Quantum Codes on Hurwitz Surfaces

by

Isaac H. Kim

Submitted to the Department of Physics
in partial fulfillment of the requirements for the degree of

Bachelor of Science in Physics

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

May 2007
[June 2007]

Author .....
Department of Physics
May 4th, 2007

Certified by .....
Peter Shor
Morss Professor of Applied Mathematics
Thesis Supervisor

Accepted by .....
Professor David Pritchard
Senior Thesis Coordinator, Department of Physics

Quantum Codes on Hurwitz Surfaces

by

Isaac H. Kim

Submitted to the Department of Physics
on May 4th, 2007, in partial fulfillment of the
requirements for the degree of
Bachelor of Science in Physics

Abstract

Ever since the birth of the first quantum error correcting code , many error correcting techniques and formalism has been constructed so far. Among those, generating a quantum code on a locally planar geometry have lead to some interesting classes of codes. Main idea of this thesis stems from Kitaev's Toric code, which was the first surface code, yet it suffered from having a asymptotically vanishing encoding rate. In this paper, we propose a quantum surface code on a more complicated closed surface which has large genus, namely the Hurwitz surface. This code admits a constant encoding rate in the asymptotic limit that the number of genus goes to infinity. However, we give evidence that $\frac{t}{n}$, where n is the number of qubits and t is the number of correctible errors, converges to 0 asymptotically. This is based on numerically generating many Hurwitz surfaces and observing the corresponding quantum code in the limit that genus number goes to infinity.

Thesis Supervisor: Peter Shor
Title: Morss Professor of Applied Mathematics

Acknowledgments

My undergraduate years would not have been as fruitful as it was without professor Shor. I personally owe so much to him, ranging from the quantum information science class to my undergraduate research experience. I enjoyed them very much, and it was one of the most rewarding experience to work on theoretical research in this field. Likewise, I would not have been able to write up this thesis without his help. For that matter, I would like to express my deepest thanks to him, for his kind and helpful guidance of my undergraduate years which lead me to the realm of quantum information science. I would also like to thank my academic advisor, professor Chuang as well. His comments on course choices, research experiences, and school life in general were a cornerstone of my undergraduate life. It was also through him in which I first encountered quantum information science. He was my best mentor throughout my undergraduate years. I would also like to thank my family, who have always given me a constant support.

Contents

1 Introduction **9**

2 Preliminary Results **11**

2.1 Qubit 11

2.2 Density Operator 12

2.3 Quantum Operation 13

2.3.1 Operator-Sum Representation 14

2.3.2 Canonical OSR 15

2.4 Quantum Channel for Many-Qubit System 17

3 Quantum Code **19**

3.1 Shor 9-Qubit Code 21

3.2 Stabilizer Code 22

3.3 Toric Code 23

4 Quantum Code on a Hurwitz Surface **25**

4.1 Riemann's Uniformization Theorem and Hyperbolic Geometry 25

4.1.1 Uniformization Theorem 25

4.1.2 Hyperbolic Geometry 26

4.2 Tiling Group 27

4.3 Hurwitz Group 28

4.4 Quantum Code on a Hurwitz Surface 30

4.4.1 Generating Hurwitz Group 31

4.4.2 Computing the Length of the Systole 32
4.4.3 Relation between Systole Length and Correctible Errors . . . 33
4.4.4 Numerical Results 35

5 Conclusion **39**

Chapter 1

Introduction

One of the pioneers of quantum computing was Richard Feynmann. In the early 1980s, he realized that with digital computer, it would be computationally hard to simulate a quantum system efficiently, since it seems that we need to process $2^n - 1$ complex numbers to specify n spin-$\frac{1}{2}$ particles system. Even with only 100 particles, 10^{30} number of complex amplitudes change in time and it will take more than the age of the universe to compute all these numbers. However, instead of viewing this as a negative result, he proposed that maybe we will be able to perform some computational tasks much faster than classical computation by exploiting the laws of quantum mechanics.[1] In 1984, David Deutsch proposed a model of universal quantum Turing machine, which, in a naïve sense, is a generalization of Universal Turing Machine(UTM) that allows a superposition and interference of states.[2] Following this was the birth of the first quantum algorithms, the Deutsch-Jozsa algorithm.[3] In 1994, Peter Shor, then at AT&T Lab, found a quantum algorithm that solves factoring and discrete logarithm problem in polynomial time, which had been a key ingredient for modern day public key cryptosystem.[4] This drew a huge interest in quantum computation and resulted in a rapid development of this field.

There was one problem nonetheless. It seemed that quantum computation- although it promised a great computational power over a classical computer- cannot operate robustly. Since quantum states are extremely fragile to the interaction with the external systems, this can result in a unwanted collapse of quantum state. Fur-

thermore, since measurement collapses quantum state, error-correction seemed impossible. This had been one of the most well-known reasonings against quantum computer. Skeptics had claimed that quantum computer is not very much different from analog computer. Their claim was that both analog computer and quantum computer have better computational power over digital computer, but their inability to correct the error with little overhead makes them useless in practice.

In fact, for the case of quantum computer, it was proved that the claim of the skeptics were false. In 1995, Peter Shor showed that one can detect errors of the quantum system and correct it without disturbing the essential information encoded in the quantum state.[5] Other quantum codes have been discovered thereafter and from the existence of such quantum codes, a *threshold theorem* was proved independently by many authors.[6] [7] [8] [9] [10] [11] *Threshold theorem* tells us that as far as the error rate ϵ is small enough, noisy quantum computer can simulate a noiseless quantum computer efficiently with poly($\log \frac{1}{\epsilon}$) overhead. This is slightly different from a modern digital computer, since its error rate is so low that we do not usually need to actively correct the errors. Therefore, a role of quantum code for quantum computer seems to be more important than the role of digital error-correcting code for digital computer at this point.

In this paper, I will introduce a quantum code which can be embedded on Hurwitz surfaces. I will first start by describing quantum operation in Chapter 2, which will lead to the error models on quantum states. Chapter 3 will describe examples of well-known quantum codes, such as Shor 9-Qubit code, stabilizer code, and Toric code. Chapter 4 contains main result of this paper. I will start this chapter with mathematical background related to Riemannian geometry. Then I will compare this geometrical structure to Toric code to help the readers in understanding the idea. The code's asymptotic behavior will be discussed as well. Chapter 5 will give a summary of the result.

Chapter 2

Preliminary Results

2.1 Qubit

As bit is the basic unit of information in classical digital computation, qubit is the basic unit of information in quantum computation. More formally, qubit is a two-state quantum system whose basis states are denoted as $|0\rangle$ and $|1\rangle$. We can impose a unitary transformation on any state so that it can transform into another state. However, this itself does not draw any interesting picture, since in this way, each of the qubits act independently with other qubits. Therefore, to make something more interesting, we need the qubits to interact with each other. Furthermore, to fully exploit the quantum mechanical effects of the system, we would like to perform arbitrary unitary transformation on the system. In other words, if we have n qubits, the dimension of the Hilbert space is 2^n. What we want is to perform arbitrary unitary transformation in $SU(2^n)$. If we can do so, we call such scheme as a univeral quantum computation.

One of the founding results of quantum computation is that with discrete set of gates, we can perform arbitrary unitary transformation in $SU(2)$ *efficiently.*[12] Furthermore, together with a controlled-NOT(CNOT) gate, we can perform arbitrary unitary transformation in $SU(2^n)$, although this can be done efficiently only for certain classes of operations.

2.2 Density Operator

Density operator is a generalization of normalized projection operator onto a one-dimensional subspace. Due to the probabilistic nature of quantum mechanics, it might be somewhat ambiguous to discuss the probability distribution of quantum states. However, density operator captures this precisely, and has several nice properties. It is defined as following.

Definition 1 *Density operator ρ on a normalized quantum state $\{|\psi_k\rangle\}$ with probability distribution $\{\pi_k\}$ is defined as*

$$\rho \equiv \sum_k \pi_k |\psi_k\rangle \langle\psi_k| . \tag{2.1}$$

As one can see from this definition, density operator is a natural generalization of quantum state. Its trace is always 1, due to the relation $\mathrm{Tr}(\rho) = \sum_k \pi_k \mathrm{Tr}(|\psi_k\rangle \langle\psi_k|) = \sum_k \pi_k$. Moreover, this fact can be generalized to arbitrary hermitian operator A. $\mathrm{Tr}(\rho A) = \sum_k \pi_k \langle\psi_k| A |\psi_k\rangle$, which is an expected value of the hermitian operator. Note that $\mathrm{Tr}(\rho^2) = 1$ if and only if ρ is a normalized rank-1 projection operator. If this is the case, ρ is called as a *pure state*. Otherwise, it is called *mixed state*. This is a very convenient notation when we want to quantify error on the quantum state. For instance, suppose we initially had a pure state $|\psi\rangle \langle\psi|$. After certain amount of time goes, due to the interaction with the external system, the state may not be in its original form. If we can specify the probability that the state remains the same as p_0, the density operator of the state will be the following.

$$\rho' = p_0 |\psi\rangle \langle\psi| + (1 - p_0)\rho'' \tag{2.2}$$

where ρ'' is some mixed state caused by the error. From now on, we will denote $\mathcal{H}_d$ as a Hilbert space of dimension d. Correspondingly, $\mathcal{D}_d$ is a set of density operators in $\mathcal{H}_d$.

2.3 Quantum Operation

In a Copenhagen interpretation of quantum mechanics, there are two procedures allowed on a quantum system. One is a unitary transformation, which is reversible, and another one is measurement, which is not reversible in general. However, these two seemingly disjoint notions can be unified into a single picture. Suppose we are looking at a d-dimensional quantum system. The state of this system can be expressed as a density operator. Therefore, in a closed system, any physical operation is required to map any density operator into another density operator. As discussed in the previous subsection, any density operator must have a trace equal to 1. Furthermore, any density operator is nonnegative- that is, all its eigenvalues are not negative.

Lemma 1 *Any density operator is nonnegative.*

Proof Suppose we have a density operator ρ. Since it is a linear combination of rank-1 hermitian operators, it is hermitian. If ρ has a negative eigenvalue, $\langle\psi|\,\rho\,|\psi\rangle < 0$ for some $|\psi\rangle$. However, this is impossible because of the following relation.

$$\langle\psi|\,\rho\,|\psi\rangle = \sum_k \pi_k \,|\langle\psi|\psi_k\rangle|^2 \geq 0 \qquad \square \tag{2.3}$$

Therefore, it will be interesting to see a family of linear operators that preserves nonnegativity and trace. A family of map corresponding to this operation is called completely-positive trace-preserving map(CPTP). Let $\mathcal{B}(\mathcal{H}_d)$ be a set of linear operators on $\mathcal{H}_d$. A map being completely positive means the following.

Definition 2 *A map Φ is completely positive if $I_k \otimes \Phi : \mathbb{C}^{k\times k} \otimes \mathbb{C}^{d\times d}$ is a positive map for all k.*

Definition 3 *Let Φ be a linear map such that $\Phi : \mathcal{H}_{d_1} \to \mathcal{H}_{d_2}$. Φ is a completely positive trace-preserving(CPTP) map if it is completely positive and preserves trace.*

Note that a naïve thought may lead a quantum operation to be only positive. However, one must note that if we only impose the condition of positivity, there

might be a larger Hilbert space where the density operator of this new system is not positive anymore. This is the reason the map must be completely positive. We define the quantum operation to be a completely positive operation. Typical examples are the unitary evolution and Von Neumann measurement which play central roles in Copenhagen interpretation of quantum mechanics. In fact, these two are just a trivial example of quantum operation. For instance, consider a unitary operation U. For any density operator ρ,

$$\Phi_U(\rho) = U\rho U^\dagger. \tag{2.4}$$

Since the eigenvalue spectrum is invariant under unitary evolution, this map is completely positive. Similarly, for the measurement operation,

$$\Phi_M(\rho) = \sum_i |i\rangle \langle i| \langle i| \rho |i\rangle , \tag{2.5}$$

where $|i\rangle$ forms a countable set of orthonormal basis. Since each $|i\rangle \langle i|$ is positive and have nonnegative coefficients, $\Phi_M(\rho)$ is nonnegative as well. Therefore, measurement operation is positive as well. Readers should note that even though quantum operation encompasses large range of physical change of states, there are physical changes which cannot be expressed as quantum operation. These examples can be found in [13]. Nevertheless, we assume our error model based on the quantum operation formalism, since most of the operations can be described by it.

2.3.1 Operator-Sum Representation

It is quite certain that quantum operation covers wide range of operations, including the canonical examples such as unitary operation and Von Neumann measurement operation. However, there clearly seems to exist a family of quantum operations which are neither a unitary operation nor a Von Neumann measurement operation. In those cases, how can we represent a quantum operation? It turns out that for *any* quantum operation, there exists an extremely useful representation which is called as Operator Sum Representation(OSR).

Definition 4 *OSR of a linear map* $\Phi : \mathcal{B}(\mathcal{H}_d) \rightarrow \mathcal{B}(\mathcal{H}_d)$ *is a l-tuple* $(E_1, E_2, \cdots E_l)$ *such that*

$$\Phi(\rho) = \sum_{k=1}^{l} E_k \rho E_k^\dagger \tag{2.6}$$

Theorem 1 *Any quantum operation admits OSR and vice versa. [13]*

Proof of this theorem can be found in [13].

2.3.2 Canonical OSR

By the virtue of Theorem 1, now we know that any quantum operation can be expressed as OSR. However, there still remains a question. That is, is there a canonical way of expressing the quantum operation as OSR? Not surprisingly, the answer is yes if the input Hilbert space and output Hilbert space have same dimension, and these turn out to be a complete orthonormal basis of $\mathcal{B}(\mathcal{H}_d)$.

Theorem 2 *Any quantum operation whose input and output Hilbert space are same can be represented as a OSR whose components are complete orthogonal basis of* $\mathcal{B}(\mathcal{H}_d)$

Proof : To see this, suppose we have a quantum operation $(E_1, E_2, \cdots, E_k)$ for some k. Consider a matrix A whose entries are defined as $A_{ij} = \mathrm{Tr}(E_i E_j^\dagger)$. Since each of the E_k are $d \times d$ matrix, there exists at most d^2 linearly independent E_ks. Since $A_{ij}^\dagger = A_{ij}$, A is hermitian, and therefore there exists a unitary matrix which diagonalizes A with at most d^2 nonzero entries. In other words, there exists $u \in U(k^2)$ such that

$$uAu^\dagger = A', \tag{2.7}$$

where A' is a diagonalized matrix with at most d^2 entries. Furthermore, we have the following relation.

$$\sum_l \sum_m (u_{lm} E_m) \rho (u_{lm} E_m)^\dagger = \sum_l \sum_m E_m \rho E_m^\dagger u_{lm} u_{lm}^* \tag{2.8}$$

Since

$$\sum_m u_{lm} u_{lm}^* = \sum_m u_{lm} u_{ml}^\dagger = \delta_{ml}, \tag{2.9}$$

we have

$$\sum_l \sum_m (u_{lm} E_m) \rho (u_{lm} E_m)^\dagger = \sum_l E_l \rho E_l^\dagger. \tag{2.10}$$

In other words, if we have a quantum operation, it can be expressed as

$$\Phi(\rho) = \sum_{l=1}^{d^2} E_k \rho E_k^\dagger, \tag{2.11}$$

where $\mathrm{Tr}(E_i^\dagger E_j) = d\delta_{ij}$. □

In other words, given any quantum operation, we can use the orthonormal basis of $\mathcal{B}(\mathcal{H}_d)$ with some coefficients in front of them. When the dimension of the Hilbert space is 2, we can find a well-known example, which is a Pauli matrix.

$$X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \quad Y = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix} \quad Z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} \tag{2.12}$$

As one can easily check, together with the identity matrix, these four matrices satisfy the orthonormality condition. Furthermore, these matrices are unitary and hermitian at the same time. For higher dimensions (> 2), there are two well-known generalizations of pauli matrices that exists in all dimensions. Unitary generalization corresponds to generalized Pauli matrices and hermitian generalization corresponds to generalized Gell-mann matrices. However, in fact we do not need a generalization for Pauli matrices for all dimensions. Since the basic unit of quantum computation is qubit, a complete orthonormal basis for $\mathcal{B}(\mathcal{H}_{2^n})$ will be useful enough. In fact, a concatenation of pauli matrices serves this role quite nicely.

Definition 5 *Pauli group is a multiplicative group with elements of form* $\alpha \otimes_{k=1}^n \sigma_k$, *where* $\alpha = 1, -1, i, -i$ *and* $i_k = 0, x, y, z$. $(\sigma_0 = I, \sigma_1 = X, \sigma_2 = Y, \sigma_3 = Z)$

If we take a *projective* look at the elements, there are $4^n = (2^n)^2$ elements of the group, and clearly these are orthogonal to each other since each pauli matrices have

trace 0. Moreover, the group elements are unitary and has eigenvalue spectrum which are 2^{n-1}-fold degenerate.

2.4 Quantum Channel for Many-Qubit System

Quantum channel is not so different from a quantum operation. In fact, those two are different two names of the same entity. However, since we are interested in a many-qubit system, it will be much more helpful in gaining insight to restrict our Hilbert space to be of dimension 2^n for some n. In this procedure, we assume that each qubits act independently, so that we can look at the behavior of single qubit. Furthermore, if we are looking how the error acts on quantum state, we may set the input Hilbert space and output Hilbert space to be the same dimension.

In this case, by Theorem 2, we know that any quantum operation can be written as a OSR with orthonormal basis of $\mathcal{B}(\mathcal{H}_d)$ with some coefficients in front of them. The most convenient and the most widely used OSR is the Pauli group in such cases. Using the elements of the Pauli group, any quantum operation on a qubit system can be written as the following.

$$\Phi(\rho) = \alpha_0\rho + \alpha_1 X\rho X + \alpha_2 Y\rho Y + \alpha_3 Z\rho Z, \tag{2.13}$$

where $\sum_k \alpha_k = 1$, by the trace-preserving property. When $\alpha_1 = \alpha_2 = \alpha_3$, we say this channel to be a depolarizing channel. If α_0, α_1 are the only nonvanishing coefficients, it is called as a bit flip channel. If α_0, α_3 are the only nonvanishing coefficients, it is called as a phase flip channel.

Chapter 3

Quantum Code

Classical error correcting code was devised to gain resilience against noise. Simplest among those is a repetition code, which encodes 0 and 1 as

$$0 \rightarrow 000 \qquad 1 \rightarrow 111. \tag{3.1}$$

Encoding 0 and 1 this way, if we set the probability of each bit flip as p, the probability that the *bit flip* occurs on this code becomes $O(p^2)$. The reason is simple. When we receive the message, we simply look at each of the bits and take the majority vote. If only one of the bits is flipped, we can detect this error and correct it. Only case we detect the error in a wrong way is when all of three or two bits are flipped, and probability of this instance is $3p^2 + p^3 = O(p^2)$. Therefore, if p is small enough in a sense of

$$p < \frac{1}{2} - \epsilon, \quad \epsilon > 0, \tag{3.2}$$

using large enough repetition code, we can lower the error rate sufficiently enough. This is the idea of error correction.

In quantum world, this may not seem possible, since no-cloning theorem states that we cannot replicate a quantum state in general.[14] Furthermore, if the qubits are entangled, it becomes highly nonintuitive to imagine whether even error *detection* is possible. Contrary to common belief, Peter Shor showed that in fact error correcting code for quantum state exists. Further developments on quantum code lead to more

remarkable results, which will be discussed in the latter part of this section.

The main idea of quantum error correction is that we can measure the *syndrome*- the quantity that quantifies the error - without destroying the original state. For instance, in analogy with the classical error correcting code, if we set the code space as

$$|0_L\rangle \rightarrow |000\rangle \quad |1_L\rangle \rightarrow |111\rangle\,, \tag{3.3}$$

we can detect a bit flip error and correct it. Here the subscript L denotes a *logical* qubit. Since $|000\rangle$ and $|111\rangle$ form a complete basis for two state system, we can detect any bit flip error as far as the bit flip occurred at only one place. This can be done by measuring ZZI, ZIZ, IZZ. What is remarkable is that $|000\rangle$ and $|111\rangle$ are simultaneous eigenstate of these operators with same eigenvalues. Furthermore, if one of the bits is flipped, two of the syndrome measurements ZZI, ZIZ, IZZ outputs -1. This can be summarized in a table.

Flipped Bit	ZZI	ZIZ	IZZ
1	-1	-1	$+1$
2	-1	$+1$	-1
3	$+1$	-1	-1
No Bit Flip	$+1$	$+1$	$+1$

As one can see, when a single bit flip occurs, the syndrome measurement can distinguish which qubit underwent a bit flip operation. After we detect this error, by applying an operator X to the corresponding qubit, we can fix the error. Of course this error correcting scheme cannot correct a phase flip error or a combination of phase flip and bit flip error. However, we can devise a quantum code which is capable of correcting a phase flip error. Consider the following code space.

$$|0_L\rangle \rightarrow |+++\rangle \quad |1_L\rangle \rightarrow |---\rangle\,, \tag{3.4}$$

where

$$|+\rangle \equiv \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) \tag{3.5}$$

$$|-\rangle \equiv \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle). \tag{3.6}$$

Syndrome measurements XXI, XIX, IXX can successfully detect the phase flip error. Since $|0_L\rangle$ and $|1_L\rangle$ are simultaneous eigenstate of XXI, XIX, IXX with same eigenvalue, we can measure any superposition of these two states without disturbing it. Now the syndrome measurement result becomes the following.

Flipped Bit	XXI	XIX	IXX
1	-1	-1	$+1$
2	-1	$+1$	-1
3	$+1$	-1	-1
No Bit Flip	$+1$	$+1$	$+1$

Thus we can see that the syndrome measurement indeed detects the error unambiguously. By applying a Z operation to the phase flipped qubit, we can correct the error.

3.1 Shor 9-Qubit Code

As we have seen, there are simple ways of correcting a bit flip error or phase flip error, yet a code that can correct *both* of these may seem elusive. Peter Shor was the one who first found a way to correct both of these errors. Since I, X, Y, Z form a canonical OSR of any quantum operation on qubit, and $Y = iZX$, this implies that *any* error can be detected and corrected. Let us see the construction. Logical qubit of Shor's code is the following.

$$|0_L\rangle = \frac{1}{\sqrt{8}}(|000\rangle + |111\rangle)^{\otimes 3} \tag{3.7}$$

$$|1_L\rangle = \frac{1}{\sqrt{8}}(|000\rangle - |111\rangle)^{\otimes 3} \tag{3.8}$$

Note that the codewords of Shor's code are simultaneous eigenstates of following operators with same eigenvalues.

$$IZZIIIIII \quad ZIZIIIIII \quad ZZIIIIIII \tag{3.9}$$

$$IIIZZIIII \quad IIIZIZIII \quad IIIIZZIII \tag{3.10}$$

$$IIIIIIZZI \quad IIIIIIZIZ \quad IIIIIIIZZ \tag{3.11}$$

$$IIIXXXXXX \tag{3.12}$$

$$XXXIIIXXX \tag{3.13}$$

$$XXXXXXIII \tag{3.14}$$

For detecting bit flip, by using the first nine measurements, we can easily detect the bit flip error and correct it. For detecting the phase flip error, we can use the last three measurements and easily detect the error, and correct it. Note that the error correcting procedure is exactly the same as for the quantum error correcting code that can only correct bit flip error or the one that can only correct phase flip error. Therefore, as far as the error occurs in a single qubit, we can correct any bit flip error and phase flip error. Furthermore, note that the only remaining canonical element of quantum operation, Y, is simply iZX. Therefore, in fact any error can be detected and corrected as far as the error occurs in only one qubit.

3.2 Stabilizer Code

Stabilizer code encompasses large class of well-known quantum codes, including Shor 9-Qubit code[5], CSS code[15], and toric code[16]. Stabilizer formalism lies on a group structure of the Pauli group G_n, whose projective elements can form a canonical OSR, as we discussed in Chapter 2. Given a group element $g \in G_n$, we say that a quantum state $|\psi\rangle$ is stabilized if $g|\psi\rangle = |\psi\rangle$. Furthermore, the state $|\psi\rangle$ is stabilized by group $S \subset G_n$ if for all elements in $g \in S$, $g|\psi\rangle = |\psi\rangle$. Note that the pauli operators are hermitian and unitary at the same time. The logical quantum state, which we use to process quantum information, are the states that are simultaneously stabilized by the

stabilizer operators. Let the vector space spanned by these vectors as V_S. Suppose we measure the quantum state by stabilizer operator, which is hermitian. If the vector lies in V_S, the out come will be a positive eigenvalue, $+1$. However, if it does not lie in V_S. It will either measure $+1$, and project the state into V_S, or measure -1, which makes the vector outside of V_S. However, since from the syndrome measurements we can uniquely determine which error this is, we can recover the error. Of course, more detailed procedure and the error-correction criterion must be discussed. For these materials, readers should consult [17]. There is one fact that is worth mentioning, since it helps in determining the dimension of the logical code space.

Lemma 2 *Let $S =< g_1, \cdots, g_{n-k} >$ be generated by $n-k$ independent and commuting elements from G_n, and such that $-I \in S$. Then V_S is a 2^k-dimensional vector space.*

The proof can be found in [13].

3.3 Toric Code

Toric Code was first introduced by Kitaev.[16] It is a special case of stabilizer code, since the stabilizer formalism applies to the Toric code as well. In addition to the properties of the Stabilizer codes, toric code has an interesting feature : it can be mapped to a gridded torus. Suppose we have a torus. In general, any point on the torus can be represented by a two-dimensional coordinate system with identification.

$$(x, y) \sim (x + a, y + b) \quad \forall x, y \in \mathbb{R} \tag{3.15}$$

By Eq.3.15, we can divide the surface of the torus into smaller *fundamental regions.* That is, we can divide the torus into smaller regions such that finite number of replication reconstructs the torus. Since Eq.3.15 is topologically identical to a rectangle, we can divide this into $k \times k$ grid. Now, each of the edges become a qubit and we define following operators.

$$\sigma_s^{(z)} = \Pi_{a \in A(s)} \sigma_a^z \quad \sigma_v^{(x)} = \Pi_{b \in B(v)} \sigma_b^x, \tag{3.16}$$

where $A(s)$ is a set of edges surrounding a surface s, and $B(v)$ is a set of edges surrounding a surface v. Since each operators $\sigma_s^{(z)}$ and $\sigma_v^{(x)}$ only simultaneously acts on even number of edges, all of them commute with each other. Note that if the number of edges are $2k^2$, number of vertices are k^2 and the number of faces are k^2, since the local structure is the same everywhere on the surface. However, this does not necessarily mean that a minimal number of elements that generate this stabilizer code is $2k^2$. In fact, since the multiplication of all Z-type operator results in an identity, we have 1 constraint. Similarly, multiplication of all X-type operator results in an identity as well. Therefore, $2k^2 - 2$ is the minimal number of elements that generate the stabilizer code. Therefore, toric code encodes 2 qubits into $2k^2$ qubits. Furthermore, one can see that the maximal number of correctible qubits are k. Note that X or Z error for any qubits can be represented as a set of lines. If the line is not closed, it anticommutes with at least one of the stabilizer measurements. Therefore, any error represented by an open line is correctible. Furthermore, any error represented by a contractible closed line is in fact one of the stabilizer measurements. Therefore, these 'errors' are in fact not errors. Only remaining errors are closed line which are not contractible. Since the minimum length of this line is k, with this quantum code we can correct k errors. Therefore, toric code encodes 2 qubits into $2k^2$ qubits, and can correct at most k errors. In fact, for any closed 2-dimensional surface, the number of encoded qubits is identical to the number of noncontractible loops which cannot be mapped to each other with continuous transformation. This quantity turns out to be $2g$ for hyperbolic surfaces, where g is a number of genuses.

Chapter 4

Quantum Code on a Hurwitz Surface

As Kitaev's construction of Toric code, in fact *any* graph on a closed two-dimensional surface suffices to construct a quantum code, since the commutativity of the operators do not depend on the topology of the surface, but rather comes from the construction of the operators themselves. Therefore, the stabilizer formalism can be applied to these geometrical quantum codes as well. Hurwitz surface is a two-dimensional closed surface with some additional structure that we will explain and exploit in constructing a quantum code. As a Toric code can be constructed by tiling a torus with squares, code on a Hurwitz surface can be constructed by tiling a Hurwitz surface with triangles.

4.1 Riemann's Uniformization Theorem and Hyperbolic Geometry

4.1.1 Uniformization Theorem

Riemann's uniformization theorem says that any surface admits a Riemannian metric of constant Gauss curvature. Furthermore, from this, a surface is a quotient of either a sphere, Euclidean plane, or hyperbolic plane by a free action of a discrete subgroup of

an isometry group. If we want to study a structure of some Riemann surface, we can impose a quotient on the universal cover, which is a Poincaré disk. Poincaré disk is a unit circle centered at the origin on the complex plane. Since the geometry on this disk can be understood in a straightforward manner, by looking the surface in this way, we can compute several interesting quantities, such as the shortest noncontractible geodesic on the surface, which is related to the number of correctible errors.

4.1.2 Hyperbolic Geometry

Hyperbolic geometry is a non-Euclidean geometry, which means that the parallel postulate of Euclidean geometry is not true any more. It is a curved space. There are many models of hyperbolic geometry, which are conformally equivalent to each other. In this paper, we will use the hyperbolic geometry of Poincaré disk model. In this model, a metric g_{ij} is given by

$$g_{ij} = 2\delta_{ij}\frac{1}{1-\sum_k x_k^2}, \tag{4.1}$$

where the Euclidean norm satisfies $|\vec{x}| < 1$. Therefore, we have

$$ds^2 = 4\frac{dx^2+dy^2}{1-x^2-y^2}, \tag{4.2}$$

and

$$dA = 4\frac{dxdy}{1-x^2-y^2}. \tag{4.3}$$

Here the notion of 'straight line' is not identical to the notion of straight line in Euclidean space. In fact, one can see that the notion of 'straight line' in Euclidean geometry can be just generalized to a geodesic. Geodesic in Poincaré disk is part of a circle that is perpendicular to the unit disk. Using these geodesic lines, we can define a 'triangle', which is formed as an intersection of three geodesic lines. Of course, we must require each of the geodesics to intersect with the other two in two different points. Calculating area of a triangle directly from Eq.4.3 is somewhat nontrivial.

The derivation procedure is in [18], and the result is

$$A = \pi - \alpha - \beta - \gamma, \tag{4.4}$$

where α, β, and γ are the internal angles of the triangle.

4.2 Tiling Group

Tiling group, as the name suggests, is a group whose elements can completely tile a surface. Simplest example will be a tiling of a 2-dimensional Euclidean plane with finite sized squares. Except for the identity element, elements of the tiling group must map one square to another square. It can either be rotation around the vertex, or a translation which has two degrees of freedom. In fact, this group is infinite since there can be infinitely many faces. However, if we identify some points, thereby wrapping around the surface, the surface becomes closed. Since there are finitely many faces, now we have a finite group. Furthermore, one can easily see that the *local* structure of the surface does not change, whereas the *global* structure, such as the translation, becomes different. Instead of allowing infinitely long translation, long enough translation eventually comes back to the starting point due to the identification of the points.

Now let us see how the geometry and the corresponding group structure changes. Before we identify points, we have translations which are commutative to each other, rotations, and reflections which reverses the orientation. If we identify points so that the surface becomes a torus with discretized coordinates $(i, j), i, j \in \{1, 2, \cdots, n-1, n\}$, there are n^2 squares which can be mapped to each other by the group element. Therefore, including the orientation-reversing isometries, the order of the group is $|G| = 2n^2$. Note that this example can be generalized naturally to more complicated surfaces. If we take a look at how the tiling worked, one can see that tiling elements, which was a square in the example, are identical geometrical objects, and therefore have same areas. Each of these geometrical objects are called as *fundamental region.*

By mapping any fundamental region via the $2n^2$ group elements, we can fully recover the original surface. However, the group that includes the orientation-reversing elements will not be used in this paper, since it induces a redundancy in counting the number of tilings that tiles the surface. Therefore, in this example, although the full symmetry group has order $2n^2$, once we only take care of the orientation-preserving group, the order of the symmetry group will have order n^2, and it will only include translations and rotations.

As one can see in this example, tiling of the surface and its corresponding tiling group are intimately related to each other. Particularly, since we are looking at the structure of the quantum code, the essential features such as the encoding rate and the rate of correctible errors, can be computed from the group elements. For instance, computing the encoding rate is trivial since the number of qubits encoded on the surface is $2g$, where g is the genus of the surface, and the total number of qubits is proportional to the number of tilings on the surface. For the number of correctible errors, once we can represent the group elements as translation and rotation operators, we can compute the minimal translation lengths that is equal to the identity operation. Of course the latter computation will not be easy in general, since representing an abstract group element is not an easy task to do. However, for Hurwitz group, which is an orientation-preserving tiling group of Hurwitz surface, such representation exists in a form that is easy to compute its translation lengths. Furthermore, Vogeler showed a combinatorial approach on calculating all the lengths of closed geodesics on Hurwitz surfaces by using the representation of the group elements.[19] Therefore, at least for the case of Hurwitz group, it is in fact possible to do this computation.

4.3 Hurwitz Group

Hurwitz group is a special tiling group on a hyperbolic geometry which is, in a sense, a maximal symmetry group on a closed hyperbolic surface with constant Gauss cur-

vature. To understand this, let us start with Gauss-Bonnet theorem.

$$\int\int KdA = 2\pi\chi(M), \tag{4.5}$$

where K is the curvature of the surface, χ is Euler characteristic, and M is the corresponding manifold. Since we are concerned with a surface with constant Gauss curvature, we have

$$A = 4\pi(g-1). \tag{4.6}$$

Since we are looking for a discrete symmetry group on this surface, this area must be an integer multiple of the area of the fundamental region. If the fundamental region is a hyperbolic triangle, it has an area

$$\pi(1-\frac{1}{l}-\frac{1}{m}+\frac{1}{n}), \tag{4.7}$$

where $\frac{\pi}{l}, \frac{\pi}{m}, \frac{\pi}{n}$ are the angles on each of the vertices. Furthermore, l, m, n must be integer-valued since otherwise the group will not be discrete or the fundamental regions will overlap with each other. From this condition, we can derive Riemann-Hurwitz equation for genus g.

$$g = 1 + \frac{|G|}{2}(1-(\frac{1}{l}+\frac{1}{m}+\frac{1}{n})) \tag{4.8}$$

Since we are studying the geometry on a hyperbolic plane, the maximum area of a triangle is bounded by the following inequality. [18]

$$\pi(\frac{1}{l}+\frac{1}{m}+\frac{1}{n}) < \pi. \tag{4.9}$$

It is easy to check that in fact the maximum area for integer values of l, m, n can be achieved when $(l,m,n) = (2,3,7)$. In this case, the Riemann-Hurwitz equation can be expressed as the following.

$$g = 1 + \frac{|G|}{84} \tag{4.10}$$

A group G which achieves this bound is called as Hurwitz group. Note that Eq.4.10 is merely a bound, and does not guarantee the existence of such group. However, in fact, there are infinitely many Hurwitz groups with different group orders.[20] This is the reason that we can see the asymptotic behavior on such surfaces.

4.4 Quantum Code on a Hurwitz Surface

Since Hurwitz group maps one of the tiles to another tile on the surface, it is easy to see that for a Hurwitz surface of genus g, total number of faces is $84(g-1)$. Since the fundamental region is triangle, each faces have 3 edges. Therefore, dividing by the overcounting factor, total number of edges become $126(g-1)$. Since this surface encodes $2g$ qubits, encoding rate becomes

$$\frac{k}{n} = \frac{2g}{126(g-1)} = \frac{1}{63} + O(\frac{1}{g}) \tag{4.11}$$

in the asymptotic limit. However, ratio between the correctible errors and total number of qubits, which is

$$\frac{t}{n} = \frac{f(sys(M))}{126(g-1)} \tag{4.12}$$

still remains elusive. Here $sys(M)$ is a length of the shortest noncontractible geodesic on a Riemannian manifold M, and is called as *systole*. f is some function which satisfies $t = f(sys(M))$. We will see how f behaves later in this paper. Obviously, for same encoding rate, one would like to prefer a quantum code which is capable of correcting more errors. Therefore, it is worth mentioning the bound on how $\frac{t}{n}$ behaves. Quantum Gilbert-Varshamov bound for stabilizer code, which gives an asymptotic limit of the quantum code, was derived in [17].

$$\frac{k}{n} \geq 1 - \frac{2\log(3)t}{n} - H(\frac{2t}{n}) \tag{4.13}$$

Asymptotically the left hand side converges to $\frac{1}{63}$. Numerically computing the

bound on $\frac{2t}{n}$, we find that the bound is achieved near

$$\frac{2t}{n} \sim 0.1852 \tag{4.14}$$

in an asymptotic limit. One might think that the code will behave better by dividing up the fundamental regions. For instance, one can imagine dividing the fundamental region into m identical pairs. The encoded qubit still remains as $2g$ and the total number of qubits grows as $126(g-1)m$. However, since the length of the systole is roughly proportional to $\sqrt{m}$, in an asymptotic limit that $m \to \infty$, we get the error correcting ratio $\frac{t}{n} \to 0$, no matter what the g is. In fact, we can never escape from this problem as far as the quantum code is a surface code, or a generalization of it.[21] There is a known Lower bound on the systole length. [22]

$$sys(M) \geq \frac{4}{3}\log(g), \tag{4.15}$$

but we do not know whether the bound on the inequality is saturated or not. If the bound is saturated for Hurwitz surfaces, the corresponding quantum code will have $\frac{t}{n}$ asymptotically converging to 0. We will examine this by numerically generating Hurwitz group and calculating the length of closed translations.

4.4.1 Generating Hurwitz Group

Note that the fundamental region of the Hurwitz surface is a triangle which has angle $\frac{\pi}{2}$, $\frac{\pi}{3}$, and $\frac{\pi}{7}$. Furthermore, one can see that two consecutive reflection on different edges result in a rotation of order 2, 3, and 7, and applying these three rotations consecutively results in an identity element. Since Hurwitz group only includes the orientation-preserving group elements, we can see that any Hurwitz group will have a group presentation of the following form.

$$G =< a, b | a^3, b^7, (ab)^2, \cdots >, \tag{4.16}$$

where $\cdots$ are the extra relators that 'wrap up' the surface. In [19], the author showed that in fact any extra relator must be a hyperbolic translation and any hyperbolic translation can be written as a repetition of two letters $R = b$, $L = ba^{-1}$. Here R and L can be thought as a 'right turn', and 'left turn' on a slightly modified tiled surface. Instead of tiling Hurwitz surface with $(2, 3, 7)$ triangle, we can gather 14 triangles to form a heptagon, and let this heptagon be a fundamental region. Even though the surface has changed, note that the group structure remains the same, since order 7 rotation corresponds to the rotation through the center of the heptagon, 3 to the rotation through the vertices, and 2 to the rotation through the center of each edges. In this tiling structure, R can be thought as taking a right turn on the vertex,and L can be thought as taking a left turn on the vertex. Since each vertex meets with 3 edges, this definition makes sense.

He also showed that any hyperbolic translation can be written in a *proper label*, which does not have three consecutive Rs or Ls in its sequence, starts with R, and ends with L.

Definition 6 *A label is proper if it begins with R, ends with L, and contains neither RRR nor LLL as a consecutive subsequence.*

This significantly reduces the number of possible extra relators, and thus we can generate Hurwitz groups with reduced redundancy. By choosing extra relators, we generated the corresponding Hurwitz group by using MAGMA.

4.4.2 Computing the Length of the Systole

A fractional linear transformation T is a transformation defined on a complex plane.

$$T_M(z) = \frac{az + b}{cz + d} \tag{4.17}$$

It is convenient to think this as a

$$M = \begin{pmatrix} a & b \\ c & d \end{pmatrix}, \tag{4.18}$$

since one can show that

$$T_{M_1} \circ T_{M_2} = T_{M_1 M_2}. \tag{4.19}$$

In [19], the author shows that there is a group representation of Hurwitz group with $a \to A$, $b \to B$, where

$$A = \begin{pmatrix} \cos\frac{\pi}{3} & \sin\frac{\pi}{3} \\ -\sin\frac{\pi}{3} & \cos\frac{\pi}{3} \end{pmatrix}, \qquad \begin{pmatrix} \cos\frac{\pi}{7} & b\sin\frac{\pi}{7} \\ -\frac{1}{b}\sin\frac{\pi}{7} & \cos\frac{\pi}{7} \end{pmatrix}. \tag{4.20}$$

Here $b = (\cot\frac{\pi}{3})(\cot\frac{\pi}{7}) + \sqrt{(\cot\frac{\pi}{3})(\cot\frac{\pi}{7}) + 1}$. Furthermore, the translation length can be computed as

$$2\cosh^{-1}\left|\frac{1}{2}\mathrm{Tr}(M)\right|. \tag{4.21}$$

Therefore, once we generate Hurwitz group, we can compute all the translation length using Eq.4.21. Furthermore, since we have the group in our hand, we can find a translation that generates all the translation on that axis. By finding the order of this translation, one can compute the length of closed geodesic. The minimum length among these will correspond to the length of the systole.

4.4.3 Relation between Systole Length and Correctible Errors

Roughly speaking, if we embed a graph on Hurwitz surface, as the systole length increases, the number of correctible errors should increase as well. However, those two do not have a strict linear relationship, since the number of correctible errors is proportional to the length of the shortest closed translation which follows the edges of the heptagon, following the R and L label. Since there is always at least one translation caused by R and L which deviates from the geodesic line, the length of the systole cannot be directly translated to a number of correctible errors. This can be seen in FIG.4-1

Of course this inconvenience can be remedied by dividing up the fundamental regions into smaller pieces. In the limit this division goes to infinity, the systole

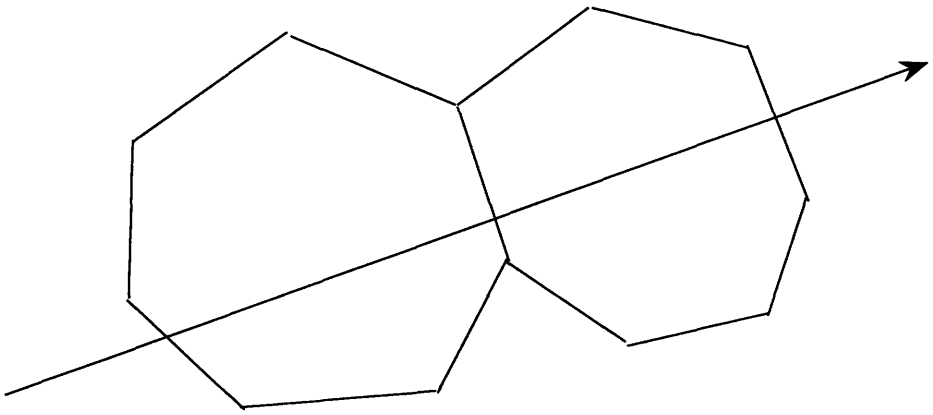

Figure 4-1: Arrowed line is a geodesic, and the heptagons are the fundamental regions. As one can see, except for some trivial cases, the geodesic does not overlap with the edges of the fundamental region.

length becomes proportional to the number of correctible errors. However, as we have explained, further division of the fundamental region decreases both $\frac{k}{n}$ and $\frac{t}{n}$, which is not desirable. In fact, we can set a bound between a number of correctible errors and systole lengths. To show this, I will first introduce some definitions and useful facts which were introduced in [19].

Definition 7 *Suppose t is a hyperbolic translation. Label-length of t is the minimal length taken by the label $\lambda(E)$ as E ranges over all edge-paths from e_1 to $t(e_1)$, where e_1 ranges over all edges in the tiling.*

Definition 8 *Suppose X is the axis of a translation t. The corridor $C(X)$ of X is the collection of faces in the tiling which have non-empty intersection with X.*

Lemma 3 *Suppose E is an edge-path corresponding to a translation g_E with axis X. If $\lambda(E)$ is proper, then E is contained in $C(X)$. (Vogeler, 2003)*

Theorem 3 *Let $l(t)$ be a minimal length of edge-path for hyperbolic translation t. Let $l_E(t)$ be a length of translation t. For any hyperbolic translation t, $l(t) \leq \alpha l_E(t)$ for some α.*

Proof : Note that any hyperbolic translation admits a proper label. Let X be the translation axis of t. The edge-path corresponding to this proper label lies in $C(X)$. Let t_F be a segment of t which intersects with one of the tiling face F. Divide t into t_Fs. Note that for each t_F, the corresponding truncated minimal edge-path is always smaller than $\alpha l_E(t_F)$, for some universal constant α. Therefore, summing up these truncated pieces, we have

$$l(t) \leq \alpha \sum_F l_E(t_F) = \alpha l_E t. \tag{4.22}$$

□

Since $l(t) \geq l_E(t)$ by definition, we have

$$l_E(t) \leq l(t) \leq \alpha l_E(t) \tag{4.23}$$

for some α. Since $l(t)$ is proportional to the number of correctible errors, one can see that indeed the length of the systole is a good measure of the number of correctible errors, even in the absence of further division of fundamental regions.

4.4.4 Numerical Results

We generated as many Hurwitz groups as possible by experimenting on different extra relators. However, there were many instances where the program failed to generate a group. This may be due to the memory limit of the computer. As a result, among the Hurwitz groups we have generated, the number of genus was always smaller than the maximal genus numerically found in [19]. The plot between the genus and the systole length is shown in FIG.4-2

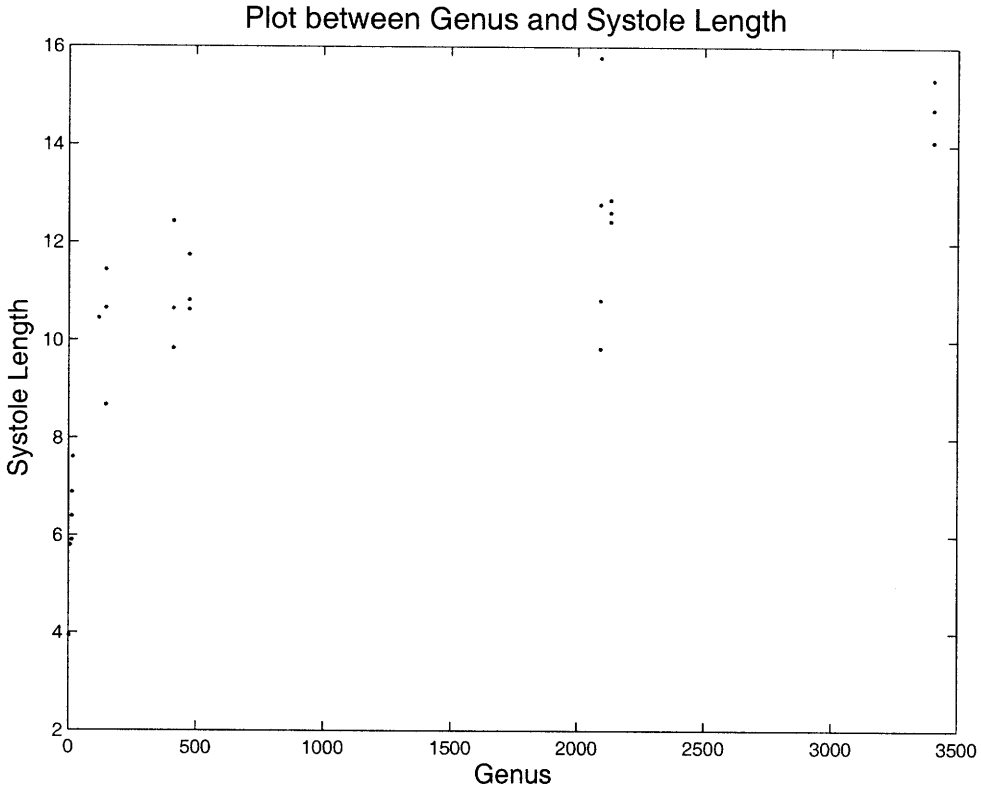

Figure 4-2: Plot between the genus and the systole length. One can see that the systole length increases at the beginning, but the increasing rate dramatically falls as the genus grows larger.

Similarly, the plot between the genus and the $\frac{sys(M)}{g}$ was also obtained. Note that in, FIG.4-3 the $\frac{sys(M)}{g}$ is plotted in log scale. Although the decreasing rate seems subexponential, it still seems to go down superpolynomially, or at least polynomially with a large degree. Since the number of correctible errors is linearly bounded by the length of the systole, it will not be able to make a difference in the order of magnitude change shown in FIG.4-3. Since $\frac{\mathrm{sys}(M)}{g}$ is linearly bounded by $\frac{t}{n}$, FIG.4-3 suggests that asymptotically $\frac{t}{n}$ should vanish very fast in the limit $g \to \infty$. Therefore, it seems highly unlikely that the code generated on Hurwitz surfaces are good quantum error correcting code, in a sense that it cannot saturate the Gilbert-Varshamov bound.

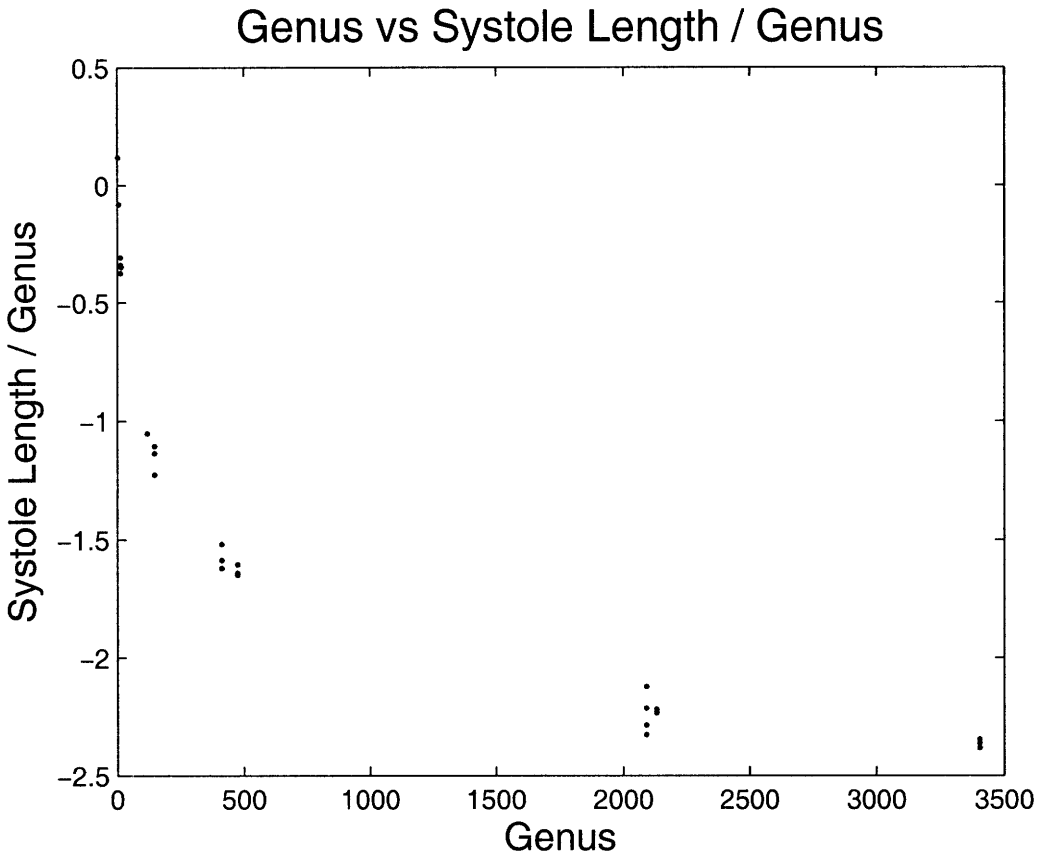

Figure 4-3: Plot between the genus and the (systole length)/genus. Note that the (systole length)/genus is in $\log_{10}$ scale.

Chapter 5

Conclusion

In this paper, We studied properties of quantum code on Hurwitz surfaces. As we could see in the numerical result, in the asymptotic limit $g \to \infty$, $\frac{t}{n}$ seems to converge to 0. However, in this limit, we observe that the encoding rate converges to $\frac{1}{63}$, which is a constant value. Our work can be compared to [23]. In [23], the authors construct a quantum code whose encoding rate converges to 1 but the number of correctible errors remains as 1. On the contrary, our code, achieves a smaller encoding rate, but is able to correct more errors, since asymptotically the systole length grows at least logarithmically.[22] From these results, it seems that there is a tradeoff between $\frac{k}{n}$ and $\frac{t}{n}$ for topological quantum codes. It will be interesting to elucidate this aspect, since it seems to be largely dependant on the structure of large-genus surface.

Bibliography

[1] R. Feynman, Simulating physics with computers. Int. J. Theor. Phys. **21**, pp.467-488 (1982).

[2] D. Deutsch, Quantum-theory, the Church-Turing principle and the universal quantum computer. Proc. R. Soc. Lond. A **400**, pp.97-117 (1985).

[3] David Deutsch, Richard Jozsa. Rapid solutions of problems by quantum computation. Proc. R. Soc. Lond. A, **439**, pp. 553 (1992).

[4] P. W. Shor, Polynomial-Tiem Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer. SIAM J. Computing **26**, pp. 1484-1509 (1997).

[5] P. W. Shor, Schemes for reducing decoherence in quantum computer memory. Phys. Rev. A **52**, pp. R2493-2496 (1995).

[6] P. W. Shor, Fault-tolerant quantum computation, Proc. 37nd Annual Symposium on Foundations of Computer Science, p. 56 (1996).

[7] D. Gottesman, A theory of fault-tolerant quantum computation, Phys. Rev. A **57**, pp. 127-137 (1998).

[8] D. Aharonov, Fault tolerant quantum computation with constant error, Proc. 29th Annual Symposium on Theory of Computing, pp. 176-188 (1997).

[9] E. Knill, Quantum computing with very noisy devices, quant-ph/0410199 (2004).

[10] P. Aliferis, D. Gottesman, J. Preskill, Quantum accuracy threshold for concatenated distance 3-codes, Quant. Inf. Comput. **6**, pp. 97-165 (2006).

[11] P. Aliferis, Level reduction and the quantum threshold theorem, Caltech Ph.D thesis, quant-ph/0703230 (2007).

[12] C. M. Dawson, M. A. Nielsen, The Solovay-Kitaev algorithm, Quant. Inf. Comput. **6**, pp. 81-95 (2006).

[13] I. L. Chuang, M. A. Nielsen, *Quantum Computation and Quantum Information* (Cambridge University Press, Cambridge, 2000).

[14] W. K. Wootters and W. H. Zurek, A Single Quantum Cannot be Cloned, Nature **299**, pp. 802-803 (1982).

[15] A. R. Calderbank, P. W. Shor, Good quantum error-correcting code exists, Phys. Rev. A **54**, pp.1098-1105 (1996).

[16] A. Y. Kitaev, Fault-tolerant quantum computation by anyons, Annals of Physics **303**, Issue 1, pp. 2-30 (2003).

[17] D. Gottesman, Stabilizer codes and quantum error correction, Caltech Ph.D thesis, quant-ph/9705052 (1997).

[18] J. W. Anderson, *Hyperbolic Geometry* (Springer, London, 1999).

[19] R. Vogeler, On the geometry of Hurwitz surfaces, Florida State University Ph.D thesis (2003).

[20] A. M. Macbeth, On a theorem of Hurwitz, Proc. Glasgow Math. Assoc. **5** pp. 90-96 (1961).

[21] M. H. Freedman, D. A. Meyer, and F. Luo, Z_2-systolic freedom and quantum codes, 287-320, Comput. Math. Ser. (Chapman & Hall/CRC, Boca Raton, FL, 2002).

[22] M. G. Katz, M. Schaps, U. Vishnes, Logarithmic growth of systole of arithmetic Riemann surfaces along congruence subgroups, math.DG/0505007 (2005).

[23] H. Bombin and M. A. Martin-Delgado, Topological quantum error correction with optimal encoding rate, Phys. Rev. A **73**, 062303 (2006).