Journal of Computer Sciences and Applications, 2015, Vol. 3, No. 2, 46-51
Available online at http://pubs.sciepub.com/jcsa/3/2/5
© Science and Education Publishing
DOI:10.12691/jcsa-3-2-5

Selection of Fittest Key Using Genetic Algorithm and Autocorrelation in Cryptography

Sania Jawaid, Anam Saiyeda*, Naba Suroor

Department of Computer Science, Jamia Hamdard, New Delhi, India
*Corresponding author: anam.7sd@gmail.com

Received March 18, 2015; Revised April 03, 2015; Accepted April 16, 2015

Abstract Secure communication is a necessity in every field. Network security aims at providing a safe and unassailable correspondence by using cryptography. In cryptography data is sent in an encrypted form to ensure security. Encryption requires impregnable keys. A key is used to encrypt or decrypt data and should be unpredictable and not easily breakable. In this paper we use genetic algorithms which is a soft computing technique for key generation, the process used for generating keys. The keys obtained are tested for randomness by using the autocorrelation test. The final key is selected based on the autocorrelation value and thus it is as random and unique as possible. Java Technology is used to implement the proposed technique and analysis of the observations gives satisfactory results. The final key obtained can further be used to perform encryption. In our paper for verification and validation Data Encryption Standard cipher program is used. It can also be used in in-house ERP System to ensure the stability of important data. It will help in obtaining good standards of security in cryptography.

Keywords: *genetic algorithms, data encryption standard, cryptography, key generation*

Cite This Article: Sania Jawaid, Anam Saiyeda, and Naba Suroor, “Selection of Fittest Key Using Genetic Algorithm and Autocorrelation in Cryptography.” *Journal of Computer Sciences and Applications*, vol. 3, no. 2 (2015): 46-51. doi: 10.12691/jcsa-3-2-5.

1. Introduction

Safe and secure transfer of data is a stipulation in all domains ranging from two people's conversation to the country's defense and military. Data and transmission over the network is at threat from the hackers and the attackers spread everywhere [3]. Data can be stolen, changed, corrupted or lost. Thus network security is indispensable. The major role of network security lies in avoiding the tampering of data transmitted across the network. Cryptography is the art of protecting information by transforming it (encrypting it) into an unreadable format, called cipher text [17]. Cryptography can be categorized as symmetric key or asymmetric key. Symmetric key includes techniques like DES, AES and asymmetric includes Diffie-Hellman, RSA and other such techniques.

Encryption requires a key which is the backbone of encryption. It is a parameter necessary for making the data undecipherable. It aids in determining the functional output of the cryptographic algorithm. It specifies the particular transformation of the given plaintext into a cipher. Thus the key needs to be random so that cryptanalysis is a difficult task.

GAs approach is used along with the concept of autocorrelation in order to get a key which is unpredictable.

The paper has been organized in sections. Section 2 of the paper introduces GAs, section 3 gives a brief on cryptography and section 4 discusses autocorrelation. Section 5 and 6 cover the past work and literature review. Section 7 discusses the proposed work, section 8 illustrates observations, section 9 discusses inference and the last section concludes.

2. Genetic Algorithms

Genetic Algorithms (GAs) is based on an analogy with the process of evolution as given by Charles Darwin which involves descent with modification. It is a function that imitates the process of natural selection in the field of Artificial Intelligence (AI) [11,14]. The process is initiated with a population of chromosomes (solutions) with associated fitness values. Based on the fitness chromosomes are selected to engender the next generation which is fitter than the earlier one. The aim is to get better solutions over successive generations while the least fit solutions die out i.e. the survival of the fittest.

2.1. Population Generation

An initial population is randomly generated composed of several chromosomes which are either binary or hex depending on the type of population. There are various operators used for selection of individuals. The individuals are selected based on their probability, various genetic operations and their fitness value [8,15].

2.2. Crossover

After generating a new population crossover operator is applied on an individual. Crossover is the process of taking more than one parent solutions. Taking attributes from both the parents, crossover is performed. The offspring resulting from this has genetic material from both. We can have different types of crossover like single point crossover, two point crossover, uniform crossover.

Single point crossover is done by selecting one crossover point. The binary string from beginning of chromosome to the crossover point is taken from one parent, the rest of the string is copied from the second parent to obtain the offspring.

Two point crossover involves two crossover points. The first part is the string from beginning of chromosome to the first crossover point. It comes from one parent. The part from the first to the second crossover point is procured from the second parent and the rest is obtained from the first parent.

Uniform crossover is the random copying of bits from the first or from the second parent to obtain the progeny.

2.3. Mutation

Mutation means a change. It incites a random walk through the search space. Each gene is altered independently depending on the mutation rate thus aiding in maintaining genetic diversity among the different generations [4,8]. After the mutation operator is applied, the population remains same. But the less fit chromosomes are replaced by the more fit chromosomes. The main motive of the mutation process is to beget such chromosomes which have least similarities among themselves [6,15].

2.4. Selection

Selection stage is analogous to the theory of natural selection which states that individuals with characteristics which increase their probability of survival will have more opportunities to reproduce. After the population has been generated, the crossover and mutation operators have been applied and the individual fitness of each chromosome is calculated selection procedure is used for selecting the better individuals. More fit chromosomes are selected out of the existing population. These individuals have a higher chance over the other individuals based on the proportion to fitness [9,13]. Several methods are available to select the best chromosomes like roulette wheel selection, Boltzman selection and many others.

Roulette wheel selection takes fitness as the basis for selection. It is a genetic operator for selecting potentially useful solutions for recombination. It is also known as Fitness proportionate selection. It follows the concept that Fitter individuals have a better probability of survival and go forward to form the mating pool for the next generation. It is done by choosing the strings statistically based on their relative fitness values or simply by the percentage. [18] An analogy of a roulette wheel is considered where all chromosomes of the population are placed based on their fitness values they are given the space on the wheel. The wheel is either spun or a dice is thrown on it.

The number of times the roulette wheel is spun is equal to size of the population. In the result the probability of the fitter chromosome to occur is greater than the other population members.

These steps are repeated N number of times till we observe improvements in the new population.

3. Cryptography

'Cryptography' is a Greek word which means the style of secret writing [2,13]. It is used in network security to encrypt or alter the messages to get an unreadable format which is secure and an intruder is unable to decode the text. This way communication is possible over the internet or in presence of any third person without endangering the data. The plain text is converted into cipher text when the encryption algorithm is practiced. At the receiver's end cipher text is transmuted into the plain text. Ciphering of the plain text ensures no interruption or hacking of our message. Cryptography is divided into two types i.e. symmetric and asymmetric key cryptography.

3.1. Symmetric Key Cryptography

In this type, the sender and the receiver employ the equivalent key for encryption and decryption and hence the key is shared [1]. The sender uses this key and encryption algorithm to transform the plain text into cipher text [3]. The receiver uses the same key and decryption algorithm to convert the cipher text back into the plain text.

Some examples of symmetric cryptography are the ciphers like block ciphers - AES and DES.

DES denotes Data Encryption Standard. In this algorithm a 64 bit key is taken as input known as the plain text and yields a 64 bit ciphered text. Here two P boxes are found which are implemented as initial and final permutations. And also 16 rounds and for each round a key is produced.

AES denotes Advanced Encryption Standard. This algorithm was introduced because length of the key in DES was small [1]. To escalate the length of the key AES is employed. It has three kinds of rounds with equivalent bits of text. The functioning and operation is same but the alteration is in this order keys are used.

3.2. Asymmetric Key Cryptography

This type of cryptography is also called public key cryptography. In this, both the parties (sender and receiver) do not use the same key. The key is not shared. The sender uses the public key and encryption algorithm to encrypt the message [10]. The receiver uses the private key and decryption algorithm to convert the message back to plain text.

Asymmetric key cryptography examples are RSA and Diffie-Hellman Key Exchange [7,12].

RSA signifies Ron Rivest, Adi Shamir and Leonard Adleman [1]. It is a public key cryptography method. Here two large prime numbers are selected. Following the final calculation of numbers these are incorporated for encryption and decryption.

Diffie- Hellman Key Exchange is also a public key cryptography technique [3,14]. Two numbers 'x' and 'y' are picked up. Next R1 and R2 are calculated for exchange between Alice and Bob. After the exchange is processed the session key (K) is produced. Subsequently the session key is shared.

4. Autocorrelation

Autocorrelation denotes the correlation of a time series with its own past and future values. Autocorrelation is occasionally termed as “lagged correlation” or sometimes “serial correlation”, which speaks of the correlation amongst members of a series of numbers arranged or organized in time. Positive autocorrelation also symbolizes a precise form of “persistence”, a tendency for a system to continue in the same state from one observation to the subsequent one [22].

Karl Pearson's Coefficient of Correlation is the most widely used correlation coefficient. It is extensively used in the sciences to determine the degree of linear dependency between two variables. It recapitulates in one significant value, the degree of correlation & also the direction of correlation.

It is a measure of the linear correlation (dependence) between two variables *X* and *Y*, giving a value between +1 and −1 inclusive. A correlation of 1 signifies total positive correlation, 0 is no correlation, and −1 is total negative correlation. This coefficient can thus be used to check the randomness of numbers. The value of “r” which is the coefficient of correlation if is found to be nearer to 0 (zero) then it is attributed to be more random.

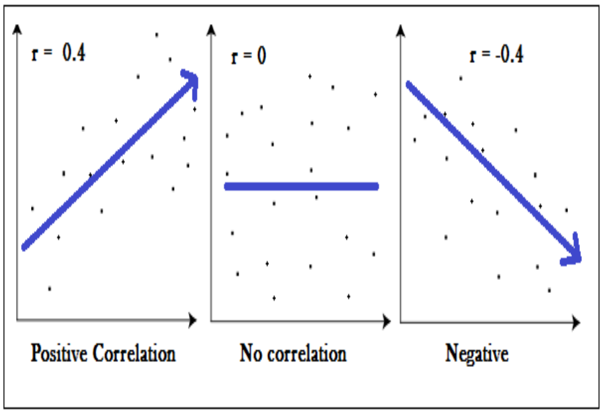

Figure 1. Graphs showing a correlation of -1 (a negative correlation), 0 and +1 (a positive correlation)

Table 1. Literature review of GAs techniques

S No.	Title	Author	Technique
1.	Using Genetic Algorithm for Symmetric key [20]	Aarti Soni and Suyash Agrawal	The pseudo random number sequence using the current date and the millisecond function was used to generate the initial population and each chromosome of that population is then divided into two equal halves and the crossover operation is applied on it. Swapping of bits is done in the next step. The whole process is repeated two times. The key is then used in the AES cipher.
2.	A Symmetric Key Encryption Technique Using Genetic Algorithm [5]	Sindhuja K and Pramela Devi S	The procedure has three main steps: 1) Key Generation Process: A “n” size block is chosen and the user input key is divided into the given block size and character z is appended if the last block has empty spaces remaining. Finally, the key is changed into its ASCII characters and shift operation is performed on its equivalent binary and the matrix of the order nxn is generated. 2) Substitution Algorithm: This algorithm works like a Shift cipher. 3) Encryption Process: Plain Text is converted into a text matrix. An additive matrix is generated which represents the summation of the plain text matrix and the key matrix. The generated matrix is then undergone the substitution process once to produce an intermediate cipher. Crossover and Mutation operations are applied on the intermediate cipher to generate the final cipher text.
3.	Key Generation for Cryptography using Genetic Algorithm [4]	Harsh Bhasin and Nakul Arora	After the initial population is generated, the chromosomes are converted into decimal numbers and the threshold check is performed on it. After this the crossover and mutation processes are applied and the threshold check is performed again. The generated new Population is stored in the data store and the Coefficient of autocorrelation is calculated for each sample. The sample having the CC value nearest to zero is selected and for each chromosome in that sample the fitness function value is calculated. The chromosomes are arranged in the decreasing order of their fitness value. Next the Roulette wheel selection is used for key replication and again crossover and mutation is performed, followed by the threshold check and finally the Chromosome with the highest fitness value is selected.
4.	Using Modified Genetic Algorithm in Private Key Cryptosystems: Key Generation and Expansion [21]	A Abdali Rashed	The first generation of population is produced using the random parameters as the cipher key. First Generation Initialization: Sixteen random numbers (cipher keys) are generated for the first generation. Each gene is of one byte (8 bits). All the elements are converted from the cipher key to the binary string. A random number is generated for a crossover point. Individuals are recombined and then mutated. Finally the new population is added to the older population. 2. Selection Operator: All elements found to be fit and are selected for the next Genetic operation.
5.	Generating the Best Fit Key in Cryptography Using Genetic Algorithm [19]	Sania Jawaid and Adeeba Jamal	Crossover and mutation operations are applied on the initial population generation and fitness value of each chromosome is calculated on the basis of Gap Test and Frequency Test. Ordering and Dominance Testing was then performed and the whole process was repeated n-times. Final key was then selected from repository having n-keys using Dominance Testing again. The final key was then used for encryption and decryption process in the DES cipher.

5. Background

GA was used to find the best fit key for the cryptographic algorithm. An approach of a pseudo random number generator was used to produce unique keys further used in the various ciphers. To make the key strong and almost unpredictable, a method was used which was based on the theory of natural selection. The basic processes in GAss, such as Initial Population Generation, Crossover, Mutation, Fitness Function Calculation and Final Key Selection were used. For calculating the fitness function Gap test and Frequency Test were performed. A 48-bit key Data Encryption Standard (DES) Cipher was used to show the implementation of the research.

Random samples were formed by generating a preliminary or initial random population of 100 chromosomes. Numerous tests were implemented on the samples and the results were observed.

Afterwards, crossover function was executed taking the total population size and the crossover rate into consideration for calculation. Mutation rate value was also selected. The fitness values of the keys were calculated and scrutinized by making use of the Frequency and Gap Tests. Thus the maximum frequency was detected in the sample. This resulted in the finding that chromosomes were repeated at most that many times. This shows the randomness of the sample used. Therefore, the final outcome came out to be as random, unique and exclusive as possible. The application further encompasses the use of DES cipher for data encryption. The whole solution was of seven rounds and the complete method was repeated 100 times. Even after this the key gets generated in a very less amount of time [19].

6. Literature Review

Literature Reviews are essential especially while doing an in depth analysis of the previous researches and providing better and feasible solution to currently existing frameworks. A paper is considered good only when it is well "re-searched" and has different ideas for an optimal solution. So let’s look at some of the proposed work by different authors in the area of cryptography and GAs.

7. Proposed Work

In this research paper, we use the processes of initial population generation, perform crossover and mutation on the population generated and check their randomness. Autocorrelation is used as fitness function. Autocorrelation is a statistical test that determines whether a random number generator is producing independent random numbers in a sequence [16].

To check the dependence between numbers within a sequence the test is implemented. After the initial population generation we perform the autocorrelation on the generated population to check for randomness.

Next we perform the autocorrelation on the population generated after Crossover. Similarly, after Mutation is performed autocorrelation is implemented on the obtained population.

In the result set we get three sets of population from each step and choose the population having the autocorrelation value nearest to zero which gets stored in the repository. All this process is executed “N” number of times and the final population having the best autocorrelation value is chosen. Thus from this final set of keys we choose a key randomly which goes for processing in the DES Cipher.

The proposed solution is depicted in the following flow chart

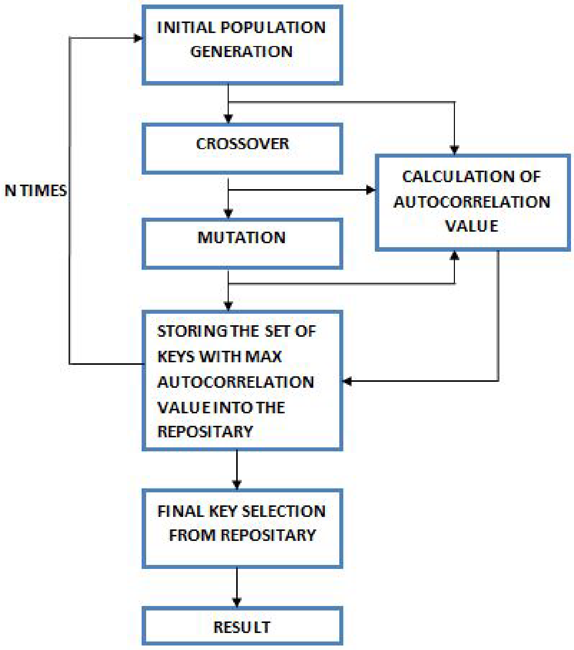
INITIAL POPULATION GENERATION
CROSSOVER
CALCULATION OF AUTOCORRELATION VALUE
N TIMES
MUTATION
STORING THE SET OF KEYS WITH MAX AUTOCORRELATION VALUE INTO THE REPOSITARY
FINAL KEY SELECTION FROM REPOSITARY
RESULT

Figure 2. Steps involved in Random key generation using GAs and autocorrelation in cryptography.

7.1. Population Generation

The process of GA initiates with randomly generated keys known as chromosome population. The Key size used is 48-bit key. The size of population will be influenced and vary on a huge number of solutions. Once the preliminary population is produced, the population set will go through various Genetic operations which escalate the number of chromosomes. Some of these individuals are probabilistically selected from the population to participate in the genetic operations. After this selection we perform the Autocorrelation test on the population set and further forward the result.

7.2. Crossover

Crossover can be easily understood as genetic recombination. Number ways are available by which it can be applied in GAs. Crossover is implemented on two randomly chosen individuals from the population set.

The successors generated from crossover are very diverse from their parents. The resulting individuals from Crossover again go through the Autocorrelation test and further passed on to the next GA operation. A crossover rate is chosen and then number of crossovers is calculated with the formula: noco = cor * m * n / 100.

Where noco = number of crossovers, cor = crossover rate, m = key length, n = number of keys

7.3. Mutation

Mutation is also one of the Genetic operations used here to preserve diversity from one generation of a population of chromosomes to the subsequent generation. It is equivalent to biological mutation. In mutation, there is a lot of chance that the result may change entirely from the former result. Mutation is a step occurred during evolution based on user-defined probability. The probability is set low.

Mutation is executed in such a way that the algorithm avoids the population of chromosomes from becoming too analogous to one another. Here also number of mutations is calculated via the formula, nom = mr * m * n / 100,

Where nom = number of mutations, mr = mutation rate, m = key length, n = number of keys

Again the resulting individuals from Mutation undergo the Autocorrelation test and result is forwarded.

7.4. Fitness Function

Fitness function is basically an objective function which defines how close the result is to the expected goal value. In the proposed solution, all the keys which are in binary format are first converted into decimal format. Autocorrelation test is then performed on them. The formula used for the autocorrelation test is the Karl Pearson's Coefficient of Correlation.

The correlation coefficient between *x* and *y* is given below:

$$r=\frac{\sum\left(X-\bar{X}\right)\left(Y-\bar{Y}\right)}{\sqrt{\sum\left(X-\bar{X}\right)^{2}}\sqrt{\sum\left(Y-\bar{Y}\right)^{2}}}$$

Where there is one dataset $\{x_1,....x_n\}$ containing “*n*” values and another dataset $\{y_1,....y_n\}$ containing “*n*” values.

$$\overline{X} = \frac{1}{n}\sum X$$

And similarly $\overline{Y}$ is also defined for Y.

Finally the final key is selected from the repository.

7.5. Final Key Selection from Repository

The complete process of Population generation, Crossover and Mutation is repeated “N” times and all the keys generated from “N” iterations are stored in the repository. The “N” value used in our work is 3 (N=3).

In N=1, we perform the three steps and from the various keys generated one can be chosen as the one with the greatest value. Again at N=2 and 3, the same procedure is used.

Finally the final key is selected on the basis of random key selection.

7.6. Result

The resulting final key can be used as an input for the encryption and decryption procedure in the DES cipher.

8. Observation

The technique proposed was accomplished using Java Technology and observations were scrutinized. These observations steered to the conclusions expressed in the next section.

The observations for the population set of 20 keys from the implementation of the proposed algorithm are as follows:

8.1. Iteration 1

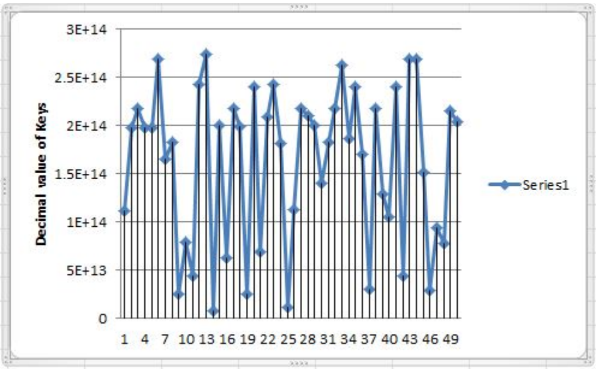

Figure 3. Sample set of highest autocorrelation value

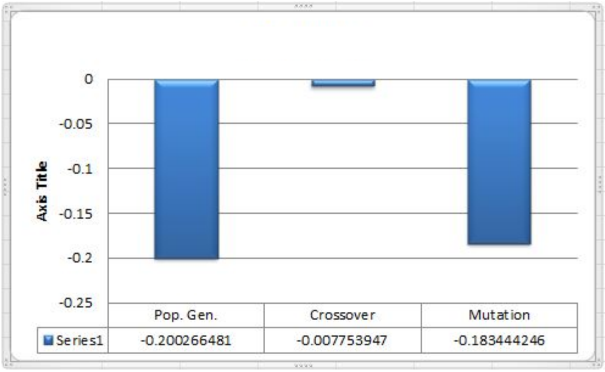

Figure 4. Comparison of autocorrelation coefficients

In the implementation of the proposed algorithm, when N=1, a random initial population of chromosomes is generated and the autocorrelation test is performed on it. The test is performed individually for Crossover and Mutation also. And we find that amongst all the resulting population the greatest value is that of the Initial Population Generation.

8.2. Iteration 2

In the second step when N=2, the above process is repeated again. A random initial population of chromosomes is generated and the autocorrelation test is performed on it. Similarly autocorrelation tests is performed for the population of Crossover and Mutation also. And we find that amongst all the resulting population when N=2 the best autocorrelation value is that of the Mutation Generation having the highest order of randomness.

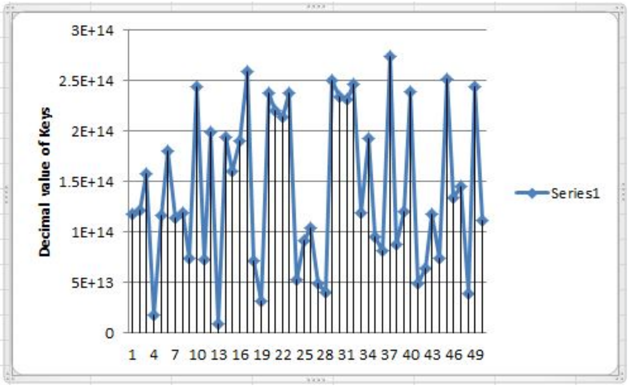

Figure 5. Sample set of highest autocorrelation value

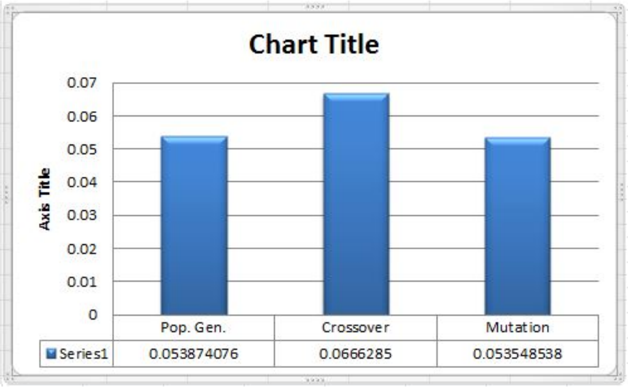

Figure 6. Comparison of autocorrelation coefficients

8.3. Iteration 3

In the third step when N=3, the above process is repeated again last time. A random initial population of chromosomes is generated and the autocorrelation test is performed on it. The test is performed individually for Crossover and Mutation also. And we find that amongst all the resulting population when N=3 the greatest value is that of the Crossover result.

Table 2. Comparison of autocorrelation values from repository

	Autocorrelation values
Iteration 1 Crossover set	-0.0077539468
Iteration 2 Population Generation set	0.0535485376
Iteration 3 Crossover set	-0.0111042183

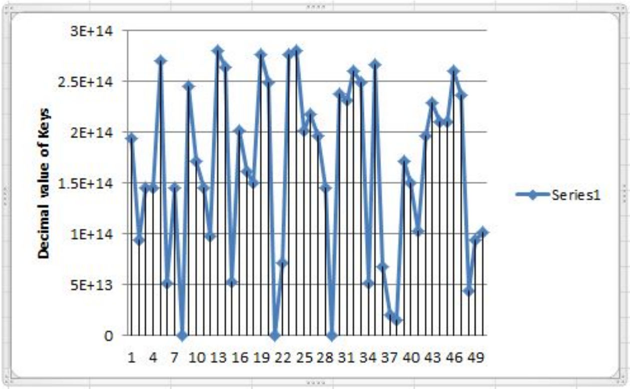

Figure 7. Sample set of highest autocorrelation value

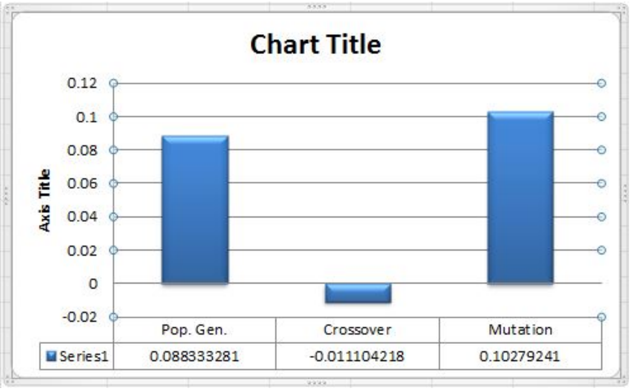

Figure 8. Comparison of autocorrelation coefficients

8.4. Final Key Selection from Repository

This final key selected is from the first iteration crossover sample set because it had the autocorrelation value which was the nearest to 0. Thus, the Final key is selected randomly from the sample set of first iteration crossover data which came out to be key number 12:

1001010111001010001000111110000111111110010000101

9. Results

The work has been instigated, evaluated and implemented. The process was implemented by designing a Java program. Random samples were produced by generating an initial random population of chromosomes. The autocorrelation test was applied on the samples and the result was obtained and found to be satisfactory.

10. Conclusion

It can be perceived from the above graphs that the produced key is random and challenging to decipher. The Genetic Operations involved are very intricate and when used one after the other, they produce the most random, arbitrary and non-repeating key. The operation execution further comprises the use of DES cipher used for data encryption which is very complex and makes the attack on data from cryptanalysts almost unmanageable and impossible. The suggested solution has a total of three rounds and the whole process can be repeated N times. Regardless of this, the key are generated in a little amount of time. This demonstrates to be a great benefit and advantage as the computational time of producing the key is smaller than encrypting the data using DES.

References

[1] Behrouz A Forouzan, “Data Communication and Networking” Tata McGraw- Hill Publishing Company Limited, Special Indian Edition 2006.

[2] William Stallings, “Network Security Essentials,” Fourth edition.

[3] William Stallings, “Cryptography and Network security”, Fifth Edition.

[4] Harsh Bhasin and Nakul Arora, “Key Generation for Cryptography using Genetic Algorithm”.

[5] Sindhuja K and Pramela Devi S, "A Symmetric Key Encryption Technique Using Genetic Algorithm", Sindhuja K et al, / (IJCSIT) International Journal of Computer Science and Information Technologies, ISSN: 0975-9646 Vol. 5 (1), 2014, pg 414-416.

[6] Y.V. Srinivasa Murthy, Dr. S. C. Satapathy, P. Srinivasu and A.A.S. Saranya, "Key Generation for Text Encryption in Cellular Networks using Multi-point Crossover Function", International Journal of Computer Applications (0975-8887) Volume 32-No. 9, October 2001.

[7] Bethany Delman, Genetic Algorithms in Cryptography, MS Thesis 2004.

[8] A. Tragha, F. Omary, A. Kriouile, “Genetic Algorithms Inspired Cryptography”, A.M.S.E Association for the Advancement of Modeling & Simulation Techniques in Enterprises, Series D: Computer Science and Statistics, November 2007.

[9] A Kumar, N Rajpal, Application of Genetic Algorithm in the Field of Steganography, in Journal of Information Technology, Vol. 2, No. 1, Jul-Dec. 2004, pg 12-15.

[10] Oded Goldreich, Foundations of Cryptography, Volume 1: Basic Tools, Cambridge University Press, 2001, ISBN 0-521-79172-3.

[11] A. J. Bagnall, “The Applications of Genetic Algorithms in Cryptanalysis”, School of Information Systems, University Of East Anglia, 1996.

[12] N. Koblitz, “A Course in Number Theory and Cryptography”, Springer-Verlag, New York, Inc., 1994.

[13] Harsh Bhasin, “Test Data Generation Using Artificial Life and Cellular Automata”, ACM SIGsoft Software Engineering Notes, January 2014.

[14] Harsh Bhasin, Neha Singla, “Cellular Genetic Test data Generation”: ACM Sigsoft Software Engineering Notes, September Edition, 2013.

[15] Harsh Bhasin et. al., “Cellular Automata based Test Data Generation”, ACM Sigsoft Software Engineering Notes.

[16] Dr. Albert D. Ritzhaupt, “Autocorrelation Random Number Test”, http://www.Aritzhaupt.com/resource/autocorrelation

[17] Vangie Beal, “Crytpography”, http://www.webopedia.com/TERM/C/cryptography.html

[18] Daniel Knight, “Ivory research”, http://www.ivoryresearch.com/writers/daniel-knight-ivory-research-writer/

[19] Sania Jawaid and Adeeba Jamal, “Generating the Best Fit Key in Cryptography using Genetic Algorithm”, International Journal of Computer Applications (0975 – 8887)Volume 98 – No. 20, July 2014.

[20] Aarti Soni and Suyash Agrawal, "Using Genetic Algorithm for Symmetric key", International Journal of Advanced Research in Computer Engineering & Technology (IJARCET), ISSN: 2278 – 1323 Volume 1, Issue 10, December 2012, http://ijarcet.org/wp-content/uploads/IJARCET-VOL-1-ISSUE-10-137-140.pdf

[21] A Abdali Rashed, "Using Modified Genetic Algorithm in Private Key Cryptosystems: Key Generation and Expansion", https://www.uop.edu.jo/CSIT2006/vol2%20pdf/pg100.pdf

[22] David M. Meko, "Applied Time Series Analysis", http://www.ltrr.arizona.edu/~dmeko/notes_3.pdf