**International Journal of Engineering Research and Development**
**e**-ISSN: 2278-067X, **p**-ISSN : 2278-800X, www.ijerd.com
Volume 5, Issue 3 (December 2012), PP. 07-12

# Comparative Cryptanalysis of Simplified-Data Encryption Standard Using Tabu Search and Simulated Annealing Methods

Rajashekarappa[1], Dr. K M Sunjiv Soyjaudah[2]
[1]Department of Computer Science and Engineering, JSS Academy of Technical Education, Vacoas, Mauritius.
(Research Scholar, Jain University, Bangalore)
[2]Department of Electrical & Electronic Engineering, University of Mauritius, Reduit, Mauritius.

**Abstract:-** This paper presents an approach for the comparative cryptanalysis of Simple Data Encryption Standard (S-DES) using Tabu Search and Simulated Annealing methods. In this paper, cipher text only attack is adopted and varieties of optimum keys are generated based on the cost function values. The goal of this paper is three fold. First we want to make a study about how evolutionary computation techniques can efficiently solve the NP-Hard combinatorial problem. For achieving this goal we test several evolutionary computation techniques like Tabu Search algorithm and simulated annealing for the cryptanalysis of simplified data encryption standard problem (SDES). Second was a comparison between Tabu Search algorithm and simulated annealing were made in order to investigate the performance for the cryptanalysis on SDES. The methods were tested and extensive computational results show that Tabu Search algorithm performs better than simulated annealing for such type of NP-Hard combinatorial problem. This paper represents our first effort toward efficient Tabu Search algorithm for the cryptanalysis of SDES. And third was cryptanalysis data could be store for long time with help of Self Monitoring Analysis and Reporting Technology (SMART) Copyback technique.

**Keywords:-** Cryptanalysis, Simplified Data Encryption Standard (S-DES), Tabu Search, Plain text, Cipher text, Simulated Annealing, Cipher text attack.

## I. INTRODUCTION

The cryptanalysis of simplified data encryption standard can be formulated as NP-Hard combinatorial problem. Solving such problems requires effort (e.g., time and/or memory requirement) which increases with the size of the problem. Techniques for solving combinatorial problems fall into two broad groups – traditional optimization techniques (exact algorithms) and nontraditional optimization techniques (approximate algorithms). A traditional optimization technique guarantees that the optimal solution to the problem will be found. The traditional optimization techniques like branch and bound, simplex method, Simulated Annealing search algorithm etc methodology is very inefficient for solving combinatorial problem because of their prohibitive complexity (time and memory requirement). Nontraditional optimization techniques are employed in an attempt to find an adequate solution to the problem. A nontraditional optimization technique – Tabu search algorithm, simulated annealing and tabu search were developed to provide a robust and efficient methodology for cryptanalysis. The aim of these techniques to find sufficient "good" solution efficiently with the characteristics of the problem, instead of the global optimum solution, and thus it also provides attractive alternative for the large scale applications. These nontraditional optimization techniques demonstrate good potential when applied in the field of cryptanalysis. The objective of the study is to determine the efficiency and accuracy of Tabu Search algorithm for the cryptanalysis of SDES. To compare the relative performance of Simulated Annealing with tabu search[1]. In this paper users have variety of requirements of data storage that can be addressed using Self Monitoring Analysis and Reporting Technology (SMART) Copyback.

The rest of the paper is organized as follows: Section 2 presents the literature review. Section 3 gives a brief overview of S-DES, Section 4 gives the overview of Tabu Search, Section 5 gives the algorithm of Simulated Annealing and Section 6 gives the Self Monitoring Analysis and Reporting Technology (SMART) Copyback. Experimental results are discussed in Section 7. Section 8 Concludes the paper and Future works.

## II. RELATED WORK

If a particular solution is reached it becomes 'tabu' for some number of transitions, generally referred to as the solution's tabu tenure. If a solution is tabu, the search is normally prevented from moving to that

solution, i.e., the local neighborhood from which the next solution is chosen excludes those solutions that are currently tabu. Conceptually, the currently tabu solutions together with their remaining tabu tenures form a 'tabu list'. In its simplest form, with common tabu tenure of the list becomes a FIFO queue. The most recently visited solution is added and the solution visited moves ago is removed. The tabu list implements a recency criterion. It prevents the search revisiting solutions in the short term and so short cycles are prevented. In Garg applied an attack on transposition cipher using tabu search & simulated annealing [1].

## III. THE S-DES ALGORITHM

This section briefly gives the overview of S-DES Algorithm. The SDES encryption algorithm takes an 8-bit block of plaintext and a 10-bit key as input and produces an 8-bit block of ciphertext as output. The decryption algorithm takes an 8-bit block of ciphertext and the same 10-bit key used for encryption as input and produces the original 8-bit block of plaintext as output. The encryption algorithm uses five basic functions: 1. An initial permutation (IP). 2. A complex function called fK which involves both permutation and substitution operations and depends on a key input 3. A simple permutation function (SW) that switches the two halves of the data. 4. The function fK a TS in and 5. A permutation function that is the inverse of the initial permutation ($IP^{-1}$). The function fK takes as input the data passing through the encryption algorithm and an 8-bit key [4].

### A. Key Generation

For key generation, a 10-bit key is considered from which two 8-bit sub keys are generated. In this case, the Key is first subjected to a permutation P10= [3 5 2 7 4 10 1 9 8 6], then a shift operation is performed. The numbers in the array represent the value of that bit in the original10-bit key. The output of the shift operation then passes through a permutation function that produces an 8-bit output P8 = [6 3 7 4 8 5 10 9] for the first sub key (K1). The output of the shift operation also feeds into another shift operation and another instance of P8 to produce the second sub key K2. In all bit strings, the leftmost position corresponds to the first bit.

### B. Encryption Algorithm

The block schematic of the SDES encryption algorithm is the Encryption process involves the sequential application of five functions:

**1. Initial and final permutation (IP):** The input to the algorithm is an 8-bit block of plaintext, which is first permuted using the IP function IP = [2 6 3 1 4 8 5 7]. This retains all 8-bits of the plaintext but mixes them up. At the end of the algorithm, the inverse permutation is applied; the inverse permutation is done by applying, $IP^{-1}$ = [4 1 3 5 7 2 8 6] Where, $IP^{-1}(IP(X)) = X$.

**2. Function fK:** The function fK, which is the complex component of S-DES, consists of a combination of permutation and substitution functions. The functions are given as follows:
fK (L, R) = (L XOR f(R, key), R)where, L, R be the left 4-bits and right 4-bits of the input, XOR is the exclusive-OR operation and key is a sub -key. Computation of f(R, key) is done as follows.
i. Apply expansion/permutation E/P= [4 1 2 3 2 3 4 1] to input 4-bits.
ii. Add the 8-bit key (XOR).
iii. Pass the left 4-bits through S-Box S0 and the right 4-bits through S-Box S1.
iv. Apply permutation P4 = [2 4 3 1].
The two S-boxes are defined as follows:

| S0 | S1 |
|---|---|
| 1032 | 0123 |
| 3210 | 2013 |
| 0213 | 3010 |
| 3132 | 2103 |

The S-boxes operate as follows: The first and fourth input bits are treated as 2-bit numbers that specify a row of the S-box and the second and third input bits specify a column of the S box. The entry in that row and column in base 2 is the 2-bit output.

**3. The Switch Function (SW):** Since the function fK allows only the leftmost 4-bits of the input, the switch function (SW) interchanges the left and right 4-bits so that the second instance of fK operates on different 4-bits. In this second instance, the E/P, S0, S1 and P4 functions are the same as above but the key input is K2 [6].

## IV. TABU SEARCH

This section presents application of tabu search includes as a subroutine a local search procedure that seems appropriate for the problem being addressed [2]. A local search procedure operates just like a local improvement procedure except that it may not require that each new trial solution must be better than the preceding trial solution [10]. The process begins by using this procedure as a local improvement procedure in the usual way (i.e, only accepting an improved solution at each iteration) to find a local optimum [3]. A key strategy of tabu search is that it then continues the search by allowing non-improving moves to the best solutions in the neighborhood of the local optimum [8]. Once a point is reached where better solutions can be found in the neighborhood of the current trial solution, the local improvement procedure is reapplied to find a new local optimum [6]. This use of memory to guide the search by using tabu lists to record some of the recent history of the search is a distinctive feature of tabu search. This feature has roots in the field of artificial intelligence [7]. Initialization: Start with a feasible initial trial solution. Iteration: Use an appropriate local search procedure to define the feasible moves into the local neighborhood of the current trial solution [5]. Eliminate from consideration any move on the current tabu list unless that move would result in a better solution than the best trial solution found so far. Update the tabu list to forbid cycling back to what had been the current trial solution. If the tabu list already had been full, delete the oldest member of the tabu list to provide more flexibility for future moves. Two randomly chosen key elements are swapped to generate candidate solutions. In each iteration, the best new key formed replaces the worst existing one in the tabu list [9].

The algorithm is presented as below.

1. Input: Intercepted ciphertext, the key size P, and the language statistics.
2. Initialise parameters: The size of the tabu list STABU, the size of the list of possibilities considered in each iteration SPOSS, and the maximum number of iterations MAX.
3. Initialise the tabu list with random and distinct keys and calculate the cost for each key in the tabu list.
4. For I =1,… , MAX do:

a. Find the best key with the lowest cost in the current tabulist, $K_{BEST}$.

b. For j=1,…, SPOSS do:

i. apply the perturbation mechanism to produce a new key $K_{NEW}$.

ii. Check if $K_{NEW}$ is already in the list of possibilities generated for this iteration or the tabu list. If so, return to step 4(b) i.

iii. Add $K_{NEW}$ to the list of possibilities for this iteration.

c. From the list of possibilities for this iteration, find the key with the lowest cost, $P_{BEST}$.

d. From the tabu list, find the key with the highest cost, $T_{WORST}$.

e. While the cost of $P_{BEST}$ is less than the cost of $T_{WORST}$:

i. Replace $T_{WORST}$ with $P_{BEST}$.

ii. Find the new $P_{BEST}$.

iii. Find the new $T_{WORST}$.

5. Output the best solution from the tabu list, $K_{BEST}$(the one with the least cost).

An important distinction in TS arises by differentiating between short term memory and longer term memory. Each type of memory is accompanied by its own special strategies. However, the effect of both types of memory may be viewed as modifying the neighbourhood $\mathbf{N}(x)$ of the current solution $x$. The modified neighborhood, which we denote by $\mathbf{N}^*(x)$, is the result of maintaining a selective history of the states encountered during the search.

In the TS strategies based on short term considerations, $\mathbf{N}^*(x)$ characteristically is a subset of $\mathbf{N}(x)$, and the tabu classification serves to identify elements of $\mathbf{N}(x)$ excluded from $\mathbf{N}^*(x)$. In TS strategies that include longer term considerations, $\mathbf{N}^*(x)$ may also be expanded to include solutions not ordinarily found in $\mathbf{N}(x)$. Characterized in this way, TS may be viewed as a dynamic neighborhood method. This means that the neighborhood of $x$ is not a static set, but rather a set that can change according to the history of the search. This feature of a dynamically changing neighborhood also applies to the consideration of selecting different component neighborhoods from a *compound* neighborhood that encompasses multiple types or levels of moves, and provides an important basis for parallel processing. Characteristically, aTS process based strictly on short term strategies may allow a solution $x$ to be visited more than once, but it is likely that the corresponding reduced neighborhood $\mathbf{N}^*(x)$ will be different each time. With the inclusion of longer term considerations, the likelihood of duplicating a previous neighborhood upon revisiting a solution, and more generally of making choices that repeatedly visit only a limited subset of $\mathbf{X}$, is all but nonexistent. From a practical stand point, the method will characteristically identify an optimal or near optimal solution long before a substantial portion of $\mathbf{X}$ is examined. A crucial aspect of TS involves the choice of an appropriate definition of $\mathbf{N}^*(x)$. Due to the exploitation of memory, $\mathbf{N}^*(x)$ depends upon the trajectory followed in moving from one solution to the next (or upon a collection of such trajectories in a parallel processing environment).

## V. SIMULATED ANNEALING

In this paper annealing is the process of slowly cooling a heated metal in order to attain a minimum energy state. The idea of mimicking the annealing process has been efficiently exploited by Kirkpatrick et al.[11] to solve combinatorial optimization problems. The algorithm is initialized with a random solution to the problem being solved and a starting temperature T0. The temperature is slowly decreased and at each temperature, a number of attempts are made to perturb the current solution. At each perturbed temperature, a change in the cost function ΔE is determined. If ΔE<0, then the proposed perturbation is accepted; otherwise it is accepted with a probability indicated by the Metropolis equation given by Probability (E1→E2) = $e^{(-\Delta E/T)}$, where E1 and E2 are the cost functions, ΔE is the change in cost function and T is the current temperature. If the proposed change is accepted, then the current solution is updated. The temperature is reduced when a predefined number of attempts have been made to update the current solution. Possibilities of termination are when a certain minimum temperature is reached or a certain number of temperature reductions have occurred; or the current solution has not changed for a number of iterations. The Simulated Annealing algorithm is presented as below:

1. Set the initial temperature, $T^{(0)}$ .
2. Generate an initial solution - arbitrarily set to the identity transformation (could be randomly generated or otherwise).
3. Evaluate the cost function for the initial solution. Call this $C^{(0)}$.
4. For temperate T do many (eg., 100 × M) times:

Generate a new solution by modifying the current one in some manner evaluate the cost function for the newly proposed solution. Consult the Metropolis function to decide whether or not the newly proposed solution will be accepted. If accepted, update the current solution and its associated cost. If the number of accepted transitions for temperature T exceeds some limit (eg. 10 × M) then jump to Step 5.

5. If the number of accepted transitions for temperature T was zero then stop (return the current solution as the best), otherwise reduce the temperature (eg. $T^{(i+1)=} T^{i} \times 0.95$) and return to step 4.

## VI. SELF MONITORING ANALYSIS AND REPORTING TECHNOLOGY(SMART) COPYBACK

In this paper users have variety of requirements of data storage that can be addressed using Self Monitoring Analysis and Reporting Technology (SMART) Copyback [12]. It is estimated that over 94% of all new information produced in the world is being stored on magnetic media, most of it on Physical Disks (PD). Despite their importance, there is relatively little published work on the failure patterns of Physical Disks (PD), and the key factors that affect their lifetime. Most available data are either based on extrapolation from accelerated aging experiments or from relatively modest sized field studies. Moreover, larger population studies rarely have the infrastructure in place to collect health signals from components in operation, which is critical information for detailed failure analysis. It presents the data collected from detailed observations of a large disk drive population in a production Internet services deployment. The population observed is many times larger than that of previous studies. In addition to presenting failure statistics and analyze the correlation between failures and several parameters generally believed to impact longevity. Analysis identifies several parameters from the Physical Disks (PD), self monitoring facility (SMART) that correlate highly with failures. Despite this high correlation conclude that models based on SMART parameters alone are unlikely to be useful for predicting individual drive failures. Surprisingly, it found that temperature and activity levels were much less correlated with Physical Disk (PD) failures

In this study we report on the failure characteristics of consumer-grade Physical Disks (PD). Analysis is made possible by a new highly parallel health data collection and analysis infrastructure, and by the sheer size of our computing deployment.

Our results confirm the findings of some of the SMART Copyback parameters are well-correlated with higher failure probabilities. First errors in reallocation, offline reallocation, and probational counts are also strongly correlated to higher failure probabilities. Despite those strong correlations, we find that failure prediction models based on SMART Copyback parameters alone are a likely to be severely limited in their prediction accuracy, given that a large fraction of our failed Physical Disks have shown no SMART error signals whatsoever. This results suggests that SMART Copyback models are more useful in predicting trends for large aggregate populations than for individual components. It also suggests that powerful predictive models need to make use of signals beyond those provided by SMART Copyback. In this thesis we report on the failure characteristics of consumer-grade Physical Disks (PD). The drive vendors builds a logic in to the drives to make drives smart so that the user gets warning signal as a “predictive failure” whenever the drive is about to go bad for some reason. The drives built with this kind of logic are called a “SMART” drive which is an acronym for

"Self-Monitoring Analysis and Reporting Technology". Using this technique and make use of cryptanalysis data keeping for long time without fail.

## VII. PERFORMANCE COMPARISION OF TABU SEARCH AND SIMULATED ANNEALING

This section presents performance and comparison among Tabu Search algorithm is recovered from the key as compared with Simulated Annealing as shown in Table 1. A number of experiments is carried out to outline the effectiveness of Tabu Search. The Tabu Search algorithm is coded in MATLAB 7, and tested on more than sixty different ciphertext. Among the unigrams, bigrams and trigrams, Unigram is more useful and the benefit of trigram over digram is small. For S-DES, the key size required is 10bits. The plain text and cipher text size are 8 bits. Initially, start with a feasible initial trial solution size of 40 is taken with a chromosome size of 10 bits i.e., 40 sets of 10 bit keys are taken randomly and the known cipher text is decrypted using the initial keys. The results are promising, yielding solutions between 60 and 112 seconds on a standard Intel Pentium Desktop computer with a 2.93GHz processor and 2GB of RAM. Four independent runs are executed for each of the 60 problems.

Initialization 40, Chromosome Size 10 bits, No. of Iteration 10, Mating Scheme Best-Worst Mating. The total number of generations taken is 10. The key was recovered on an average of 5 generations. The size of the search space used by TS is only 40 where as in Simulated Annealing attack, it is 65. Thus, in case of cryptanalysis using TS there is reduction in search space by a factor of 4 approximately. In the worst case the number of generations is increased to 15.

In this section a number of experiments are carried out which outlines the effectiveness of both the algorithm described above. The purpose of these experiments is to compare the performance of Simulated Annealing algorithm approach with tabu search approach for the cryptanalysis of simplified SDES algorithm. The experiments were implemented in MATLAB 7 on a Pentium IV(2.93 GHz). Experimental results obtained from these algorithms were generated with 150 runs per data point e.g. ten different messages were created for both the algorithms and each algorithm was run 20 times per message. The best result for each message was averaged to produce data point.

This paper shows the average number of key elements (out of 10) correctly recovered versus the amount of cipher text and the computation time to recover the keys from the search space. The experimental result shows for amounts of cipher text ranging from 100 to 1000 character. Tabu Search results are better than the Simulated Annealing results. The amount of Cipher text was 1000 and Tabu Search took ten minutes to recovered from the number of bits matched in the key was 9(out of 10), but in Simulated Annealing took fifteen minutes to recovered from the number of bits matched in the key was 7(out of 10). Tabu Search can be extended to attack DES which uses 64 bit key size.

## VIII. CONCLUSIONS

In this paper has demonstrated that the tabu search and simulated annealing are ideally suited for the cryptanalysis of Simplified Data Encryption Standard. Thus these techniques offer a lot of promises for attacks of the ciphers. The time complexity of the proposed approach has been reduced drastically when compared to the Simulated Annealing Algorithm. Experimental results demonstrate good performance for tabu search than simulated annealing few parameters need to be tuned for the best possible performance. Though SDES is a simple encryption algorithm, this is a promising method and can be adopted to handle other complex block ciphers like DES and AES. The cost function values used here can be applied for other block ciphers also. The future works are extending this approach for attacking DES and AES ciphers. Cryptanalysis data could be store for long time with help of Self Monitoring Analysis and Reporting Technology (SMART) Copyback technique.

## REFERENCES

[1]. Garg Poonam, Shastri Aditya, Agarwal D. C, "Genetic Algorithm, Tabu Search & Simulated annealing Attack on Transposition Cipher", proceeding of Third AIMS International conference on management at IIMA – 2006, pg 983-989.

[2]. Frederick S. Hillier, Gerald J. Lieberman, "Introduction to Operations Research Concepts and Cases", Eighth edition, McGraw- Hill, 2009.

[3]. William Stallings, "Cryptography and Network Security Principles and Practices", Fourth edition, McGraw- Hill, 2003.M. Young, The Technical Writer's Handbook. Mill Valley, CA: University Science, 1989.

[4]. Behrouz A. Forouzan, "Cryptography and Network Security", Firstedition, McGraw- Hill, 2006

[5]. James Kennedy and Russell Eberhart, "Particle Swarm Optimisation", Proceedings of the IEEE International Conference on Neural Networks,pp.1942-1948, 1995.

[6]. Chanas S. and P. Kobylanski, "A New Heuristic Algorithm Solving the Linear Ordering Problem", Computational Optimization and Applications, Vol. 6, pp. 191-205, 1996.
[7]. Derisi A. Donateo, D. Paolo, C. Ficarella A, "A combined optimization method for common rail Diesel engines", Proceedings of 2002 Spring Technical Conference of the ASME Internal Combustion Engine Division, Rockford Illinois, 2002.
[8]. Lan Sommerville, "Software Engineering", Sixth Edition, Pearson Education Asia,2006.
[9]. Atul Kahate, " Cryptography and Network Security", TMH, 2003.
[10]. Glover F, "Tabu Search — Part I", ORSA Journal on Computing 2: 1, 4-32, 1990.
[11]. Kirkpatrick S, C. D. Gelatt. Jr. and Vecchi M. P, "Optimisation by Simulated Annealing", Science, Vol. 220, No. 4598, pp.671-680,1983.
[12]. Rajashekarappa and Dr. K M S Soyjaudah "Self Monitoring Analysis and Reporting Technology (SMART) Copyback", proceedings of ICIP 2011, pp 463 - 469, 2011. © Springer-Verlag Berlin Heidelberg 2011.

Mr. Rajashekarappa is a Lecturer since July 2010 in the Department of Computer Science and Engineering, JSS Academy of Technical Education, Avenue Droopnath Ramphul, Bonne Terre, Vacoas, Mauritius. He has one and half years of experienced as a Project Assistant at Indian Institute of Science (IISc), Bangalore, India. He worked as Project Internee in Indian Space Research Organization (ISRO), Bangalore, India. He has one and half years of experienced as a Project Trainee at LSI Technologies Pvt. Ltd, Bangalore, India.

Mr. Rajashekarappa obtained his Bachelor of Engineering in Computer Science and Engineering from Anjuman Engineering College, Bhatkal, India. He has qualified in Graduate Aptitude Test in Engineering (GATE), Computer Science and Engineering, 2006. He received his Master Degree in Computer Science and Engineering from R. V. College of Engineering, Bangalore, India. He is pursuing his Ph. D in Computer Science and Engineering at Jain University, Bangalore, India, under the guidance of Dr. K M Sunjiv Soyjaudah, University of Mauritius, Reduit, Mauritius. His area of interest and research include Cryptography, Data mining, Mobile Communication, Computer Networks and Cloud Computing. He has published several Research papers in international journal/conferences. He has guided many students of Bachelor degree in Computer Science and Engineering in their major projects. He is a member of ISTE, IETE, IACSIT, IAEST, IAENG and AIRCC.

Professor K M Sunjiv Soyjaudah received his B. Sc (Hons) degree in Physics from Queen Mary College, University of London in 1982, his M.Sc. Degree in Digital Electronics from King's College, University of London in 1991, and his Ph. D. degree in Digital Communications from University of Mauritius in 1998. He is presently Professor of Communications Engineering in the Department of Electrical and Electronic Engineering of the University of Mauritius. His current interest includes source and channel coding modulation, cryptography, voice and video through IP, as well as mobile communication. Professor K M S Soyjaudah is a member of the IEEE, Director in the Multicarrier (Mauritius), Technical Expert in the Energy Efficiency Management Office, Mauritius.