CHAPTER 2

LITERATURE SURVEY

2.1 INTRODUCTION

There have been many researches done on genetic algorithms and their application to various problems, there are relatively few papers that apply genetic algorithms to cryptanalysis. The earliest papers that use a genetic algorithm approach for a cryptanalysis problem were written in 1993, almost twenty years after the primary paper on genetic algorithms by John Holland (Holland 1975). These papers focus on the Cryptanalysis of some classical ciphers.

2.2 SIMPLE SUBSTITUTION CIPHER

Spillman et al (1993) focuses in their paper, on the cryptanalysis of a simple substitution cipher using genetic algorithm. In their work, genetic algorithm is selected as an approach so that a directed random search of the key space can be achieved. The key representation is an ordered list of alphabet characters, where the first character in the list is the plaintext character for the most frequent ciphertext character, the second character is the plaintext character for the second most frequent ciphertext character, and so on. The fitness function uses unigrams and bigrams and it is as follows:

\[
\text{fitness} = (1 - \frac{\sum_{i=1}^{26} (|SF[i] - DF[i]| + \sum_{j=1}^{26} |SDF[i, j] - DDF[i, j]|)}{4})^8 \quad (2.1)
\]
SF[i] is the standard frequency of character i in English plaintext, DF[i] is the measured frequency of the decoded character i in the ciphertext. SDF is the standard frequency for a bigram and DDF the measured frequency. The result has shown that by increasing the size of the gene pool, the number of generations required to get the acceptable key can be reduced.

Andrew John Clark (1998) in his research has investigated the use of various optimization heuristics in the fields of automated cryptanalysis and automated cryptographic function generation. The various optimization heuristics are Simulated Annealing, Genetic Algorithm and Tabu Search. New attacks on these ciphers are proposed which utilize Simulated Annealing and the Tabu Search. Existing attacks which make use of the genetic algorithm and simulated annealing are compared with the new simulated annealing and Tabu Search techniques. In the cryptanalysis of simple substitution cipher, he used the frequency analysis as a fitness function in Genetic Algorithm. The result has shown that, all three techniques were found to successfully cryptanalyse simple substitution ciphers.

Grundlingh and van Vuuren (2003) have shown how the use of Genetic Algorithm leads to efficient cryptanalysis of monoalphabetic substitution ciphers. The fitness function which is used in the Genetic Algorithm involves discrepancies between the expected number of occurrences and the observed number of occurrences of a letter. The paper concludes by describing a decision tool created to assist cryptanalysts who routinely attack substitution ciphers. This tool is graphical in nature and is based on the Genetic Algorithm attack on substitution ciphers.

Gester (2003) used Genetic Algorithm in cryptanalysis of simple substitution cipher. Each individual in the population represents a single guess at the correct key for the substitution cipher. Each gene represents a
translation table mapping the characters of the cipher-text to the characters of plain-text. This is implemented as a list of characters, the first of which is the character mapped to by 'a', the second mapped to by 'b' and so on. This is simply implemented as a list indexed by the characters 'a' through 'z'. So, to preform a decryption using an individual is simple, just look up each character of the plain-text using the key and record the value found. The fitness function for rating the quality of each individual in the population’s solution is based on trigram and digram counts. A corpus of the text must be supplied. From this, a table of the number of times each trigram and diagram occurs is generated. Mutation is implemented simply as a random swap between two elements of the mapping list. Crossover is more complex. In this implementation, a pair of individuals produces a single new individual. A copy of one of the parent individual is selected and a section of the other parent’s gene/key is chosen. Then the character in each element of the selected region is found in the child’s key and is swapped with the character currently in the desired position. The average fitness does climb for several generations and occasionally climbs quite high but rarely is an usable result produced.

Verma et al (2007) presents a cryptanalysis method based on Genetic Algorithm and Tabu Search to break a Mono-Alphabetic Substitution Cipher in Adhoc networks. He has compared and analyzed the performance of these algorithms in automated attacks on Mono-alphabetic Substitution Cipher with two types of metrics, first metric is ciphertext size and the second is the time taken by the algorithm.

2.3 POLYALPHABETIC SUBSTITUTION CIPHER

Dimovski and Gligoroski (2003a) proposed an automated attack on the polyalphabetic substitution cipher. The attack described here effectively
reduces the complexity of a polyalphabetic substitution cipher attack to that of a monoalphabetic one, if there is a computer with $B$ processing nodes, where $B$ is the period of the polyalphabetic substitution cipher. The frequency analysis was the method to evaluate the fitness function. Frequency of unigrams is used in the partial keys. For evaluating the whole key, bigrams and trigrams of English letters were used. The attack was implemented with a polyalphabetic substitution cipher with a block size of three. The attack was run 100 times for different sizes of known ciphertext characters per key. The results showed that the parallel genetic algorithm is powerful technique for attack.

2.4 VIGENERE CIPHER

Kasiski (www.trincoll.edu/depts/cpsc/cryptography/vigenere.html) developed a technique to deduce the length of the keyword used in the polyalphabetic substitution cipher. The Kasiski examination allows a cryptanalyst to deduce the length of the keyword used in the polyalphabetic substitution cipher. Once the length of the keyword is discovered, the cryptanalyst lines up the ciphertext in $n$ columns, where $n$ is the length of the keyword. Then, each column can be treated as the ciphertext of a monoalphabetic substitution cipher. As such, each column can be attacked with frequency analysis. The Kasiski examination involves looking for strings of characters that are repeated in the ciphertext. The strings should be three characters long or more for the examination to be successful. Then, the distances between consecutive occurrences of the strings are likely to be multiples of the length of the keyword. Thus, finding more repeated strings narrow down the possible lengths of the keyword, since the greatest common divisor of all the distances can be taken.
The reason this test works is that, if a repeated string occurs in the plaintext and the distance between them is a multiple of the keyword length, the keyword letters will line up in the same way with both occurrences of the string.

Jones and Christman (2001) developed solutions to a Vigenere alphabetic code. The Genetic Algorithm uses the number of characters in valid English words as the measure of a solution’s fitness; breeding is accomplished by simple crossover and mutation consists of random variations of a single keyword position. They introduced a new concept by including a small portion of the worst solutions as well as the best. Their method for breaking the code relied on a large ratio between the length of the encoded text and the length of keyword: \[ R = \frac{\text{length(ciphertext)}}{\text{length(codeword)}} \]. As this ratio decreases, it becomes increasingly difficult to create a statistically accurate analysis of letter frequencies. This ratio shows that when the codeword and encoded text are of the same length it is impossible to decode the text. For that they presented an alternate technique, using a Genetic Algorithm to discover the code word and the plain text. The research has shown that a genetic algorithm can be used to quickly and “intelligently” solve a problem that would have otherwise required a long exhaustive search. In the implementation of the Genetic Algorithm the concept of “anti-elitism” is introduced, the idea that some of the least fit individuals of a population should survive on to the next generation. This concept guaranteed preservation of a certain amount of diversity in the gene pool of a population, and improved the performance of the GA. While the GA outperforms the exhaustive search method for a low text to keyword ratio, it is not guaranteed to outperform an exhaustive search for extremely low ratios. As the text to keyword ratio decreases it becomes increasingly more difficult for the GA to decipher the code, eventually reaching a point at which an exhaustive search will outperform the Genetic
Algorithm. The GA is not guaranteed to arrive at a solution at all. The result has shown that the Genetic Algorithm decodes the Vigenère code exceptionally well in situations in which traditional methods of code breaking are inefficient.

Wilson and Mario Garcia (2006) proposed, “A Modified Version of the Vigenère Algorithm”, by adding a few bits of random padding to each byte; one can diffuse the statistical retentiveness found within most messages. The exact quantity of pad will be determined by a one way function, in an effort to eliminate the distinguishably of the message bits from the padded random bits. This methodology moderately increases the size of the cipher text, but greatly increases the security of the cipher

2.5 VERNAM CIPHER

Meier and Staffelbach (1989) proposed a Fast Correlation Attack (FCA) for cryptanalysis of stream cipher, this uses a probability vector and Bayesian bit-by-bit error correction in order to recover the initial linear feedback shift register (LFSR) contents. The attack uses a probabilistic model of the stream cipher. The fast correlation attack also requires a number of parity checks based on the feedback polynomial $f(x)$.

Feng-Tse Lin and Cheng-Yan Kao (1995) proposed an approach to cryptanalysis Vernam cipher by the use of Genetic Algorithm, the fitness value which is used depends on constructed dictionary of common English trigrams plaintext. The result has shown that method was able to guess some of the plaintext.
2.6 TRANSPOSITION CIPHER

Matthews (1993) used an order-based Genetic Algorithm to cryptanalysis a simple transposition cipher. The Genetic Algorithm system used to attack the transposition incorporates all the standard Genetic Algorithm features, the fitness scoring system based on the number of times common English bigrams and trigrams appear in the decrypted text. Ten bigrams and trigrams were used; they are shown in Table 2.1.

**Table 2.1 The fitness weight proposed by Matthews**

<table>
<thead>
<tr>
<th>Bi/trigram</th>
<th>Score</th>
<th>Bi/trigram</th>
<th>Score</th>
</tr>
</thead>
<tbody>
<tr>
<td>TH</td>
<td>+2</td>
<td>ED</td>
<td>+1</td>
</tr>
<tr>
<td>HE</td>
<td>+1</td>
<td>THE</td>
<td>+5</td>
</tr>
<tr>
<td>IN</td>
<td>+1</td>
<td>ING</td>
<td>+5</td>
</tr>
<tr>
<td>ER</td>
<td>+1</td>
<td>AND</td>
<td>+5</td>
</tr>
<tr>
<td>AN</td>
<td>+1</td>
<td>EEE</td>
<td>-5</td>
</tr>
</tbody>
</table>

The result has shown that transposition ciphers can be cryptanalysis using genetic algorithm and concluded that by using different mutation and crossover rate combinations may give better results.

Dimovski and Gligoroski (2003b) proposed three optimization heuristics in attacks on the transposition cipher. These heuristics are Simulated Annealing, Genetic Algorithm and Tabu Search. The assessment function used in Genetic Algorithm was based on bigram statistics only to reduce the complexity from $O(N^3)$ (where N is the alphabet size) when
trigram statistics are being determined to $O(N^2)$ when bigrams are the largest statistics being used.

Russel et al (2003b) proposed a solver based on an Ant Colony Optimization (ACO) algorithm to cryptanalysis transposition cipher. Two heuristics are used: one for recognizing plaintext using a dictionary, Dict. and another for indicating adjacent columns using bigrams, $\text{Adj}_{(I;J)}$. The result has shown that the cryptanalysis of transposition ciphers can be completely automated in a ciphertext-only attack.

2.7 GENETIC ALGORITHM WITH OTHER CIPHERS

Albassal and Wahdan (2003) presented a method to discover the key of a Substitution Permutation Network (SPN) using Genetic Algorithms, to measure the fitness based on the differential characteristics of the SPN. The result has shown that the complexity of the proposed attack is less than half of normal differential cryptanalysis of the same SPN.

Yaseen and Sahasrabuddhe (1999) developed a genetic algorithm as a method for cryptanalysing the Chor-Rivest knapsack Public Key Cryptosystem (PKC), the results showed how the algorithm is effectively used to break this scheme by examining a very small fraction of the space of possible solutions. The algorithm found the exact solution in all attempted cases.

Nalini and Raghavendra Rao (2005) presented three optimization heuristics, Simulated Annealing, Tabu Search and Genetic Algorithm for the cryptanalysis of Simplified Data Encryption Standard (SDES). Results have
shown a good performance for Tabu Search and Simulated Annealing, and comparison with Genetic Algorithms are also presented.

Clark and Dawson (1998) presented three optimization heuristics in automated attacks on a simple classical cipher. The three optimization heuristics considered are Simulated Annealing, the Genetic Algorithm and the Tabu Search. Efficiency and speed are investigated as performance criteria. The research has shown that the attack on small amounts of known ciphertext is more effective with the use of bigrams in cost function of genetic algorithm than using trigrams. Also it is concluded that trigrams are generally the most effective basis for a cost function used in attacks on the substitution cipher.

Ho Yean Li et al (2005) examined the effects of weakly chosen password-keys on the security of block ciphers. A new hybrid optimization heuristic cryptanalytic attack (Tabu Search and Genetic Algorithm) was used to conduct an intelligent key-search attack on classical ciphers and modern ciphers. The algorithm chosen to represent modern block ciphers is the Advanced Encryption Standard (AES) algorithm. AES is an algebraic product cipher which combines the elements of substitution and transposition. Therefore, the primary aims of this paper is to study the effects of an optimization heuristic cryptanalytic attack on block cipher. Generally, the Genetic Algorithm attack proved to be most efficient against the transposition cipher. The attack succeeded in recovering the original plaintext message in less than an hour for each trial run. It was also observed that the Genetic Algorithm attack produced weak results for the substitution cipher (the Hill Cipher) and the modem cipher (the AES product cipher). This is due to the fact that the original plaintext message may be recovered by using an alternative key with similar properties as the original encryption key on the transposition cipher, but never on the substitution cipher or on the product cipher. This is because of the confusion property inherent in both the
substitution cipher and the product cipher. Generally, the results suggested that a parallel implementation of the Genetic Algorithm would produce better results than the serial implementation done here. The results have shown that the transposition cipher (Columnar Transposition Cipher) is most susceptible to the Tabu Search and Genetic Algorithm attacks on weak passwords. This is followed by the Polygraphic Substitution Cipher (Hill Cipher), which is also vulnerable to the Tabu Search attack as well as the Genetic Algorithm attack, but at a greater time cost (provided the encryption key is a weakly chosen password). The product cipher (the AES cipher) is the most secure among the three. Unlike the other two, the product cipher is rather stable in terms of its vulnerability towards the optimization heuristic attacks. Nevertheless, the product cipher is still susceptible to weak password attacks by the average hacker or script kiddie using a basic personal computer system. This is especially obvious from the average 52% - 53% key search efficiency using the Tabu Search algorithm. In short, regardless of the strength and security of a cryptographic cipher, all categories of cipher algorithms are vulnerable to optimization heuristic attacks by a basic personal computer if the encryption key is a weakly chosen password.

Ya-Ping Zhang et al (2004) presented a new word-oriented stream cipher, RAINBOW, based on conventional encryption algorithms. The core of this algorithm is Key stream generator. What used to generate pseudo-random key stream is composed of "real key" and "temporal key". The real key" is just like key in block cipher, which is known by two users. But the "temporal key" is generated at the beginning of communication. The "real key" and "temporal key" are blended and divided into two parts. One part is taken as plaintext; the other is treated as key in conventional encryption algorithm (such as Triple DES, IDEA and so on). The output of encryption is a pseudo-random key stream, which is then XOR'ed with the plaintext to generate the cipher text. Because of diffusion and confusion of the
conventional encryption, the "real key", pseudo-random key streams, plain
text and ciphertext hold very complex and nonlinear relations. We have
performed several detailed security analysis. The cryptanalysis of RAINBOW
did not reveal an attack better than exhaustive key search. The Speed of this
algorithm is as fast as commonly block ciphers.

Albassall and Wahdan (2004) presented a method to discover the
key of a Feistel Cipher (FC) using Genetic Algorithms is described. A fitness
measure based on the differential characteristics of the FC is proposed and
complete problem formulation is described. The complexity of the proposed
attack is shown to be less than half of normal differential cryptanalysis of the
same FC. The work here is an extension to the previous work. The concept
introduced is proven and the work can be applied to other vulnerable ciphers
to differential cryptanalysis. The Genetic algorithm has been used as a
cryptanalysis tool for a basic FC and a complete problem formulation and a
proposal is described. The complexity of the new attack is less than $\frac{1}{2}$ the
differential attack.

The quality of reference works found in the literature varies widely;
the results of some references have not achieved good solution for their
problems.

No work has included the space character in Vigenere cipher and
no work has used Genetic Algorithm to find the expected key length.

Some results have to be improved so the motivation of this work to
achieve the better results.