International DFG Workshop 2007
Quantum Information Processing
Cochem/Ediger-Eller, 28.–30. März 2007

Quantum Convolutional Codes

Markus Grassl

joint work with

Martin Rötteler, NEC Laboratories America, Inc.
David Forney, Saikat Guha, MIT
Salah A. Aly, Andreas Klappenecker, Pradeep Kiran Sarvepalli, Texas A&M

Fakultät für Informatik

Institut für Algorithmen und Kognitive Systeme
Fakultät für Informatik, Universität Karlsruhe (TH)
Germany
http://iaks-www.ira.uka.de/home/grassl

Motivation

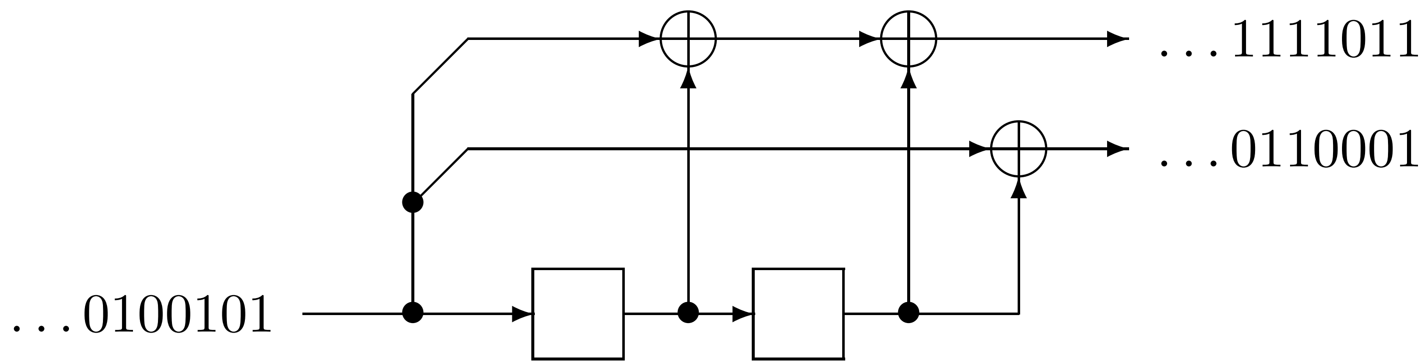
Problem: Send a stream of qubits along a channel

- Solution 1: Use the code $[\![5, 1, 3]\!]$ for every qubit
  $\Longrightarrow$ rate $1/5$, correcting one error in every block of five qubits
- Solution 2: Use a long code to encode larger blocks/all qubits
  $\Longrightarrow$ better rate & more errors can be corrected, but
  - more complicated operations
  - large delay
- Solution 3: Quantum Convolutional Codes
  - "online" encoding/decoding
  - local encoding/inverse encoding operations
  - good classical convolutional codes are used in practice

Classical Convolutional Codes

Based on linear shift registers

Example: Code of rate 1/2, min. distance 5:



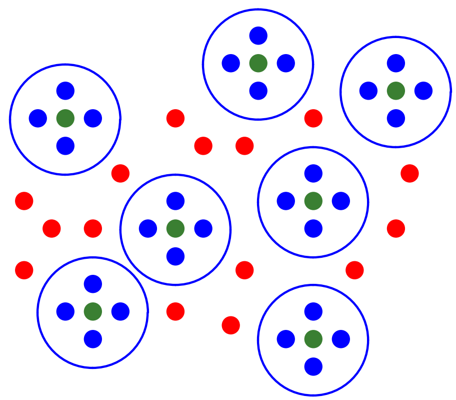
in general:

- outputs are linear functions of the input and the memory
- the new memory state is a linear function of the input and the old memory

The Basic Idea of QECC

Classical codes

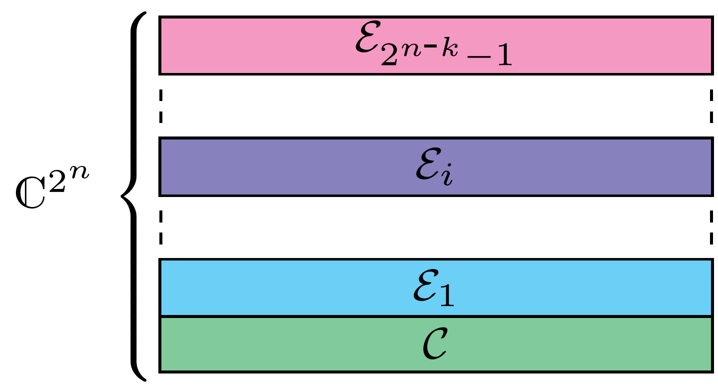
Partition of the set of all words of length n over an alphabet of size 2.



- codewords
- errors of bounded weight
- other errors

Quantum codes

Orthogonal decomposition of the vector space $\mathcal{H}^{\otimes n}$, where $\mathcal{H} \cong \mathbb{C}^2$.



$$\mathcal{H}^{\otimes n} = \mathcal{C} \oplus \mathcal{E}_1 \oplus \ldots \oplus \mathcal{E}_{2^{n-k}-1}$$

Encoding: $|\boldsymbol{x}\rangle \mapsto U_{\text{enc}}(|\boldsymbol{x}\rangle\,|0\rangle)$

Discretization of Quantum Errors

Consider errors $E = E_1 \otimes \ldots \otimes E_n$, $\qquad E_i \in \{I, X, Y, Z\}$.

"Pauli" matrices (real version):

$$I, \quad X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \quad Z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}, \quad Y = XZ = \begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix}$$

The weight of E is the number of $E_i \neq \mathbf{1}$. E. g., the weight of $I \otimes X \otimes Z \otimes Z \otimes I \otimes Y \otimes Z$ is 5.

Theorem: If a code $\mathcal{C}$ corrects all errors E of weight t or less, then $\mathcal{C}$ can correct arbitrary errors affecting $\leq t$ qubits.

Simple Quantum Error-Correcting Code

Repetition code: $|0\rangle \mapsto |000\rangle$, $|1\rangle \mapsto |111\rangle$

Encoding of one qubit:

$$\alpha\,|0\rangle + \beta\,|1\rangle \mapsto \alpha\,|000\rangle + \beta\,|111\rangle\,.$$

This defines a two-dimensional subspace $\mathcal{H}_\mathcal{C} \le \mathcal{H}_2 \otimes \mathcal{H}_2 \otimes \mathcal{H}_2$

bit-flip	quantum state	subspace
no error	$\alpha\,|000\rangle + \beta\,|111\rangle$	$(\mathbb{1} \otimes \mathbb{1} \otimes \mathbb{1})\mathcal{H}_\mathcal{C}$
1^{st} position	$\alpha\,|100\rangle + \beta\,|011\rangle$	$(X \otimes \mathbb{1} \otimes \mathbb{1})\mathcal{H}_\mathcal{C}$
2^{nd} position	$\alpha\,|010\rangle + \beta\,|101\rangle$	$(\mathbb{1} \otimes X \otimes \mathbb{1})\mathcal{H}_\mathcal{C}$
3^{rd} position	$\alpha\,|001\rangle + \beta\,|110\rangle$	$(\mathbb{1} \otimes \mathbb{1} \otimes X)\mathcal{H}_\mathcal{C}$

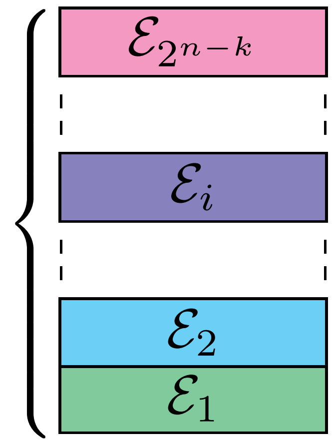
Hence we have an orthogonal decomposition of $\mathcal{H}_2 \otimes \mathcal{H}_2 \otimes \mathcal{H}_2$

Stabilizer Codes

Observables

$\mathcal{C}$ is a common eigenspace of the stabilizer group $\mathcal{S}$

decomposition into common eigenspaces



the orthogonal spaces are labeled by the eigenvalues
$\Longrightarrow$ operations that change the eigenvalues can be detected

The Stabilizer of a Quantum Code

Pauli group:

$$\mathcal{G}_n = \Big\{ \pm E_1 \otimes \ldots \otimes E_n : E_i \in \{I, X, Y, Z\} \Big\}$$

Let $\mathcal{C} \le \mathbb{C}^{2^n}$ be a quantum code.

The stabilizer of $\mathcal{C}$ is defined to be the set

$$\mathcal{S} = \Big\{ M \in \mathcal{G}_n : M \,|v\rangle = |v\rangle \text{ for all } |v\rangle \in \mathcal{C} \Big\}.$$

A quantum code $\mathcal{C}$ with stabilizer $\mathcal{S}$ is called a stabilizer code if and only if

$$M \,|v\rangle = |v\rangle \text{ for all } M \in \mathcal{S} \Longrightarrow |v\rangle \in \mathcal{C}.$$

$\mathcal{S}$ is an abelian (commutative) group and $\mathcal{C}$ is the joint $+1$-eigenspace of all $M \in \mathcal{S}$.

Stabilizer Codes and Classical Codes

Notation:

Denote by $X_{\boldsymbol{a}}$ where $\boldsymbol{a} = (a_1, \ldots, a_n) \in \mathbb{F}_2^n$ and $\mathbb{F}_2 = \{0, 1\}$ the operator

$$X_{\boldsymbol{a}} = X^{a_1} \otimes X^{a_2} \otimes \ldots \otimes X^{a_n}.$$

Similar

$$Z_{\boldsymbol{b}} = Z^{b_1} \otimes Z^{b_2} \otimes \ldots \otimes X^{b_n}.$$

For instance $X_{110} = X \otimes X \otimes I$ and $Z_{101} = Z \otimes I \otimes Z$.

Hence, any operator in $\mathcal{G}_n$ is of the form $\pm X_{\boldsymbol{a}} Z_{\boldsymbol{b}}$ for some $\boldsymbol{a}, \boldsymbol{b} \in \mathbb{F}_2^n$, written as $(\boldsymbol{a}|\boldsymbol{b})$.

Stabilizer Codes and Classical Codes

Example: The repetition code is a stabilizer code with stabilizer
$\mathcal{S} = \{III, ZZI, ZIZ, IZZ\} = \langle ZIZ, IZZ \rangle$.

$$\mathcal{S} \mathrel{\hat{=}} \left(\begin{array}{ccc|ccc} 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 \end{array} \right)$$

in general

$$\mathcal{S} \mathrel{\hat{=}} \big(\mathbf{X}|\mathbf{Z}\big) \in \mathbb{F}_2^{(n-k)\times 2n}$$

The matrix $(\mathbf{X}|\mathbf{Z})$ generates a symplectic self-orthogonal code with

$$\mathbf{X}\mathbf{Z}^t - \mathbf{Z}\mathbf{X}^t = \mathbf{0}.$$

Encoding Stabilizer Codes

Basic idea:

Use operations of the *generalized Clifford group* (or Jacobi group) to transform the stabilizer $\mathcal{S}$ into a trivial stabilizer $\mathcal{S}_0 := \left\langle Z^{(1)}, \ldots, Z^{(n-k)} \right\rangle$, corresponding to the code $\left|0^{n-k}\right\rangle |\phi\rangle$.

- row/column operations on the binary matrix $(\mathbf{X}|\mathbf{Z})$ to obtain "normal form" $(\mathbf{0}|\mathbf{I}\mathbf{0})$
- operations on $(\mathbf{X}|\mathbf{Z})$ correspond to
  - "elementary" single-qubit gates
  - CNOT-gate
  - single qubit gate $P := \begin{pmatrix} 1 & 0 \\ 0 & e^{\pi i/2} \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 0 & i \end{pmatrix} \in \mathbb{C}^{2\times 2}$

Action on Pauli Matrices

Hadamard matrix H	$HXH = Z$ $(1,0) \mapsto (0,1)$	$HYH = -Y$ $(1,1) \mapsto (1,1)$	$HZH = X$ $(0,1) \mapsto (1,0)$
	exchange X and Z		
matrix P	$P^\dagger XP = -Y$ $(1,0) \mapsto (1,1)$	$P^\dagger YP = X$ $(1,1) \mapsto (1,0)$	$P^\dagger ZP = Z$ $(0,1) \mapsto (0,1)$
	multiply X by Z		

in $\mathbb{C}$

$\bmod 2$

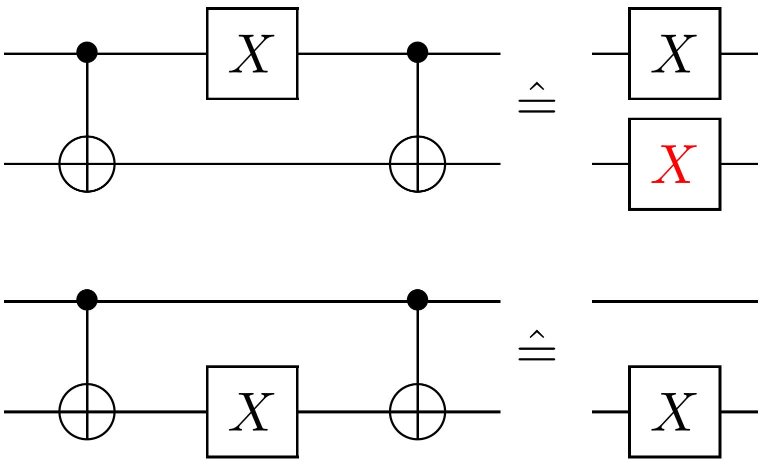
operation on binary row vectors: $(a,b)\overline{M} = (a', b')$ (arithmetic $\bmod 2$)

$$\overline{H} \mathrel{\hat{=}} \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \quad \text{and} \quad \overline{P} \mathrel{\hat{=}} \begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix}$$

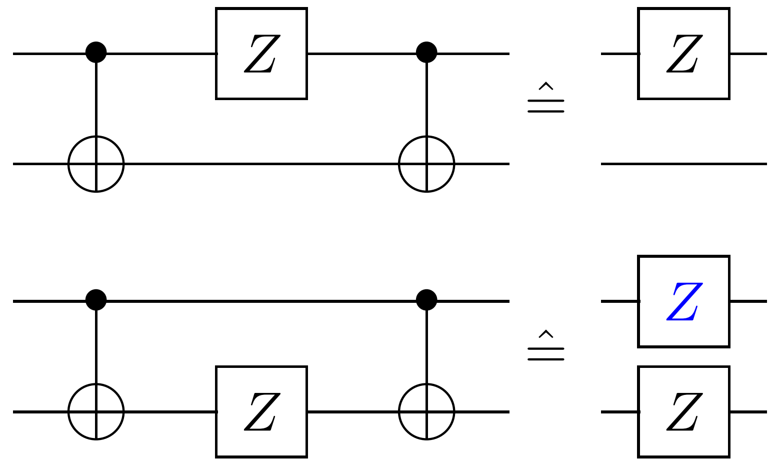
local operation on $(\mathbf{X}|\mathbf{Z})$:
multiplying column i in submatrix $\mathbf{X}$ and column i in submatrix $\mathbf{Z}$ by $\overline{M}$

Action of CNOT

Modifying stabilizers



add X from source to target



add Z from target to source

$$\mathrm{CNOT} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} \in \mathbb{C}^{4\times 4}$$

$$\overline{\mathrm{CNOT}} \mathrel{\hat=} \left(\begin{array}{cc|cc} 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ \hline 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 \end{array}\right) \in \mathbb{F}_2^{4\times 4}$$

Example: 5 Qubit Code $[\![5,1,3]\!]$

Generators of stabilizer

$$\begin{bmatrix} XXZIZ \\ ZXXZI \\ IZXXZ \\ ZIZXX \end{bmatrix} \mathrel{\hat{=}} \left(\begin{array}{ccccc|ccccc} 1 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 & 1 & 0 & 0 \end{array}\right)$$

Step I: X-Only Generator

Local operations $\hat{=}$ operations on corresponding $\mathbf{X}$ and $\mathbf{Z}$ columns

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

$$T_1 := I \otimes I \otimes I \otimes I \otimes I$$

Step I: X-Only Generator

Local operations $\hat{=}$ operations on corresponding $\mathbf{X}$ and $\mathbf{Z}$ columns

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

$$T_1 := I \otimes I \otimes H \otimes I \otimes I$$

Step I: X-Only Generator

Local operations $\hat{=}$ operations on corresponding $\mathbf{X}$ and $\mathbf{Z}$ columns

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

$$T_1 := I \otimes I \otimes H \otimes I \otimes H$$

Step II: X-Generator of Weight One

CNOT $\hat{=}$ operations on pairs of corresponding $\mathbf{X}$ and $\mathbf{Z}$ columns

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

$$T_1 := I \otimes I \otimes H \otimes I \otimes H$$

$$T_2 := I$$

Step II: X-Generator of Weight One

$\mathrm{CNOT} \mathrel{\hat{=}}$ operations on pairs of corresponding $\mathbf{X}$ and $\mathbf{Z}$ columns

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

$$T_1 := I \otimes I \otimes H \otimes I \otimes H$$

$$T_2 := I$$

Step II: X-Generator of Weight One

$\mathrm{CNOT} \hat{=}$ operations on pairs of corresponding $\mathbf{X}$ and $\mathbf{Z}$ columns

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

$$T_1 := I \otimes I \otimes H \otimes I \otimes H$$

$$T_2 := \mathrm{CNOT}^{(1,2)}$$

Step II: X-Generator of Weight One

$\mathrm{CNOT} \hat{=}$ operations on pairs of corresponding $\mathbf{X}$ and $\mathbf{Z}$ columns

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

$$T_1 := I \otimes I \otimes H \otimes I \otimes H$$

$$T_2 := \mathrm{CNOT}^{(1,2)}$$

Step III: Row Operations

multiplying generators $\hat{=}$ adding/permuting rows

$$\left(\begin{array}{ccccc|ccccc} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \hline 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 1 \end{array}\right)$$

$$T_1 := I \otimes I \otimes H \otimes I \otimes H$$

$$T_2 := \mathrm{CNOT}^{(1,2)}\mathrm{CNOT}^{(1,3)}\mathrm{CNOT}^{(1,5)}$$

Pre-Final Result: Only X-Generators

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

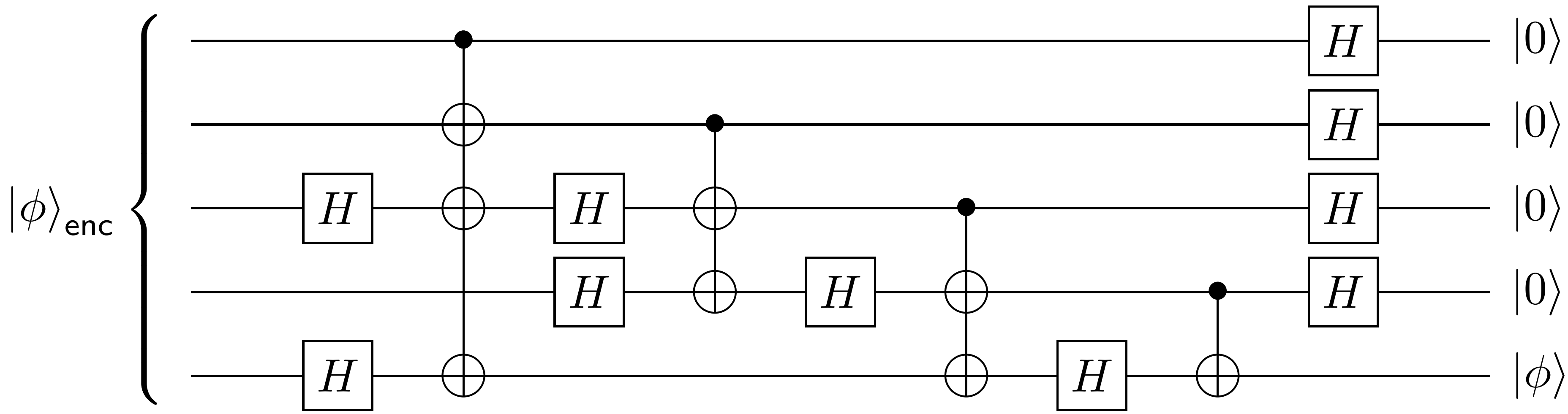
Final Result: Only Z-Generators

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

$$T_{\text{final}} := H \otimes H \otimes H \otimes H \otimes I$$

combining all transformations:
quantum circuit that maps the encoded state $|\phi\rangle_{\text{enc}}$
to the un-encoded state $|0000\rangle\,|\phi\rangle$

Inverse Encoding Circuit



Quantum circuit mapping a state $|\phi\rangle_{\text{enc}}$ of the code $[\![5,1,3]\!]$ to an unencoded one-qubit state $|\phi\rangle$.

Quantum Convolutional Codes

Quantum Block Codes

The code is the common eigenspace of the stabilizers.

Quantum Convolutional Codes

Idea: impose local constraints by stabilizers

Example:

$$s_1 = \ldots\ III\ XXX\ XZY\ III\ III\ \ldots$$
$$s_2 = \ldots\ III\ ZZZ\ ZYX\ III\ III\ \ldots$$

shift the stabilizers by three qubits:

$$s_1' = \ldots\ III\ III\ XXX\ XZY\ III\ \ldots$$
$$s_2' = \ldots\ III\ III\ ZZZ\ ZYX\ III\ \ldots$$

Semi-infinite Stabilizer

Compact representation of the semi-infinite stabilizer matrix

$$\begin{pmatrix} XXX & XZY & & \\ ZZZ & ZYX & & \\ & XXX & XZY & \\ & ZZZ & ZYX & \\ & & & \ddots \end{pmatrix}$$

$$\hat{=} \left(\begin{array}{cccc|cccc} 111 & 101 & & & 000 & 011 & & \\ 000 & 011 & & & 111 & 110 & & \\ & 111 & 101 & & & 000 & 011 & \\ & 000 & 011 & & & 111 & 110 & \\ & & & \ddots & & & & \ddots \end{array}\right)$$

$$\hat{=} \left(\begin{array}{ccc|ccc} 1+D & 1 & 1+D & 0 & D & D \\ 0 & D & D & 1+D & 1+D & 1 \end{array}\right) = \mathbf{S}(D)$$

Quantum Convolutional Codes

Quantum Block Codes

The stabilizer $\mathcal{S}$ corresponds to a binary code generated by the stabilizer matrix $(\mathbf{X}|\mathbf{Z})$.

Quantum Convolutional Codes

The semi-infinite stabilizer corresponds to a binary convolutional code generated by the matrix $\big(\mathbf{X}(D) \bigm| \mathbf{Z}(D)\big)$ with

$$\mathbf{X}(D)\mathbf{Z}(1/D)^t - \mathbf{Z}(D)\mathbf{X}(1/D)^t = \mathbf{0}$$

Example:

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

Catastrophic (Quantum) Convolutional Codes

Bad example:

$$\begin{pmatrix} Z & Z & & & \\ & Z & Z & & \\ & & Z & Z & \\ & & & & \ddots \end{pmatrix} \mathrel{\hat{=}} \big(0 \,|1 + D\big) = \mathbf{S}(D)$$

Quantum code with basis states $|\underline{0}\rangle = |000\ldots\rangle$ and $|\underline{1}\rangle = |111\ldots\rangle$, contains in particular "infinite cat state"

$\Longrightarrow$ local errors spread unboundedly

$\Longrightarrow$ further constraints on $\mathbf{S}(D)$ (must have polynomial inverse)

Quantum Convolutional Codes: Error Correction

Basic Ideas:

- Every stabilizer has bounded support.
- Measure the eigenvalue of the stabilizer when all correspondings qubits have been received.
  $\Longrightarrow$ syndrome of the corresponding classical convolutional code
- Use your favorite algorithm to decode the classical convolutional code (e. g. Viterbi algorithm).

Operations on $\mathbf{S}(D) = (\mathbf{X}(D)|\mathbf{Z}(D))$

$$H = \frac{1}{\sqrt{2}}\begin{pmatrix}1 & 1\\ 1 & -1\end{pmatrix} \in \mathbb{C}^{2\times 2} \qquad \overline{H} = \begin{pmatrix}0 & 1\\ 1 & 0\end{pmatrix} \in \mathbb{F}_2^{2\times 2}$$

$$P = \begin{pmatrix}1 & 0\\ 0 & \exp(i\pi/2)\end{pmatrix} \in \mathbb{C}^{2\times 2} \qquad \overline{P} = \begin{pmatrix}1 & 1\\ 0 & 1\end{pmatrix} \in \mathbb{F}_2^{2\times 2}$$

$$\mathrm{CNOT}^{(i,j+\ell n)}, i \not\equiv j \pmod{n} \qquad \overline{\mathrm{CNOT}} = \left(\begin{array}{cc|cc}1 & D^\ell & 0 & 0\\ 0 & 1 & 0 & 0\\ \hline 0 & 0 & 1 & 0\\ 0 & 0 & D^{-\ell} & 1\end{array}\right)$$

$$P_\ell := \mathrm{CSIGN}^{(i,i+\ell n)}, \ell \neq 0 \qquad \overline{P_\ell} = \begin{pmatrix}1 & D^{-\ell} + D^\ell\\ 0 & 1\end{pmatrix}$$

Quantum Circuits

Single qubit gates

operation on stabilizer matrix in D-transform notation

$\Longrightarrow$ expand to semi-infinite matrix

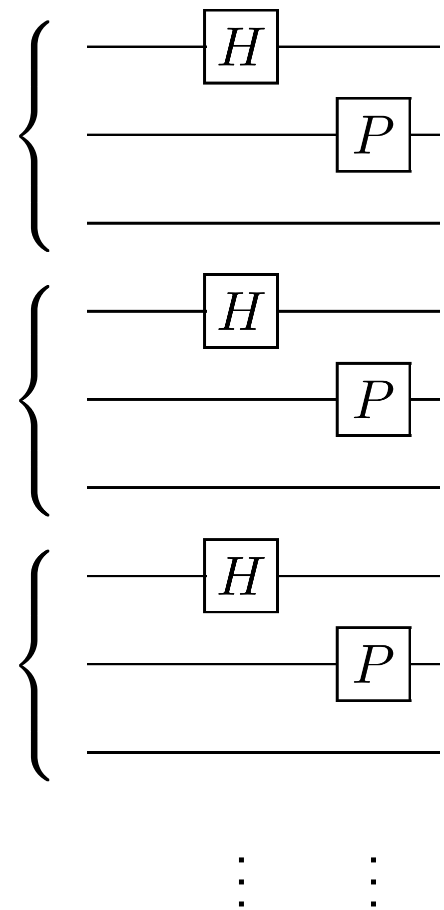
$\Longrightarrow$ repeat the operations infinitely often

Example:

blocks of three qubits each

operation H on first position

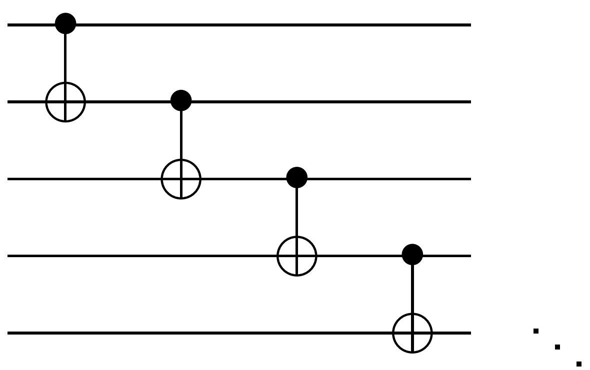
operation P on second position



Quantum Circuits

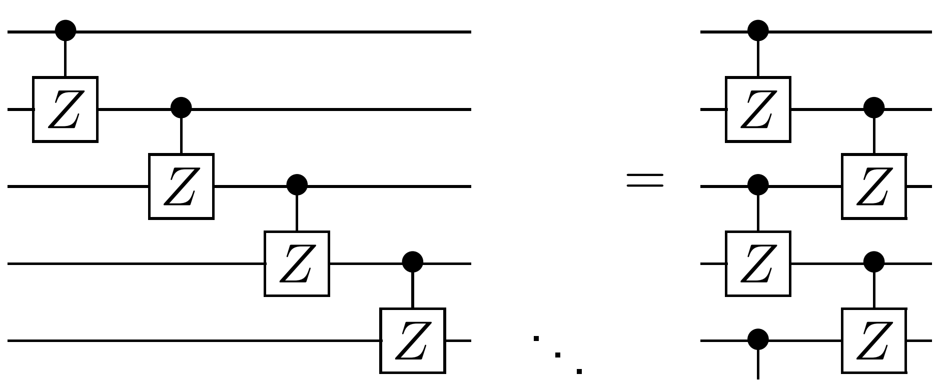
Two-qubit gates

- CNOT on qubit j in block ℓ and qubit j in block $\ell+1$:



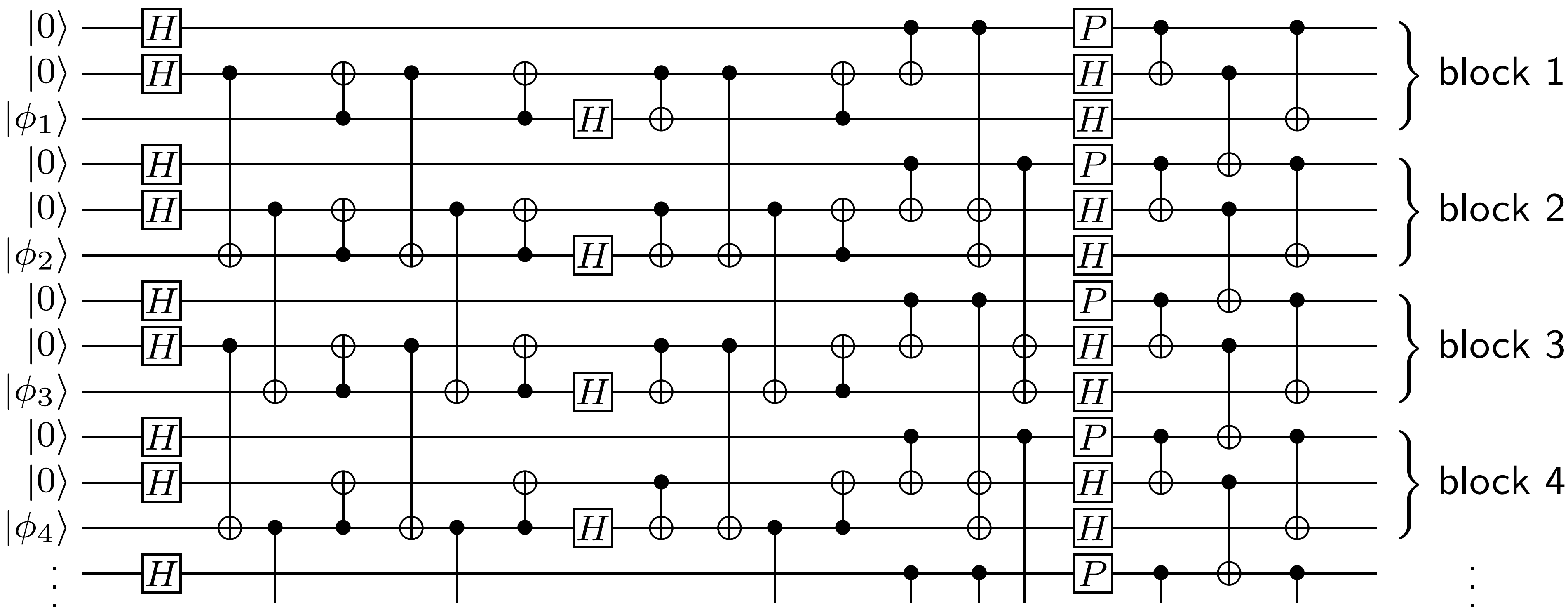
$\Longrightarrow$ infinite depth

- CSIGN on qubit j in block ℓ and qubit j in block: $\ell+1$:



$\Longrightarrow$ finite depth

Example: Rate $1/3$ Quantum Convolutional Code



Every gate has to be repeatedly applied shifted by one block.

Summary

- Quantum stabilizer codes are common eigenspaces of the stabilizer
- The stabilizer corresponds to a classical code
- Use both block and convolutional codes to define stabilizer codes

- Encoding circuits from transformations on the binary matrix
- Quantum circuits with finite depth for convolutional quantum codes with non-catastrophic generator matrix
- Two-qubit gates span a bounded number of blocks.
- Generators of stabilizer act non-trivially on a bounded number of qubits.

References

- M. Grassl, M. Rötteler, and Th. Beth, "Efficient Quantum Circuits for Non-Qubit Quantum Error-Correcting Codes", *International Journal of Foundations of Computer Science*, 14(5):757–775, 2003, quant-ph/0211014.
- H. Ollivier and J.-P. Tillich, "Quantum convolutional codes: fundamentals," 2004, quant-ph/0401134.
- G. D. Forney, Jr., M. Grassl, and S. Guha, "Convolutional and tail-biting quantum error-correcting codes," *IEEE Trans. Inform. Theory*, vol. 53, no. 3, pp. 865–880, 2007, quant-ph/0511016.
- M. Grassl and M. Rötteler, "Non-catastrophic encoders and encoder inverses for quantum convolutional codes," in *Proc. 2006 IEEE Intl. Symp. Inform. Theory*, Seattle, USA, 2006, pp. 1109–1113, quant-ph/0602129.
- S. A. Aly, M. Grassl, A. Klappenecker, M. Rötteler, P. K. Sarvepalli, "Quantum Convolutional BCH Codes," *2007 Canadian Information Theory Workshop*, quant-ph/0703113.