nu-1$. The generalized RM code is the singly extended code of length q^m. It is denoted as $\mathrm{GRM}_q(\nu,m)$. The dual of a GRM code is also a GRM code (Kasami *et al.* 1968*a*; Assmus & Key 1998; Blahut 2003). It is known that

$$\mathrm{GRM}_q(\nu,m)^{\perp}=\mathrm{GRM}_q(\nu^{\perp},m),\quad \text{with } \nu^{\perp}=m(q-1)-1-\nu. \tag{4.4}$$

Let C be a linear code over $\mathbb{F}_{q^s}^n$. Then, we define $C\,|_{\mathbb{F}_q}$, the *subfield subcode* of C over $\mathbb{F}_q^n$ as the code words of C, which are entirely in $\mathbb{F}_q^n$ (see Huffman & Pless 2003, pp. 116–120). Formally, this can be expressed as

$$C\,|_{\mathbb{F}_q}=\{c\in C\,|\,c\in\mathbb{F}_q^n\}. \tag{4.5}$$

Let $C\subseteq\mathbb{F}_{q^l}^n$. Then, the *trace code* of C over $\mathbb{F}_q$ is defined as

$$\mathrm{tr}_{q^l/q}(C)=\{\mathrm{tr}_{q^l/q}(c)\,|\,c\in C\}, \tag{4.6}$$

where $\mathrm{tr}_{q^l/q}(x)=\sum_{i=0}^{l-1}x^{q^i}$. There are interesting relationships between the trace code and the subfield subcode. One of which is the following result, which we will need later.

Lemma 4.7. *Let $C\subseteq\mathbb{F}_{q^l}^n$. Then $C\,|_{\mathbb{F}_q}$, the subfield subcode of C is contained in $\mathrm{tr}_{q^l/q}(C)$, the trace code of C. In other words,*

$$C\,|_{\mathbb{F}_q}\subseteq\mathrm{tr}_{q^l/q}(C).$$

Proof. Let $c\in C\,|_{\mathbb{F}_q}\subseteq\mathbb{F}_q^n$ and $\alpha\in\mathbb{F}_{q^l}$. Then, $\mathrm{tr}_{q^l/q}(\alpha c)=c\,\mathrm{tr}_{q^l/q}(\alpha)$ as $c\in\mathbb{F}_q^n$. Since trace is a surjective form, there exists some $\alpha\in\mathbb{F}_{q^l}$, such that $\mathrm{tr}_{q^l/q}(\alpha)=1$. This implies that $c\in\mathrm{tr}_{q^l/q}(C)$. Since c is an arbitrary element in $C\,|_{\mathbb{F}_q}$, it follows that $C\,|_{\mathbb{F}_q}\subseteq\mathrm{tr}_{q^l/q}(C)$. ∎

The other relation due to Delsarte is the following.

Lemma 4.8 (Delsarte 1975). *Let $C\subseteq\mathbb{F}_{q^l}^n$. Then,*

$$C\,|_{\mathbb{F}_q}^{\perp}=\mathrm{tr}_{q^l/q}(C^{\perp}).$$

Let $q=p^s$, then the Euclidean geometry code of order r over $\mathrm{EG}(m,p^s)$ is defined as the dual of the subfield subcode of $\mathrm{GRM}_q((q-1)(m-r-1),m)$ (Blahut 2003, p. 448). The type-I LDPC code of $C^{(1)}_{\mathrm{EG}}(m,\mu,0,s,p)$ code is an Euclidean geometry code of order $\mu-1$ over $\mathrm{EG}(m,p^s)$ (see Tang *et al.* 2005). Hence, its dual is the subfield subcode of $\mathrm{GRM}_q((q-1)(m-\mu),m)$ code. In other words,

$$C^{(1)}_{\mathrm{EG}}(m,\mu,0,s,p)^{\perp} = \mathrm{GRM}_q((q-1)(m-\mu),m)\,|_{\mathbb{F}_p}. \tag{4.7}$$

Furthermore, Delsarte's result (lemma 4.8) states that

$$\begin{aligned}\mathrm{GRM}_q((q-1)(m-\mu),m)\,|^{\perp}_{\mathbb{F}_p} &= \mathrm{tr}_{q/p}(\mathrm{GRM}_q((q-1)(m-\mu),m)^{\perp})\\ &= \mathrm{tr}_{q/p}(\mathrm{GRM}_q(\mu(q-1)-1,m)).\end{aligned}$$

Hence, $C^{(1)}_{\mathrm{EG}}(m,\mu,0,s,p)$ code can also be related to $\mathrm{GRM}_q(\mu(q-1)-1,m)$ as

$$C^{(1)}_{\mathrm{EG}}(m,\mu,0,s,p) = \mathrm{tr}_{q/p}(\mathrm{GRM}_q(\mu(q-1)-1),m). \tag{4.8}$$

In theorem 4.9, we use all these facts together to construct a family of asymmetric stabilizer LDPC codes.

Theorem 4.9 (Asymmetric EG LDPC codes). *Let p be a prime, with $q=p^s$ and $s\geq 1$, $m\geq 2$. Let $1<\mu_z<m$ and $m-\mu_z+1\leq\mu_x<m$. Then, there exists an*

$$[[p^{ms}, k_x+k_z-p^{ms}, d_x/d_z]]_p$$

asymmetric EG LDPC code, where

$$k_x = \dim C^{(1)}_{\mathrm{EG}}(m,\mu_x,0,s,p) \quad \textit{and} \quad k_z = \dim C^{(1)}_{\mathrm{EG}}(m,\mu_z,0,s,p),$$

$$d_x \geq A_{\mathrm{EG}}(m,\mu_x,\mu_x-1,s,p)+1 \textit{ and } d_z \geq A_{\mathrm{EG}}(m,\mu_z,\mu_z-1,s,p)+1.$$

Proof. Let us choose $C_z = C^{(1)}_{\mathrm{EG}}(m,\mu_z,0,s,p)$. Then, from equation (4.8) we have

$$C_z = \mathrm{tr}_{q/p}(\mathrm{GRM}_q(\mu_z(q-1)-1),m).$$

By lemma 4.7, we know that

$$\begin{aligned}\mathrm{tr}_{q/p}(\mathrm{GRM}_q(\mu_z(q-1)-1),m) &\supseteq \mathrm{GRM}_q(\mu_z(q-1)-1,m)\,|_{\mathbb{F}_p}\\ C_z &\supseteq \mathrm{GRM}_q((q-1)(m-(m-\mu_z+1)),m)\,|_{\mathbb{F}_p},\end{aligned}$$

where the last inclusion follows from the nesting property of the generalized RM codes. For any order μ_x such that $m-\mu_z+1\leq\mu_x<m$, let $C_x = C^{(1)}_{\mathrm{EG}}(m,\mu_x,0,s,p)$. Then C_x is an LDPC code whose dual $C_x^{\perp} = \mathrm{GRM}_q((q-1)(m-\mu_x),m)\,|_{\mathbb{F}_p}$ is contained in C_z. Thus, we can use lemma 3.1 to form an asymmetric code with the parameters

$$[[p^{ms}, k_x+k_z-p^{ms}, d_x/d_z]]_p.$$

The distance of C_z and C_x are lower bounded as $d_x\geq A_{\mathrm{EG}}(m,\mu_x,\mu_x-1,s,p)+1$ and $d_z\geq A_{\mathrm{EG}}(m,\mu_z,\mu_z-1,s,p)+1$ (see Tang *et al.* 2005). ∎

In the construction just proposed, we should choose C_x to be a larger code compared with C_z, so that C_z is a stronger code. We have the construction over a non-binary alphabet, although in the context of asymmetric quantum codes, one might be more interested in the case $p=2$.

We briefly turn our attention back to the (symmetric) depolarizing channel. It turns out that the presented LDPC codes, which are designed for asymmetric channels, will, in general, not perform equally well on the depolarizing channel. In fact, constructing good quantum LDPC codes for the depolarizing channel

remains a difficult problem and a satisfactory solution is yet to be advanced. However, we point out in corollary 4.10 to theorem 4.9 that, under certain conditions on the rate of the desired code, it is possible to construct quantum LDPC codes also for the symmetric depolarizing channel.

Corollary 4.10 (EG LDPC codes for depolarizing channel). *Let p be a prime, with $q=p^s$ and $s\geq 1$, $m\geq 2$. Let $\lceil (m+1)/2\rceil \leq \mu < m$. Then, there exists an $[[p^{ms}, 2k-p^{ms}, d]]_p$ symmetric EG LDPC code, where $k=\dim C^{(1)}_{\mathrm{EG}}(m,\mu,0,s,p)$. For the distance, $d\geq A_{\mathrm{EG}}(m,\mu,\mu-1,s,p)+1$ holds.*

Our next construction is an alternative to the method proposed by Ioffe & Mézard (2007). Now we shall make use of the cyclic finite geometry codes. Our goal will be to find a BCH code whose dual is contained in a cyclic Euclidean geometry LDPC code. First, we need the notion of q-ary weight. Let $0\leq h<q^m$, then we define

$$W_q(h)=\sum h_i, \quad \text{where } h=h_0+h_1q+\cdots+h_{m-1}q^{(m-1)}. \tag{4.9}$$

If $h\geq q^m$, then we compute $W_q(h)$ as the weight of h mod q^m-1.

Theorem 4.11 (Asymmetric BCH–LDPC stabilizer codes). *Let C be the code $C^{(1)}_{\mathrm{EG},c}(m,\mu,0,s,p)$ and $\delta\leq\delta_0=p^{\mu s}-1$. Then, there exists an*

$$[[p^{ms}-1, k_x+k_y-n, d_x/d_z]]_p$$

asymmetric stabilizer code, where $d_z\geq A_{\mathrm{EG}}(m,\mu,\mu-1,s,p)$ and $d_x\geq\delta$.

Proof. For proving this theorem, we need the cyclic structure of $C^{(1)}_{\mathrm{EG},c}(m,\mu,0,s,p)$. Let α be a primitive element in $\mathbb{F}_{p^{ms}}$. Then, the roots of generator polynomial of $C^{(1)}_{\mathrm{EG},c}(m,\mu,0,s,p)$ are given by Kasami & Lin (1971, theorem 6), see also Kasami *et al.* (1968*b*) and Lin & Costello (2004),

$$Z=\left\{\alpha^h \,|\, 0<\max_{0\leq l<s} W_{p^s}(hp^l)\leq (p^s-1)(m-\mu)\right\},$$

where $W_q(h)$ is as defined in equation (4.9). The code $C^{(1)}_{\mathrm{EG},c}(m,\mu,0,s,p)$ is actually an $(\mu-1,p^s)$ Euclidean geometry code. The roots of the generator polynomial of the dual code are given by

$$Z^{\perp}=\left\{\alpha^h \,|\, \min_{0\leq l<s} W_{p^s}(hp^l)<\mu(p^s-1)\right\}.$$

In fact, the dual code is the even-like subcode of a primitive polynomial code of length $p^{ms}-1$ over $\mathbb{F}_p$ and order $m-\mu$. By Kasami *et al.* (1968*b*, theorem 6), the generator polynomial of the polynomial code has the roots

$$Z_p=\left\{\alpha^h \,|\, 0<\min_{0\leq l<s} W_{p^s}(hp^l)<\mu(p^s-1)\right\}.$$

Thus, $Z^{\perp}=Z_p\cup\{0\}$. Now by Kasami & Lin (1971, theorem 6), Z_p and therefore $Z^{\perp}$ contain the sequence of consecutive roots, $\alpha, \alpha^2, \ldots, \alpha^{\delta_0-1}$, where

$$\delta_0=(R+1)p^{Qs}-1 \quad \text{and} \quad m(p^s-1)-(m-\mu)(p^s-1)=Q(p^s-1)+R.$$

Simplifying, we see that $R=0$ and $Q=\mu$ giving $\delta_0=p^{\mu s}-1$. It follows that

$$C^{(1)}_{\mathrm{EG},c}(m,\mu,0,s,p)^{\perp} = \mathrm{GRM}_q((q-1)(m-\mu),m)\,|_{\mathbb{F}_p} \subseteq \mathcal{BCH}(\delta_0).$$

Thus, we have solved the problem of construction of the asymmetric stabilizer codes in dual fashion to that of Ioffe & Mézard (2007). Instead of finding an LDPC code whose parity-check matrix is contained in a given BCH code, we have found a BCH code whose parity-check matrix is contained in a given finite geometry LDPC code. Choosing any BCH code whose design distance is less than $p^{\mu s}-1$ gives a BCH code, which contains the dual of the finite geometry LDPC code $C^{(1)}_{\mathrm{EG},c}(m,\mu,0,s,p)$. Thus, we can apply lemma 3.1 to obtain the code stated in the statement of the theorem. ■

In what follows, we give a small example to illustrate this construction.

Example 4.12. Let $m=s=p=2$ and $\mu=1$. Then, $C^{(1)}_{\mathrm{EG},c}(2,1,0,2,2)$ is a cyclic code whose generator polynomial has roots given by

$$\begin{aligned} Z &= \{\alpha^h \,|\, 0 < \max_{0\le l<2} W_{2^2}(2^l h) \le (m-\mu)(p^s-1) = (2-1)(2^2-1)\} \\ &= \{\alpha^1,\alpha^2,\alpha^3,\alpha^4,\alpha^6,\alpha^8,\alpha^9,\alpha^{12}\}. \end{aligned}$$

As there are four consecutive roots and $|Z|=8$, it defines a $[15, 7, \ge 5]$ code. The roots of the generator polynomial of the dual code are given by

$$\begin{aligned} Z^{\perp} &= \{\alpha^h \,|\, 0 < \min_{0\le l<2} W_{2^2}(2^l h) \le \mu(p^s-1) = (2^2-1)\} \\ &= \{\alpha^0,\alpha^1,\alpha^2,\alpha^4,\alpha^5,\alpha^8,\alpha^{10}\}. \end{aligned}$$

We see that $Z^{\perp}$ has two consecutive roots excluding 1; therefore, the dual code is contained in a narrow-sense BCH code with design distance 3. Note that $p^{\mu s}-1=3$. Thus, we can choose $C_x=\mathcal{BCH}(3)$ and $C_z=C^{(1)}_{\mathrm{EG},c}(2,1,0,2,2)$ and apply lemma 3.1 to construct a $[[15, 3, 3/5]]_2$ asymmetric code.

We can also state the above construction in a dual fashion, i.e. given a primitive BCH code of design distance δ, find an EG LDPC code whose dual is contained in it. It must be pointed out that in the case of asymmetric codes derived from LDPC codes, the asymmetry factor d_x/d_z is not as indicative of the code performance as in the case of bounded distance decoders. For $m=p=2$, we can derive explicit relations for the parameters of the codes.

Corollary 4.13. *Let* $C=C^{(1)}_{\mathrm{EG},c}(2,1,0,s,2)$ *and* $\delta=2t+1\le 2^s-1$. *Then, there exists an*

$$[[2^{2s}-1, 2^{2s}-3^s-s(\delta-1), \delta/2^s+1]]_2$$

asymmetric stabilizer code.

Proof. The parameters of C are $[2^{2s}-1,2^{2s}-3^s,2^s+1]_2$ (see Lin & Costello 2004). Since $C^{\perp}$ is contained in a BCH code of length $2^{2s}-1$ whose design distance $\delta\le 2^s-1$, we can compute the dimension of the BCH code as $2^{2s}-1-s(\delta-1)$ (see MacWilliams & Sloane 1977, corollary 8). By lemma 3.1, the quantum code has the dimension $2^{2s}-3^s-s(\delta-1)$. ■

Table 3. Asymmetric quantum codes constructed from the construction based on pairs of LDPC and BCH codes.

s	δ	code $[[n, k, d_x/d_z]]_2$	asymmetry d_z/d_x	rate
4	15	$[[255, 119, 15/17]]_2$	≈ 1	0.467
4	13	$[[255, 127, 13/17]]_2$	≈ 1.25	0.498
4	11	$[[255, 135, 11/17]]_2$	≈ 1.5	0.529
4	9	$[[255, 143, 9/17]]_2$	≈ 2	0.561
4	7	$[[255, 151, 7/17]]_2$	≈ 2.5	0.592
4	5	$[[255, 159, 5/17]]_2$	≈ 3	0.624
4	3	$[[255, 167, 3/17]]_2$	≈ 6	0.655

Example 4.14. For $m=p=2$ and $s=4$, we can obtain a $[255, 175, 17]$ LDPC code. We can choose any BCH code with design distance $\delta \leq 2^4-1=15$ to construct an asymmetric code. Possible codes are given in table 3.

The previous method of using a BCH code and a LDPC code can also be used in conjunction with any LDPC code that is cyclic. In particular, LDPC codes derived from projective finite geometry are amenable to this approach. First, let us recall some facts about projective geometries. An m-dimensional projective geometry is denoted as $\mathrm{PG}(m, p^s)$. The points in $\mathrm{PG}(m, p^s)$ can be put in correspondence with the non-zero elements of $\mathbb{F}_{p^s}^{m+1}$. A μ-flat in $\mathrm{PG}(m, p^s)$ is described by

$$\lambda_0 a_0 + \lambda_1 a_1 + \lambda_2 a_2 + \cdots + \lambda_\mu a_\mu; \quad \lambda_i \in \mathbb{F}_{p^s} \text{ and not all } \lambda_i = 0, \tag{4.10}$$

and $a_i \in \mathbb{F}_{p^s}^{m+1}$ are linearly independent. Any non-zero $(m+1)$-tuple is a point in the projective geometry, $\mathrm{PG}(m, p^s)$. Two tuples are equivalent if one is non-zero ($\mathbb{F}_{p^s}$) scalar multiple of other. Consequently, there are $n=(p^{(m+1)s}-1)/(p^s-1)$ points in $\mathrm{PG}(m, p^s)$. Let α be a primitive element in $\mathbb{F}_{p^{(m+1)s}}$ and $\beta=\alpha^n$. Then, β is a primitive element of $\mathbb{F}_{p^s}$. A point in $\mathrm{PG}(m, p^s)$ can also be represented as α^i for some power of α. Since $\mathbb{F}_{p^s}$ scalar multiples of α^i denote the same point, we denote this equivalence by using (α^i), where $(\alpha^i)=\{\alpha^i, \beta\alpha^i, \ldots, \beta^{p^s-2}\alpha^i\}$ and $0 \leq i \leq n-1$. Let $0 \leq \mu_1 < \mu_2 \leq m$, then we define $N_{\mathrm{PG}}(\mu_2, \mu_1, s, p)$ to be the number of μ_1-flats that are contained in a given μ_2-flat and $A_{\mathrm{PG}}(\mu_2, \mu_1, s, p)$ to be the number of μ_2-flats that contain a given μ_1-flat. These are given as

$$N_{\mathrm{PG}}(\mu_2, \mu_1, s, p) = \prod_{i=0}^{\mu_1} \frac{p^{s(\mu_2-i+1)}-1}{p^{s(\mu_1-i+1)}-1} \tag{4.11}$$

and

$$A_{\mathrm{PG}}(m, \mu_2, \mu_1, s, p) = \prod_{i=\mu_1+1}^{\mu_2} \frac{p^{s(m-i+1)}-1}{p^{s(\mu_2-i+1)}-1}. \tag{4.12}$$

Index the μ_1-flats in $\mathrm{PG}(m, p^s)$ as F_j from $j=1$ to $N=N_{\mathrm{PG}}(\mu_2, \mu_1, s, p)$. The incidence vector associated with a μ_2-flat $\mathfrak{F}$ with respect to the μ_1-flats is given (as in the Euclidean case) by

$$I_{\mathfrak{F}} = \left\{ i_j \,\middle|\, \begin{array}{ll} i_j = 1, & \text{if } F_j \text{ is contained in } \mathfrak{F}, \\ i_j = 0, & \text{otherwise.} \end{array} \right\}$$

Index all the μ_2-flats from $i=1$ to $J=N_{\mathrm{PG}}(m, \mu_2, s, p)$ and form the $J\times N$ incidence matrix $H^{(1)}_{\mathrm{PG}}$ from the incidence vectors of all the μ_2-flats. The null space of the matrix defines the type-I projective geometry LDPC code. With this preparation, we are ready to construct asymmetric BCH–LDPC stabilizer codes from projective geometries.

Theorem 4.15. *Let* $C=C^{(1)}_{\mathrm{PG}}(m, \mu, 0, s, p)$, $n=(p^{(m+1)s}-1)/(p^s-1)$ *and* $\delta\leq\delta_0=(p^{(\mu+1)s}-1)/(p^s-1)$. *Then, there exists an*

$$[[n, k_x + k_y - n, d_x/d_z]]_p$$

asymmetric stabilizer code, where $k_x=\dim \mathcal{BCH}_p(\delta, n)$, $k_z=\dim C^{(1)}_{\mathrm{PG}}(m, \mu, 0, s, p)$, $d_z\geq A_{\mathrm{EG}}(m, \mu, \mu-1, s, p)$ *and* $d_x\geq\delta$.

Proof. Let α be a primitive element of $\mathbb{F}_{p^{(m+1)s}}$. The code $C^{(1)}_{\mathrm{PG}}(m, \mu, 0, s, p)$ is a cyclic code with α^h as roots of its generator polynomial if and only if p^s-1 divides h and $\max_{0\leq l<s} W_{p^s}(p^l h)=j(p^s-1)$ for some $0\leq j\leq m-\mu$ (see Tang *et al.* 2005, eqn (27)). In other words, the roots are given by

$$Z=\{\alpha^h \,|\, p^s-1 \,|\, h \text{ and } \max_{0\leq l<s} W_{p^s}(p^l h) = j(p^s-1) \text{ for some } \; 0\leq j\leq (m-\mu)\}.$$

The roots of the dual code are given by

$$Z^{\perp}=\{\alpha^h \,|\, p^s-1 \,|\, h \text{ and } \min_{0\leq l<s} W_{p^s}(p^l h) = j(p^s-1) \text{ for some } 0<j\leq (\mu+1)\}.$$

Now by Kasami *et al.* (1968*b*, theorem 11), $Z^{\perp}$ contains a sequence of δ_0 consecutive roots given by $\{\beta, \beta', \ldots, \beta^{\delta_0-1}\}$ in $Z^{\perp}$, where $\beta=\alpha^{p^s-1}$ and δ_0 is given as $\delta_0=((R+1)q^{Qs}+1)/(p^s-1)$, where

$$(m+1)(p^s-1)-(m-\mu)(p^s-1) = Q(p^s-1)+R.$$

It follows that $R=0$ and $Q=\mu+1$. Therefore, there exists a BCH code, D of design distance $\delta\leq\delta_0=(p^{(\mu+1)s}-1)/(p^s-1)$ that contains the dual of the LDPC code. Additionally, observe that the order of β is $\mathrm{ord}(\alpha)/\gcd(p^s-1, \mathrm{ord}(\alpha))=(p^{(m+1)s}-1)/\gcd(p^s-1, p^{(m+1)s}-1)=n$. Therefore, D is a (non-primitive) narrow-sense BCH code. In conjunction with lemma 3.1, $\mathcal{BCH}_p(\delta, n)$ and $C^{(1)}_{\mathrm{PG}}(m, \mu, 0, s, p)$ give the code with the parameters stated in the theorem. ∎

It will be useful to have exact expressions for the dimensions of the code constructed using theorem 4.15. In general, these expressions are fairly complicated, but for the special case of $m=2$ and $\mu=1$, we can compute dim $C^{(1)}_{\mathrm{PG}}(m, \mu, 0, s, p)$ explicitly. Additionally, the following fact (Aly *et al.* 2007, theorem 10) on the dimension of non-primitive BCH codes is required.

Lemma 4.16 (Aly *et al.* 2007). *Let* q *be a power of prime and* $\gcd(n, q)=1$ *with* $\mathrm{ord}_n(q)=m$. *Then a narrow-sense BCH code of length* $q^{\lfloor m/2\rfloor}\leq n\leq q^m-1$ *over* $\mathbb{F}_q$ *with designed distance* δ *in the range* $2\leq\delta\leq\min\{\lfloor nq^{\lceil m/2\rceil}/(q^m-1)\rfloor, n\}$ *has dimension*

$$k = n - m(\delta - 1)(1 - 1/q).$$

Putting all these together, we get corollary 4.17.

Corollary 4.17. *Let* $C = C^{(1)}_{\mathrm{PG}}(2, 1, 0, s, 2)$, $n = 2^{2s} + 2^s + 1$ *and* $\delta \leq 2^{s/2} + 1$. *Then there exists an asymmetric*

$$[[n, n - 3^s - 3s\lceil(\delta - 1)/2\rceil - 1, \delta/2^s + 2]]_2$$

stabilizer code.

Proof. From Kou *et al.* (2001, eqn (21)), we know that $C = C^{(1)}_{\mathrm{PG}}(2, 1, 0, s, 2)$ is a $[n,\ n - 3^s - 1,\ 2^s + 2]_2$ code. It can be easily verified that the narrow-sense binary BCH code $\mathcal{BCH}(\delta \leq 2^{s/2} + 1, n)$ containing the dual of C satisfies the requirements of lemma 4.16 and is an $[n, n - 3s\lceil(\delta - 1)/2\rceil, \geq\delta]_2$ code. The asymmetric stabilizer code now follows from theorem 4.15. ■

5. Performance results

We now give the performance results of some of the codes constructed in §4. First, we will give the details about the channel model and how the simulations are performed.

Basically, asymmetric stabilizer codes can give us two benefits. Firstly, over an asymmetric quantum channel, they can give lower error rates compared with symmetric codes. We assume in this case we are comparing codes of same rates. Secondly, for the same error rates over asymmetric channels, the asymmetric codes can give higher data rates. Let us demonstrate these benefits of asymmetric quantum codes by means of some simple examples. In order to be fair, we will compare codes of similar decoding complexity.

In the simulations, we make a simplifying assumption about the channel model which was first used in MacKay *et al.* (2004) to simplify the simulation of quantum LDPC codes. We assume that the overall probability of error in the channel is given by p, while the individual probabilities of X, Y and Z errors are $p_x = p/(A+2)$, $p_y = p/(A+2)$ and $p_z = pA/(A+2)$, respectively. The exact performance would require us to simulate a 4-ary channel and also account for the fact that some errors can be estimated modulo the stabilizer. However, we observed that if we ignored the effect of the stabilizer it does not change the error rates because these codes are non-degenerate. The 4-ary channel can be modelled as two binary symmetric channels (BSCs) one modelling the bit-flip channel and the other the phase-flip channel. For exact performance, these two channels should be dependent; however, a good approximation is to model the channel as two independent BSCs with crossover probabilities $p_x + p_y = 2p/(A+2)$ and $p_y + p_z = p(A+1)/(A+2)$. In this case, the overall error rate in the quantum channel is the sum of the error rates in the two BSCs. While this approach is going to slightly overestimate the error rates, nonetheless it is useful and commonly used in literature. Since the X-channel uses a BCH code and decoded using a bounded distance decoder, we can just compute P^x_e, the X-error rate, in closed form as

$$P^x_e = 1 - \sum_{j=0}^{\lfloor (d-1)/2 \rfloor} \binom{n}{j} (p_x + p_y)^j (1 - p_x - p_y)^{n-j}. \tag{5.1}$$

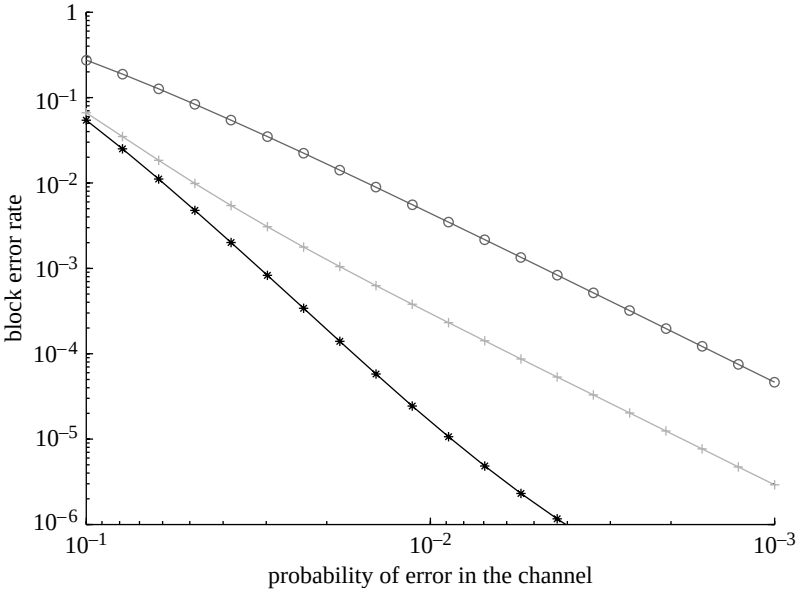

Figure 1. Performance of a [[15, 1, 3/7]] code described in the text for choices $A=1$ (circles), 10 (pluses), 100 (asterisks) of the channel asymmetry.

The error rate in the Z-channel, P_e^z, is obtained through simulations. The overall error rate is

$$P_e = 1-(1-P_e^x)(1-P_e^z) = P_e^x + P_e^z - P_e^x P_e^z \approx P_e^x + P_e^z.$$

Example 5.1 (Illustrating improvement of error rate). The highest distance possible for a (symmetric) [[15, 1]] code is 5. However, should we use an asymmetric [[15, 1]] code, then we can get an [[15, 1, 3/7]] code. This code can be constructed by choosing $C_x = \mathcal{BCH}(3)$ and $C_z = \mathcal{BCH}(7)$ in lemma 3.1, both of length 15. We now consider the performance of quantum codes over a quantum channel of the form

$$\mathcal{E}(\rho) = (1-p)\rho + p_x X\rho X + p_y Y\rho Y + p_z Z\rho Z,$$

where $p=p_x+p_y+p_z$; $p_x=p_y=p_z/A$. As we vary the channel asymmetry $A=p_z/p_x=p_z/p_y$, we keep the probability of error $p=p_x+p_y+p_z$ constant. On a symmetric channel, the asymmetric code is going to fare worse than the symmetric code. If we increase the asymmetry of the channel then we can see that the code performance improves. Figure 1 shows the performance of [[15, 1, 3/7]] code over various asymmetric channels.

While it is not shown here, the performance of the symmetric code does not change too much as the channel asymmetry is varied. The key observation is that increasing channel asymmetry improves the performance of the asymmetric code. Figure 2 compares the performance of the symmetric and asymmetric codes. For clarity, the asymmetric code's performance is shown over three values of channel asymmetry, while the symmetric code's performance is shown only when the channel asymmetry is 100, since this is the point of interest for us.

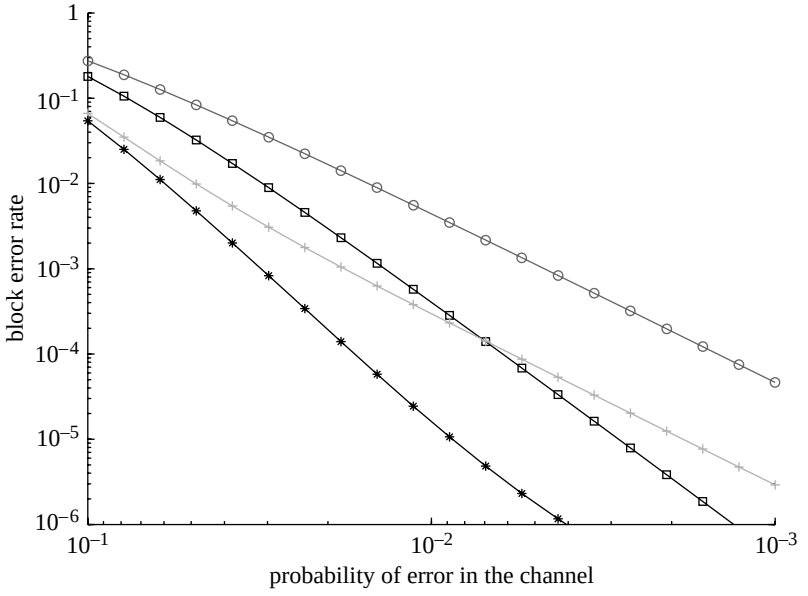

Figure 2. Comparison between the performance of a [[15, 1, 3/7]] that has been described in the text with the performance of a symmetric [[15, 1, 5]]. The shown curves are the plots for choices of asymmetry parameters $A=1, 10, 100$ for the asymmetric code and the $A=100$ for the symmetric code. Squares, QECC $A=100$; circles, AQECC $A=1$; pluses, AQECC $A=10$; asterisks, AQECC $A=100$.

Example 5.2 (Illustrating data rate gain). Let us now compare the performance of a [[31, 1, 7]] quantum BCH code with its asymmetric counterpart. There exists a [[31, 11, 3/7]] quantum BCH code. In figure 3, we see that over a channel with asymmetry $A=100$, the performance of the asymmetric code is comparable to that of the symmetric code, but we are able to operate at a rate 10 times larger.

(*a*) *Decoding LDPC codes*

The LDPC codes were decoded using an algorithm similar to the hard decision bit-flipping algorithm given by Kou *et al.* (2001). This is an instance of the bit-flipping algorithm originally given by Gallager. The maximum number of iterations for decoding is set to 50. A small modification had to be made to accommodate the special situation of quantum syndrome decoding. By measuring the generators of the stabilizer group, we obtain a classical syndrome, which, due to the fact that only ± 1 eigenspaces occur in all of the generators, is hard information. We use the syndrome as shown in figure 4 and initialize all the bit nodes with 0 at the start of the algorithm. Then the algorithm proceeds in the usual fashion as in Kou *et al.* (2001). We implemented this algorithm and ran several simulations, which are described next.

In figure 5, we see the performance of [[255, 159, 5/17]], given in table 3, as the channel asymmetry is varied from 1 to 100. As we observed with the BCH asymmetric codes in §5, here also we see that as we increase the asymmetry the code starts to perform better. As the asymmetry is increased, eventually the performance of the quantum code approaches the performance of the classical LDPC code.

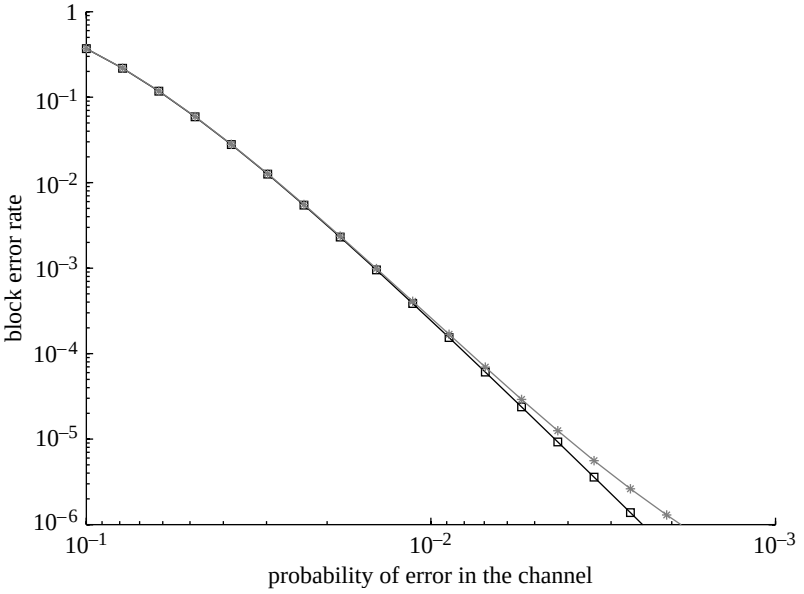

Figure 3. Comparison of the performance of a symmetric [[31, 1, 7]] (squares) code and the performance of an asymmetric [[31, 11, 3/7]] (asterisks) code for channel asymmetry $A = 100$.

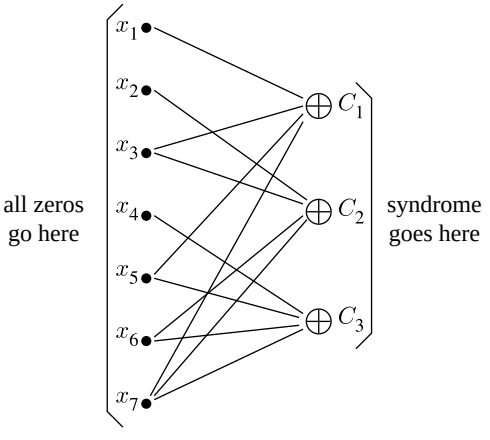

Figure 4. Modification of the iterative message-passing algorithm to the quantum case. The initialization step is different from the classical case as no soft information from the channel is available but rather only hard information about the measured syndrome is available. The algorithm begins with initializing all bit nodes to 0 and the check nodes with the syndrome. From then on, any classically known method for iterative decoding can be applied. In the figure, this principle is shown for the example of a classical [7, 4, 3] Hamming code. Application to the quantum case is straightforward as the decoding algorithm only works with classical information to compute the most likely error.

Tolerating a little rate loss improves the performance as can be seen from figure 6, especially at low channel asymmetries. If we increase the distance of the BCH code, the code becomes more tolerant to variations in channel asymmetry as can be seen by the performance of [[255, 143, 9/17]] in figure 7. This plot also

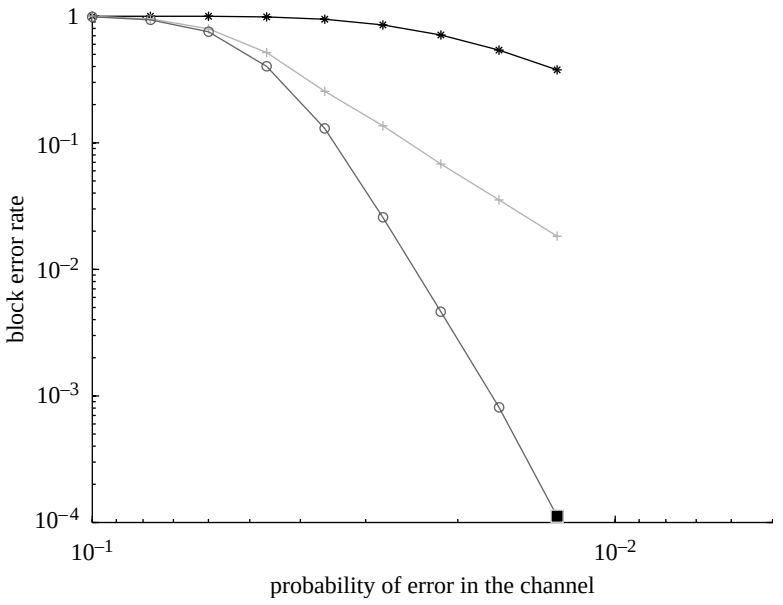

Figure 5. Performance of a [[255, 159, 5/17]] code described in the text for choices $A=1$ (asterisks), 10 (pluses), 100 (circles) of the channel asymmetry; filled square = (X: 0.01292, Y: 0.0001121).

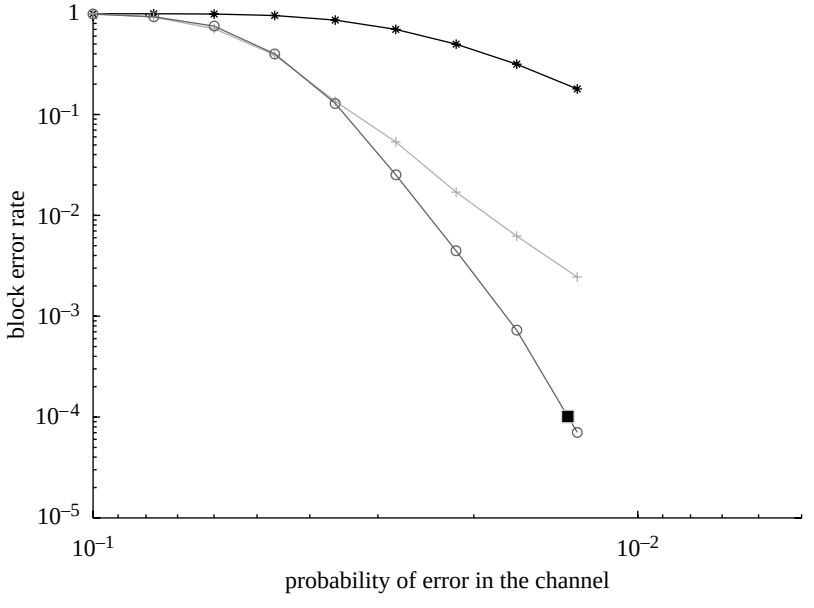

Figure 6. Performance of a [[255, 151, 7/17]] code described in the text for choices $A=1$ (asterisks), 10 (pluses), 100 (circles) of the channel asymmetry; filled square = (X: 0.01344, Y: 0.0001013).

illustrates an important point. Our channel model assumes that as we vary the channel asymmetry we keep the total probability of error in the channel fixed. This implies that while the probability of X-errors goes down, the probability of Z-errors tends to p, the total probability of error. Hence, the reduction in error

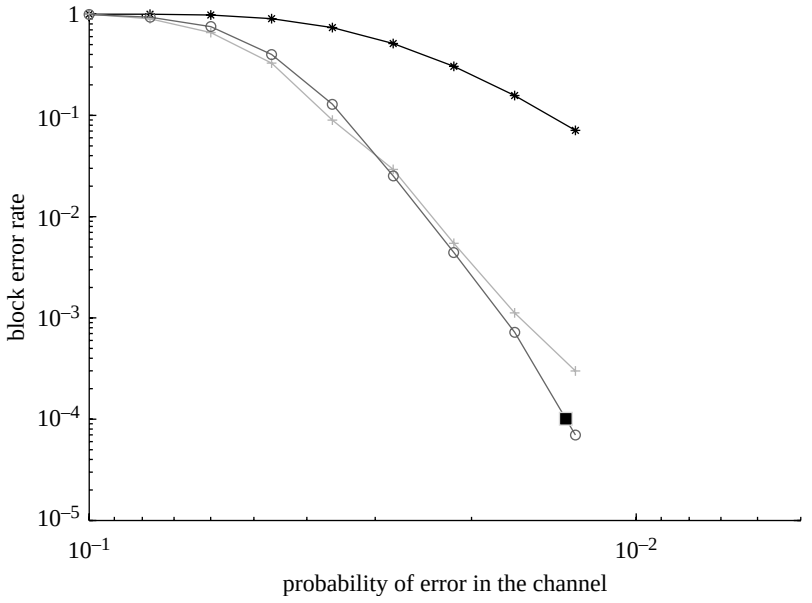

Figure 7. Performance of a [[255, 143, 9/17]] code described in the text for choices $A=1$ (asterisks), 10 (pluses), 100 (circles) of the channel asymmetry; filled square=(X: 0.01345, Y: 0.000101).

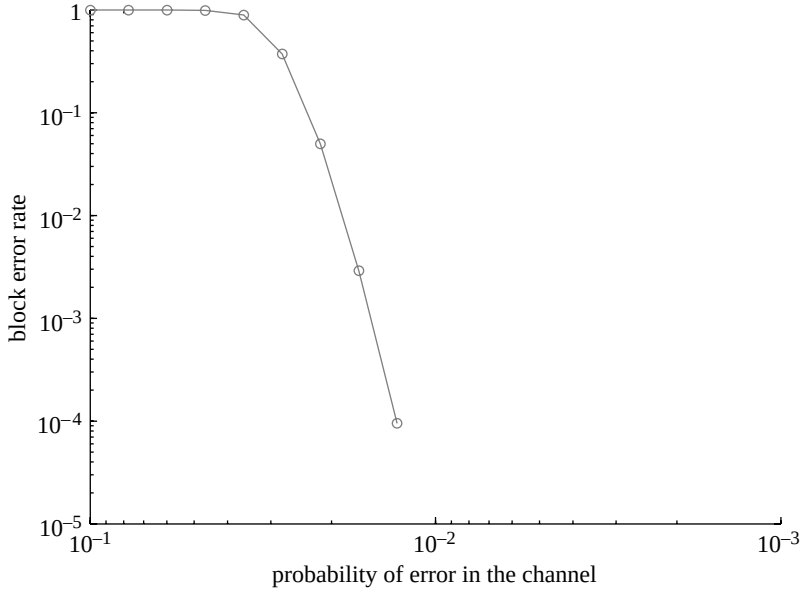

Figure 8. Performance of [[1023, 731, 11/33]] code for $A=100$.

rate in the X-channel must more than compensate for the increase in Z-error rate. If on the other hand, we had fixed the probability of error in the Z-channel and varied the channel asymmetry then we would observe a monotonic improvement in the error rate because on one hand the Z-error rate does not change but the X-error rate does.

Comparing with the short length codes of §5 we see that the error rate falls much more sharply. Larger length codes can potentially give better error rates. The performance of a code with length 1023 is shown in figure 8.

6. Conclusion and discussion

We have presented several new results regarding asymmetric quantum error-correcting codes, namely linear programming bounds on the feasible parameters, constructions based on nested families of classical codes and a CSS construction based on LDPC/BCH pairs. Furthermore, we have emphasized that for the simulation of these codes a slight modification of the standard classical belief propagation type simulation is necessary. We have carried out performance simulations for some asymmetric quantum LDPC codes and found that as expected the performance is a function of the channel asymmetry.

The question naturally raises how the codes presented in this work, in particular those obtained by the LDPC/BCH construction presented in §4, compare with the codes proposed by Ioffe & Mézard (2007). Strictly speaking, both constructions have regimes where they can perform better than the other. But it appears that the algebraically constructed asymmetric codes have the following benefits with respect to the randomly constructed ones of Ioffe & Mézard (2007).

— They give comparable performance and higher data rates with shorter lengths.
— The benefits of classical algebraic LDPC codes are inherited as low error floors compared with the random LDPC codes.
— The code construction is of lower complexity.

Our rationale for these benefits is as follows. Please note that the method of Ioffe & Mézard (2007) relies on random LDPC codes. The claim that we require short lengths follows from the fact that the algebraic constructions (considered in this paper) are better than random LDPC codes with respect to finite-length effects such as error floors. Since the quantum codes inherit the properties of the associated classical codes, we expect the codes proposed here to fare better than those of Ioffe & Mézard (2007) in this regard. This is in addition to the empirical observations that we construct codes at lengths of 256 with rates of approximately 1/2 and performance comparable with those of Ioffe & Mézard (2007), whose lengths are 1024 or greater. The construction of Ioffe & Mézard (2007) is not systematic in the sense it relies on a random choice of the code words of the BCH to construct the LDPC code. The construction is also more complex and it is not clear if the method actually terminates with a good LDPC code with high probability. By contrast, for the codes proposed in this paper, the design parameters completely determine the structure of the LDPC code and the associated asymmetric quantum code.

Our codes also offer flexibility in the rate and performance of the code because we can choose many possible BCH codes for a given finite geometry LDPC code or vice versa. The flip side, however, is that the codes given here have a slightly higher complexity of decoding.

Open problems are to study the performance of alternatives to the hard decision bit algorithm used in the performance simulations. Alternatives would be to study other ways of message passing such as weighted bit flipping, belief propagation, etc.

Finally, it will be imperative to study whether the codes constructed in this paper can be used to perform universal fault-tolerant quantum computing on them without changing the bias in the error model.

The authors would like to thank Marcus Silva for many useful discussions and for proposing the combined amplitude damping and dephasing channel, which is our main motivating example. We would also like to thank the referees for their detailed comments on the paper and for bringing to our attention relevant work by Knill *et al.* (1996) and Steane (1996). One of us (P.K.S.) would like to acknowledge the hospitality of NEC Laboratories America, Inc., wherein part of this research was conducted. This research was also supported by NSF CAREER award 0347310 and NSF grant CCF 0622201.

Appendix A. Proof of theorem 2.1

Theorem A.1. *Given a combined amplitude damping and dephasing channel $\mathcal{E}$ as above, the associated Pauli-twirled channel is of the form*

$$\mathcal{E}_T(\rho) = (1-p_x-p_y-p_z)\rho + p_x X\rho X + p_y Y\rho Y + p_z Z\rho Z,$$

where $p_x = p_y = (1-\mathrm{e}^{-t/T_1})/4$ and $p_z = 1/2 - p_x - (1/2)\mathrm{e}^{-t/T_2}$. In particular,

$$\frac{p_z}{p_x} = 1 + 2\frac{1-\exp(t/T_1(1-T_1/T_2))}{\mathrm{e}^{t/T_1}-1}.$$

If $t \ll T_1$, then we can approximate this ratio as $2T_1/T_2 - 1$.

Proof. A Kraus operator-sum decomposition of $\mathcal{E}$ is given by

$$\left.\begin{aligned} \mathcal{E}(\rho) = \sum_{k=0}^{2} A_k \rho A_k^\dagger, \quad \text{where } A_0 = \begin{bmatrix} 1 & 0 \\ 0 & \sqrt{1-\lambda-\gamma} \end{bmatrix}; \\ A_1 = \begin{bmatrix} 0 & 0 \\ 0 & \sqrt{\lambda} \end{bmatrix}; \quad A_2 = \begin{bmatrix} 0 & \sqrt{\gamma} \\ 0 & 0 \end{bmatrix}; \end{aligned}\right\} \tag{A 1}$$

and $\sqrt{1-\gamma-\lambda} = \mathrm{e}^{-t/T_2}$, $1-\gamma = \mathrm{e}^{-t/T_1}$. We can rewrite the Kraus operators A_i as

$$A_0 = \frac{1+\sqrt{1-\lambda-\gamma}}{2}\mathbf{1} + \frac{1-\sqrt{1-\lambda-\gamma}}{2} Z,$$

$$A_1 = \frac{\sqrt{\lambda}}{2}\mathbf{1} - \frac{\sqrt{\lambda}}{2} Z,$$

$$A_2 = \frac{\sqrt{\gamma}}{2} X - \frac{\sqrt{\gamma}}{2i} Y.$$

Rewriting $\mathcal{E}(\rho)$ in terms of Pauli matrices yields

$$\mathcal{E}(\rho) = \frac{2-\gamma+2\sqrt{1-\lambda-\gamma}}{4}\rho + \frac{\gamma}{4}X\rho X + \frac{\gamma}{4}Y\rho Y + \frac{2-\gamma-2\sqrt{1-\lambda-\gamma}}{4}Z\rho Z - \frac{\gamma}{4}\mathbf{1}\rho Z - \frac{\gamma}{4}Z\rho\mathbf{1} + \frac{\gamma}{4i}X\rho Y - \frac{\gamma}{4i}Y\rho X. \tag{A2}$$

It follows that the Pauli-twirled channel $\mathcal{E}_T$ is of the claimed form (see Dankert *et al.* 2006, lemma 2). In particular, note that twirling removes the asymmetric terms in equation (A 2).

Computing the ratio p_z/p_x, we obtain

$$\frac{p_z}{p_x} = \frac{2-\gamma-2\sqrt{1-\lambda-\gamma}}{\gamma} = \frac{1+\mathrm{e}^{-t/T_1}-2\mathrm{e}^{-t/T_2}}{1-\mathrm{e}^{-t/T_1}} = 1+2\frac{\mathrm{e}^{-t/T_1}-\mathrm{e}^{-t/T_2}}{1-\mathrm{e}^{-t/T_1}}$$

$$= 1+2\frac{1-\mathrm{e}^{t/T_1-t/T_2}}{\mathrm{e}^{t/T_1}-1} = 1+2\frac{1-\exp(t/T_1(1-T_1/T_2))}{\mathrm{e}^{t/T_1}-1}.$$

If $t \ll T_1$, then we can approximate the ratio as $2T_1/T_2-1$, as claimed. ■

References

Aliferis, P. & Preskill, J. 2008 Fault-tolerant quantum computation against biased noise. *Phys. Rev. A* **78**, 052331. (doi:10.1103/PhysRevA.78.052331)

Aly, S. A., Klappenecker, A. & Sarvepalli, P. K. 2007 On quantum and classical BCH codes. *IEEE Trans. Inform. Theory* **53**, 1183–1188. (doi:10.1109/TIT.2006.890730)

Assmus Jr, E. & Key, J. 1998 Polynomial codes and finite geometries. In *Handbook of coding theory*, vol. II (eds V. Pless & W. Huffman), pp. 1269–1343. Amsterdam, The Netherlands: Elsevier.

Blahut, R. E. 2003 *Algebraic codes for data transmission*. Cambridge, UK: Cambridge University Press.

Calderbank, A., Rains, E., Shor, P. & Sloane, N. 1998 Quantum error correction via codes over GF(4). *IEEE Trans. Inform. Theory* **44**, 1369–1387. (doi:10.1109/18.681315)

Camara, T., Ollivier, H. & Tillich, J.-P. 2007 A class of quantum LDPC codes: construction and performances under iterative decoding. In *Proc. 2007 IEEE Int. Symp. on Information Theory, Nice, France*, pp. 811–815.

Dankert, C., Cleve, R., Emerson, J. & Livine, E. 2006 Exact and approximate unitary 2-designs: constructions and applications. (http://arxiv.org/abs/quant-ph/0606161)

Delsarte, P. 1975 On subfield subcodes of Reed–Solomon codes. *IEEE Trans. Inform. Theory* **21**, 575–576. (doi:10.1109/TIT.1975.1055435)

DiVincenzo, D. P., Leung, D. W. & Terhal, B. M. 2002 Quantum data hiding. *IEEE Trans. Inform. Theory* **48**, 580–599. (doi:10.1109/18.985948)

Emerson, J., Alicki, R. & Zyczkowski, K. 2005 Scalable noise estimation with random unitary operators. *J. Opt. B: Quantum Semiclassical Opt.* **7**, S347–S352. (doi:10.1088/1464-4266/7/10/021)

Emerson, J., Silva, M., Ryan, C., Laforest, M., Baugh, J., Cory, D. G. & Laflamme, R. 2007 Symmetrized characterization of noise quantum processes. *Science* **317**, 1893–1896. (doi:10.1126/science.1145699)

Evans, Z. W. E., Stephens, A. M., Cole, J. H. & Hollenberg, L. C. L. 2007 Error correction optimisation in the presence of X/Z asymmetry. (http://arxiv.org/abs/0709.3875)

Fletcher, A. S., Shor, P. W. & Win, M. Z. 2008 Structured near-optimal channel-adapted quantum error correction. *Phys. Rev. A* **77**, 012320. (doi:10.1103/PhysRevA.77.012320)

Hagiwara, M. & Imai, H. 2007 Quantum quasi-cyclic LDPC codes. In *Proc. 2007 IEEE Int. Symp. on Information Theory, Nice, France*, pp. 806–810.

Huffman, W. C. & Pless, V. 2003 *Fundamentals of error-correcting codes*. Cambridge, UK: Cambridge University Press.

Ioffe, L. & Mézard, M. 2007 Asymmetric quantum error-correcting codes. *Phys. Rev. Lett. A* **75**, 032345. (doi:10.1103/PhysRevA.75.032345)

Kasami, T. & Lin, S. 1971 On majority-logic decoding for duals of primitive polynomial codes. *IEEE Trans. Inform. Theory* **17**, 322–331. (doi:10.1109/TIT.1971.1054640)

Kasami, T., Lin, S. & Peterson, W. W. 1968*a* New generalizations of the Reed–Muller codes. Part I: primitive codes. *IEEE Trans. Inform. Theory* **14**, 189–199. (doi:10.1109/TIT.1968.1054127)

Kasami, T., Lin, S. & Peterson, W. W. 1968*b* Polynomial codes. *IEEE Trans. Inform. Theory* **14**, 807–814. (doi:10.1109/TIT.1968.1054226)

Knill, E., Laflamme, R. & Zurek, W. 1996 Threshold accuracy for quantum computation. (http://arxiv.org/abs/quant-ph/9610011)

Kou, Y., Lin, S. & Fossorier, M. P. C. 2001 Low-density parity check codes based on finite geometries: a rediscovery and new results. *IEEE Trans. Inform. Theory* **47**, 2711–2736. (doi:10.1109/18.959255)

Leung, D. W., Nielsen, M. A., Chuang, I. L. & Yamamoto, Y. 1997 Approximate quantum error correction can lead to better codes. *Phys. Rev. A* **56**, 2567–2573. (doi:10.1103/PhysRevA.56.2567)

Lin, S. & Costello Jr, D. J. 2004 *Error control coding*, 2nd edn. Englewood Cliffs, NJ: Prentice Hall.

MacKay, D. J. C., Mitchison, G. & McFadden, P. L. 2004 Sparse-graph codes for quantum error correction. *IEEE Trans. Inform. Theory* **50**, 2315–2330. (doi:10.1109/TIT.2004.834737)

MacWilliams, F. & Sloane, N. 1977 *The theory of error-correcting codes*. Amsterdam, The Netherlands: North-Holland.

Nielsen, M. & Chuang, I. 2000 *Quantum computation and quantum information*. Cambridge, UK: Cambridge University Press.

Rahn, B., Doherty, A. C. & Mabuchi, H. 2002 Exact performance of concatenated quantum codes. *Phys. Rev. A* **66**, 032304. (doi:10.1103/PhysRevA.66.032304)

Sarvepalli, P. K., Rötteler, M. & Klappenecker, A. 2008 Asymmetric quantum LDPC codes. In *Proc. 2008 IEEE Int. Symp. on Information Theory, Toronto, Canada*, pp. 305–309.

Silva, M., Magesan, E., Kribs, D. W. & Emerson, J. 2007 Experimentally scalable protocol for identification of correctable codes. (http://arxiv.org/abs/0710.1900)

Steane, A. 1996 Simple quantum error correcting codes. *Phys. Rev. A* **54**, 4741–4751. (doi:10.1103/PhysRevA.54.4741)

Steane, A. 1999*a* Enlargement of Calderbank–Shor–Steane quantum codes. *IEEE Trans. Inform. Theory* **45**, 2492–2495. (doi:10.1109/18.796388)

Steane, A. 1999*b* Quantum Reed–Muller codes. *IEEE Trans. Inform. Theory* **45**, 1701–1703. (doi:10.1109/18.771249)

Stephens, A. M., Evans, Z. W. E., Devitt, S. J. & Hollenberg, L. C. L. 2007 Asymmetric quantum error correction via code conversion. (http://arxiv.org/abs/0708.3969)

Tang, H., Lin, S. & Abdel-Ghaffar, K. A. S. 2005 Codes over finite geometries. *IEEE Trans. Inform. Theory* **51**, 572–596. (doi:10.1109/TIT.2004.840867)