A Comparison between Single and Multi- Crossover Points to Break Hill Cipher Using Heuristic Search: MA & GA

Dalal A. Hammood
Computer Engineerin Department / College of Electrical & Electronic Techniques
Foundation of Technical Education/Baghdad
Email: Alsaady_Dalal@yahoo.com

Received on: 22/12/2011 & Accepted on: 6/12/2012

ABSTRACT
Hill cipher is a classical cipher which is based on linear algebra. In this method, matrices and matrix multiplication have been used to combine the plaintext.

Heuristic search is a search techniques. The methods of HS are: (GA, SE, EP, MA, TS). Genetic algorithms are one of Heuristic search, it is search techniques which is used natural selection. GAs select optimal solution through three operations, they are: selection, crossover and mutation. The parameters are kept in memory and the best values of fitness have been selected to represent next generation.

Memetic Algorithm is one of Heuristic search, a memetic algorithm is an extension of the traditional genetic algorithm. It uses a local search technique to reduce the likelihood of the convergence, to reach the best solution.

This paper focuses on using MA and GA to find optimal solution to cryptanalyse Hill cipher. Then comparing two methods of crossover to see which one has best solution, and comparing between GA and MA to see which one has best solution.

MATLAB is used as M-FILE. The results of cryptanalysis cleared as following:
1- Without genetic algorithms: The number of correct letters for the key was 1 out of 9.
2- Using genetic algorithms: two methods are used, and they have been compared of crossover, they are single and multi- crossover points randomly. After (250) generation, the number of correct letters was 4 out of 9 when single crossover point is used. The number of correct letters was 8 out of 9 when multi crossover point are used. So multi crossover point have best solution. Genetic algorithms are applied successfully.
3- Using Memetic Algorithms. After (100) generation, the number of correct letters was 8 out of 9. So MA is better than Genetic algorithms.
4- the number of correct letter was 9 out of 9 when the MA is used.

Keywords: Hill cipher, Memetic Algorithm, Genetic Algorithms, Cryptanalyse,
A Comparison between Single and Multi-Crossover Points to Break Hill Cipher Using Heuristic Search: MA & GA

Key Search, single & multi crossover point.

A Comparison between Single and Multi-Crossover Points to Break Hill Cipher

A comparison between single and multi crossover points to break Hill Cipher using heuristic search: MA & GA.

Summary

Hill cipher is a classic cipher based on linear algebra. The method used in this research is a combination of classical Hill cipher and the genetic algorithm. Genetic algorithms are a class of search algorithms that mimic natural evolution. Genetic algorithms use three basic operators: selection, crossover, and mutation. Uber uses these operators to find the best solution. Genetic algorithms are typically used in optimization problems and are often used in computer science.

The best solution is reached by minimizing the fitness function. The fitness function is a measure of how well a solution meets the problem's requirements. The fitness function is usually a mathematical function that takes a solution as input and returns a scalar value as output. Genetic algorithms use a population of solutions, and each solution is represented by a chromosome. The chromosome is a string of genes, and each gene represents a variable in the problem. The genetic algorithm uses three basic operators: selection, crossover, and mutation. Selection is the process of choosing a subset of the population to be used in the next generation. Crossover is the process of combining two parent chromosomes to create offspring. Mutation is the process of randomly changing the value of a gene in a chromosome.

The best solution is reached by minimizing the fitness function. The fitness function is a measure of how well a solution meets the problem's requirements. The fitness function is usually a mathematical function that takes a solution as input and returns a scalar value as output. Genetic algorithms use a population of solutions, and each solution is represented by a chromosome. The chromosome is a string of genes, and each gene represents a variable in the problem. The genetic algorithm uses three basic operators: selection, crossover, and mutation. Selection is the process of choosing a subset of the population to be used in the next generation. Crossover is the process of combining two parent chromosomes to create offspring. Mutation is the process of randomly changing the value of a gene in a chromosome.
INTRODUCTION

Cryptography means hidden and secrecy in writing. The aim of cryptography to render a message incomprehensible to an unauthorized reader[1]. Cryptanalysis a set of rules which are in turn intended to make that unauthorized decryption more difficult (encryption security)[1]. Plaintext is the message that will be encrypted[2].

There are many papers that have been published about hill cipher, but these papers have not used genetic algorithms.

In 2007 A.Mahapatra & R. Dash presented data encryption and decryption by Using hill cipher technique and self Repetitive matrix[3].


In 2009 M. Toorani, & A. Falahati presented A Secure Variant of the Hill Cipher, that overcomes all the security drawbacks of the Hill cipher. The proposed scheme includes an encryption algorithm that is a variant of the Affine Hill cipher for which a secure cryptographic protocol is introduced[5].

In this paper two methods of crossover are used, and then comparing with them to see which one has the best solution. The two methods are: single crossover point, and multi crossover point.

Two algorithms are used Genetic and Memetic Algorithm, and comparing between them.

HILL CIPHER

Hill cipher is a classical cipher which is based on linear algebra. In this method, matrices and matrix multiplication have been used to combine the plaintext.

This cipher was invented in 1929 by Laster S. Hill . Let m be a positive integer. The idea is to take m linear combinations of the m alphabetic characters in one plaintext element, thus producing the m alphabetic characters in one ciphertext element.

For example, if m=3, the plaintext element is written as x=(x_1, x_2, x_3) and a ciphertext element as y=(y_1, y_2, y_3). Here, y_1 would be a linear combination of x_1, x_2, and x_3, as would y_2 and y_3, as shown below[1,2,6-8]:

\[
(y_1, y_2, y_3) = (x_1, x_2, x_3) \begin{pmatrix} k_{11} & k_{12} & k_{13} \\ k_{21} & k_{22} & k_{23} \\ k_{31} & k_{32} & k_{33} \end{pmatrix}
\]

Where x_1, ..., x_m are plaintext, k=(k_{ij}) is a key to encrypt or decrypt text, and y_1, ..., y_m are ciphertext. The matrix must be square (m*m).

The plaintext is X, which X is:

492
The Key cipher is  
Table 1 shows the number of each alphabetic letters[8-10].
Now, to encrypt plaintext, the key is following:

\[ k = \begin{pmatrix}
    17 & 17 & 5 \\
    21 & 18 & 21 \\
    2 & 2 & 19 \\
\end{pmatrix} \]

Plaintext= x = (algorithms are quite general definition of arithmetic processes). Equation (1) represents the encryption process.

\[ E = k \cdot x \mod 26 \] ……(1)

Where k is a key, x is a plaintext, and E is a ciphertext.
The message is broken into blocks, each block must be 3 letters and then multiplying each block with key[2].
alg= k.(0 11 6) mod 26 = J, rit= k.(17 8 19) mod 26 = M, and so on.
Then, the ciphertext is

JMGVOGIXUBHVMELWABDHTAQYFAACNFCDWGLUMAQVFFK
BNOQFSKYQWWW

To decrypt ciphertext, equation (2) shows the decryption process.

\[ D = X^{-1} \cdot K \] ……(2)

Where k is a key, x is the inverse ciphertext, and D is a decipher.

GENETIC ALGORITHMS
The Genetic Algorithm (GA) is a stochastic optimization strategy. It derives its behavior from a nature selection, and works by the creation a population of individuals represented by chromosomes. The individuals in the population then go through a process of evolution[11].

Figure 1 shows the structure of a simple genetic algorithm. Genetic algorithms work on populations of individuals instead of single solutions. In this way the search is performed in a parallel manner.
At the beginning of the computation a number of individuals (the population) are randomly initialized. The objective function is then evaluated for these individuals. The first/initial generation is produced.
If the optimization criteria are not met with the creation of a new generation starts. Individuals are selected according to their fitness for the production of offspring.
Parents are recombined to produce offspring. All offsprings will be mutated with a certain probability. The fitness of the offspring is then computed. The
offsprings are inserted into the population replacing the parents, producing a new generation.

In practice, the genetic model of computation can be implemented by having arrays of bits or characters to represent the chromosomes. Simple bit manipulation operations allow the implementation of selection, crossover, and mutation[11,12].

When the genetic algorithm is implemented it is usually done in a manner that involves the following cycle: Evaluate the fitness of all of the individuals in the population. Create a new population by performing operations such as crossover, fitness-proportionate reproduction and mutation on the individuals whose fitness has just been measured. Discard the old population and iterate using the new population[11].

One iteration of this loop is referred to as a generation. There is no theoretical reason for this as an implementation model. Indeed, behavior in populations in nature is not found as a whole, but it is a convenient implementation model[10-12].

The first generation (generation 0) of this process operates on a population of randomly generated individuals. From there on, the genetic operations, in concert with the fitness measure, operate to improve the population[9-13].

**PROPOSED ALGORITHM OF GA**

t=0;
initialize population P(t);
evaluate P(t);
until (done)
t=t+1;
parent selection P(t);
crossover P(t)
   1- single point crossover
   2- multi point crossover
mutate P(t);
swapping between position randomly
evaluate P(t);
survive P(t);

**MEMETIC ALGORITHM**

The genetic algorithm is a global search, this is not well suited to find optimal solution[14].

The memetic algorithms can be viewed as a marriage between a population-based global technique and a local search made by each of the individuals. They are a special kind of genetic algorithms with a local hill climbing. Like genetic algorithms, memetic Algorithms are a population-based approach. They have shown that they are orders of magnitude faster than traditional genetic Algorithms for some problem domains[15]. In a memetic algorithm the population is initialized at random or using a heuristic. Then, each individual makes local search to
improve its fitness. To form a new population for the next generation, higher quality individuals are selected. The selection phase is identical inform to that used in the classical genetic algorithm selection phase. Once two parents have been selected, their chromosomes are combined and the classical operators of crossover are applied to generate new individuals. The latter are enhanced using a local search technique. The role of local search in memetic algorithms is to locate the local optimum more efficiently then the genetic algorithm[15].

**THE ALGORITHM OF MA[16]**

\[
t = 0; \\
\text{set of population } P(t), \text{ max_gen, gen}=0; \\
cross_rate=0.2, \text{mut_rate}=0.2; \\
\text{initialize population} \\
evaluate P(t); \\
\text{until (done)} \\
t = t+1; \\
parent selection P(t); \\
crossover P(t) \\
\quad \text{multi point crossover} \\
\quad \text{mutate P(t);} \\
\quad \text{swapping between position randomly} \\
evaluate P(t); \\
apply local search \\
survive P(t); \\
applying final local search to best offspring
\]

**FITNESS MEASURE**

The technique used to compare candidate keys is to compare uni-gram statistics of the decrypted message with those of the language (which are assumed known)[13].

\[
Fitness = \alpha \left(1 - \sum_{i=1}^{26} \frac{\|SF[i] - DF[i]\|}{4}\right)^8 + \\
\beta \left(1 - \sum_{i=1}^{26} \frac{\|SDF[i][j] - DDF[i][j]\|}{4}\right)^8 \cdots (3)
\]

where SF and SDF denote the relative frequencies of single characters and digrams in the English language (respectively), and DF and DDF denote the relative frequencies of single characters and digrams in the message decrypted using key. Varying \( \alpha \) and \( \beta \) allows the weighing in favour of either the single character frequencies of the di-gram frequencies[13].
USING GENETIC ALGORITHMS TO ATTACK A HILL CIPHER

In the first phase, an initial population, describing representatives of the potential solution, is created to initiate the search process. The elements of the population are encoded into bit-strings, called chromosomes[17].

Another process is a selection. It is supposed to be able to compare each individual (chromosomes) in the population. Selection is done by using a fitness function.

After selection, two methods have been used to apply crossover. They are single point and two points crossover, to see which else has optimal solution.

1- Single point crossover

The traditional genetic algorithm uses single point crossover, where the two mating chromosomes are cut once at corresponding points and the sections after the cuts exchanged. Here, a cross-site or crossover point is selected randomly along the length of the mated strings and bits next to the cross-sites are exchanged. If appropriate site is chosen, better children can be obtained by combining good parents else it severely hampers string quality[12,17-19].

2- Two point crossover

In two-point crossover, two crossover points are chosen and the contents between these points are exchanged between two mated parents, as shown in Figure (2)[12,17-19].

After crossover, some keys are subjected to mutation. Mutation prevents the algorithm to be trapped in a local minimum. The mutation operation is used in this cipher. randomly select two elements in the child and swap those elements as shown in Figure (3)[12,17-19].

USING MEMETIC ALGORITHMS TO ATTACK A HILL CIPHER

For the memetic algorithm, the population size was set to 20; the probabilities for crossover and mutation were both 0.2 for all the test problems because it was the best configuration found empirically for the memetic algorithm. Stochastic selection is used. multi crossover is used to reproducing child. A marriage between a population-based global and local search is used to reproducing a new generation[16].

RESULTS

The attack to hill cipher was implemented for population size 20, different numbers of generations and mutation rate 0.2.

1- Using Genetic Algorithm.

Figure (4) shows the single point crossover process. It shows the relation between the number of correct letters and different number of generations for population 20. It is clear that the number of correct letters is 4 out of 9 after 250 generation when the population size 20. This solution is not enough.
A Comparison between Single and Multi-Crossover Points to Break Hill Cipher
Using Heuristic Search: MA & GA

Figure (5) shows the two point crossover process. It shows the relation between number of correct letters and different number of generations for population 20. It is clear that the number of correct letters is 8 out of 9 after 250 generation when the population size 20.

Figure (6) shows the elapsed time between the number of correct letters and different number of generations for population 20. It is clear that the elapsed time is increased when the number of generations increase for two point crossover.

2- Using Memetic Algorithm

Figure 7 shows the two point crossover process. It shows the relation between number of correct letters and different number of generations for population 20. It is clear that the number of correct letters is 8 out of 9 after 100 generation when the population size 20, and 9 out of 9 after 150 generation.

Figure (8) shows a comparison between GA and MA. It is clear that the MA is better than GA to find a good solution.

COMPARISON RESULTS
The true key is:
R R F V S V C C T

Table (2) shows decryption ciphertext without genetic algorithm. It is clear that the number of correct letter is 1 out off 9.

Table (3) shows decryption ciphertext with genetic algorithms using two point crossover. It is clear that the number of correct letter is 8 out off 9.

Table (4) shows decryption ciphertext with memetic algorithm using two point crossover. It is clear that the number of correct letter is 9 out off 9.

The best key is:
\[
\begin{bmatrix}
R \\
U \\
F
\end{bmatrix}
\begin{bmatrix}
V \\
S \\
V
\end{bmatrix}
\begin{bmatrix}
C \\
C \\
T
\end{bmatrix}
\]

CONCLUSIONS
In this paper memetic and genetic algorithms, were implemented successfully to break a hill cipher.

This work has used single letters frequency. Using different number of generations, and population 20. the text length is 59 letters and mutation rate is 0.2, key size is 9 letters.

The algorithms were implemented using the MATLAB program. Different parameters were tested such as the number of population and the time required finishing the algorithm for different number of generations.

Two algorithms are used to see which one has a best key.

Using Genetic Algorithms: The best number of correct letters was 4 out of 9 letters after 250. generation for single point crossover. When Multi points crossover is used, the number of correct letters was 8 out of 9, after 250 generation.
Using Memetic Algorithm: The best number of correct letters was 9 out of 9 letters after 100 generation for multi (two) point crossover which represent the best solution.

So memetic algorithms gives best solution.

The algorithms were run in MATLAB for the processor 2.70 GHz and RAM 512MB.

REFERENCES
A Comparison between Single and Multi-Crossover Points to Break Hill Cipher Using Heuristic Search: MA & GA


Table (1) numbers of each alphabetic letters.

<table>
<thead>
<tr>
<th>a</th>
<th>b</th>
<th>c</th>
<th>d</th>
<th>e</th>
<th>f</th>
<th>g</th>
<th>h</th>
<th>i</th>
<th>j</th>
<th>k</th>
<th>l</th>
<th>m</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td>9</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
</tr>
<tr>
<td>n</td>
<td>o</td>
<td>p</td>
<td>q</td>
<td>r</td>
<td>s</td>
<td>t</td>
<td>u</td>
<td>v</td>
<td>w</td>
<td>x</td>
<td>y</td>
<td>z</td>
</tr>
</tbody>
</table>

Table (2) Number of correct letters Without using Gas.

<table>
<thead>
<tr>
<th>population</th>
<th>Keys</th>
<th>Number of correct letters</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>HIEFBCDGA</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>BFAHGIEDC</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>BCEHIFGDA</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>GCEDFAHIB</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>CGBFEDHIA</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>AECBFGHDI</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>FDHAEBGCI</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>HEGFACBD</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>CDBEFHIGA</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>DBFEGAHIC</td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>HHEFBRDSSB</td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>FFRRBDTUR</td>
<td></td>
</tr>
<tr>
<td>13</td>
<td>UGGRYTFWREY</td>
<td></td>
</tr>
<tr>
<td>14</td>
<td>FHHHDTRRR</td>
<td></td>
</tr>
<tr>
<td>15</td>
<td>VTFHRETTR</td>
<td></td>
</tr>
<tr>
<td>16</td>
<td>BAEWTRETO</td>
<td></td>
</tr>
<tr>
<td>17</td>
<td>HTREWQIUY</td>
<td></td>
</tr>
<tr>
<td>18</td>
<td>REERIITURE</td>
<td></td>
</tr>
<tr>
<td>19</td>
<td>KJUYTTEWQ</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>ACTRVAAUT</td>
<td></td>
</tr>
</tbody>
</table>
Table (3) Number of correct letters after 250 generation for two point crossover using GA.

<table>
<thead>
<tr>
<th>Population</th>
<th>Keys</th>
<th>Number of correct letters</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>HHFRBDSSB</td>
<td>\</td>
</tr>
<tr>
<td>2</td>
<td>TUTRVVCCT</td>
<td>4</td>
</tr>
<tr>
<td>3</td>
<td>ACTRVVFUT</td>
<td>\</td>
</tr>
<tr>
<td>4</td>
<td>HEGFIACBD</td>
<td>\</td>
</tr>
<tr>
<td>5</td>
<td>FRHDDTFHA</td>
<td>\</td>
</tr>
<tr>
<td>6</td>
<td>RFHVSVCCT</td>
<td>8</td>
</tr>
<tr>
<td>7</td>
<td>HHFRBDSSB</td>
<td>\</td>
</tr>
<tr>
<td>8</td>
<td>RFHFIIVCCT</td>
<td>\</td>
</tr>
<tr>
<td>9</td>
<td>REERJUTURE</td>
<td>\</td>
</tr>
<tr>
<td>10</td>
<td>RFHFIACCT</td>
<td>\</td>
</tr>
<tr>
<td>11</td>
<td>KJUYYTTEWQ</td>
<td>\</td>
</tr>
<tr>
<td>12</td>
<td>RFHFIJVCCE</td>
<td>4</td>
</tr>
<tr>
<td>13</td>
<td>AECBFIGHDI</td>
<td>\</td>
</tr>
<tr>
<td>14</td>
<td>RFHFIIVCCA</td>
<td>\</td>
</tr>
<tr>
<td>15</td>
<td>UGGYTREWY</td>
<td>\</td>
</tr>
<tr>
<td>16</td>
<td>RRRTVVCCT</td>
<td>5</td>
</tr>
<tr>
<td>17</td>
<td>HTREWQUY</td>
<td>\</td>
</tr>
<tr>
<td>18</td>
<td>RRFTVVCCT</td>
<td>7</td>
</tr>
<tr>
<td>19</td>
<td>RFHFIIVCCT</td>
<td>7</td>
</tr>
<tr>
<td>20</td>
<td>GCEFDFAHIB</td>
<td>\</td>
</tr>
</tbody>
</table>

Table (4) Number of correct letters after 100 generation for two point crossover using MA.

<table>
<thead>
<tr>
<th>Population</th>
<th>Keys</th>
<th>Number of correct letter</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>RUTRVSCCA</td>
<td>4</td>
</tr>
<tr>
<td>2</td>
<td>RFHFIIVCCT</td>
<td>7</td>
</tr>
<tr>
<td>3</td>
<td>RFHVSVCCT</td>
<td>8</td>
</tr>
<tr>
<td>4</td>
<td>RFHFIIVCCT</td>
<td>\</td>
</tr>
<tr>
<td>5</td>
<td>RFHFIACCT</td>
<td>\</td>
</tr>
<tr>
<td>6</td>
<td>RFHFIIVCCE</td>
<td>4</td>
</tr>
<tr>
<td>7</td>
<td>RFHFIIVCCA</td>
<td>\</td>
</tr>
<tr>
<td>8</td>
<td>RRRTVVCCT</td>
<td>5</td>
</tr>
<tr>
<td>9</td>
<td>RRFTVVCCT</td>
<td>7</td>
</tr>
<tr>
<td>10</td>
<td>RFHFIIVCCT</td>
<td>7</td>
</tr>
<tr>
<td>11</td>
<td>RFHFIIVCCA</td>
<td>7</td>
</tr>
<tr>
<td>12</td>
<td>RFHFSVCCCT</td>
<td>9</td>
</tr>
<tr>
<td>13</td>
<td>RFHFIIVCCT</td>
<td>7</td>
</tr>
<tr>
<td>14</td>
<td>RRRTVVCCT</td>
<td>5</td>
</tr>
<tr>
<td>15</td>
<td>RFHFSVCCU</td>
<td>8</td>
</tr>
<tr>
<td>16</td>
<td>RFHFSVCCU</td>
<td>8</td>
</tr>
<tr>
<td>17</td>
<td>RFHFIIVCCA</td>
<td>6</td>
</tr>
<tr>
<td>18</td>
<td>RFHFIIVCCT</td>
<td>7</td>
</tr>
<tr>
<td>19</td>
<td>RFHFIIVCCA</td>
<td>6</td>
</tr>
<tr>
<td>20</td>
<td>RFHFSVCCCT</td>
<td>9</td>
</tr>
</tbody>
</table>
A Comparison between Single and Multi-Crossover Points to Break Hill Cipher Using Heuristic Search: MA & GA

Figure (1) Cycle Of Gas.

<table>
<thead>
<tr>
<th>Parent 1:</th>
<th>GCE</th>
<th>DFA</th>
<th>HIB</th>
</tr>
</thead>
<tbody>
<tr>
<td>Parent 2:</td>
<td>DBF</td>
<td>EGA</td>
<td>HIC</td>
</tr>
</tbody>
</table>

Child 1: HIC DFA DBF
Child 2: HIB EGA GCE

Figure (2) Applying Crossover.

<table>
<thead>
<tr>
<th>Parent 1:</th>
<th>HIC</th>
<th>DFA</th>
<th>DBF</th>
</tr>
</thead>
<tbody>
<tr>
<td>Parent 2:</td>
<td>HIB</td>
<td>EGA</td>
<td>GCE</td>
</tr>
</tbody>
</table>

Child 1: HIB DFA GBF
Child 2: HIC EGA DCE

Figure (3) Applying Mutation.
A Comparison between Single and Multi-Crossover Points to Break Hill Cipher Using Heuristic Search: MA & GA

Figure (4) Single point crossover, pop. 20, No. of correct letters & different No. of generations.

Figure (5) Two point crossover, pop. 20, no. of correct letters & different no. of generations.
Figure (6) Elapsed time between no. of correct letters & different no. of generation.

Figure (7) Two point crossover, population 20, number of correct letters and different number of generations using MA.
A Comparison between Single and Multi-Crossover Points to Break Hill Cipher Using Heuristic Search: MA & GA

Figure (8) A comparision between GA and MA.