NUMERICAL COMPUTATIONS FOR THE DESIGN OF ELECTRONIC MAIL BOXES ON CANTOR SET

BY

MASHO JIMA

ADVISORS: ALEMAYEHU SHIFERAW (PhD)

GENANAW GOFE (PhD)

HABTAMU GAROMA (MSc)

A THESIS SUBMITTED TO THE DEPARTMENT OF MATHEMATICS, IN THE PARTIAL FULFILLMENT FOR THE REQUIREMENTS OF THE DEGREE OF MASTERS OF SCIENCE IN MATHEMATICS

JIMMA, ETHIOPIA

JUNE, 2014

Declaration

I, undersigned declare that this research entitled with **"Numerical Computations For The Design Of Electronic Mail Boxes On Cantor Set"** is original and it has not been submitted to any institution elsewhere for the award of any academic degree or like, where other sources of information have been used, they have been acknowledged.

Name: Masho Jima

Signature: _____

Date: _____

The work has been done under the supervision of:

Alemayehu Shiferaw (PhD)

Signature: _____

Date: _____

Genanaw Gofe (PhD)

Signature: _____

Date: _____

Acknowledgement

First of all I'm indebted to my almighty God who gave me long life and helped me to reach the precious time of my research report writing completion. Next, I would like to appreciate Dr. Alemayehu Shiferaw, Dr.Genanaw Gofe and Habtamu Garoma (MSc) for their painstakingly gone through all my research from scratch to the last submission.

Furthermore, valuable thanks go to Mr. Bira Wolteji, Akalu Abriham, and Rida Tassew (MSc graduates of 2014 from Jimma University) and for their excellent idea generation and encouragement for the work.

Lastly, I pass heartfelt thanks go to my family for moral and financial support they pay for the success of this work.

Masho Jima

Table of Contents

List of Figures

Abstract

This paper extends the algorithm introduced by (Al-Rammahi, 2014)*[2] by using the Cantor sets and cubic spline interpolating function in the design of electronic mailboxes. The cantor set was used as the domain of the function for the mail design while spline functions were used as the formula. The password of the mailbox was calculated in line with that of cantor set of intervals and spline interpolating functions in respective of the governing polynomial function of degree* $N-1$*. The mathematical meaning of sending, receiving, and opening of message or mail inbox were also discussed. The software package termed as MATLAB was in a position to design and calculate the intended numerical values. Finally, the Newton-Raphson Method was used for the computation of the password and mathematically the interpretations were given.*

CHAPTER ONE

1. Introduction

1.1 Background of the Study

Electronic mail is very important at present in terms of its wide or widely accepted medium of interpersonal communication for many years. It have ample amount of applications, especially in transaction of business activities such as people's communication, military activities, academics and others. Many researchers have been working in designing, improving, securing, making connections so fast, to benefit the good usage of electronic mail. For instance, (Cheung, 2011) [6] addressed the impact of electronic direct mail on the design of the messages using Chi square distribution. (Bothma, 2008)[14] Studied ways of combating the corporate paper war: Electronic Mail Abuse paper war and electronic mail abuse. As cited on [2], (Bahreman, 1994)[3] Proposed two families of protocols to certify electronic mail with enabling to exchange a receipt extracting the ideas from [3] and [6]. (Hui, 1993)[5] Examined the research and developed a prototype object-based multimedia electronic mail system based on the ideas taken from [3] and [14].

Moreover, (Al-Rammahi, 2014)[2] came with the paper that concerns for designing new proposed algorithm for the design of electronic mail. Different mail servers have different mechanisms to control the customers mailing activities by designing their own system controlling algorithms. Hence, three ideas were composed in the design, Cantor sets, spline, and Newton –Raphson's method. As indicated on [2], Cantor sets have good topological properties represented in bounded, closure, compactness, measurable, infiniteness, and countable. So, it was used as the area (or domain) of the design. For the smoothness of numerical spline method (Lipschutz, 1965) [13] and (Zhou, 1996)[17], it is used as a functioning or controlling the design. For fast time and less error, Newton –Raphson's method were used for the computation of the approximated roots of the governing interpolating polynomial function which was derived from the cubic spline interpolating functions. The user name was served as initial point while the roots were used as the password. Two procedures were introduced, the one, named as (send message) used for putting the message in box mail while the second, named as (open mail) used for owner mail (Blundo, 2004) [9] and (Cyders, 2014)[7].

When a function f defined on interval $[x_o, x_N]$ and a set of nodes $\{x_o, x_1, \cdots, x_N\}$ such that $a = x_o, x_1, \cdots, x_N = b$.

A cubic spline interpolating S for f is a function that satisfies the following conditions (Al-Rammahi, 2014)[2]:

1. $S(x)$ is a cubic polynomial, denoted $S_i(x)$ on subinterval $[x_i, x_{i+1}]$ for each $j = 0,1,..N-1.$
2. $S(x_i) = f(x_{i+1})$ for each $i = 0,1,2\cdots, N$
3. $S_{i+1}(x_{i+1}) = S_i(x_{i+1})$ for each $i = 0,1,2,\cdots N-2.$
4. $S'_{i+1}(x_{i+1}) = S'_i(x_{i+1})$ for each $i = 0,1,2,\cdots N-2.$
5. $S''_{i+1}(x_{i+1}) = S''_i(x_{i+1})$ for each $i = 0,1,2,\cdots N-2.$
6. One of the following set of boundary conditions is satisfied

   $S''(x_o) = 0 = S''(x_N)$ For free or natural boundary and

   $S'(x_o) = f'(x_o),\ and\ S'(x_N) = f'(x_N)$ For coupled boundary.

Remark: To construct the cubic spline interpolating S for the function f which defined on the values [9]. Let $a = x_o < x_1 < \cdots < x_N = b$

Satisfying $S''(x_o) = S''(x_N)$

$and\ S(x) = S_i(x) = a_i + b_i(x - x_i) + c_i(x - x_i)^2 + d_i(x - x_i)^3\ for\ x_i \le x \le x_{i+1}$:

Picard's Theorem: If $f(x, y)$ and $\frac{\partial f}{\partial y}$ are both continuous functions on a closed rectangle R, then through each point (x_o, y_o) in the interior of R, then there exists a unique curve of the equation $\frac{dy}{dx} = f(x, y)$ that passes through it.

1.2. Statement of the Problem

(Al-Rammahi, 2014)[2] came with the new algorithm on Cantor interpolating Spline to design Electronic mail boxes. Even though the result obtained was very promising it was limited $[0,1]$ which is C_o of the Cantor set C. The design invited a single mail address having only two data set regardless of the server used. Thus the design was case study and not set for different servers on different intervals of Cantor set. So, this paper tried to extend the idea of (Al-Rammahi, 2014)[2]from the interval C_o to $C_{1,1}$ by relating Newton Raphson-Method and the spline function used for the design of mailbox on a defined domain. Moreover, this paper has tried to address the following basic questions.

1. How can we apply spline function, Cantor set and their properties in the design of Mailbox?
2. How can we formulate an interpolating polynomial function for the design of mailboxes?
3. How can we associate the roots to the password for domain name in mailbox system?
4. What are the mathematical meanings of sending, receiving, and opening of mail inbox?

1.3. Objectives of the Study

1.3.1 General Objective

The main objective of this study is to deal with the numerical computations for the design of electronic mail boxes on Cantor set

1.3.2 Specific Objectives

The research was subjected to the following specific objectives:

1. To apply spline function, Cantor set and their properties in the design of Mailbox.
2. To formulate an interpolating polynomial function for the design of mailboxes.
3. To associate the roots to the password for domain name in mailbox system.
4. To discuss real mathematical meanings of sending, receiving, and opening of mail inbox.

1.4 Significance of the Study

Numerical computations for the design of electronic mail boxes on Cantor set is an interesting area of research with abundant real applications. There are many works about the design of electronic mail boxes in line with the application of numerical analysis. The researcher hopes that the result obtained in this study will contribute to research activities in this area. The researcher was also beneficial from this study since it was a good exposure to develop scientific research writing skill and scientific communication in Mathematics. Moreover, this study will lay a base for other researchers who need to conduct a research on this area.

1.5 Delimitation of the Study

This study was delimited to the numerical computations for the design of electronic mail boxes on Cantor set using Gmail and Yahoo servers. The study has been done under numerical analysis stream of mathematics department in 2014 at Jimma University.

CHAPTER TWO

2. Review of Related Literatures

2.1 The Cantor Set

Recent multimedia research (Fairs, 1984)[10] and (Micali, 1997)[15], efforts have resulted in the slow emergence of multimedia mail systems as a viable enhancement and replacement of the traditional ASCII-text only mail systems. Existing multimedia systems, which are mainly homogeneous systems, are generally more concerned with providing the necessary tools to allow multimedia mail to be composed and electronic communication to take place among users (Zhou, 1996)[16]. Provision of basic functionality alone is insufficient to assure long term success and large scale users' acceptance. There is a distinctive need to address both aspects of efficiency and effectiveness in multimedia electronic mail systems so that it can be a viable alternative to traditional text mail. The Cantor set has many definitions and many different constructions (Pugh, 2002)[13] and (Aliprantis, 1981) [1]. Although Cantor originally provided a purely abstract definition, the most accessible is the Cantor middle-thirds or ternary set construction. Begin with the closed real interval $[0,1]$ and divide it into three equal open subintervals. Remove the central open interval $I_1 = \left(\frac{1}{3}, \frac{2}{3}\right)$ such that $[0,1] - I_1 = [0,1] - \left(\frac{1}{3}, \frac{2}{3}\right)$

Next, subdivide each of these two remaining intervals into three equal open subintervals and from each remove the central third. Let I_2 be the removed set, then

$$I_2 = \left(\frac{1}{3^2}, \frac{2}{3^2}\right) \cup \left(\frac{7}{3^2}, \frac{8}{3^2}\right) \text{And} [0,1] - (I_1 \cup I_2) = \left[0, \frac{1}{3^2}\right] \cup \left[\frac{2}{3^2}, \frac{2}{3^2}\right] \cup \left[\frac{6}{3^2}, \frac{7}{3^2}\right] \cup \left[\frac{8}{3^2}, 1\right]$$

We can then subdivide each of the intervals that comprise $[0,1] - (I_1 \cup I_2)$ in to three subintervals, removing their middle thirds, and continue in the previous manner. The sequence of open sets I_n is then disjoint, and we traditionally define the Cantor set C as the closed interval with the union of these I_n subtracted out. That is, $C = [0,1] - \cup I_n$

And moreover, the Cantor set C is perfect and totally disconnected, nonempty, closed and nowhere dense, and uncountable. Although our construction of the Cantor set in the first section

used the typical middle-thirds or ternary rule, we can easily generalize this one-dimensional idea to any length other than $\frac{1}{3}$, excluding of course the degenerate cases of 0 and 1 [16].

2.2 The Newton-Raphson Method

The studies done by (Al-Rammahi, 2014)[2] and (Cyders, 2014)[7] shows as that the Newton-Raphson Method (or simply Newton's Method) is another so-called local method to determine the root of an equation or function. This method uses a single starting point (as opposed to the bounds required by the bisection method), and repeatedly uses a derivative to project a line to the axis of the root in question.

2.3 The Electronic Mail Design

As it was cited on (Fairs, 1984) [10], the Electronic Mail Designer focuses on two terms, the (ID) and the password. It is clear the (ID) is public while the password should be top secret. So, suitable mathematics must be used carefully for issue sub-domain for each mail which is not related with other mails and it is not possible to insert other domain in the sub-domains series.

CHAPTER THREE

3. Research Design and Methodology

3.1. Study Site

This study was conducted at Department of Mathematics, Jimma University, College of Natural Science, in 2014. In particular, the study was done under Numerical Analysis stream. That is numerical computations for the design of electronic mail boxes on Cantor set.

3.2. Study Design

The study design was Quasi Experimental which is both experimental and documentary analysis. Because, the related documents were compiled, analyzed, and their numerical values were manipulated by using MATLAB software package.

3.3. Sources of Information

This study mostly depends on secondary data so, the available sources of information for the study were books, journals, different studies related to the topic, internet services, and the computational software called MATLAB.

3.4 Procedures of the study

The procedure the researcher used was the procedure used by (Al-Rammahi, 2014)[2] and (Blundo, 2004) [4] in their papers. In addition to these the following procedures were employed:

Step 1: The interval of a Cantor set was selected.

Step 2: The domain name of one user was selected.

Step 3: By using MATLAB the domain name was converted into its data set points.

Step 4: The cubic interpolating Spline function with their respective conditions were defined.

Step 5: The diagonal coefficient matrix were obtained.

Step 6: The coefficients were evaluated on their proper interval.

Step 7: Interpolating polynomial of degree $N-1$ was obtained.

Step 8: Using Newton-Raphson method the iteration was done.

Step 9: The system's password was obtained for the chosen domain name.

3.5 Ethical Consideration

The study needs books, journals, different studies related to the topic, internet services, MATLAB, and other related materials. But there were a problem for collecting all the above listed required materials without any legal letter. So, the researcher took a letter of permission from mathematics department of Jimma University and explained the aim of collecting those materials. And finally, the researcher appreciated the respondent or venders for the consent and cooperation they made.

CHAPTER FOUR

4. Result and Discussion

4.1. Preliminaries

A polynomial spline of degree m is a function $S(x)$ for $a = x_o < x_1 < \cdots < x_{N-1} < x_N = b$ which satisfies the following conditions:

1. $For\ x \in [x_i, x_{i+1}],\ S(x) = S_i(x)$: polynomial of degree $\leq m$
2. $S^{(m-1)}$ exists and continuous at the interior points $x_1, x_2, \cdots, x_N$ i.e $\lim_{x \to x_i^-} S_{i-1}^{(m-1)}(x) = \lim_{x \to x_i^+} S_i^{(m-1)}(x)$

Definition:- A cubic spline $S(x)$ is a piecewise defined function that satisfies the following conditions:

1. $S(x) = S_i(x)$ is a cubic polynomial on each sub interval $[x_i, x_{i+1}]$ for $i = 0,1,\cdots, N-1$
2. $S(x_i) = u_i\ for\ i = 0,1,...,N$ (S have to interpolate all the points)
3. $S(x),\ S'(x)\ ,and\ S''(x)$ are continuous on $[a,b]$ (S is smooth). So, we write the m cubic polynomial pieces as $S_i(x) = a_i + b_i(x - x_i) + c_i(x - x_i)^2 + d_i(x - x_i)^3$, $i = 0,1,...,N$ -1 where $a_i, b_i, c_i\ and\ d_i$ represents 4'N unknown coefficients.

From cubic polynomial pieces between each data points we have:

$$S_i(x) = a_i + b_i(x - x_i) + c_i(x - x_i)^2 + d_i(x - x_i)^3, i = 0,1,2\ldots, N-1 \quad (1)$$

$$S'(x) = b_i + 2c_i(x - x_i) + 3d_i(x - x_i)^2 \quad (2)$$

$$S''(x) = 2c_i + 6d_i(x - x_i) \quad (3)$$

Let $S(x) = u_i\ for\ i = 1,2,3,\ldots, N-1$ since $x_i \in [x_i, x_{i+1}]$ (4)

$$S(x_i) = S_i(x_i)$$

$$u_i = S_i(x_i)$$

$$u_i = a_i + b_i(x_i - x_i) + c_i(x_i - x_i)^2 + d_i(x_i - x_i)^3$$

$$u_i = a_i \text{ for each } i = 1, 2, 3, \ldots, N-1 \tag{5}$$

From continuous properties of cubic spline method across each interval we have

$$S(x_i) = S_i(x_i)$$

$$S_i(x_i) = S_{i-1}(x_i) \text{ for } i = 1, 2, 3, \cdots, N-1 \tag{6}$$

From (5) we have $S_i(x_i) = a_i$ and

$$S_{i-1}(x_i) = a_{i-1} + b_{i-1}(x_i - x_{i-1}) + c_{i-1}(x_i - x_{i-1})^2 + d_{i-1}(x_i - x_{i-1})^3$$

So, $a_i = a_{i-1} + b_{i-1}(x_i - x_{i-1}) + c_{i-1}(x_i - x_{i-1})^2 + d_{i-1}(x_i - x_{i-1})^3$

For $i = 2, 3, ..., N$ -1 and $h = x_i$ - x_{i-1}

$$a_i = a_{i-1} + b_{i-1}h + c_{i-1}h^2 + d_{i-1}h^3 \tag{7}$$

To make a curve smooth across each interval, the derivative must be equal at the data points.

$$i.e, S_i'(x_i) = S_{i-1}'(x_i)$$

$$\Rightarrow S_i'(x_i) = b_i \text{ and} \tag{8}$$

$$S'_{i-1}(x_i) = b_{i-1} + 2c_{i-1}(x_i - x_{i-1}) + 3d_{i-1}(x_i - x_{i-1})^2$$

$$b_i = b_{i-1} + 2c_{i-1}(x_i - x_{i-1}) + 3d_{i-1}(x_i - x_{i-1})^2 \tag{9}$$

From equation (3) $S_i''(x) = 6d_i(x - x_i) + 2c_i$

$$S_i'(x_i) = 2c_i \text{ for } i = 2, 3, 4, \cdots, N-2 \tag{10}$$

Lastly, since $S_i''(x)$ has to be continuous across the interval,

$$S''_i(x_{i+1}) = 6d_i(x_{i+1} - x_i) + 2c_i \tag{11}$$

And letting $h = x_{i+1} - x_i$, using the conclusion from equation (10) and (11):

$$S''_{i+1}(x_{i+1}) = 6d_i(x_{i+1} - x_i) + 2c_i$$

$$2c_{i+1} = 6d_i h + 2c_i \tag{12}$$

The equation can be much simplified by substituting M_i for $S_i''(x_i)$ and expressing the above equation in terms of M_i and u_i. This makes the determination the weights a_i, b_i, c_i and d_i a much easier task. Each c_i can be represented by:

$$S_i''(x_i) = 2c_i \Rightarrow M_i = 2c_i \Rightarrow c_i = \frac{M_i}{2} \tag{13}$$

And a_i has already been determined to be $a_i = u_i$

Similarly, using equation (12) d_i can be written as:

$$2c_{i+1} = 6d_i h + 2c_i \quad \Rightarrow 6d_i h = 2c_{i+1} - 2c_i$$

$$\Rightarrow d_i = \frac{2c_{i+1} - 2c_i}{6h} = \frac{2(\frac{M_{i+1}}{2}) - 2(\frac{M_i}{2})}{6h}$$

$$d_i = \frac{M_{i+1} - M_i}{6h} \tag{14}$$

From equation (7) b_i can be written as:

$$a_{i+1} = a_i + b_i h + c_i h^2 + d_i h^3 \Rightarrow b_i h = -a_i - c_i h^2 - d_i h^3 + a_{i+1}$$

$$\Rightarrow b_i = \frac{-a_i - c_i h^2 - d_i h^3 + a_{i+1}}{h}$$

$$b_i = \frac{u_{i+1} - u_i}{h} - \frac{h}{6}\left(M_{i+1} + 2M_i\right) \tag{15}$$

We now have our equation for determining the weight of our $N-1$ equations:

$$a_i = u_i \ , \ b_i = \frac{u_{i+1} - u_i}{h} - \frac{h}{6}(M_{i+1} + 2M_i) \ , \ c_i = \frac{M_i}{2}, d_i = \frac{M_{i+1} - M_i}{6h} \tag{16}$$

These systems can be handled more conveniently by putting them in Matrix form as follows

From (9), $b_{i+1} = b_i + 2c_i h + 3d_i h^2 \ \ for \ i = 1, 2, ..., \ N\text{-}1$

$$\Rightarrow 3d_i h^2 + 2c_i h = b_{i+1} - b_i \tag{17}$$

When we substitute the values of equation (16) into (17) and rearrange the values; we get:

$$M_i + 4M_{i+1} + M_{i+2} = \frac{6}{h^2}\left[u_i - 2u_{i+1} + u_{i+2}\right] \text{ for } i = 1, 2, 3, \cdots, N-1 \tag{18}$$

By substituting the values of i in to i-1, we get:

$$M_{i-1} + 4M_i + M_{i+1} = \frac{6}{h^2}\left[u_{i-1} - 2u_i + u_{i+1}\right], \text{ for } i = 1, 2, 3, \cdots, N-1 \tag{19}$$

4.2. Modeling Electronic Mail

In order to provide some form of protection, cryptographic techniques have been employed to obtain additional guarantees on the mail service. A number of certified email protocols have been presented in literatures, ensuring that the message exchange procedure provides the participants with different security properties. Usually such protocols involve a trusted third party (TTP for brief) which controls the behavior of the participants, helping them in the message exchange, and resolving any dispute if necessary. According to the role played by the TTP protocols have been classified as inline or optimistic. In inline protocols [16], the TTP is actively involved in each message exchange. In optimistic protocols [15], the sender and the receiver perform the message exchanging without the intervention of the TTP but they can invoke the TTP to resolve any dispute, caused for example by cheating attempt from one of the party.

We consider a distributed system consisting of n nodes (processors) $P = \{1, 2, ..., n\}$ and special node, namely the Trusted Third Party (TTP) which is delegated by the participants to control the behavior of the parties, assist them during the exchange of messages and resolve any dispute if necessary. The TTP is a fully trusted party, meaning that the senders and receivers have complete trust in it. Moreover, there is a communication channel between each node of the set $P \cup \{TTP\}$ [4].

The cryptographic primitives used in this paper are intensively selected. Such as $Sig_A(m)$, $h(m)$, $PK_B(m)$, and $E_k(m)$.

Notations 4.2.1[4]:

(i) $Sig_A(m)$ denoted the digital signature of the message m using the private key of user A under a public-key signature algorithm.

(ii) $h(m)$ Indicates the hash of message m using some collusion resistant hashing scheme. A collusion resistant hash function maps arbitrary length messages to constant size message such that it is computationally infeasible to find any two distinct messages hashing to the same value.

(iii) $PK_B(m)$: denotes the encryption of message m using the public key of user B in some public key encryption algorithm. The algorithm should provide non malleability, i.e., given a cipher text it is impossible to generate another cipher text such that the respective plaintexts are related.

(iv) $E_k(m)$: denotes the encryption of message m using the key k under some symmetric encryption algorithm.

To send a mail message containing m to the receiver R, the sender S first digitally signs (S,R,ttp,m) with his private key to produce $Sig_s(S,R,ttp,m)$. Then, S generates a session key k and encrypts the signed data under k using a symmetric key cryptosystem. Finally, S computes $h(m)$ and send the message

$M_1 = \langle S,R,ttp,\mathrm{h}(m), PK_{ttp}(k), E_k(Sig_s(S,R,ttp,m))\rangle$ to R. The clear text part

$(i.e.,\ S,R,ttp,h(m))$ In this message will serves as the mail identifier. This message informs R that there is a certified mail from S to him. After receiving this message, has two choices. He may ignore the message. In this case, the protocol is aborted. He may choose to receive the message. In this case, he signs $(S,R,ttp,h(m))$ using his private key and sends the message $M_2 = \langle Sig_R(S,R,ttp,h(m)), PK_{ttp}(k), E_k(Sig_s(S,R,ttp,m))\rangle$ *to TTP*. Upon receiving this message, the TTP first check the validity of $Sig_R(S,R,ttp,m)$ using public key of R. Then it decrypts $PK_{ttp}(k)$ using its private key, and decrypts $E_k(Sig_s(S,R,ttp,m))$ using k. Next the TTP checks the validity of $Sig_s(S,R,ttp,m)$ using $S's$ public key, computes $h(m)$, and compares this $h(m)$ with the one received in $Sig_R(S,R,ttp,h(m))$. If the two value match, the TTP knows that m is the mail content that S wanted to send to R, and that R is willing to receive m. In this case, the TTP is able to compute the messages $M_3 = \langle Sig_{ttp}(Sig_R(S,R,ttp,h(m))),R,m\rangle$ corresponding to the proof of origin and $M_4 = \langle Sig_{ttp}(Sig_s(S,R,ttp,m))\rangle$ corresponding to the proof of delivery and sends them to R and to S, respectively.

Non-repudiation of origin [4]: The protocol provides the recipient of an email with an irrefutable proof that the mail content received was the same as the one sent by the originator.

This proof-of-origin can protect against any attempt by the originator to falsely deny sending that message.

Non-repudiation of delivery [4]: The protocol provides the mail originator with an irrevocable proof that the mail content received by the recipient was the same as the one sent by the originator. This proof-of-delivery can protect against any attempt by the recipient to falsely deny receiving the message.

Fairness [4]: Proper execution of the protocol ensures that the proof-of-delivery from the mail recipient and the proof-of-origin from the mail originator are available to the mail originator and recipient, respectively. Moreover, the protocol must be fail-safe. That is, incomplete execution of the protocol will not result in a situation where the proof-of-delivery is available to the originator but the proof-of origin is not available to the recipient, or vice versa.

As it is sited on [3], the first invariant shows that if $\text{StatusSnd}(\text{id}) = done$ the message M_4 of the protocol has been delivered to the sender.

4.1 [4]: In any reachable state s,

if $\text{s.StatusSnd}(\text{id}) = \text{done}$ the $\text{s.CHANNEL}_{ttp,s}.\text{HChanRec}(\text{id}) = \text{yes}$. If i receives a message from the TTP the message has been sent by the TTP.

4.2 [4]: In any reachable state s, $if\ \ s.CHANNEL_{ttp,i}.HChan\,\text{Re}\,c_{ttp,i}(id) = yes$

then $s.HChanSnd(id) = yes.$

4.3 [4]: In any reachable state s, if $\text{s.CHANNEL}_{\text{ttp,s}}.\text{HChanSnd(id)=yes}$. Then $s.StatusTtp(id) = done.$

Remark: If the receiver has completed the protocol, it received the message M_3.

4.4 [4]: In any reachable state s,

if $\text{s.StatusRcv(id)=done}$ then we have $\text{s.CHANNEL}_{ttp,R}.HChan\,\text{Re}\,c(id) = yes.$

Remark: If the message is in transit on the channel from the TTP to the receiver, the message was sent by the TTP.

4.5 [4]: In any reachable state s, if $s.CHANNEL_{ttp,R}.HChanSnd(id) = yes$ then $s.HTtpTo\,\mathrm{Re}c(id) = yes.$

4.6 [4]: In any reachable state s, if $s.StatusTtp(id) = done$ then, we have $s.HTtpToRcv(id) = yes.$

4.7 [4]: In any reachable state s, if $s.StatusTtp(id) \in \{Send - rcv, Send - snd, done\}$ then $s.Hcheck(id) = yes.$

Remark: Once the TTP has sent message M_3 to the receiver, in order to complete the protocol it only needs to send message M_4 to the sender.

4.8 [4]: In any reachable state s, if s.HTtpToRcv(id)=yes then we have

$s.StatusTtp(id) \in \{Send - snd, done\}.$

Definition 4.2.1 [1]: The Cantor set C is defined as $C = \cap_{n=1}^{\infty} I_n$ where I_{n+1} is constructed by trisecting I_n and removing the middle third, I_0 being the closed real interval $[0,1]$.

Several interesting properties of the Cantor set are immediately apparent. Since it is defined as the set of points not excluded, the "size" of the set can be thought of as the proportion of the interval $[0,1]$ removed. If we add up the contribution from $\frac{2}{3}$ removed n times we find that $\sum_{n=0}^{\infty} \frac{2^n}{3^{n+1}} = \frac{1}{3} + \frac{2}{9} + \frac{4}{27} + \cdots = \frac{1}{3}\left(\frac{1}{1-\frac{2}{3}}\right) = 1$ where the geometric sum has its well-known solutions. As a result, the proportion remaining "in" the Cantor set is 1-1=0, and it can contain no intervals of non-zero length. For assume by contradiction it does contain some interval (a,b). Choose $n \in \mathbb{N}$ such that $\frac{1}{3^n} < b - a$. Since the Cantor set is contained in the finite intersection of closed intervals, all of length less than $(b-a)$, we have that this intersection and so C cannot contain (a,b).

Theorem 4.2.1 [17]: The Cantor set is nonempty.

Proof: Let consider the interval I_n as defined on the above Cantor set definition. Each trisection of I_n to form I_{n+1} leaves exactly two end points. For example removing $\left(\frac{1}{3},\frac{2}{3}\right)$ from $[0,1]$ leaves the points $p_0 = \frac{1}{3}$ *and* $p_1 = \frac{2}{3}$. In fact, since the Cantor set is the infinite intersection of each I_n, C contains the end points of each subinterval, and is clearly non empty. In fact it is infinite.

Definition 4.2.2 [17]: A subset A of a metric space M, is nowhere dense if its closure has an empty interior. That is if $\text{int}(\overline{A}) = \varnothing$.

Theorem 4.2.2 [17]: A Cantor set is closed and nowhere dense.

Proof: We have already seen that C is the intersection of closed sets, which implies that C is itself closed. Furthermore, as previously discussed the Cantor set contains no intervals of non-zero length, and so, $\text{int}(\overline{C}) = \varnothing$.

Definition 4.2.3 [13]: A metric space M is totally disconnected if, for any $\varepsilon > 0$ and $p \in M$ there exists a clopen subset U of M such that $p \in U \subset M_\varepsilon(p)$. That is, there is an arbitrarily small clopen neighborhood centered on every point of M. With this definition we can prove two more important facts about the Cantor set.

Theorem 4.2.3 [13]: The Cantor set C is perfect and totally disconnected.

Proof: Fix any $\varepsilon > 0$ and point $p \in C$. Let $n \in \mathbb{N}$ be sufficiently large such that $\frac{1}{3^n} < \varepsilon$. Then, p is guaranteed to be in one of the intervals $(I_n$ for some $n \in \mathbb{N})$ that make up C, each of length $\frac{1}{3^n}$, the endpoints of the Cantor set in this interval are infinite number, and contained in the open interval $(p-\varepsilon, p+\varepsilon)$, so p is a cluster point of $C, M_\varepsilon(p)$ containing an infinite number of points. And since we are considering any $p \in C$, C is perfect. Furthermore, this interval I_n is closed in $\mathbb{R}$ and in the Cantor set C as well. Since $I_n^c = C \setminus I_n$ consists of a countable number of closed intervals, itself closed. We can then represent C as the disjoint union of two clopen sets, $(C \cap I_n)$ and $(C \cap I_n^c)$, the result being that the Cantor set C is totally disconnected.

Theorem 4.2.4 [17]: Cantor set C is compact.

Proof: Each C_n is a finite union of closed sets, so C_n is closed for $\forall n$. Then, $C = \cap C_n$ is also closed. Also, C is bounded since $C \subseteq [0,1]$. So, by Heine-Borel theorem C is compact.

Generalization, although our construction of the Cantor set used the typical "middle third" or ternary rule, we can easily generalize this one dimensional idea to any length other than $\frac{1}{3}$, excluding of course the degenerate cases of 0 and 1.

4.3 Main Results

In this section we are going to treat and compute the actual system based password of gmail domain name. With the wide spreading of the internet and the World Wide Web, our society is becoming more and more dependent on communication data which are transmitted over computer networks. A large number of transactions involving a growing number of people have been actually replaced by their digital analogues, in which electronic "objects" are exchanged among two or more parties. An example comes from the diffusion of the electronic mail services which allows users to exchange messages containing text or multimedia files.

Because of its features, such as low cost, rapidity and accessibility the email service is increasingly used in place of ordinary mail. In many cases, email messages are recognized as recipients evidences of online transactions, such as buying air lines tickets, or submission of papers in conferences or journals, and so on. However, the use of email poses some problems, since in its simplest form the email service does not have many features that are usually required in such cases. The standard email service is based on Simple Mail Transfer Protocol [5] and Post Office Protocol [10], which do not offer guarantees on the delivery and the integrity of the messages. Messages are usually stored and transmitted in plain text allowing a malicious adversary to tap the connection during the transfer and making him able to access sensible data.

Now a day's most of the mail users are in a position to use mail address from the prominent servers such as Gmail, Yahoo, Hotmail, and the like. Gmail sever has 6518MB of storage, which allows the users the ability to save their own email without worrying that any new emails will not get through because they will reach the allowed limit.

Now for just comparison, the researcher used mail address or domain names of users of Gmail and Yahoo server. The reason why these two servers were selected was that most of the users in the world (about 98% (Micali, 1997)[14]) were using these two servers. So, the general graphic representations of these two were sketched through MATLAB by using cubic interpolating spline function which was defined on Cantor set.

For the analysis, the domain name 'maashookoo@gmail.com' was taken and by using MATLAB built in package which corresponds each character (twenty of them) of the domain into numbers (twenty corresponding numbers), (i.e 'm' is represented by '109', 'a' by '97', etc), which can be written as a vector

$[109 \;\; 97 \;\; 97 \; 115 \; 104 \; 111 \; 111 \; 107 \; 111 \; 111 \;\; 64 \; 103 \; 109 \;\; 97 \; 105 \; 108 \;\; 46 \;\; 99 \; 111 \; 109]$.Hence, the domain name now is transformed, and the analysis has been done by considering this vector as shown below. Moreover, the MATLAB code was attached as an appendix at the end.

Example1. Design the cubic spline of domain name maashookoo@gmail.com

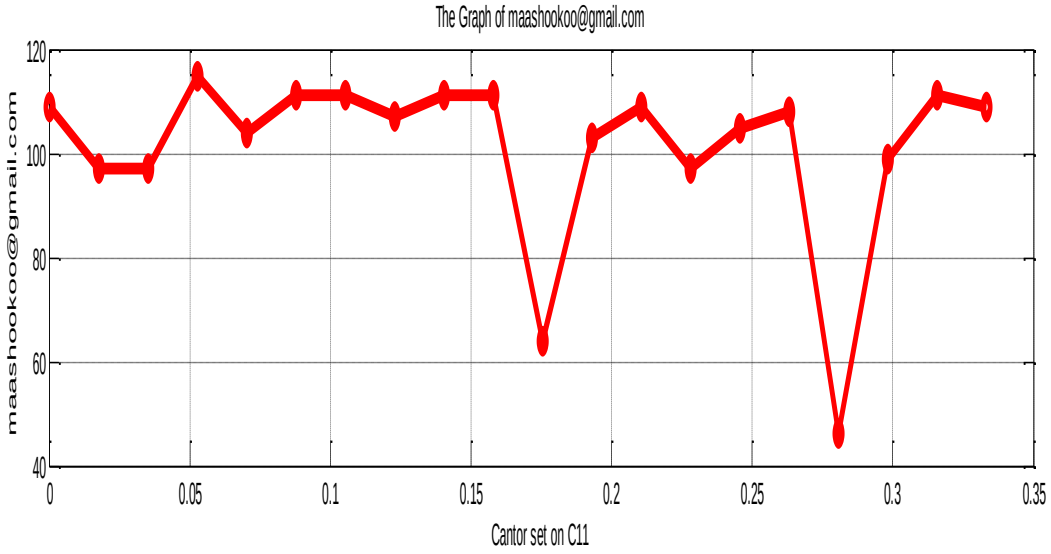

Figure 1

Approximated equation for the above mail address is

$y_j = a * e^{-bx_j} + c$; where a ,b, c are constants to be calculated and y_j's are evaluated at each mesh poins for finite elements

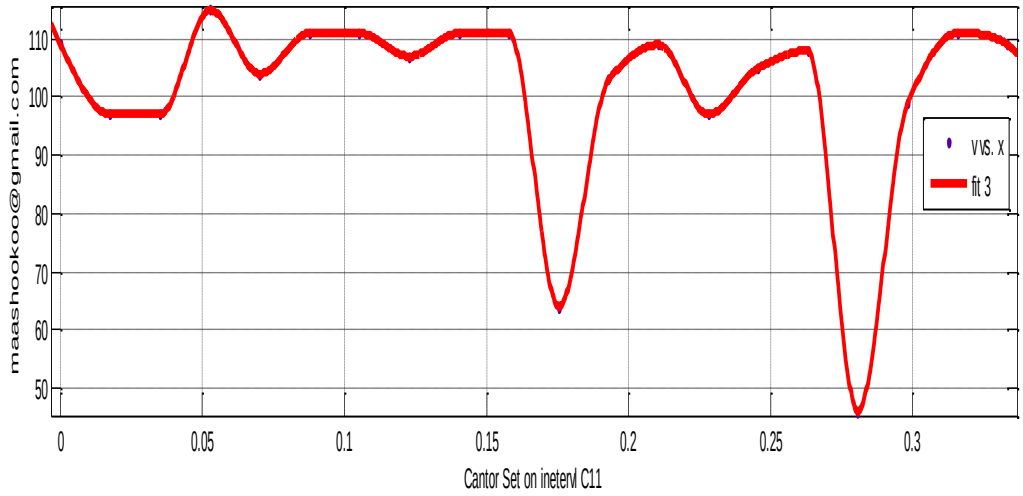

Figure 2

$y_j = a * e^{-bx_j} + c$; where a ,b, c are constants to be calculated and y_j's are evaluated at each mesh poins for finite elements

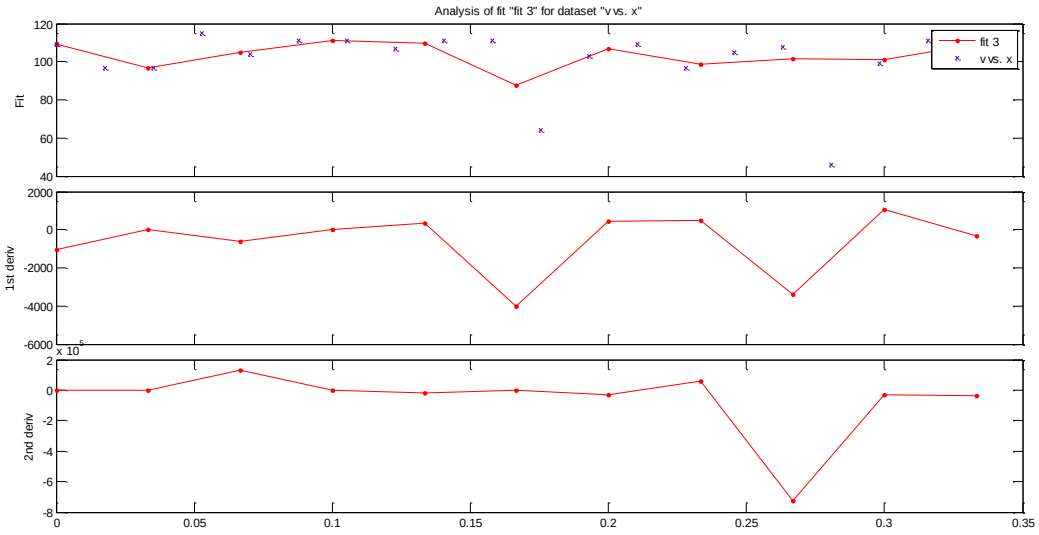

Figure 3

From equation (19) and equations (1-18) the coefficients of the spline interpolating function was calculated by reducing the expressions into tri-diagonal. That is

$$M_{i-1} + 4M_i + M_{i+1} = \frac{6}{h^2}[u_{i-1} - 2u_i + u_{i+1}] \quad , \; for \; i = 1, 2, ..., N\text{-}1.$$

For $i = 1$ we have the following expressions;

$$M_o + 4M_1 + M_2 = \frac{6}{h^2}[u_o - 2u_1 + u_2]$$

In a similar fashion one can obtain a $20X20$ tri-diagonal matrix for the left hand expression and a column matrix say $t_{i'}s$ for the right hand side expression as follows. Let A be a coefficient matrix for the left hand side expression.

Where $t_i = \frac{6}{h^2}\left(u_{i-1} - 2u_i + u_{i+1}\right)$ for i=1,2,$\cdots$18 (20)

and $u_i's$ were calculated directly from the domain name maashookoo@gmail.com which was

transformed into vector representation format by using MATLAB code attached on the appendix.

$$U = u_i = \left[u_o, u_1, \cdots, u_{17}\right]^T$$

$$= \left[109\ \ 97\ \ 97\ 115\ 104\ 111\ 111\ 107\ 111\ 111\ \ 64\ 103\ 109\ \ 97\ 105\ 108\ \ 46\ \ 99\ 111\ 109\right]^T \qquad (21)$$

For $i = 0,1,\cdots 17$

$$\begin{bmatrix}
1 & 4 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 4 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 4 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 4 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 4 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 4 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 4 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 4 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 4 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 4 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 4 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 4 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 4 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 4 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 4 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 4 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 4 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1
\end{bmatrix} * \begin{bmatrix} M_0 \\ M_1 \\ M_2 \\ M_3 \\ M_4 \\ M_5 \\ M_6 \\ M_7 \\ M_8 \\ M_9 \\ M_{10} \\ M_{11} \\ M_{12} \\ M_{13} \\ M_{14} \\ M_{15} \\ M_{16} \\ M_{17} \end{bmatrix} = \begin{bmatrix} t_1 \\ t_2 \\ t_3 \\ t_4 \\ t_5 \\ t_6 \\ t_7 \\ t_8 \\ t_9 \\ t_{10} \\ t_{11} \\ t_{12} \\ t_{13} \\ t_{14} \\ t_{15} \\ t_{16} \\ t_{17} \\ t_{18} \end{bmatrix} \qquad (22)$$

From equation (20) t_i's for i=1,2,$\cdots$18, were calculated by using MATLAB and the result were displayed as follows in matrix form

t_i=[2.5941e+005 3.8911e+005 -6.2690e+005 3.8911e+005 -1.5132e+005 -8.6469e+004 1.7294e+005 -8.6469e+004 -1.0160e+006 1.8591e+006 -7.1337e+005 -3.8911e+005 4.3235e+005 -1.0809e+005 -1.4051e+006 2.4860e+006 -8.8631e+005 -3.0264e+005]^T, where i=1,2,$\cdots$18.

From equation (22) M_i's were computed by taking the following expression.

$M_i = A^{-1} * t_i$, for i=1,2,$\cdots$18 So, A^{-1} is computed and the result was as follows:

$$A^{-1}=1.0e+009 * \begin{bmatrix}
0.0000 & -0.0000 & 0.0000 & -0.0000 & 0.0000 & -0.0000 & 0.0000 & -0.0000 & 0.0000 & -0.0002 & 0.0006 & -0.0021 & 0.0079 & -0.0294 & 0.1096 & -0.4089 & 1.5259 & -5.6946 \\
0 & 0.0000 & -0.0000 & 0.0000 & -0.0000 & 0.0000 & -0.0000 & 0.0000 & -0.0000 & 0.0000 & -0.0002 & 0.0006 & -0.0021 & 0.0079 & -0.0294 & 0.1096 & -0.4089 & 1.5259 \\
0 & 0 & 0.0000 & -0.0000 & 0.0000 & -0.0000 & 0.0000 & -0.0000 & 0.0000 & -0.0000 & 0.0000 & -0.0002 & 0.0006 & -0.0021 & 0.0079 & -0.0294 & 0.1096 & -0.4089 \\
0 & 0 & 0 & 0.0000 & -0.0000 & 0.0000 & -0.0000 & 0.0000 & -0.0000 & 0.0000 & -0.0000 & 0.0000 & -0.0002 & 0.0006 & -0.0021 & 0.0079 & -0.0294 & 0.1096 \\
0 & 0 & 0 & 0 & 0.0000 & -0.0000 & 0.0000 & -0.0000 & 0.0000 & -0.0000 & 0.0000 & -0.0000 & 0.0000 & -0.0002 & 0.0006 & -0.0021 & 0.0079 & -0.0294 \\
0 & 0 & 0 & 0 & 0 & 0.0000 & -0.0000 & 0.0000 & -0.0000 & 0.0000 & -0.0000 & 0.0000 & -0.0000 & 0.0000 & -0.0002 & 0.0006 & -0.0021 & 0.0079 \\
0 & 0 & 0 & 0 & 0 & 0 & 0.0000 & -0.0000 & 0.0000 & -0.0000 & 0.0000 & -0.0000 & 0.0000 & -0.0000 & 0.0000 & -0.0002 & 0.0006 & -0.0021 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.0000 & -0.0000 & 0.0000 & -0.0000 & 0.0000 & -0.0000 & 0.0000 & -0.0000 & 0.0000 & -0.0002 & 0.0006 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.0000 & -0.0000 & 0.0000 & -0.0000 & 0.0000 & -0.0000 & 0.0000 & -0.0000 & 0.0000 & -0.0002 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.0000 & -0.0000 & 0.0000 & -0.0000 & 0.0000 & -0.0000 & 0.0000 & -0.0000 & 0.0000 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.0000 & -0.0000 & 0.0000 & -0.0000 & 0.0000 & -0.0000 & 0.0000 & -0.0000 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.0000 & -0.0000 & 0.0000 & -0.0000 & 0.0000 & -0.0000 & 0.0000 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.0000 & -0.0000 & 0.0000 & -0.0000 & 0.0000 & -0.0000 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.0000 & -0.0000 & 0.0000 & -0.0000 & 0.0000 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.0000 & -0.0000 & 0.0000 & -0.0000 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.0000 & -0.0000 & 0.0000 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.0000 & -0.0000 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.0000
\end{bmatrix}$$

$M_i = A^{-1} * b_i,$ for i=1,2,$\cdots$18 ,

M_i= 1.0e+014 *[-7.9265 2.1239 -0.5691 0.1525 -0.0409 0.0109 -0.0029 0.0008 -0.0002 0.0001 -0.0 0.0 -0.0 0.0 -0.0 0.0 0.0 -0.0]^T

From equation (17) we have $a_i = u_i$, $b_i = \frac{u_{i+1} - u_i}{h} - \frac{h}{6}(M_{i+1} + 2M_i)$, $c_i = \frac{M_i}{2}, d_i = \frac{M_{i+1} - M_i}{6h}$

$$\Rightarrow c_i = \frac{M_i}{2} = 1.0e{+}014 * \begin{bmatrix} -3.9633 \\ 1.0620 \\ -0.2845 \\ 0.0762 \\ -0.0204 \\ 0.0055 \\ -0.0015 \\ 0.0004 \\ -0.0001 \\ 0.0000 \\ -0.0000 \\ 0.0000 \\ -0.0000 \\ 0.0000 \\ -0.0000 \\ 0.0000 \\ 0.0000 \\ -0.0000 \end{bmatrix}; d_i = \begin{bmatrix} 2.7907e{+}012 \\ -7.4776e{+}011 \\ 2.0036e{+}011 \\ -5.3701e{+}010 \\ 1.4383e{+}010 \\ -3.8318e{+}009 \\ 1.0274e{+}009 \\ -2.7767e{+}008 \\ 8.33e{+}007 \\ -2.7767e{+}007 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}; b_i = \begin{bmatrix} 3.8121e{+}012 \\ -1.0215e{+}012 \\ 2.7370e{+}011 \\ -7.3332e{+}010 \\ 1.9687e{+}010 \\ -5.2479e{+}009 \\ 1.3883e{+}009 \\ -3.8873e{+}008 \\ 83300000 \\ -5.5536e{+}007 \\ 2.3409e{+}003 \\ 360.1441 \\ -720.2881 \\ 480.1921 \\ 180.0720 \\ -3.7215e{+}003 \\ 3.1813e{+}003 \end{bmatrix} \& a_i = \begin{bmatrix} 109 \\ 97 \\ 97 \\ 115 \\ 104 \\ 111 \\ 111 \\ 107 \\ 111 \\ 111 \\ 64 \\ 103 \\ 109 \\ 97 \\ 105 \\ 108 \\ 46 \\ 99 \end{bmatrix}$$

Before we are going to use Newton-Raphson Method firs we have to determine the governing function f that agrees with S on the mesh points. Since we have 20 data sets, we need a $20X20$ matrix.

Let $f(x) = \alpha_1 + \alpha_2 x + \alpha_3 x^2 + \cdots + \alpha_{20} x^{19}$ (23)

From the property of spline function S and its governing function f we have:

$f(x_i) = S(x_i) = S_i(x_i)$, For all i=1,2,3$\cdots$,20. Consider the first interval of the Cantor set (say) $C_{11} = \left[0, \frac{1}{3}\right]$. The step size or the discretization length $h = \frac{x_N - x_1}{N} = \frac{\frac{1}{3} - 0}{20} = \frac{1}{60}$.

From equation (23) above we have

$$f(x_1) = \alpha_1 + \alpha_2(x_1) + \alpha_3(x_1)^2 + \cdots + \alpha_{20}(x_1)^{19} = 109, \text{ since } x_1 = 0,$$

$$\Rightarrow f(0) = \alpha_1 + \alpha_2(0) + \alpha_3(0)^2 + \cdots + \alpha_{20}(0)^{19} = 109,$$

$$\Rightarrow f(0) = \alpha_1 = 109$$

For x_2, $f(x_2) = \alpha_1 + \alpha_2(x_2) + \alpha_3(x_2)^2 + \cdots + \alpha_{20}(x_2)^{19}$, $x_2 = h = \frac{1}{60}$.

$$f(x_2) = f(h) = \alpha_1 + \alpha_2(h) + \alpha_3(h)^2 + \cdots + \alpha_{20}(h)^{19} = 97$$

$$f(x_3) = f(2h) = \alpha_1 + \alpha_2(2h) + \alpha_3(2h)^2 + \cdots + \alpha_{20}(2h)^{19} = 97$$
$$\vdots \qquad \vdots \qquad \vdots \qquad \vdots \qquad \vdots \qquad \vdots \qquad \vdots \qquad \vdots$$
$$f(x_{20}) = f(20h) = \alpha_1 + \alpha_2(20h) + \alpha_3(20h)^2 + \cdots + \alpha_{20}(20h)^{19} = 109$$

So, the coefficient matrix Q (say), can be expressed as;

$$Q = \begin{bmatrix} (0h) & (0h)^2 & (0h)^3 & \cdots & (0h)^{19} \\ (1h) & (1h)^2 & (1h)^3 & \cdots & (1h)^{19} \\ (2h) & (2h)^2 & (2h)^3 & \cdots & (2h)^{19} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ (20h) & (20h)^2 & (20h)^3 & \cdots & (20h)^{19} \end{bmatrix} \text{ and } \alpha_i = \begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \alpha_3 \\ \vdots \\ \alpha_{20} \end{bmatrix} \text{ for } i = 1, 2, \cdots, 20$$

Thus,

$$\begin{bmatrix} (0h) & (0h)^2 & (0h)^3 & \cdots & (0h)^{19} \\ (1h) & (1h)^2 & (1h)^3 & \cdots & (1h)^{19} \\ (2h) & (2h)^2 & (2h)^3 & \cdots & (2h)^{19} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ (20h) & (20h)^2 & (20h)^3 & \cdots & (20h)^{19} \end{bmatrix} * \begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \alpha_3 \\ \vdots \\ \alpha_{20} \end{bmatrix} = \begin{bmatrix} a_1 \\ a_2 \\ a_3 \\ \vdots \\ a_{20} \end{bmatrix}$$

$$\Rightarrow \begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \alpha_3 \\ \vdots \\ \alpha_{20} \end{bmatrix} = \begin{bmatrix} (0h) & (0h)^2 & (0h)^3 & \cdots & (0h)^{19} \\ (1h) & (1h)^2 & (1h)^3 & \cdots & (1h)^{19} \\ (2h) & (2h)^2 & (2h)^3 & \cdots & (2h)^{19} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ (20h) & (20h)^2 & (20h)^3 & \cdots & (20h)^{19} \end{bmatrix}^{-1} * \begin{bmatrix} a_1 \\ a_2 \\ a_3 \\ \vdots \\ a_{20} \end{bmatrix} \qquad (24)$$

From these expressions, we have the following values for α_i's.

$Q =$

1.6667e-002	2.7778e-004	4.6296e-006	7.7160e-008	1.2860e-009	2.1433e-011	3.5722e-013	5.9537e-015	9.9229e-017	1.6538e-018	2.7564e-020	4.5939e-022	7.6566e-024	1.2761e-025	2.1268e-027	3.5447e-029	5.9078e-031	9.8464e-033	1.6411e-034
3.3333e-002	1.1111e-003	3.7037e-005	1.2346e-006	4.1152e-008	1.3717e-009	4.5725e-011	1.5242e-012	5.0805e-014	1.6935e-015	5.6450e-017	1.8817e-018	6.2723e-020	2.0908e-021	6.9692e-023	2.3231e-024	7.7435e-026	2.5812e-027	8.6039e-029
5.0000e-002	2.5000e-003	1.2500e-004	6.2500e-006	3.1250e-007	1.5625e-008	7.8125e-010	3.9063e-011	1.9531e-012	9.7656e-014	4.8828e-015	2.4414e-016	1.2207e-017	6.1035e-019	3.0518e-020	1.5259e-021	7.6294e-023	3.8147e-024	1.9073e-025
6.6667e-002	4.4444e-003	2.9630e-004	1.9753e-005	1.3169e-006	8.7791e-008	5.8528e-009	3.9018e-010	2.6012e-011	1.7342e-012	1.1561e-013	7.7073e-015	5.1382e-016	3.4255e-017	2.2837e-018	1.5224e-019	1.0150e-020	6.7664e-022	4.5109e-023
8.3333e-002	6.9444e-003	5.7870e-004	4.8225e-005	4.0188e-006	3.3490e-007	2.7908e-008	2.3257e-009	1.9381e-010	1.6151e-011	1.3459e-012	1.1216e-013	9.3464e-015	7.7887e-016	6.4905e-017	5.4088e-018	4.5073e-019	3.7561e-020	3.1301e-021
1.0000e-001	1.0000e-002	1.0000e-003	1.0000e-004	1.0000e-005	1.0000e-006	1.0000e-007	1.0000e-008	1.0000e-009	1.0000e-010	1.0000e-011	1.0000e-012	1.0000e-013	1.0000e-014	1.0000e-015	1.0000e-016	1.0000e-017	1.0000e-018	1.0000e-019
1.1667e-001	1.3611e-002	1.5880e-003	1.8526e-004	2.1614e-005	2.5216e-006	2.9419e-007	3.4322e-008	4.0042e-009	4.6716e-010	5.4502e-011	6.3586e-012	7.4184e-013	8.6548e-014	1.0097e-014	1.1780e-015	1.3743e-016	1.6034e-017	1.8706e-018
1.3333e-001	1.7778e-002	2.3704e-003	3.1605e-004	4.2140e-005	5.6187e-006	7.4915e-007	9.9887e-008	1.3318e-008	1.7758e-009	2.3677e-010	3.1569e-011	4.2092e-012	5.6123e-013	7.4831e-014	9.9775e-015	1.3303e-015	1.7738e-016	2.3650e-017
1.5000e-001	2.2500e-002	3.3750e-003	5.0625e-004	7.5937e-005	1.1391e-005	1.7086e-006	2.5629e-007	3.8443e-008	5.7665e-009	8.6498e-010	1.2975e-010	1.9462e-011	2.9193e-012	4.3789e-013	6.5684e-014	9.8526e-015	1.4779e-015	2.2168e-016
1.6667e-001	2.7778e-002	4.6296e-003	7.7160e-004	1.2860e-004	2.1433e-005	3.5722e-006	5.9537e-007	9.9229e-008	1.6538e-008	2.7564e-009	4.5939e-010	7.6566e-011	1.2761e-011	2.1268e-012	3.5447e-013	5.9078e-014	9.8464e-015	1.6411e-015
1.8333e-001	3.3611e-002	6.1620e-003	1.1297e-003	2.0711e-004	3.7971e-005	6.9613e-006	1.2762e-006	2.3398e-007	4.2896e-008	7.8642e-009	1.4418e-009	2.6433e-010	4.8460e-011	8.8843e-012	1.6288e-012	2.9861e-013	5.4745e-014	1.0037e-014
2.0000e-001	4.0000e-002	8.0000e-003	1.6000e-003	3.2000e-004	6.4000e-005	1.2800e-005	2.5600e-006	5.1200e-007	1.0240e-007	2.0480e-008	4.0960e-009	8.1920e-010	1.6384e-010	3.2768e-011	6.5536e-012	1.3107e-012	2.6214e-013	5.2429e-014
2.1667e-001	4.6944e-002	1.0171e-002	2.2038e-003	4.7749e-004	1.0346e-004	2.2415e-005	4.8567e-006	1.0523e-006	2.2799e-007	4.9398e-008	1.0703e-008	2.3190e-009	5.0245e-010	1.0886e-010	2.3587e-011	5.1105e-012	1.1073e-012	2.3991e-013
2.3333e-001	5.4444e-002	1.2704e-002	2.9642e-003	6.9165e-004	1.6138e-004	3.7656e-005	8.7865e-006	2.0502e-006	4.7837e-007	1.1162e-007	2.6045e-008	6.0771e-009	1.4180e-009	3.3087e-010	7.7202e-011	1.8014e-011	4.2032e-012	9.8075e-013
2.5000e-001	6.2500e-002	1.5625e-002	3.9063e-003	9.7656e-004	2.4414e-004	6.1035e-005	1.5259e-005	3.8147e-006	9.5367e-007	2.3842e-007	5.9605e-008	1.4901e-008	3.7253e-009	9.3132e-010	2.3283e-010	5.8208e-011	1.4552e-011	3.6380e-012
2.6667e-001	7.1111e-002	1.8963e-002	5.0568e-003	1.3485e-003	3.5959e-004	9.5892e-005	2.5571e-005	6.8190e-006	1.8184e-006	4.8490e-007	1.2931e-007	3.4482e-008	9.1952e-009	2.4521e-009	6.5388e-010	1.7437e-010	4.6498e-011	1.2400e-011
2.8333e-001	8.0278e-002	2.2745e-002	6.4445e-003	1.8259e-003	5.1735e-004	1.4658e-004	4.1532e-005	1.1767e-005	3.3341e-006	9.4466e-007	2.6765e-007	7.5835e-008	2.1487e-008	6.0879e-009	1.7249e-009	4.8872e-010	1.3847e-010	3.9233e-011
3.0000e-001	9.0000e-002	2.7000e-002	8.1000e-003	2.4300e-003	7.2900e-004	2.1870e-004	6.5610e-005	1.9683e-005	5.9049e-006	1.7715e-006	5.3144e-007	1.5943e-007	4.7830e-008	1.4349e-008	4.3047e-009	1.2914e-009	3.8742e-010	1.1623e-010
3.1667e-001	1.0028e-001	3.1755e-002	1.0056e-002	3.1843e-003	1.0084e-003	3.1931e-004	1.0112e-004	3.2020e-005	1.0140e-005	3.2109e-006	1.0168e-006	3.2198e-007	1.0196e-007	3.2288e-008	1.0224e-008	3.2377e-009	1.0253e-009	3.2467e-010

The inverse of the coefficient matrix calculated by using MATLAB and the result was displayed as follows.

$$Q^{-1} = \begin{bmatrix}
1.1586e+003 & -5.2905e+003 & 2.0254e+004 & -6.1504e+004 & 1.4925e+005 & -2.9317e+005 & 4.7105e+005 & -6.2357e+005 & 6.8283e+005 & -6.1906e+005 & 4.6359e+005 & -2.8509e+005 & 1.4254e+005 & -5.7035e+004 & 1.7835e+004 & -4.2003e+003 & 7.0077e+002 & -7.3850e+001 & 3.6970e+000 \\
-1.7806e+005 & 9.7094e+005 & -3.9166e+006 & 1.2192e+007 & -3.0017e+007 & 5.9514e+007 & -9.6247e+007 & 1.2802e+008 & -1.4070e+008 & 1.2792e+008 & -9.6012e+007 & 5.9155e+007 & -2.9621e+007 & 1.1868e+007 & -3.7154e+006 & 8.7582e+005 & -1.4624e+005 & 1.5423e+004 & -7.7258e+002 \\
1.2438e+007 & -7.6272e+007 & 3.2455e+008 & -1.0388e+009 & 2.6007e+009 & -5.2141e+009 & 8.4989e+009 & -1.1371e+010 & 1.2553e+010 & -1.1453e+010 & 8.6208e+009 & -5.3238e+009 & 2.6711e+009 & -1.0720e+009 & 3.3606e+008 & -7.9314e+007 & 1.3258e+007 & -1.3995e+006 & 7.0160e+004 \\
-5.2632e+008 & 3.5068e+009 & -1.5639e+010 & 5.1466e+010 & -1.3117e+011 & 2.6620e+011 & -4.3777e+011 & 5.8960e+011 & -6.5427e+011 & 5.9942e+011 & -4.5270e+011 & 2.8034e+011 & -1.4098e+011 & 5.6690e+010 & -1.7802e+010 & 4.2077e+009 & -7.0424e+008 & 7.4422e+007 & -3.7348e+006 \\
1.5163e+010 & -1.0741e+011 & 4.9833e+011 & -1.6832e+012 & 4.3668e+012 & -8.9763e+012 & 1.4902e+013 & -2.0217e+013 & 2.2564e+013 & -2.0769e+013 & 1.5745e+013 & -9.7813e+012 & 4.9320e+012 & -1.9878e+012 & 6.2545e+011 & -1.4809e+011 & 2.4822e+010 & -2.6267e+009 & 1.3197e+008 \\
-3.1656e+011 & 2.3494e+012 & -1.1266e+013 & 3.8961e+013 & -1.0283e+014 & 2.1409e+014 & -3.5893e+014 & 4.9073e+014 & -5.5110e+014 & 5.0983e+014 & -3.8814e+014 & 2.4198e+014 & -1.2238e+014 & 4.9454e+013 & -1.5596e+013 & 3.6998e+012 & -6.2125e+011 & 6.5841e+010 & -3.3127e+009 \\
4.9737e+012 & -3.8280e+013 & 1.8870e+014 & -6.6636e+014 & 1.7871e+015 & -3.7678e+015 & 6.3798e+015 & -8.7925e+015 & 9.9389e+015 & -9.2446e+015 & 7.0702e+015 & -4.4251e+015 & 2.2455e+015 & -9.1003e+014 & 2.8772e+014 & -6.8410e+013 & 1.1510e+013 & -1.2220e+012 & 6.1580e+010 \\
-6.0241e+013 & 4.7724e+014 & -2.4076e+015 & 8.6603e+015 & -2.3573e+016 & 5.0297e+016 & -8.6002e+016 & 1.1949e+017 & -1.3599e+017 & 1.2721e+017 & -9.7768e+016 & 6.1449e+016 & -3.1297e+016 & 1.2724e+016 & -4.0343e+015 & 9.6163e+014 & -1.6216e+014 & 1.7251e+013 & -8.7091e+011 \\
5.7111e+014 & -4.6313e+015 & 2.3823e+016 & -8.7095e+016 & 2.4029e+017 & -5.1853e+017 & 8.9511e+017 & -1.2537e+018 & 1.4367e+018 & -1.3520e+018 & 1.0444e+018 & -6.5938e+017 & 3.3716e+017 & -1.3756e+017 & 4.3750e+016 & -1.0458e+016 & 1.7678e+015 & -1.8850e+014 & 9.5358e+012 \\
-4.2759e+015 & 3.5343e+016 & -1.8483e+017 & 6.8538e+017 & -1.9142e+018 & 4.1746e+018 & -7.2724e+018 & 1.0267e+019 & -1.1847e+019 & 1.1216e+019 & -8.7112e+018 & 5.5259e+018 & -2.8375e+018 & 1.1621e+018 & -3.7085e+017 & 8.8916e+016 & -1.5073e+016 & 1.6112e+015 & -8.1699e+013 \\
2.5374e+016 & -2.1308e+017 & 1.1301e+018 & -4.2431e+018 & 1.1982e+019 & -2.6388e+019 & 4.6371e+019 & -6.5972e+019 & 7.6649e+019 & -7.3016e+019 & 5.7021e+019 & -3.6350e+019 & 1.8749e+019 & -7.7095e+018 & 2.4694e+018 & -5.9405e+017 & 1.0101e+017 & -1.0828e+016 & 5.5048e+014 \\
-1.1920e+017 & 1.0143e+018 & -5.4447e+018 & 2.0668e+019 & -5.8950e+019 & 1.3100e+020 & -2.3210e+020 & 3.3267e+020 & -3.8913e+020 & 3.7297e+020 & -2.9290e+020 & 1.8768e+020 & -9.7257e+019 & 4.0163e+019 & -1.2915e+019 & 3.1183e+018 & -5.3202e+017 & 5.7208e+016 & -2.9167e+015 \\
4.4050e+017 & -3.7903e+018 & 2.0558e+019 & -7.8795e+019 & 2.2676e+020 & -5.0811e+020 & 9.0716e+020 & -1.3095e+021 & 1.5419e+021 & -1.4869e+021 & 1.1743e+021 & -7.5641e+020 & 3.9389e+020 & -1.6340e+020 & 5.2765e+019 & -1.2790e+019 & 2.1900e+018 & -2.3630e+017 & 1.2086e+016 \\
-1.2643e+018 & 1.0982e+019 & -6.0104e+019 & 2.3234e+020 & -6.7403e+020 & 1.5219e+021 & -2.7367e+021 & 3.9773e+021 & -4.7131e+021 & 4.5723e+021 & -3.6316e+021 & 2.3517e+021 & -1.2307e+021 & 5.1296e+020 & -1.6638e+020 & 4.0499e+019 & -6.9624e+018 & 7.5402e+017 & -3.8702e+016 \\
2.7589e+018 & -2.4161e+019 & 1.3326e+020 & -5.1904e+020 & 1.5167e+021 & -3.4484e+021 & 6.2426e+021 & -9.1310e+021 & 1.0887e+022 & -1.0624e+022 & 8.4854e+021 & -5.5244e+021 & 2.9061e+021 & -1.2172e+021 & 3.9664e+020 & -9.6978e+019 & 1.6743e+019 & -1.8206e+018 & 9.3809e+016 \\
-4.4203e+018 & 3.8981e+019 & -2.1648e+020 & 8.4882e+020 & -2.4966e+021 & 5.7126e+021 & -1.0406e+022 & 1.5313e+022 & -1.8364e+022 & 1.8024e+022 & -1.4476e+022 & 9.4752e+021 & -5.0103e+021 & 2.1091e+021 & -6.9067e+020 & 1.6966e+020 & -2.9425e+019 & 3.2138e+018 & -1.6630e+017 \\
4.9004e+018 & -4.3477e+019 & 2.4290e+020 & -9.5808e+020 & 2.8346e+021 & -6.5239e+021 & 1.1952e+022 & -1.7688e+022 & 2.1333e+022 & -2.1053e+022 & 1.7001e+022 & -1.1188e+022 & 5.9469e+021 & -2.5163e+021 & 8.2820e+020 & -2.0446e+020 & 3.5633e+019 & -3.9103e+018 & 2.0329e+017 \\
-3.3568e+018 & 2.9939e+019 & -1.6815e+020 & 6.6677e+020 & -1.9832e+021 & 4.5888e+021 & -8.4518e+021 & 1.2575e+022 & -1.5247e+022 & 1.5127e+022 & -1.2280e+022 & 8.1240e+021 & -4.3413e+021 & 1.8466e+021 & -6.1094e+020 & 1.5161e+020 & -2.6559e+019 & 2.9295e+018 & -1.5307e+017 \\
1.0700e+018 & -9.5870e+018 & 5.4098e+019 & -2.1553e+020 & 6.4417e+020 & -1.4977e+021 & 2.7722e+021 & -4.1453e+021 & 5.0514e+021 & -5.0373e+021 & 4.1105e+021 & -2.7335e+021 & 1.4684e+021 & -6.2791e+020 & 2.0886e+020 & -5.2110e+019 & 9.1784e+018 & -1.0180e+018 & 5.3486e+016
\end{bmatrix}$$

So, by multiplying Q^{-1} *by* $a_i's$ we will have the following values.

$$\begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \alpha_3 \\ \vdots \\ \alpha_{20} \end{bmatrix} = \begin{bmatrix} 109 & 2.3224e{+}007 & -4.7923e{+}009 & 4.2859e{+}011 & -2.2408e{+}013 & 7.7563e{+}014 & -1.9023e{+}016 \\ 3.4465e{+}017 & -4.7388e{+}018 & 5.0323e{+}019 & \cdots & & & \\ -4.1717e{+}020 & 2.7136e{+}021 & -1.3850e{+}022 & 5.5171e{+}022 & -1.6951e{+}023 & 3.9352e{+}023 & \\ -6.6706e{+}023 & 7.7850e{+}023 & -5.5889e{+}023 & 1.8596e{+}023 & & & \end{bmatrix}^T$$

This computation leads us to get the governing function f with their respective conditions as displayed in the following lines.

$$f(x) = 109 + (6234133056968353/268435456)*x - (2512548999448485/524288)*x^2 + (1755502906181383/4096)*x^3$$
$$-(5736392578750985/256)*x^4 + (6205046247124737/8)*x^5 - (19023112608783088)*x^6 + (344651962205307456)*x^7$$
$$-(4738841091626776576)*x^8 + (50323312017772527616)*x^9 - (417172156483298852864)*x^{10}$$
$$+(2713558697764479041536)*x^{11} - (13849897859614441996288)*x^{12} + (55170702892070133039104)*x^{13}$$
$$-(169511024182191928115200)*x^{14} + (393524688526126484553728)*x^{15} - (667058782115569686544384)*x^{16}$$
$$+(778499302591270089654272)*x^{17} - (558889075366920378646528)*x^{18} + (185955788415513546194944)*x^{19}$$

Now, the next step is to use Newton-Raphson Method to approximate the password of the user name maashookoo@gmail.com

Newton-Raphson method is defined as

$$x_1 = x_o - \frac{f(x_o)}{f'(x_o)}, \qquad (25)$$

by putting

$x_o = (a_i's)^T = [109 \quad 97 \quad 97 \quad 115 \quad 104 \quad 111 \quad 111 \quad 107 \quad 111 \quad 111 \quad 64 \quad 103 \quad 109 \quad 97 \quad 105 \quad 108 \quad 46 \quad 99 \quad 111 \quad 109]$.

Since $(a_i's)^T$ is a row matrix, the governing function has been written in $f((a_i's)^T)$. Then take the first derivative of f and substitute in equation (23), and iterate the function till the absolute error

$\xi = |p_a - p_{app}| \le \mu$, *where* $\mu = 1.0 \times 10^{-6}$, $\xi = absolute\ error$,

$p_a = actual\ password\ and\ p_{app} = the\ approximated\ password$.

Now the derivative of $f(x)$ is computed and the result was recorded as follows:

$$f'(x) = (6234133056968353/268435456) - (2512548999448485/262144)*x + (5266508718544149/4096)*x^2$$
$$-(5736392578750985/64)*x^3 - (31025231235623685/8)*x^4 - (114138675652698528)*x^5$$
$$+(2412563735437152192)*x^6 - (37910728733014212608)*x^7 + (45290980815995274854 4)*x^8$$
$$-(417172156483298852864 0)*x^9 - (298491456754092694568 96)*x^{10} - (166198774315373303955456)*x^{11}$$
$$+(717219137596911729508352)*x^{12} - (237315433855068699361 2800)*x^{13} + (590287032789189726830 5920)*x^{14}$$
$$-(106729405138491149847 10144)*x^{15} + (132344881440515915241 22624)*x^{16} - (100600033566045668156 37504)*x^{17}$$
$$+(353315997989475737770 3936)*x^{18}$$

Let $p_i's$ be the successive iterated value of the function up to the range of tolerable error. Then the following are the results obtained.

$p_o = (a_i's)^T = [109 \quad 97 \quad 97 \quad 115 \quad 104 \quad 111 \quad 111 \quad 107 \quad 111 \quad 111 \quad 64 \quad 103 \quad 109 \quad 97 \quad 105 \quad 108 \quad 46 \quad 99 \quad 111 \quad 109]$.

$p_1 = p_o - \left(\frac{f(p_o)}{f'(p_o)} \right)$; where p_1 is the first iterated value. Since matrix division is not allowed, the iteration was done in an element wise manner. Then by continuing this procedure up to the 38th (to have a uniform and consistent method for all) iterations we will obtain the best approximated value which is termed as the system's password. Hence, we can design any mail server by this fashion, thus numerical analysis is the key for the design.

4.4 Discussion

On this paper, different mail servers such as yahoo.com, and gmail.com were treated to obtain the pattern or the similarities and differences among them. After the two servers design were seen the mathematical meaning of electronic mail design was analyzed thoroughly.

For any domain name or email address, we obtained a unique curve that uniquely identifies the user name. No two or more mail addresses have the same curve on any mail server. By Picard's theorem, this curve has a unique solution. This unique solution(s) was/were calculated by using Newton-Raphson method for its being fast and easily converges to the required solution.

The cryptographic primitives were used to see mathematical meanings such as $Sig_A(m)$, $h(m)$, $PK_B(m)$, and $E_k(m)$ [4.2.1]. To send a mail message containing m to the

receiver R, the sender S first digitally signs (S, R, ttp, m) with his private key to produce $Sig_s(S, R, ttp, m)$. Then, S generates a session key k and encrypts the signed data under k using a symmetric key cryptosystem. Finally, S computes $h(m)$ and send the message sequentially from $M_1, \cdots, M_4$. By exp *ressions* $\mathbb{F}$ *nder* 4.1, 4.2, 4.3, 4.4, 4.5, 4.6, 4.7, *and* 4.8 the sent message have been checked where it is correct or corrupted. In addition to this, the results on 4.2.1, 4.2.2, 4.2.3, *and* 4.2.4 together with definitions 4.2.2, 4.2.3, *and* 4.2.4 validates the properties of Cantor set in line with its topological properties that makes the mail boxes stable.

CHAPTER FIVE

5. Conclusion, and Future Scope

The numerical computations have the power to design electronic mail boxes on Cantor set. Therefore, we conclude that any interested one can design a mail box by deriving spline functions and the interpolating polynomial of degree $N-1$ and using Newton-Raphson method along with the MATLAB software for computation of system based password.

In the future to utilize the best of internet access and mail box design on different servers I want to work in this direction by applying different mathematical methods.

APPEDICES

MATLAB codes for the programs and formulas used

```
U=double('maashookoo@gmail.com');
for i=1:18;
    b=
t=(6/(0.01666)^2)*(U(i)-2*U(i+1)+U(i+2))
end
```

```
U=double('maashookoo@gmail.com');
%U=[109 97 97 115 104 111 111 107 111 111 64 103 109 97 105 108 46 99 111 109];
for i=1:18;
t=(6/(0.01666)^2)*(U(i)-2*U(i+1)+U(i+2))
end
```

```
A=[1 4 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
   0 1 4 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0
   0 0 1 4 1 0 0 0 0 0 0 0 0 0 0 0 0 0
   0 0 0 1 4 1 0 0 0 0 0 0 0 0 0 0 0 0
   0 0 0 0 1 4 1 0 0 0 0 0 0 0 0 0 0 0
   0 0 0 0 0 1 4 1 0 0 0 0 0 0 0 0 0 0
   0 0 0 0 0 0 1 4 1 0 0 0 0 0 0 0 0 0
   0 0 0 0 0 0 0 1 4 1 0 0 0 0 0 0 0 0
   0 0 0 0 0 0 0 0 1 4 1 0 0 0 0 0 0 0
   0 0 0 0 0 0 0 0 0 1 4 1 0 0 0 0 0 0
   0 0 0 0 0 0 0 0 0 0 1 4 1 0 0 0 0 0
   0 0 0 0 0 0 0 0 0 0 0 1 4 1 0 0 0 0
   0 0 0 0 0 0 0 0 0 0 0 0 1 4 1 0 0 0
   0 0 0 0 0 0 0 0 0 0 0 0 0 1 4 1 0 0
   0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 4 1 0
   0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 4 1
   0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 4
   0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1];
B=inv(A)
```

```
M=1.0e+014*[-7.9265 2.1239 -0.5691 0.1525 -0.0409 0.0109 -0.0029 0.0008 -
0.0002 0.0001 -0.0 0.0 -0.0 0.0 -0.0 0.0 -0.0 0.0];
U=double('maashookoo@gmail.com');
for i=1:17
b=(U(i+1)-U(i))/0.01666-(0.01666/6)*(M(i+1)+2*M(i))
end
```

```
format shorte;
h=1/60;
A=zeros(19);
for i=1:19;
for j=1:19;
A(i,j)=(i*h)^j;
end
end
Q=inv(A);
F=double('maashookoo@gmail.com');
P=transpose(F);
T=Q*P;
d=T';
display(T);
Z=transpose(T);
w=[109    97    97   115   104   111   111   107   111   111    64   103
109    97   105   108    46    99   111 109];
symsx;
x=w;
for s=1:25
count=s
for l=1:19
     f=d(l)*(x).^(l);
     g=polyder(f);
     h=x-(f./g)
diff(f)
end
end

%MATLAB code used to compare gmail and yahoo servers/ to obtain the figures
s1=double('maashookoo@yahoo.com')
s11=s1;
s2=double('sherqpotyi@gmail.com');
s22=s2;
t=length(s1);
x=linspace(0,1/3,t);
xx=0:1/3;
ss=spline(x,s11,xx);
```

```
rr=spline(x,s22,xx);
plot(x,s11,'ko',x,s22,'bo',xx,ss,'r-',xx,rr,'b-')

%MATLAB code used to compare domain names on gmail server.
s1=double('maashookoo@gmail.com');
s2=double('sherqpotyi@gmail.com');
t=length(s1);
x=linspace(0,1/3,t);
xx=0:0.05:0.3333;
yy=spline(x,s1,xx);
% ss=spline(x,s1,xx);
rr=spline(x,s2,xx);
plot(x,s1,'ko',x,s2,'bo',xx,yy,'r-',xx,rr,'b-')
```

```
D=double('maashookoo@gmail.com');
H=D;
% D=double('maashookoo@gmail.com');
H=D;
for i=1:20
H(i)
End
A=[1 4 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
   0 1 4 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0
   0 0 1 4 1 0 0 0 0 0 0 0 0 0 0 0 0 0
   0 0 0 1 4 1 0 0 0 0 0 0 0 0 0 0 0 0
   0 0 0 0 1 4 1 0 0 0 0 0 0 0 0 0 0 0
   0 0 0 0 0 1 4 1 0 0 0 0 0 0 0 0 0 0
   0 0 0 0 0 0 1 4 1 0 0 0 0 0 0 0 0 0
   0 0 0 0 0 0 0 1 4 1 0 0 0 0 0 0 0 0
   0 0 0 0 0 0 0 0 1 4 1 0 0 0 0 0 0 0
   0 0 0 0 0 0 0 0 0 1 4 1 0 0 0 0 0 0
   0 0 0 0 0 0 0 0 0 0 1 4 1 0 0 0 0 0
   0 0 0 0 0 0 0 0 0 0 0 1 4 1 0 0 0 0
   0 0 0 0 0 0 0 0 0 0 0 0 1 4 1 0 0 0
   0 0 0 0 0 0 0 0 0 0 0 0 0 1 4 1 0 0
   0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 4 1 0
   0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 4 1
   0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 4
   0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1];
 B=inv(A);
```

```
ui=[2.5941e+005
    3.8911e+005
    -6.2690e+005
    3.8911e+005
    -1.5132e+005
    -8.6469e+004
    1.7294e+005
    -8.6469e+004
    -1.0160e+006
    1.8591e+006
    -7.1337e+005
    -3.8911e+005
    4.3235e+005
    -1.0809e+005
    -1.4051e+006
    2.4860e+006
    -8.8631e+005
    -3.0264e+005];
for i=1:18
M(i)=B*ui;
di=(M(i+1)-M(i))/(6*0.01666)
end

CI=MI/2
for i=1:18
H(i);
    t=(6/(0.01666)^2)*(H(i)-2*H(i+1)+H(i+2));
M(i)=B*t
end
for i=1:18;
t=(6/(0.01666)^2)*(H(i)-2*H(i+1)+H(i+2));
s(i)=t
end
```

BIBLIOGRAPHY

[1] Aliprantis, C. a. (1981). Principleof Real Analysis. *Book*, 26-51.

[2] Al-Rammahi, A. (2014). Cantor Interpolating Spline to Design Electronic Mail Boxes. *International Journal of Mathematical, Computational Science and Engineering*, 10-12.

[3] Bahreman, A. (1994). Certified Electronic Mail, In proceedings of the Network and Distributed Systems Security Conference. *DNA*, 3-19.

[4] Blundo, C. C. (2004). Modeling A Certified E-Mail Protocol Using I/O Automata. *Elsevier*, 344-357.

[5] Bothma, M. M. (2008). Combating The Corporate Paper War: Electronic Mail Abuse. *Journal of Information Management*, 1-12.

[6] Cheung, M. (2011). Factors affecting the Design of Electronic Direct Mail Messages: Implications for Professional Communicators. *IEEE*, 279-298.

[7] Cyders, T. S. (2014). *Basic Numerical Methods and FreeMat.*

[8] DiBenedetto, E. (2002). *Real Analysis.* New York: Birkhauser Boston Inc.

[9] Fairs, R. L. (1984). Numerical Analysis. *Brooks/Cole Publishing Company*, 2-8.

[10] Foo, S. H. (1996). A Hetrogeneous Multimedia electronic mail system. *Intenational Journal of Information Technology,*, 55-77.

[11] Gerganew, D. (n.d.). *en.wkipedia.org/wiki/Cantor_Set*. Retrieved Jan 22, 2014, from www.mathworks.org: http://en.wkipedia.org/wiki/Cantor_Set

[12] Hui, S. (1993). Amultimedia ElectronicMail System on Hetrogeneous Environment. *IEEE, Tenscon Proceeding*, 61-64.

[13] Lipschutz, S. (1965). *Theory and Problems in General Topology.* New York: Mc Graw-Hill Book Company.

[14] Micali, S. (1997). Certified email with invisible post offices, . *RSA*, 22-30.

[15] Nelson, D. R. (1990). The Cantor Set Breif Introduction. *University of California- Berkely, Berkely CA94704*, 6-11.

[16] Pugh, C. C. (2002). *Real Mathematical Analysis.* Verlag, UTM: Springer.

[17] Zhou, J. G. (1996). A fair non Repudation protocol. *Research in security and privacy*, 55-61.