Bachelor Thesis
Decoding the Color Code

Clemens Schumann,
Advised by Peter-Jan Derks

December 6, 2022

Contents

1 Introduction to the Algebra **3**
1.1 Schroedinger picture 3
1.2 Heisenberg Picture and stabilizer Formalism 4
1.2.1 Stabilizer Group 4
1.2.2 Effect of Gates on stabilizers 4
1.2.3 Effect of measurements on stabilizers 5
1.2.4 Circuit Analysis in Stabilizer formalism 5
1.3 The Clifford Gates 7
1.4 Error Detection and Correction 8
1.4.1 Repetition Code 8
1.4.2 2D Codes 9

2 Conclusion **10**

3 Appendix **11**
3.1 Schroedinger picture calculation of CNOT circuit 11

1 Introduction to the Algebra

In the following we will give an overview of the utilized algebra in quantum information theory.
To this end we must first return to the fundamentals of quantum mechanics.

1.1 Schroedinger picture

In the Schroedinger picture, we focus on the time evolution of states:

$$|\psi\rangle = |\psi\rangle(t) \tag{1}$$

In this picture we can introduce quantum circuit diagram notation, whereby:

- States progress in time along horizontal parallel lines
- Time goes from left to right
- Gates denoted X,Y,Z are the single qubit pauli operators $\sigma_x, \sigma_y, \sigma_z$
- Gates can act on one or multiple qubits, whereby an X gate on qubit 1 in a 3-qubit system should be interpreted as:
  $(X \otimes \mathbb{I} \otimes \mathbb{I})(|\psi_1\rangle \otimes |\psi_2\rangle \otimes |\psi_3\rangle)$

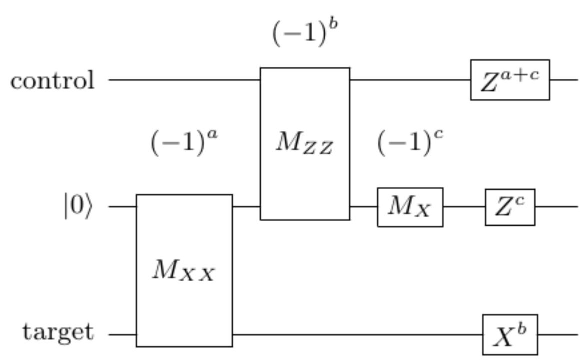

Figure 1: A Quantum Circuit to implement a measurement based Controlled-$X_{|\psi\rangle_{control} \rightarrow |\psi\rangle_{target}}$ Gate, where $|0\rangle$ is the +1 eigenstate in σ_z-basis.

As can be seen explicitly calculated in the familiar Schroedinger picture in Appendix 3.1, the circuit from figure 1 implements a CNOT-gate from the control qubit to the target qubit.

We will now analyze this circuit in the Heisenberg picture, finding that it results in an equal output.

1.2 Heisenberg Picture and stabilizer Formalism

1.2.1 Stabilizer Group

We call an operator/gate S, to which the input state is an eigenvector ($S|\psi\rangle = |\psi\rangle$), a "stabilizer" of that input state.
The "stabilizer group" is a generating subset of the set of such operators.

We write these stabilizers as tensor-products of pauli operators $P \in P_G$, where pauli operators are the operators on $\mathbb{F}_2$ such that:

$\forall P \in P_G, n \in \mathbb{N} : P^{2n} = \mathbb{I}$.

In the Heisenberg picture, stabilizers are tracked instead of states. The stabilizer group S_G is the group generated by the set of stabilizers:

$$S_G = \langle S_0, .., S_n \rangle : S|\psi_{in}\rangle = |\psi_{in}\rangle \forall S \in S_G \tag{2}$$

So for the example in figure 1 it is the group of operators to whom $|\psi_{control}\rangle \otimes |0\rangle \otimes |\psi_{target}\rangle$ is an eigenstate, namely $\mathbb{I} \otimes Z \otimes \mathbb{I}$ (and trivially $\mathbb{I} \otimes \mathbb{I} \otimes \mathbb{I}$, which we choose to ignore as a stabilizer since any three-qubit state is stabilized by it, and it can be generated by squaring any pauli-constructed stabilizer).

The stabilizer group is always an abelian group and its elements commute, since if:

$$\forall A, B \in S : AB|\psi\rangle = BA|\psi\rangle = |\psi\rangle \Rightarrow [A, B]|\psi\rangle = 0 \tag{3}$$

1.2.2 Effect of Gates on stabilizers

In order to determine the effect a Gate operation A has on a stabilizer, consider the following:

If $S|\psi\rangle = |\psi\rangle$ then:

$$A|\psi\rangle = AS|\psi\rangle = AS\mathbb{I}|\psi\rangle = \underbrace{ASA^\dagger}_{=S'} A|\psi\rangle \tag{4}$$

So we now know that the post-gate state is an Eigenstate of S'.

Therefore $S'_G = \langle AS_0A^\dagger, ..., AS_nA^\dagger \rangle$.

1.2.3 Effect of measurements on stabilizers

A pauli measurement operator M can either commute with all stabilizer operators, in which case M itself is a stabilizer already and the measurement has no effect on the state, or anticommute with at least one operator in S_G, since pauli operators as well as their tensor-products can only commute or anti-commute with each other.

In that case, to obtain the new stabilizers S'_G, proceed as follows:

- Identify $S \in S_G : S, M = 0$
- Remove S from S_G
- Add M to S_G
- for $N \in S_G \cup \overline{X} \cup \overline{Z}$: if $\{N, M\} == 0 : N = SN$

where $\overline{X}$ and $\overline{Z}$ are the sets of logical Xs and Zs respectively.

1.2.4 Circuit Analysis in Stabilizer formalism

After a measurement M, an n qubit input state will always collapse into either the +1 or the -1 eigenstate of the measurement operator.

In the first case the acting measurement operator becomes $\mathbb{I}^{\otimes n} + M$, in the second it becomes $\mathbb{I}^{\otimes n} - M$. Therefore, in the circuit shown in figure 1, the measurements are:

$$P_1^\pm = \frac{1}{2}\left(\mathbb{I}^{\otimes 3} \pm \mathbb{I} \otimes X \otimes X\right) \tag{5}$$

$$P_2^\pm = \frac{1}{2}\left(\mathbb{I}^{\otimes 3} \pm X \otimes X \otimes \mathbb{I}\right) \tag{6}$$

$$P_3^\pm = \frac{1}{2}\left(\mathbb{I}^{\otimes 3} \pm \mathbb{I} \otimes X \otimes \mathbb{I}\right) \tag{7}$$

In the following stabilizers will be written without the tensorproduct symbols, so in our case the stabilizer is initially: $S_G^0 = \langle IZI \rangle$, the logical $\overline{X}$ operator is XXX and the logical $\overline{Z}$ operator is ZIZ.

After the first Measurement, the state is stabilized by IXX, since it collapses into an eigenstate of the measurement operator. Notably, if the measurement operator M anticommutes with some element of the stabilizer S:

$$SP_{-}S^{\dagger} = \frac{1}{2}S\left(\mathbb{I}^{\otimes 3} - M\right)S^{\dagger} = \frac{1}{2}\left(\mathbb{I}^{\otimes 3} + M\right)SS^{\dagger} = P_{+} \tag{8}$$

So by applying an anticommuting previous stabilizer operator after the measurement one can ensure that the state is in the P_+ projected state $P_+|\psi_{init}\rangle$ (in short, +1 and -1 eigenstates have the same stabilizers if we add conditional gates accordingly).

For the logical operators, if $\overline{X}$ or $\overline{Z}$ do not commute with the measurement operator, we know that their product with another anticommuting operator from the previous stabilizer must then commute with the measurement operator: $[S_{prev}\overline{X}M, MS_{prev}\overline{X}] = 0$ (recall the previous statement that all paulis and their tensorproducts must either commute or anti-commute).

In our case, IZI and IXX anticommute, so now the state is stabilized by $S_G^1 = \langle IXX\rangle$. Both initial logical operators commute with the first measurement operator, so are left unchanged.

After the second measurement M_2=ZZI, since this measurement anticommutes with the IXX stabilizer, the new stabilizers are: $S_G^2 = \langle ZZI\rangle$. The logical $\overline{X}$ and $\overline{Z}$ operators are unaffected , since they commute with the measurement operator.

After the third measurement M_3 =IXI, since this measurement anticommutes with the stabilizer, the new stabilizers are: $S_G^3 = \langle IXI\rangle$. The logical $\overline{Z}$ operator anticommutes with the measurement, so is replaced by $\overline{Z_3}$=ZZI · ZIZ = IIZ. The logical $\overline{X}$ is unaffected, since it commutes with the measurement operator.

The stabilizer for the control and target qubit are still identity, and logical $\overline{Z} : ZIZ \to IIZ$.

Since CNOT takes: $Z \otimes Z \to I \otimes Z$, this implements a logical CNOT.

1.3 The Clifford Gates

It has been proven in *reference source* that operators that take a state stabilized by some member of the Pauli-Group to a state stabilized by another member of the Pauli-Group can be simulated efficiently on a classical computer. The Group of operators that satisfy this condition is called the Clifford-Group.

For the decoder we wish to implement in this thesis it therefore makes sense to focus on those first and foremost, as applying corrective gates is a computationally/ experimentally expensive task that should be put off to the latest possible moment, and the propagation of an error until then can be simulated efficiently.

The Clifford-Group can be generated by:

- The Hadamard-Gate, which performs single qubit basis changes from eigenstates of X to eigenstates of Z and vice-versa:

  $H|+\rangle = |0\rangle$, $H|0\rangle = |+\rangle$, $H|-\rangle = |1\rangle$, $H|1\rangle = |-\rangle$

- The Phase-Gate, which performs single qubit sign flips on the state parts which are $|1\rangle$ in the computational basis:

  $P(\alpha|0\rangle \pm \beta|1\rangle) = \alpha|0\rangle \mp \beta|1\rangle$

- The CNOT-Gate, which on a two qubit system performs an X gate on the second qubit if the first qubit is $|1\rangle$, so takes:

  $\alpha|00\rangle + \beta|01\rangle + \delta|10\rangle + \gamma|11\rangle$
  $\Rightarrow \alpha|00\rangle + \beta|01\rangle + \gamma|10\rangle + \delta|11\rangle$

In the σ_z-basis their matrix representations are:

- $H = \frac{1}{\sqrt{2}} \cdot \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$; $P = \begin{pmatrix} 1 & 0 \\ 0 & i \end{pmatrix}$

- $CNOT = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}$

1.4 Error Detection and Correction

One of the big challenges of physically realising a quantum computer is its subjection to noise in the real world. Unlike classical computers, the type of error is not limited to a bitflip, as even single qubit states have a theoretically infinite amount of differing states to it on a bloch sphere, and therefore an infinite amount of types of errors can have occured in the presence of noise such as thermal or electromagnetical noise.

Fortunately, this noise can be modeled as successive pauli gates. Since an identity noise occuring is irrelevant to us, and XY as well as ZY (anti-) commute, we need only correct for X and Z errors occuring in order to correct any pauli errors.

1.4.1 Repetition Code

In order to correct errors, they must first be detected. From classical computer science there are well known existing codes, such as the repetition code. For this error code information is encoded by repeating the intended message some amount of times, and then decoding it by performing a majority vote on the transmitted message. In Quantum error correction, we speak of [[n,k,d]] stabilizer codes if an encoding scheme allows for n physical qubits to encode k logical qubits to a distance d.

A quantum equivalent of the 3-bit repetition code performed on the message $|1\rangle$ is is the $[[3,1,\frac{1}{2}]]$ repetition code depicted in figure 2, including so-called "Syndrome Extraction". A Syndrome is a stabilizer that can be measured in order to detect wether and where an error has occured in a multi-qubit system. It is crucial that the measurement of such Syndromes occurs without harming the actual quantum information stored in the "data-qubits". Therefore two additional "Ancilla-qubits" (both initialized to $|0\rangle$) are attached to the circuit via CNOTs. This circuit is stabilized by IZZ and ZZI, measured by ancilla 1/2. The measurement result will therefore be a vector of length two, with each entry either being +1 or -1. To simplify the algebra this will be changed to the binary representation of 0 for +1 and 1 for -1.

To represent the code, Stabilizers can be stacked together to a so-called parity-check-matrix, which satisfies:

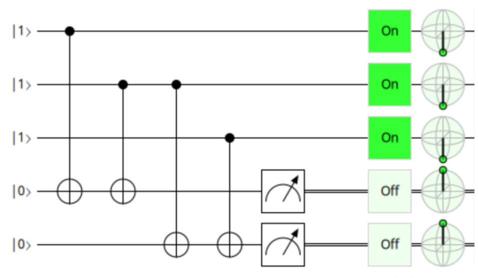

Figure 2: Bitflip Syndrome extractor
+1 measurement result on first Ancilla indicates a bitflip error on qubits 1 or 2, +1 result on second ancilla indicates bitflip on second or third qubit

This way of encoding information however leaves two notable issues:

For one, it only detects bitflip, or pauli-X errors occuring on the stored quantum information. While using Hadamard gates one could trivially adapt this code to instead detect pauli-Z errors, it is not possible to use linear codes like the repetition code to *simultaneously* detect pauli-X and pauli-Z errors occuring.

Secondly, it also assumes a noise model of a "Noisy Channel", which is not compatible with the actually encountered errors in real physical quantum computers.

1.4.2 2D Codes

Previous research in computer science provides a toolset for generating valid codes from existing encoding schemes. Hypergraph product codes, introduced by Tillich and Zémor, of two existing codes will always remain a valid detection code. We can therefore form a hypergraph product code of two repetition codes for X error detection and Z error detection respectively, obtaining the [[d^2,1,d]] "Surface-Code" which can detect up to d of *both* X and Z errors, and therefore any pauli error happening.

The parity check matrix of a hypergraph product code is generated by the parity check matrices of two valid codes in the following way:

PUT IN A NICE DRAWING OF THE SURFACE CODE.

2 Conclusion

3 Appendix

3.1 Schroedinger picture calculation of CNOT circuit

In the quantum circuit depicted in figure 1 the input state can be written as $|\psi_{control}\rangle \otimes |0\rangle \otimes |\psi_{target}\rangle$ and the measurement in the first timestep can be expressed as $\mathbb{I} \otimes X \otimes X$.

The initial state $|\phi_{t=0}\rangle = |\psi_{control}\rangle \otimes |\psi_{ancilla}\rangle \otimes |\psi_{target}\rangle$
where
$|\psi_{control}\rangle = \alpha|0\rangle + \beta|1\rangle$
$|\psi_{ancilla}\rangle = |0\rangle$
$|\psi_{target}\rangle = \gamma|0\rangle + \delta|1\rangle$
therefore:

$$|\phi_{t=0}\rangle = \alpha\left(\gamma|000\rangle + \delta|001\rangle\right) + \beta\left(\gamma|100\rangle + \delta|101\rangle\right) \tag{9}$$

If the first measurement result is +1, the state becomes:

$$\begin{aligned}|\phi^{+}_{t=1}\rangle &= \frac{1}{2}\left(\mathbb{I}\otimes\mathbb{I}\otimes\mathbb{I} + \mathbb{I}\otimes X\otimes X\right)|\phi_{t=0}\rangle\\ &= \alpha\left(\gamma\left(|000\rangle + |011\rangle\right) + \delta\left(|001\rangle + |010\rangle\right)\right)\\ &+ \beta\left(\gamma\left(|100\rangle + |111\rangle\right) + \delta\left(|101\rangle + |110\rangle\right)\right)\end{aligned}$$

if the result is -1, it becomes:

$$\begin{aligned}|\phi^{-}_{t=1}\rangle &= \frac{1}{2}\left(\mathbb{I}\otimes\mathbb{I}\otimes\mathbb{I} - \mathbb{I}\otimes X\otimes X\right)|\phi_{t=0}\rangle\\ &= \alpha\left(\gamma\left(|000\rangle - |011\rangle\right) + \delta\left(|001\rangle - |010\rangle\right)\right)\\ &+ \beta\left(\gamma\left(|100\rangle - |111\rangle\right) + \delta\left(|101\rangle - |110\rangle\right)\right)\end{aligned}$$

In the case of the +1 Measurement $\rightarrow$ a=0:

$$\begin{aligned}|\phi^{++}_{t=2}\rangle &= \frac{1}{2}\left(\mathbb{I}\otimes\mathbb{I}\otimes\mathbb{I} + Z\otimes Z\otimes\mathbb{I}\right)|\phi^{+}_{t=1}\rangle\\ &= (|000\rangle\langle 000| + |001\rangle\langle 001| + |110\rangle\langle 110| + |111\rangle\langle 111|)|\phi^{+}_{t=1}\rangle\\ &= \alpha\left(\gamma|000\rangle + \delta|001\rangle\right) + \beta\left(\gamma|111\rangle + \delta|110\rangle\right)\end{aligned}$$

$$\begin{aligned}|\phi^{+-}_{t=2}\rangle &= \frac{1}{2}\left(\mathbb{I}\otimes\mathbb{I}\otimes\mathbb{I} - Z\otimes Z\otimes\mathbb{I}\right)|\phi^{+}_{t=1}\rangle\\ &= (|010\rangle\langle 010| + |011\rangle\langle 011| + |100\rangle\langle 100| + |101\rangle\langle 101|)|\phi^{+}_{t=1}\rangle\\ &= \alpha\left(\gamma|011\rangle + \delta|010\rangle\right) + \beta\left(\gamma|100\rangle + \delta|101\rangle\right)\end{aligned}$$

In the case of the -1 Measurement $\rightarrow$ a=1:

$$\begin{aligned}|\phi_{t=2}^{-+}\rangle &= \frac{1}{2}\left(\mathbb{I}\otimes\mathbb{I}\otimes\mathbb{I} + Z\otimes Z\otimes\mathbb{I}\right)|\phi_{t=1}^{-}\rangle \\ &= \alpha\left(\gamma|000\rangle + \delta|001\rangle\right) - \beta\left(\gamma|111\rangle + \delta|110\rangle\right)\end{aligned}$$

$$\begin{aligned}|\phi_{t=2}^{--}\rangle &= \frac{1}{2}\left(\mathbb{I}\otimes\mathbb{I}\otimes\mathbb{I} - Z\otimes Z\otimes\mathbb{I}\right)|\phi_{t=1}^{-}\rangle \\ &= -\alpha\left(\gamma|011\rangle + \delta|010\rangle\right) + \beta\left(\gamma|100\rangle + \delta|101\rangle\right)\end{aligned}$$

Now the applied measurement is $\mathbb{I}\otimes X\otimes\mathbb{I}$, which means:

$$\begin{aligned}|\phi_{t=3}^{+++}\rangle &= \frac{1}{2}\left(\mathbb{I}\otimes\mathbb{I}\otimes\mathbb{I} + \mathbb{I}\otimes X\otimes\mathbb{I}\right)|\phi_{t=2}^{++}\rangle \\ &= \frac{1}{2}((|010\rangle + |000\rangle)\langle 000| + (|011\rangle + |001\rangle)\langle 001| \\ &+ (|000\rangle + |010\rangle)\langle 010| + (|001\rangle + |011\rangle)\langle 011| \\ &+ (|110\rangle + |100\rangle)\langle 100| + (|111\rangle + |101\rangle)\langle 101| \\ &+ (|100\rangle + |110\rangle)\langle 110| + (|101\rangle + |111\rangle)\langle 111|)|\phi_{t=2}^{++}\rangle \\ &= \frac{1}{2}(\alpha\left(\gamma(|000\rangle + |010\rangle) + \delta(|011\rangle + |001\rangle)\right) \\ &+ \beta\left(\gamma(|101\rangle + |111\rangle) + \delta(|100\rangle + |110\rangle)\right))\end{aligned}$$

In this case, a,b and c would each be zero, therefore no further gate would be applied.
As intended, this state is equivalent to $CNOT_{|\psi_{Control}\rangle\rightarrow|\psi_{Target}\rangle}|\phi_{t=0}\rangle$.
If the last measurement result is -1:

$$\begin{aligned}|\phi_{t=3}^{++-}\rangle &= \frac{1}{2}\left(\mathbb{I}\otimes\mathbb{I}\otimes\mathbb{I} - \mathbb{I}\otimes X\otimes\mathbb{I}\right)|\phi_{t=2}^{++}\rangle \\ &= \frac{1}{2}((|010\rangle + |000\rangle)\langle 000| + (|001\rangle - |011\rangle)\langle 001| \\ &+ (|010\rangle - |000\rangle)\langle 010| + (|011\rangle - |001\rangle)\langle 011| \\ &+ (|100\rangle - |110\rangle)\langle 100| + (|101\rangle - |111\rangle)\langle 101| \\ &+ (|110\rangle - |100\rangle)\langle 110| + (|111\rangle - |101\rangle)\langle 111|)|\phi_{t=2}^{++}\rangle \\ &= \frac{1}{2}(\alpha\,(TODOTODOTODOTODOTODO\gamma(|000\rangle + |010\rangle) + \delta(|011\rangle + |001\rangle)) \\ &+ \beta\left(\gamma(|101\rangle + |111\rangle) + \delta(|100\rangle + |110\rangle)\right))\end{aligned}$$

Notably, each measurement sequence has a differing resulting ancilla state, however we do not care since ancillas are meant to be discarded.
For now, the other 7 final computation steps are left as an exercise to the reader, however I probably will still finish that.