
[All ETDs from UAB](#)

[UAB Theses & Dissertations](#)

2013

A Generalized Noisy Communication Channel Approach for Quantum Computing

Fan Xiong
University of Alabama at Birmingham

Follow this and additional works at: <https://digitalcommons.library.uab.edu/etd-collection>

Recommended Citation

Xiong, Fan, "A Generalized Noisy Communication Channel Approach for Quantum Computing" (2013). *All ETDs from UAB*. 3390.
<https://digitalcommons.library.uab.edu/etd-collection/3390>

This content has been accepted for inclusion by an authorized administrator of the UAB Digital Commons, and is provided as a free open access item. All inquiries regarding this item or the UAB Digital Commons should be directed to the [UAB Libraries Office of Scholarly Communication](#).

A GENERALIZED NOISY COMMUNICATION CHANNEL APPROACH FOR
QUANTUM COMPUTING

by

FAN XIONG

MURAT M. TANIK, CHAIR
GREGORY A. FRANKLIN
MURAT N. TANJU
GREGG L. VAUGHN
B. EARL WELLS

A DISSERTATION

Submitted to the graduate faculty of The University of Alabama at Birmingham,
in partial fulfillment of the requirements for the degree of
Doctor of Philosophy

BIRMINGHAM, ALABAMA

Aug. 2013

Copyright by
FAN XIONG
2013

A GENERALIZED NOISY COMMUNICATION CHANNEL APPROACH FOR
QUANTUM COMPUTING

FAN XIONG

COMPUTER ENGINEERING

ABSTRACT

Quantum computing is considered to be a promising technology for some specific types of computing purposes. Modeling of quantum computing could help study and research of this field. In this dissertation, we approach quantum computing from a communication perspective that is to model quantum computing with a generalized noisy communication channel approach.

We proposed a generalized noisy communication channel approach (GNCom) for quantum computing. The proposed GNCom approach is applied to quantum computing as a universal quantum register. GNCom further extended the noisy communication channel framework by associating it with Quantum Electrodynamics (QED). Based on GNCom and QED, we proposed a quantum computer design technique.

Keywords: Quantum computing, quantum electrodynamics, information theory, noisy communication channel, Fourier analysis

DEDICATION

To my lovely daughter, Cara.

ACKNOWLEDGMENTS

Looking back on my graduate study in the United States, I have many people to thank. This dissertation is not possible without the help and support of professors, staff, lab fellows, family and friends.

First and foremost, I would like to express my deepest thanks to my academic advisor and committee chair, Dr. Murat M. Tanik. Thanks so much for being a mentor, father figure and great friend. I am indebted to him for each step of my academic and professional growth. This dissertation is not possible without his guidance, support and encouragement. Dr. Tanik's broad knowledge, rich experience, inspiring passion and enthusiasm, and hard-working spirit motivated me and will be a lifetime inspiration.

Thanks to Dr. Tanju for serving on my committee, and thanks for the sharp feedback and comments he has provided. I am grateful for spending his valuable time to help me practice and improve my work. Thanks to Dr. Vaughn and Dr. Wells for serving on my committee, and thanks for their comments and feedback with my dissertation and course plans. Special thanks to Dr. Wells for driving all the way back and forth from Huntsville on my defense day. Thanks to Dr. Franklin for serving in my committee.

I would like to especially thank Dr. Bunyamin Ozaydin for his critical review of my dissertation. His suggestions have improved the presentation of the results and distinction among various contributors to this project in our research group. I stood on his shoulders to finish this dissertation. I also thank him for the meetings and discussions which inspired my research. Thanks again for reviewing my dissertation and thanks for the valuable feedback and comments. Also I wish to thank my lab fellow, Luai Najim, for the discussions, comments, and feedback for my dissertation.

Special thanks to Jessica Bonner, Lori and Dave Henderson for editing and proof reading my dissertation.

I also owe many thanks to the faculty and staff in the ECE department. I'm deeply indebted to the department for the financial support all through my graduate study here at UAB. Thanks to the current chair Dr. Tannenbaum, previous chair Dr. Massoud and Dr. Vaughn. I truly appreciate their great help. Many thanks go to Dr. Jannett, Dr. Massoud, Dr. Kathik, Dr. Haider, Dr. Callahan, Dr. Nelson, and Jon Marstrander. Thanks for being great professors and friends to me. Special thanks to Jan, Sandra, and Debbie, wonderful women who have offered me tremendous help with paperwork and lab equipment. Thanks for always thinking of me and always being available. Thanks to the other staff women in the mechanical and material science department; I will remember their friendly smiles and greetings.

Many thanks to Dr. Nadar I. Rafla, Dr. Sinming Loo, and Dr. Janet Callahan from Boise State University for help, support and encouragement during my graduate study. Thanks to Dr. Susan Vasana from the University of North Florida for collaboration on the medical research. Thanks to Dr. Skellum in the computer science department for involving me in the high-performance computing projects. Thanks to Dr. John Tanik, Dr. Ali H. Dogru, and Dr. Varadraj P. Gurupur for their help and collaboration for paper publication.

Thanks very much to my industry mentors YvonneYang, Jessica Manfredi, and Lori Henderson. They have been great role models and support for me. Special thanks to Lori and Dave for their prompt and detail review of my dissertation.

Thanks very much to my friends in the graduate program, Shanshan, Vikas, Tauhid, Manar, Vinaya, Pawan, Qingyun, Yanguo, and Shahed. We had a great time working together. I value and appreciate their friendship.

Special thanks go to Dr. Dale Callahan from the IEM program, Karen King from UAB career service for their help with my job hunting. Special thanks to my friends Vikas Singal from Cactus semiconductor, Zheng Xu from Texas A&M, Tai Wang from Broadcom, Qian Liu from Qualcomm, Kaijun Li from Intel, Yingting Li from Micron, Paul Korn and Frank Shen from Bio-rad Lab, Zhonghua Dong and Michelle Hu from Hermes Microvision, Xueyi Wang, Stephen Parke and Dan Lawrence from Northwest Nazarene University, Kevin Hall, GD Kalra and Marvin Tom from Xilinx, Meagan Harn and Zhongping Cai from KLA Tencor, Sheng Hu and Mingqiang Cui from Apple, Stan Thompson from ATT, Priyanka Parikh from NVIDIA, Jessica Ford from Seagate, Bob Gardner from Missing Link Electronics, Xiaoping Li from Stryker, Tolga Bayik from Broadcom for their help and references for my job-seeking process. Thanks to Johanna Sartori and Sean Kao for my first interview within the industry. I'm very thankful for Dr. Tanik, Dr. Rafla, Dr. Susan, and Vikas for providing impressive references for me for my job application.

Many thanks to my dear friends: Yujiao; Ming, Haibo and Janet; Ya, Yu and Veronica, Xin, Jianwei and Andy; Qiling and Sophia; Zhuo, Jie, Zhuangzhuang and Melody; Jing and Cindy; we had a lot of happy times together. They have always been my great and strong supporters. Many thanks go to my friends Lori and Dave Henderson for their friendship and help with all aspects of my life and study.

Last but not least, I would like to express my deepest thanks and appreciation to my family. To my cousins JiangXia Liu, Liangliang Cao, Iris and Rosy: thanks for being so close to me. Jim and Kathi, thanks for being my American parents. My lovely daughter, Cara, thanks for being my greatest inspiration. Thanks to my husband Zhiyong Li for all the times we spent together. My mother Yufen Xu, Dad Yangbin Xiong, my brother Nian Xiong and my sister-in-law Xi Long, thanks for always standing by me and giving me selfless love.

TABLE OF CONTENTS

	<i>Page</i>
ABSTRACT	iii
DEDICATION.....	iv
ACKNOWLEDGEMENT	v
LIST OF TABLES.....	xii
LIST OF FIGURES	xiv
LIST OF ABBREVIATIONS.....	xx
CHAPTER	
1. INTRODUCTION.....	1
Contributions.....	4
2. THE GENERALIZED NOISY COMMUNICATION CHANNEL APPROACH	6
Shannon’s noisy communication channel paradigm.....	6
Noisy communication polygon representation.....	9
U matrix.....	10
Fourier series representation of regular polygon.....	10
U transform.....	27
Riemann zeta function.....	30
U matrix grouping	34

Chebyshev matrix eigenvalues, Polygon geometry and Amplitwist.....	42
3. GNCOM FOR QUANTUM COMPUTING	50
Introduction to quantum computing.....	50
History of quantum computation.....	50
Basics.....	52
Algorithms.....	59
Applications: Field programmable Gate Array (FPGA)	61
Noisy communication channel representation of quantum computing.....	61
Entangled quantum states and Fourier polygon	61
Multiple-input Multiple-output (MIMO) channels representation of quantum computing	69
4. THE EXTENDED NOISY COMMUNICATION CHANNEL FRAMEWORK.....	73
The extended noisy communication channel framework.....	73
QED.....	75
Light partial reflection.....	76
Probability Amplitude	77
Polygon geometry and probability amplitude	86
Generalized noisy communication channel approach for QED analysis	92
Quantum computer design technique based on QED and Quantum Fourier Series..	95
5. CONCLUSIONS	98

Future research	98
Engineering application of the transform	98
Information content	100
LIST OF REFERENCES	102
APPENDIX.....	107
Appendix A: Math Background.....	107
Complex numbers.....	107
Complex function	112
Permutation matrices	113
Fourier matrices.....	116
Pauli matrices	118
Riemann zeta function.....	119
Chebyshev polynomial	120
Multivalued logic.....	122
Appendix B: The Noisy Communication Channel Framework	125
Communication channel representations.....	125
Appendix C: More results of U matrix	132

LIST OF TABLES

Tables	Page
Table 1 Relationship of equation parameters and U matrix	22
Table 2 p changes from 1 to 10 for n=7 and q=1	23
Table 3 U matrix changes as row number increases	24
Table 4 Inverse Fourier transform of U matrix changes as row number increases	25
Table 5. FFT and U transform	28
Table 6 FFT, U transform, Zeta transform	33
Table 7 Fourier polygon grouping based on N changes	34
Table 8 Fourier polygon grouping with α change.....	38
Table 9 The-state-of-the-art of quantum computers	52
Table 10 Truth table of a NOT gate.....	56
Table 11 Information content.....	101
Table 12 Chebyshev polynomial of degree 10	121
Table 13 Chebyshev polynomial of the second kind up to degree 10	122
Table 14 p changes from 1 to 9 for n=7 and q=2.....	132
Table 15 U matrix changes as row number increases.....	133
Table 16 Inverse Fourier transform of U matrix changes as row number increases	135
Table 17 p changes from 1 to 10 for n=7 and q=3.....	137
Table 18 U matrix changes as row number increases.....	138
Table 19 Inverse Fourier transform of U matrix changes as row number increases	139

Table 20 p changes from 1 to 10 for n=7 and q=4.....	141
Table 21 U matrix changes as row number increases.....	142
Table 22 Inverse Fourier transform of U matrix changes as row number increases	144
Table 23 p changes from 1 to 10 for n=7 and q=5.....	146
Table 24 U matrix changes as row number increases.....	147
Table 25 Inverse Fourier transform of U matrix changes as row number increases	148
Table 26 p changes from 1 to 10 for n=7 and q=6.....	150
Table 27 U matrix changes as row number increases.....	151
Table 28 Inverse Fourier transform of U matrix changes as row number increases	153

LIST OF FIGURES

Figure	Page
Figure 1 Modeling paradigms.....	2
Figure 2 Communication-based modeling.....	3
Figure 3 An example of communication channel.....	7
Figure 4 A noisy communication channel interpretation.....	8
Figure 5 Shannon’s representation of noisy communication channel.....	8
Figure 6 Error content graph representation of noisy communication channel of Figure 5	9
Figure 7 A communication channel with non-convex irregular polygon error content graph representation.....	10
Figure 8 Fourier series for polygons with $p=1$, and $p=2$	11
Figure 9 Explicitly written U matrix.....	14
Figure 10 Evolution of 5 by 2 polygons with increasing resolution.....	15
Figure 11 Evolution of 11 by 4 polygons with increasing resolution.....	15
Figure 12 Some Polygon examples for $u_{6,1p,t}$, $u_{7,3p,t}$, $u_{8,2p,t}$, and $u_{9,3p,t}$	16
Figure 13 Irregular polygon formed by permutation matrix of Example 2.	18
Figure 14 The regular polygons used to construct irregular polygons.	19
Figure 15 The process of the irregular polygon formed by pieces taken from regular polygons.....	20
Figure 16 Fourier series representation of irregular polygons of 8 vertices.....	20

Figure 17 Shannon’s representation and Fourier polygon representation of noisy communication channel	21
Figure 18 U matrix, SumU matrix, and Inverse FFT of U matrix for $u7,6p, t$	22
Figure 19 Rows of the U matrix of $u7,6p, t$	27
Figure 20 Fourier matrix and U matrix.....	28
Figure 21 FFT and U transform of a harmonic signal	29
Figure 22 FFT and U transform of a harmonic signal with more samples.....	29
Figure 23 FFT and U transform of a random signal	30
Figure 24 FFT and U transform of a random signal with more samples.....	30
Figure 25 Deriving Riemann Zeta Matrix from U matrix	31
Figure 26 Riemann Zeta matrix	32
Figure 27 The side and top views of a compound system of communication channels...	32
Figure 28 Comparison of FFT, U transform, and Zeta transform	33
Figure 29 An example of amplitwist	43
Figure 30 A simple model of particle movement model	44
Figure 31 Polygon geometry and probability amplitude	45
Figure 32 Amplitwist and “particle movement” model.....	47
Figure 33 Corresponding polygon geometry example.....	48
Figure 34 The Blotch sphere representation of a single qubit	54
Figure 35 The CNOT quantum gate	58
Figure 36 Entangled states in a particle chain	62
Figure 37 Fourier polygon with 6 vertices.....	63
Figure 38 Example of entangled states	64

Figure 39 Irregular polygon formed by permutation matrix.....	65
Figure 40 The regular polygons used to construct irregular polygons.	66
Figure 41 U matrix and entangled quantum states.....	67
Figure 42 Shannon’s representation and Fourier polygon representation of noisy communication channel	68
Figure 43 Relationship of probability amplitude and communication channel	69
Figure 44 A multiple-input multiple-output channel	70
Figure 45 U matrix as a quantum register.....	72
Figure 46 The noisy communication channel framework for general system modeling..	74
Figure 47 The extended noisy communication channel framework.....	75
Figure 48 Feynman diagram for photon and electron.....	76
Figure 49 Light partial reflection experiment with one surface	77
Figure 50 Light partial reflection experiment with two surfaces.....	77
Figure 51 Probability representation of arrow length.....	78
Figure 52 Partial reflection event and the probability amplitude	79
Figure 53 The final probability of photons reflected by two surfaces	80
Figure 54 The probability change based on the thickness of the glass for reflection	81
Figure 55 Addition of multiple probability amplitudes	82
Figure 56 Probability amplitudes of 6 surfaces	83
Figure 57 Irregular polygon formed by permutation matrix with 7 elements	84
Figure 58 Regular polygons used to construct the irregular polygon with 7 vertices	85
Figure 59 Richard Feynman’s probability amplitude and communication channel “probability amplitude”	86

Figure 60 Polygon geometry and probability amplitude	87
Figure 61 Amplitwist and multiple probability amplitudes.....	90
Figure 62 Amplitwist example with multiple vectors.....	90
Figure 63 Amplitwist analysis of QED.....	91
Figure 64 Relationship of probability amplitude and communication channel.....	94
Figure 65 Fourier polygons and multiple probability amplitudes	96
Figure 66 Paths of reflecting light	97
Figure 67 Harmonic signal and FFT, U transform, Zeta transform signal in the frequency domain.....	99
Figure 68 Damping signal and FFT, U transform in the frequency domain.....	100
Figure 69 Polar form of a complex number in the complex plane.	108
Figure 70 Roots of unity for $n = 1, 2, \dots, 9$	111
Figure 71. Regular polygon generated by 7×7 Fourier Matrix.	117
Figure 72 Classical two-valued logic.....	123
Figure 73 Three-valued logic.....	123
Figure 74 N-valued logic	124
Figure 75 Communication error representation of noisy communication channel.....	125
Figure 76 Example state sets of the error content graph representations	128
Figure 77 Roots of unity of a 7 symbol communication channel.....	130
Figure 78 The communication channel framework and its connections to common analysis.....	131
Figure 79 Row drawing of $u(7,2)$	136
Figure 80 Row drawing of $u(7,3)$	141

Figure 81 Row drawing of $u(7,4)$	145
Figure 82 Row drawing of $u(7,5)$	150
Figure 83 Row drawing of $u(7,6)$	154

LIST OF ABBREVIATIONS

QED	Quantum Electrodynamics
FPGA	Field Programmable Gate Array
FFT	Fast Fourier Transform

1. INTRODUCTION

“System” is a very complex concept, but it could be simplified as follows: a system is composed of two parts: a target and the associated processes to reach the target [1,4]. People usually use modeling to study complex systems. Mathematical models of real world applications help us understand the phenomena around us in a more effective way. More specifically, a model is a simpler realization or idealization of more complex real world entities in the form of a physical miniature, mathematical equations, or a computer simulation of a real world system. The purpose of such models is to increase one’s understanding of real world systems, to help analyze the system, or to reveal some new information about it [2,3]. In addition, it could provide a computation tool to help people in study and research.

Traditionally, the natural or artificial phenomena had been mathematically modeled with either wave or particle equations. Within this study, however, we approach modeling from a new perspective---a communication-based perspective. Based on Dr. Murat M. Tanik’s synthesis [1,4], the modeling of natural or artificial phenomena could be classified into three categories: material-based modeling, energy-based modeling, and communication-based modeling, as shown in Figure 1. The first category of modeling has been very well developed, and not much space is left for study; the second category is also well developed, though not as much as the first category; compared to the first two categories, communication-based modeling is not as well developed. Not much work has

been done, and a lot of space, possibilities and potentials remain open [1,4]. The core of our research is to approach modeling using the communication paradigm, and to apply the communication-based modeling technique to quantum computing.

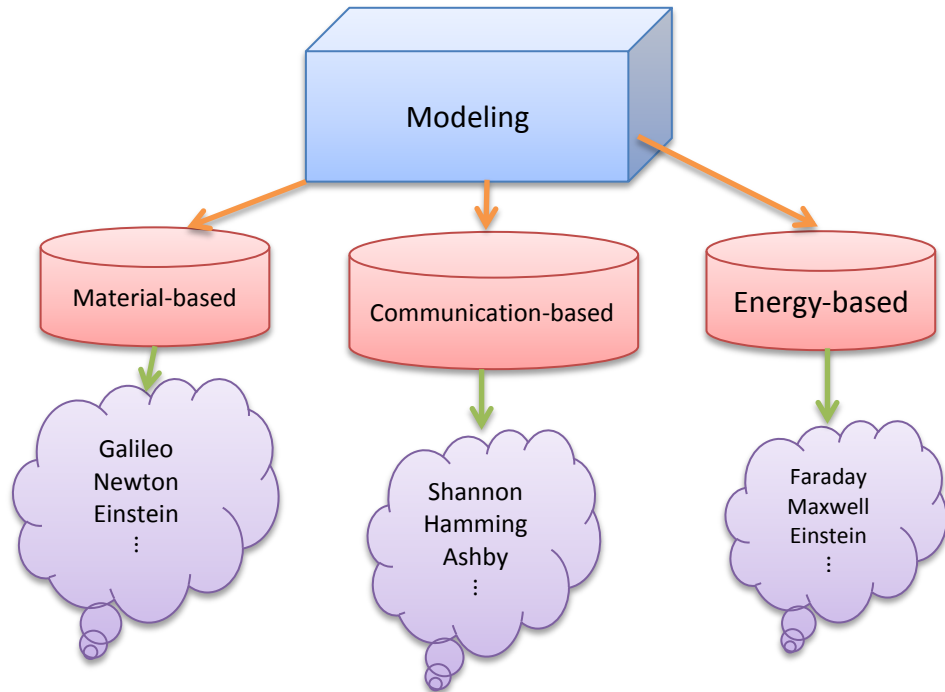


Figure 1 Modeling paradigms

The communication-based model is an original idea which is beyond particle or wave equations [1,4]. The communication-based model is a universal model and can potentially behave at the limiting end points like particle or wave equations as the energy of the system being modeled varies, as shown in Figure 2 [1,4].

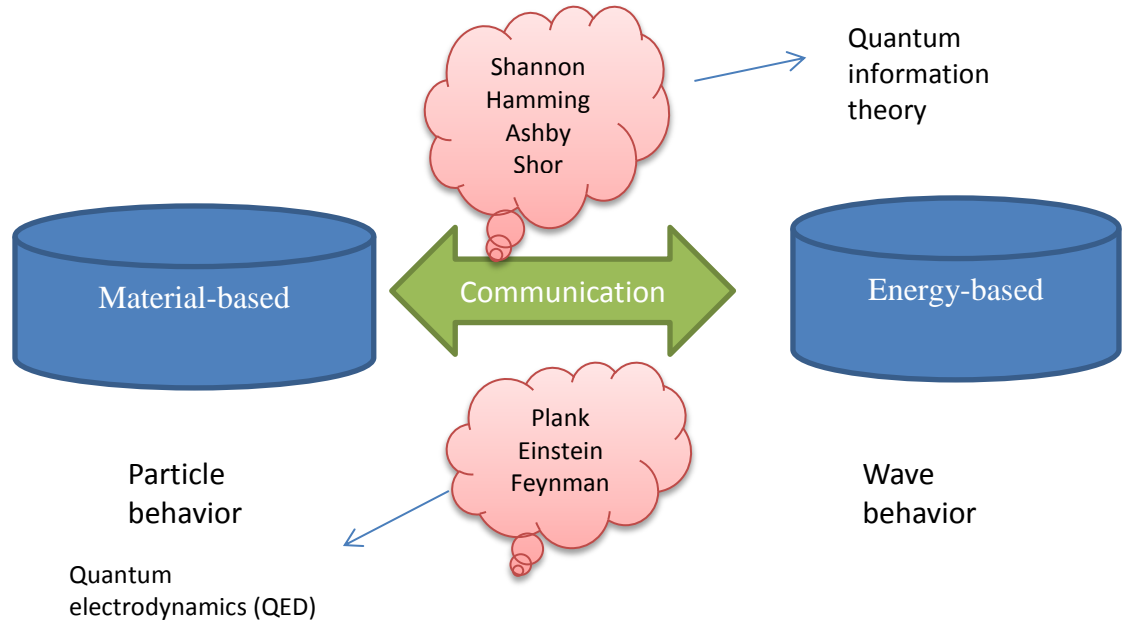


Figure 2 Communication-based modeling

Due to the rapid change of technology in today's environment, interdisciplinary research has played an increasing role to deal with the problems brought by such changes. Moore's law has been the reference to the rate of technological progress for computer chips during the past decades. There are indications that Moore's law is reaching its limit: the switch size cannot be further scaled due to the physical limitations [5-8]. Suppose that alternative materials could be found to take the place of silicon for chip design, and we still follow Moore's law in the future; then computers would be the size of an atom around the year 2050 [9,10]. "Strange" phenomena happen within atomic-sized systems since atomic systems function under the laws of quantum theory. On the other hand, quantum computing is also considered to be a promising technology for some specific types of computing purposes. However quantum computing is still a young research field, and modeling of quantum computing could help study and research of this field. Thus

communication based modeling of quantum computing is a promising modeling technique.

Keeping the multidisciplinary demand of system modeling and analysis in mind; we proposed a generalized noisy communication channel approach (GNCom) for quantum computing building on the noisy communication channel model developed in our research group [11-33]. In this dissertation, following the research program of our research group, we cover a broad variety of research fields: quantum computing, information theory, Fourier analysis, complex analysis, combinatorial analysis, Riemann Zeta analysis, and quantum electrodynamics. [13-16,19-33]

For the convenience of the reader, the necessary background on these concepts is given in the appendices.

Contributions

This dissertation develops a generalized noisy communication channel approach (GNCom) and develops a quantum computing model using the generalized noisy communication channel approach. The GNCom approach includes new matrices and transformations, which are utilized to represent quantum computing with the noisy communication channel. The proposed GNCom approach is also applied to quantum computing as a universal quantum register to replace quantum gates. GNCom further improved the noisy communication channel framework [11,12] by connecting quantum electrodynamics (QED). The framework is also improved by new techniques such as Amplitwist and multivalued logic. Based on GNCom and QED, we further proposed a quantum computer design technique. The contributions are listed below:

- 1) This dissertation develops a generalized noisy communication channel approach (GNCom) building on the noisy communication channel model developed in our research group.
- 2) This dissertation develops the noisy communication channel representation of quantum computing.
- 3) The GNCom approach extended the noisy communication channel framework by connecting QED.

2. THE GENERALIZED NOISY COMMUNICATION CHANNEL

APPROACH

The generalized noisy communication channel approach is developed based on Claude E. Shannon's communication paradigm [34, 35]. Our group had intensively researched the noisy communication channel and had successfully applied it to software component modeling [17,19] and general system modeling [11]. In this dissertation, we develop a generalized noisy communication channel framework for quantum computing.

The noisy communication could be described with a bipartite graph [34]. Based on the communication error probability, it could be represented as an error content graph either in a regular polygon or an irregular polygon shape [36]. We proposed a numerical algorithm based on Fourier series to connect the noisy communication channel with the polygons [15,31]. We developed the generalized noisy communication channel to represent quantum computing by introducing Fourier series to each edge of the polygon or to each communication channel, thus making it powerful to model quantum computing.

Shannon's noisy communication channel paradigm

A communication channel is a medium through which information can be sent or conveyed from the source (transmitter) to the sink (receiver). However due to noise, the receiver may not receive the original information the transmitter sends. For example, a

symbol received differs from a symbol that has been sent. Claude E. Shannon proposed his famous zero error capacity channel theorem in his work [34]. An example of a typical noisy communication channel is shown in Figure 3.

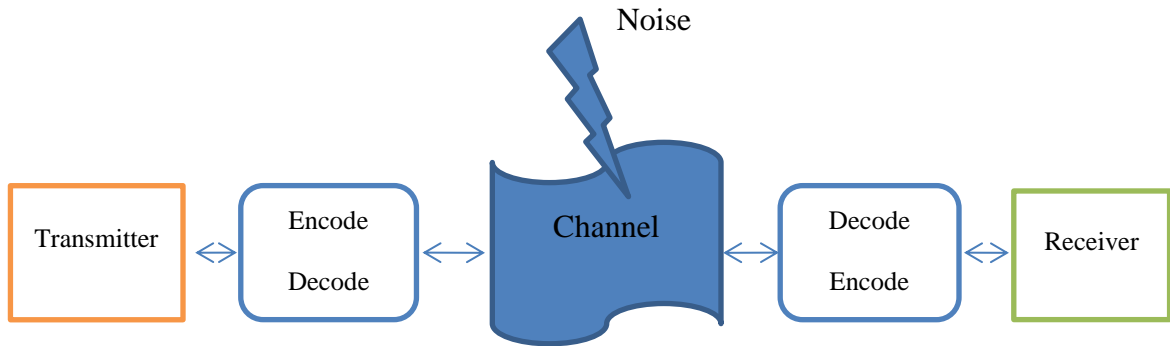


Figure 3 An example of communication channel

The noisy communication channel can be depicted as in Figure 4. The left side lower case letters represent the source symbols, and the right side upper case letters represent the received symbols. If the receiver side letter is the same as the source letter, then there is no error during transmissions, but it is possible that a source symbol can be received as different symbols.

The symbols sent and received through the communication channel can be expressed in detail as shown in Figure 4. Symbols a, b, c, d, and e are the 5 signals that need to be transmitted through a finite, discrete, and memory-less communication channel. The receiver can interpret each of these signals in two different ways. For example, a can be received as either A or B, b can be received as either B or C, etc.

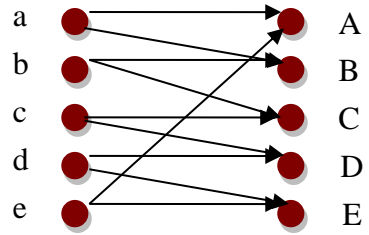


Figure 4 A noisy communication channel interpretation

Shannon represents the noisy communication channel in terms of error probabilities during communication [35], thus the channel capacity could be derived based on error probabilities. Figure 5 shows Shannon’s representation of the noisy communication channel. In this representation, the vertices represent the input and output signals, and the arrows show the probability of the signals received at the receiver side. For example for the first two vertices, a is received as A with probability P_{aa} , and a is received as B with probability P_{ab} . For a given input signal, the sum of all the probabilities should be equal to 1.

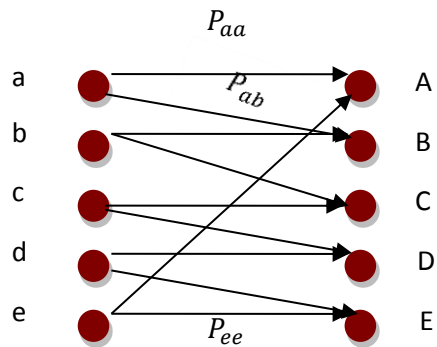


Figure 5 Shannon’s representation of noisy communication channel

Noisy communication polygon representation

The noisy communication channel could be further represented as an error content graph [36]. Graphs of this type provide a visual representation of the transmitter-receiver connections specified by the communication error probability representation.

By observing Figure 5, we can see that the error pairs of the symbols are $a \rightarrow B$, $b \rightarrow C$, $c \rightarrow D$, $d \rightarrow E$, and $e \rightarrow A$, which produces the error content graph shown in Figure 6.

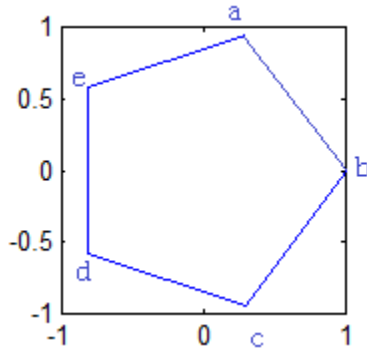


Figure 6 Error content graph representation of noisy communication channel of Figure 5

Here we can see that the communication channel can be represented by polygons in the form of error content graph (regular polygons in this case). For other kinds of noisy communication channels, the error content graph representation could be non-convex irregular polygons [15]. Figure 7 gives an example of a noisy communication channel and its corresponding non-convex irregular polygon.

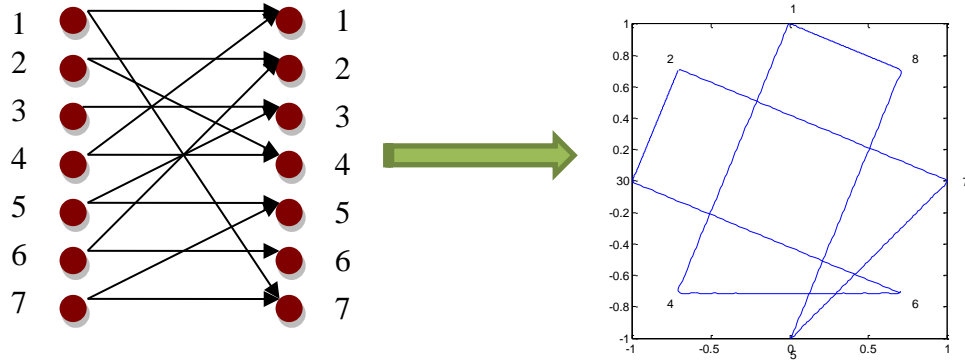


Figure 7 A communication channel with non-convex irregular polygon error content graph representation

U matrix

Robert discussed the relationship of Fourier series and polygons [37]. Fourier representation of noisy communication channel had been studied by our group as well [20, 22-24]. Following our research group's work, we further explored the U matrix, which is a complex matrix obtained by introducing Fourier series to the noisy communication channel. The U matrix is the basis of the generalized noisy communication channel approach. We will cover the Fourier representation of polygons first, and then introduce U matrix features and new transforms.

Fourier series representation of regular polygon

The Fourier series of a regular polygon is given as

$$F_n(t) = \sum_{k \equiv l(n)} \frac{e^{jkt}}{k^2} \quad (1.1)$$

where $k = 1 + nl$ with $l \in [-p, p]$, or $l = -p, -p + 1, \dots, p - 1, p$, and $l, p \in Z$ [37].

By expanding equation (1.1), we introduce an algorithm to construct regular polygons using Fourier series. An example of the expansion is shown below.

For $p = 1, n = 5, l = -1, 0, 1, k = 1 + nl = 1 + 5l = -4, 1, 6$.

$$F_5(t) = \sum_{k \equiv l(5)} \frac{e^{jkt}}{k^2} = \frac{e^{j(-4)t}}{16} + \frac{e^{j(1)t}}{1} + \frac{e^{j(6)t}}{36}$$

For $p = 2, n = 5, l = -2, -1, 0, 1, 2$,

$$k = 1 + nl = 1 + 5l = -9, -4, 1, 6, 11.$$

$$F_5(t) = \sum_{k \equiv l(5)} \frac{e^{jkt}}{k^2} = \frac{e^{j(-9)t}}{81} + \frac{e^{j(-4)t}}{16} + \frac{e^{j(1)t}}{1} + \frac{e^{j(6)t}}{36} + \frac{e^{j(11)t}}{121}$$

The corresponding polygons are shown in Figure 8.

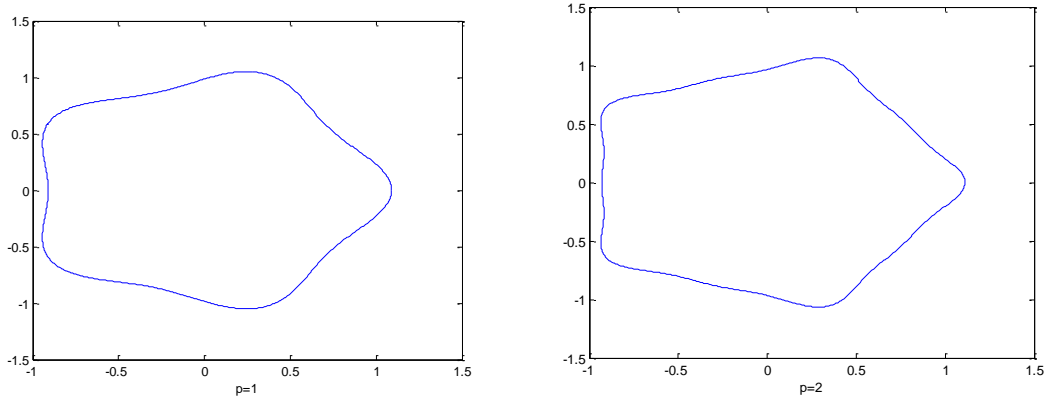


Figure 8 Fourier series for polygons with $p=1$, and $p=2$.

It is highly observable that as p increases, the number of the elements increases as does the resolution. The following parameters are defined to represent the polygon with

Fourier series:

- ❖ $q \in Z^+$ Corresponds to the jump step value of uniform polygon, more specifically, defining the destination point to go from the present point.
- ❖ $n \in Z^+$ Corresponds to the number of vertices of a uniform polygon.

- ❖ $l \in Z^+$ Determines the interval $[-p, p]$ in which the Fourier series will be calculated.
- ❖ $N \in Z^+$ is the sampling factor for the numerical algorithms, and it determines the number of samples $t \in [t_0, 2\pi + t_0]$.

Introducing all the parameters of the algorithm, we obtain our new equation,

$$u(l, t) = \sum_{k \equiv l(n)} \frac{e^{j(q+nl)t_r}}{(q + nl)^2} \quad (1.2)$$

where $t_r = t_0, t_1, \dots, t_{N-1} = (2\pi + t_0)$, and $l = -p, -p + 1, \dots - 1, 0, 1, \dots, p - 1, p$.

Since the resulting Fourier series calculation is the sum of each Fourier term at time t_0 , the equation (1.2) can be rewritten as

$$u(l, t) = \sum_{l=-p}^p \frac{e^{j(q+nl)t_r}}{(q + nl)^2} \quad (1.3)$$

Then an algorithm would be expected to realize the following equation:

$$u(l, t) = \sum_{l=-p}^p \frac{e^{j(q+nl)t}}{(q + nl)^2} \quad (1.4)$$

The Fourier series matrix defined by (1.4) has the following form [22-24]. Note that the parameter l denotes the rows of the resulting matrix.

$$u_{n,q}(p, t) = \begin{bmatrix} \frac{e^{j[q+n(0-p)]t}}{[q+n(0-p)]^2} \\ \frac{e^{j[q+n(1-p)]t}}{[q+n(1-p)]^2} \\ \frac{e^{j[q+n(2-p)]t}}{[q+n(2-p)]^2} \\ \vdots \\ \frac{e^{j[q+n(-1)]t}}{[q+n(-1)]^2} \\ \frac{e^{j[q+n(0)]t}}{[q+n(0)]^2} \\ \frac{e^{j[q+n(1)]t}}{[q+n(1)]^2} \\ \vdots \\ \frac{e^{j[q+n(p-2)]t}}{[q+n(p-2)]^2} \\ \frac{e^{j[q+n(p-1)]t}}{[q+n(p-1)]^2} \\ \frac{e^{j[q+np]t}}{[q+np]^2} \end{bmatrix} \quad (1.5)$$

The matrix in (1.5) has a row for each value of $l \in [-p, p]$ with a total of $2p + 1$ rows.

The matrix $u_{n,q}(p, t)$ will have N columns, each corresponding to an instant of time.

We can rewrite the matrix explicitly as shown in Figure 9.

$$u_{n,q}(p, t) = \begin{bmatrix} \frac{e^{j[q+n(-p-p)]t_1}}{[q+n(-p-p)]^2} & \frac{e^{j[q+n(-p-p)]t_{r-1}}}{[q+n(-p-p)]^2} & \frac{e^{j[q+n(-p-p)]t_r}}{[q+n(-p-p)]^2} \\ \frac{e^{j[q+n(-p+1-p)]t_1}}{[q+n(-p+1-p)]^2} & \frac{e^{j[q+n(-p+1-p)]t_{r-1}}}{[q+n(-p+1-p)]^2} & \frac{e^{j[q+n(-p+1-p)]t_r}}{[q+n(-p+1-p)]^2} \\ \vdots & \vdots & \vdots \\ \frac{e^{j[q+n(-1-p)]t_1}}{[q+n(-1-p)]^2} & \frac{e^{j[q+n(-1-p)]t_{r-1}}}{[q+n(-1-p)]^2} & \frac{e^{j[q+n(-1-p)]t_r}}{[q+n(-1-p)]^2} \\ \frac{e^{j[q+n(0-p)]t_1}}{[q+n(0-p)]^2} & \dots & \frac{e^{j[q+n(0-p)]t_r}}{[q+n(0-p)]^2} \\ \frac{e^{j[q+n(1-p)]t_1}}{[q+n(1-p)]^2} & \frac{e^{j[q+n(1-p)]t_{r-1}}}{[q+n(1-p)]^2} & \frac{e^{j[q+n(1-p)]t_r}}{[q+n(1-p)]^2} \\ \vdots & \vdots & \vdots \\ \frac{e^{j[q+n(p-1-p)]t_1}}{[q+n(p-1-p)]^2} & \frac{e^{j[q+n(p-1-p)]t_{r-1}}}{[q+n(p-1-p)]^2} & \frac{e^{j[q+n(p-1-p)]t_r}}{[q+n(p-1-p)]^2} \\ \frac{e^{j[q+n(p-p)]t_1}}{[q+n(p-p)]^2} & \frac{e^{j[q+n(p-p)]t_{r-1}}}{[q+n(p-p)]^2} & \frac{e^{j[q+n(p-p)]t_r}}{[q+n(p-p)]^2} \end{bmatrix}$$

Figure 9 Explicitly written U matrix

Since the resulting Fourier series calculation is the sum of each Fourier term for every time t_r , the equation for $u_{n,q}(p, t)$ could be rewritten as

$$u_{n,q}(p, t) = \sum_{l=-p}^p \frac{e^{j(q+nl)t}}{(q+nl)^2} \quad (1.6)$$

By changing the limits of the sum, we get

$$u_{n,q}(p, t) = \sum_{l=0}^{2p} \frac{e^{j[q+n(l-p)]t}}{[q+n(l-p)]^2} \quad (1.7)$$

We implemented the numerical algorithm using Matlab.

The polygons of $q = 2, n = 5, p = 1, 2, \dots, 10$ are shown in Figure 10.

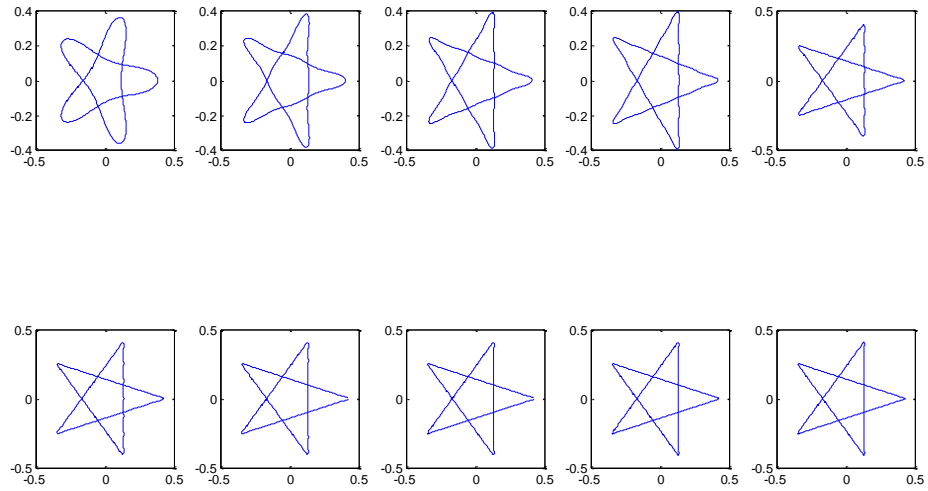


Figure 10 Evolution of 5 by 2 polygons with increasing resolution

The polygons of $q = 4, n = 11, p = 1, 2, \dots, 10$ are shown in Figure 11.

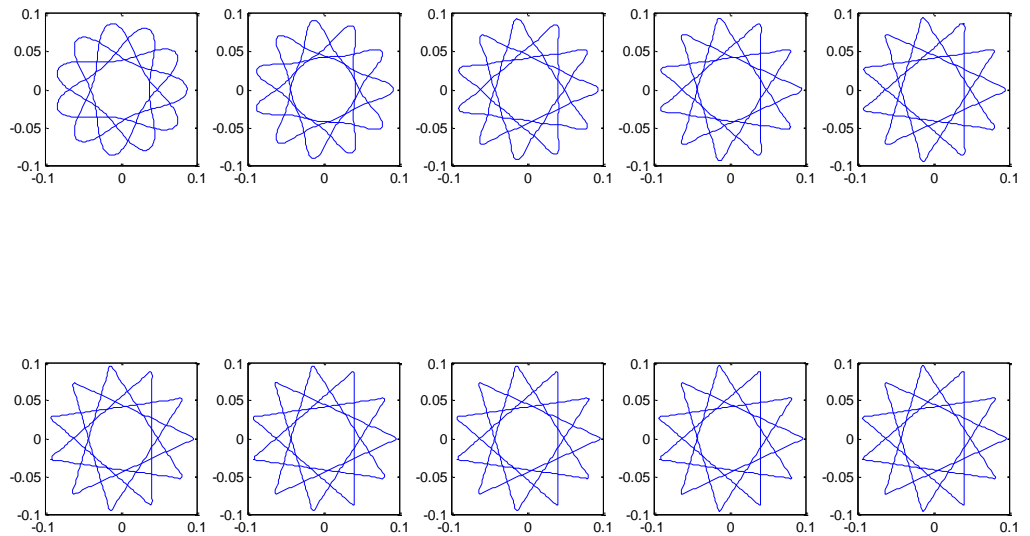


Figure 11 Evolution of 11 by 4 polygons with increasing resolution

Figure 12 shows the polygons for $u_{6,1}(p, t)$, $u_{7,3}(p, t)$, $u_{8,2}(p, t)$, and $u_{9,3}(p, t)$. The resolution parameter for all the drawings is constant ($p=10$).

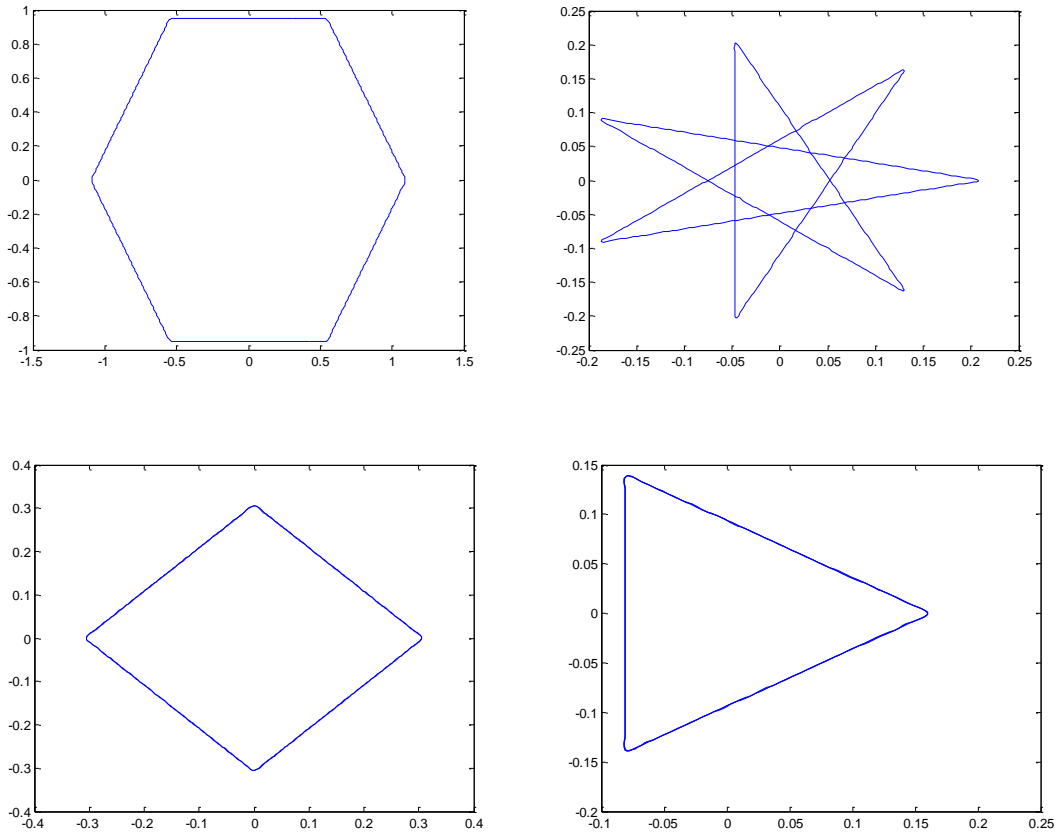


Figure 12 Some Polygon examples for $u_{6,1}(p, t)$, $u_{7,3}(p, t)$, $u_{8,2}(p, t)$, and $u_{9,3}(p, t)$

Note that when n and q are not relatively prime, numbers of vertices in the resulting polygon are not equal to n as shown in Figure 12 for $u_{8,2}(p, t)$, and $u_{9,3}(p, t)$.

Fourier series of irregular polygon

The Fourier series representation of regular polygons is very powerful and is easy to use, but not all the real world problems could be represented by regular polygons [22].

Therefore, we make further efforts to extend the regular polygon Fourier series representation to irregular polygon representation.

Let $u_{n,q}(p, t)$ denotes a regular polygon that has n vertices and a jump speed of q .

Time $t \in 2\pi$ and

$$0 \leq t_0 < t_1 < \dots < t_i < t_{i+1} < \dots < t_{n-1} < t_n = t_0 + 2\pi$$

The function $u_{n,q}(p, t)$ is linear in every time interval $[t_i, t_{i+1}]$.

Correspondingly, let $\varphi_{n,[q]}([p], t)$ denote an irregular polygon that has n vertices.

Instead of having a constant jump speed, the jump speed q is a vector that has n elements.

In addition, $[p]$ does not have to be constant for each side of the irregular polygon.

Example 1. Let $q = [c, d, e, \dots, h, q, d]$, the length of the jump speed vector q is n . An irregular polygon $\varphi_{n,[q]}([p], t)$ is composed of pieces of the regular polygons:

$$u_{n,c}([p_1], t), u_{n,d}([p_1], t), u_{n,e}([p_1], t), \dots, u_{n,h}([p_1], t), u_{n,q}([p_1], t), u_{n,d}([p_1], t).$$

Thus, the irregular polygon $\varphi_{n,[q]}([p], t)$ is defined as:

$$\varphi_{n,[q]}([p], t) = \sum_{i=0}^{n-1} \tau_i u_{n,q_i}([p_i], t) \quad \text{where } t_i \leq t < t_{i+1} \quad (1.8)$$

with $t_0 = 0$, and $t_n = 2\pi$. The overall time span $t \in 2\pi$ is divided into n equal intervals. τ_i in (1.8) is a transformation, which rotates the regular polygon $u_{n,q_i}([p], t)$ counterclockwise by an amount of $2\pi q_i/n$, and scales the inscribed circle radius to be 1, which is an amplitwist operation referred to in Appendix 2. The scale is important, since the jump speed and number of vertices of the original regular polygons vary a lot. It is difficult to define what point on the complex plane maps to what time. So the process for constructing the irregular polygon involves starting from time 0 for all the regular polygons, rotating them, scaling them and extracting pieces needed. The rotation of

regular polygons enables using the first side of the corresponding regular polygon for constructing the irregular polygon. The final irregular polygon is made of different Fourier series concatenated one after another for different time intervals. An example is shown to illustrate how an irregular polygon was constructed with Fourier series.

Example 2. Give a permutation matrix which forms an irregular polygon

$$\begin{bmatrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \\ 2 & 7 & 5 & 8 & 1 & 4 & 6 & 3 \end{bmatrix}$$

The irregular polygon representation is shown in Figure 13.

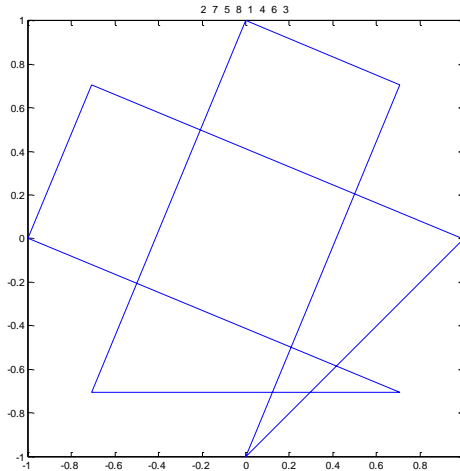


Figure 13 Irregular polygon formed by permutation matrix of Example 2.

The jump “speed” for the line going from 2 to 7 is $7-2=5$. Similarly, the “speed” for the line going from 7 to 5 is $5-7=-2$. Since the direction to plot the irregular polygon is defined as counterclockwise, the negative speed corresponds to the direction moving clockwise with the specified amount. In order to make it clear, we will use mod n for the “speed vector.” The speed vector for this example is $[5, -2, 3, -7, 3, 2, -3, -1]$. After the mod n operation, the speed vector turns out to be $[5, 6, 3, 1, 3, 2, 5, 7]$. This vector

corresponds to the jump speed of the irregular polygons, which indicates that the irregular polygon is constructed from pieces out of various regular polygons:

$$u_{8,5}(t), u_{8,6}(t), u_{8,3}(t), u_{8,1}(t), u_{8,3}(t), u_{8,2}(t), u_{8,5}(t), u_{8,7}(t)$$

The resolution parameters p_i are all set to 10. Figure 14 shows the Fourier series representation regular polygons used to construct the irregular polygon.

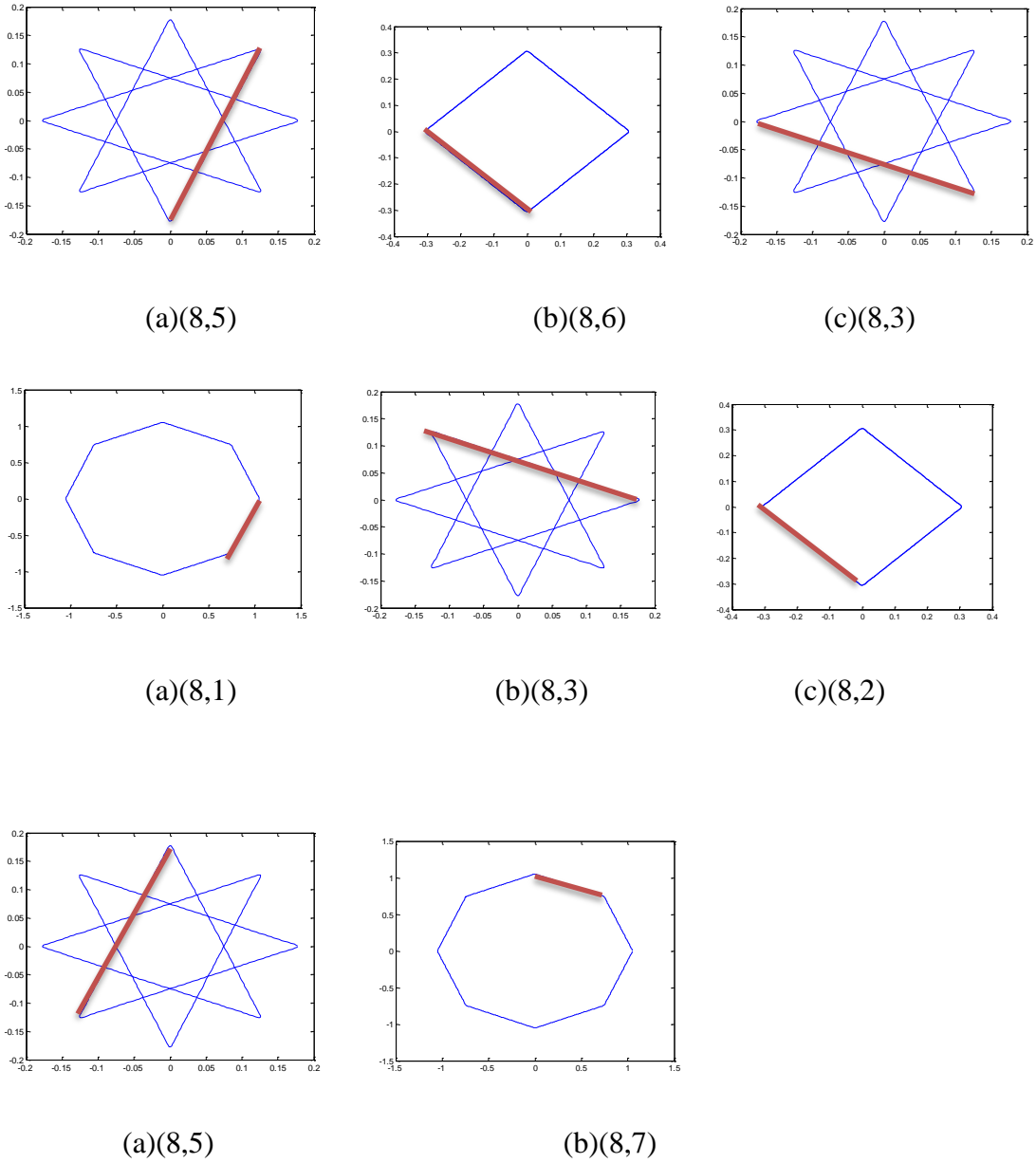


Figure 14 The regular polygons used to construct irregular polygons.

The detailed construction process of the irregular polygon by taking pieces from regular polygons is shown in Figure 15.

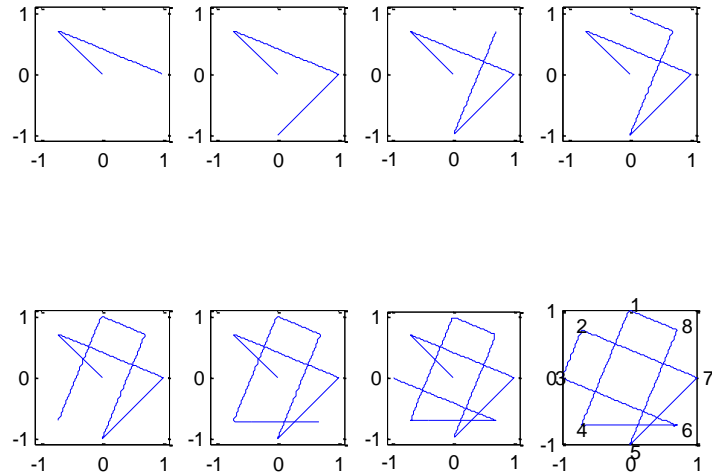


Figure 15 The process of the irregular polygon formed by pieces taken from regular polygons

The Fourier series representation of irregular polygons of $n=8$ is shown in Figure 16.

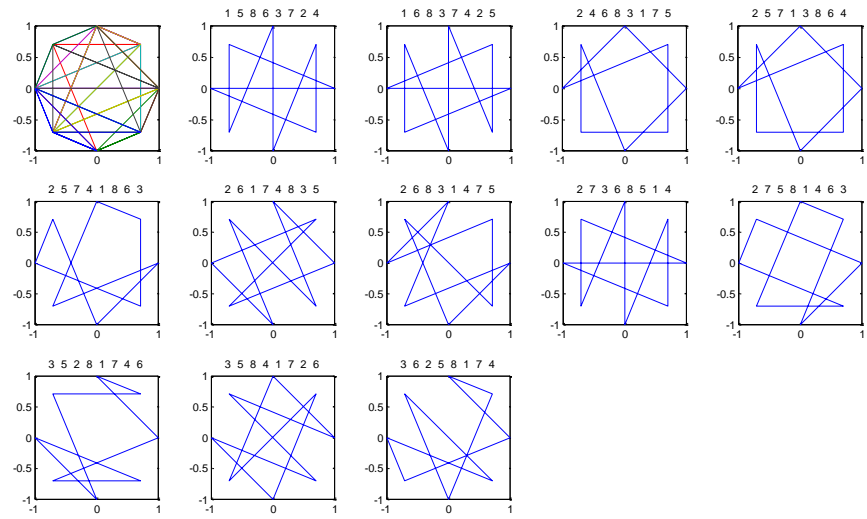


Figure 16 Fourier series representation of irregular polygons of 8 vertices

Based on our analysis, there exist regular or irregular Fourier polygon representations for the noisy communication channel in the form of error content graph [23-25]. For the noisy communication channel in permutation form $\begin{bmatrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \\ 2 & 7 & 5 & 8 & 1 & 4 & 6 & 3 \end{bmatrix}$, the corresponding Shannon's representation of the noisy communication channel and the Fourier polygon representation are shown in Figure 17.

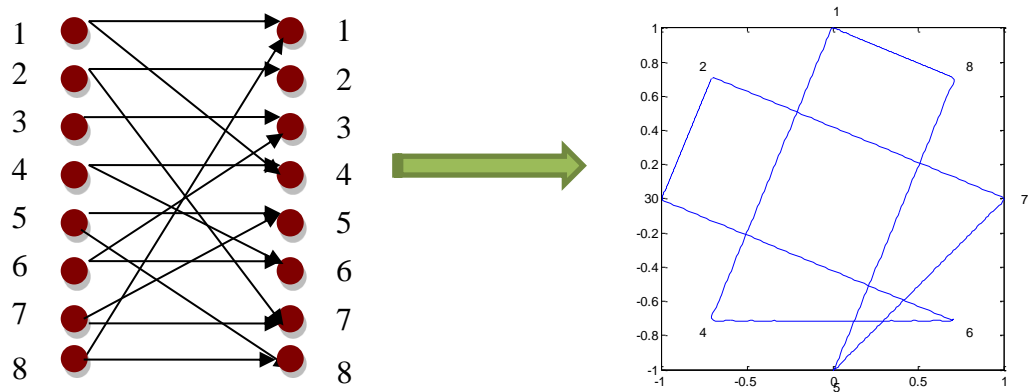


Figure 17 Shannon's representation and Fourier polygon representation of noisy communication channel

We further visually investigated the Fourier series numerical algorithm

$$u_{n,q}(p, t) = \sum_{l=-p}^p \frac{e^{j(q+nl)t}}{(q + nl)^2},$$

especially the Fourier series matrix $u_{n,q}(p, t)$. An example plot of the U matrix, SumU matrix and inverse FFT of U matrix is shown in Figure 18.

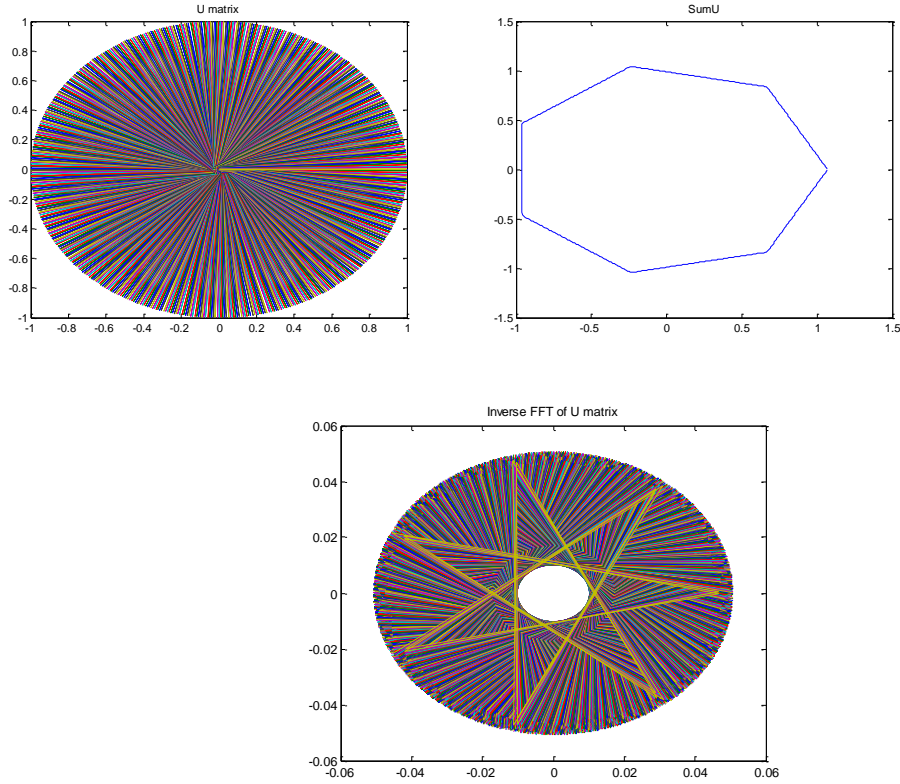


Figure 18 U matrix, SumU matrix, and Inverse FFT of U matrix for $u_{7,6}(p, t)$

Besides changing n and q , which corresponds to number of polygon vertices and jump step, we investigate further by tuning the parameters p and t . The relationship of the parameters, the U matrix and the polygon is shown in Table 1.

Table 1 Relationship of equation parameters and U matrix

Parameters of numerical algorithm	U matrix	Polygon
n		Number of vertices
q		Jump step
l	Rows of the U matrix	Summation interval
p		Resolution
t	Columns of the U matrix	Sampling factor

Table 2 shows an example of the polygon plot of $n=7$ and $q=1$. The results of changing p from 1 to 10 are shown in Table 2.

Table 2 p changes from 1 to 10 for n=7 and q=1

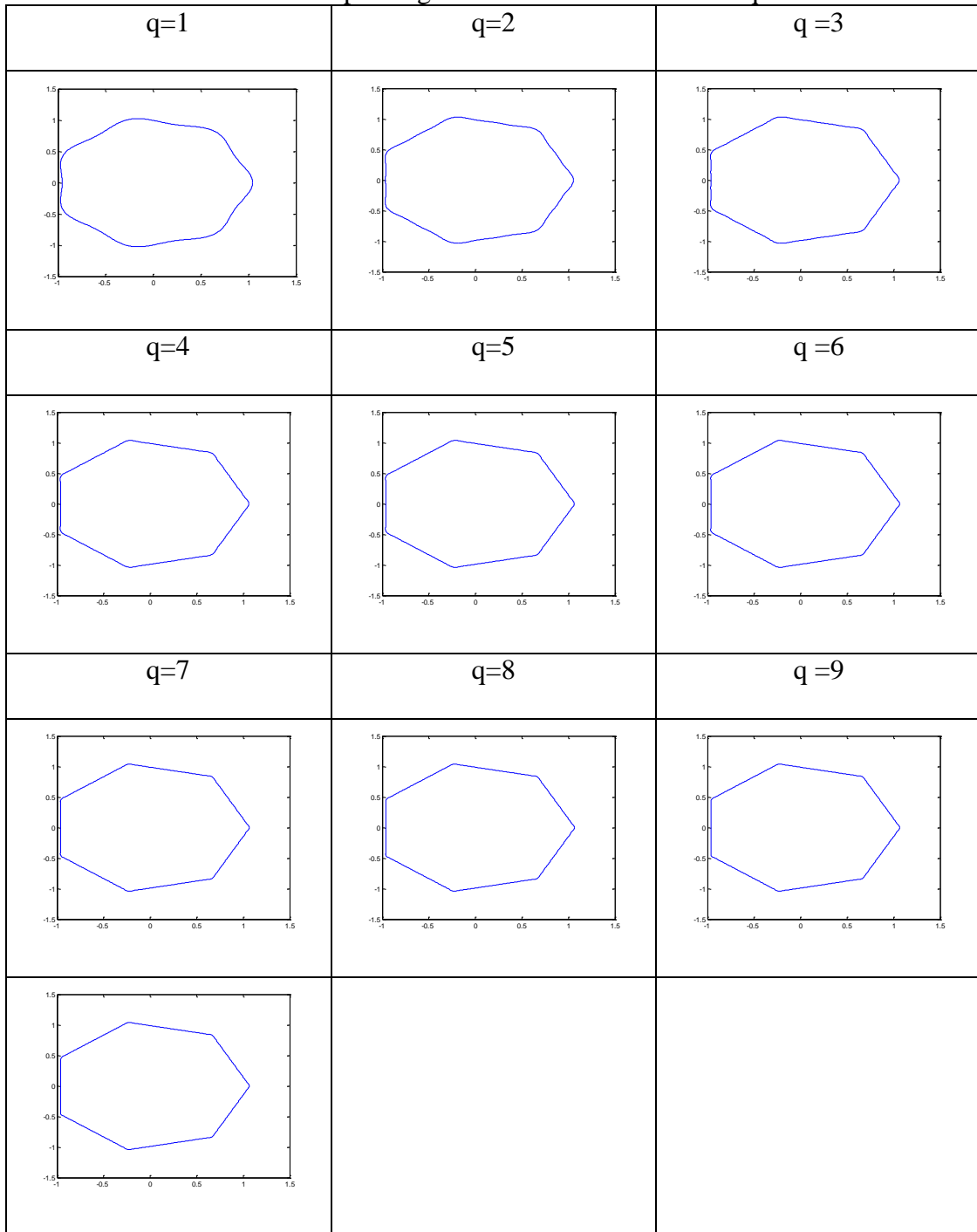
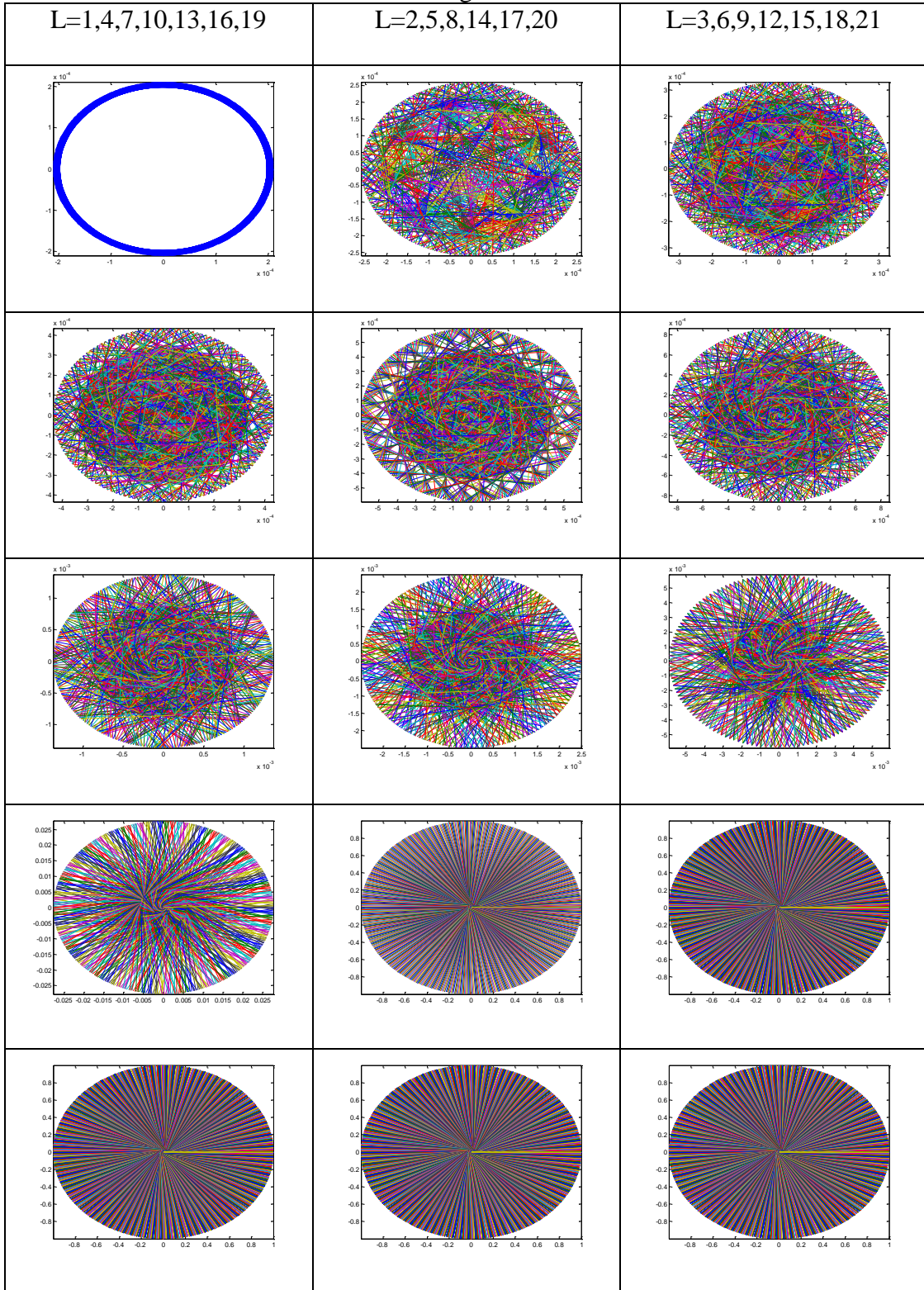
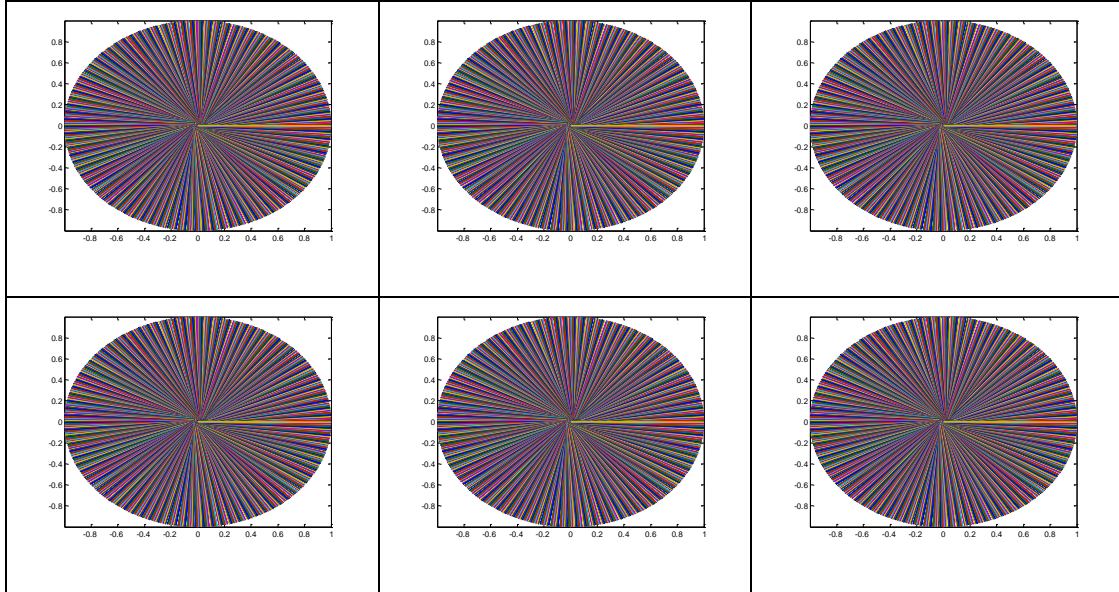


Table 3 shows the plots of the U matrix changing as row number increases or as p changes from -10 to 10. The first plot corresponds to U matrix with only one row with p=-10. The second row corresponds to the U matrix with two rows with p=-10, -9.

Table 3 U matrix changes as row number increases

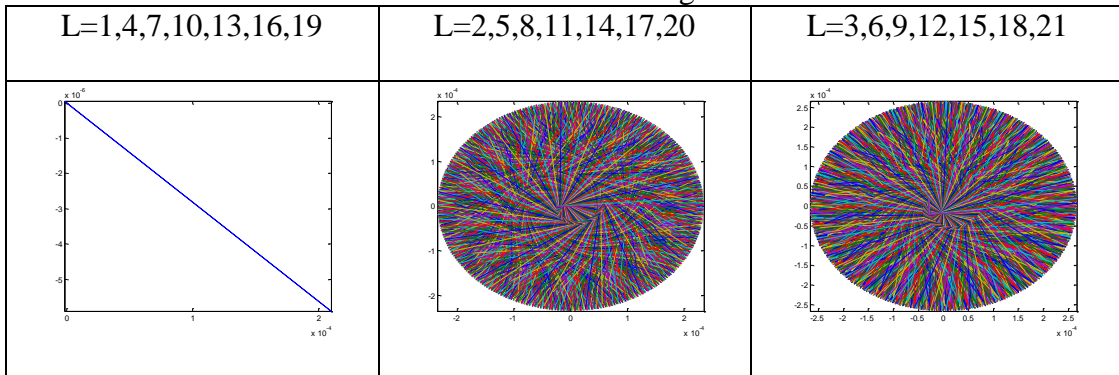


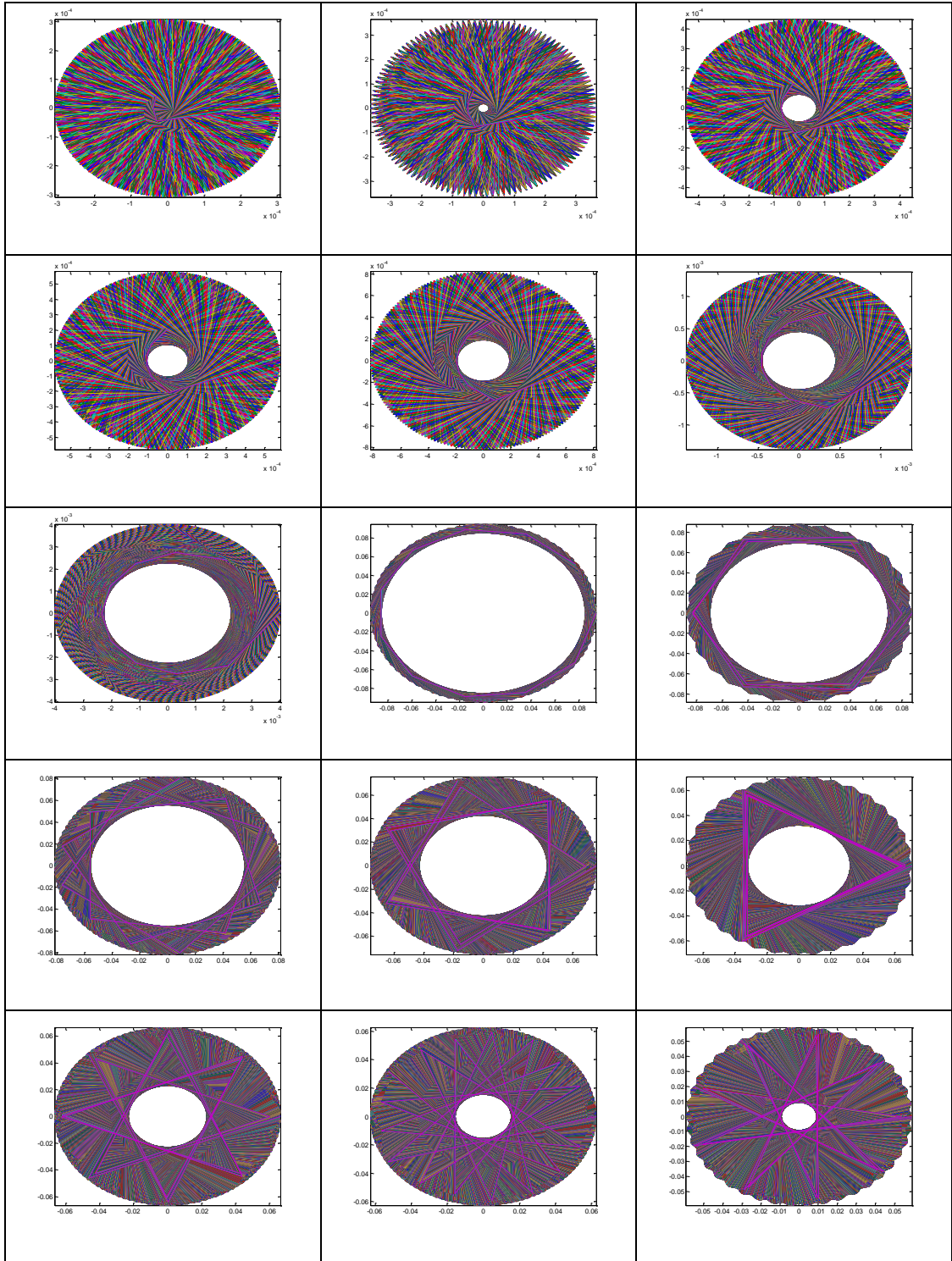


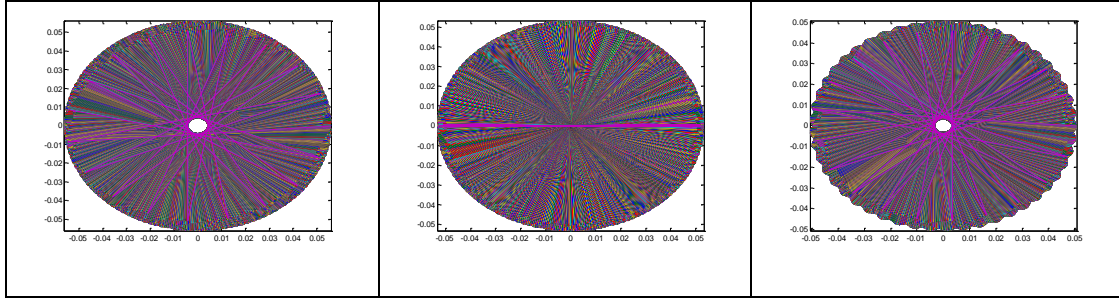
These plots show the expansion of the U matrix by row numbers. For example, $L=1$ corresponds to a U matrix with only one row; $L=2$ corresponds to a U matrix with two rows; $L=21$ corresponds to the U matrix with 21 rows, since $P=10$ in this example, so $l \in [-10, 10]$, which indicated 21 rows [31].

The inverse Fourier transform of the U matrix is shown in Table 4.

Table 4 Inverse Fourier transform of U matrix changes as row number increases







If we draw each row of the U matrix, we could obtain Figure 19.

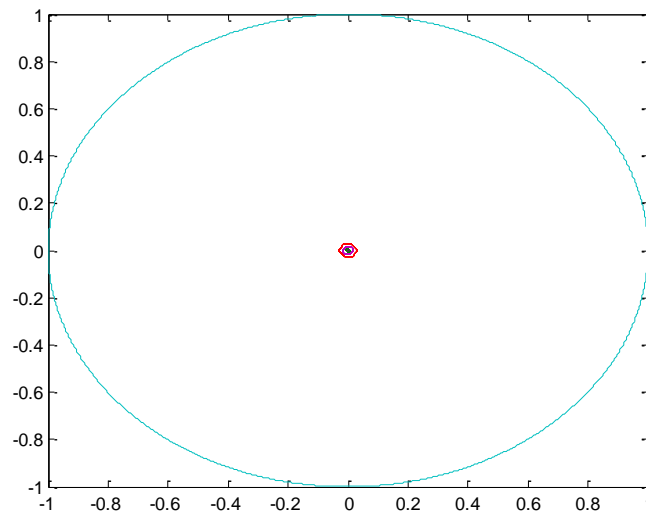


Figure 19 Rows of the U matrix of $u_{7,6}(p, t)$

The other plots of U matrix with $n=7$, $q=2,3,4,5,6$ are shown in Appendix C.

U transform

Let us recall the Fourier Matrix and Fast Fourier Transform (FFT): for a given signal S , the FFT process of the signal could be considered as the multiplication of the Fourier Matrix and the signal, which is indicated as

$$S \rightarrow \text{FFT}(S) \quad \longleftrightarrow \quad S \rightarrow \text{Fourier Matrix} \times S$$

We propose a new terminology U transform based on the U matrix. The U transform is simply defined as the multiplication of U matrix and a signal. In this section, we compare the results of FFT and U transform. The two transforms are shown in Table 5.

Table 5. FFT and U transform

	Matrix	
Fast Fourier Transform	Fourier Matrix	$S \times \text{Fourier Matrix}$
U transform	U matrix	$S \times U \text{ matrix}$

Figure 20 shows example plots of a 21x21 Fourier matrix and U matrix with $n=7$, $q=1$, $p=10$

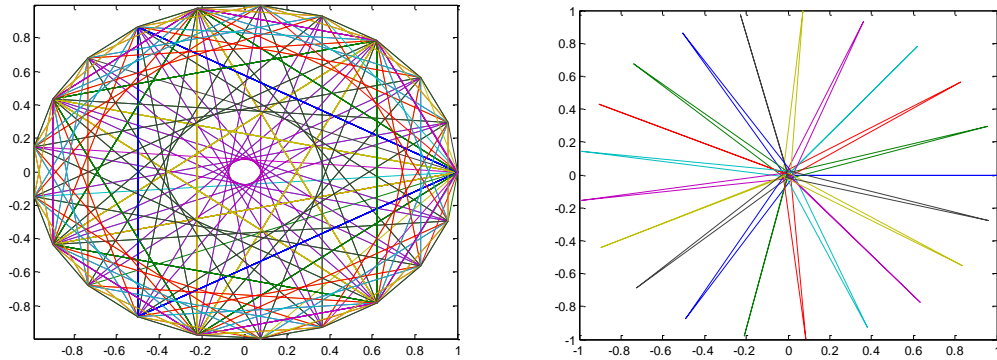


Figure 20 Fourier matrix and U matrix

For visual inspection and comparison purposes, we showed two examples of applying FFT and U transform to a harmonic signal $S = \sin(t) + \sin(2t) + \sin(2.5t)$ and a random signal. The comparison results show that the U transform has better symmetry and is more visually uniform from a geometric point of view. The harmonic signal S and its FFT and U transform are shown in Figure 21.

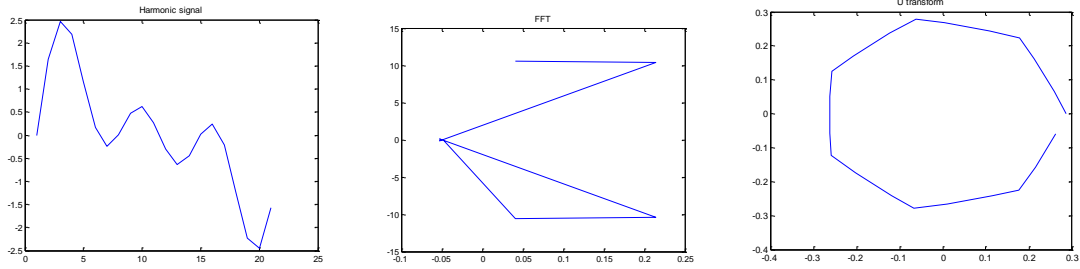


Figure 21 FFT and U transform of a harmonic signal

We also carried out simulations of U transform with more samples to compare the results with FFT. The harmonic signal S and its FFT and U transform with 629 samples are shown in Figure 22. The results further confirm the uniform and symmetric feature of the U transform. Furthermore, it can tell visually that FFT only captures part of the result U transform captures.

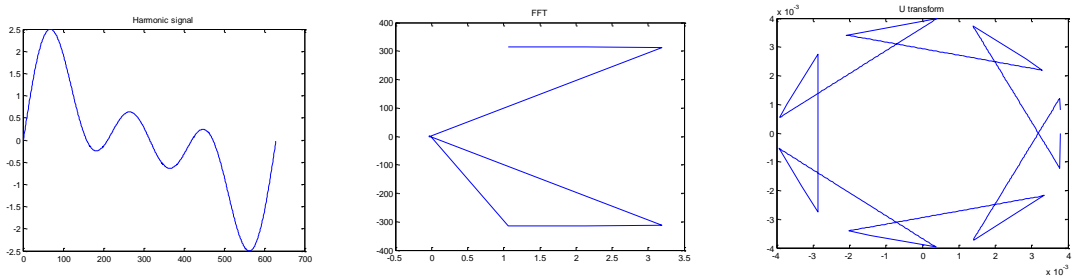


Figure 22 FFT and U transform of a harmonic signal with more samples

Other than harmonic signals, we carried out more simulations to compare the results of FFT and U transform. Comparison is also done for random signals which could show more generality. The random signal and its corresponding FFT and U transform plots are shown in Figure 23.

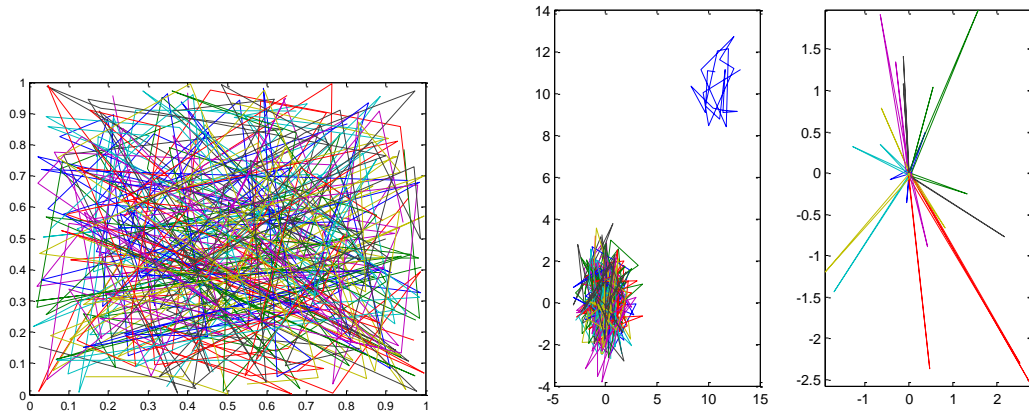


Figure 23 FFT and U transform of a random signal

We also applied the simulation to a random signal with more samples, and the comparison results are shown in Figure 24.

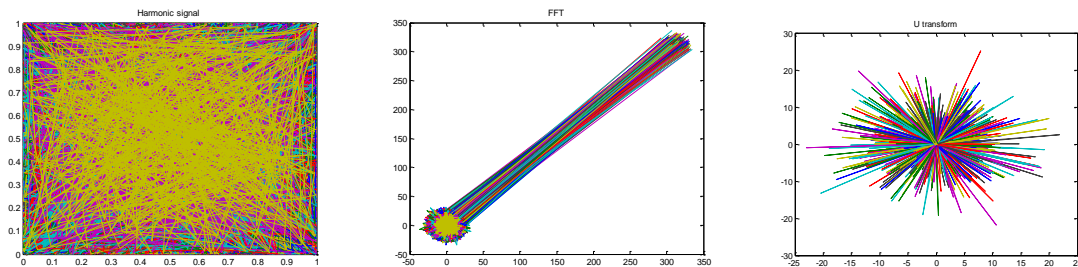


Figure 24 FFT and U transform of a random signal with more samples

Zeta Transform

Riemann zeta function

The Riemann zeta function is an important special function, which is expressed by Riemann in his 1859 paper [38]. The Riemann zeta function is defined over the complex plane for one complex variable. The Riemann zeta function $\zeta(s)$ is a function of complex variables that analytically continues the sum of the infinite series $\sum_{n=1}^{\infty} \frac{1}{n^s}$,

which converges when the real part of s is greater than 1 [38]. It is implemented in Matlab as function `zeta` and also implemented in Mathematica as `Zeta`.

The Riemann zeta function $\zeta(s)$ is a function of a complex number $s = \sigma + it$.

$\zeta(s)$ is defined as

$$\zeta(s) = \sum_{n=1}^{\infty} \frac{1}{n^s} = \frac{1}{1^s} + \frac{1}{2^s} + \frac{1}{3^s} + \dots \quad \sigma = R(s) > 1$$

The infinite series converges for all complex numbers s with real parts greater than 1.

Research shows that there was a connection between the Riemann Zeta function and atomic energy level [39-42]. The spacing between consecutive zeros of the zeta function behaves statistically like the spacing between consecutive eigenvalues of Gaussian Unitary Ensemble large random matrices [40].

We further apply our proposed algorithm to the Riemann Zeta function. First, we replace the S term in the equation with our U matrix, and we can obtain a raw Riemann Zeta matrix. Then, we filter the infinite elements in the matrix and obtain our Riemann Zeta matrix. The process is shown in Figure 25.



Figure 25 Deriving Riemann Zeta Matrix from U matrix

The plots of the Riemann zeta matrix are shown in Figure 26.

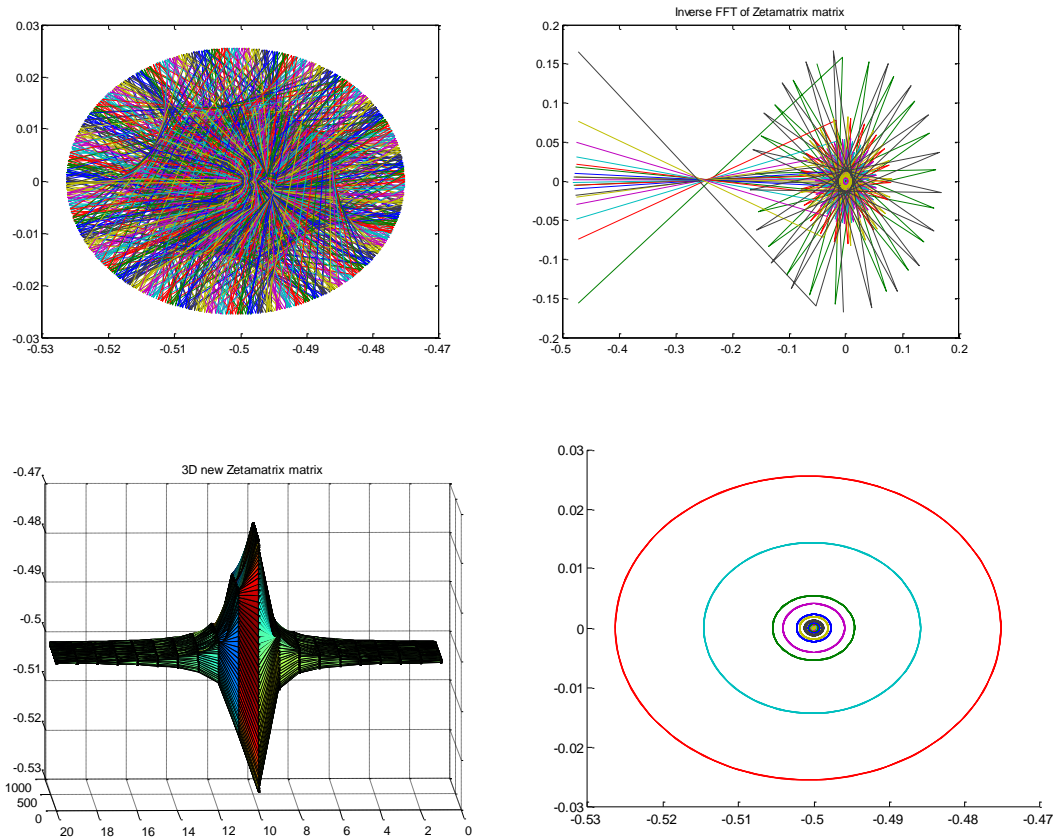


Figure 26 Riemann Zeta matrix

We can observe a more complex compound system of communication channels as shown in Figure 27.

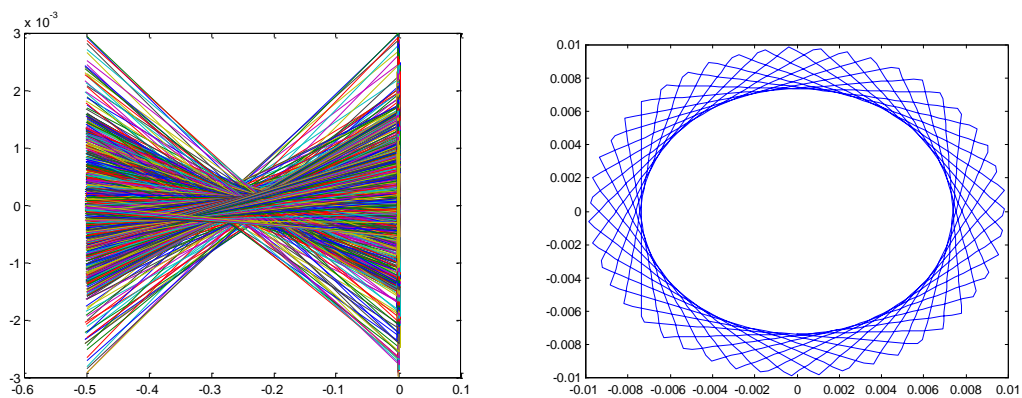


Figure 27 The side and top views of a compound system of communication channels

We further extended the U transform to Zeta Transform by replacing the U matrix with Zeta matrix as shown in Table 6.

Table 6 FFT, U transform, Zeta transform

	Matrix	
Fast Fourier Transform	Fourier Matrix	$S \times \text{Fourier Matrix}$
U transform	U matrix	$S \times U \text{ matrix}$
Zeta transform	Zeta matrix	$S \times \text{Zeta matrix}$

An example to compare FFT, the U transform, and Zeta transform is shown in

Figure 28.

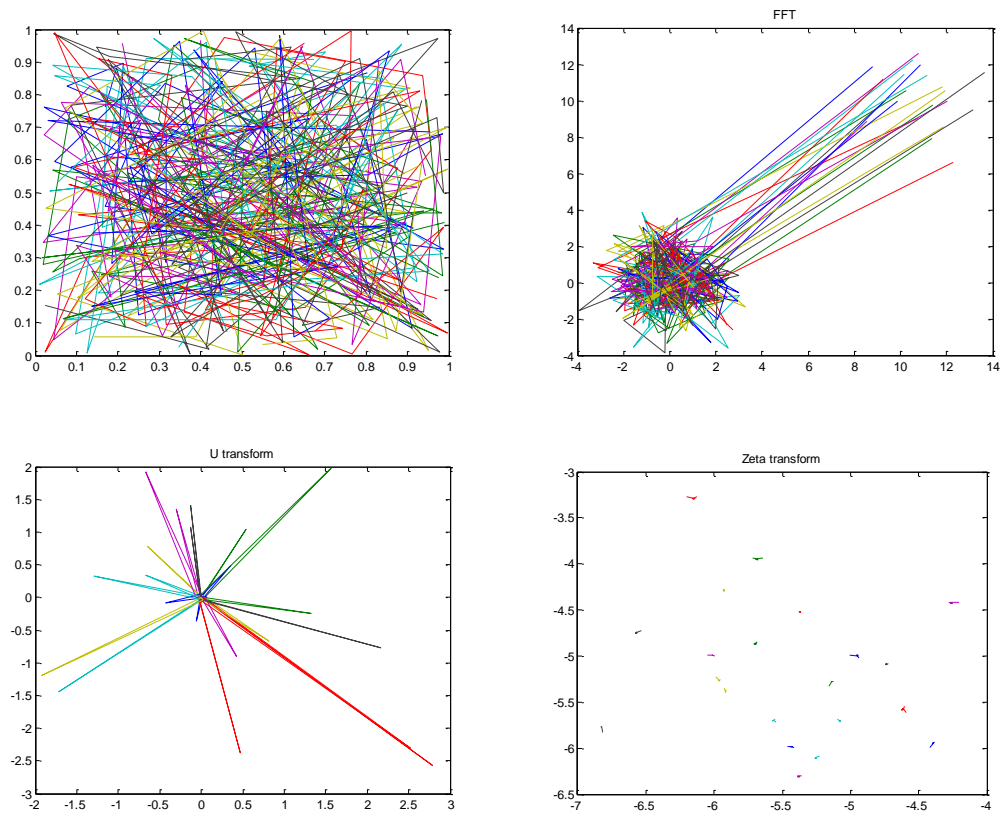
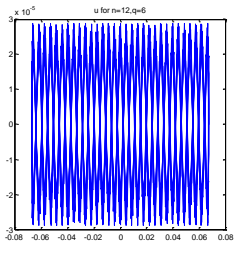
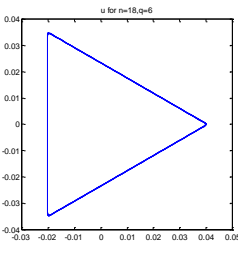
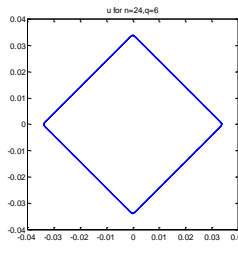
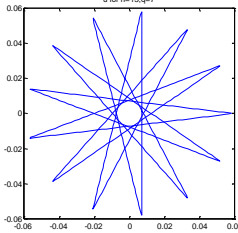
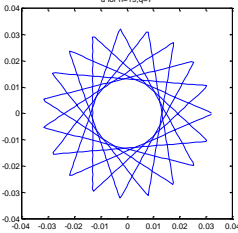
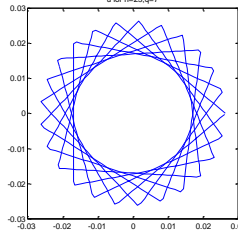


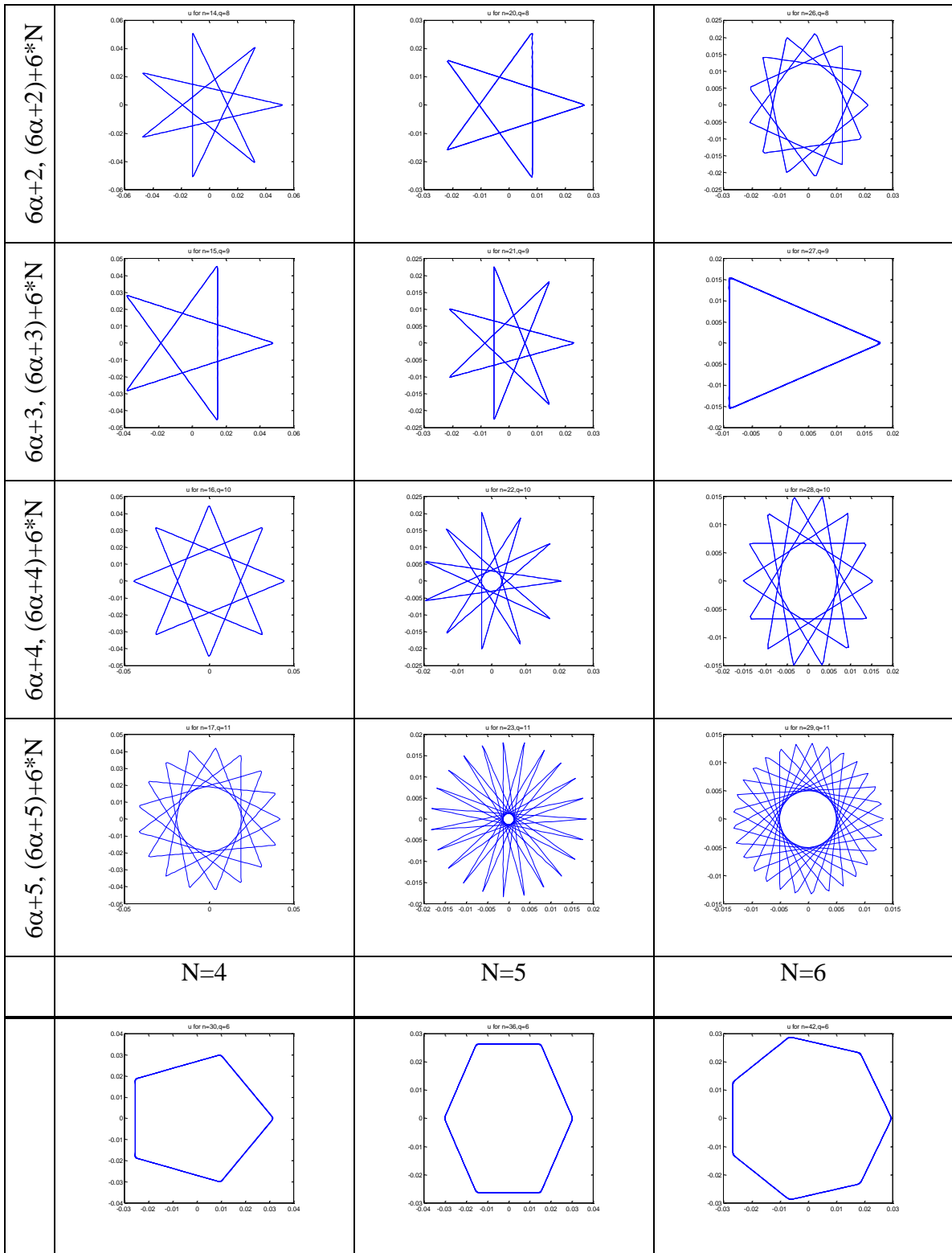
Figure 28 Comparison of FFT, U transform, and Zeta transform

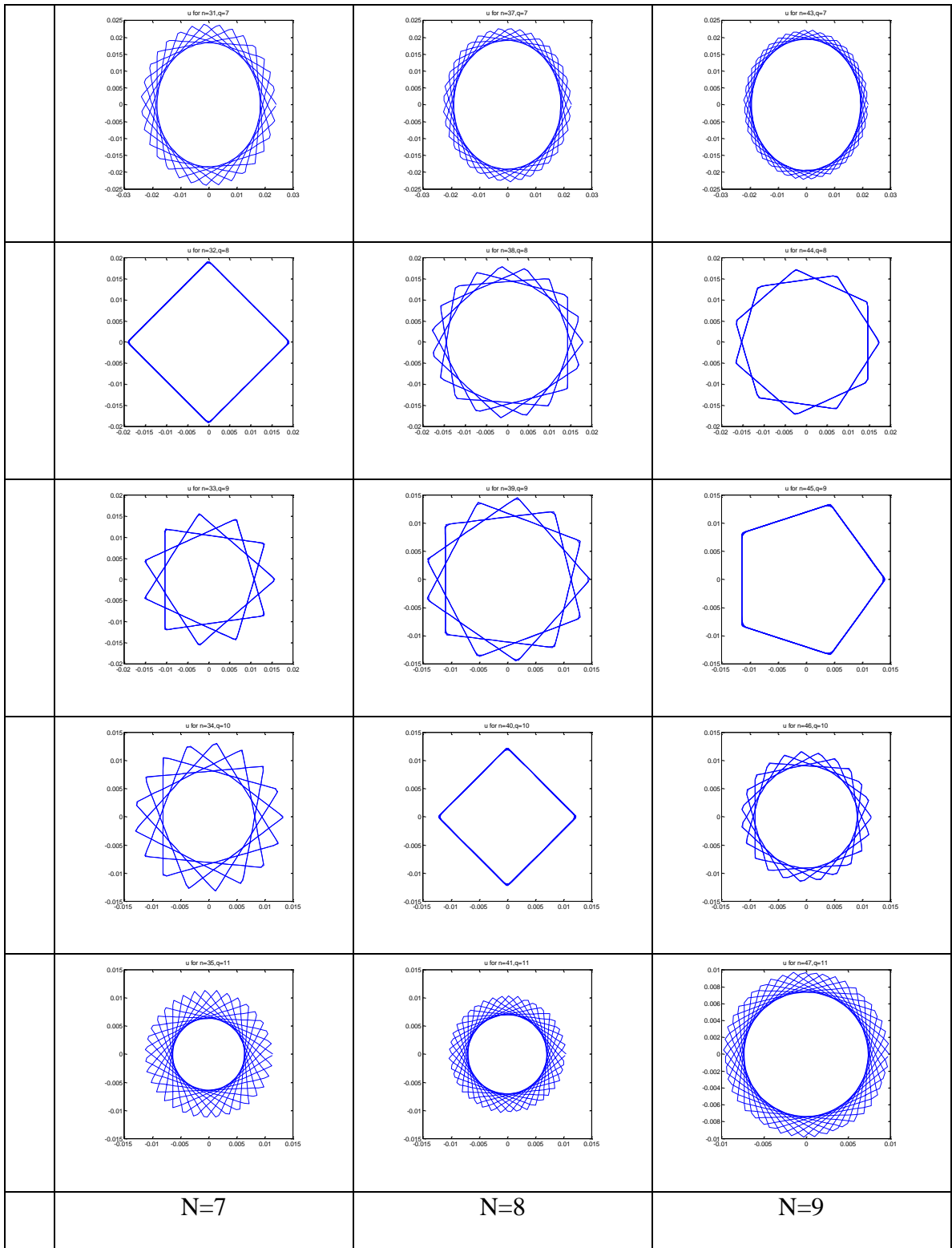
U matrix grouping

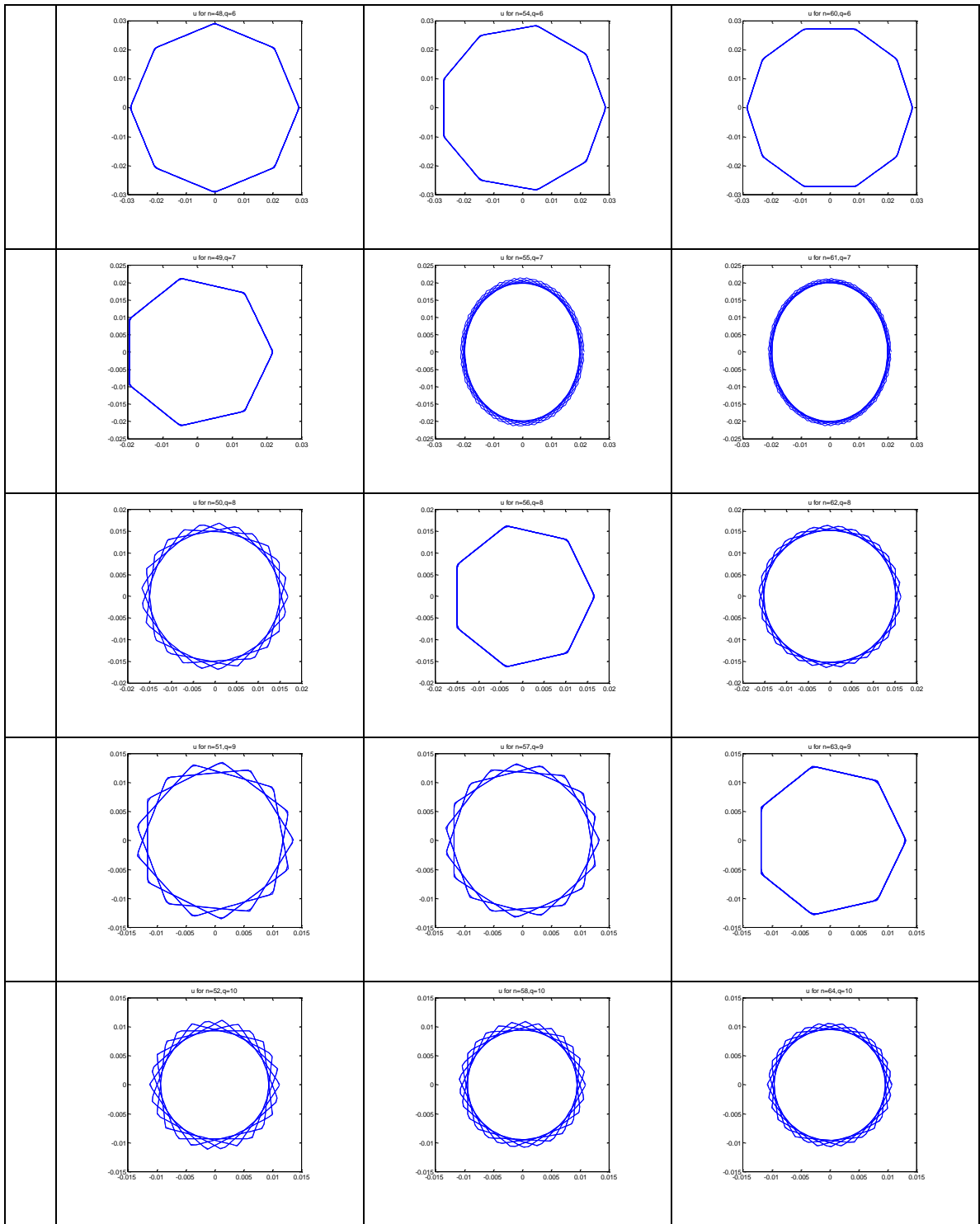
By tuning the parameters of the numerical equation, the proposed U matrix could produce any polygons as needed. We grouped the polygons based on six alpha principles to show the modular pattern [33]. The polygons shown in Table 7 are constructed based on the following pattern. Each polygon is a $u_{(n,q)}$ polygon; n and q are given as shown in the rows and columns in the table. Table 7 shows the case of $\alpha = 1$, so the plot of the first row and first column is $u_{(n,q)} = u_{(6\alpha+6N,6\alpha)}$. Substituted with $\alpha = 1, N = 1$, the plot is for $u_{(12,6)}$. The plot of the second row and first column is $u_{(n,q)} = u_{(6\alpha+1+6N,6\alpha+1)}$. Substituted with $\alpha = 1, N = 1$, the plot is for $u_{(13,7)}$. Following this pattern, it is easy to obtain the corresponding U matrix of the plots.

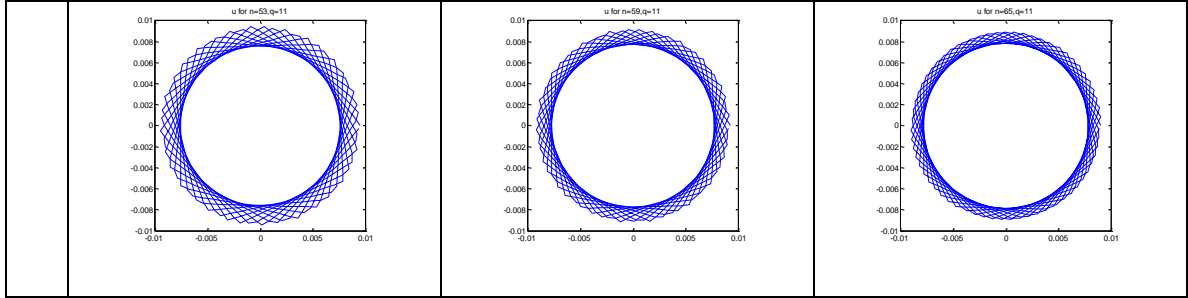
Table 7 Fourier polygon grouping based on N changes

	N=1	N=2	N=3
$6\alpha, 6\alpha+6*N$			
$6\alpha+1, (6\alpha+1)+6*N$			





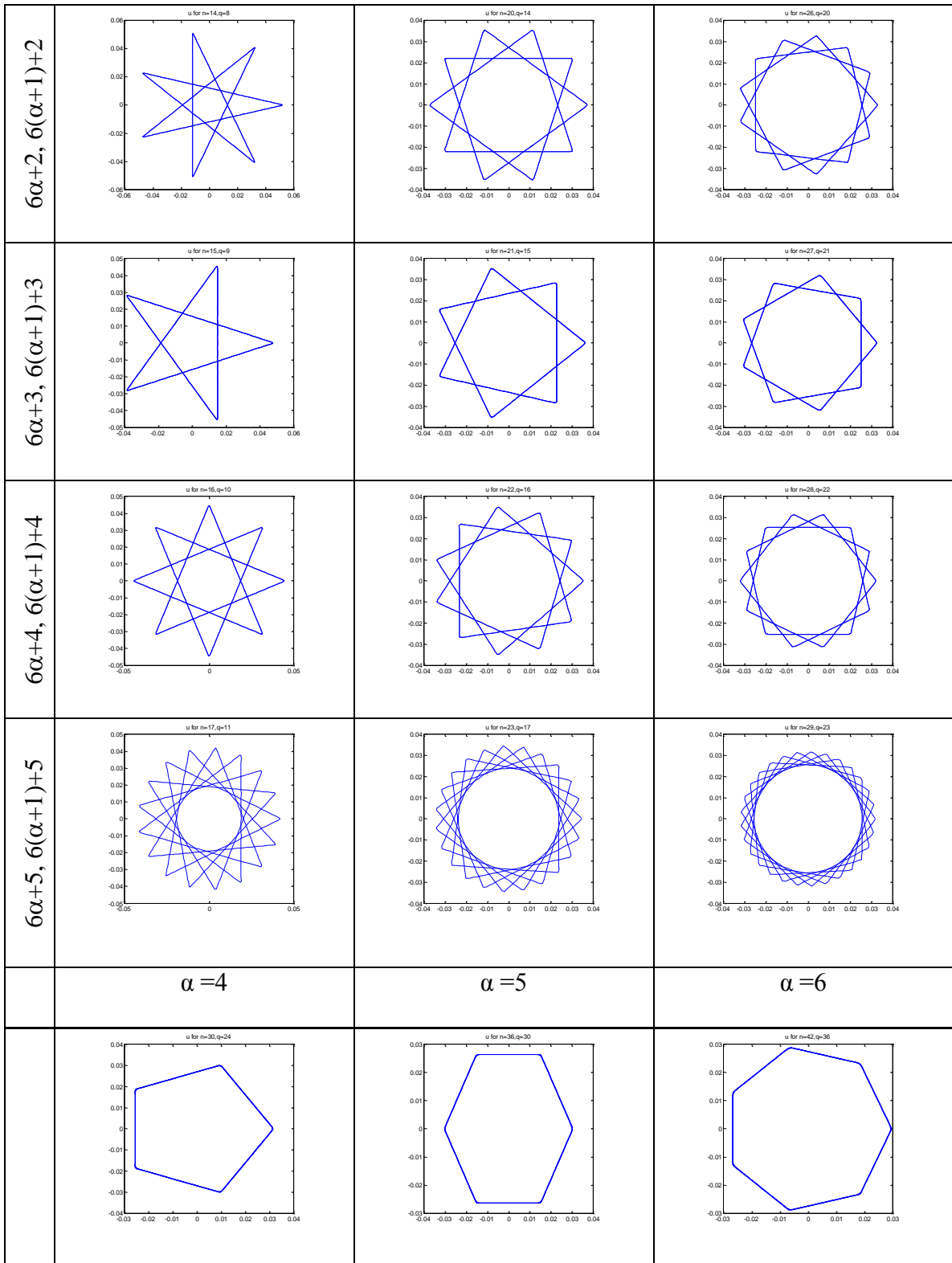


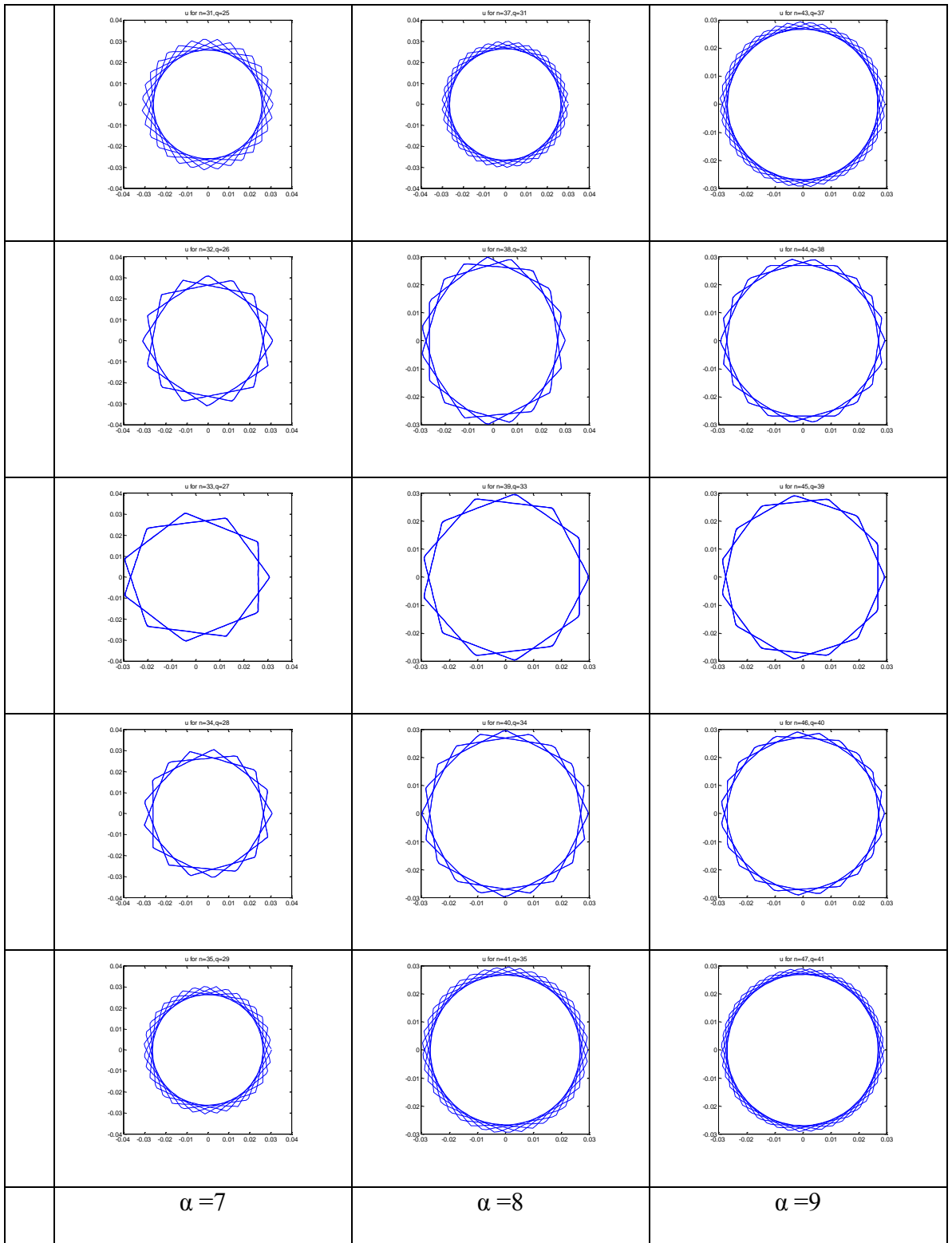


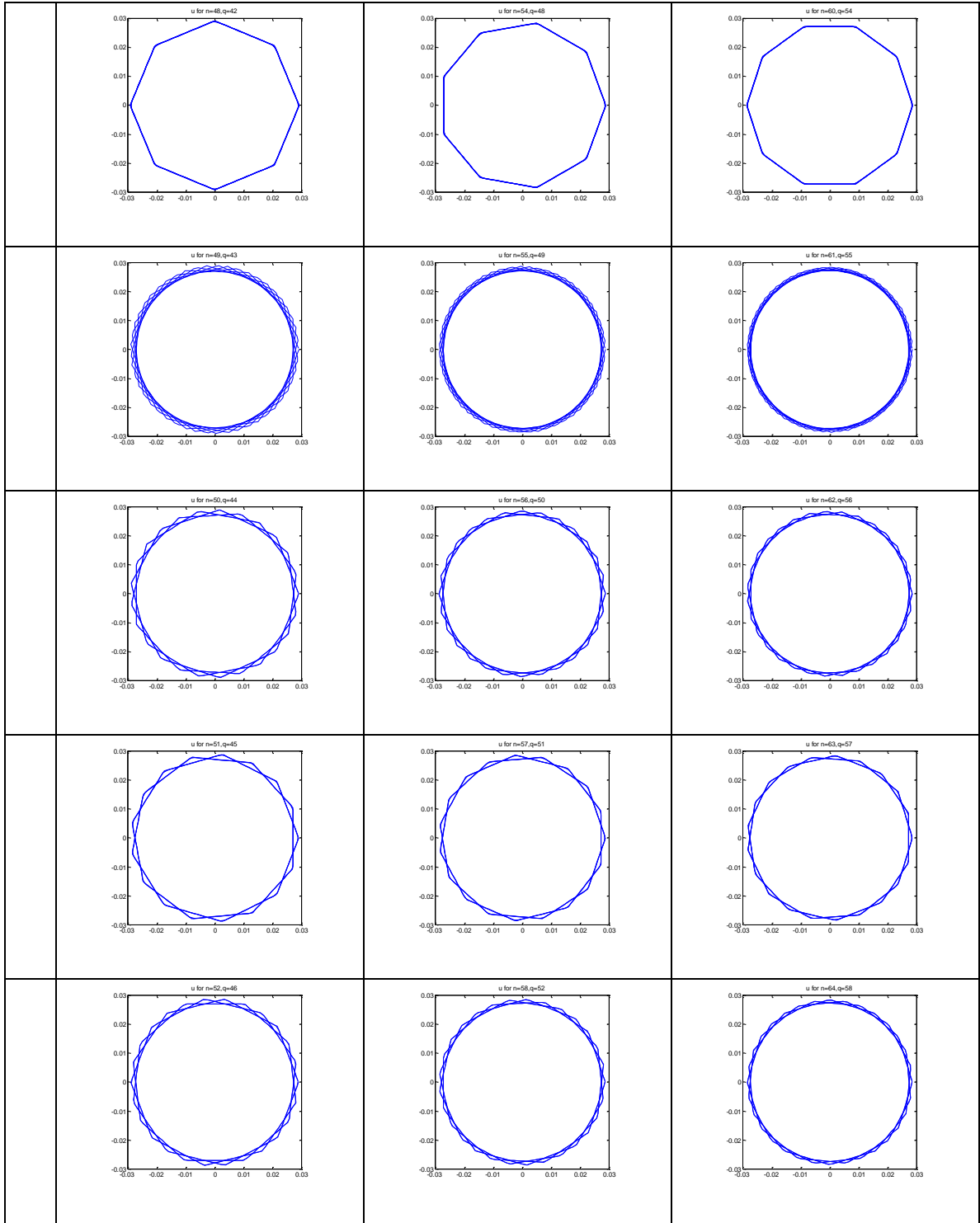
We also studied another way of grouping the polygons by changing α [33]. The polygons shown in Table 8 are constructed with the similar pattern used to construct polygons in Table 7. Each polygon is a $u_{(n,q)}$ polygon; n and q are given as shown in the rows and columns in the table. The plot of the first row and first column of Table 8 is $u_{(n,q)} = u_{(6(\alpha+1),6\alpha)}$. Substituted with $\alpha = 1$, the plot is for $u_{(12,6)}$. The plot of the second row and first column is $u_{(n,q)} = u_{(6(\alpha+1)+1,6\alpha+1)}$. Substituted with $\alpha = 1$, the plot is for $u_{(13,7)}$. Following this pattern, it is easy to obtain the corresponding U matrix of the plots. There are some overlaps of Table 7 and Table 8.

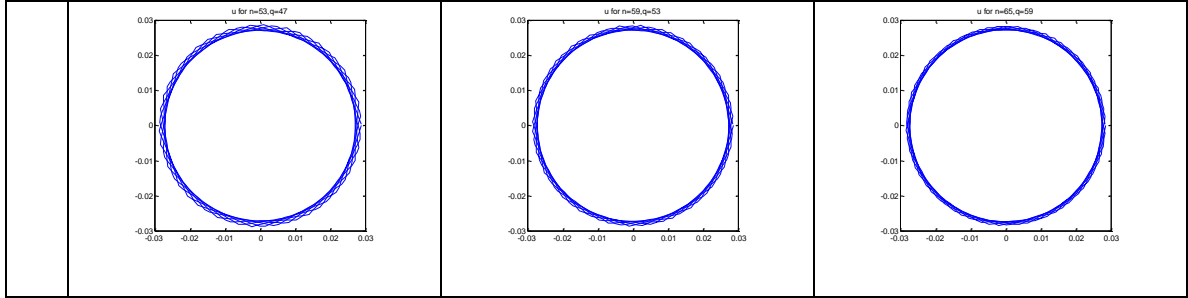
Table 8 Fourier polygon grouping with α change

	$\alpha=1$	$\alpha=2$	$\alpha=3$
$6\alpha, 6(\alpha+1)$			
$6\alpha+1, 6(\alpha+1)+1$			









Chebyshev matrix eigenvalues, Polygon geometry and Amplitwist

In our research group we found that there was a close connection between the polygon geometry and the Chebyshev matrix eigenvalues; that is, the eigenvalue of the Chebyshev matrix equals to the polygon edge and radius ratio [15,21]. Combined with the amplitwist analysis, the tool set could be used to further study these relationships and their implications.

Amplitwist

The term amplitwist comes from Tristan Needhan [43]. The idea is that a complex-valued function is differentiable at a point where it has an “amplitwist” there. The amplitwist is a combination of amplification and twisting, or rotation. The “amplification” is the expansion factor, and the “twist” is the angle of rotation. Under this definition, the amplitwist of f , $f'(z)$ may be thought of locally as a linear transformation,

$$\begin{aligned}
 f'(z) &= \text{the amplitwist of } f \text{ at } z \\
 &= (\text{amplification})e^{i(\text{twist})} \\
 &= |f'(z)|e^{i\text{arg}[f'(z)]}
 \end{aligned}$$

Example 3. If $A = ae^{i\alpha}$, this represents the combination of an origin-centered expansion by a , and a rotation by α as shown in Figure 29. Hence

$$(AZ)' = \text{amplitwist of } (AZ) = A$$

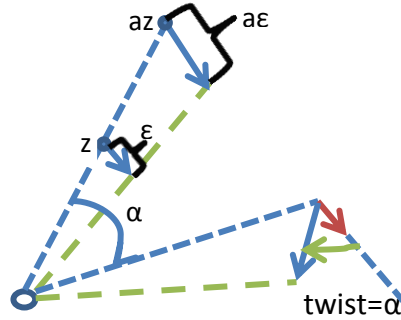


Figure 29 An example of amplitwist

Eigenvalues of Chebychev matrix

The corresponding Chebyshev polynomial matrix [15, 46] is defined as

$$U_p(x) = \begin{bmatrix} 2x & 1 & 0 & 0 & \cdots & 0 \\ 1 & 2x & 1 & 0 & \cdots & 0 \\ 0 & 1 & 2x & 1 & \cdots & 0 \\ 0 & 0 & 1 & 2x & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1 & 2x \end{bmatrix}$$

The eigenvalues of $U_p(x)$ is

$$2(x - 1) + 4\sin^2 \frac{q\pi}{2p + 2} \quad q = 1, 2, \dots, p$$

The corresponding Chebyshev matrix of the second kind is

$$V_p(x) = \begin{bmatrix} 2x & 1 & 0 & 0 & \cdots & 0 \\ 1 & 2x & 1 & 0 & \cdots & 0 \\ 0 & 1 & 2x & 1 & \cdots & 0 \\ 0 & 0 & 1 & 2x & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1 & 2x + 1 \end{bmatrix}$$

The eigenvalues of $V_p(x)$ is

$$2(x - 1) + 4\sin^2 \frac{q\pi}{2p + 1} \quad q = 1, 2, \dots, p$$

Amplitwist, Eigenvalue and polygon geometry

Let us start from Heisenberg’s uncertainty principle in quantum mechanics [53]

$$\Delta x \Delta p \geq \frac{h}{4\pi}$$

Here Δx is the deviation of position x , and Δp is the deviation of momentum p of a particle. Let us rewrite the equation in the form of

$$2\pi \Delta x \Delta p \geq \frac{h}{2}$$

Let $\Delta x \Delta p = R$, we obtain

$$2\pi R \geq \frac{h}{2}$$

which corresponds to a circle in geometry, and a simple model of the “particle movement” is shown in Figure 30.

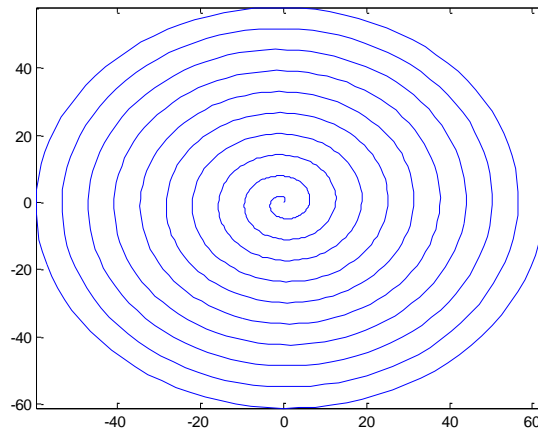


Figure 30 A simple model of particle movement model

As demonstrated by Bunyamin Ozadin [11, 46], there is a connection between the eigenvalues of the Chebychev matrix and polygon geometry. The radius and polygon edge ratio equals to the eigenvalues of the certain Chebychev matrices [15,46].

Let us take the regular polygon with 8 vertices for example. Figure 31 shows the edges of regular polygons $\begin{Bmatrix} 8 \\ 1 \end{Bmatrix}, \begin{Bmatrix} 8 \\ 2 \end{Bmatrix}, \begin{Bmatrix} 8 \\ 3 \end{Bmatrix}$, which correspond to AB, AC and AD respectively.

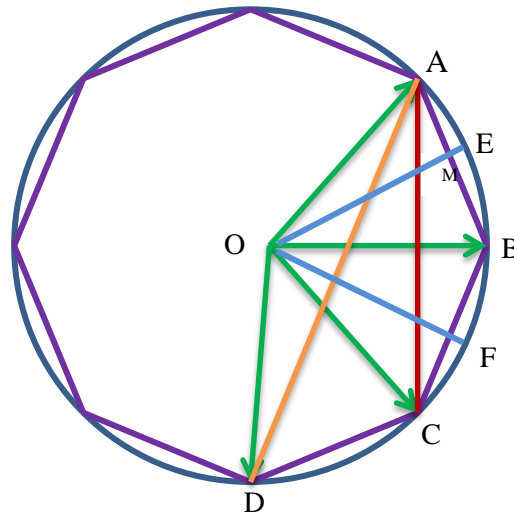


Figure 31 Polygon geometry and probability amplitude

A, B, C, D equally distributed the circle and they correspond to roots of unity. It is obvious that

$$\widehat{AB} = \widehat{BC} = \widehat{CD} = \frac{2\pi}{n}$$

Triangle OAB is an isosceles triangle, and if we make E the middle point of A and B, OM is the altitude of triangle AOB, and OM is perpendicular to AB. Thus, we have

$$\angle AOM = \angle BOM = \frac{1}{2} \angle AOB = \frac{1}{2} \frac{2\pi}{n} = \frac{\pi}{n}$$

Let us look at right triangle AOM,

$$\overline{OA} \sin \angle AOM = \overline{OA} \sin \frac{\pi}{n} = \overline{AM} = \frac{1}{2} \overline{AB}$$

Thus, we can obtain

$$\frac{\overline{AB}}{\overline{OA}} = 2 \sin \frac{\pi}{n}$$

Applying similar analysis to isosceles triangles OAC and OAD, we have

$$\frac{\overline{AC}}{\overline{OA}} = 2 \sin \frac{2\pi}{n}$$

$$\frac{\overline{AD}}{\overline{OA}} = 2 \sin \frac{3\pi}{n}$$

Since \overline{AB} , \overline{AC} , and \overline{AD} correspond to the edges of the regular polygon $\left\{ \frac{n}{q} \right\}$ for the case $n=8$, $q=1, 2, 3$ in the example, let us use l_q to replace \overline{AB} , \overline{AC} , and \overline{AD} , and we can have the equation

$$\frac{l_q}{\overline{OA}} = 2 \sin \frac{q\pi}{n}$$

or

$$\frac{l_q}{R} = 2 \sin \frac{q\pi}{n}$$

Let us recall the Chebyshev matrix, the eigenvalues of $U_p(x)$ and $V_p(x)$ equal to the square of a polygon edge and radius ratio $\left(\frac{l_q}{R}\right)^2$ under certain conditions [11,15,21].

Let us look back to Figure 30, and we could connect this model with amplitude and polygon geometry as shown in Figure 32.

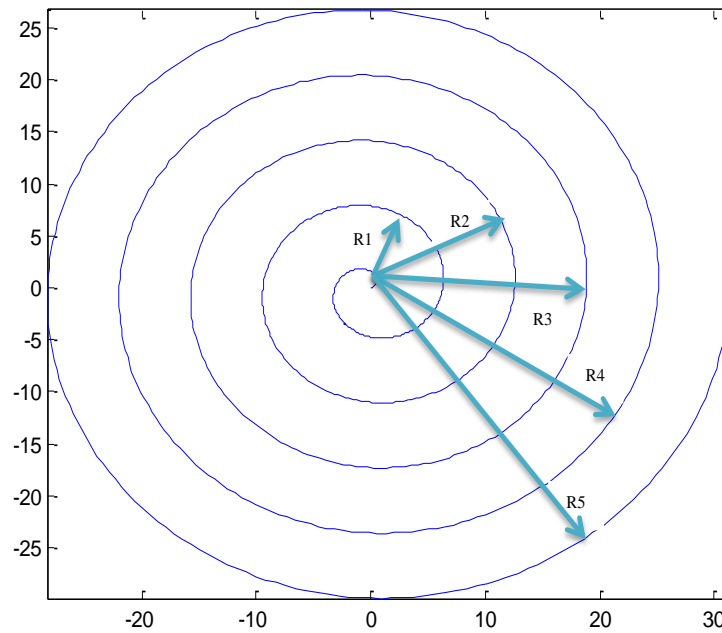


Figure 32 Amplitwist and “particle movement” model

Let us look back to the polygon geometry equation:

$$\frac{l_q}{R} = 2 \sin \frac{q\pi}{n}$$

Giving $R_1, R_2, R_3, R_4, R_5, \dots$, let us start with R_1 and R_2 first; we have

$$\frac{l_q}{R_1} = 2 \sin \frac{q\pi}{n}$$

$$\frac{l_q}{R_2} = \frac{R_1}{R_2} 2 \sin \frac{q\pi}{n}$$

Let us take the example of $n=8$ and $q=1$, the equations correspond to the geometry shown in Figure 33.

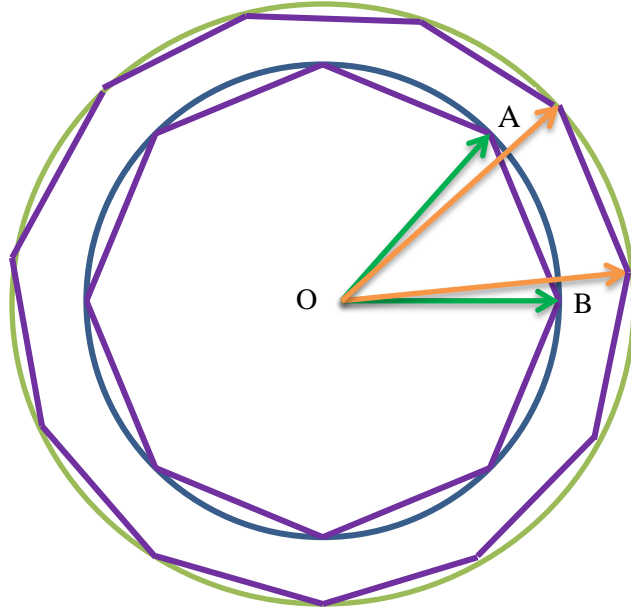


Figure 33 Corresponding polygon geometry example

$$\frac{l_q}{R_1} = 2\sin \frac{\pi}{8}$$

$$\frac{l_q}{R_2} = \frac{R_1}{R_2} 2\sin \frac{\pi}{8}$$

From the two equations, if we keep l_q and the angle the same, we just change R_2 , and the new ratio $\frac{l_q}{R_2}$ corresponds to a new eigenvalue of the Chebychev matrix.

We could obtain another form of the equation for R_2 . If we replace the edge with the same length for polygon $n=10$, $q=1$, for example, then we have

$$\frac{l_q}{R_2} = 2\sin \frac{\pi}{10}$$

To be strictly mathematical, there always exists another polygon $\left\{ \begin{matrix} n \\ q \end{matrix} \right\}$ such that

$$R_1 \sin \frac{\pi}{n1} = R_2 \sin \frac{\pi}{n2}$$

Thus, we have

$$n_2 = \frac{\pi}{\arcsin\left(\frac{R_1}{R_2} \sin \frac{\pi}{n_1}\right)}$$

The deduction and observation could further be used in our analysis of the new quantum computer design technique.

3. GNCOM FOR QUANTUM COMPUTING

In this chapter, we illustrate in detail how the generalized noisy communication channel approach is applied to quantum computing. First we start with a brief introduction to quantum computing, then follow with a discussion of the GNcom representation of quantum computing, and conclude with an application of a universal linear equation derived from the multiple-input multiple-output communication channel to quantum computing by using the U matrix as a quantum register.

Introduction to quantum computing

Quantum computing has become a fascinating research area for physicists and computer scientists over the last 20 years. Quantum computing is an interdisciplinary research field, since it combines physics, mathematics, and computer science. Quantum computing shows another perspective of computation, which is to think computation physically. Furthermore, quantum computation provides new tools to solve the class of problems that are hard or impossible to solve with traditional computing [67].

History of quantum computation

Quantum computation is a branch of theoretical physics. Dating back to 1970s, the interest of developing single quantum systems contributes to the development of

quantum computation [47, 53]. Richard Feynman's speech on the simulation of quantum mechanical systems further inspired the research. Overall, much progress had been made in the 20th century in the field of information theory, which connects the abstract notion of information to the laws of thermodynamics [64]. A great deal of excitement was triggered in the mid-1990s when Shor's factoring algorithm and Grover's search algorithm were discovered. Both of the algorithms have been applied to real world problems other than just simulations; in addition, both show significant speed up over the best known classical algorithms. For example, Shor's factoring algorithm enables someone to break an encryption scheme in a very short time.

Due to these intense interests, a great deal of efforts has been made to build quantum computers, but the state of the art of quantum computer design is limited to small quantum computers with a few qubits operation. The state of the art of quantum computers is listed in Table 9. The development of large-scale quantum computers still remains as a great challenge.

Table 9 The-state-of-the-art of quantum computers

Year	Institute	Chip	Function
2005	University of Michigan	Ion trap chip[48]	scalable quantum computing tools
2009	Yale university	rudimentary solid-state quantum processor[49,50]	Two-qubit superconducting chip could run elementary algorithms
2009	University of Bristol	silicon-based quantum computing chip[51-53]	Could run Shor's algorithm
2011	University of New South Wales	quantum teleportation[54,55]	successfully transferred a complex set of quantum data
2011	D-Wave Systems	the first commercial quantum annealer[56]	128 qubit processor chipset.
2011	University of Bristol	an all-bulk optics system[57]	run an iterative version of Shor's algorithm
2011	University of California, Santa Barbara	Quantum Processor Hooks Up with Quantum Memory[58]	proof of a quantum computer can be made with a Von Neumann architecture
2012	IBM	Sapphire chip[59]	close to the minimum requirements for a full-scale quantum computing system
2012	University of Southern California, Delft University of Technology, the Iowa State University of Science and Technology, and the University of California, Santa Barbara	two-qubit quantum computer[60]	Grover's algorithm for four variants of search has generated the right answer from the first try in 95% of cases
2012	University of New South Wales	the first working "quantum bit" based on a single atom in silicon[61,62]	The same technological platform that forms the computers, laptops and phones.

Basics

Bits and functions operating on bits are the basic theory of classical computing.

By analogy, quantum bits and their associated linear transformations form the basics of

quantum computing. A classical computer is built from electrical circuits with logic circuits and interconnects. A quantum computer is built from quantum circuits with interconnects and quantum logic circuits. A special interest of quantum computing research is quantum behaviors that are hard to simulate on a classical computer, like superposition and interference.

In this section, we will give a simple introduction to quantum computing. Starting with the basic unit of quantum computing - qubits, and then followed by the basic logical operations - quantum gates.

The Qubit

The fundamental concept of quantum computation and information is the quantum bit, or qubit for short. Qubits are mathematical objects with specific properties and can be realized as actual physical systems. Treating qubits mathematically has the flexibility to construct a general theory of quantum computation that is independent of a specific system. On the other hand, the beauty of qubits is that computation is very much a physical process and not just a mathematical formalism [53, 69].

A classical bit has two states, 0 or 1. Analogous to this, a qubit also has a state. Two possible states corresponding to the classical bit 0 and 1 of a qubit are $|0\rangle$ and $|1\rangle$. A notation like $|\rangle$ is called the Dirac Notation, which is a standard notation for states in quantum mechanics. However unlike a classical bit, a qubit can have more than just two states. Qubits can also be a normalized superposition, as linear combination of the basic states,

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$$

where α and β are complex numbers, and $|\alpha|^2 + |\beta|^2 = 1$.

However, if we make a measurement of superposition states, it is impossible for us to measure all the terms in the superposition states. When we measure the qubit, it will produce $|0\rangle$ with probability $|\alpha|^2$ and $|1\rangle$ with probability $|\beta|^2$, since $|\alpha|^2 + |\beta|^2 = 1$.

A qubit can be rewritten in a geometric representation.

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$$

$$|\alpha|^2 + |\beta|^2 = 1$$

which yields[63]

$$|\psi\rangle = \cos\frac{\theta}{2}|0\rangle + e^{i\varphi}\sin\frac{\theta}{2}|1\rangle.$$

The numbers θ and φ are real numbers, and they define a point on the unit three dimensional sphere, which is often called the Bloch sphere [64]. This geometric mapping is very helpful to visualize the state of a single qubit as shown in Figure 34.

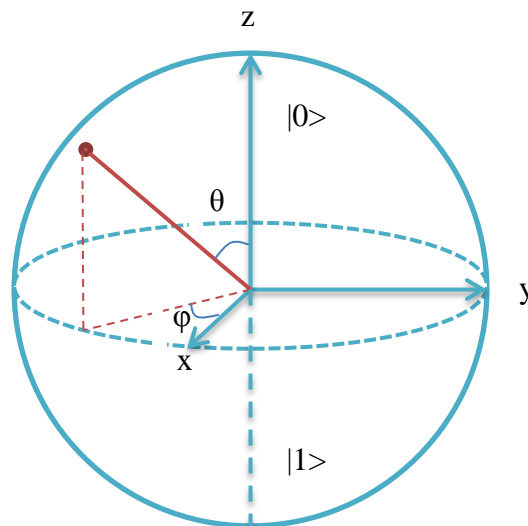


Figure 34 The Bloch sphere representation of a single qubit

Multiple qubits

Analogous to classical bits, multiple qubits also exist. For two classical bits, there exist four possible states, 00, 01, 10 and 11. Correspondingly, a two-qubit system has four basic states $|00\rangle$, $|01\rangle$, $|10\rangle$ and $|11\rangle$.

If we have two qubits whose states are given by

$$|\psi_1\rangle = a_1|0\rangle + b_1|1\rangle \text{ and}$$

$$|\psi_2\rangle = a_2|0\rangle + b_2|1\rangle,$$

then the state of the whole system is:

$$|\psi\rangle = |\psi_1\rangle \otimes |\psi_2\rangle$$

$$|\psi\rangle = (a_1|0\rangle + b_1|1\rangle) \otimes (a_2|0\rangle + b_2|1\rangle)$$

$$|\psi\rangle = a_1b_1|00\rangle + a_1b_2|10\rangle + a_2b_1|01\rangle + a_2b_2|11\rangle$$

Thus we obtain the general two-qubit state form

$$|\psi\rangle = a|00\rangle + b|10\rangle + c|01\rangle + d|11\rangle,$$

$$|a|^2 + |b|^2 + |c|^2 + |d|^2 = 1.$$

The equation above can only be written as a product of two single qubit states if $ad=bc$. This gives a definition for an extremely important phenomenon in quantum computing, entanglement. An entangled state is a multi-qubit state that cannot be separated into a product of single qubits [65].

Extending the concept to n qubits, we can denote the n bit binary expansion of x , $|x\rangle_n$. A general n qubit state is written as

$$|\psi\rangle = \sum_{x=0}^{2^n-1} c_x |x\rangle_n$$

$$\sum_{x=1}^{2^n-1} |c_x| = 1.$$

Quantum gates

A single qubit only shows classical values 0 or 1 with certain probability after measurement. As a result, the superposition state will irreversibly change to a basic state, “collapsing” it from the superposition to the outcome of the measurement. From the above description, we can see that, unless the superposition state is measured, the amount of “hidden” information is still stored under the dynamic evolution. This makes manipulation of the information stored in unmeasured qubits possible with quantum gates.

The well-known classical Boolean logic gates are: NOT, AND, OR, NAND, NOR, XOR etc. Let us start with the simplest 1 bit logic gate NOT; the truth table of a classical NOT gate is shown in Table 10.

Table 10 Truth table of a NOT gate

Input	Output
0	1
1	0

Due to the superposition state of qubit, a quantum NOT gate is not just simply interchange $|0\rangle$ to $|1\rangle$, or $|1\rangle$ to $|0\rangle$. The quantum NOT gate behaves linearly; that is, the roles of $|0\rangle$ and $|1\rangle$ are interchanged.

$$\alpha|0\rangle + \beta|1\rangle \xrightarrow{\text{yields}} \alpha|1\rangle + \beta|0\rangle.$$

The quantum NOT gate could be written in matrix form. Let us define a matrix

$$X = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$$

for the quantum NOT gate. The state $\alpha|0\rangle + \beta|1\rangle$ can be written as a vector,

$$\begin{bmatrix} \alpha \\ \beta \end{bmatrix}$$

and the output state $\alpha|1\rangle + \beta|0\rangle$ is [47]

$$X \begin{bmatrix} \alpha \\ \beta \end{bmatrix} = \begin{bmatrix} \beta \\ \alpha \end{bmatrix}.$$

The matrix representation of quantum gate is unitary. If U is a matrix corresponding to a single qubit gate, then $U^\dagger U = I$, where U^\dagger is the adjoint of U , obtained by transposing and then complex conjugating of U . There is only one single bit gate for classical NOT gate, but there are many single qubit gates. Any unitary matrix specifies a valid quantum gate.

An important gate in quantum computing is the Hadamard gate. The Hadamard gate H is defined as

$$H|0\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle), \quad H|1\rangle = \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle),$$

The matrix form is [66]

$$H = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$$

For the classical computing case, there exists a universal gate NAND that could be used to represent any combinations of other gates for computation. By introducing the phase operator φ , together with the Hadamard gate, one can construct an arbitrary one-qubit gate [67]. The phase operators φ are defined by

$$\varphi|0\rangle = |0\rangle, \quad \varphi|1\rangle = e^{i\varphi}|1\rangle,$$

and the corresponding matrix form is

$$\varphi = \begin{bmatrix} 1 & 0 \\ 0 & e^{i\varphi} \end{bmatrix}.$$

In quantum computing, any multiple qubit logic gate may be composed from a quantum CNOT gate and a single-qubit gate [68]. This CNOT gate has two input qubits: the control qubit and the target qubit. If the control qubit is 0, the target qubit remains unchanged. If the control qubit is 1, the target qubit is flipped. The action can be written as:

$$|00\rangle \rightarrow |00\rangle, |01\rangle \rightarrow |01\rangle, |10\rangle \rightarrow |11\rangle, |11\rangle \rightarrow |10\rangle$$

The circuit representation for the CNOT is shown in Figure 35. The top line indicates the control qubit, and the bottom line represents the target qubit.

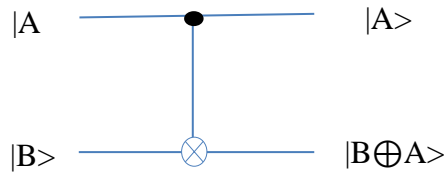


Figure 35 The CNOT quantum gate

The CNOT gate could also be written in Matrix format.

$$U_{CN} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

Another important feature of CNOT gate is that it generates an entangled state.

For example, given an input state

$$|\psi\rangle = |\psi_1\rangle|\psi_2\rangle = (a|0\rangle + b|1\rangle)|0\rangle$$

applying the CNOT gate to it, we obtain

$$C|\psi\rangle = a|00\rangle + b|11\rangle,$$

which cannot be separated into a product of two single qubit states.

Any unitary transformation acting on n qubits can be constructed by combinations of CNOT, Hadamard, and phase gates [69]. These form the universal sets of quantum gates. However, CNOT is not the only choice in the set. The CNOT gate could be replaced by any other two qubit gates that can produce entanglement.

One unique feature of quantum gates is that they are always reversible. Since the inverse of a unitary matrix is also a unitary matrix, a quantum gate can always be inverted by another quantum gate.

Algorithms

The features of quantum computer make it possible to design new algorithms to solve problems which are hard to solve or require tremendous resources to solve with classical computers. However, algorithm design is a highly complicated and difficult task, especially for quantum computing, due to the following reasons. First, the algorithms are designed to speed up computation or reduce complexity with quantum mechanical features. Second, we are lacking a general approach: current method approaches each problem in its own way. Third, quantum computing is so different from classical computing that developers find it difficult to overcome classical concepts and develop new algorithms. Due to the reasons mentioned above, there are only a few powerful quantum algorithms that have been proposed. Among these, Shor's algorithm [71] was considered as one of the major milestones in this field and generated a great deal of excitement in the quantum computing research domain.

Shor's factoring algorithm

Shor's algorithm [71] could successfully factor, a positive integer in polynomial time, while the most successful classical algorithm takes sub-exponential time. Here is the argument:

Given a positive integer N , two prime factors p and q that satisfy $N = pq$ can be found by finding the period of a function

$$f(x) = yx \bmod N$$

where any $y < N$ has no common factors with N other than 1.

The period r of $f(x)$ is determined by y and N . Once the period is known, N can be factored if r is even and $\frac{y^r}{2} \not\equiv -1 \pmod N$. The factors of N are the greatest common divider of $\frac{y^r}{2} \pm 1$ and N , which can be found in polynomial time. In short, Shor's algorithm reduces the problem of finding the period of a certain periodic function, $f(x) = f(x + a)$ [71].

Shor's algorithm became so popular because it could break the heavily used RSA encryption protocol in an exponentially shorter time than a classical computer. This example shows that quantum computing can also have a real-world advantage over classical computing. The quantum speed up of computation shown by Shor's algorithm had a dramatic effect economically and scientifically.

Applications: Field programmable Gate Array (FPGA)

Quantum computing has been utilized to solve many hard computational problems. FPGA routing is such a computation intensive problem. Existing methods are unable to perform the breakthrough in terms of time, cost, and complexity. Quantum computing shed some light on it [72]. In the proposed method, the FPGA routing problem was transferred to a Boolean satisfiability (SAT) equation. The candidates that satisfy the SAT equation to “1” indicate a valid routing. Quantum search algorithms are applied to the Boolean equation to solve the routing problems with properties of quantum superposition and quantum parallelism. The proposed approach could reduce the number of iterations to find a solution [72].

Noisy communication channel representation of quantum computing

In this section, we will discuss the noisy communication channel representation of quantum computing. We start with entangled quantum states and the polygon connection; then we connect quantum computing and noisy communication channel through Fourier polygons [32]; finally, we further connect quantum computing with multiple input multiple output communication channels with the proposed U matrix.

Entangled quantum states and Fourier polygon

As introduced in the previous section, entanglement is a very important feature in quantum computing. In short, the entangled states are dependent on each other, and the more entangled a system is, the more computation power it offers.

The possible quantum states of a chain of particles could be represented as particles in space as shown in Figure 36, and the lines connecting states could be swapped without changing the energy of the chain [73,74].

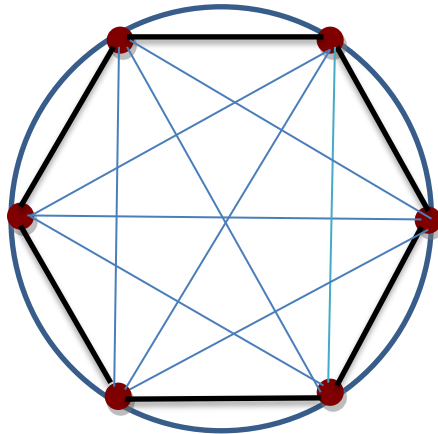


Figure 36 Entangled states in a particle chain

Let us recall our Fourier representation of polygons with 6 vertices as shown in Figure 37.

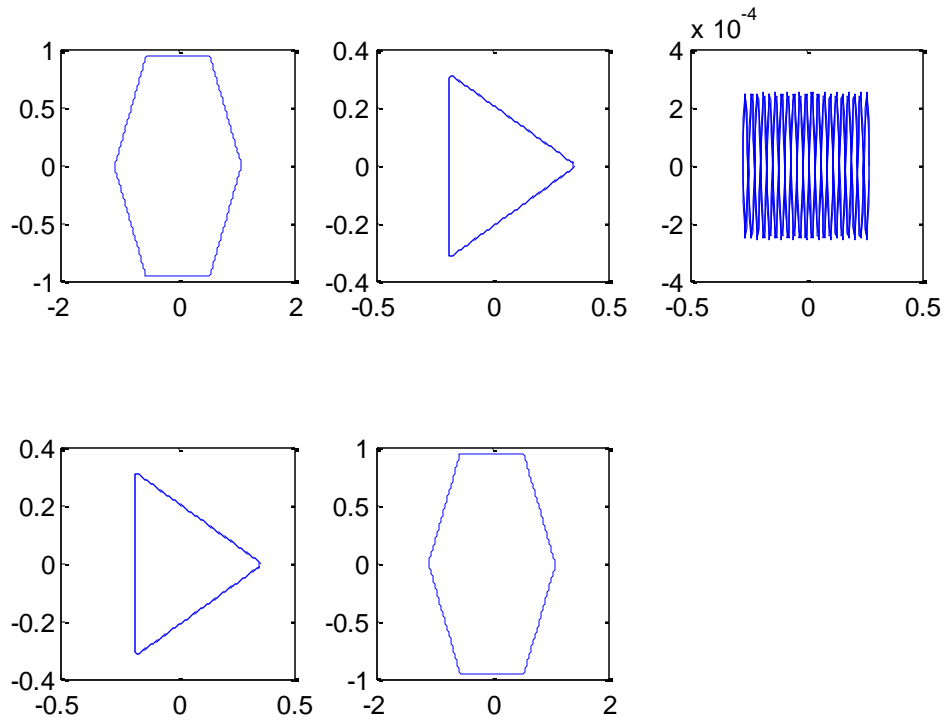


Figure 37 Fourier polygon with 6 vertices

Here we see a close visual similarity between the particle chain and the Fourier polygon representation. The particle chain is the vertices of Fourier polygons, and the entangled quantum states are edges of the Fourier polygons. This observation implies the possibility of modeling the quantum states in a particle chain with Fourier polygons.

Under ideal cases, the particle chain could form a regular polygon, but for the real cases either the particle chain or the entangled states could be in the form of irregular polygons. An example of entangled quantum states is shown in Figure 38.

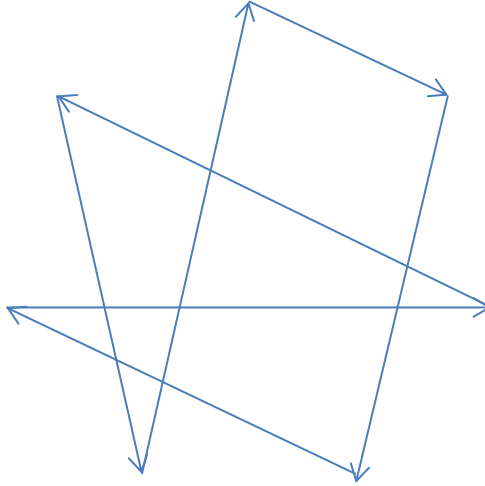


Figure 38 Example of entangled states

As shown in Figure 38, each arrow could be modeled as a quantum state in the form of

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$$

The example of the entangled quantum states is shown in the form of

$$|\psi\rangle = |\psi_1\rangle \otimes |\psi_2\rangle \otimes \cdots \otimes |\psi_n\rangle$$

$$|\psi\rangle = \sum_{x=1}^{2^n-1} c_x |x\rangle_n$$

$$\sum_{x=1}^{2^n-1} |c_x| = 1$$

The entangled states could be modeled with Fourier representation of irregular polygons as shown in Figure 39 by taking pieces from regular polygons.

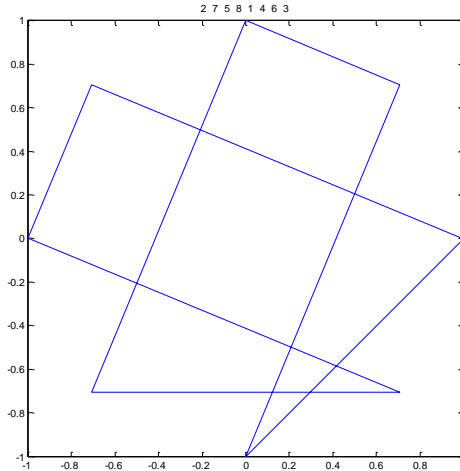


Figure 39 Irregular polygon formed by permutation matrix

The “jump speed” for the line goes from 2 to 7 is $7-2=5$ [15, 31]. Similarly, the speed for the line goes from 7 to 5 is $5-7=-2$. Since the direction to plot the irregular polygon is defined as counterclockwise, the direction of the negative speed moves clockwise with the specified amount. In order to make it clear, we will use mod n for the “speed vector.” The speed vector for this example is $[5, -2, 3, -7, 3, 2, -3, -1]$. After mod n operation, the speed vector turns out to be $[5, 6, 3, 1, 3, 2, 5, 7]$. This vector corresponds to the jump speed of the irregular polygons, which indicates that the irregular polygon is constructed from pieces of regular polygons:

$$u_{8,5}(t), u_{8,6}(t), u_{8,3}(t), u_{8,1}(t), u_{8,3}(t), u_{8,2}(t), u_{8,5}(t), u_{8,7}(t)$$

The resolution parameters p_i are all set to 10. Figure 40 shows the Fourier series representation regular polygons used to construct the irregular polygon. One side of each polygon is the piece needed to construct the irregular polygon shown in Figure 39.

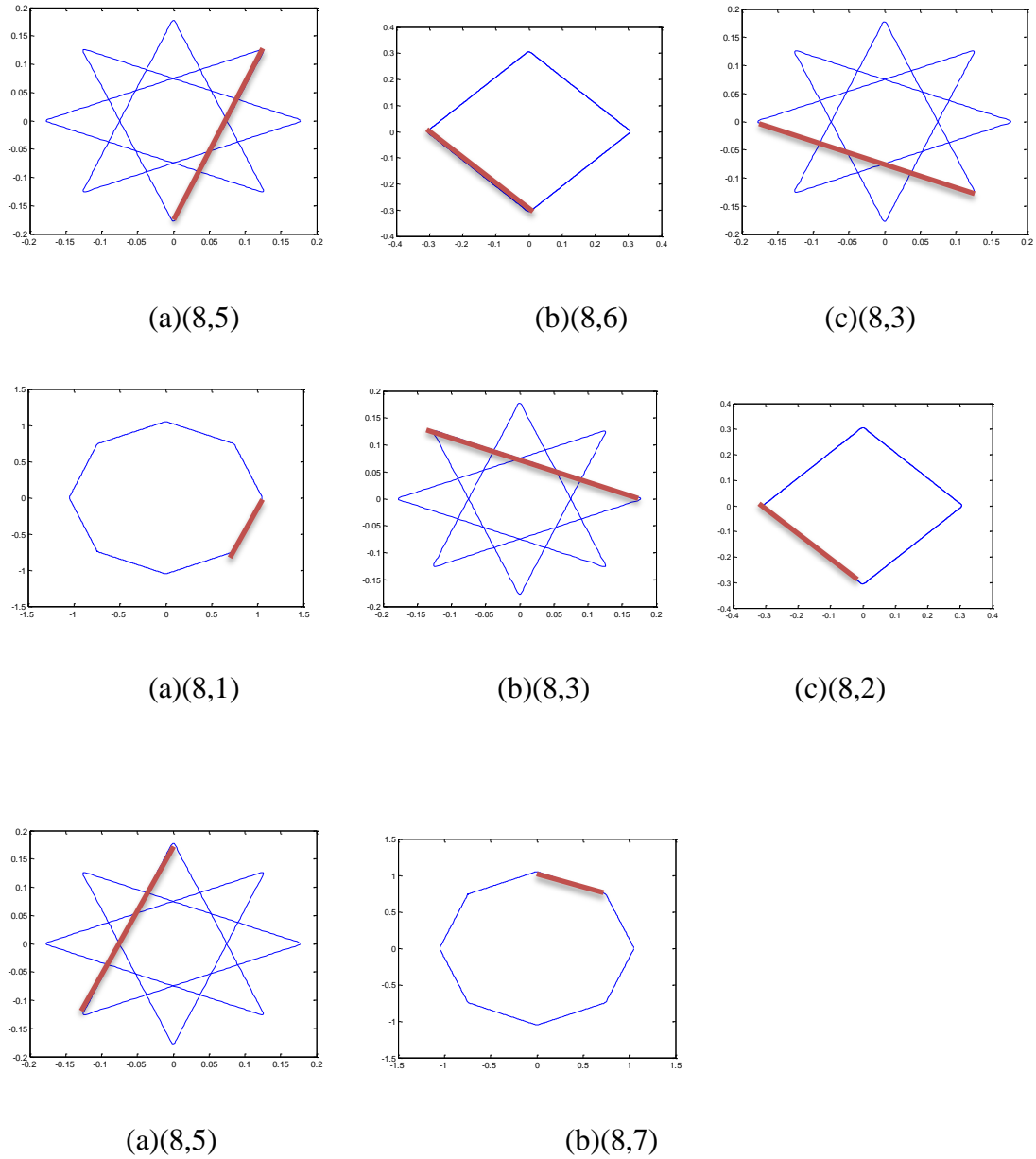


Figure 40 The regular polygons used to construct irregular polygons.

Tuning the parameters of our U matrix, more complicated entangled quantum states could be modeled as shown in Figure 41 [16, 31].

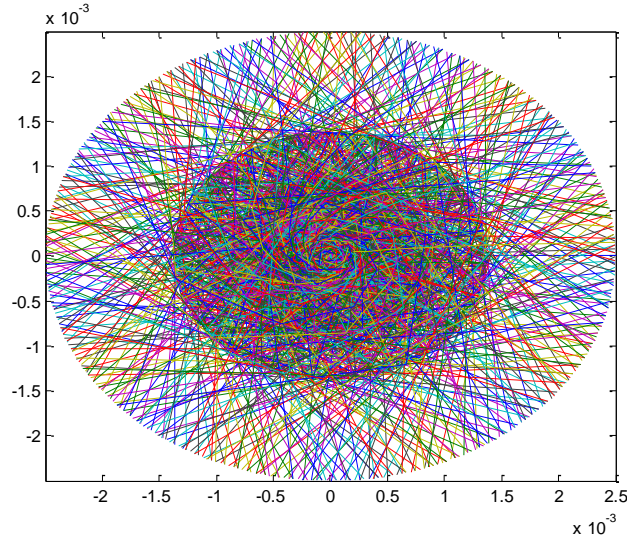


Figure 41 U matrix and entangled quantum states

Noisy communication channel representation for quantum computing

In this section, we summarize the connection between entangled quantum states and the noisy communication channel based on the analysis in the previous sections.

Based on our research group's analysis [15-27, 31-33], there exist regular or irregular Fourier polygon representations for the noisy communication channel in the form of an error content graph. For the noisy communication channel in permutation form $\begin{bmatrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \\ 2 & 7 & 5 & 8 & 1 & 4 & 6 & 3 \end{bmatrix}$, the corresponding Shannon's representation of the noisy communication channel and the Fourier polygon representation are shown in Figure 42.

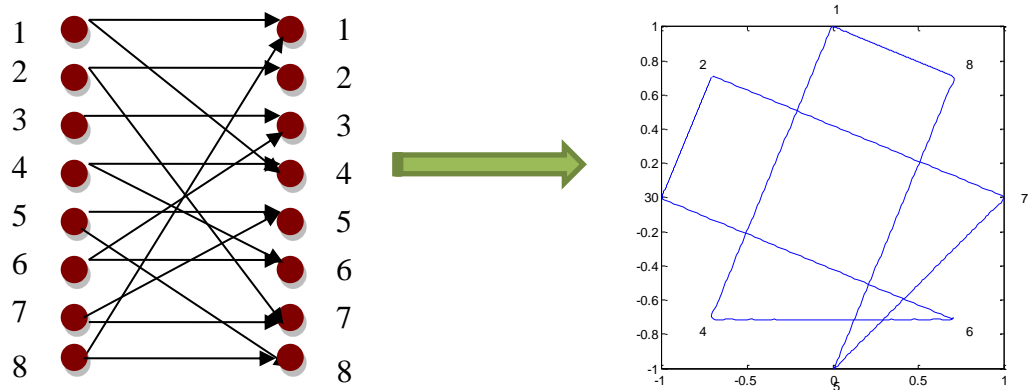


Figure 42 Shannon's representation and Fourier polygon representation of noisy communication channel

The relationship of the noisy communication channel, entangled quantum states and polygons are shown in Figure 43. The connection lies in the following aspects: 1) Shannon's noisy communication channel can be represented as polygons; 2) there is a geometric connection between the entangled quantum states and polygons; 3) the probability amplitude expression in a complex plane can be represented by the edges of a Fourier polygon; 4) applying Fourier analysis, the noisy communication channel can be represented as quantum states.

As illustrated in Appendix B, the noisy communication channel has been modeled with common tool sets: combinatorial analysis in terms of permutation, Fourier analysis in terms of Fourier series, and complex analysis in terms of an error content graph. Analysis of entangled quantum states with Fourier series had been illustrated in the previous sections. It is observable that we can apply combinatorial analysis, and complex analysis to the entangled quantum states.

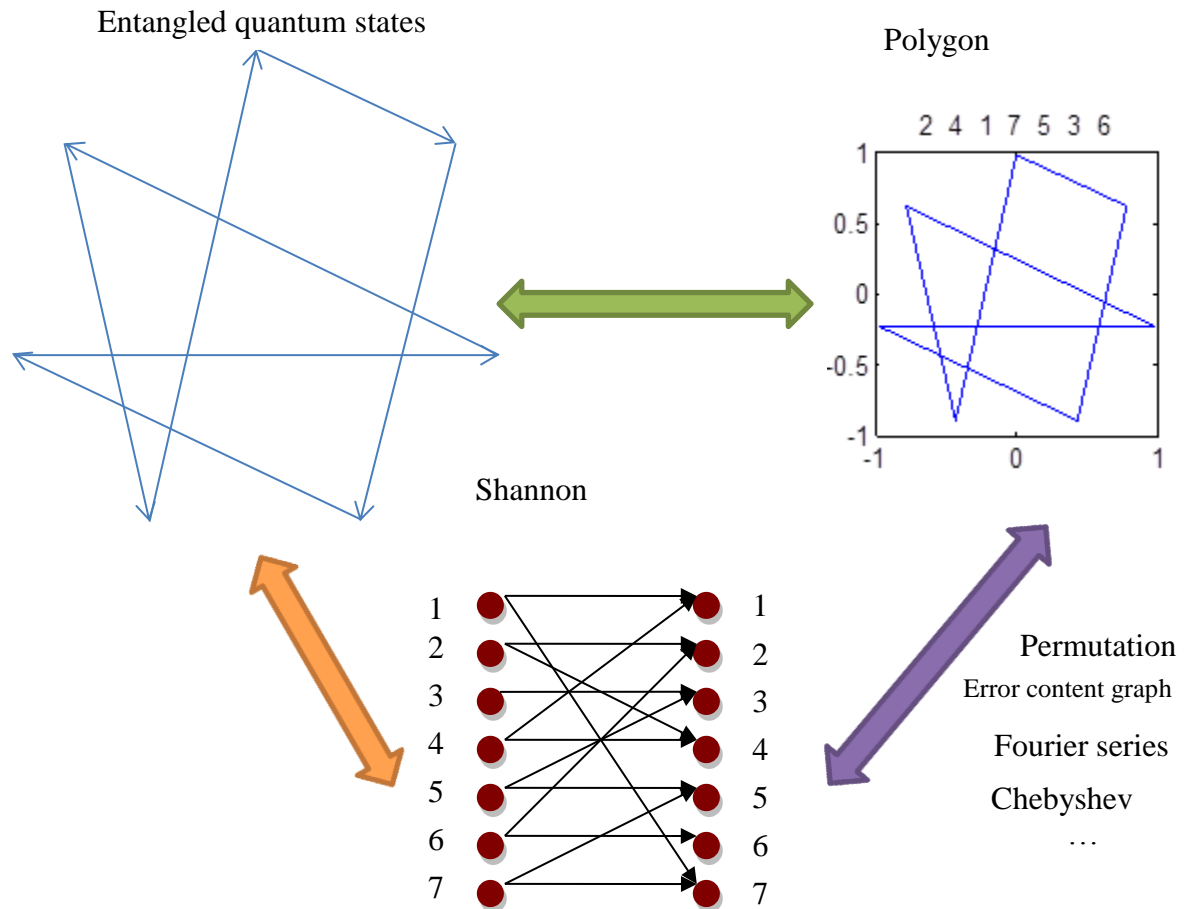


Figure 43 Relationship of probability amplitude and communication channel

Multiple-input Multiple-output (MIMO) channels representation of quantum computing

A baseband discrete-time model for a flat fading MIMO channel is shown in Figure 44. As this model depicts, there are N signals transmitted from N transmitters simultaneously. In a wireless communication system, the N signals of the MIMO system are with M outputs. Each transmitted signal arrives at each of the M receivers through the wireless channels. Each output of the channel is a linear superposition of the faded version of the inputs added by noise [75].

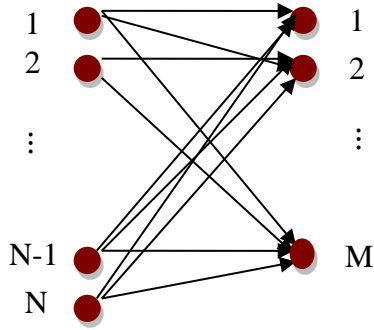


Figure 44 A multiple-input multiple-output channel

The signal $r_{t,m}$, which is received at time t at antenna m , is in the form of

$$r_{t,m} = \sum_{n=1}^N \alpha_{n,m} C_{t,n} + N_{1,m}$$

where $N_{1,m}$ is the noise sample of the receiver antenna m at time t .

The input-output relationship could be extended to the form of

$$R = C \times H + N$$

where C is a matrix that represents the signals transmitted from N transmit antennas

during T time slots:

$$C = \begin{bmatrix} C_{1,1} & C_{1,2} & \cdots & C_{1,N} \\ C_{2,1} & C_{2,2} & \cdots & C_{2,N} \\ \vdots & \vdots & \ddots & \vdots \\ C_{T,1} & C_{T,2} & \cdots & C_{T,N} \end{bmatrix}$$

R is a $T \times M$ matrix constructed to represent all the received signals in T time slots:

$$R = \begin{bmatrix} R_{1,1} & R_{1,2} & \cdots & R_{1,M} \\ R_{2,1} & R_{2,2} & \cdots & R_{2,M} \\ \vdots & \vdots & \ddots & \vdots \\ R_{T,1} & R_{T,2} & \cdots & R_{T,M} \end{bmatrix}$$

H is the path gains in a $N \times M$ channel matrix

$$H = \begin{bmatrix} \alpha_{1,1} & \alpha_{1,2} & \cdots & \alpha_{1,M} \\ \alpha_{2,1} & \alpha_{2,2} & \cdots & \alpha_{2,M} \\ \vdots & \vdots & \ddots & \vdots \\ \alpha_{N,1} & \alpha_{N,2} & \cdots & \alpha_{N,M} \end{bmatrix}$$

N is the $T \times M$ noise matrix

$$N = \begin{bmatrix} N_{1,1} & N_{1,2} & \cdots & N_{1,M} \\ N_{2,1} & N_{2,2} & \cdots & N_{2,M} \\ \vdots & \vdots & \ddots & \vdots \\ N_{T,1} & N_{T,2} & \cdots & N_{T,M} \end{bmatrix}$$

The MIMO model linear equation could be applied to quantum computing. Recalling our

U matrix

$$u_{n,q}(p, t) = \begin{bmatrix} \frac{e^{j[q+n(-p-p)]t_1}}{[q+n(-p-p)]^2} & \frac{e^{j[q+n(-p-p)]t_{r-1}}}{[q+n(-p-p)]^2} & \frac{e^{j[q+n(-p-p)]t_r}}{[q+n(-p-p)]^2} \\ \frac{e^{j[q+n(-p+1-p)]t_1}}{[q+n(-p+1-p)]^2} & \frac{e^{j[q+n(-p+1-p)]t_{r-1}}}{[q+n(-p+1-p)]^2} & \frac{e^{j[q+n(-p+1-p)]t_r}}{[q+n(-p+1-p)]^2} \\ \vdots & \vdots & \vdots \\ \frac{e^{j[q+n(-1-p)]t_1}}{[q+n(-1-p)]^2} & \frac{e^{j[q+n(-1-p)]t_{r-1}}}{[q+n(-1-p)]^2} & \frac{e^{j[q+n(-1-p)]t_r}}{[q+n(-1-p)]^2} \\ \frac{e^{j[q+n(0-p)]t_1}}{[q+n(0-p)]^2} & \cdots & \frac{e^{j[q+n(0-p)]t_r}}{[q+n(0-p)]^2} \\ \frac{e^{j[q+n(1-p)]t_1}}{[q+n(1-p)]^2} & \frac{e^{j[q+n(1-p)]t_{r-1}}}{[q+n(1-p)]^2} & \frac{e^{j[q+n(1-p)]t_r}}{[q+n(1-p)]^2} \\ \vdots & \vdots & \vdots \\ \frac{e^{j[q+n(p-1-p)]t_1}}{[q+n(p-1-p)]^2} & \frac{e^{j[q+n(p-1-p)]t_{r-1}}}{[q+n(p-1-p)]^2} & \frac{e^{j[q+n(p-1-p)]t_r}}{[q+n(p-1-p)]^2} \\ \frac{e^{j[q+n(p-p)]t_1}}{[q+n(p-p)]^2} & \frac{e^{j[q+n(p-p)]t_{r-1}}}{[q+n(p-p)]^2} & \frac{e^{j[q+n(p-p)]t_r}}{[q+n(p-p)]^2} \end{bmatrix}$$

and we could rewrite the equation as

$$R = C \times U + N$$

Here U functions as a quantum register to store the information in the computation process, which is similar to the function of quantum gates. Since

measurement collapse superposition states, a quantum register could store the information with superposition states. Using U matrix as a quantum register provides a possible new technique to simulate and eventually build a quantum computer. This equation also corresponds to the U transformation we proposed earlier with a special case for $N=0$.

The plots of the U matrix and entangled quantum state results are shown in Figure 45. The entangled quantum states were randomly generated, and the inverse FFT of the computation result is really visually intriguing, which shows certain patterns from the pseudo randomness.

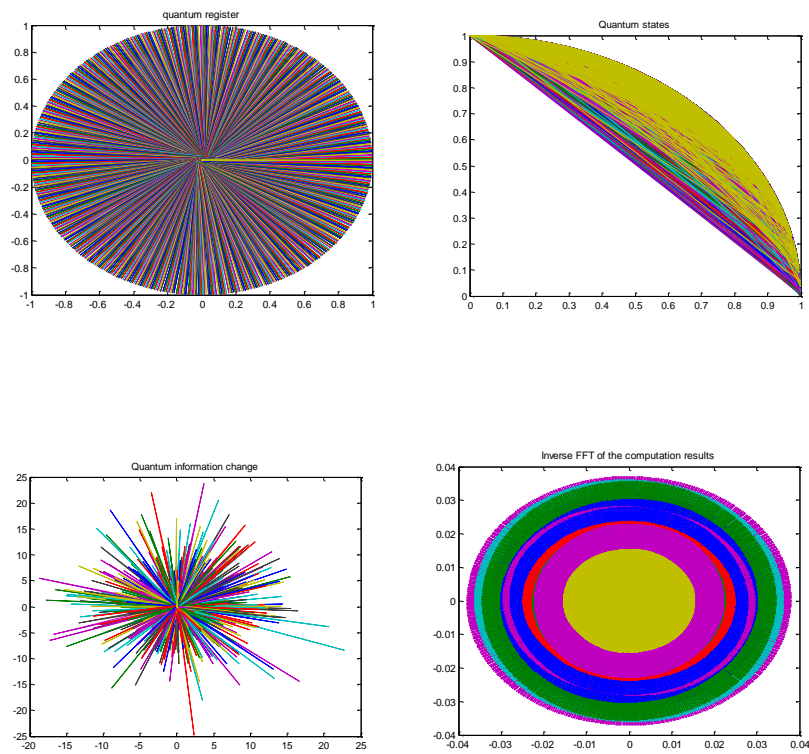


Figure 45 U matrix as a quantum register

4. THE EXTENDED NOISY COMMUNICATION CHANNEL FRAMEWORK

The GNCom approach improved the noisy communication framework by further associating quantum electrodynamics to the framework [31, 32]. Finally, GNCom results in an enhanced noisy communication channel framework to a broader range.

The extended noisy communication channel framework

Our research group initiated the research of communication based modeling [11-27, 31-33]. We approached systems from an information theoretical perspective and proposed a noisy communication channel framework for general system modeling. We proposed to model systems by noisy communication channels and analyzed systems by a structure of relationships whose elements could commonly be used as analysis tools in science and engineering. The formal mechanisms we integrated include: roots of unity, special permutations and polynomials, Fourier series, and uniform polygons. The framework connected the noisy communication channel with a series of mathematical analysis tools including combinational analysis, complex analysis, Chebyshev analysis and Fourier analysis, as shown in Figure 46 [11-15]. The noisy communication channel is connected to complex analysis through roots of unity and an error content graph. Also, the noisy communication channel is connected to combinatorial analysis through

permutation. Further, the noisy communication channel is connected to Chebyshev analysis through the Chebyshev matrix and polygon geometry. Finally, the noisy communication channel is connected to Fourier analysis through Fourier polygon representations.

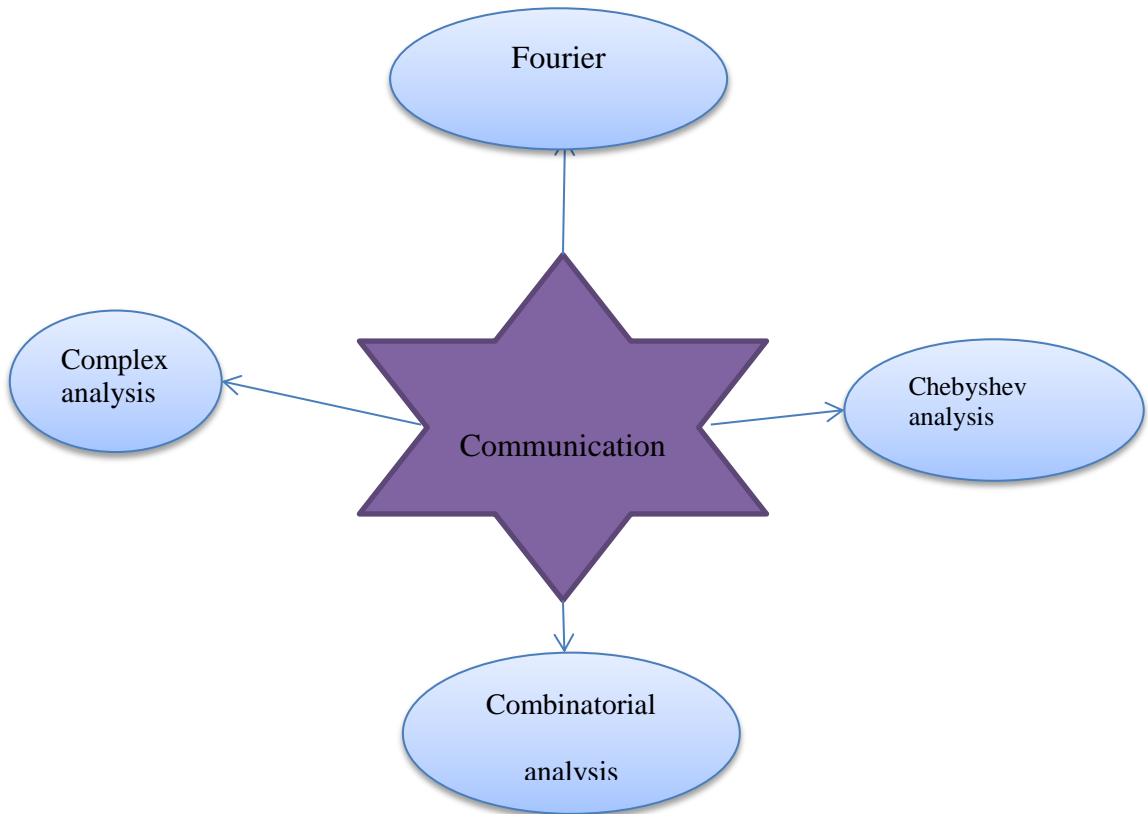


Figure 46 The noisy communication channel framework for general system modeling

The GNCom approach extended the noisy communication channel framework to the quantum computing field so that people can use it as a computation tool to model real-world problems. The extended framework is shown in Figure 47. One implication tool set of the GNCom approach was added to the framework: QED.

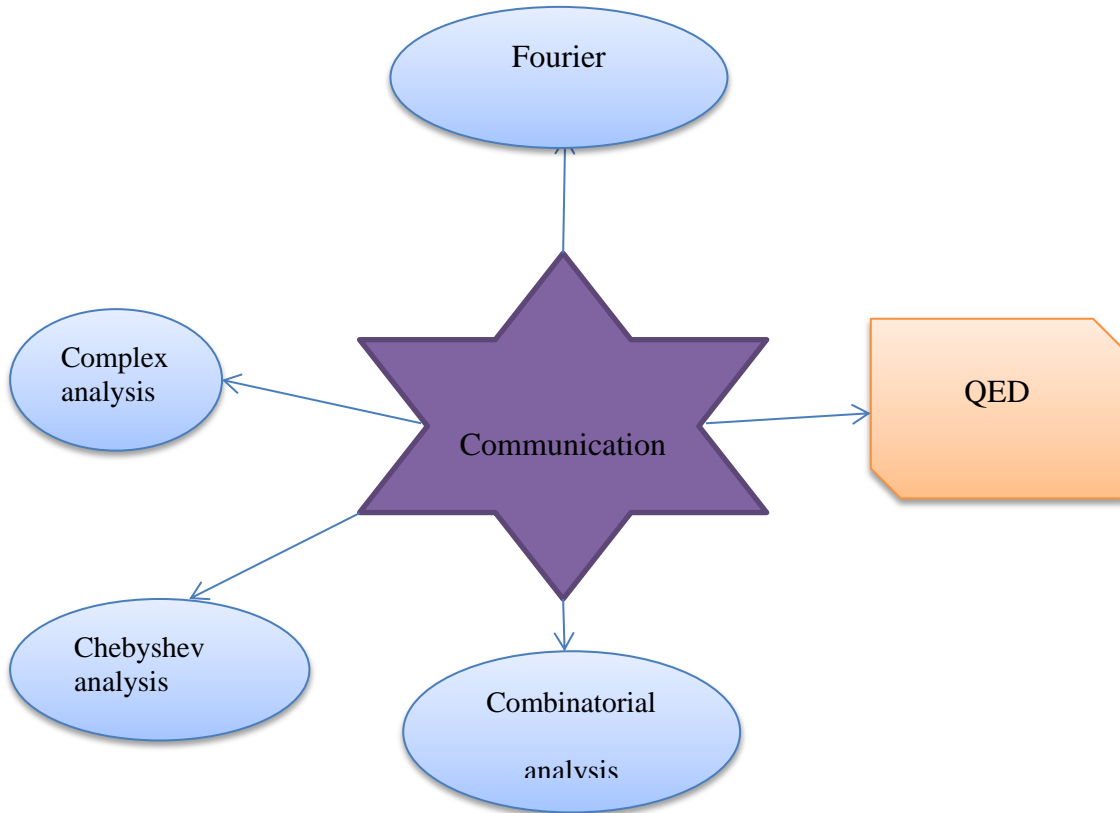


Figure 47 The extended noisy communication channel framework

QED

In this section, we will start with the introduction of QED, then Fourier analysis of QED, and follow with Amplitwist analysis of QED.

QED is short for quantum electrodynamics. QED is the quantum field theory of electrodynamics. It describes the interaction of light and matter. In particular, the QED theory talks about the interaction of photon and electron. Photons and electrons “move” from one point to another. Electrons can emit and absorb photons. The QED theory calculates the probabilities of this phenomenon happening.

One of the founders of QED, Richard Feynman, proposed three basic visual elements to represent photon, electron, emit and absorb, which are called Feynman

diagrams, as shown in Figure 48 [76]. In this figure, a wavy line represents photon; a straight line represents electron; and, two straight lines and a wavy line represent a photon emission or absorption by an electron.

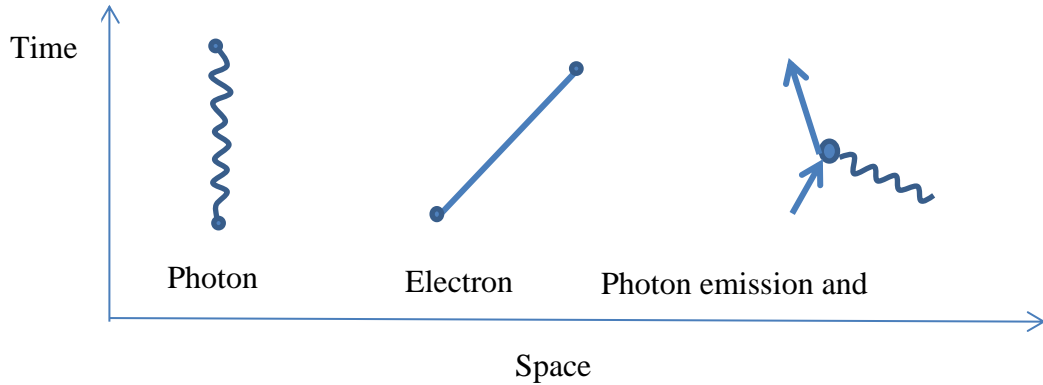


Figure 48 Feynman diagram for photon and electron

Light partial reflection

We start explaining QED with a series of light partial reflection experiments. Light partial reflection is a very commonly observable phenomenon in our daily life. For example, if someone is standing in a room and looking at the view outside through the window, he will see the view outside; and, he will also see part of his body image shown on the window glass. This is a light partial reflection phenomenon [76].

Monochromatic Light is emitted from a source to the surface of a block of glass as shown in Figure 49. In an experiment, a photon detector was placed at point A above the surface to detect the number of photons reflected. Based on the experiment, an average of 4% photons were detected at point A, and the other 96% got through.

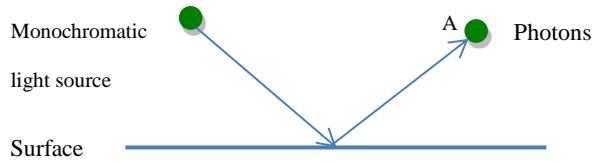


Figure 49 Light partial reflection experiment with one surface

The experiment also considered the effects on the back surface of the glass block, as shown in Figure 50. Logically thinking, we should measure 4% of the photons from the front surface and another 4% of the remaining 96% of the photons from the back surface at point A. That is approximately 8% of the photons. However, the results were quite surprising. The photons measured at point A changed from 0 to a maximum value of 16% periodically, as shown in Figure 50.

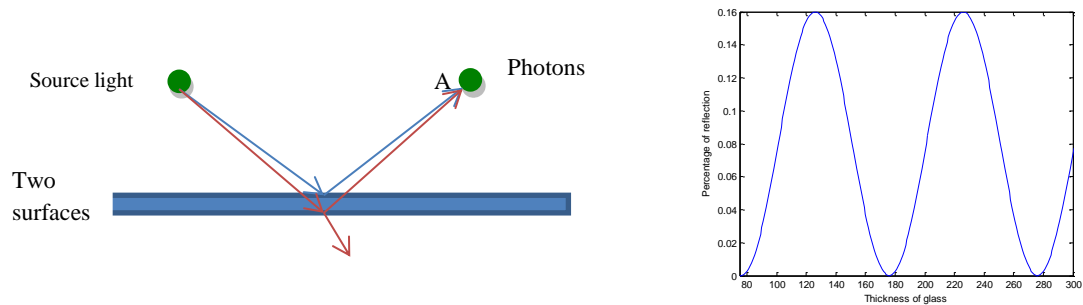


Figure 50 Light partial reflection experiment with two surfaces

From Figure 50, we can see the continuous nature of the partial reflection. This phenomenon cannot be explained with any other theory but QED [76].

Probability Amplitude

Feynman explained the calculation of the probability of photons moving by a series of light partial reflection experiments [76]. In an experiment to measure the partial

reflection of light, it was found that only 4% of the photons were reflected by a single surface, while the other 96% were transmitted by the front surface. In order to explain this feature, more experiments were carried out by adding the number of surfaces and changing the thickness of the surface. After repeating these experiments a number of times, it was found that the photons reflected by the first surface changes periodically from 0 to a maximum value of 16%.

In order to calculate the probability of the phenomenon, we define the problem as follows: Light reflects from two surfaces, and there are two possible routes: one route is from the source to point A from the front surface; the other route is from the source to point B from the back surface. The probability of reflection can be calculated from the two surfaces.

The resultant probability can be calculated by adding the probabilities of the two routes. We use the concept of probability amplitude to do the addition. The probability of an event is equal to the square of the length of an arrow representing the probability amplitude. We represent each route with an arrow, and the length of the arrow is the square root of the probability of the event. For example, in the experiment described in section II, the probability of light reflection of a single surface is 4%, and the length of the arrow is 0.2, as shown in Figure 51.

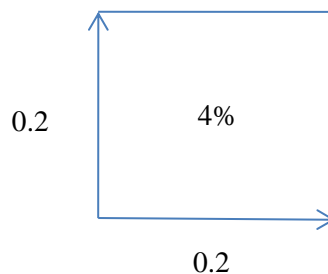


Figure 51 Probability representation of arrow length

A similar rule applies to the second surface. The probability of reflection from the second surface is also 4%, and the length of the arrow is 0.2. When adding the two arrows, vector algebra is utilized. The resultant probability amplitude arrow of the event must be squared to calculate the probability.

After defining the length of the arrow, one needs to define the direction of the arrow. Feynman adopted an external imaginary stopwatch to define the direction of the arrow. When the light leaves the source, we start the stopwatch. When it reaches the photon detector, we stop the stopwatch. The direction of the arrow will point in the direction of the stopwatch hand if the arrow represents a back surface reflection. For the front surface, the direction of the arrow will be opposite to that of the stopwatch hand.

The direction of the arrow is related to the thickness of the reflection surfaces, or, more specifically, it reflects the time. See Figure 52 for an example. In this example, the two arrows are with a small angle. Based on the law of cosines, the length of the third arrow is calculated; thus, we can obtain the final probability by the square of the final arrow length.

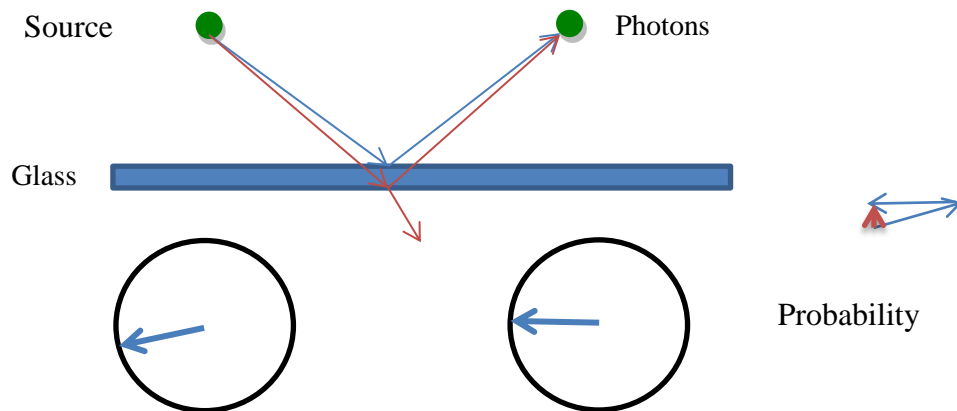


Figure 52 Partial reflection event and the probability amplitude

When vectors are added, the sum of the vectors shows the probability. An example of two surfaces in Figure 53 shows how it works. Within the example shown, upon adding the two arrows with length 0.2 and making a small angle, the probability amplitude of the final product, based on the law of cosines, is calculated as follows:

$$\sqrt{0.2^2 + 0.2^2 - 2 \times 0.2 \times 0.2 \times \cos(0.2507)} = 0.05$$

Thus, the probability is $0.05^2 = 0.0025$.

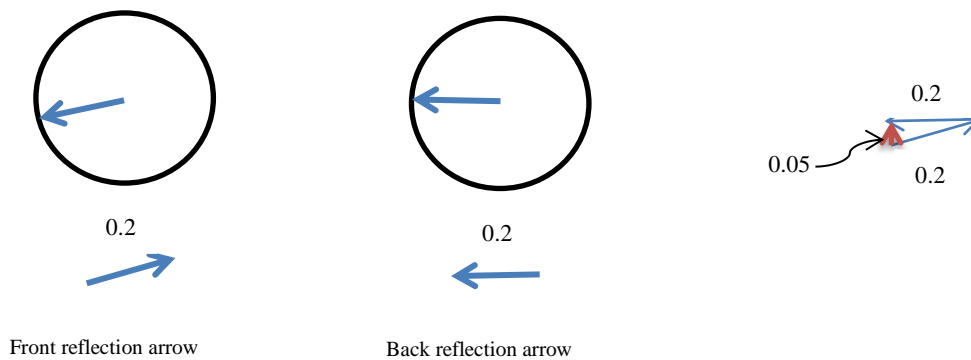


Figure 53 The final probability of photons reflected by two surfaces

Based on the thickness of the glass, the probability change from 0 to maximum 16% is shown in Figure 54. When the two arrows are in opposite directions, they cancel out, and the final probability amplitude is 0, which results in a probability of 0. When the two arrows are in the same direction, the probability amplitude is of the maximum 0.4, which results in a probability of 16%. All the other probabilities are the cases between.

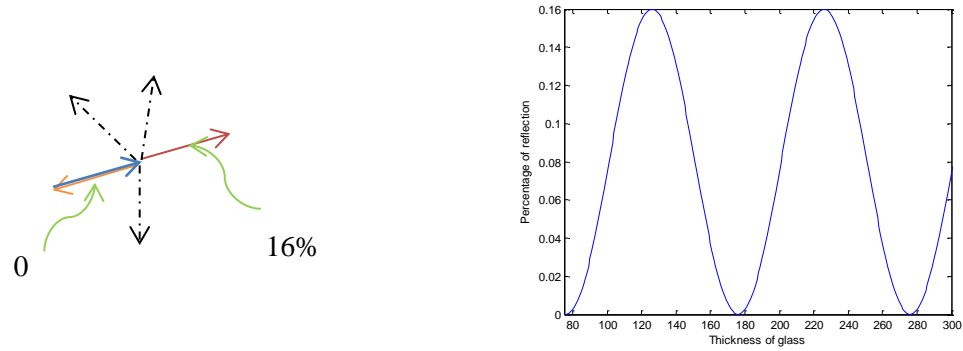


Figure 54 The probability change based on the thickness of the glass for reflection

All the cases discussed in previous sections are related to two surfaces. But two surfaces cannot cover all the cases in the real world. Rather, there can be multiple surfaces; also the thickness of the glass can also change. We expand the two surface cases to the probability of multiple surfaces. With expansion of the addition of two vectors, the probability of multiple surfaces can be calculated with similar vector algebra, as shown in Figure 55. The final probability amplitude can be drawn by connecting the tail of the first arrow to the head of the last arrow. These multiple probability amplitudes shed light on our research. Importantly, the probability amplitude could be mapped to a complex plane and be represented as complex numbers. All the connection points could be mapped to a circle, and we could further study this phenomenon with analysis tools like complex analysis and Fourier analysis [31-33].

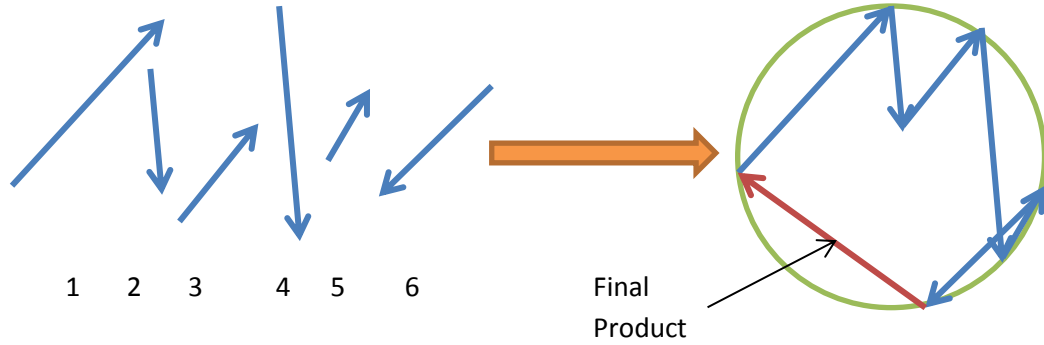


Figure 55 Addition of multiple probability amplitudes

Fourier series and QED

In this section, we will introduce the connections of Fourier series and QED by removing the external stopwatch concept. Instead we integrate space, phase, and time domain by introducing an internal “clock” using Fourier series. Let us recall Fourier polygons and its noisy communication channel representation. As can be seen intuitively, there is a connection between the probability amplitude of QED and the Fourier series representation of polygons. Introducing Fourier series to the amplitude probability calculation, we could conceptually integrate an internal clock to the event. Note that in the Fourier series representation of polygon,

$$0 \leq t_0 < t_1 < \dots < t_i < t_{i+1} < \dots < t_{n-1} < n = t_0 + 2\pi.$$

The approach to calculate probability amplitude can be analyzed using Fourier series polygon representations; that is, each arrow corresponds to a value of the Fourier series by introducing the internal clock. Therefore, we can see the analogy of the quantum electrodynamics and the communication channel model. More specifically we can use the communication channel model to analyze quantum electrodynamics theory [31-33].

When we apply our approach to the light partial reflection phenomenon, we will show that the probability amplitudes can be represented as a noisy communication channel. The entanglement of the probability amplitudes is considered as a group of communication channels. In detail, we can model the light partial reflection phenomenon as Fourier polygons of the noisy communication channel on the complex plane.

Let us recall the Fourier series equations for the noisy communication channel

$$u(l, t) = \sum_{l=-p}^p \frac{e^{j(q+nl)t}}{(q + nl)^2}$$

Tuning the parameter of this equation, any probability amplitude can be represented as a piece of a Fourier polygon.

Example 4. The probability amplitude calculation shown in Figure 56, can be connected to the irregular Fourier polygon, which is formed by parts of regular polygons.

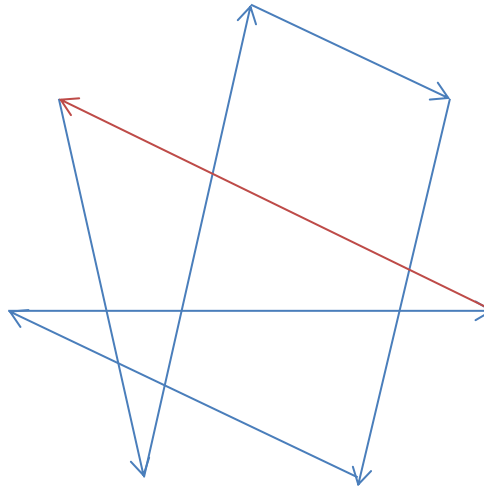


Figure 56 Probability amplitudes of 6 surfaces

The probability amplitude corresponds to an irregular Fourier polygon formed by the

permutation $\begin{matrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 \\ 2 & 4 & 1 & 7 & 5 & 3 & 6 \end{matrix}$, as shown in Figure 57.

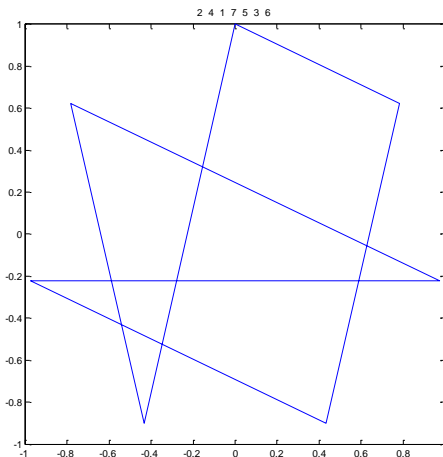
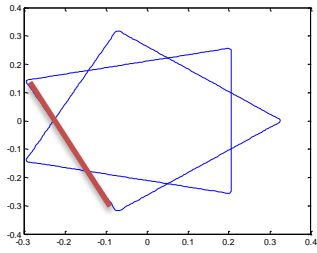


Figure 57 Irregular polygon formed by permutation matrix with 7 elements

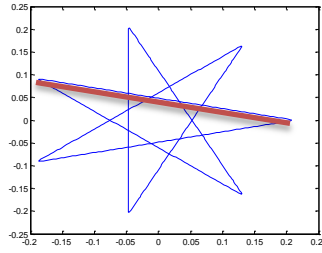
Following the procedure described earlier, the jump speed is $[2 -3 6 -2 -2 3 -4]$, and it turns out to be $[2 4 6 5 5 3 3]$ after mod 7. The irregular polygon is formed by pieces from the regular polygons:

$$u_{7,2}(t), u_{7,4}(t), u_{7,6}(t), u_{7,5}(t), u_{7,5}(t), u_{7,3}(t), u_{7,3}(t).$$

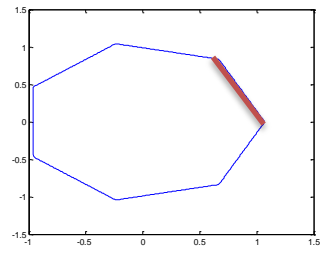
The regular polygons and the pieces are shown in Figure 58 with the pieces marked.



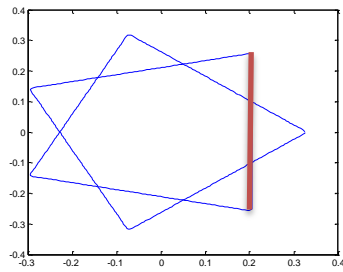
(a)(7,2)



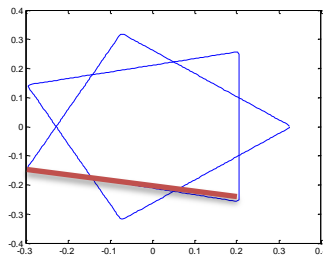
(b)(7,4)



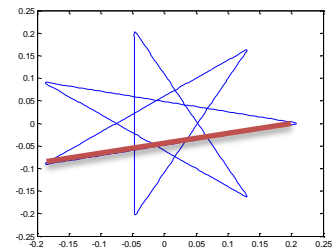
(c)(7,6)



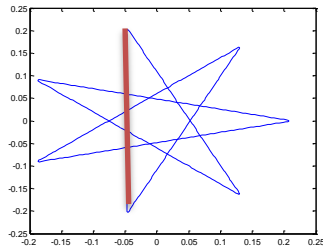
(d)(7,5)



(e)(7,5)



(f)(7,3)



(g)(7,3)

Figure 58 Regular polygons used to construct the irregular polygon with 7 vertices

As can be seen from the analysis, the probability amplitude could be directly broken into pieces taken out from regular polygons.

Probability amplitude representation of noisy communication channel

As illustrated in Chapter 1, the noisy communication channel could be represented as polygons. Each edge of the polygon is part of a communication channel. By introducing Fourier series analysis, each edge of the polygon could be represented as probability amplitude, and the frequency can be varied. The relationship of Richard Feynman’s probability amplitude in green and the communication channel “probability amplitude” in red are shown in Figure 59.

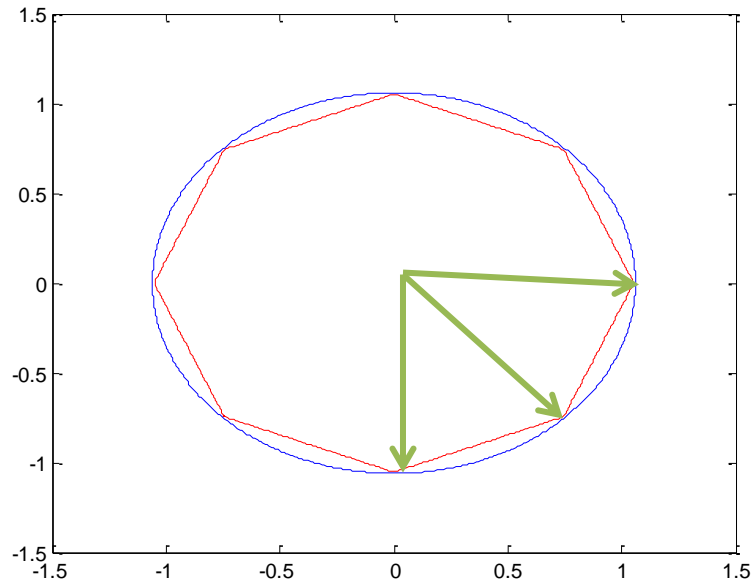


Figure 59 Richard Feynman’s probability amplitude and communication channel “probability amplitude”

Polygon geometry and probability amplitude

In this section, we analyze the connection of QED, or probability amplitude and Fourier representation of polygons. Let us recall Feynman’s “arrow”—the probability amplitude, the connection point of the arrow and the clock, or the “circle” with roots of unity or vertices of a Fourier polygon based on our equations. There is a connection

between the magnitude of the probability amplitude and the length of the Fourier polygon edge. Let us take the regular polygon with 8 vertices for example. Figure 60 shows the edges of the regular polygons $\{\frac{8}{1}\}, \{\frac{8}{2}\}, \{\frac{8}{3}\}$, which correspond to AB, AC and AD, respectively.

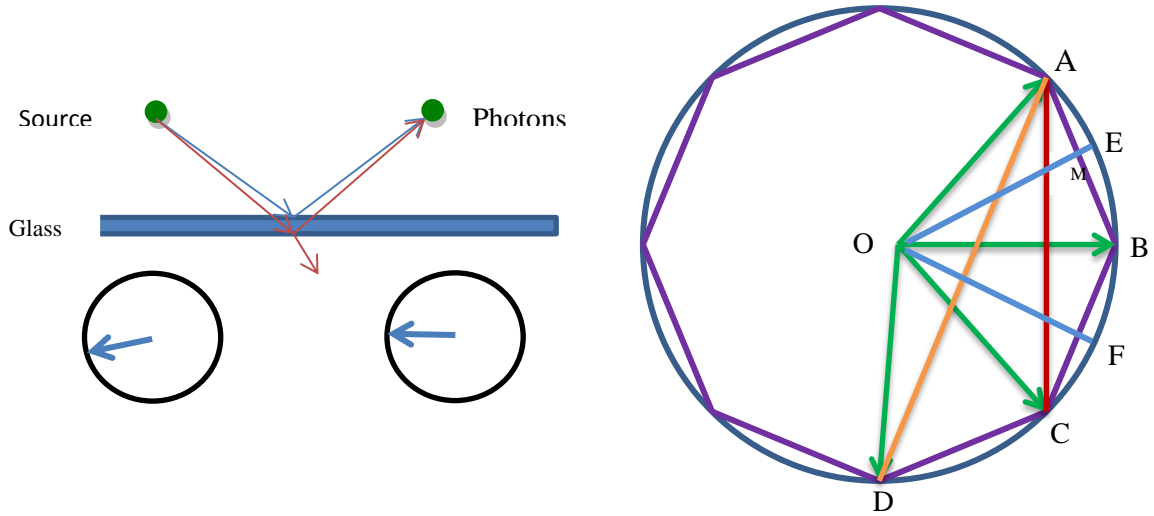


Figure 60 Polygon geometry and probability amplitude

Similar to the polygon edge and radius ratio derivation, we can derive the relationship of the magnitude of the probability amplitude and the edge length of the Fourier polygon. A, B, C, D are equally distributed throughout the circle, and they correspond to roots of unity. It is observed that

$$\widehat{AB} = \widehat{BC} = \widehat{CD} = \frac{2\pi}{n}$$

Triangle OAB is an isosceles triangle, and with E as the middle point of A and B, OM is the altitude of triangle AOB, and OM is perpendicular to AB. Thus we have

$$\angle AOM = \angle BOM = \frac{1}{2} \angle AOB = \frac{1}{2} \frac{2\pi}{n} = \frac{\pi}{n}$$

Let us look at right triangle AOM,

$$\overline{OA} \sin \angle AOM = \overline{OA} \sin \frac{\pi}{n} = \overline{AM} = \frac{1}{2} \overline{AB}$$

Thus, we can obtain

$$\overline{OA} = \frac{1}{2} \frac{\overline{AB}}{\sin \frac{\pi}{n}}$$

Applying similar analysis to isosceles triangles OAC and OAD, we have

$$\overline{OA} = \frac{1}{2} \frac{\overline{AC}}{\sin \frac{2\pi}{n}}$$

$$\overline{OA} = \frac{1}{2} \frac{\overline{AD}}{\sin \frac{3\pi}{n}}$$

Since \overline{AB} , \overline{AC} , and \overline{AD} correspond to the edges of the regular polygon $\left\{ \begin{matrix} n \\ q \end{matrix} \right\}$ in the case of $n=8$, $q=1, 2, 3$ in the example, let us use l_q to replace \overline{AB} , \overline{AC} , and \overline{AD} , and we can have a universal equation

$$\overline{OA} = \frac{1}{2} \frac{l_q}{\sin \frac{q\pi}{n}}$$

Since \overline{OA} is the magnitude of probability amplitude, we substitute it with $|PA|$, and we have

$$|PA| = \frac{1}{2} \frac{l_q}{\sin \frac{q\pi}{n}}$$

Applying complex analysis, we can represent the probability amplitude on a complex plane,

$$\overrightarrow{OA} = \frac{1}{2} \frac{l_q}{\sin \frac{q\pi}{n}} e^{i \frac{2\pi}{n}}$$

Tuning the parameters of the proposed Fourier polygon equation, we can obtain any regular polygon and thus we can represent all the probability amplitudes. In this example, by further applying the amplitwist concept, we can obtain the complex expression of all probability amplitudes by twisting the angle of $\frac{2\pi}{n}$ with the same magnitude from the previous probability amplitude. More specifically here, $\overrightarrow{OB} = \overrightarrow{OA}e^{i\frac{2\pi}{n}}, \overrightarrow{OC} = \overrightarrow{OB}e^{i\frac{2\pi}{n}}, \dots$

Amplitwist analysis of QED

Let us recall the Amplitwist concept, Feynman's probability amplitude, and the imaginary stopwatch. In fact the imaginary clock Feynman adopted was a phaser. Let us look at the multiple probability amplitudes example first and then apply amplitwist analysis to it. We can map all the probability amplitudes into a complex plane, and arrow 1 and arrow 2 are mapped as an example, as shown in Figure 61. If we can discover the magnitude difference, or the amplification of arrow 1 and arrow 2, and the phase difference of arrow 1 and arrow 2, we can obtain an expression of the arrows by applying the amplitwist concept.

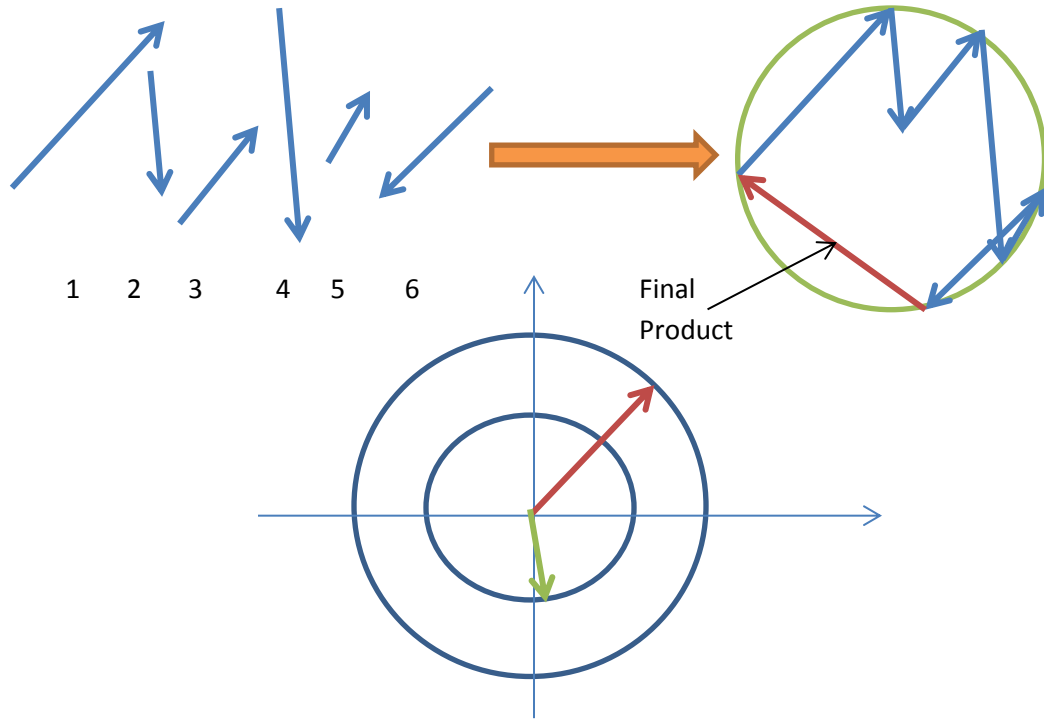


Figure 61 Amplitwist and multiple probability amplitudes

Let us extend amplitwist to more arrows, as shown in Figure 62. It is easy to calculate every arrow based on the amplification and rotation.

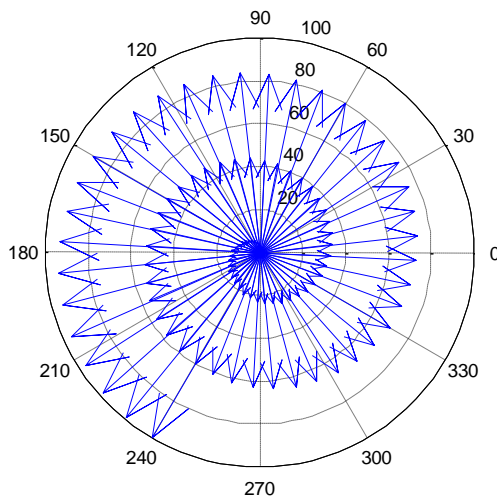


Figure 62 Amplitwist example with multiple vectors

As we explained in the previous section on the close relationship of probability amplitude and Fourier polygon geometry, we can easily model the behavior of probability amplitude by combining amplitwist analysis. An example is shown in Figure 63.

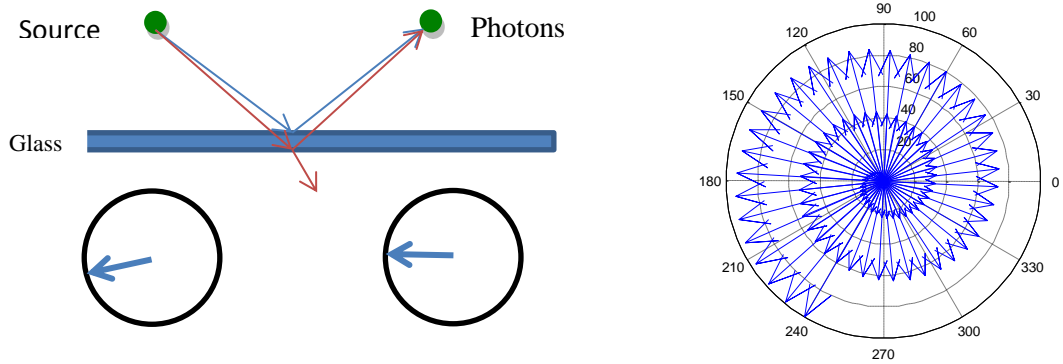


Figure 63 Amplitwist analysis of QED

As Feynman defined, the direction of the arrow was determined by an imaginary stopwatch. The frequency of the quantum clock rotation is

$$f = \frac{L}{h} = \frac{K - U}{h}$$

where h is the quantum of action called Planck constant, and L is called the Lagrangian. For single-particle moves at non-relativistic speeds, the Lagrangian is the difference between the Kinetic energy K and the potential energy U [76].

We consider a free particle with potential energy $U = 0$ and $L = K = \frac{1}{2}mv^2$; thus, the rate of arrow rotation is

$$f = \frac{L}{h} = \frac{K - U}{h} = \frac{mv^2}{2h}$$

Assume the length of the glass is d , and the wavelength of the light used in the experiment is λ . Over a time t , the photon goes through the first surface, incident to the second surface and bounce back to the first surface. During this period, the rotation of the arrow is the difference of the phase change between the first arrow and second arrow; the number of rotations can be represented as,

$$n = \frac{mv^2}{2h} t$$

On the other hand, the number of rotations can be represented as $\frac{2d}{\lambda}$; thus, we have

$$n = \frac{mv^2}{2h} t = \frac{2d}{\lambda}$$

Solving this equation, we have

$$t = \frac{4dh}{\lambda mv^2}$$

So the rotation angle is

$$\theta = 2\pi f t = \frac{4d\pi}{\lambda}$$

We demonstrated by example that by applying the polygon geometry, we can represent the probability amplitude. Further applying amplitwist, we can obtain the compute the next probability amplitude.

Generalized noisy communication channel approach for QED analysis

In this section, we summarize the connection between QED and the noisy communication channel based on the analysis in the previous sections. The relationship between the noisy communication channel, probability amplitudes and polygons are shown in Figure 64. The connection lies in the following aspects: 1) Shannon's

representation of the noisy communication channel could be represented as polygons; 2) there is a geometry connection of the probability amplitudes and polygons; 3) the probability amplitude expression in a complex plane could be mathematically represented by the edges of the Fourier polygon; 4) applying Fourier analysis, the noisy communication channel could be represented as probability amplitudes.

As illustrated in Appendix B, the noisy communication channel has been modeled with common tool sets: combinatorial analysis in terms of permutation, Fourier analysis in terms of Fourier series, and complex analysis in terms of error content graph. Analysis of QED with Fourier series has been illustrated in the previous sections. It is observable that we can apply combinatorial analysis and complex analysis to the QED. Furthermore, Fourier series representation of each “arrow” incorporates Feynman clock.

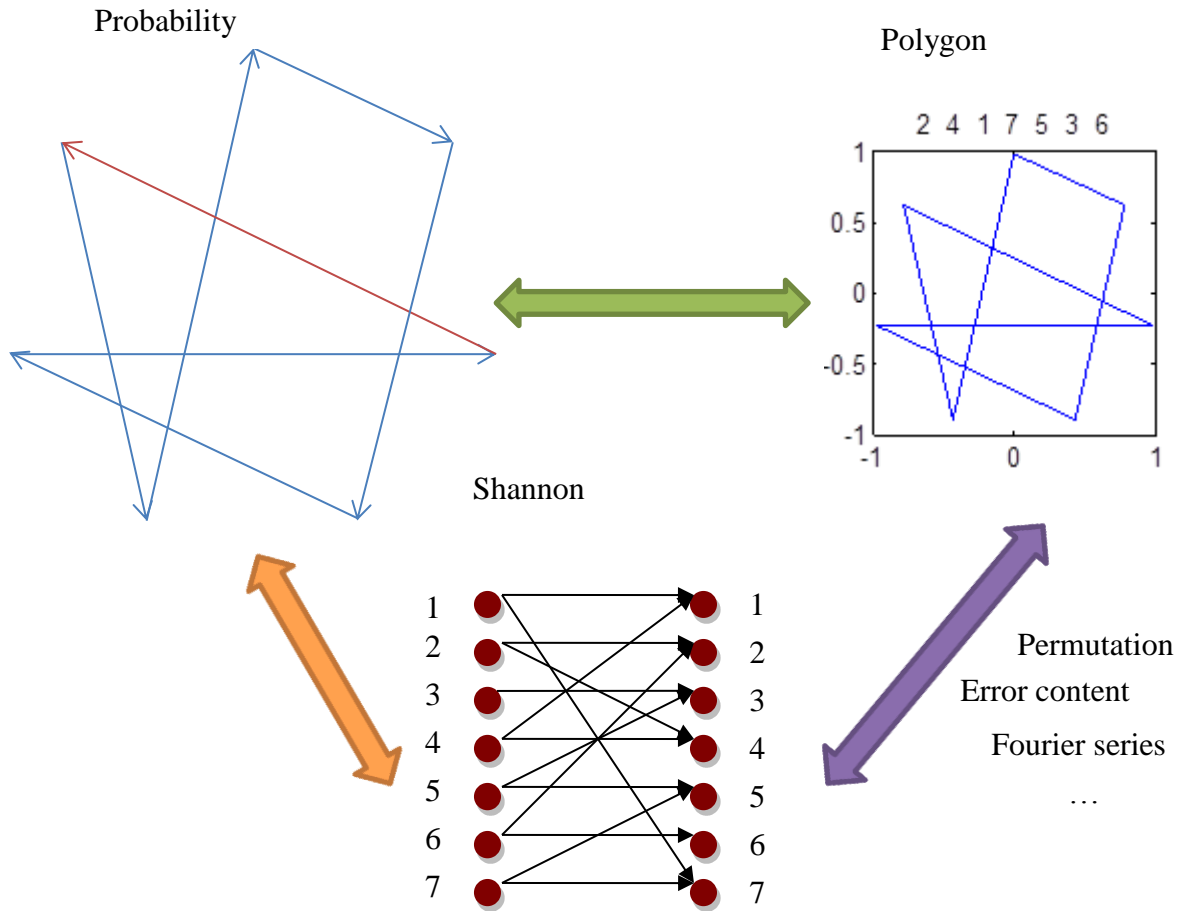


Figure 64 Relationship of probability amplitude and communication channel

QED and quantum computing

In addition, QED could be considered as a mechanism for quantum computing.

As shown in Figure 56, each arrow in the multiple probability amplitude diagram can be studied as a quantum state in the form

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$$

The multiple probability amplitude diagram can be studied as entangled quantum states in the form

$$|\psi\rangle = |\psi_1\rangle \otimes |\psi_2\rangle \otimes \dots \otimes |\psi_n\rangle$$

$$\psi\rangle = \sum_{x=1}^{2^n-1} c_x |x\rangle_n$$

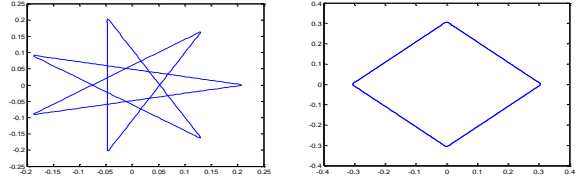
$$\sum_{x=1}^{2^n-1} |c_x| = 1$$

Quantum computer design technique based on QED and Quantum Fourier Series

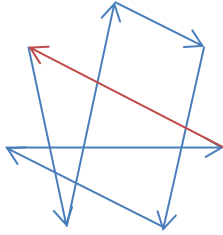
Even though quantum computer design is still at its infancy, researchers are mimicking the classical computer design technique starting from quantum gates [52, 53]. In this dissertation, we propose a quantum computer design technique inspired from our communication channel model. Let us recall the Fourier polygons and the entangled multiple probability amplitude shown in Figure 65. By introducing Fourier series to the communication channel, we proposed the U matrix and related transforms. Any polygons could be represented as noisy communication channel with the GNCom approach.

In terms of quantum computing, the entangled quantum states could be represented with the noisy communication channel based on the GNCom approach; also the proposed U matrix could further be utilized as a quantum register to store quantum information in quantum computing process as illustrated in chapter 3.

$$u_{n,q}(p, t) = \sum_{l=0}^{2p} \frac{e^{j[(q+n(l-p))t]}{[q + n(l - p)]^2}$$



(a) Fourier polygons



$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$$

$$|\psi\rangle = |\psi_1\rangle \otimes |\psi_2\rangle \otimes \dots \otimes |\psi_n\rangle$$

$$|\psi\rangle = \sum_{x=1}^{2^n-1} c_x |x\rangle_n \quad \sum_{x=1}^{2^n-1} c_x = 1$$

(b) Multiple entangled states

Figure 65 Fourier polygons and multiple probability amplitudes

The generalized noisy communication model approach provides an approach for quantum computing. Based on the approach, we further proposed a quantum computer design technique from the example of light partial reflection. We enhanced the noisy communication channel framework by connecting QED to the framework and applying new techniques like amplitwist and eigenvalues of Chebyshev matrix. The probability amplitude or the quantum state could be modeled by the polygon geometry, the polygon edge and circle radius ratio or the eigenvalues of Chebyshev matrix. Based on the generalized noisy communication model, we hypothesize that a quantum computer can be designed by shining light through multiple-layer glass or crystal materials if controlling the reflection coefficient is possible, as shown in Figure 66. Quantum computing can be done by masking or etching the material based on the communication channel model. Our future research will further investigate this technique.

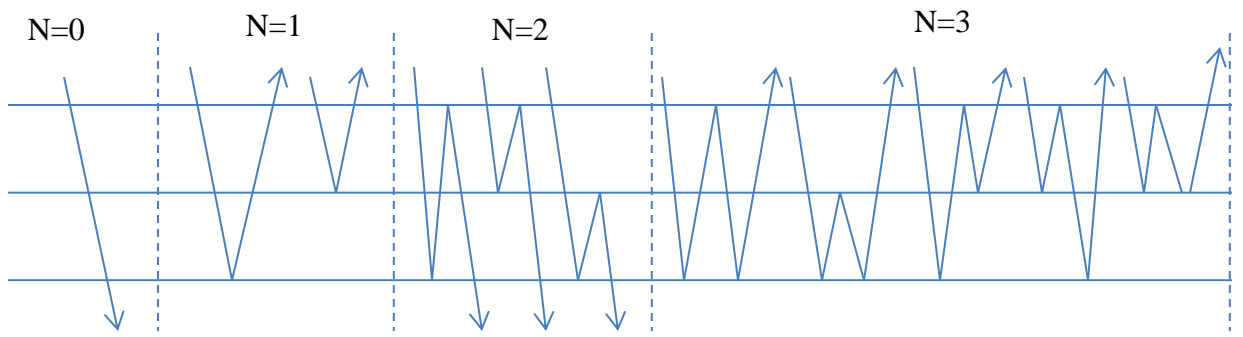


Figure 66 Paths of reflecting light

5. CONCLUSIONS

In this dissertation, we developed a generalized noisy communication channel approach and developed the noisy communication representation for quantum computing based on the GNCom approach. The GNCom improved the noisy communication channel framework by connecting QED analysis. Ultimately, the dissertation provides a computation tool for study purposes that can be used as a stepping stone for further research. It is also useful for education and interdisciplinary research purposes because it connects practical and complex concepts from multiple disciplines.

Future research

The U transform and Zeta transform we proposed show potential advantages visually, and we carried out preliminary experiments to apply the transforms to textbook engineering problems. The results were visually compared with Fast Fourier Transform. The U transform seemed to show more symmetry than FFT, but practical applications still needs to be further explored.

Engineering application of the transform

Fast Fourier Transform is very popular in engineering applications. One example of it is signal filtering. If we mix a couple of signals with different frequencies, FFT can successfully separate the frequencies, and the original signal with different frequencies

can be successfully restored by inverse FFT. Here we are going to show a signal filtering example with U transform and Zeta transform, then compare the results with FFT.

The original signal is a harmonic signal: $S = \sin(t) + \sin(2t) + \sin(2.5t)$. The signal and the corresponding FFT, U transform, and Zeta transform results in the frequency domain, are shown in Figure 67.

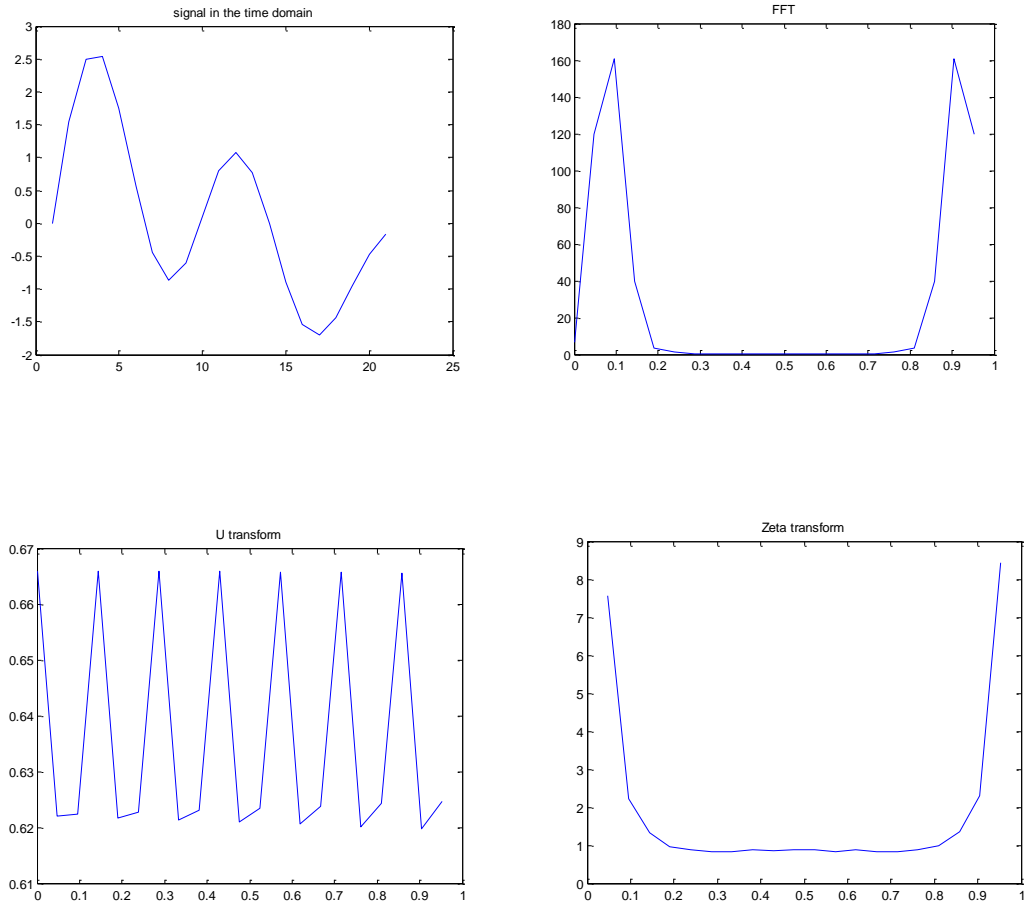


Figure 67 Harmonic signal and FFT, U transform, Zeta transform signal in the frequency domain

Another experiment was also carried out using a damping signal, and the results are shown in Figure 68. Due to the large computation cost of the Zeta transform, the Zeta transform result is not shown here.

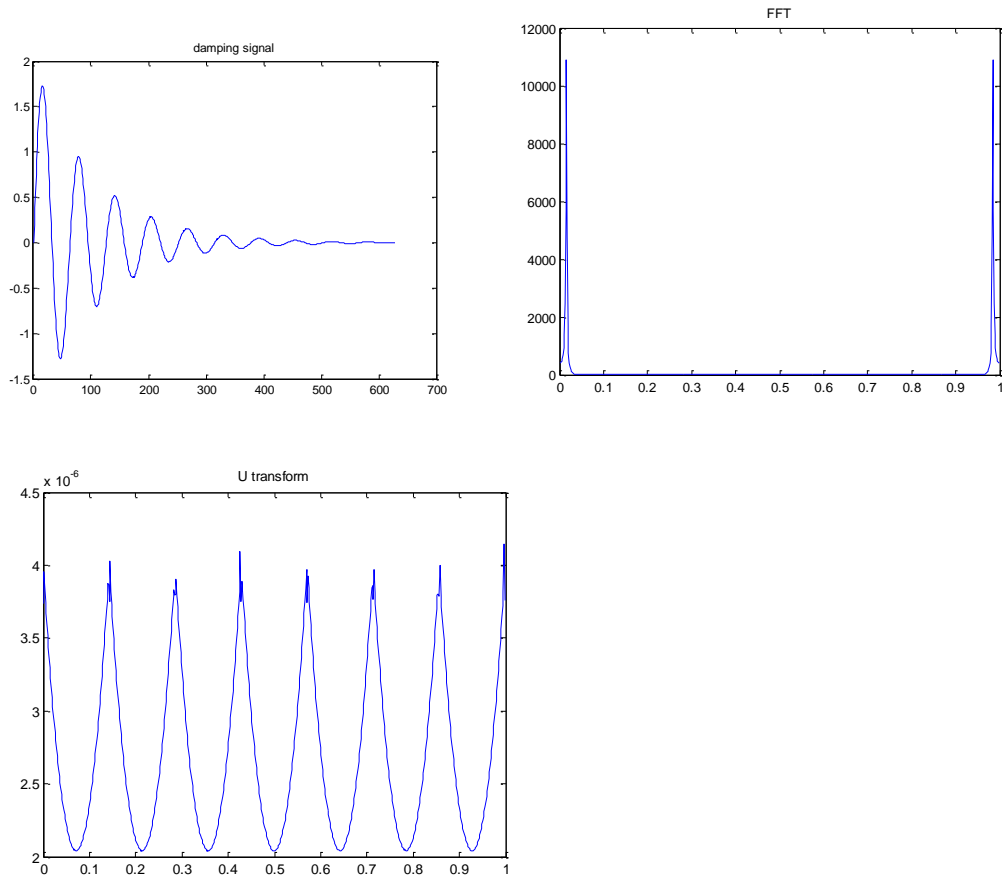


Figure 68 Damping signal and FFT, U transform in the frequency domain

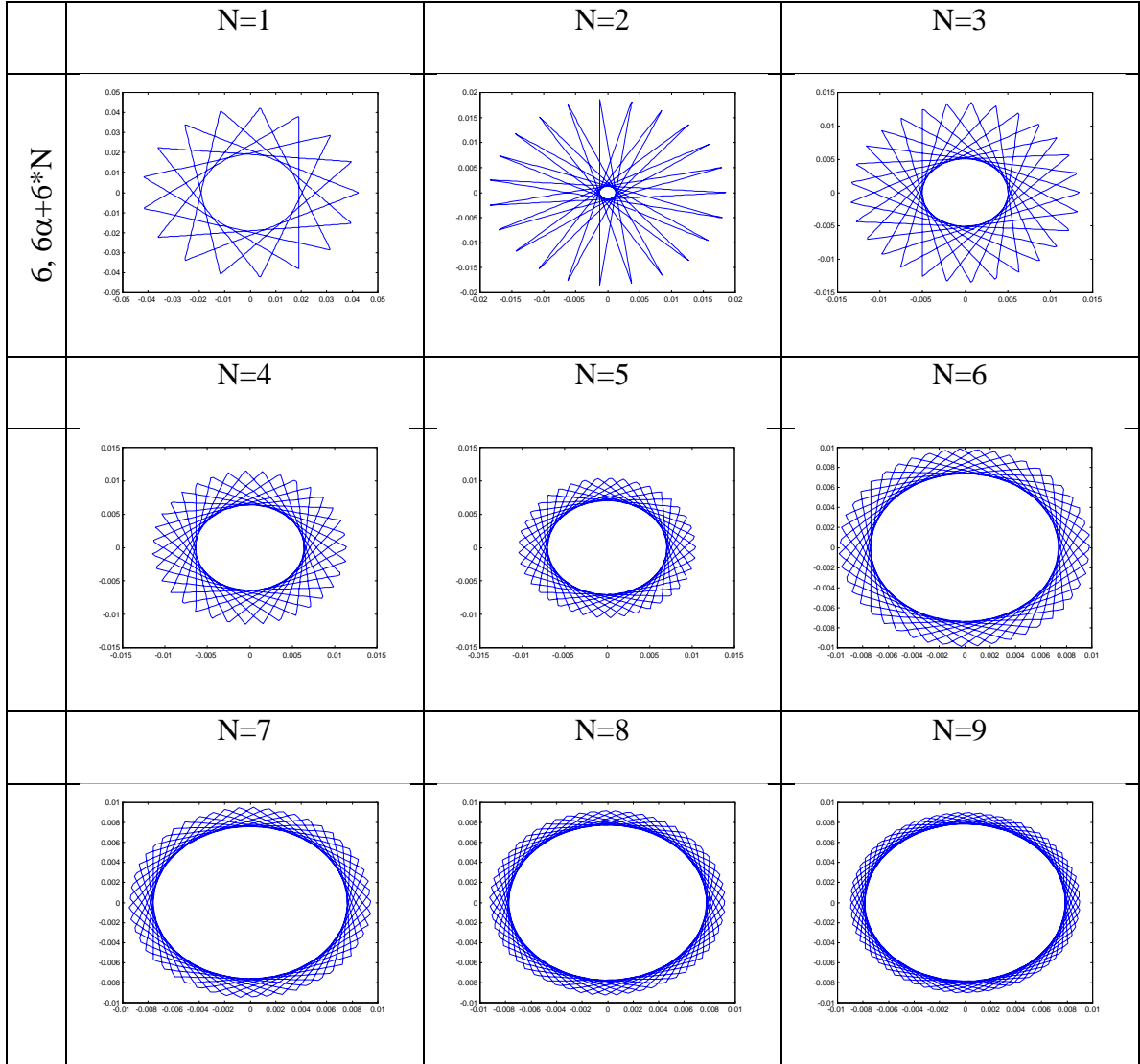
Information content

Usually information is measured by entropy [35]. When applying our algorithm to the noisy communication channel, we can explain how much information a system has by observation. Here is how it works: observe the polygons; check the size of the circle in the middle. The smaller the circle is, the larger information it contains.

As shown in Table 7, Table 8, and Table 11, the information content shows periodicity if we look at the y axis direction of the table. If we look at the x axis of the table, we can tell that the information is getting smaller with the increase of N. This

phenomenon also matches a quantum level example: atomic model [16]. In the future, we will study in our research group further grouping of the U matrix and information content.

Table 11 Information content



LIST OF REFERENCES

- [1]. M. M. Tanik and E. S. Chan, Fundamentals of Computing for Software Engineers. New York: Van Nostrand Reinhold, 1991.
- [2]. H. A. Simon, The Science of the Artificial. Third. Cambridge, MA: The MIT Press. 1999.
- [3]. R. W. Hamming, "The unreasonable effectiveness of mathematics," The American Mathematical Monthly, vol. 87, no. 2, February 1980.
- [4]. M. M. Tanik, "The Third Original Idea: Communication and Clocks," CAS interdisciplinary innovation forums, University of Alabama at Birmingham. January 16, 2013
- [5]. [ITRS] International Technology Roadmap for Semiconductors. www.itrs.org
- [6]. R.R.Schaller, "Moore's law: past, present and future," Spectrum, IEEE , vol.34, no.6, pp.52,59, Jun 1997
- [7]. A. A. Chien, V. Karamcheti. "Moore's Law: The First Ending and A New Beginning." Technical report, TR-2012-06, University of Chicago, 17 August, 2012.
- [8]. S. Borkar, A. A. Chien. "The future of microprocessors". Commun. ACM Vol. 54, No. 5, pp. 67-77. May 2011.
- [9]. C P. Williams, Explorations in Quantum Computing. Springer, 2011
- [10]. D. Bacon, "Quantum Computing Introduction and Basics of Quantum Theory," CSE 599d lecture notes, University of Washington
- [11]. B. Ozaydin, "An information theoretical modeling and analysis of systems," Ph.D dissertation, University of Alabama at Birmingham, 2012.
- [12]. B. Ozaydin, "On the relationships among Chebyshev polynomials, n-queens solutions, Fourier series, and models of atom," Master's thesis, University of Alabama at Birmingham, 2002.
- [13]. B. Ozaydin, Murat M. Tanik. "System modeling and analysis using communication channels." Journal of Integrated Design and Process Science, vol 15, no. 1, 2011, pp. 1-33
- [14]. B. Ozaydin, Murat M. Tanik. "Modeling and analysis of systems: An information theoretical approach," in Proceedings of the Thirteenth World Conference on Integrated Design and Process Technology, Dallas, TX, June 2010.

- [15]. B. Ozaydin, R. Seker, and M. M. Tanik, "Uniform polygons, Chebyshev polynomials, and N-queens solutions," Univ. of Alabama at Birmingham, Tech. Rep. 2001-04-ECE-002, 2001.
- [16]. B. Ozaydin, R. Seker, and M. M. Tanik, "Fourier series and Bohr's atomic model," Univ. of Alabama at Birmingham, Tech. Rep. 2001-10-ECE-010, 2001.
- [17]. R. Seker, "Component-based software modeling based on Shannon's information channels," Ph.D. dissertation, Univ. of Alabama at Birmingham, 2002.
- [18]. R. Seker, O. Aktunc, B. Ozaydin, M. M. Tanik, and L. Jololian, "Pervasive Shannon metrics and component-based software," *International Journal of Computer and Information Science*, vol. 5, no. 1, January 2004.
- [19]. R. Seker and M. M. Tanik, "An information-theoretical framework for modeling component-based systems," *IEEE Transactions on Systems, Man, and Cybernetics- Part C: Applications and Reviews*, vol. 34, no. 4, pp. 475–484, November 2004.
- [20]. R. Seker, M. M. Tanik, T. C. Jannett, and B. Ozaydin, "A new approach to the information channel: Fourier series," in *Proceedings of the Sixth World Conference on Integrated Design and Process Technology (IDPT)*, Pasadena, CA, 2002.
- [21]. R. Seker, M. M. Tanik, B. Ozaydin, and E. Ozturk, "Analysis of N-queens solutions using Chebychev polynomials," in *Proceedings of the Fifth World Conference on Integrated Design and Process Technology (IDPT)*, Dallas, TX, 2000.
- [22]. R. Seker, B. Ozaydin, and M. M. Tanik, "Polygons and Fourier series," Univ. of Alabama at Birmingham, Tech. Rep. 2001-07-ECE-004, 2001.
- [23]. R. Seker, B. Ozaydin, and M. M. Tanik, "A relationship between Fourier series and communication channel," Univ. of Alabama at Birmingham, Tech. Rep. 2001-08-ECE-008, 2001.
- [24]. R. Seker, B. Ozaydin, and M. M. Tanik, "On the relationship between Shannon's information channel and Fourier series," Univ. of Alabama at Birmingham, Tech. Rep. 2001-10-ECE-014, 2001.
- [25]. R. Seker and M. M. Tanik, "Division ring, information theory, and discrete noiseless communication channel," Univ. of Alabama at Birmingham, Tech. Rep. 2001- 12-ECE-018, 2001.
- [26]. R. Seker, M. M. Tanik, and B. Ozaydin, "Algebraic structures, complex numbers, and vectors," Univ. of Alabama at Birmingham, Tech. Rep. 2001-04-ECE-001, 2001.
- [27]. Fan Xiong, Murat M. Tanik. *Information Theory-based Quantum Computing Model for Atom Sized Computer*. IEEE Texas Workshop on Integrated System Exploration (TexasWISE) 2013 TexasWISE, Winedale House, Texas, 8 Mar. 2013

- [28]. Fan Xiong, Prebhu Shrestha, Murat M. Tanik, John U. Tanik, Susan Vasana. "Visual Analysis of cardiovascular system using evolvable hardware design." *Journal of Integrated Design and Process Science*, vol 15, no. 4, 2011, pp. 49-84
- [29]. Fan Xiong, Murat M. Tanik, "Transformation of Graduate Education: A Rapid Prototyping Test-Bed For Engineering Undergraduate Development", *International Journal of Computers, Information Technology and Engineering*, vol 5, no. 2, Dec. 2011, pp. 27-33
- [30]. Fan Xiong, John U. Tanik, Murat M. Tanik. An Evolutionary Circuit Model for Cardiovascular System: An FFGA Approach. *International Journal of Computers, Information Technology and Engineering*, vol. 5, no.2, Dec. 2011, pp. 41-53.
- [31]. Fan Xiong and M. M. Tanik, "A generalized noisy communication channel approach," Univ. of Alabama at Birmingham, Tech. Rep. 2013-07-ECE-001, 2013.
- [32]. Fan Xiong and M. M. Tanik, "A generalized noisy communication channel approach for quantum computing," Univ. of Alabama at Birmingham, Tech. Rep. 2013-07-ECE-002, 2013.
- [33]. Fan Xiong and M. M. Tanik, "An extended noisy communication channel framework," Univ. of Alabama at Birmingham, Tech. Rep. 2013-07-ECE-003, 2013.
- [34]. C. E. Shannon, "The zero error capacity of a noisy channel," *IRE Trans. on Information Theory*, vol. 2, pp. 8–19, September 1956.
- [35]. C. E. Shannon and W. Weaver, *The Mathematical Theory of Communication*. Urbana: Univ. of Illinois Press, 1963.
- [36]. C. Berge, *Graphs and Hypergraphs*. Amsterdam: North-Holland, 1973.
- [37]. A. Robert, "Fourier series of polygons," *American Mathematical Monthly*, vol. 101, no. 5, pp. 420–428, May 1994.
- [38]. H. M. Edwards. *Riemann's Zeta Function*. Academic Press. 1974
- [39]. A. M. Odlyzko. Primes, quantum chaos, and computers, in *Number Theory*, National Research Council, 1990, pp. 35-46.
- [40]. A. M. Odlyzko. On the distribution of spacings between zeros of the zeta function, *Math. Comp.*, 48 (1987), pp. 273-308
- [41]. H. I. Montgomery. The pair correlation of zeros of the zeta function. *Analytic number theory*, Proc. Sympos. Pure Math. XXIV, Providence, R.I.: American Mathematical Society, pp. 181–193, 1973.
- [42]. Z. Rudnick, P. Sarnak. Zeros of principal L-functions and random matrix theory. *Duke Mathematical Journal* 81 (2): 269–322.1996
- [43]. Tristan Needham, "Visual Complex Analysis". Oxford University Press, 1998
- [44]. T. J. Rivlin, *The Chebyshev Polynomials*. Wiley Interscience, 1990.
- [45]. M. M. Chawla, "Short notes: On Chebyshev polynomials of the second kind," *SIAM Review*, vol. 9, no. 4, pp. 729–733, 1967.

- [46]. D. Y. Savio and E. R. Suryanarayan, "Chebychev polynomials and regular polygons," *American Mathematical Monthly*, vol. 100, no. 7, pp. 657–661, August / September 1993.
- [47]. B. Omer, "A procedural Formalism for Quantum Computing", Technical university of Vienna, e-mail: oemer@tph.tuwien.ac.at, Homepage: <http://tph.tuwien.ac.at/~oemer>.
- [48]. "U-M develops scalable and mass- producible quantum computer chip." PHYSOrg.com. 14 Dec 2005. <http://phys.org/news9063.html>
- [49]. Dicarlo, L; Chow, JM; Gambetta, JM; Bishop, LS; Johnson, BR; Schuster, DI; Majer, J; Blais, A et al. (2009-06-28). "Demonstration of two-qubit algorithms with a superconducting quantum processor". *Nature* 460 (7252): 240-4. Bibcode 2009Natur.460..240D. doi:10.1038/nature08121. PMID 19561592.
- [50]. "Scientists Create First Electronic Quantum Processor". 2009-07-02. http://www.eurekalert.org/pub_releases/2009-06/yu-scf062509.php
- [51]. New Scientist. "Code-breaking quantum algorithm runs on a silicon chip". 2009-10-14. <http://www.newscientist.com/article/dn17736-codebreaking-quantum-algorithm-run-on-a-silicon-chip.html>
- [52]. Gilles Brassard , "New Trends in Quantum Computation". arXiv:quant-ph/9602014
- [53]. Quantum Information Processing. Springer.com. 2011-05-19.
- [54]. Trute, Peter. "Quantum teleporter breakthrough". The University Of New South Wales. 17 April 2011.
- [55]. Richard Lai, "Engadget, First light wave quantum teleportation achieved, opens door to ultra fast data transmission". <http://www.engadget.com/2011/04/18/first-light-wave-quantum-teleportation-achieved-opens-door-to-u/>, 2011.4
- [56]. "Learning to program the D-Wave One". 11 May 2011. <http://dwave.wordpress.com/2011/05/11/learning-to-program-the-d-wave-one/>
- [57]. Enrique Martin Lopez, Anthony Laing, Thomas Lawson, Roberto Alvarez, Xiao-Qi Zhou, Jeremy L. O'Brien (2011). "Implementation of an iterative quantum order finding algorithm". arXiv:1111.4147 [quant-ph].
- [58]. "Quantum Processor Hooks Up with Quantum Memory", <http://www.technologyreview.com/news/425304/quantum-processor-hooks-up-with-quantum-memory/>
- [59]. "IBM Says It's 'On the Cusp' of Building a Quantum Computer", 2012.2 <http://www.pcmag.com/article2/0,2817,2400930,00.asp>
- [60]. "Scientists Build a Quantum Computer Inside a Diamond". 2012.4. <http://www.tomshardware.com/news/quantum-computer-diamond-qubit-computing,15230.html>
- [61]. "Australian engineers write quantum computer 'qubit' in global breakthrough". *The Australian*. Retrieved 3 October 2012.

- [62]. "Breakthrough in bid to create first quantum computer". The University of New South Wales. Retrieved 3 October 2012.
- [63]. P. Dirac. *The Principles of Quantum Mechanics*. Clarendon Press Oxford, United Kingdom, 1987
- [64]. C P. Williams, *Explorations in Quantum Computing*. Springer, 2011
- [65]. B. Good, "A brief introduction to quantum computing." May. 2007.
- [66]. NS Yanofsky, MA Manna. *quantum computing for computer scientists*. Cambridge university press, 2008.
- [67]. G.P. Berman, *Introduction to quantum computers*. NJ: World Scientific. 1998.
- [68]. P. W. Shor, *Introduction to quantum algorithms*, <http://arXiv.org/abs/quant-ph/quant-ph/0005003>, 2000.
- [69]. "Basic Concepts in Quantum Computation." Quantiki. Online. Available: http://www.quantiki.org/wiki/index.php/Basic_concepts_in_quantum_computation. 2 May 2007.
- [70]. Barenco, A. et al. , 'Elementary gates for quantum computation', *Phys. Rev., A* 52: 3457-3467.1995.
- [71]. Peter W. Shor, "Polynomial-Time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer",1994, arXiv:quant-ph/9508027
- [72]. Chopra, Vinay, "A Quantum based Method for FPGA Routing", Master's thesis, Institute of Engineering and Technology (Deemed University), Patiala, 2007
- [73]. Bravyi, Sergey and Caha, Libor and Movassagh, Ramis and Nagaj, Daniel and Shor, Peter W. Criticality without Frustration for Quantum Spin-1 Chains. *Phys. Rev. Lett.* Vol.109, no.20, pp.202-207, Nov. 2012.
- [74]. Larry Hardesty, "Proving quantum computers feasible". MIT news, Nov. 2012
- [75]. H. Jafarkhani, *Space-Time Coding: Theory and Practice*, Cambridge University Press, 2005.
- [76]. Feynman, Richard. *QED: The strange theory of light and matter*. Princeton University Press. 1985

APPENDIX

Appendix A: Math Background

In this chapter, various math tools are introduced to provide a foundation for the quantum modeling and framework. Each of the math topics covered can be written in several books. Only brief and necessary math for understanding this dissertation is introduced here.

Basically five categories of math are introduced: complex numbers and function, permutation matrix, Fourier matrix, Pauli matrix and Riemann zeta function. Only exponential form and roots of unity of complex numbers are introduced. The definition of permutation matrix is given, and a special forward shift permutation matrix and the relation of roots of unity are investigated. Fourier matrix and Riemann zeta function are quickly reviewed.

Complex numbers

Polar forms of complex numbers powers and roots

The complex numbers in the complex plane with xy-coordinates could be further extended to polar form with polar coordinates r, θ , which is defined by

$$x = r \cos \theta, \quad y = r \sin \theta .$$

Thus polar form of $z = x + iy$ is

$$z = r(\cos \theta + i \sin \theta)$$

r is called the absolute value or modulus of z and is denoted by $|z|$.

$$|z| = r = \sqrt{x^2 + y^2} = \sqrt{z\bar{z}}$$

θ is called the argument of z and is denoted by $\arg z$.

$$\theta = \arg z = \tan^{-1} \frac{y}{x}$$

The polar form of a complex number in the complex plane is shown in Figure 69.

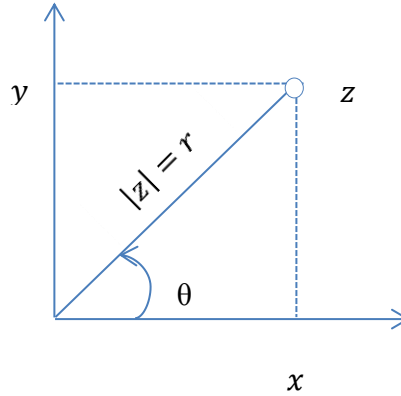


Figure 69 Polar form of a complex number in the complex plane.

Multiplication and Division in Polar form

Given $z_1 = r_1(\cos \theta_1 + i \sin \theta_1)$ and $z_2 = r_2(\cos \theta_2 + i \sin \theta_2)$, multiply z_1 and z_2 yields

$$z_1 z_2 = r_1 r_2 [\cos(\theta_1 + \theta_2) + i \sin(\theta_1 + \theta_2)]$$

Taking absolute values on both sides, we obtain

$$|z_1 z_2| = |z_1| |z_2|$$

and

$$\arg(z_1 z_2) = \arg z_1 + \arg z_2$$

Let $z = \frac{z_1}{z_2}$. Hence $|zz_2| = |z||z_2| = |z_1|$, $\arg(zz_2) = \arg z + \arg z_2 = \arg z_1$, which yields

$$\left| \frac{z_1}{z_2} \right| = \frac{|z_1|}{|z_2|}$$

and

$$\arg\left(\frac{z_1}{z_2}\right) = \arg z_1 - \arg z_2$$

$$\frac{z_1}{z_2} = \frac{r_1}{r_2} [\cos(\theta_1 - \theta_2) + i \sin(\theta_1 - \theta_2)]$$

Integer powers: De Moivre's formula

Let $z_1 = z_2 = z$, for $n = 0, 1, 2, \dots$,

$$z^n = r^n (\cos n\theta + i \sin n\theta)$$

This could be proven by induction. Similarly, with $z_1 = 1$ and $z_2 = z^n$, for $n = -1, -2, \dots$, for $|z| = r = 1$, we obtain De Moivre's formula

$$(\cos \theta + i \sin \theta)^n = \cos n\theta + i \sin n\theta$$

Roots

For a given $z \neq 0$, there exists precisely n distinct values of w , which satisfy $z = w^n$ ($n = 1, 2, \dots$). Each of these values is called an n th root of z , and $w = \sqrt[n]{z}$. w has n values. The values can be obtained in polar forms as

$$z = r(\cos \theta + i \sin \theta) \quad \text{and} \quad w = R(\cos \varphi + i \sin \varphi).$$

The equation $z = w^n$ becomes

$$w^n = R^n(\cos n\varphi + i \sin n\varphi) = z = r(\cos \theta + i \sin \theta).$$

Solving the equation, we obtain

$$R = \sqrt[n]{r}, \text{ and } \varphi = \frac{\theta}{n} + \frac{2k\pi}{n}, (k = 0, 1, 2, \dots, n - 1).$$

Finally the n values of $\sqrt[n]{z}$ is

$$\sqrt[n]{z} = \sqrt[n]{r} \left(\cos \frac{2k\pi + \theta}{n} + \sin \frac{2k\pi + \theta}{n} \right), (k = 0, 1, 2, \dots, n - 1).$$

These n values are distributed in a circle of radius $\sqrt[n]{r}$ with a

center at the origin, which form a regular polygon of n sides. For a special case $z=1$, we have $|z| = r = 1$, and $Arg z = 0$.

$$\sqrt[n]{1} = \left(\cos \frac{2k\pi}{n} + \sin \frac{2k\pi}{n} \right), (k = 0, 1, 2, \dots, n - 1)$$

These n values are called the roots of unity. They are equally distributed in the circle with a radius of 1 and a center of 0, and they are called the unit circle. Figure 70 shows the roots of unity of $n = 1, 2, \dots, 9$

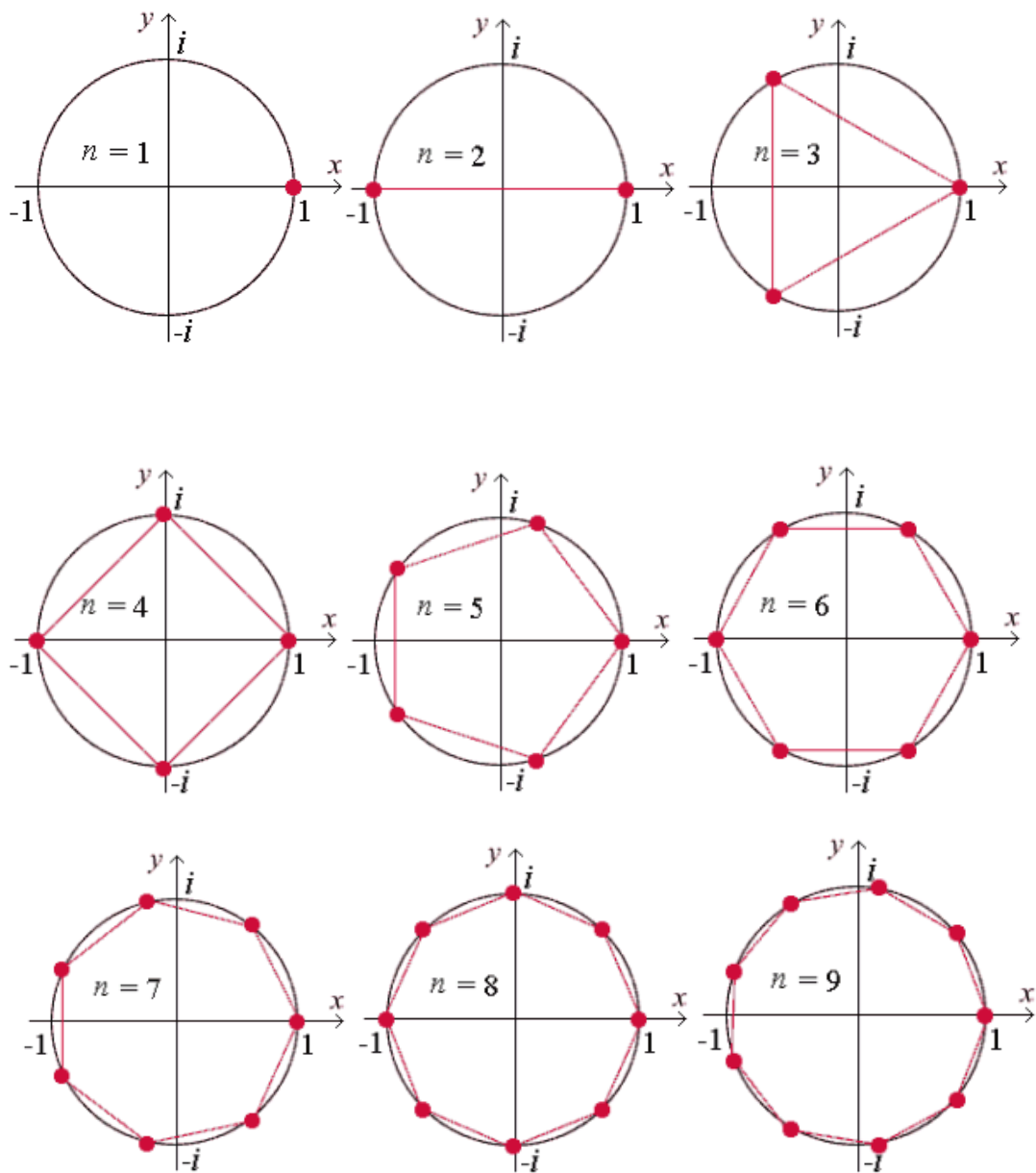


Figure 70 Roots of unity for $n = 1, 2, \dots, 9$

Exponential form

The exponential function e^z is defined in terms of e^x , $\cos y$, and $\sin y$,

$$e^z = e^x(\cos y + i \sin y)$$

For $z = iy$, we have

$$e^{iy} = \cos y + i \sin y,$$

which is also called the Euler formula.

Thus, the polar form of a complex number, $z = r(\cos y + i \sin y)$, can also be written as

$$z = r e^{i\theta}$$

Amplitwist

The term amplitwist comes from Tristan Needham. The idea is that a complex-valued function is differentiable at a point where it has an “amplitwist” there. The amplitwist is a combination of amplification and twisting, or rotation. The “amplification” is the expansion factor, and the “twist” is the angle of rotation. Under this definition, the derivative of f or the amplitwist of f , $f'(z)$, may be thought of locally as a linear transformation:

$$\begin{aligned} f'(z) &= \text{the amplitwist of } f \text{ at } z \\ &= (\text{amplification})e^{i(\text{twist})} \\ &= |f'(z)|e^{i\arg[f'(z)]} \end{aligned}$$

Complex function

For a complex function, both the independent and dependent variables are both complex numbers. A complex function’s range is in the complex numbers.

A complex function can be written as

$$z = x + iy$$

$$w = f(z) = u(x, y) + iv(x, y)$$

where $x, y \in R$, and $u(x, y), v(x, y)$ are real value functions. It is difficult to graph a complex function, since there are 2 dimensions for the range and 2 dimensions for the domain, creating a total of 4 dimensions.

Complex logarithm

A complex logarithm is the inverse of a complex exponential function. The logarithm of a complex number z is defined as every complex number ω that satisfies the equation

$$e^{\omega} = z.$$

Recalling the exponential form of complex numbers, we have

$$e^{\omega} = r e^{i\theta},$$

where $r = |z|$ and $\theta = \text{arg}z$; solving the equation, we have

$$\omega = \log z = \ln|z| + i \text{arg}z .$$

Here $\ln|z|$ is the natural logarithm of the real number $|z|$. Note that the logarithm of complex function is multivalued. If $0 \leq \theta < 2\pi$, we can write

$$\log z = \ln|z| + i\theta + n \cdot 2\pi i \quad (n = 0, \pm 1, \pm 2, \dots).$$

The values with $n=0$ are called the principal values of the logarithm.

Permutation matrices

For a set $N = \{1, 2, \dots, n\}$, there exists $n!$ distinct permutations of N , including the identity permutation. Given a permutation σ of set N ,

$$\sigma: \{1, 2, \dots, n\} \rightarrow \{1, 2, \dots, n\},$$

which can be written in two-line form

$$\left\{ \begin{array}{cccc} 1 & 2 & \dots & n \\ \sigma(1) & \sigma(2) & \dots & \sigma(n) \end{array} \right\}.$$

The permutation matrix is the $n \times n$ matrix P_σ whose entries are all 0 except that, in row i , the entry $\sigma(i)$ is 1. Therefore, every row and column contain precisely a single 1 with 0's everywhere else.

Example 5.

$$\sigma: \begin{pmatrix} 1 & 2 & 3 & 4 \\ 3 & 1 & 4 & 2 \end{pmatrix}, P_\sigma = \begin{pmatrix} 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \end{pmatrix}$$

A permutation matrix is non-singular, and the determinant is always ± 1 . Given a permutation matrix A and its transpose A^T , the following equation holds,

$$AA^T = I, \text{ where } I \text{ is the identity matrix.}$$

Forward shift permutation matrix

Permutation $\sigma(1) = 2, \sigma(2) = 3, \dots, \sigma(n-1) = n, \sigma(n) = 1$ generates a forward shift permutation matrix

$$\pi = \begin{pmatrix} 0 & 1 & 0 & 0 & \dots & 0 \\ 0 & 0 & 1 & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \dots & \vdots \\ 1 & 0 & 0 & 0 & \dots & 0 \end{pmatrix}.$$

Thus, we have

$$\pi^2 = \begin{pmatrix} 0 & 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & 0 & 1 & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \dots & \vdots \\ 0 & 1 & 0 & 0 & \dots & 0 \end{pmatrix}$$

Similarly for π^k and σ^k , the matrix π^n corresponds to $\sigma^n = I$ so that $\pi^n = I$.

Permutation matrix and roots of unity

Let $P = P_\sigma$ indicate an $n \times n$ permutation matrix, σ could be factored to a product of disjoint cycles. Suppose that the length of the cycles in the product are P_1, P_2, \dots, P_m , and the sum of P_1, P_2, \dots, P_m is n . By properly arranging the rows and columns, the cycles in P_σ can only contain contiguous indices with successive integers. Let π_{P_k} designate as the π matrix of order P_k , the characteristic polynomial of π_{P_k} is

$$(-1)^{P_k}(\lambda^{P_k} - 1)$$

The eigenvalues of the permutation matrix P are the roots of unity of all the roots of the m equations:

$$\lambda^{P_k} = 1, k = 1, 2, \dots, m$$

Example

$$P_\sigma = \begin{pmatrix} 0 & 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \end{pmatrix}$$

σ is the permutation of 1,2,3,4,5,6, and $\sigma(1) = 5, \sigma(2) = 1, \sigma(3) = 6, \sigma(4) = 4, \sigma(5) = 2, \sigma(6) = 3$.

σ can be cycled as $\sigma = (152)(4)(36)$. Thus, $m=3, p_1=3, p_2=1, p_3=2$. The eigenvalues of P_σ are the roots of $(\lambda^3 - 1)(\lambda - 1)(\lambda^2 - 1)$.

If the factorization of a permutation σ consists of one cycle of full length n , the permutation is called primitive. The eigenvalues of a primitive permutation matrix are the n th root of unity.

Fourier matrices

The Fourier matrix is defined as

$$F_n = \frac{1}{\sqrt{n}} \begin{pmatrix} 1 & 1 & 1 & \dots & 1 \\ 1 & w & w^2 & \dots & w^{n-1} \\ 1 & w^2 & w^4 & \dots & w^{2(n-1)} \\ 1 & \vdots & \vdots & \ddots & \vdots \\ 1 & w^{n-1} & w^{2(n-1)} & \dots & w^{(n-1)(n-1)} \end{pmatrix};$$

w could be roots of unity, where $w = \exp\left(\frac{2\pi i}{n}\right) = \cos\frac{2\pi}{n} + \sin\frac{2\pi}{n}$. Since the sequence $w^k, k = 0, 1, \dots$, is periodic, F_n can be rewritten as

$$F_n = \frac{1}{\sqrt{n}} \begin{pmatrix} 1 & 1 & 1 & \dots & 1 \\ 1 & w & w^2 & \dots & w^{n-1} \\ 1 & w^2 & w^4 & \dots & w^{n-2} \\ 1 & \vdots & \vdots & \ddots & \vdots \\ 1 & w^{n-1} & w^{n-2} & \dots & w \end{pmatrix}$$

We can tell that F_n is a complex symmetric matrix, and its transpose is the same as it is.

Some important characteristics of the Fourier matrix are:

- 1) The Fourier matrix is a unitary matrix, which means that the inverse is the adjoint.
- 2) The Fourier matrix plays a role in the theory of circulants, which also behaves like a shift register. Every row vector is shifted one element to the right with respect to the previous row vector.
- 3) The columns of the Fourier matrix are eigenvectors of the circulant matrix of the same size.
- 4) The multiplication of the Fourier matrix by the first row of the circulant matrix gives the eigenvalues corresponding to the eigenvectors of the circulant matrix.

The Fourier Matrix is a concept related to the roots of unity. The Fourier Matrix is a matrix organization of the Discrete Fourier Transform. Its structure for $n \times n$ size

consists of n roots of unity arranged in different orders in every row or column of the matrix. The example below shows a 7×7 Fourier matrix.

$$\begin{bmatrix} e^{-j*0*0*2\pi/7} & e^{-j*0*1*2\pi/7} & e^{-j*0*2*2\pi/7} & e^{-j*0*3*2\pi/7} & e^{-j*0*4*2\pi/7} & e^{-j*0*5*2\pi/7} & e^{-j*0*6*2\pi/7} \\ e^{-j*1*0*2\pi/7} & e^{-j*1*1*2\pi/7} & e^{-j*1*2*2\pi/7} & e^{-j*1*3*2\pi/7} & e^{-j*1*4*2\pi/7} & e^{-j*1*5*2\pi/7} & e^{-j*1*6*2\pi/7} \\ e^{-j*2*0*2\pi/7} & e^{-j*2*1*2\pi/7} & e^{-j*2*2*2\pi/7} & e^{-j*2*3*2\pi/7} & e^{-j*2*4*2\pi/7} & e^{-j*2*5*2\pi/7} & e^{-j*2*6*2\pi/7} \\ e^{-j*3*0*2\pi/7} & e^{-j*3*1*2\pi/7} & e^{-j*3*2*2\pi/7} & e^{-j*3*3*2\pi/7} & e^{-j*3*4*2\pi/7} & e^{-j*3*5*2\pi/7} & e^{-j*3*6*2\pi/7} \\ e^{-j*4*0*2\pi/7} & e^{-j*4*1*2\pi/7} & e^{-j*4*2*2\pi/7} & e^{-j*4*3*2\pi/7} & e^{-j*4*4*2\pi/7} & e^{-j*4*5*2\pi/7} & e^{-j*4*6*2\pi/7} \\ e^{-j*5*0*2\pi/7} & e^{-j*5*1*2\pi/7} & e^{-j*5*2*2\pi/7} & e^{-j*5*3*2\pi/7} & e^{-j*5*4*2\pi/7} & e^{-j*5*5*2\pi/7} & e^{-j*5*6*2\pi/7} \\ e^{-j*6*0*2\pi/7} & e^{-j*6*1*2\pi/7} & e^{-j*6*2*2\pi/7} & e^{-j*6*3*2\pi/7} & e^{-j*6*4*2\pi/7} & e^{-j*6*5*2\pi/7} & e^{-j*6*6*2\pi/7} \end{bmatrix}$$

The Fourier Matrix generates regular polygons, as shown in Figure 71. For the proper choice of points from the Fourier Matrix, any polygon can be drawn.

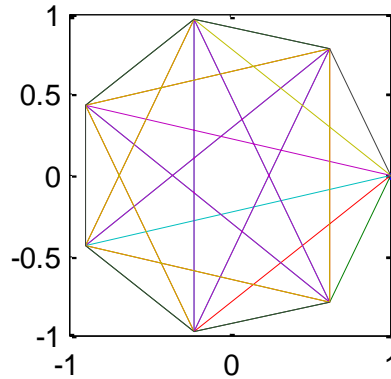


Figure 71. Regular polygon generated by 7x7 Fourier Matrix.

Discrete Fourier transform

In a complex n -tuples Z and \hat{Z} , $Z = (z_1, z_2, \dots, z_n)^T$, and $\hat{Z} = (\hat{z}_1, \hat{z}_2, \dots, \hat{z}_n)^T$.

The linear transformation $\hat{Z} = FZ$ is known as the discrete Fourier transform, where F is the Fourier matrix. The inverse is given by $F^{-1}\hat{Z} = Z$. The transform is called harmonic analysis or periodogram analysis, and the inverse transform is called harmonic synthesis.

Example 6. $P(z) = a_0 + a_1z + \dots + a_nz^{n-1}$ is a polynomial. The polynomial is determined by n distinct points $z_k, k = 1, 2, \dots, n$ in the complex plane. Choose z_k as the n roots of unity $1, w, w^2, \dots, w^{n-1}$. Then we have

$$\sqrt{n}F \begin{pmatrix} a_0 \\ a_1 \\ \vdots \\ a_{n-1} \end{pmatrix} = \begin{pmatrix} P(1) \\ P(w) \\ \vdots \\ P(w^{n-1}) \end{pmatrix}$$

The passage from coefficient values to functional values “synthesized” the function value.

Based on this equation, we also have

$$\begin{pmatrix} a_0 \\ a_1 \\ \vdots \\ a_{n-1} \end{pmatrix} = \frac{1}{\sqrt{n}}F \begin{pmatrix} P(1) \\ P(w) \\ \vdots \\ P(w^{n-1}) \end{pmatrix}$$

This passage from functional values to coefficients is an analysis of the function.

Pauli matrices

Matrices

$$\Sigma_1 = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix},$$

$$\Sigma_2 = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix},$$

$$\Sigma_3 = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$$

are called Pauli matrices. The Pauli matrices and the unit matrix $I = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$ can be used

to form 2×2 matrices. For any 2×2 matrix

$$A = \begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix}$$

There exists complex numbers z_0, z_1, z_2, z_3 such that

$$A = z_0 I + z_1 \Sigma_1 + z_2 \Sigma_2 + z_3 \Sigma_3$$

with

$$z_0 = \frac{1}{2}(a_{11} + a_{22}),$$

$$z_1 = \frac{1}{2}(a_{21} + a_{12}),$$

$$z_2 = -i \frac{1}{2}(a_{21} - a_{12}),$$

$$z_3 = \frac{1}{2}(a_{11} - a_{22})$$

For a given matrix

$$M = z_0 I + z_1 \Sigma_1 + z_2 \Sigma_2 + z_3 \Sigma_3,$$

if $z_0^2 - z_1^2 - z_2^2 - z_3^2 \neq 0$, then M has an inverse, which is

$$M^{-1} = \frac{1}{z_0^2 - z_1^2 - z_2^2 - z_3^2} (z_0 I - z_1 \Sigma_1 - z_2 \Sigma_2 - z_3 \Sigma_3).$$

Riemann zeta function

The Riemann zeta function is an important special function of mathematics and physics, which is denoted by Riemann in his 1859 paper. The Riemann zeta function is defined over the complex plane for one complex variable. The Riemann zeta function $\zeta(s)$ is a function of complex variables that analytically continues the sum of the infinite series $\sum_{n=1}^{\infty} \frac{1}{n^s}$, which converges when the real part of s is greater than 1. It is implemented in Matlab as function zeta and also implemented in Mathematica as Zeta.

The Riemann zeta function $\zeta(s)$ is a function of a complex number $s = \sigma + it$. $\zeta(s)$ is defined as

$$\zeta(s) = \sum_{n=1}^{\infty} \frac{1}{n^s} = \frac{1}{1^s} + \frac{1}{2^s} + \frac{1}{3^s} + \dots \quad \sigma = R(s) > 1$$

The infinite series converge for all complex numbers s with real parts greater than 1.

Euler product formula

Euler discovered the connection between zeta function and prime numbers. Euler proved the identity

$$\sum_{n=1}^{\infty} \frac{1}{n^s} = \prod_{p \text{ prime}} \frac{1}{1 - p^{-s}}$$

The left-hand side is $\zeta(s)$, and the right-hand side is extended to all the prime numbers.

The expression is called Euler products:

$$\prod_{\text{prime}} \frac{1}{1 - p^{-s}} = \frac{1}{1 - 2^{-s}} \cdot \frac{1}{1 - 3^{-s}} \cdot \frac{1}{1 - 5^{-s}} \cdot \frac{1}{1 - 7^{-s}} \cdot \frac{1}{1 - 11^{-s}} \dots \frac{1}{1 - p^{-s}} \dots$$

The converge condition ($\sigma = R(s) > 1$) holds on both sides.

Chebyshev polynomial

By applying the identity of cosines

$$\cos(\alpha + \beta) = \cos \alpha \cos \beta + \sin \alpha \sin \beta$$

we have

$$\cos((n + 1)\theta) = \cos n\theta \cos \theta + \sin n\theta \sin \theta$$

We could rewrite the equation for any non-negative integer n and C_i , such that

$$\cos(n\theta) = \sum_{i=0}^n C_i \cos^i \theta$$

This equation indicates that $\cos(n\theta)$ is a polynomial in $\cos \theta$. The n th Chebyshev polynomial for fixed n is defined as

$$\cos(n\theta) = T_n \cos \theta$$

Let $x = \arccos \theta$, this results in

$$T_n(x) = \cos(n \arccos x)$$

We could easily obtain the polynomial up to degree n . Let us start by $n=0$,

$$T_0(x) = \cos(0) = 1$$

$$T_1(x) = \cos(\arccos x) = x$$

$$T_2(x) = \cos(2 \arccos x) = 2 (\cos(\arccos x))^2 - 1 = 2x^2 - 1$$

Then, when we apply the recursive property of the Chebyshev polynomial [44], we have

$$T_{n+1}(x) = 2xT_n(x) - T_{n-1}(x)$$

Thus,

$$T_3(x) = 2xT_2(x) - T_1(x) = 2x(2x^2 - 1) - x = 4x^3 - 3x$$

As we continue on, the Chebyshev polynomial up to degree 10 is shown in Table 12:

Table 12 Chebyshev polynomial of degree 10

$T_0(x)$	1
$T_1(x)$	x
$T_2(x)$	$2x^2 - 1$
$T_3(x)$	$4x^3 - 3x$
$T_4(x)$	$8x^4 - 8x^2 + 1$
$T_5(x)$	$16x^5 - 20x^3 + 5x$
$T_6(x)$	$32x^6 - 48x^4 + 5x$
$T_7(x)$	$64x^7 - 112x^5 + 56x^3 - 7x$
$T_8(x)$	$128x^8 - 256x^6 + 160x^4 - 32x^2 + 1$
$T_9(x)$	$256x^9 - 567x^7 + 432x^5 - 120x^3 + 9x$
$T_{10}(x)$	$512x^{10} - 1280x^8 + 1120x^6 - 400x^4 + 50x^2 - 1$

We could also derive the Chebyshev polynomial of the second kind, which we will use in our analysis. If we differentiate $T_n(x) = \cos n\theta$ in terms of x , we have

$$T_n'(x) = \frac{n \sin n\theta}{\sin \theta}$$

The general form of the Chebyshev polynomial of the second kind with degree p is [45]

$$u_p(x) = \frac{\sin(p+1)\theta}{\sin \theta}$$

The Chebyshev polynomial of the second kind up to degree 10 is shown in Table 13.

Table 13 Chebyshev polynomial of the second kind up to degree 10

$T_0(x)$	1
$T_1(x)$	$2x$
$T_2(x)$	$4x^2 - 1$
$T_3(x)$	$8x^3 - 4x$
$T_4(x)$	$16x^4 - 12x^2 + 1$
$T_5(x)$	$32x^5 - 32x^3 + 6x$
$T_6(x)$	$64x^6 - 80x^4 + 24x^2 - 1$
$T_7(x)$	$128x^7 - 192x^5 + 80x^3 - 8x$
$T_8(x)$	$256x^8 - 448x^6 + 240x^4 - 40x^2 + 1$
$T_9(x)$	$512x^9 - 1024x^7 + 672x^5 - 160x^3 + 10x$
$T_{10}(x)$	$1024x^{10} - 2304x^8 + 1792x^6 - 5600x^4 + 60x^2 - 1$

Multivalued logic

For a classical, two-valued logic as shown in Figure 72, R and \bar{R} represent truth and falsehood, respectively. The arrow indicates the results of the inversion. The truth value is flipped upside down. Truth turns to falsehood, and falsehood turns to truth. If we flip truth twice, it will get back to the original setting, which is also called double negation.

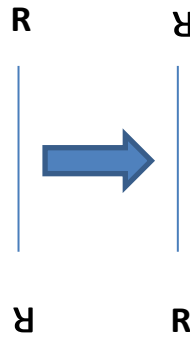


Figure 72 Classical two-valued logic

If we consider the possibility of adding a third truth value, 'I' for intermediate or indeterminate state, the representation can be illustrated, as shown in Figure 73. Since there are three values, the structure rotates 120 degrees, and it will take three negations to get back to the original settings.

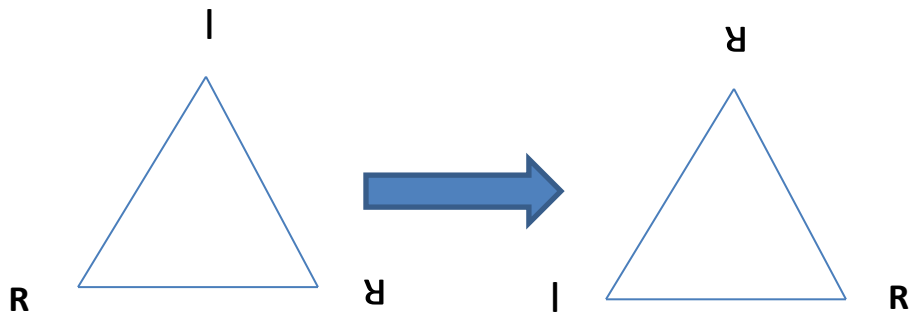


Figure 73 Three-valued logic

We can extend the examples to a general case. For any number n of distinct truth values t_1, t_2, \dots, t_n , we can imagine a corresponding regular polygon representation as the result of a rotation of $360/n$ degrees, as shown in Figure 74. Here the negation of each value is the next clockwise value, and, since the polygon is closed, the negation of the last value t_n will return to the initial value t_1 .

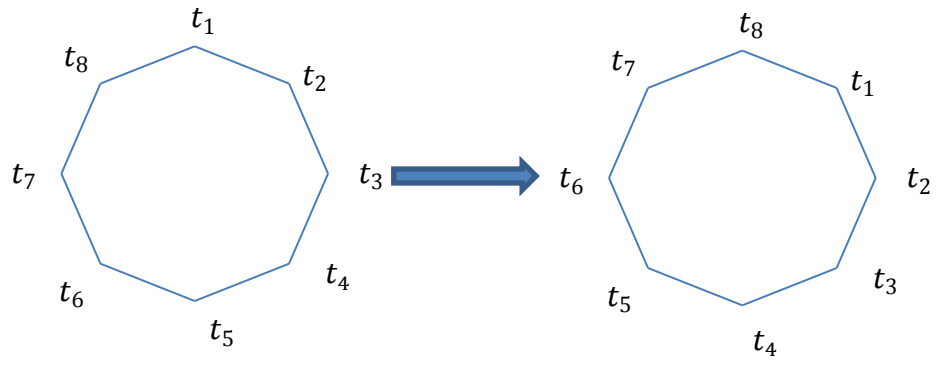


Figure 74 N-valued logic

Appendix B: The Noisy Communication Channel Framework

Communication channel representations

There are different representations of the communication channel, and this section briefly reviews the communication channel representation.

Communication error probability representation

Shannon represents a noisy communication channel with its communication error probabilities, as shown in Figure 75. In this representation, the arrow indicates the transmission from the sender to the receiver with certain probability. If there are no arrows between two symbols, then there is no possibility that the source symbol can be received as a sink symbol.

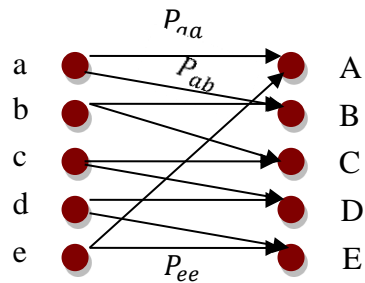


Figure 75 Communication error representation of noisy communication channel

Note that the sum of the probabilities from the same source symbol is 1. For example,

$P_{aa} + P_{ab} = 1$. The representation could be easily written in matrix format as

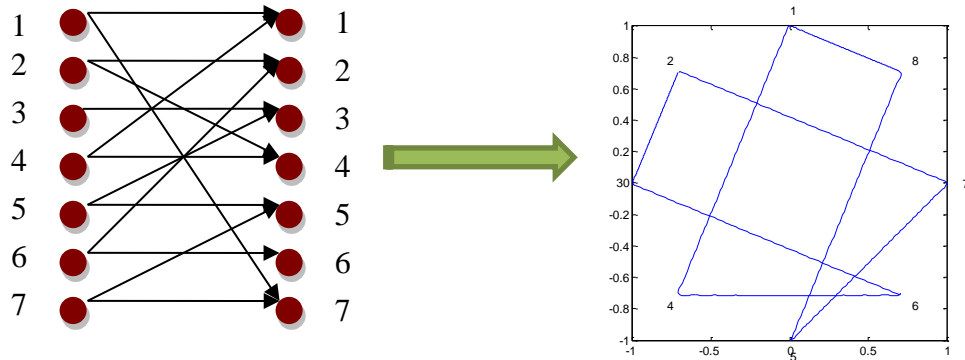
$$\begin{bmatrix} & a & b & c & d & e \\ A & p_{aa} & p_{ab} & 0 & 0 & 0 \\ B & 0 & p_{bb} & p_{bc} & 0 & 0 \\ C & 0 & 0 & p_{cc} & p_{cd} & 0 \\ D & 0 & 0 & 0 & p_{dd} & p_{de} \\ E & p_{ea} & 0 & 0 & 0 & p_{ee} \end{bmatrix}$$

Error content graph representation

Error Content Graphs are another common approach of representing the communication channel. Graphs of this type provide a visual representation of the transmitter-receiver connections specified by the communication error probability representation. This type of representation makes it easy to immediately estimate the capacity of the channel. The area in the middle of the graph corresponds to the capacity of the channel. The capacity of the channel is bigger if no lines cut close to the center of the circle, i.e. when there are no connections between symbols that are far from each other.

By observing Figure 5, we can see that the error pairs of the symbols are $a \rightarrow B$, $b \rightarrow C$, $c \rightarrow D$, $d \rightarrow E$, and $e \rightarrow A$, which produces the error content graph shown in Figure 6.

We can see that communication channel can be represented by polygons, regular polygon in this case. As described earlier cases of irregular polygons also exist.



gives an example of a communication channel and its corresponding irregular polygon.

Stability number representation

From Figure 6, we can tell that the adjacent vertices of the graph are only those signals can be confused at the receiver. For example, a can be confused as b as shown in Figure 6, so a and b are adjacent vertices in the graph.

A stable set of G is a subset of vertices with no edge between any two of them. For example, a and b are not a stable set of graph in Figure 76, since there is an edge between them; a and c, or a and d, or b and d etc are stable sets. Given a set S , the neighborhood of S is $T(S)$, if $T(s) \cap S = \Phi$, S is stable. Take a and c for example, the neighbors of a are e and b, the neighbors of c are b and d; the neighborhood set $T(S)$ is $\{b, d, e\}$, obviously $T(s) \cap S = \{b, d, e\} \cap \{a, c\} = \Phi$, so a and c is a stable set.

Stability number is not a commonly used term in English graph theory. Stability number, or independent number is the number of vertices in the largest totally disconnected sub-graph of a graph, or is the maximum cardinality (the count) of a stable set. The Stability number for the communication channel can be also seen as the count of the largest set of vertices in its error content graph with no two vertices are connected. There could be many stable sets of a graph, some of which with the same stability

number. For the graph in Figure 6, the stability number $\alpha(G)=2$ which is equal to clique number $\theta(G)$, this is a perfect graph. For the graph in Figure 7, the stability number is 3.

The example state sets of regular and irregular polygons are shown in Figure 76.

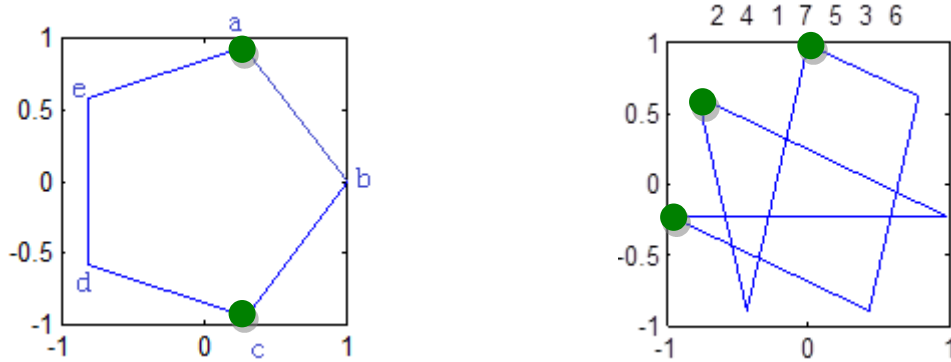


Figure 76 Example state sets of the error content graph representations

Permutation matrix representation

Similar to the communication error probability matrix form, a communication channel can also be represented by permutation matrix. The permutation matrix representation of the communication channels in Figure 6 is

$$[a \ b \ c \ d \ e]$$

The permutation matrix representation of the communication channel in Figure 76 is

$$[2 \ 4 \ 1 \ 7 \ 5 \ 3 \ 6]$$

which could be also written in the form

$$\begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}$$

These symbols are used to specify a particular location in the grid of the matrix. In case of communication channel, 1s can specify the location of particular code words in the code space. These representations are very useful in observing the separation between the code words. The symbols in adjacent rows are likely to be confused. The communication channel has higher capacity if the adjacent symbols are far from each other.

Roots of unity representation

Roots of unity can also be used to represent the communication channel. For the communication channel with n symbols, nth roots of unity are used to specify n equally spaced points on the unit circle. These n equally spaced points represent n symbols. The connections, which are specified by the transmitter-receiver misinterpretation possibilities in the channel, are then drawn between the points within the unit circle to obtain the Error Content Graph. For the 7-symbol Communication Channel shown in Figure 76, the coordinates of 7 symbols on the unit circle are

$$\sqrt[7]{1} = \begin{bmatrix} e^{-j*0*2\pi/7} \\ e^{-j*1*2\pi/7} \\ e^{-j*2*2\pi/7} \\ e^{-j*3*2\pi/7} \\ e^{-j*4*2\pi/7} \\ e^{-j*5*2\pi/7} \\ e^{-j*6*2\pi/7} \end{bmatrix}$$

The 7 roots of unity on the unit circle are shown in Figure 77, with the connections between the adjacent points. This arrangement of connections also forms a perfect communication channel (with the highest capacity) for set of 7 source symbols.

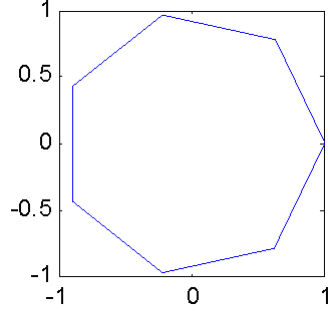


Figure 77 Roots of unity of a 7 symbol communication channel

Fourier matrix representation

As discussed in chapter 1, Fourier matrix generates regular polygons and irregular polygons. For the proper choice of points from the Fourier matrix, any polygon or any error content graph for the communication channel can be drawn.

The choices of points from the Fourier matrix, which are to be used for generating error contents graphs, are specified by permutation matrix representation of the particular communication channel. The points from the second row or column of the matrix table are mapped around the unit circle according to the permutation matrix specification. The error content graph for the 7 symbol communication channel given in Figure 76 can be generated as follows:

$$\begin{bmatrix} e^{-j*0*2\pi/7} \\ e^{-j*1*2\pi/7} \\ e^{-j*2*2\pi/7} \\ e^{-j*3*2\pi/7} \\ e^{-j*4*2\pi/7} \\ e^{-j*5*2\pi/7} \\ e^{-j*6*2\pi/7} \end{bmatrix} * \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix} = \begin{bmatrix} e^{-j*1*2\pi/7} \\ e^{-j*3*2\pi/7} \\ e^{-j*0*2\pi/7} \\ e^{-j*6*2\pi/7} \\ e^{-j*4*2\pi/7} \\ e^{-j*2*2\pi/7} \\ e^{-j*5*2\pi/7} \end{bmatrix}$$

Modeling and Analysis of Noisy Communication Channels

As illustrated in the previous sections, one can see that there exists connections among complex numbers, combinatorial theory, Fourier series, graph theory and communication channel. A noisy communication channel framework has been proposed to model systems in general with four analysis tools: complex analysis, combinatorial analysis, Chebyshev analysis and Fourier analysis [11,22].

The work can be concluded in Figure 78. The noisy communication channel framework is connected to complex analysis with roots of unity. The framework is connected to combinatorial analysis with permutation matrix. The framework is connected to Fourier analysis with polygons, which has been illustrated in detail in chapter 2. The framework is connected to Chebyshev analysis with Chbyshev polynomial.

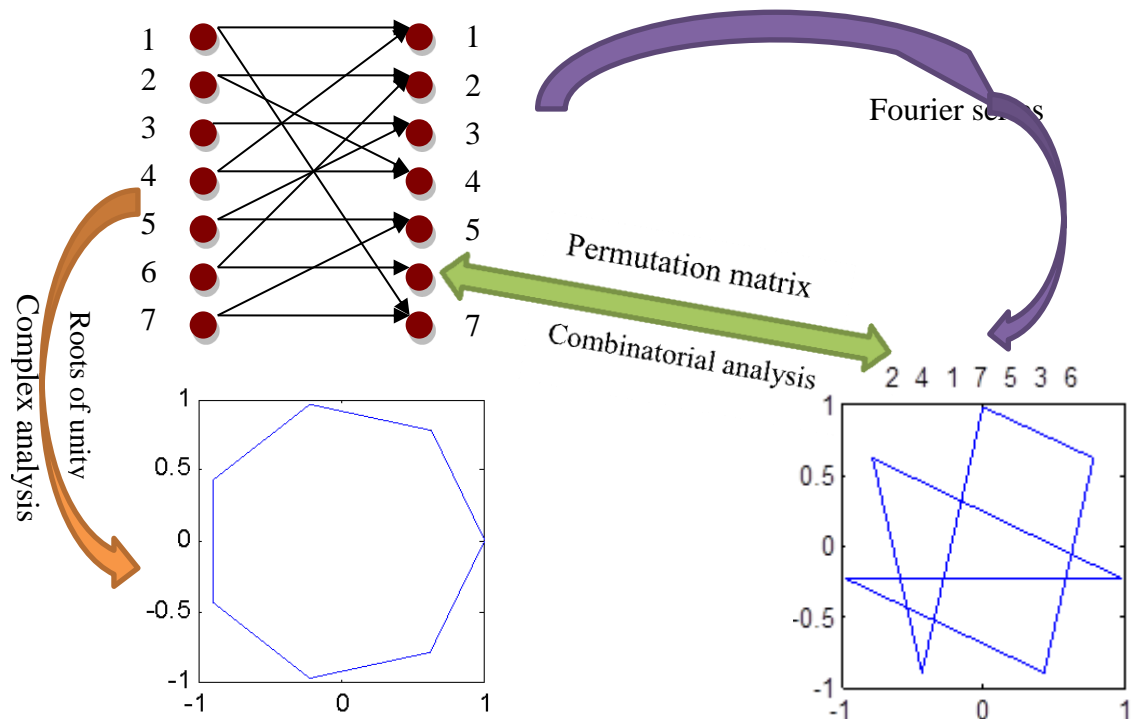


Figure 78 The communication channel framework and its connections to common analysis

Appendix C: More results of U matrix

Now let's show the example of $n=7$ and $q=2$. The results of changing p from 1 to 10 are shown in Table 14.

Table 14 p changes from 1 to 9 for $n=7$ and $q=2$

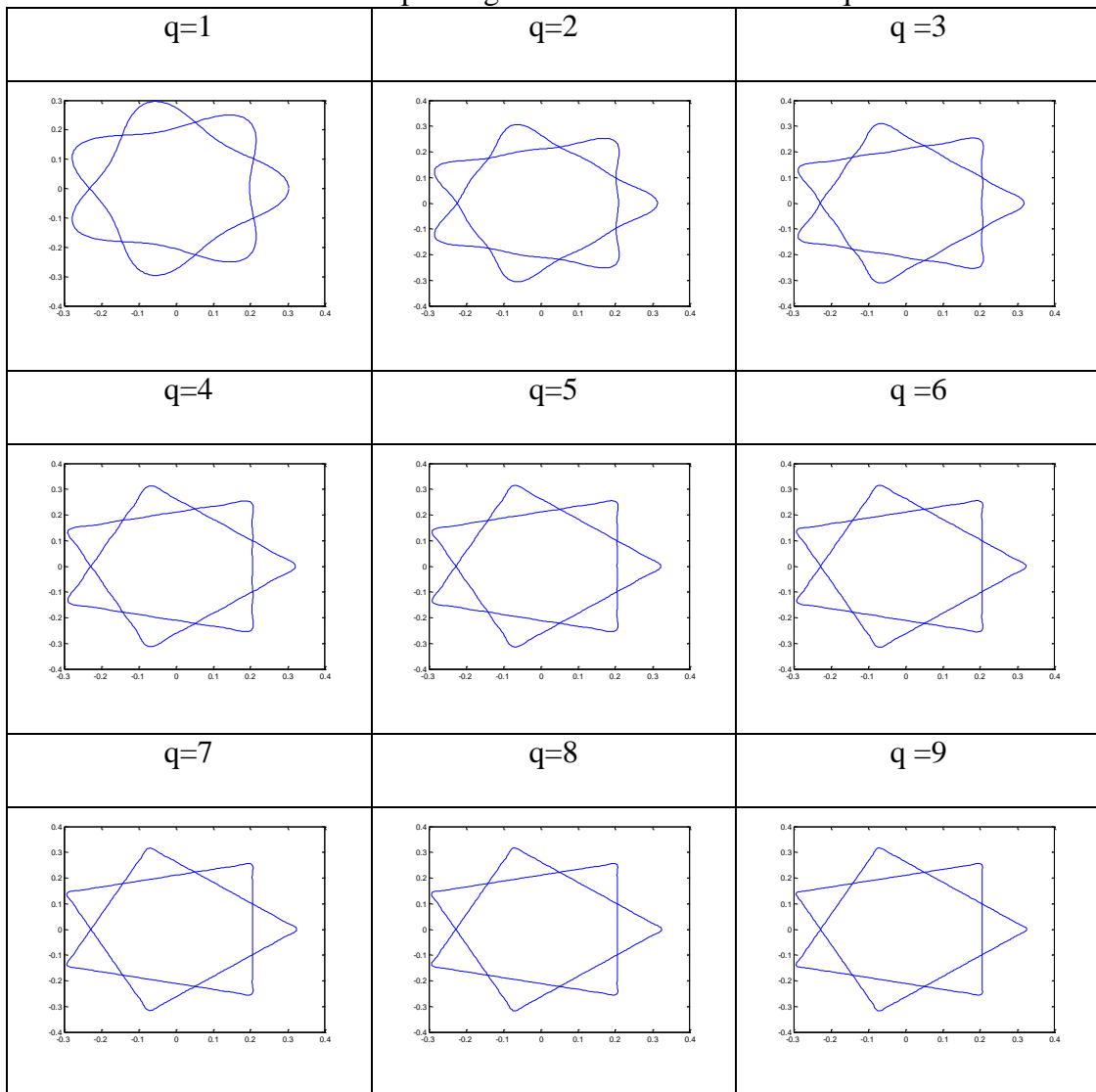
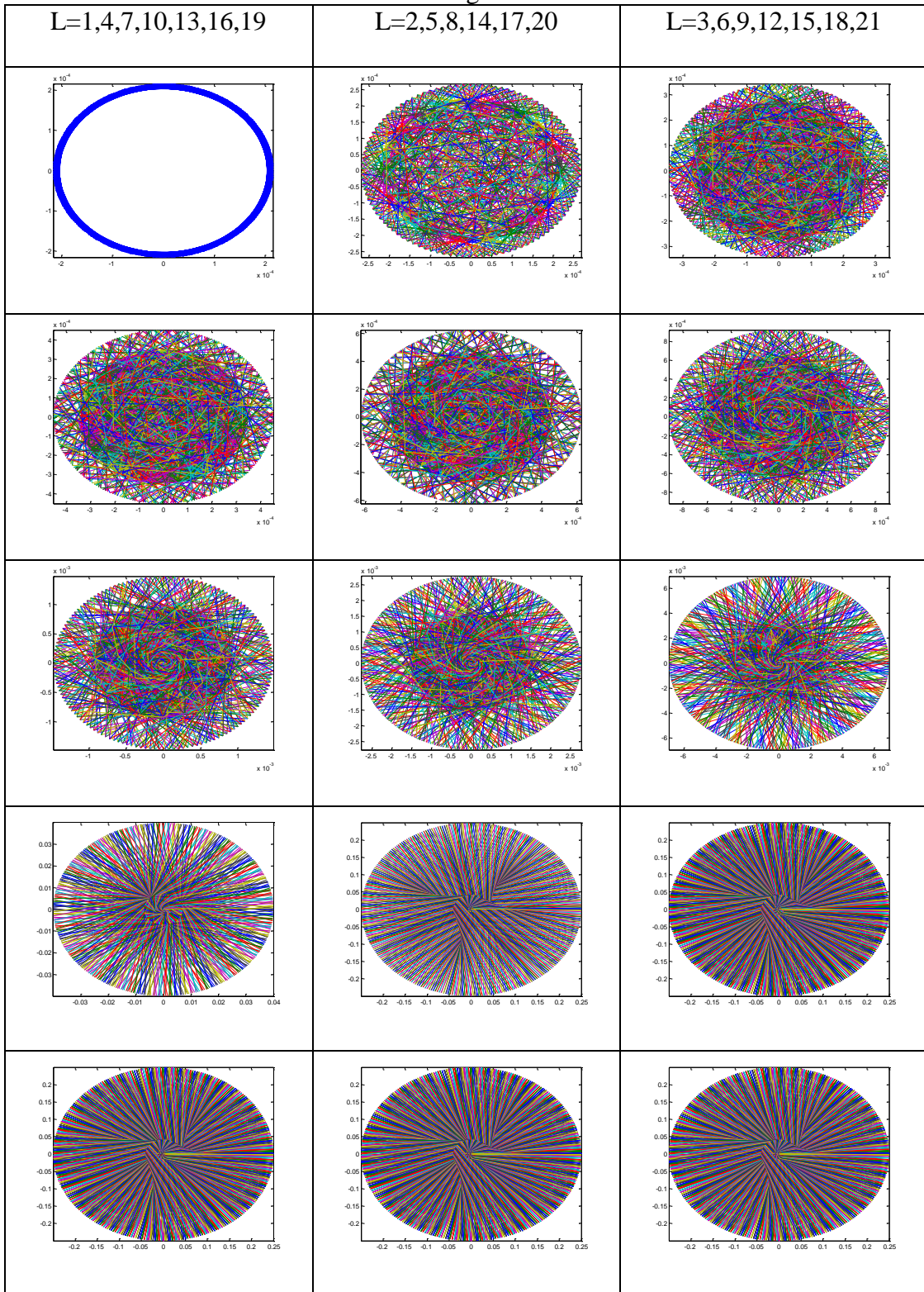
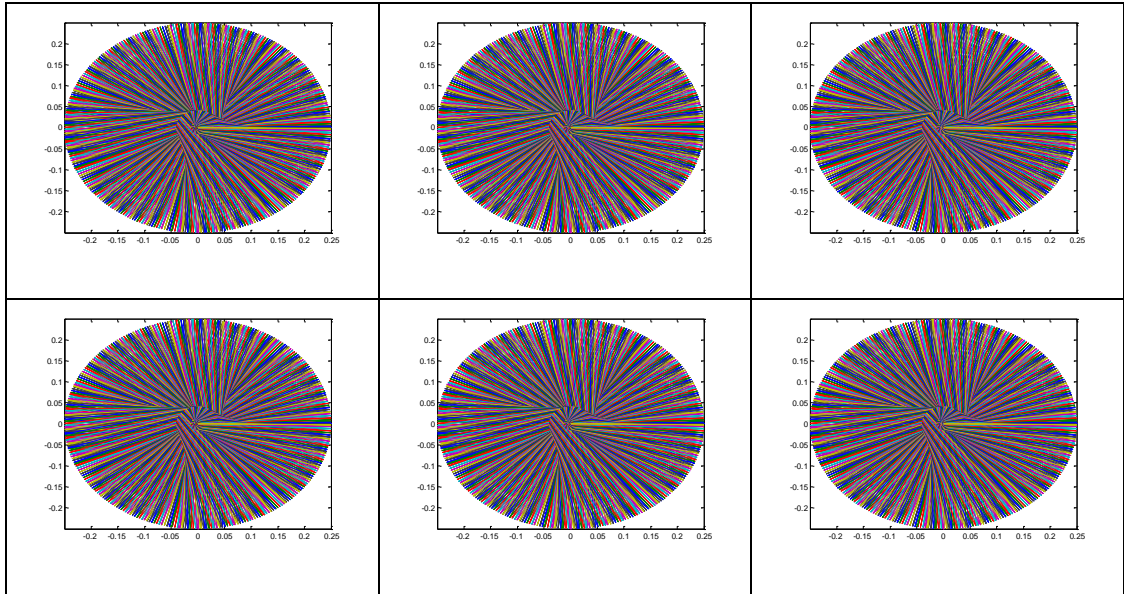


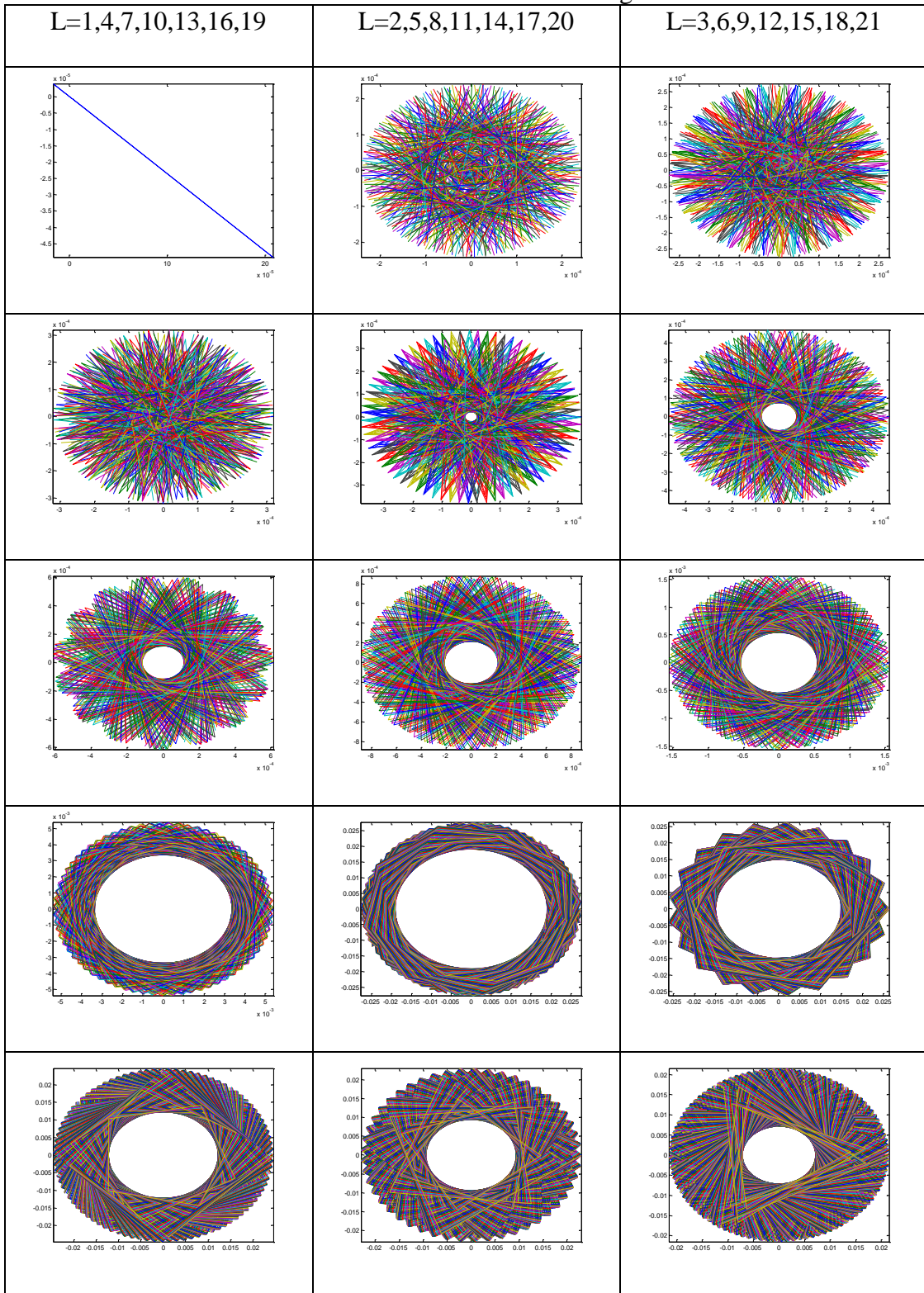
Table 15 U matrix changes as row number increases

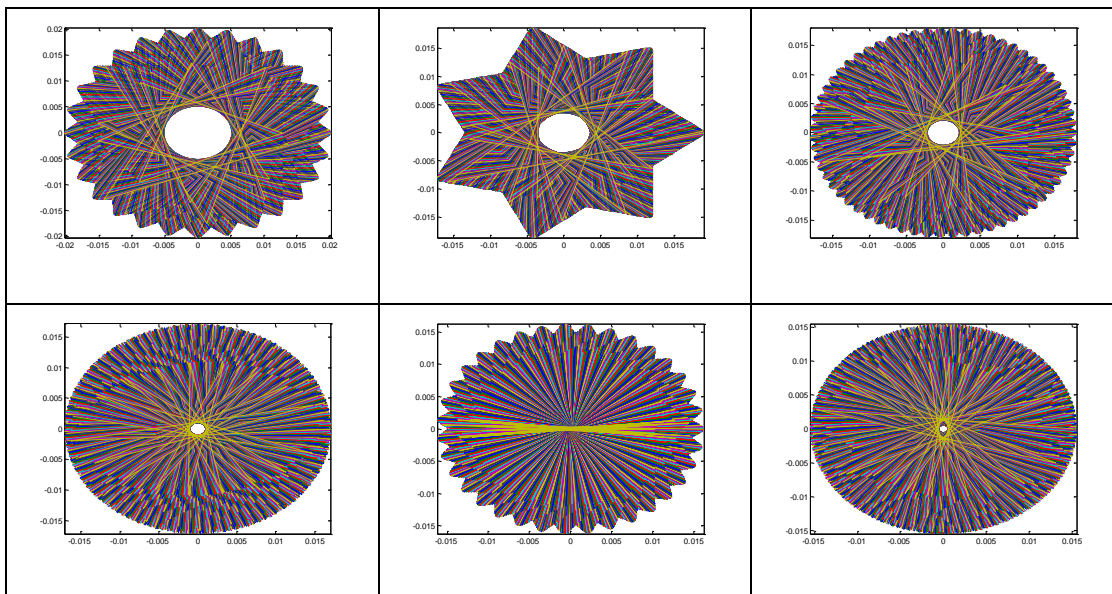




The inverse Fourier transform of the U matrix is shown in Table 16.

Table 16 Inverse Fourier transform of U matrix changes as row number increases





If we draw the each row of the U matrix, we obtain the result as shown in Figure 79, which responds to the atomic orbital.

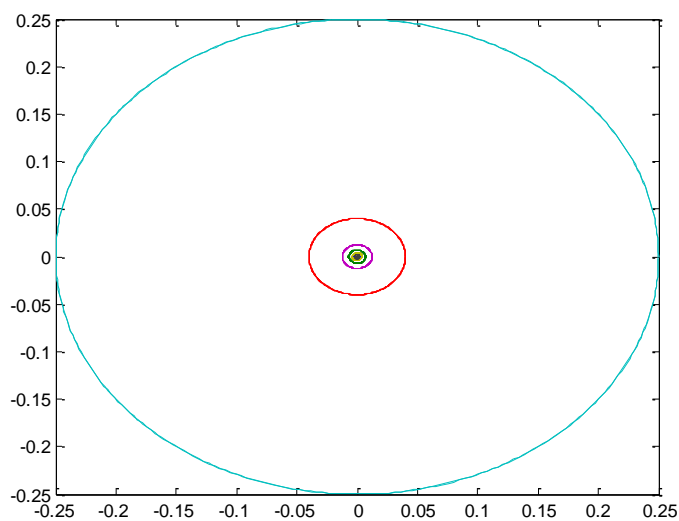


Figure 79 Row drawing of $u_{(7,2)}$

Now let's show the example of $n=7$ and $q=3$. The results of changing p from 1 to 10 are shown in Table 17.

Table 17 p changes from 1 to 10 for n=7 and q=3

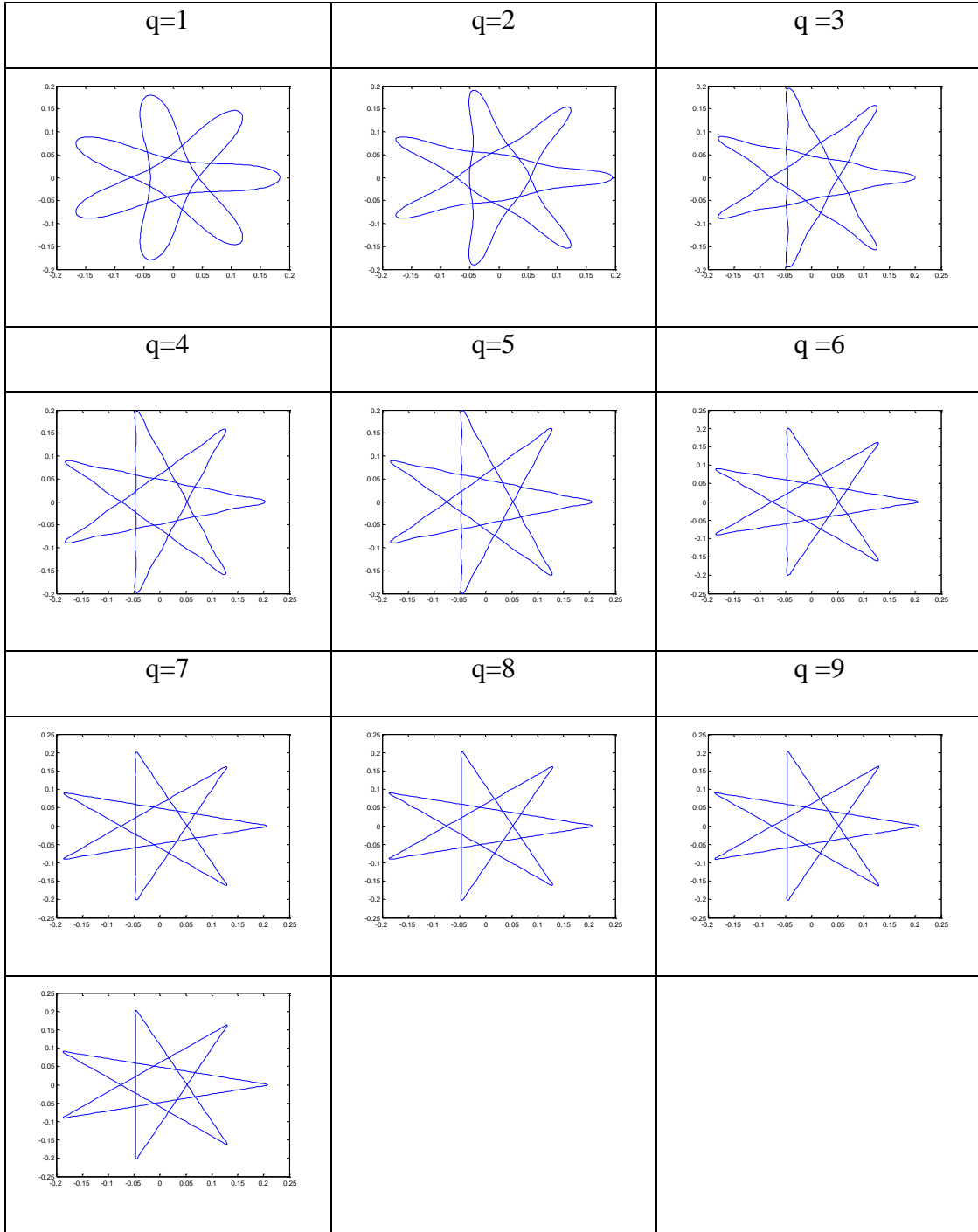
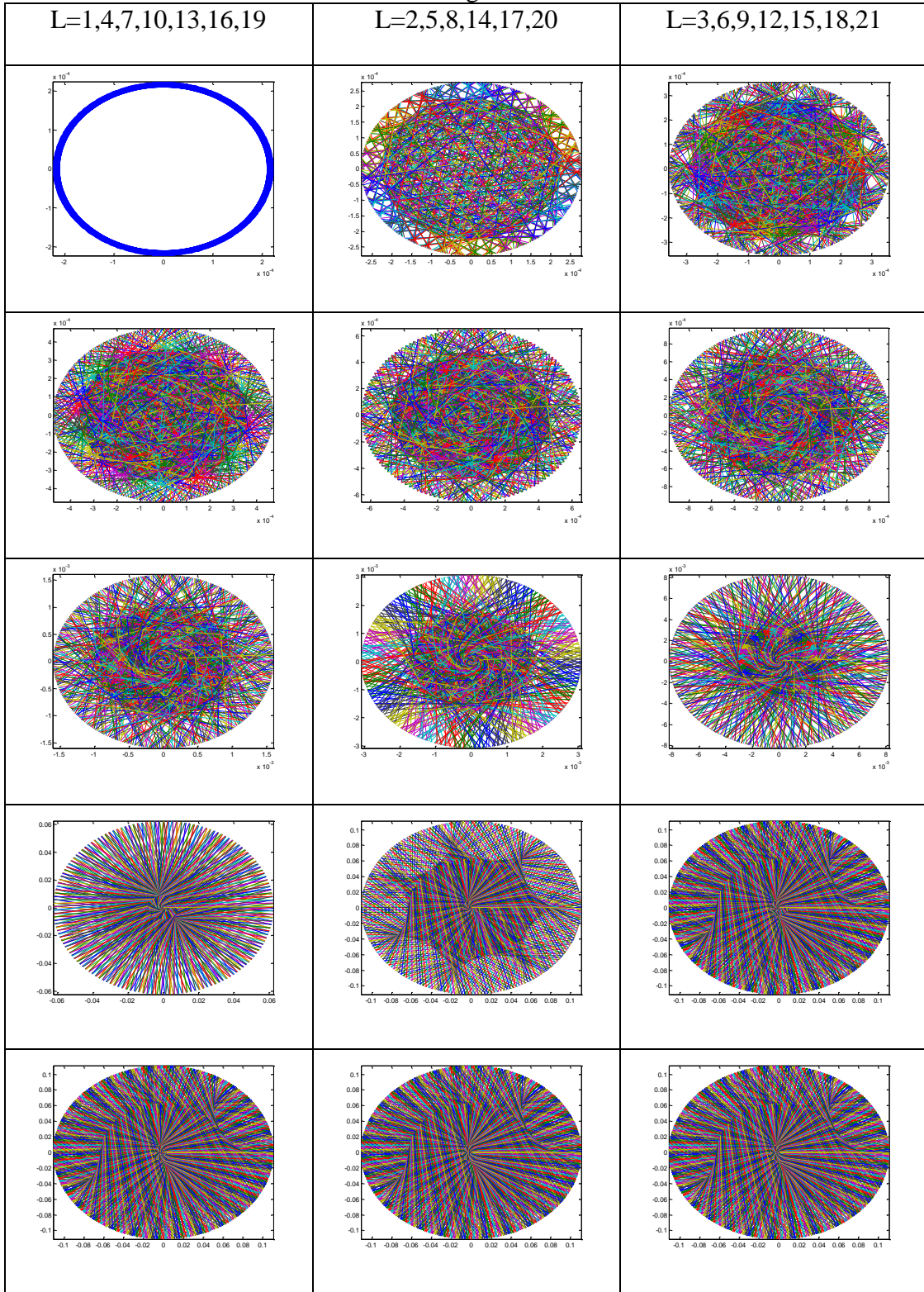


Table 18 U matrix changes as row number increases



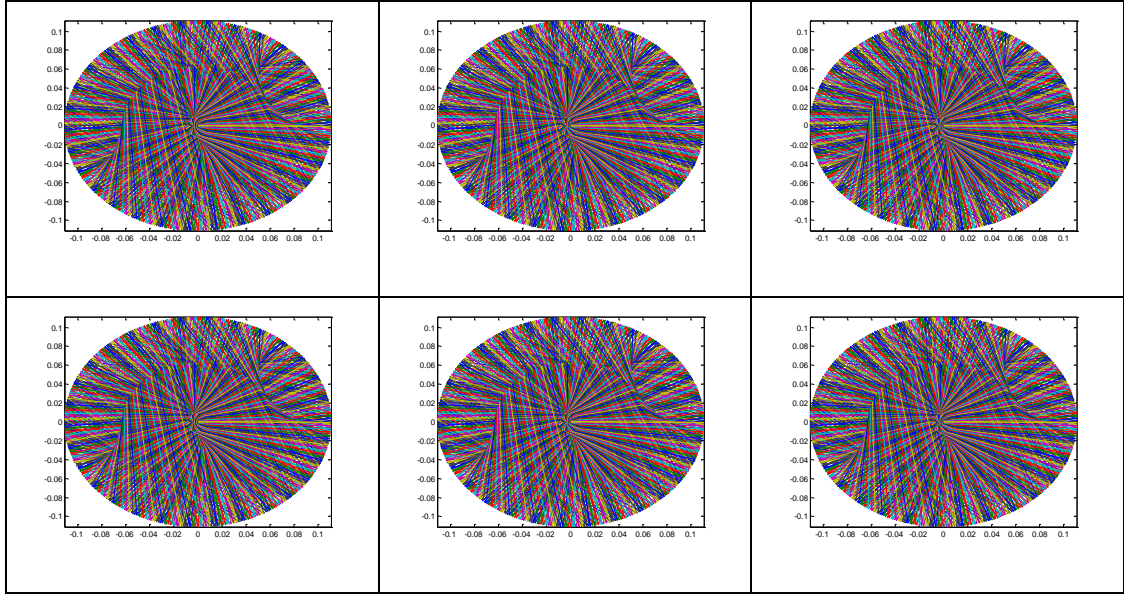
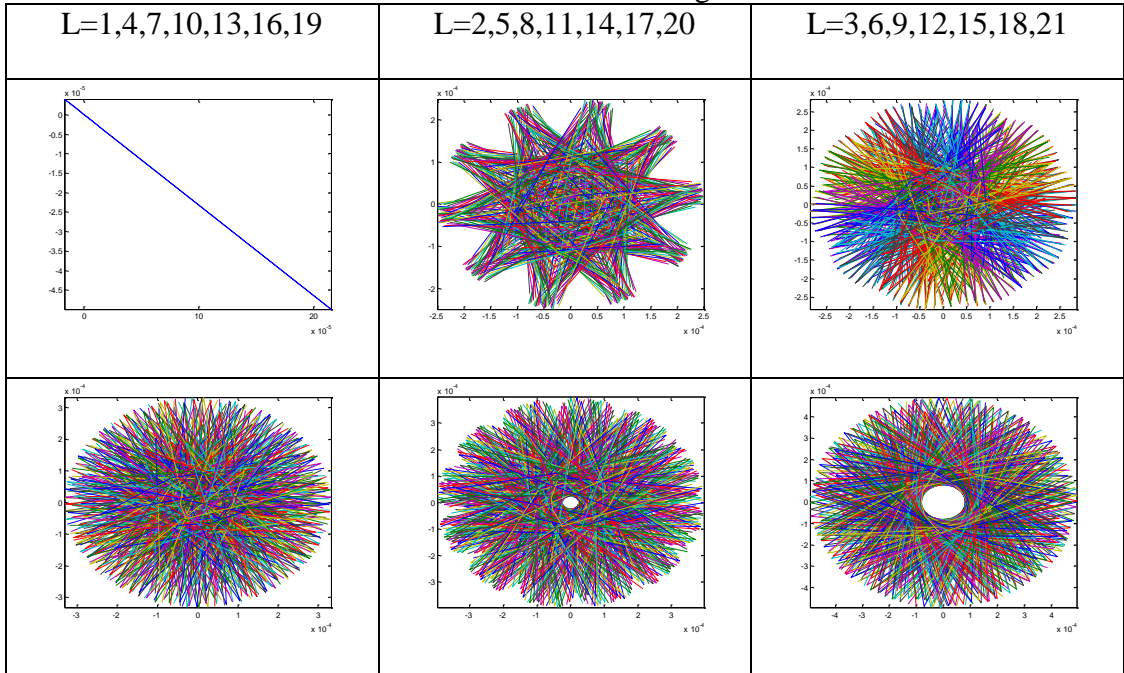
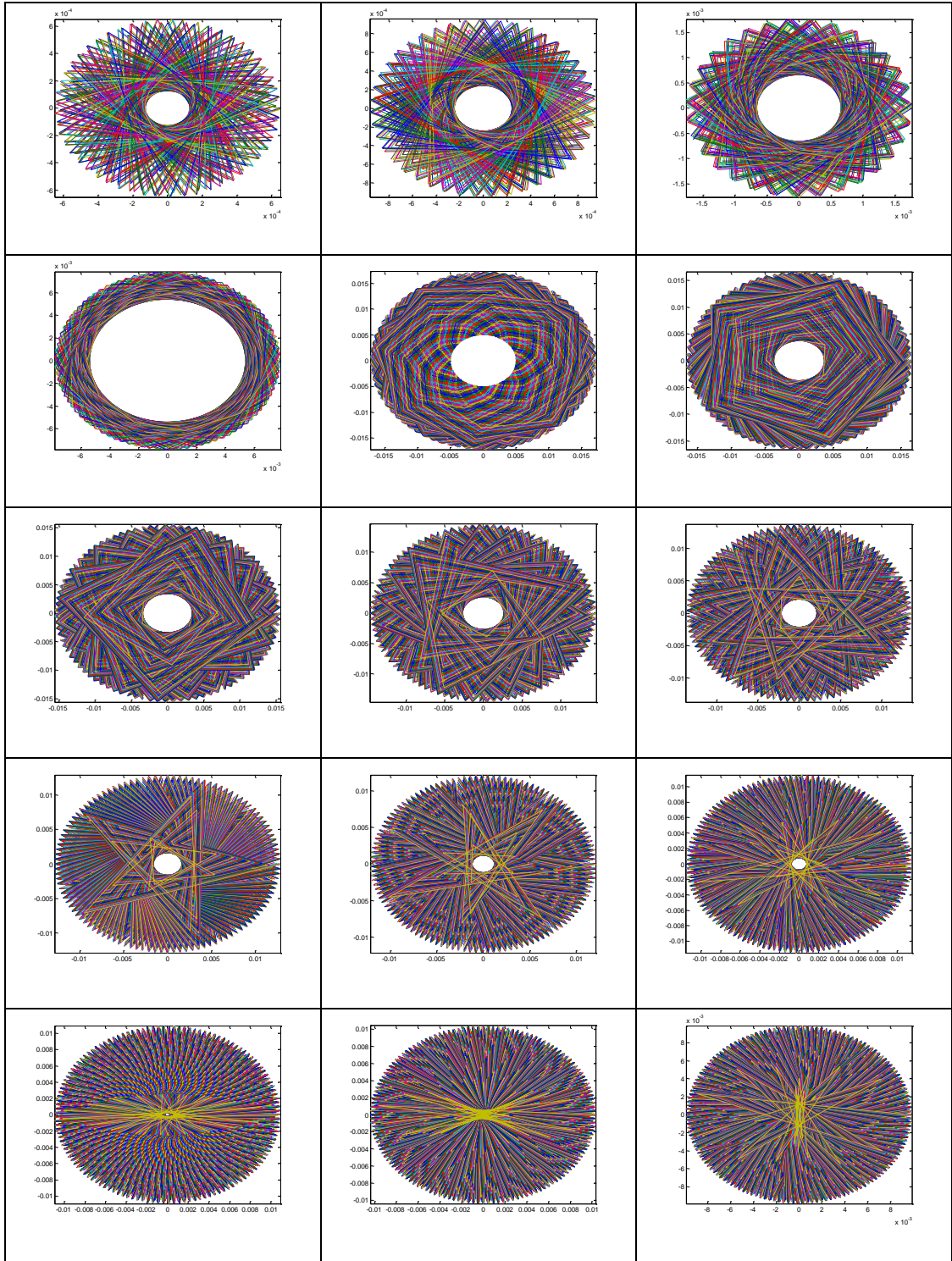


Table 19 Inverse Fourier transform of U matrix changes as row number increases





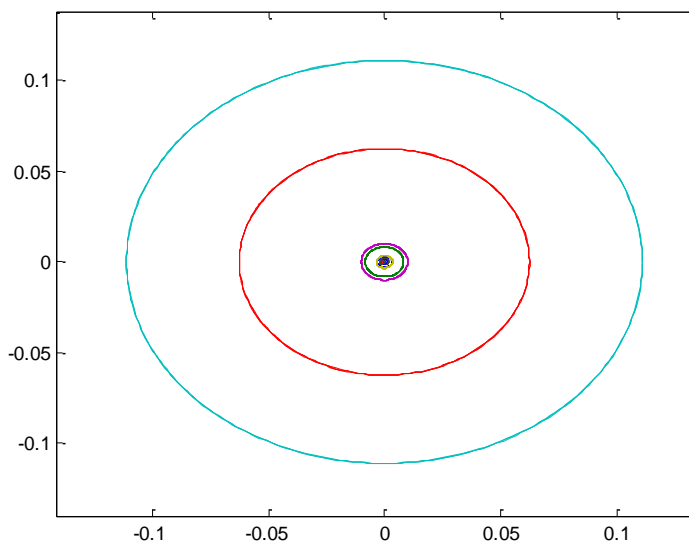


Figure 80 Row drawing of $u_{(7,3)}$

Now let's show the example of $n=7$ and $q=4$. The results of changing p from 1 to 10 are shown in Table 20.

Table 20 p changes from 1 to 10 for $n=7$ and $q=4$

q=1	q=2	q=3
q=4	q=5	q=6

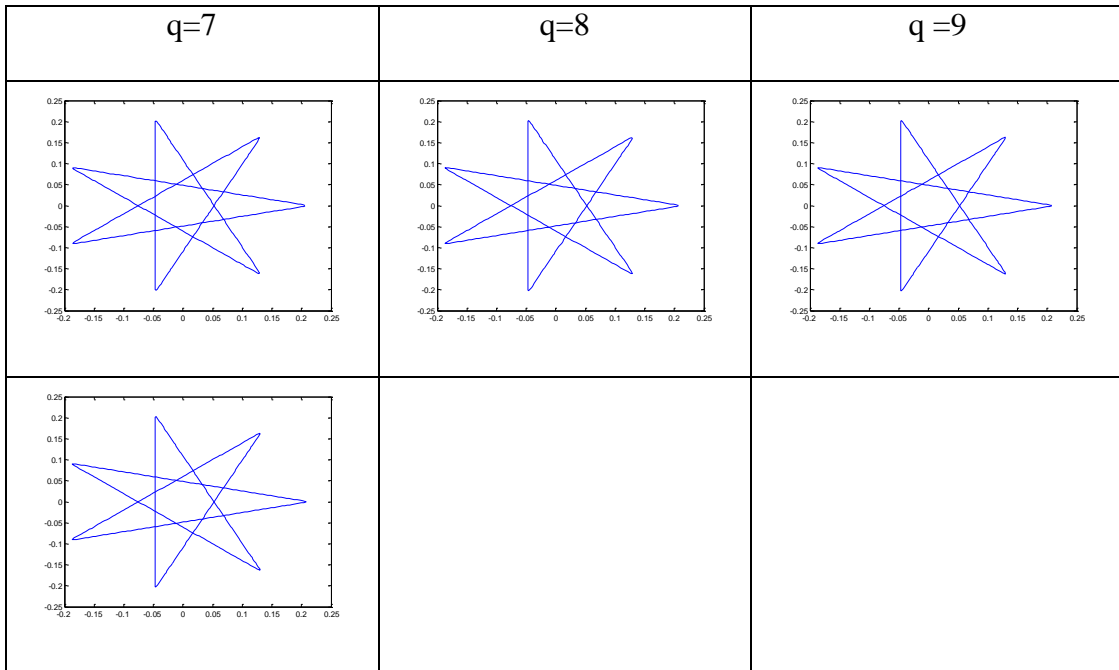
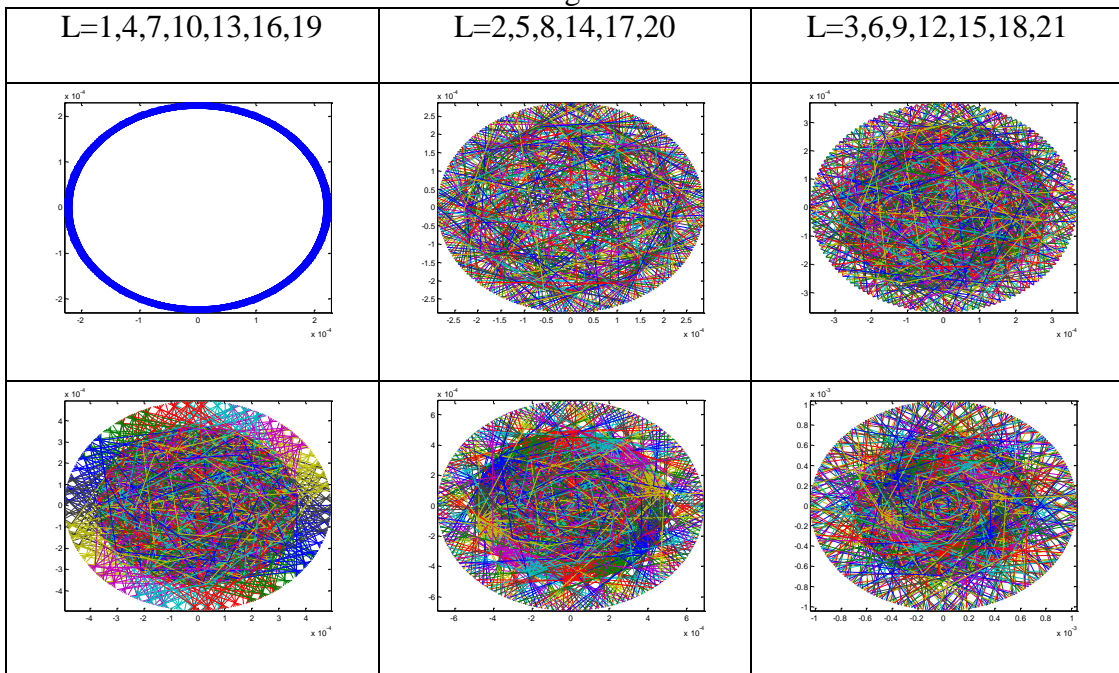


Table 21 U matrix changes as row number increases



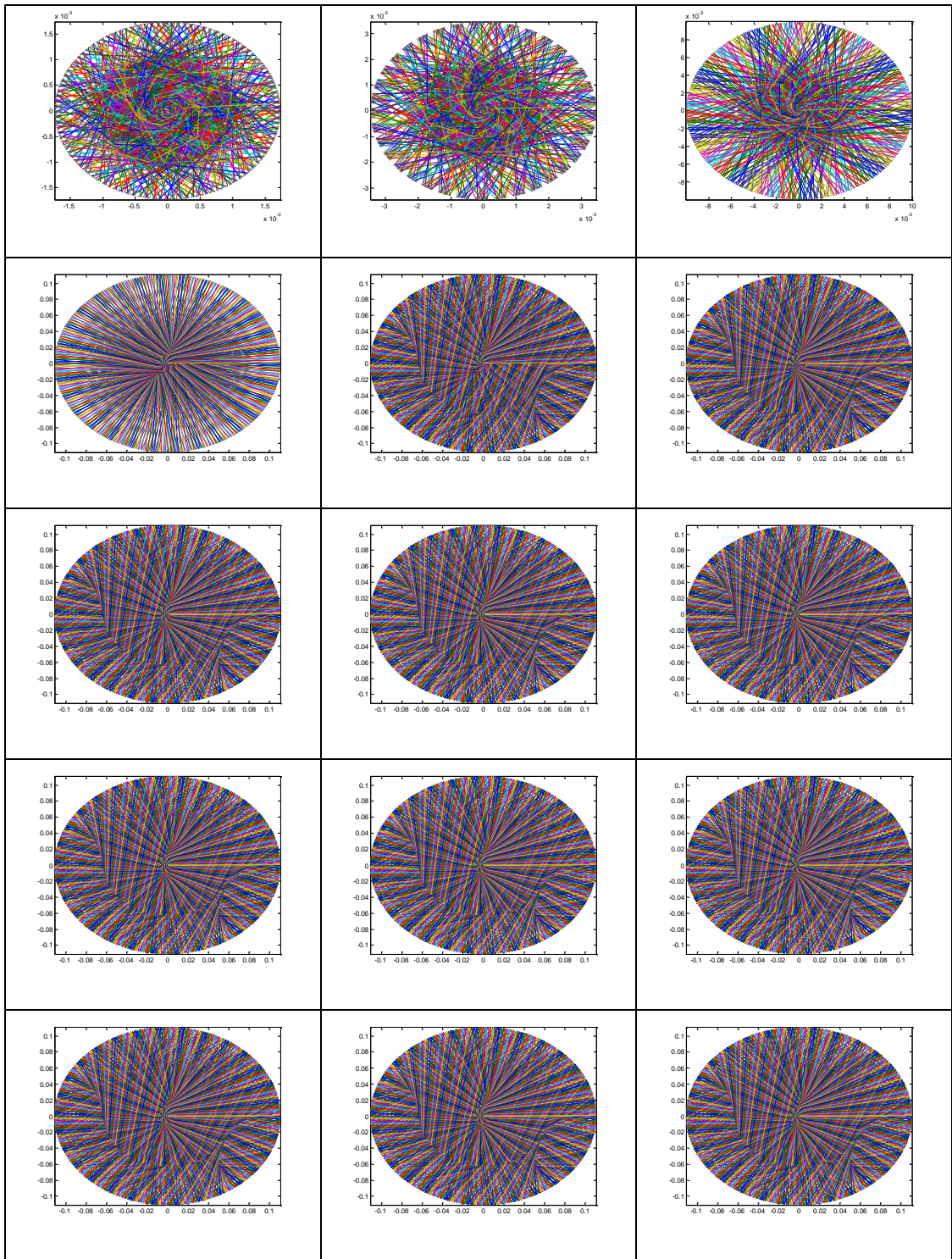
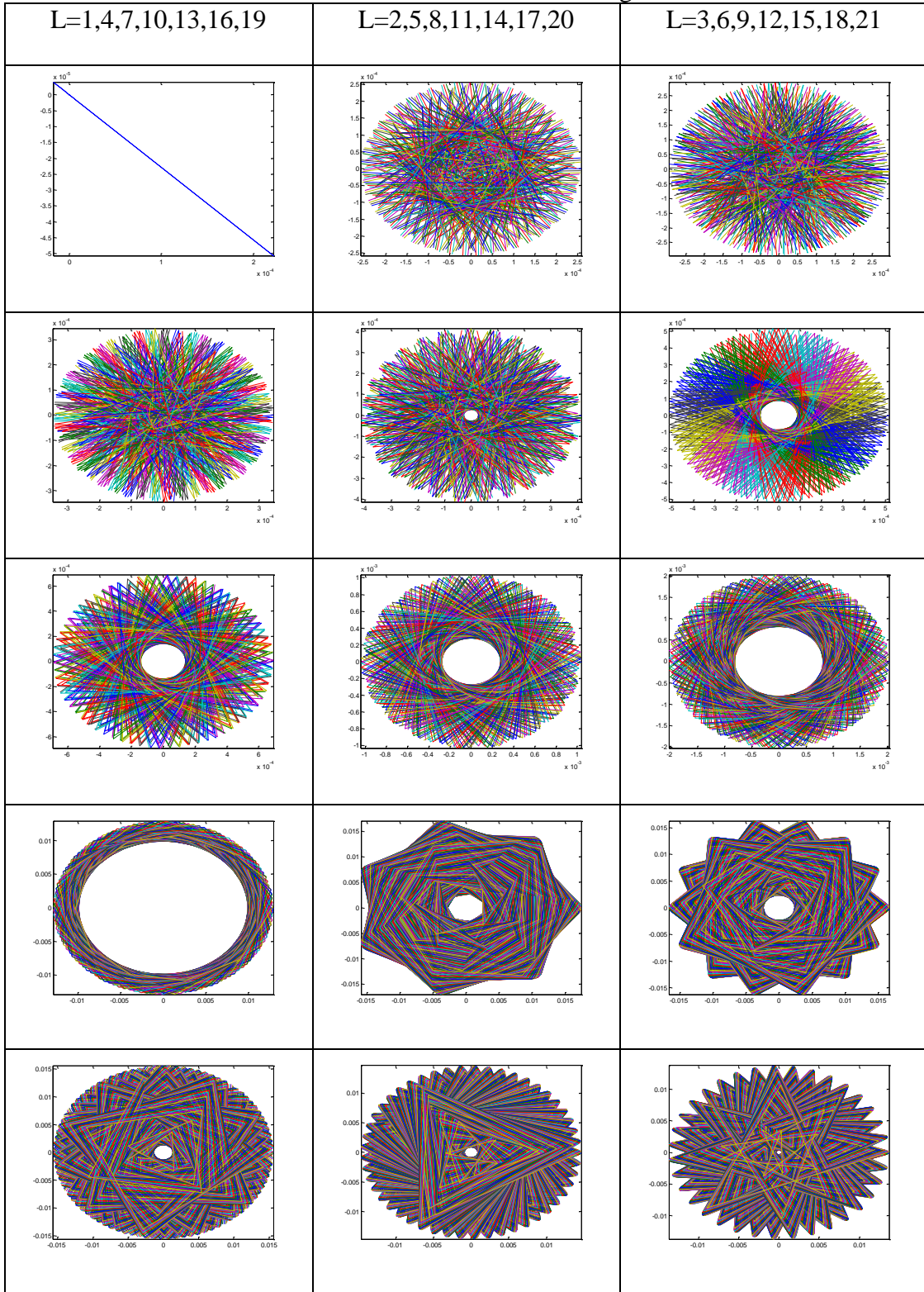


Table 22 Inverse Fourier transform of U matrix changes as row number increases



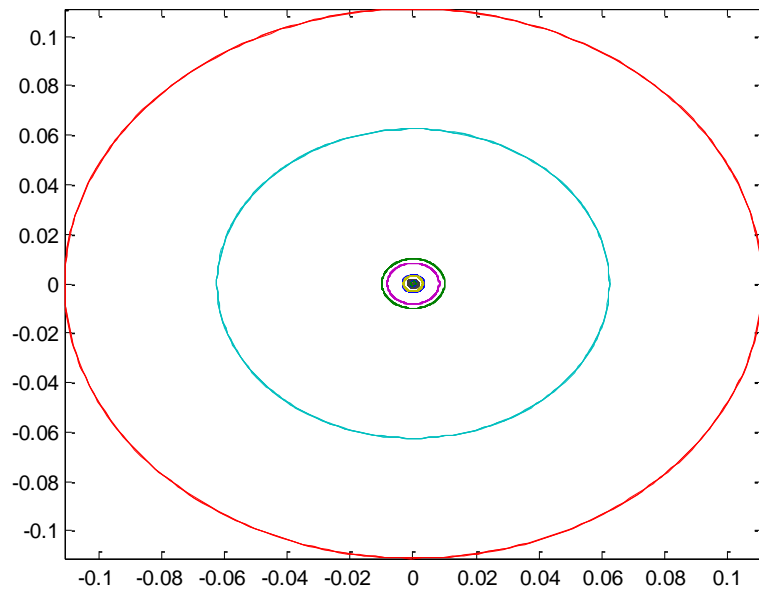
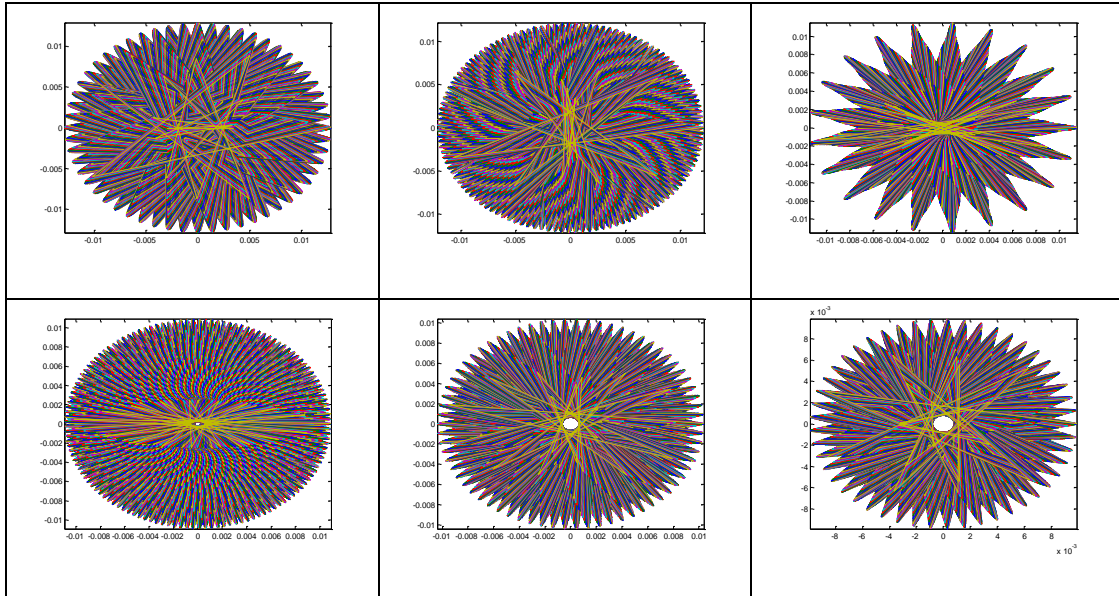


Figure 81 Row drawing of $u_{(7,4)}$

Now let us show the example of $n=7$ and $q=5$. The results of changing p from 1 to 10 are shown in Table 23.

Table 23 p changes from 1 to 10 for n=7 and q=5

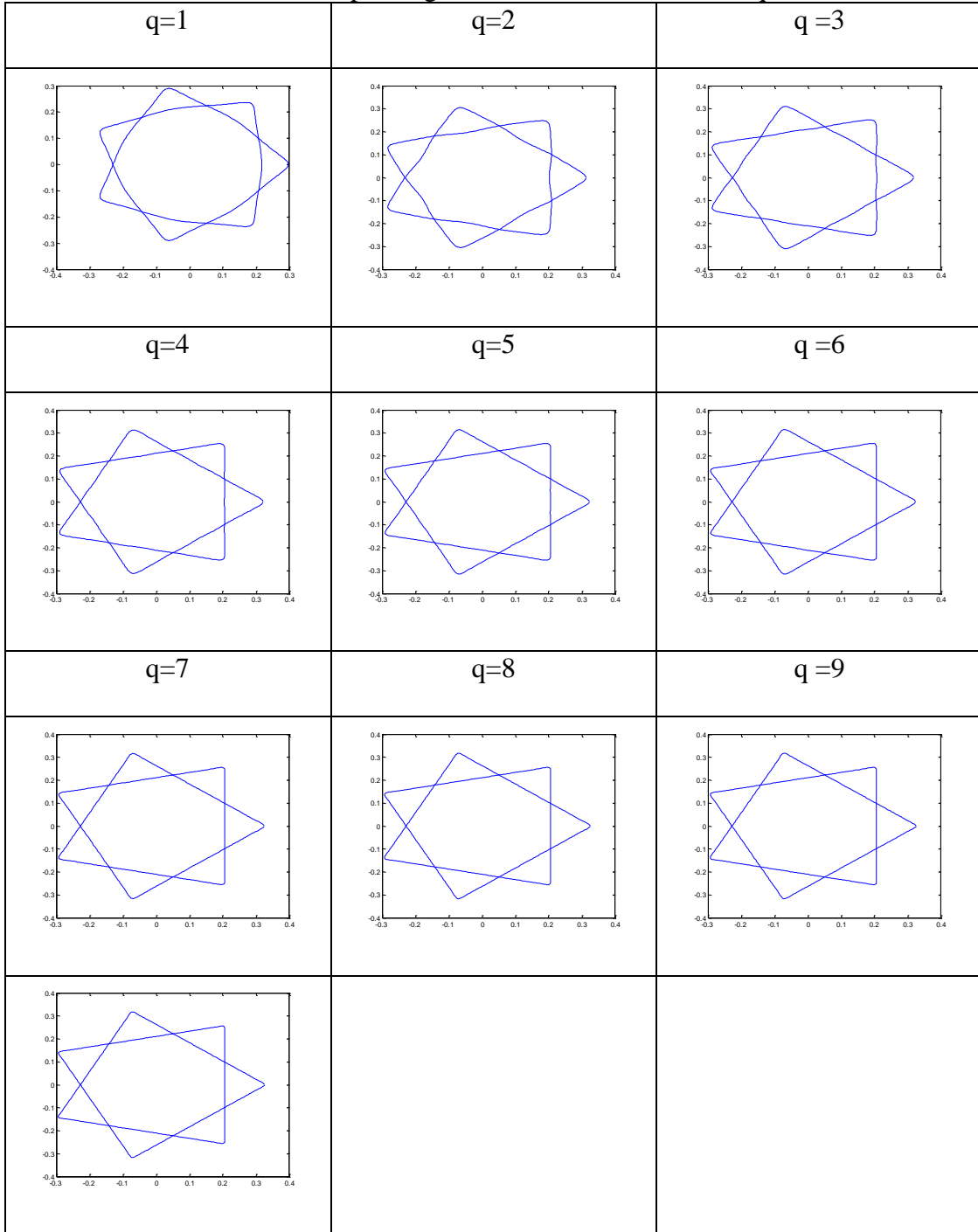
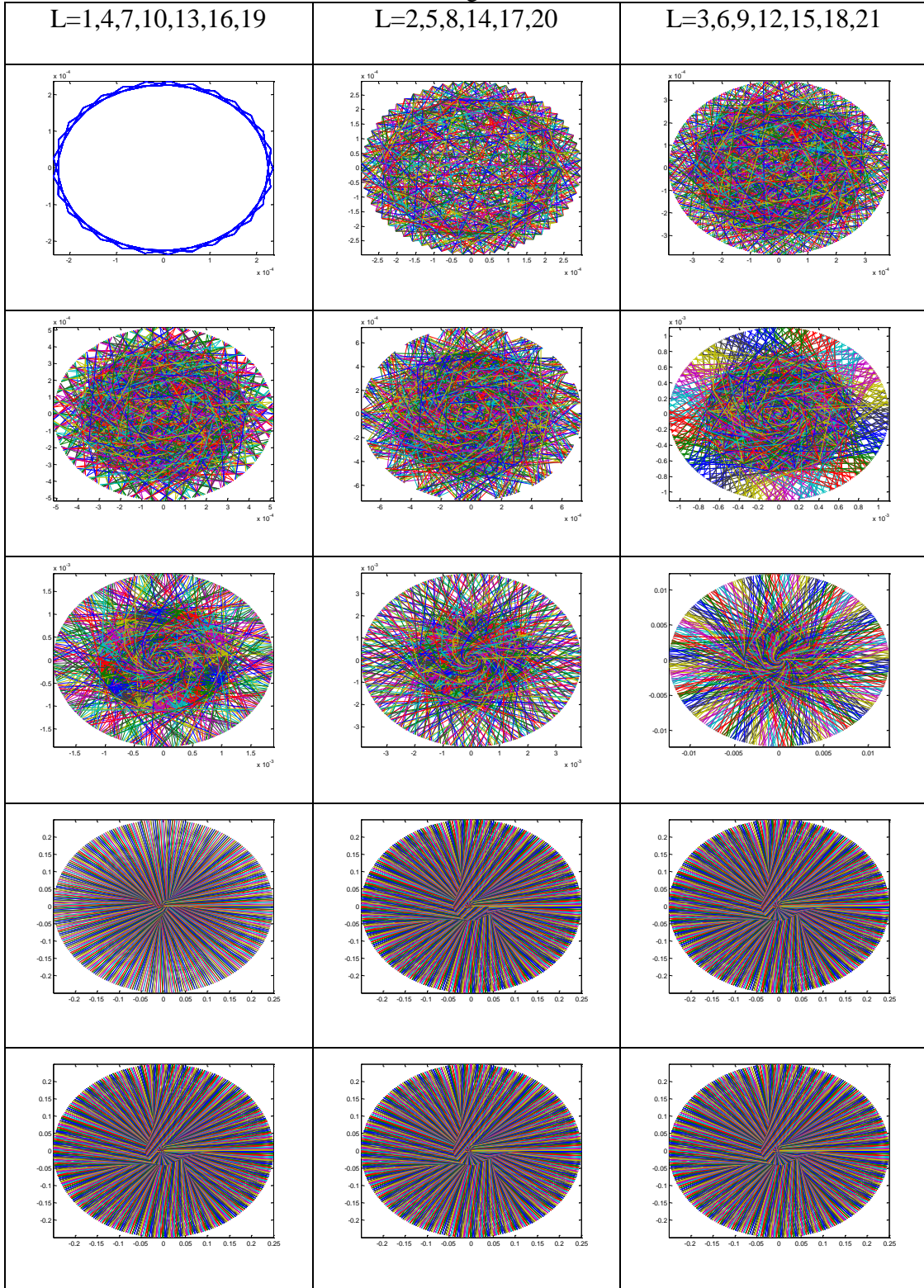


Table 24 U matrix changes as row number increases



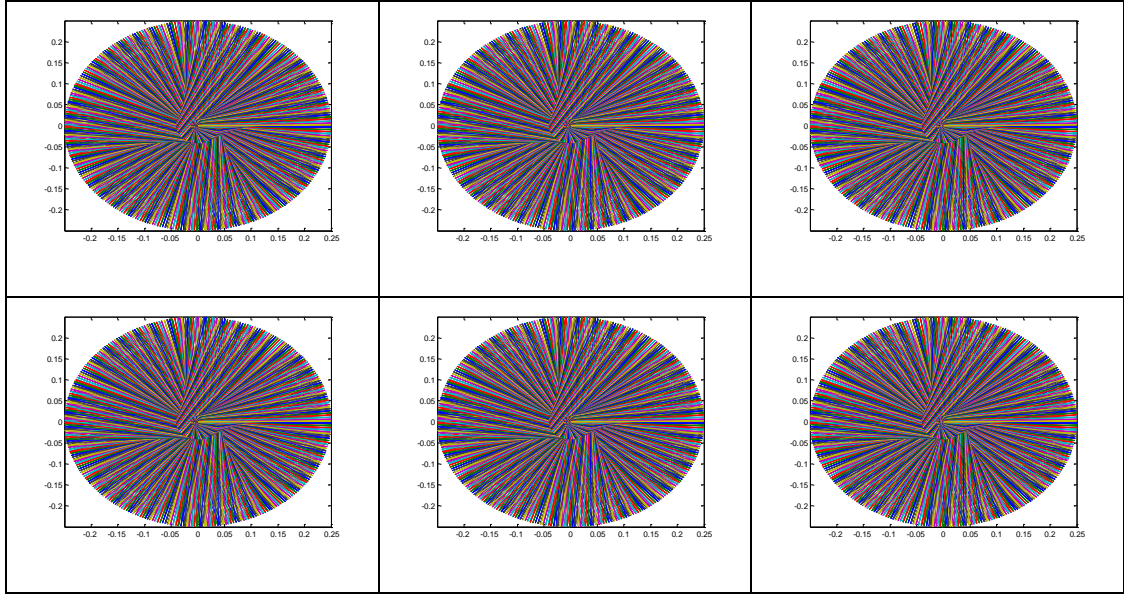
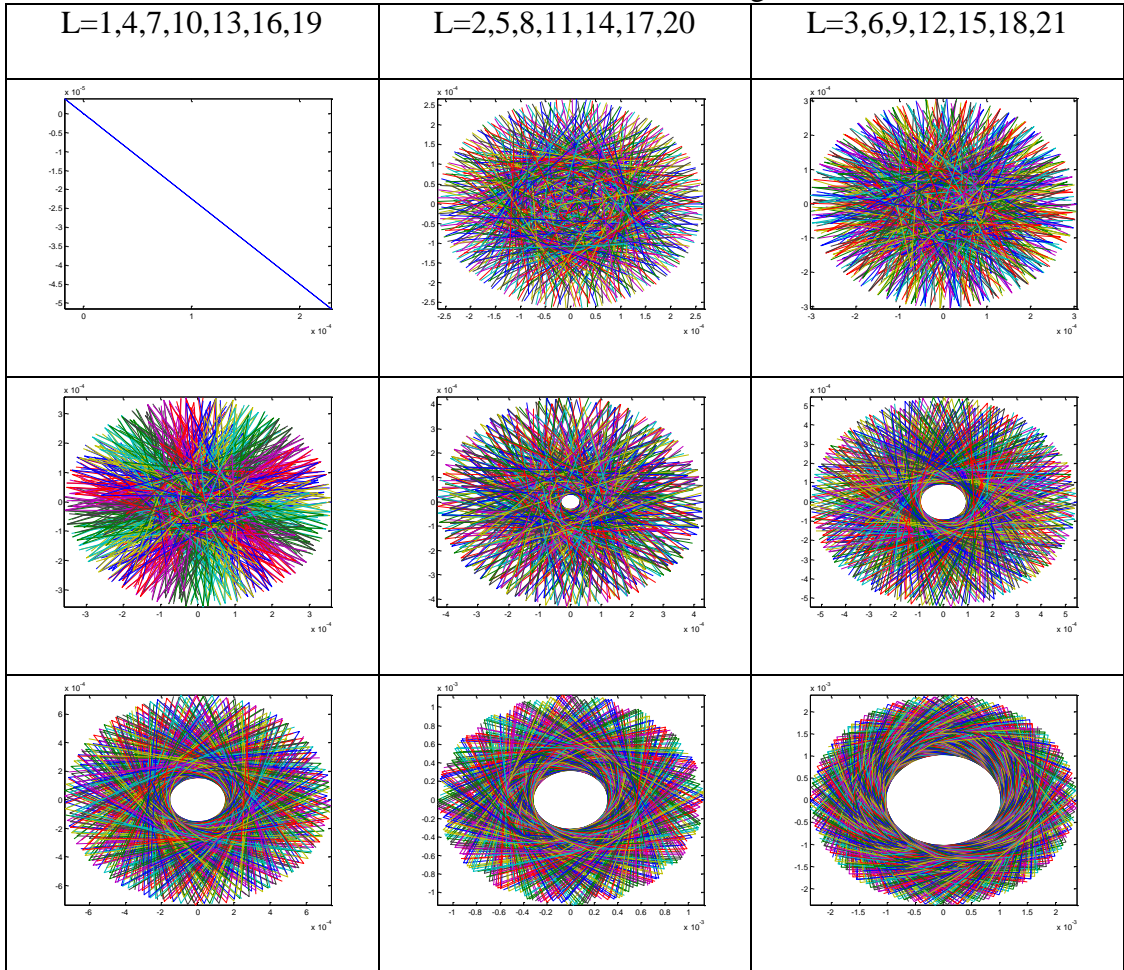
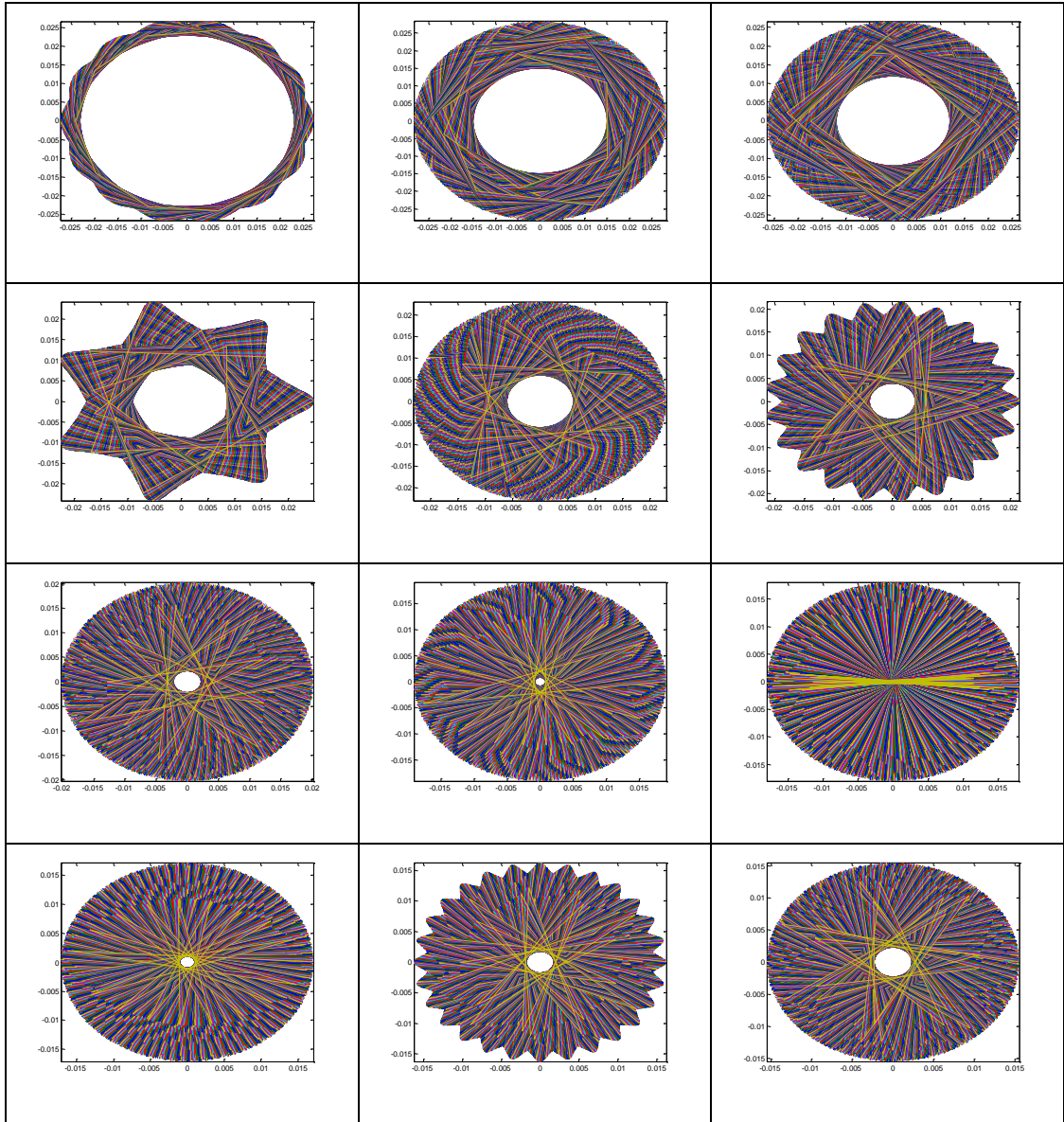


Table 25 Inverse Fourier transform of U matrix changes as row number increases





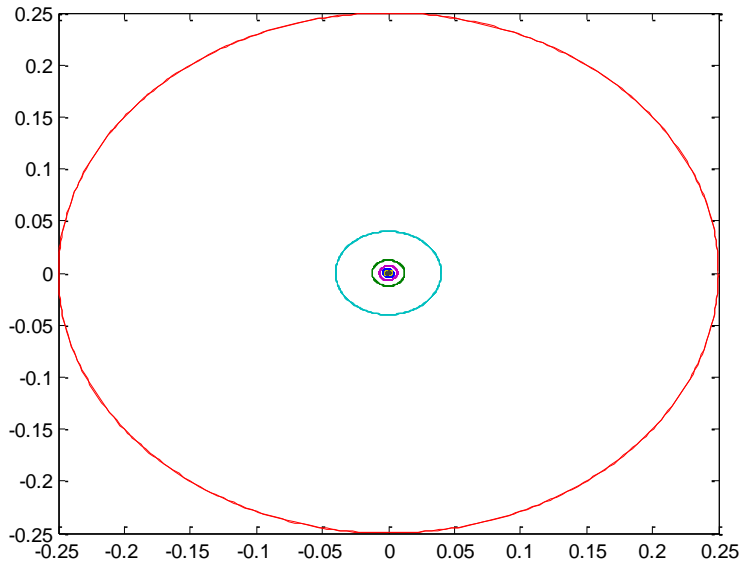
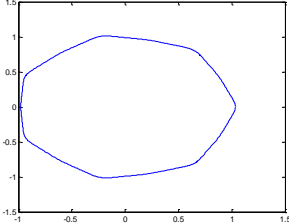
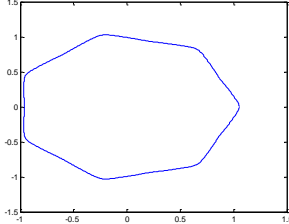
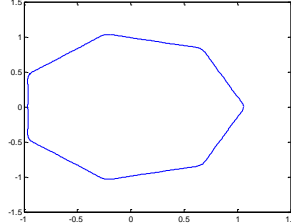
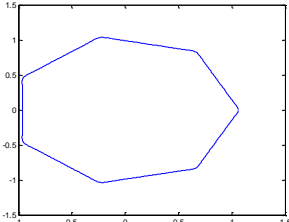
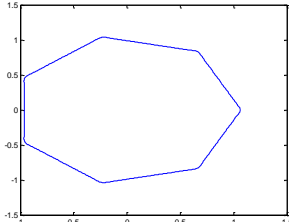
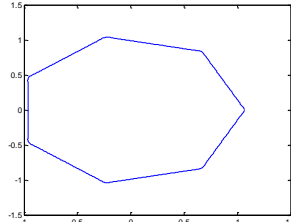


Figure 82 Row drawing of $u_{(7,5)}$

Now let's show the example of $n=7$ and $q=6$. The results of changing p from 1 to 10 are shown in Table 26.

Table 26 p changes from 1 to 10 for $n=7$ and $q=6$

q=1	q=2	q=3
		
q=4	q=5	q=6
		

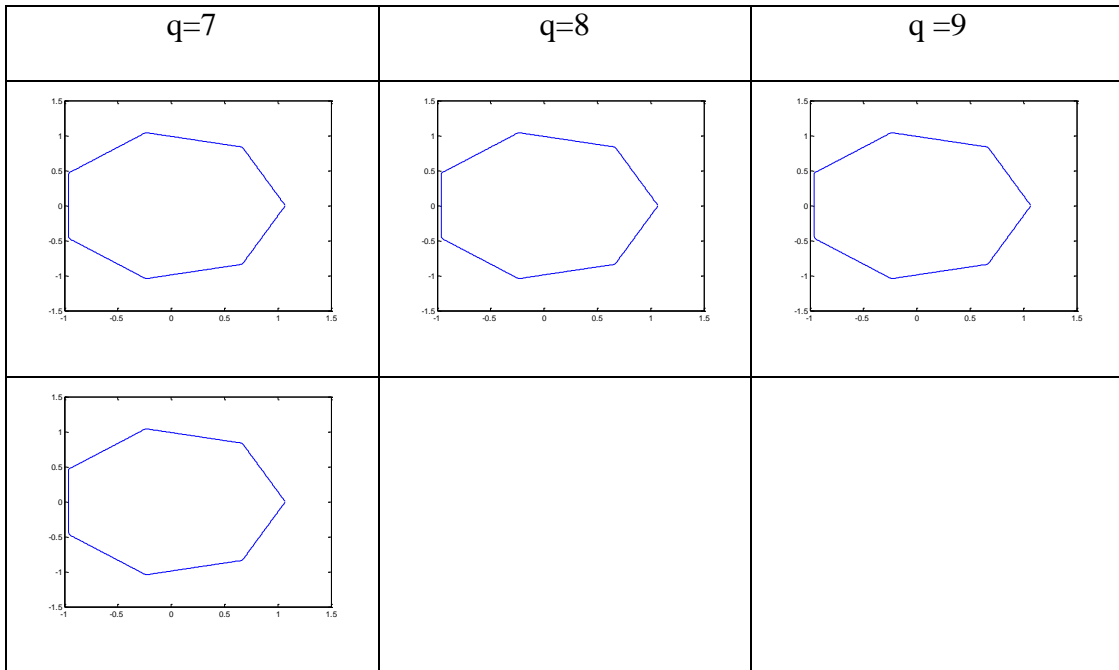
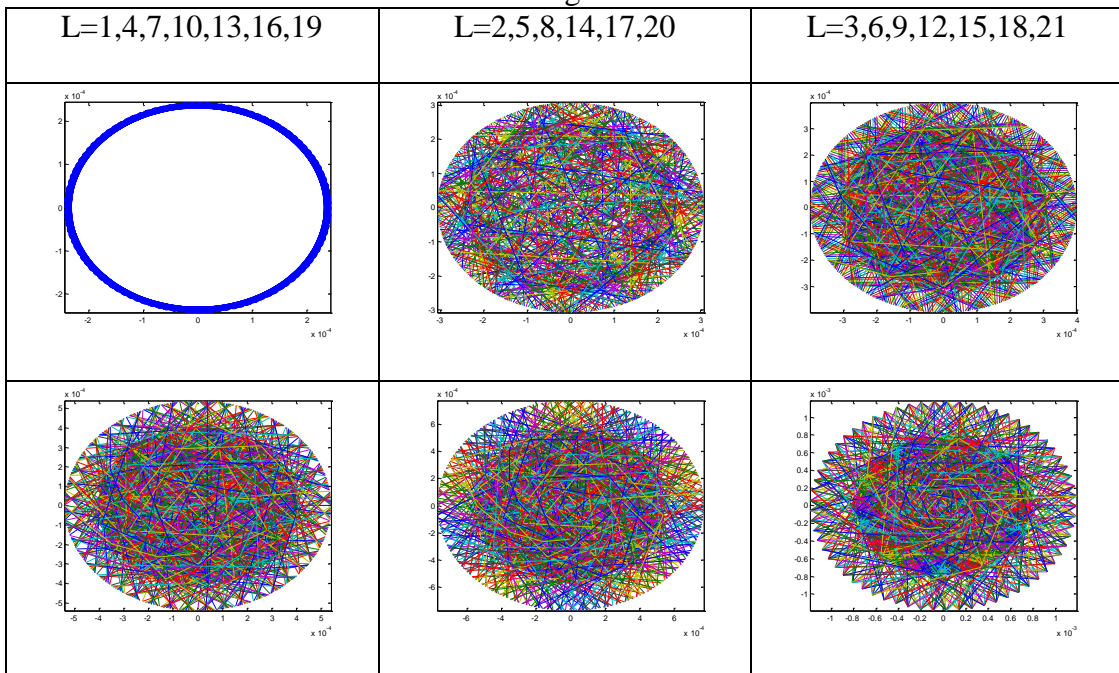


Table 27 U matrix changes as row number increases



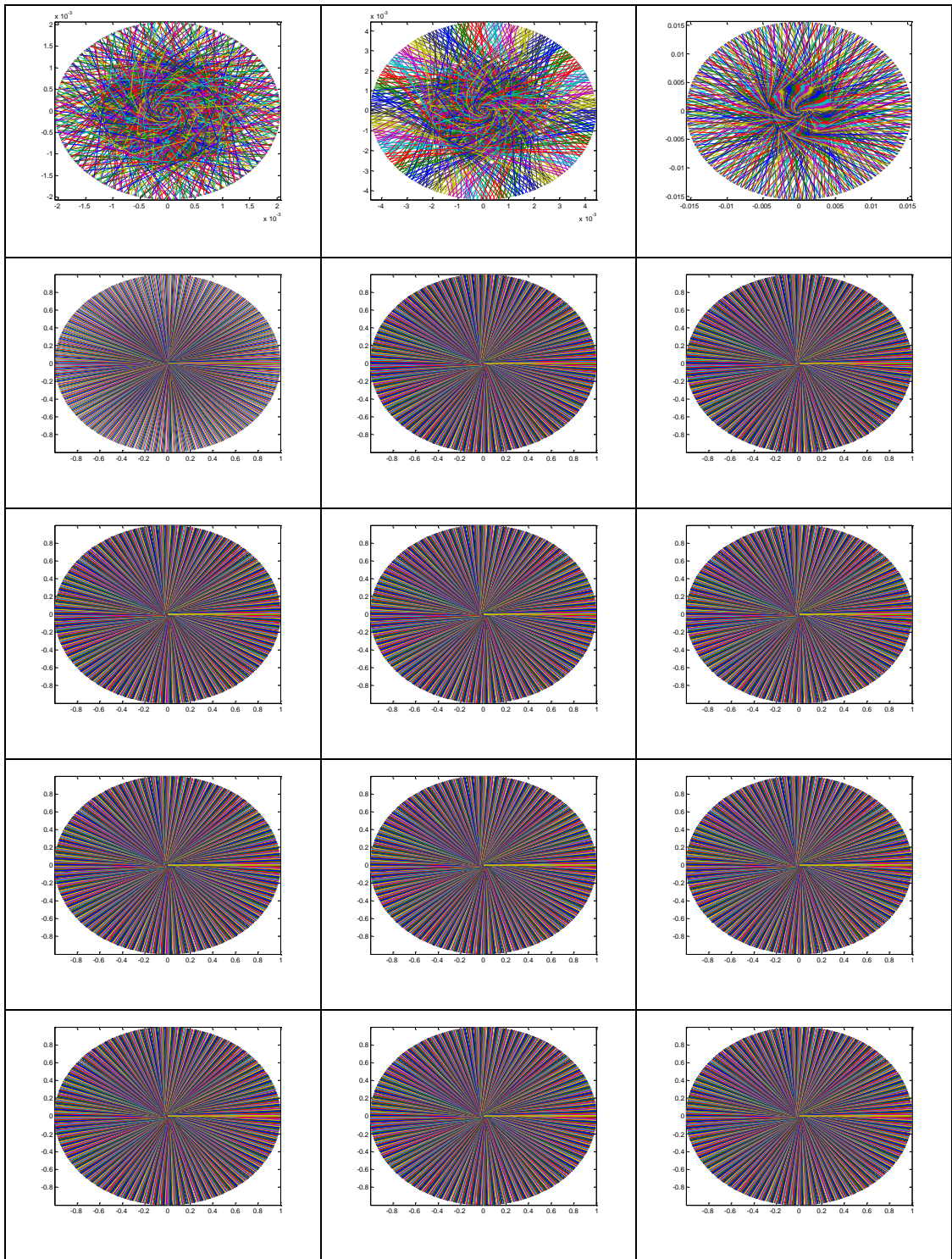
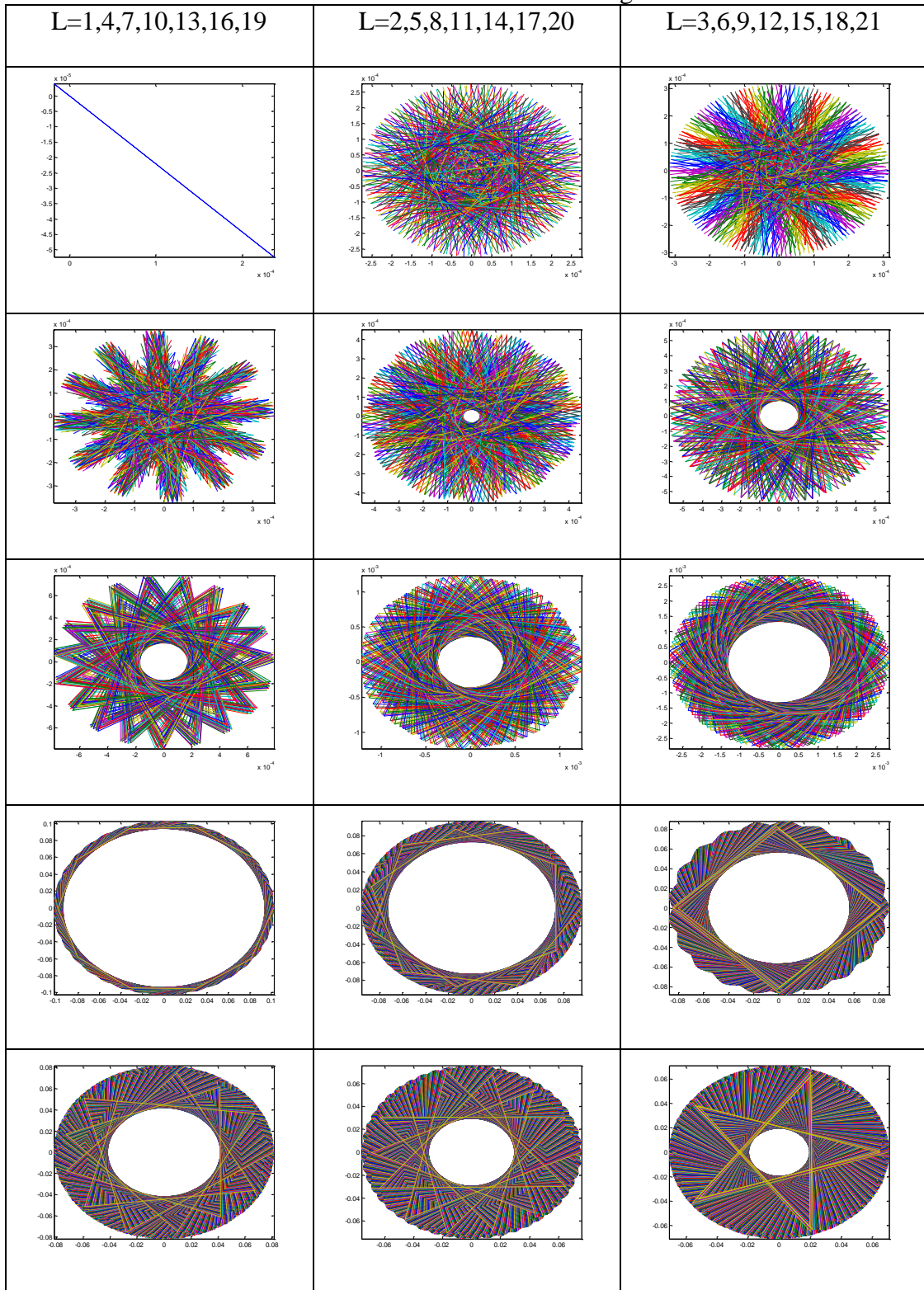


Table 28 Inverse Fourier transform of U matrix changes as row number increases



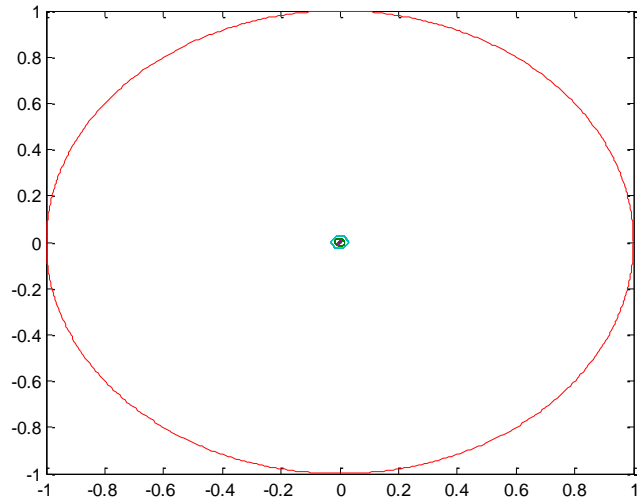
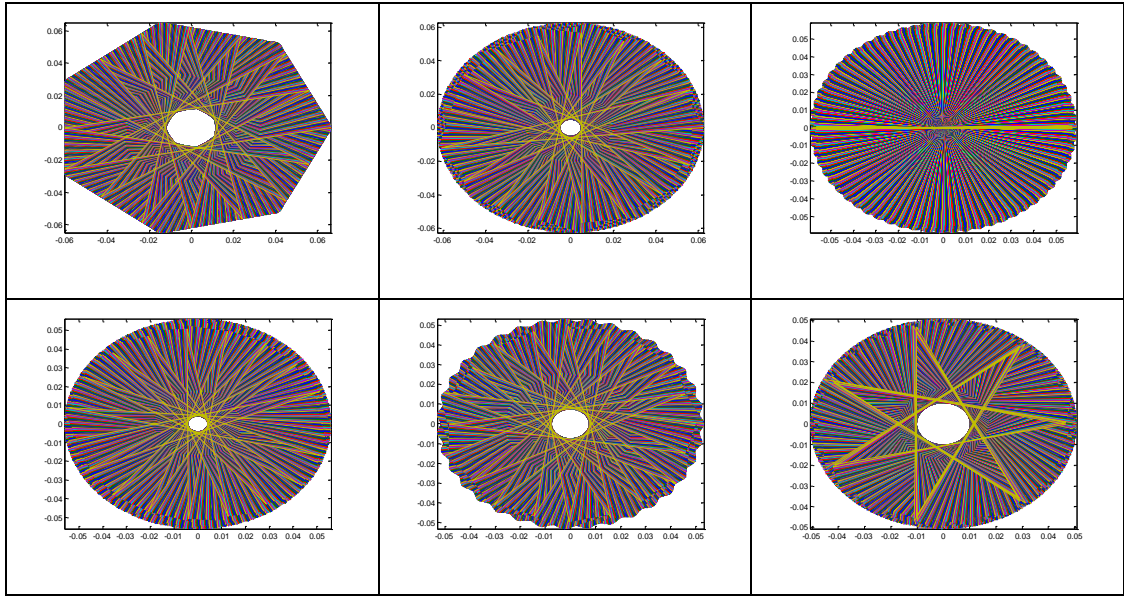


Figure 83 Row drawing of $u_{(7,6)}$