Non-catastrophic Encoders and Encoder Inverses for Quantum Convolutional Codes

Markus Grassl
Institut für Algorithmen und Kognitive Systeme
Fakultät für Informatik, Universität Karlsruhe (TH)
Am Fasanengarten 5, 76128 Karlsruhe, Germany
Email: grassl@ira.uka.de

Martin Rötteler
NEC Laboratories America, Inc.
4 Independence Way, Suite 200
Princeton, NJ 08540, U.S.A.
Email: mroetteler@nec-labs.com

Abstract— We present an algorithm to construct quantum circuits for encoding and inverse encoding of quantum convolutional codes. We show that any quantum convolutional code contains a subcode of finite index which has a non-catastrophic encoding circuit. Our work generalizes the conditions for non-catastrophic encoders derived in a paper by Ollivier and Tillich (quant-ph/0401134) which are applicable only for a restricted class of quantum convolutional codes. We also show that the encoders and their inverses constructed by our method naturally can be applied online, i. e., qubits can be sent and received with constant delay.

I. INTRODUCTION

Similar to the classical case a quantum convolutional code encodes an incoming stream of quantum information into an outgoing stream. A theory of quantum convolutional codes based on infinite stabilizer matrices has been developed recently, see [12]. While some constructions of quantum convolutional codes are known, see [2], [3], [1], [5], [6], [7], [11], [12], some very basic questions about the structure of quantum convolutional codes and their encoding circuits have not been addressed so far, respectively have been addressed only in special cases. In this paper we focus on the question of which quantum convolutional codes have non-catastrophic encoders, respectively inverse encoders.

Recall, that classically a code encoded by a catastrophic encoder has the unwanted property that—after code word estimation—a finite number of error locations can be mapped by the inverse encoder to an infinite number of error locations. For classical convolutional codes it is well-known that the non-catastrophicity condition is a property of the encoder and not of the code itself. Indeed, every convolutional code has both catastrophic and non-catastrophic encoders and therefore the choice of a good encoder is very important.

In this paper we address the analogous question whether any quantum convolutional stabilizer code has non-catastrophic encoders and encoder inverses. Here the condition to be non-catastrophic has been shown in [12] to be that it has a constant depth encoder whose elementary quantum gates can be arranged in form of a "pearl necklace", i. e., a regular structure in which blocks are only allowed to overlap with their neighbors with possibly some blocks spaced out. Furthermore, in [12] some conditions on the code have been given under which a non-catastrophic encoder exists. However, these conditions

are quite strict and not applicable to an arbitrary quantum convolutional code.

Using the matrix description of quantum convolutional stabilizer codes and transformations on this matrix which preserve the symplectic orthogonality, we show that a normal form can be achieved which corresponds to a very simple convolutional code. Reducing the dimension of this code by only a bounded factor, we obtain an even simpler code allowing online encoding and decoding. Furthermore, from the sequence of transformations one can read off a non-catastrophic encoder for a subcode of the original code whose dimension is reduced by the same factor. Asymptotically, the rate of the subcode and the original code are the same.

II. QUANTUM CONVOLUTIONAL CODES

Quantum convolutional codes are defined as infinite versions of quantum stabilizer codes. We briefly recall the necessary definitions and the polynomial formalism to describe quantum convolutional codes which was introduced in [12].

Definition 1 (Infinite Pauli Group): Let

$$X=\left(\begin{array}{cc} 0 & 1 \\ 1 & 0 \end{array}\right), Z=\left(\begin{array}{cc} 1 & 0 \\ 0 & -1 \end{array}\right), Y=XZ=\left(\begin{array}{cc} 0 & -1 \\ 1 & 0 \end{array}\right)$$

be the (real version of the) 2×2 Pauli matrices. Consider an infinite set of qubits labeled by the nonnegative integers $\mathbb{N}$. Let $M \in \{X, Y, Z\}$ be a Pauli matrix. We denote by M_i the semi-infinite tensor product $I_2 \otimes \ldots \otimes I_2 \otimes M \otimes I_2 \otimes \ldots$, where M operates on qubit i and I_2 denotes the identity matrix of size 2×2. The group generated by all X_i and Z_i for $i \in \mathbb{N}$ is called the infinite Pauli group $\mathcal{P}_\infty$. For an element $A = A_1 \otimes A_2 \otimes \ldots \in \mathcal{P}_\infty$ the positions in which A_i is not equal to $\pm I_2$ is called the support of A.

In the theory of block stabilizer codes, the elements of the Pauli group are labeled by tuples of binary vectors. Similarly, we can label the elements of the infinite Pauli group by a tuple of binary sequences, each of which is represented by a formal power series. Hence we get the correspondence

$$(-1)^c X_{\boldsymbol{\alpha}} Z_{\boldsymbol{\beta}} \quad := \quad (-1)^c \bigotimes_{\ell \geq 0} X^{\alpha_\ell} Z^{\beta_\ell}$$

$$\hat{=} \quad \left(\sum_{\ell \geq 0} \alpha_\ell D^\ell, \sum_{\ell \geq 0} \beta_\ell D^\ell \right)$$

where $c \in \mathbb{F}_2$ and $\boldsymbol{\alpha} = \sum_{\ell \geq 0} \alpha_\ell D^\ell$ and $\boldsymbol{\beta} = \sum_{\ell \geq 0} \beta_\ell D^\ell$ are formal power series with coefficients in $\mathbb{F}_2$. In this representation, multiplication of elements of $\mathcal{P}_\infty$ corresponds to addition of the power series. Furthermore, shifting an element $A \in \mathcal{P}_\infty$ one qubit to the right corresponds to the multiplication of the power series by D. As we also allow to shift the operators by a bounded number of qubits to the left, we use Laurent series instead of power series to represent the elements of $\mathcal{P}_\infty$. An element $A \in \mathcal{P}_\infty$ with finite support corresponds to a tuple of Laurent polynomials. Recall that the field of Laurent series in the variable D with coefficients in $\mathbb{F}_2$ is denoted by $\mathbb{F}_2((D))$ and recall further that it contains the ring $\mathbb{F}_2[D, D^{-1}]$ of Laurent polynomials.

We are interested in shift invariant abelian subgroups of $\mathcal{P}_\infty$, more specifically in those subgroups which can be generated by a finite number of elements and their shifted versions. The following definition introduces a shorthand notation for describing such subgroups.

Definition 2 (Stabilizer Matrix): Let $\mathcal{S}$ be an abelian subgroup of $\mathcal{P}_\infty$ which has trivial intersection with the center of $\mathcal{P}_\infty$. Furthermore, let $\{g_1, g_2, \ldots, g_r\}$ where $g_i = (-1)^{c_i} X_{\boldsymbol{\alpha}_i} Z_{\boldsymbol{\beta}_i}$ with $c_i \in \{0,1\}$ and $(\boldsymbol{\alpha}_i, \boldsymbol{\beta}_i) \in \mathbb{F}_2((D))^n \times \mathbb{F}_2((D))^n$ be a minimal set of generators for $\mathcal{S}$. Then a stabilizer matrix of the corresponding quantum convolutional (stabilizer) code $\mathcal{C}$ is a generator matrix of the (classical) additive convolutional code $C \subseteq \mathbb{F}_2((D))^n \times \mathbb{F}_2((D))^n$ generated by $(\boldsymbol{\alpha}_i, \boldsymbol{\beta}_i)$. We will write this matrix in the form

$$S(D) = (X(D)|Z(D)) = \begin{pmatrix} \boldsymbol{\alpha}_1 & \boldsymbol{\beta}_1 \\ \vdots & \vdots \\ \boldsymbol{\alpha}_r & \boldsymbol{\beta}_r \end{pmatrix} \in \mathbb{F}_2((D))^{r \times 2n}. \tag{1}$$

In what follows we are only interested in those stabilizers which have a finite description. Hence we will consider only such stabilizer matrices (1) in which all entries are actually rational functions, i.e., elements of $\mathbb{F}_2(D)$. Eventually, we will require that all entries have finite support and are hence polynomials.

Alternatively to (1) a quantum convolutional code can also be described in terms of a semi-infinite stabilizer matrix S which has entries in $\mathbb{F}_2 \times \mathbb{F}_2$. The general structure of the matrix is as follows:

$$S := \begin{pmatrix} G_0 & G_1 & \ldots & G_m & 0 & \ldots \\ 0 & G_0 & G_1 & \ldots & G_m & 0 & \ldots \\ 0 & 0 & G_0 & G_1 & \ldots & G_m & 0 & \ldots \\ \vdots & & & \ddots & \ddots & & \ddots \end{pmatrix} \tag{2}$$

The matrix S has a block band structure where each block is of size $(n-k) \times (m+1)n$. All blocks have equal size and are comprised of $m+1$ matrices $G_0, G_1, \ldots, G_m$ which are of size $(n-k) \times n$ each. In the second block, these $m+1$ matrices are shifted by n columns, hence any two consecutive blocks overlap in $(m-1)n$ positions.

Similar to the classical case the link between the polynomial description of eq. (1) and the semi-infinite matrix eq. (2) is given by $S(D) := \sum_{i=0}^m G_i D^i$. The band structure of eq. (2)

implies that for every qubit in the semi-infinite stream of qubits, there is a bounded number of generators of the stabilizer group that act non-trivially on that position. Moreover, as these generators of the stabilizer group have bounded support, their eigenvalues can be measured when the corresponding qubits have been received. Therefore, it is possible to compute the error syndrome for the quantum convolutional code online.

Writing the stabilizer in the form $S(D) = (X(D)|Z(D))$ as in eq. (1), it was shown in [12] that the condition of symplectic orthogonality of the semi-infinite matrix S can be expressed compactly in the form

$$X(D)Z(1/D)^t + Z(D)X(1/D)^t = 0. \tag{3}$$

On the other hand, we can start with an arbitrary self-orthogonal additive convolutional code over $\mathbb{F}_2((D))^n \times \mathbb{F}_2((D))^n$ to define a convolutional quantum code. In general, the generator matrix for such a code may contain rational functions, but there is always an equivalent description in terms of a matrix with polynomial entries [9]. The following theorem shows that for self-dual convolutional codes, all entries of a systematic generator matrix are in fact Laurent polynomials.

Theorem 3: Let $S(D) = (X(D)|Z(D))$ with $X(D) = I$ be a stabilizer matrix of a self-dual additive convolutional code over the rational function field $\mathbb{F}_2(D)$. Then $Z(1/D) = Z(D)^t$ and all entries of $Z(D)$ are Laurent polynomials.

Proof: From condition (3) it follows that the code is self-dual if and only if $Z(1/D) = Z(D)^t$. Assume that $Z_{ij}(D)$ is a proper rational function and not a Laurent polynomial. Then evaluating the series expansion of $Z_{ij}(D)$ at $1/D$ yields infinitely many negative powers. However, since $Z_{ji}(D)$ contains only finitely many negative powers we get a contradiction. Hence all entries of $Z(D)$ have to be Laurent polynomials.

The symmetry $Z(1/D) = Z(D)^t$ additionally implies that the diagonal terms $Z_{ii}(D)$ are Laurent polynomials of the form

$$Z_{ii}(D) = \sum_{\ell=0}^d c_\ell (D^{-\ell} + D^\ell). \tag{4}$$

∎

III. SHIFT-INVARIANT CLIFFORD OPERATIONS

We are interested in quantum circuits which encode a convolutional quantum code. Recall that the controlled-not (CNOT) maps $|x\rangle|y\rangle \mapsto |x\rangle|x \oplus y\rangle$ and that the controlled-Z (CSIGN) gates maps $|x\rangle|y\rangle \mapsto (-1)^{x \cdot y}|x\rangle|y\rangle$ (see [10]). We want that errors which happen during the encoding do not be spread out too far. A particularly bad example of spreading errors is given by the cascade $\text{CNOT}_\infty = \prod_{i=0}^\infty \text{CNOT}^{(i,i+1)}$ where gates with smaller index i are applied first. The cascade CNOT_∞ maps the finite support element $X \otimes I_2 \otimes I_2 \otimes \ldots$ to the infinite support element $X \otimes X \otimes X \otimes \ldots$ On the other hand the infinite cascade $\text{CSIGN}_\infty = \prod_{i=0}^\infty \text{CSIGN}^{(i,i+1)}$ does not have this behavior: indeed, a Pauli matrix X_i is mapped to $Z_{i-1} X_i Z_{i+1}$ by this cascade and since it furthermore commutes with all Z

operators, this shows that it maps finite support Pauli matrices to finite support Pauli matrices. The reason for this difference is that the sequence CSIGN_∞ can be parallelized to have finite depth (actually depth 2), whereas this is not possible for CNOT_∞. Clearly, any circuit of constant depth only leads to a local error expansion, i.e., Pauli matrices with finite support get mapped onto Pauli matrices with finite support. This gives rise to the following definition:

Definition 4 (Non-catastrophic encoder): Let $\mathcal{C}$ be a quantum convolutional code and let $\mathcal{E}$ be an encoding circuit for $\mathcal{C}$. Then $\mathcal{E}$ is called *non-catastrophic* if the gates in $\mathcal{E}$ can be arranged into a circuit of finite depth.

In the following, we consider infinite cascades of gates from the Clifford group that can be realized by quantum circuits with constant depth. Since the generators for the quantum convolutional code are obtained by shifting a fixed block an infinite number of times, we have to impose a shift invariance condition on any Clifford gate that we intend to apply to the code. This means that whenever a gate is applied it has to be applied also in a shifted version by an offset of n qubits. Similar to the approach in [8], the action of such operations on elements of the infinite Pauli group can be described as linear transformations on the stabilizer matrix. As an example, the action of an infinitely replicated Hadamard gate H on a qubit is described in its action on the vectors $(f(D), g(D)) \in \mathbb{F}_2(D)^2$ by the matrix $\overline{H} = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$ since $H^\dagger X H = Z$ and $H^\dagger Z H = X$. Similarly, all infinitely replicated versions of Clifford gates which only operate within a block and do not connect qubits between shifted blocks, correspond to the usual matrices in the symplectic group $\mathrm{Sp}_{2n}(\mathbb{F}_2)$.

More interesting are those operations which connect different blocks which have been shifted in time. An example is a CNOT gate which operates on a qubit i (control) and qubit j (target), where qubit j has been shifted by ℓ blocks. Recall that shifting by ℓ blocks corresponds to multiplying by D^ℓ. In this case we obtain that CNOT gate maps the stabilizer vector $(x_1, x_2 | z_1, z_2) \mapsto (x_1, x_2 + x_1 D^\ell | z_1 + z_2 D^{-\ell}, z_2)$, i.e., X errors are propagated into the future and Z errors into the past. Note that by applying a sequence of CNOT gates we can actually map $(x_1, x_2 | z_1, z_2) \mapsto (x_1, x_2 + f(D)x_1 | z_1 + f(1/D)z_2, z_2)$, where $f(D) \in \mathbb{F}_2[D]$ is an arbitrary polynomial. A summary of the gates used is shown in Table I. It is important to note that all the operations shown in Table I can be parallelized to have constant depth.

IV. COMPUTING AN ENCODING CIRCUIT

In the following we describe an algorithm which operate on the stabilizer matrix (1) in order to produce a new stabilizer which is in a simpler form. We can act in two ways: (i) by applying row operations using an invertible matrix over $\mathbb{F}_2[D, D^{-1}]$. Apart from possible shifts, this does not change the stabilizer group, i.e., up to a possibly new initial qubit sequence (of bounded length) the quantum code is unchanged. We can also apply (ii) column operations given by an arbitrary element of the Clifford group shown in Table I. Before we state

TABLE I
ACTION OF VARIOUS CLIFFORD OPERATIONS.

unitary gate U	matrix $\overline{U}$
$H = \dfrac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} \in \mathbb{C}^{2\times 2}$	$\overline{H} = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \in \mathbb{F}_2^{2\times 2}$
$P = \begin{pmatrix} 1 & 0 \\ 0 & \exp(i\pi/2) \end{pmatrix} \in \mathbb{C}^{2\times 2}$	$\overline{P} = \begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix} \in \mathbb{F}_2^{2\times 2}$
$\mathrm{CNOT}^{(i,j+\ell n)}, i \not\equiv j \pmod n$	$\overline{\mathrm{CNOT}} = \left(\begin{array}{cc\|cc} 1 & D^\ell & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & D^{-\ell} & 1 \end{array} \right)$
$\mathrm{CSIGN}^{(i,j+\ell n)}, i \not\equiv j \pmod n$	$\overline{\mathrm{CSIGN}} = \left(\begin{array}{cc\|cc} 1 & 0 & 0 & D^\ell \\ 0 & 1 & D^{-\ell} & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{array} \right)$
$P_\ell := \mathrm{CSIGN}^{(i,i+\ell n)}, \ell \neq 0$	$\overline{P_\ell} = \begin{pmatrix} 1 & D^{-\ell} + D^\ell \\ 0 & 1 \end{pmatrix}$

Conjugation of the stabilizer group $\mathcal{S}$ by the unitary gate U corresponds to the action of the matrices $\overline{U}$ on the columns of the stabilizer matrix $S(D) = (X(D)|Z(D))$.

the algorithm we recall the Smith normal form [9] of a matrix:

Theorem 5: Let $M(D) \in \mathbb{F}_2[D]^{r\times n}$ be an $r \times n$ polynomial matrix. Then there exist polynomial matrices $A(D) \in \mathrm{GL}_r(\mathbb{F}_2[D])$ and $B(D) \in \mathrm{GL}_n(\mathbb{F}_2[D])$, both having determinant one, such that $M(D) = A(D)\Gamma(D)B(D)$, where $\Gamma(D)$ is the $r \times n$ matrix

$$\Gamma(D) = \begin{pmatrix} \gamma_1(D) & & & \\ & \ddots & & \\ & & \gamma_r(D) & 0 & \cdots & 0 \end{pmatrix},$$

where the diagonal elements (*elementary divisors*) $\gamma_i \in \mathbb{F}_2[D]$ satisfy $\gamma_i | \gamma_{i+1}$ for $i = 1, \ldots, r-1$.

Note that the Smith form can be computed for any matrix over an Euclidean domain, including the ring $\mathbb{F}_2[D, D^{-1}]$ of Laurent polynomials (see, e.g., [4]). For this, define the degree of a Laurent polynomial $f = \sum_{\ell=\ell_0}^{\ell_1} c_\ell D^\ell$ with $c_{\ell_0} \neq 0 \neq c_{\ell_1}$ as $|\ell_1 - \ell_0|$.

We will also need an observation about matrices which have already been partially brought into Smith form and which contain Laurent polynomials as entries.

Lemma 6: Let $M(D) \in \mathbb{F}_2[D, D^{-1}]^{r\times n}$ be a matrix containing Laurent polynomials and which has the form $M = (\mathrm{diag}(\gamma_i(D))|U(D))$, where $U(D) \in \mathbb{F}_2[D, D^{-1}]^{r\times(n-r)}$. Assume that for at least one i we have that γ_i does not divide the Laurent polynomials contained in the ith row of $U(D)$. Then at least one of the polynomials $\gamma'_i(D)$ arising in the

Smith normal form of $M(D)$ (after the denominators have been cleared by row-wise multiplication of powers of D) has a strictly smaller degree than the corresponding $\gamma_i(D)$.

Proof: Without loss of generality, we consider the first row $(\gamma_1(D), 0, \ldots, 0, f_1(D), \ldots, f_{n-r}(D))$ of $M(D)$, where the $f_i(D)$ are Laurent polynomials. Clearing the denominators by a suitable power D^ℓ leaves us with $D^\ell \gamma_1(D)$ and polynomials $f_i'(D) := D^\ell f_i(D)$. Computing the Smith normal form we obtain the gcd of $D^\ell \gamma_1(D), f_1'(D), \ldots, f_{n-r}'(D)$ which by assumption has to be a proper divisor of $\gamma_1(D)$. ∎

Next, observe that by using Clifford gates (acting on the X-part only) we can implement the matrix $B(D)$ used in the Smith normal form. The reason for this is that in the computation of the Smith normal form only elementary operations and permutations are necessary [9, Section 2.2]. We can realize these operations using the $\overline{\text{CNOT}}$ gates and permutations of the qubits, which can also be realized by $\overline{\text{CNOT}}$. Left multiplication by an invertible matrix does not change the stabilizer, so there is no need to implement the matrix $A(D)$ as quantum gates.

Algorithm 7: Let a polynomial stabilizer matrix $S(D) = (X(D)|Z(D)) \in \mathbb{F}_2[D]^{r \times 2n}$ of full rank be given.

1) Compute matrices $A(D)$ and $B(D)$ which realize the Smith normal form for $X(D)$. Factor the matrix $B(D)$ into elementary matrices of the form $\overline{\text{CNOT}}$ and permutations of qubits. Apply these operations to the code to obtain the new stabilizer matrix

$$S(D) = \left(\begin{array}{cc|cc} \Gamma(D) & 0 & Z_1(D) & Z_2(D) \\ 0 & 0 & & \end{array} \right),$$

where $\Gamma(D)$ is a diagonal matrix with non-zero polynomial entries of rank s and $Z_1(D) \in \mathbb{F}_2[D, D^{-1}]^{r \times s}$ and $Z_2(D) \in \mathbb{F}_2[D, D^{-1}]^{r \times (n-s)}$ are matrices with Laurent polynomials as entries.

2) While the $Z_2(D)$ part of $S(D)$ is not zero, repeat the following steps:
   - Use Hadamard gates $\overline{H}$ to swap $Z_2(D)$ into the X-part yielding

   $$S(D) = \left(\begin{array}{cc|cc} \Gamma(D) & X_2(D) & Z_1(D) & 0 \\ 0 & & & \end{array} \right),$$

   with $X_2(D) = Z_2(D)$.
   - If $\Gamma(D)$ has full rank and if all polynomials in row j of $X_2(D)$ are divisible by $\gamma_j(D)$ for all $j = 1, \ldots, r$, then use $\overline{\text{CNOT}}$-gates to obtain zeros in both $X_2(D)$ and $Z_2(D)$.
   - Else recompute the Smith normal form of the X-part and get either smaller elementary divisors or all polynomials in $Z_2(D)$ are multiples of the corresponding elementary divisor. The degree of the elementary divisors decreases because of Lemma 6.

3) The stabilizer matrix $S(D)$ is now of the form $S(D) = (\Gamma(D)0|Z_1(D)0)$, where $\Gamma(D)$ has a rational inverse since $S(D)$ has full rank.

4) From Theorem 3 it follows that all entries in the rows of $Z_1(D)$ are divisible (as Laurent polynomials) by the corresponding element of $\Gamma(D)$ (consider the matrix $\Gamma^{-1}S(D) = (I\,0|\Gamma^{-1}Z_1(D)\,0)$ which contains Laurent polynomials only). Hence, using $\overline{\text{CSIGN}}$ gates, clear all off-diagonal terms in $Z_1(D)$.

5) From (4) it follows that we can cancel the diagonal of the matrix $Z_1(D)$ using the gates $\overline{P}$ and $\overline{P_\ell}$.

6) Finally, use Hadamard gates $\overline{H}$ to obtain Z-only generators in diagonal form.

This algorithm transforms the original stabilizer matrix into a stabilizer matrix $S_1(D) := (0\,0|\Gamma(D)0)$ with $\Gamma(D) = \text{diag}(\gamma_i(D))$. In case $\gamma_i(D) = 1$, the only possible sequence of states formed by the ith qubit of all blocks is $|0\rangle|0\rangle \ldots$ If $\gamma_i(D) = D^\ell$, there are no constraints on the first ℓ qubits. Otherwise, the state $|c_0\rangle|c_1\rangle \ldots$ corresponding to the power series expansion of $1/\gamma_i(D) = \sum_{\ell \geq 0} c_\ell D^\ell$ and its shifted versions are allowed, too. As the sequence $(c_\ell)_\ell$ is periodic, there are only finitely many different shifted versions. We ignore these additional states as they would require an infinite cascade of CNOT gates. As an example for this behavior consider the states $|0\rangle|0\rangle \ldots$ and $|1\rangle|1\rangle \ldots$ allowed by the single qubit Z-generator $1 + D$.

In case $\Gamma(D) \neq I$, which corresponds to catastrophic encoders in the classical case, we consider the code $\mathcal{C}_0$ with stabilizer matrix $S_0(D) := (0\,0|I\,0)$. Now, $\mathcal{C}_0$ is a proper convolutional subcode of the code $\mathcal{C}_1$ with stabilizer matrix $S_1(D)$. The dimension is only decreased by a bounded factor depending on $\Gamma(D)$. In case $\Gamma(D) = I$, we have $\mathcal{C}_0 = \mathcal{C}_1$.

The subcode $\mathcal{C}_0$ has a very simple structure: a sequence of $n - k$ qubits in the state $|0\rangle$ alternates with a sequence of k qubits $|\phi_i\rangle$. Encoding for $\mathcal{C}_0$ is done by inserting qubits in the state $|0\rangle$ into the input stream. To obtain a state of (a subcode of) the original convolutional quantum code $\mathcal{C}$, apply the gates corresponding to the elementary matrices used in the algorithm in reversed order. The corresponding elementary gates are only Clifford gates which have to be replicated infinitely often. All elementary gates used can be parallelized into finite depth which implies that the operations can be carried out online. Hence we have shown the following result:

Corollary 8: Let $S(D)$ be the stabilizer matrix of a quantum convolutional code $\mathcal{C}$. Then there exists a convolutional subcode $\mathcal{C}_{\text{sub}} \subseteq \mathcal{C}$ with a non-catastrophic encoder and encoder inverse such that asymptotically the rates of $\mathcal{C}_{\text{sub}}$ and $\mathcal{C}$ are equal. Moreover, the encoder and its inverse only use Clifford gates and allow for online encoding and inverse encoding.

V. EXAMPLE

Consider the $\mathbb{F}_4$-linear rate-$1/3$ convolutional code from [6, Table VI]) with generator matrix

$$G(D) = \left(\begin{array}{ccc} 1 + D & 1 + \omega D & 1 + \overline{\omega} D \end{array} \right).$$

The corresponding stabilizer matrix is

$$S(D) = \left(\begin{array}{ccc|ccc} 1 + D & 1 & 1 + D & 0 & D & D \\ 0 & D & D & 1 + D & 1 + D & 1 \end{array} \right).$$
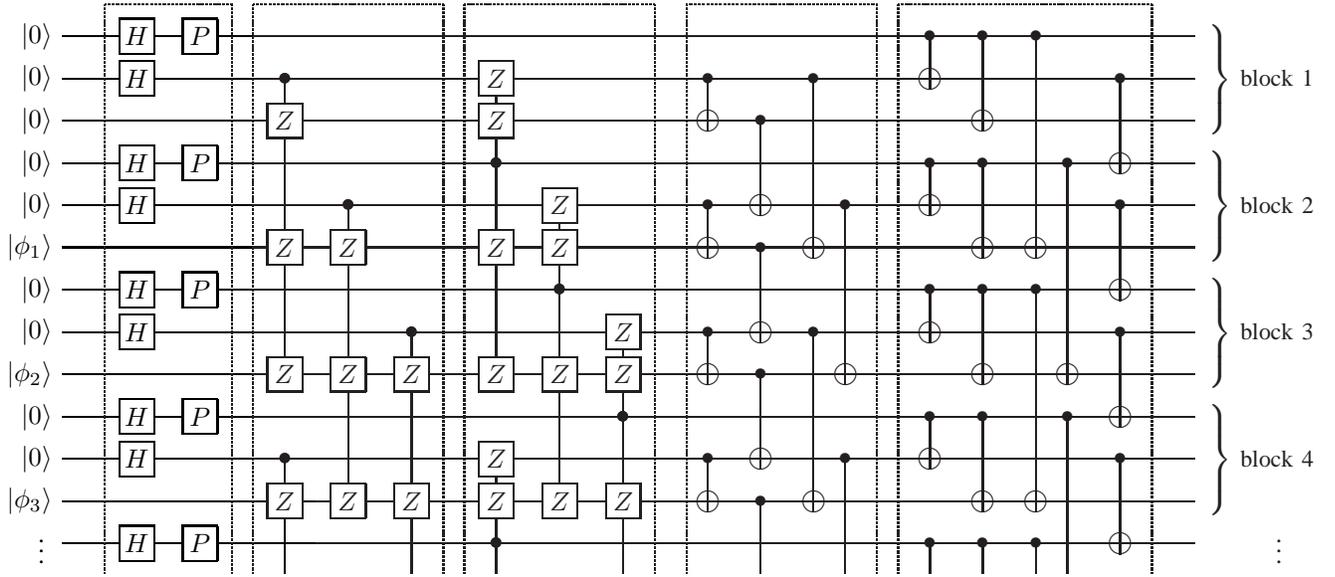
Fig. 1. Encoding circuit for a rate $1/3$ convolutional quantum codes. Every gate has to be repeatedly applied shifted by one block, i.e. three positions down. Note that the CSIGN gates are diagonal and hence can be arranged in any order. For the CNOT gates, each gate has to be repeated in its shifted version before the next gate can be applied.

The first sequence of $\overline{\text{CNOT}}$ operations transforms the first row of $X(D)$, and we obtain:

$$
\left(
\begin{array}{ccc|ccc}
1 & 0 & 0 & 1 & D & D \\
D^2 & D^2 + D & D^3 + D^2 + D & 0 & (D^2+1)/D & 1
\end{array}
\right)
$$

Invertible row-operations do not change the stabilizer group, so adding D^2 times the first row to the second yields

$$
\left(
\begin{array}{ccc|ccc}
1 & 0 & 0 & 1 & D & D \\
0 & D^2 + D & D^3 + D^2 + D & D^2 & (D^4+D^2+1)/D & D^3+1
\end{array}
\right).
$$

Again using $\overline{\text{CNOT}}$, we transform the second row of $X(D)$:

$$
\left(
\begin{array}{ccc|ccc}
1 & 0 & 0 & 1 & 1/D & (D^2+D+1)/D \\
0 & D & 0 & D^2 & 0 & D^3+D^2+D
\end{array}
\right). \tag{5}
$$

Using $\overline{\text{CSIGN}}$, we can clear the off-diagonal terms in the first row of $Z(D)$,

$$
\left(
\begin{array}{ccc|ccc}
1 & 0 & 0 & 1 & 0 & 0 \\
0 & D & 0 & 0 & 0 & D^3+D^2+D
\end{array}
\right),
$$

and similar for the second row

$$
\left(
\begin{array}{ccc|ccc}
1 & 0 & 0 & 1 & 0 & 0 \\
0 & D & 0 & 0 & 0 & 0
\end{array}
\right).
$$

Finally, using $\overline{P}$ and $\overline{H}$ we get Z-only generators:

$$
\left(
\begin{array}{ccc|ccc}
0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & D & 0
\end{array}
\right)
$$

To clear the entries in the first row of the Z-part of (5), we need $\overline{\text{CSIGN}}$ gates whose target lie in the block before that of the target (see the third set of gates in Fig. 1). Therefore, encoding can only start in the second block. In the first block, all qubits are initialized to $|0\rangle$. Additionally, we ignore that the term D in the second row implies that there is no operator Z_i

in the stabilizer acting on the second qubit of the now second block and constrain the input to $|0\rangle$.

A circuit for encoding is obtained by reversing the order of the transformations. The encoding circuit is illustrated in Fig. 1. Note that the circuit extends over three blocks, i.e., has total memory two. This is reflected by the fact that in this example the operations used by the algorithm to clear entries only involved Laurent polynomials of degree at most two. In contrast, an encoder for the classical convolutional code over $\mathbb{F}_4$ given by $G(D)$ can be realized with total memory one.

VI. CONCLUSION

We have shown that the quantum convolutional codes obtained from self-orthogonal classical convolutional codes always have a subcode which has asymptotically the same rate and allows for non-catastrophic encoders. This shows that errors affecting these codes do not propagate in an unbounded fashion during the decoding process. For simplicity, we have presented the algorithm for qubit systems only, but as in [8], the technique applies to non-qubit systems as well.

ACKNOWLEDGMENT

REFERENCES

[1] A. C. A. de Almeida and R. Palazzo, Jr., "A Concatenated $[(4,1,3)]$ Quantum Convolutional Code," in *2004 IEEE Information Theory Workshop*, San Antonio, TX, 2004.

[2] H. F. Chau, "Quantum convolutional error-correcting codes," *Phys. Rev. A*, vol. 58, no. 2, pp. 905–909, 1998.

[3] ——, "Good quantum-convolutional error-correction codes and their decoding algorithm exist," *Phys. Rev. A*, vol. 60, no. 3, pp. 1966–1974, 1999.

[4] I. Daubechies and W. Sweldens, "Factoring Wavelet Transforms into Lifting Steps," *The journal of Fourier analysis and applications*, vol. 4, no. 3, pp. 247–269, 1998.

[5] G. D. Forney, Jr. and S. Guha, "Simple Rate-$1/3$ Convolutional and Tail-Biting Quantum Error-Correcting Codes," in *Proceedings of the International Symposium on Information Theory (ISIT 05)*, 2005, pp. 1028–1032.

[6] G. D. Forney Jr., M. Grassl, and S. Guha, "Convolutional and tail-biting quantum error-correcting codes," 2005, preprint quant-ph/0511016, submitted to IEEE Transactions on Information Theory.

[7] M. Grassl and M. Rötteler, "Quantum block and convolutional codes from self-orthogonal product codes," in *Proceedings of the International Symposium on Information Theory (ISIT 05)*, 2005, pp. 1018–1022.

[8] M. Grassl, Th. Beth, and M. Rötteler, "Efficient quantum circuits for non-qubit quantum error-correcting codes," *International Journal of Foundations of Computer Science*, vol. 14, no. 5, pp. 757–775, 2003.

[9] R. Johannesson and K. S. Zigangirov, *Fundamentals of Convolutional Coding*, ser. IEEE Series on Digital and Mobile Communication. New York: IEEE Press, 1999.

[10] M. Nielsen and I. Chuang, *Quantum Computation and Quantum Information*. Cambridge University Press, 2000.

[11] H. Ollivier and J.-P. Tillich, "Description of a quantum convolutional code," *Phys. Rev. Lett.*, vol. 91, no. 17, p. 177902, 2003.

[12] ——, "Quantum convolutional codes: fundamentals," 2004, preprint quant-ph/0401134.