I. J. Mathematical Sciences and Computing, **2020, 6, 9-14**
Published Online December 2020 in MECS (http://www.mecs-press.org/)
DOI: 10.5815/ijmsc.2020.06.02

An encoding schematic based on coordinate transformations

Awnon Bhowmik
Department of Mathematics, The City College of New York, 160 Convent Ave, New York, NY 10031, USA
Corresponding Author: abhowmik901@york.cuny.edu

Received: 09 September 2020; Accepted: 13 November 2020; Published: 08 December 2020

Abstract: This paper outlines an encoding schematic that is dependent on simple Cartesian coordinate transformations. Namely, the change of axes and the rotation of axes. A combination of these two is incorporated after turning singular ASCII values into 2D points. This system is based on multiple private keys that can also act as a potential candidate for threshold cryptography. Comprehensive initial testing has been performed on certain parameters by altering their values within a range. Further testing is required for more insights about the system. For now, the list of parameters that amounts to successful decryption is to be noted down for future use with this system.

Index Terms: change of axes, rotation of axes, rotation matrix, vector geometry, threshold cryptography, Koblitz encoding, cantor pairing function, 4-tuple parameter

1. Introduction

Over the years it has been realized that if utilized appropriately, any mathematical concept can be applied to make a mathematical cryptosystem. It could be as simple as a substitution cipher, as simple as the playful use of fractal geometry in a system [1]. It can also be a system that stems from fundamental concepts of number theory such as the integer factorization problem in RSA [2], or Chinese remainder theorem in various cryptographic protocols [3] and many others. This paper is solely due to a recreational endeavor. It consists of a method for secure communication between two people or groups. There has been previous research which is based on similar approach such as Hill Cipher [4]. Unlike the Hill Cipher, which consists of matrix multiplication modulo a number, our method avoids the modular division approach which guarantees effective decryption once a good set of parameters is used. The encryption schematic described in this paper is solely dependent on analytic and vector geometry. The data preprocessing however, can be performed using various methods. Two such methods are described here.

2. Coordinate Transformation

In 2D Cartesian plane, the only transformations available are change of axes via translation and rotation of axes through an arbitrary angle. They can be used one at a time or superimposed one on top of the other. Given a coordinate (h, k), a translation of axes is given by

$$(x_1, y_1) = (x_0 + h, y_0 + k) \tag{1}$$

And a rotation of axis is given by

$$(x_2, y_2) = \begin{pmatrix} cos\,\theta & -sin\,\theta \\ sin\,\theta & cos\,\theta \end{pmatrix} \begin{pmatrix} x_1 \\ y_1 \end{pmatrix} \tag{2}$$

Expanding it and writing it in terms of x_0, y_0, h, k gives us

$$(x_2, y_2) = \begin{pmatrix} x_1 \cos\theta - y_1 \sin\theta \\ x_1 \sin\theta + y_1 \cos\theta \end{pmatrix} = \begin{pmatrix} (x_0 + h)\cos\theta - (y_0 + k)\sin\theta \\ (x_0 + h)\sin\theta + (y_0 + k)\cos\theta \end{pmatrix} \tag{3}$$

A coordinate transformation such as this, however, is not deemed to be the only approach one can take. Rather than generating a value for θ, it is possible to for the user to define a 2×2 *key* matrix and perform a matrix multiplication to generate (x_2, y_2).

3.Koblitz Encoding & Decoding Algorithm

The encoding algorithm [5] is as follows

- Given a message M, convert each character m_k into a number a_k using Unicode, where $b = 2^{16}$ and $0 < a_k < 2^{16}$
- Convert the message M into an integer using

$$m = \sum_{k=1}^{n} a_k b^{k-1} \tag{4}$$

In practice an $n \leq 160$ is chosen such that m satisfies

$$m \leq 2^{16 \cdot 160} < p$$

- A number d is fixed such that $d \leq \frac{p}{m}$. In practice the prime p is chosen large enough so that $d = 100$ can be allowed.
- For integers $j = 0,1,2, \ldots d-1$, the following are performed
- x coordinate of a point on the elliptic curve is computed as

$$x_j \equiv (dm + j) \bmod p \text{ where } m = \left\lfloor \frac{x_j}{d} \right\rfloor$$

- Compute

$$s_j \equiv \left(x_j^3 + Ax + B\right) \bmod p \tag{5}$$

- If $\left(s_j\right)^{\frac{p+1}{2}} \equiv s_j \bmod p$, then y coordinate of a point on the elliptic curve is defined as $y_j \equiv \left(s_j\right)^{\frac{p+1}{4}} \bmod p$. Return the point $\left(x_j, y_j\right)$.

Thus, the message M is encoded as an element of the Abelian group $G = E\left(\mathbb{F}_p\right)$. The following is performed for decryption.

- Considering each point (x, y) and setting

$$m = \left\lfloor \frac{x-1}{k} \right\rfloor \tag{6}$$

which is essentially means

$$a_k \equiv \left\lfloor \frac{m}{b^{k-1}} \right\rfloor \bmod b \tag{7}$$

Thus, each character is recovered and concatenated to produce the original message M.

In the case of our algorithm we set this $k = 20$ and the prime $p = 751$.

4. Cantor Pairing Function

This is an elegant pairing function proposed by the Russian mathematician George Cantor that takes in two natural numbers and turns it into a single number. This function is a primitive recursive pairing function [6].

$$\pi: \mathbb{N} \times \mathbb{N} \to \mathbb{N}$$

And is defined by

$$\pi(x, y) = \frac{1}{2}(x + y)(x + y + 1) + y \tag{8}$$

Due to the way its defined, this is a one-to-one and onto function, which means it is invertible. This consequently means that given a single number, it can be readily mapped back to a unique (x, y) ordered pair.

To retrieve an ordered pair (x, y) from a given t, the following transformations are used

$$\omega = x + y$$
$$t = \frac{1}{2}\omega(\omega + 1)$$
$$z = t + y$$

From the second equation, cross multiplying gives a quadratic in ω

$$\omega^2 + \omega - 2t = 0$$

Solving it gives us

$$\omega = \frac{\sqrt{8t+1}-1}{2}$$

which is a strictly increasing and continuous function when t is non-negative real. Since

$$t \leq z = t + y < t + (\omega + 1) = \frac{(\omega+1)^2 + (\omega+1)}{2}$$

This implies that

$$\omega \leq \frac{\sqrt{8z+1}-1}{2} < \omega + 1$$

And thus

$$\omega = \left\lfloor \frac{\sqrt{8z+1}-1}{2} \right\rfloor$$

Finally calculate x and y from z as follows

$$\omega = \left\lfloor \frac{\sqrt{8z+1}-1}{2} \right\rfloor$$
$$t = \frac{\omega^2 + \omega}{2}$$
$$y = z - t$$
$$x = \omega - y$$

5. Algorithm

- Take an input string, split the string into its constituent characters and turn them into their respective ASCII values.
- Enter the parameters for an elliptic curve (a, b) such that

$$y^2 = x^3 + ax + b \pmod{p} \tag{9}$$

- Apply Koblitz Encoding to split each ASCII data into an ordered 2-tuple, i.e. (x, y) pair.
- Input a point (h, k) such that

$$(x_1, y_1) = (x_0 + h, y_0 + k)$$

- Input an integer $n \in \mathbb{Z}$ such that

$$\theta = \frac{n\pi}{2} \tag{10}$$

And the rotation matrix

$$R = \begin{pmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{pmatrix} \tag{11}$$

To obtain

$$\begin{pmatrix} x_2 \\ y_2 \end{pmatrix} = \begin{pmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{pmatrix} \begin{pmatrix} x_1 \\ y_1 \end{pmatrix} \tag{12}$$

- Make a list plot and send to receiver.
- Receiver retrieves the list of points from the plot, and performs the following

$$p' = Rp$$
$$R^{-1}p' = R^{-1}Rp$$
$$p = R^{-1}p \tag{13}$$

Where $p = (x_1, y_1)$

- Apply

$$(x_0, y_0) = (x_1 - h, y_1 - k) \tag{14}$$

- Pass these lists of points through the Koblitz decoding function to obtain the ASCII data.
- Convert the ASCII data to characters and join to obtain the original string.

6. Sample Test Run

Below is a snapshot of a sample test run showing that the algorithm works.


```
Enter a message: Hello

Enter elliptic curve parameters (a,b): 5 6

Enter change of axes parameters (h,k): 3 4

Enter an integer to generate angle of rotation matrix: 2

Encoded:
[[1441, 32], [2022, 148], [2163, 112], [2163, 112], [2221, 236]]

Time elapsed: 0.0010139942169189453

Encrypted message:
[[-1444, -36], [-2025, -152], [-2166, -116], [-2166, -116], [-2224, -240]]

Time elapsed: 0.00025010108947753906

Decrypted message:
[[1441.0, 32.0], [2022.0, 148.0], [2163.0, 112.0], [2163.0, 112.0], [2221.0, 236.0]]

Time elapsedL 0.00042819976806640625

Decoded message:
Hello

Time elapsed: 8.559226989746094e-05
```

7. Experimental Analysis

7.1 String length vs run time

A graph was plotted for [0,26] and [0,260] character length message, and the trend was observed to be linear, along with some inadvertent spikes due to system processing time and other procedures running on the machine during the benchmarking process. A set of good parameters $(a, b, h, k, n) = (5,7,3,4,1)$ were fixed beforehand.

The following table shows some results that were obtained

Table 1. String length vs Time

String length	Time (sec.)
26	0.00269
260	0.02564
2600	0.23829
26000	2.31413

Following is a graphical representation

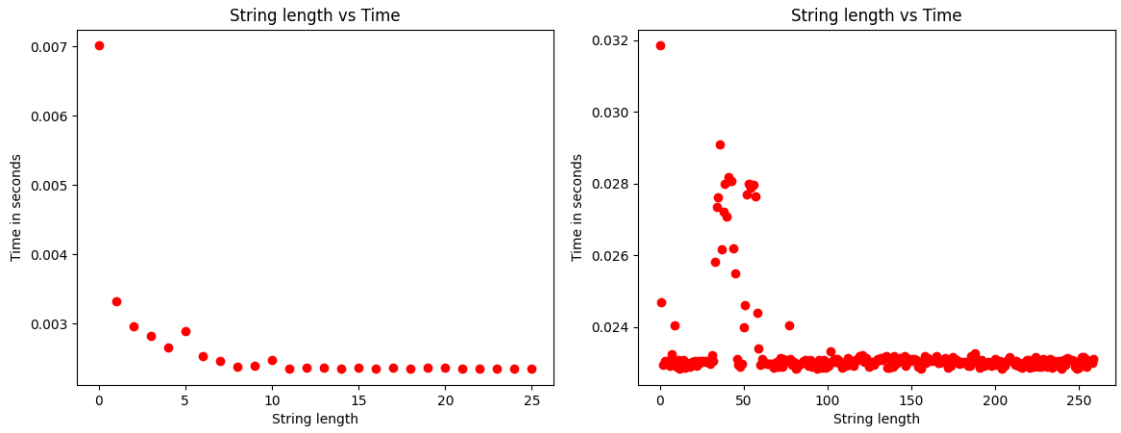

Figure 1. Effect of string length on runtime

7.2 Finding valid parameters

It was noted that not all messages decrypt correctly using all combinations of the private key parameters. So, we must have a set of valid combination of parameters that works. To achieve this, a code snippet was run, varying the parameters $(a, b, h, k) \in [-10,10]$ and $n \in [0,3]$. Finally, the decoded message was matched with the original message. It was noted that among all these combinations, more than 58% percent of the time, the message was successfully decrypted. However, $n \in \{0,2\}$ showed the most promise, with a decryption success of more than 82%. More work needs to be done in this area, but this is as much as was possible due to available equipment and processing prowess.

8. Future Work

We consider a situation in a company where there are three people A, B, C that report directly to the director. Suppose there is a situation when it was found that someone was embezzling money from the company's profits. So, this is a dire consequence that forces them to report to the boss. Once they write the secret message, one has the elliptic curve parameters a, b, the second provides the h, k pair and the third supplies the integer n. One advantage is, none of the other employees in the company can access or break the message unless all parameters are made available. If an adversary tries to intercept, it will take some time for him to succeed and by that time, the network scanner can detect anomaly and throw them off the server. On the other hand, if someone among these three people is the embezzler, they will not be able to suspect anything if the writer A reports to the boss and the other two B, C helps in encrypting it by supplying their piece of private key. If B or C is the embezzler, they can be easily tricked by this method by not suspecting that they are being reported. This means that with delicate improvements, this system is a good potential candidate for threshold cryptography [7]. A potential improvement can also be an exhaustive search to increase the domain for the parameter 4-tuple (a, b, h, k) such that valid combinations are put into lists, and the algorithm tweaked so that given a message, the application automatically suggests a valid 4-tuple combination to extract the maximum security possible.

9. Conclusion

Using the Koblitz encoding algorithm allows us to introduce two more private key parameters a, b for the elliptic curve, which is not the case if Cantor pairing function is used. Moreover, Cantor's pairing function is a generic algorithm which can be easily coded up if required. The pairing function requires two arguments x, y whereas the decoding algorithm takes a single argument which is the ASCII value.

Koblitz encoding algorithm also uses two parameters, a prime p and an arbitrary positive integer k. However, the user can be allowed to alter these values to experiment what effect it has in the runtime of the algorithm. Using the Koblitz algorithm showed us that this algorithm is not effective for all combinations of the private parameters. Further testing is required to narrow down the domain of the private key parameters. The parameters used in the sample test run was reused on a string containing 26,000 characters, which amounted to an overall runtime of about 2.3 seconds which is fair since this is not a block cipher protocol. All the results and codes for this research project has been compiled into a GitHub repository [8].

References

[1] A. Bhowmik and U. Menon, "Dragon Crypto - An Innovative Cryptosystem," International Journal of Computer Applications, vol. 176, no. 29, pp. 37-41, 2020.
[2] R. L. Rivest, A. Shamir and L. Adleman, "A method for obtaining digital signatures and public-key cryptosystems," Communications of the ACM, vol. 21, no. 2, pp. 120-126, 1978.
[3] D. Pei, A. Saloma and C. Ding, Chinese remainder theorem: applications in computing, coding, cryptography, World Scientific, 1996.
[4] L. S. Hill, "Cryptography in an algebraic alphabet," The American Mathematical Monthly, vol. 36, no. 6, pp. 306-312, 1929.
[5] R. Brady, N. Davis and A. Tracy, "Encrypting with Elliptic Curve Cryptography," in MSRI-UP, 2010.
[6] M. Suzdik, "An elegant pairing function," in Wolfram Research (ed.) Special NKS 2006 Wolfram Science Conference, Washington, DC, 2006.
[7] S. Henderson, NIST Kick-Starts 'Threshold Cryptography' Development Effort, 2020.
[8]A. Bhowmik, "CoordinateGeometryCrypto," GitHub, 8 September 2020. [Online]. Available: https://github.com/awnonbhowmik/CoordinateGeometryCrypto.

Authors' Profiles

Awnon Bhowmik received his Bachelor of Science in Mathematics and Computer Science from CUNY York College, and is currently pursuing a Master of Science degree in mathematics at The City College of New York, CUNY. His research interests are in Cryptography, Number Theory, Fractal Geometry, Mathematical Modelling and Simulation.

How to cite this paper: Awnon Bhowmik. " An encoding schematic based on coordinate transformations ", International Journal of Mathematical Sciences and Computing (IJMSC), Vol.6, No.6, pp.9-14, 2020. DOI: 10.5815/IJMSC.2020.06.02

