A Novel Method to Design S-box Based on Genetic Algorithm and Particle Swarm Optimization in AES-128 Cryptosystem

K. Kalaiselvi, Anand Kumar

1Bharathiar University, Department of Computer Science, Kristu Jayanti College, Bangalore. kalaiselvi76@yahoo.com
2Bharathiar University, Department of CSE, M.S Engineering College, Bangalore. kumaranandkumar@gmail.com

Abstract

Information security is a major concern in data communication, which needs to be provided through a strong cryptography method. Advanced Encryption Standard (AES) is proven to be a superior block cipher to provide confidentiality for the secret information. Substitution box (S-Box) forms the vital structure of AES in performing cryptographic operations which are resistant against possible attacks on the information. The design of suitably strong S-box with desirable properties to provide better outcome can be achieved using Evolutionary Algorithms (EA) which are metaheuristics optimization algorithms based on generic population. Genetic algorithm (GA) and Particle Swarm Optimization (PSO) belongs to EA, are used in this research work to design a key dependent S-Box for AES-128 bit cryptosystem. This research paper also compares the properties of S-box which is generated using hybrid EAs with that of the conventional AES-128 and evaluates the result. The performance evaluation shows that the S-box designed using this novel method is stronger in terms of non-linearity, strict avalanche effects, bits distribution and bijectivity.

Key Words: Cryptography, S-box, evolutionary algorithm, genetic algorithm, particle swarm optimization.
1. Introduction

The recent data communication and data transfer uses electronic transmission of text, audio and video information through a secured or insecure network channels. The integrity and security of the data transmitted is of much concerned since the lack of these leads to possible breach of confidentiality and disruption in the service provided. The demand towards strengthening the security has been increased and the need to achieve the privacy during data exchange has grown tremendously [2]. These integral characteristic of network security can be achieved by cryptographic mechanisms. Many block ciphers [10] and stream ciphers such as DES, AES, RSA have been developed for the purpose to provide secured data transfer. Cryptographic algorithms can be either symmetrical wherein the same key is used by both the sender and the receiver to encrypt and decrypt the data or asymmetrical as different keys are used by both the parties in communication for the cryptographic purpose. A strong cryptosystem is needed for uncompromised key generation. The strength of AES cryptosystem [7] is centric to the S-box which is constructed with generated key. The performance properties of S-box, Bijection, Non linearity, Avalanche effects and Bit independence [2] should increase the confusion and diffusion property of cryptosystems [4].

S-boxes are constructed with a predetermined key. These static S-boxes are susceptible to differential cryptanalytic and linear cryptanalytic attacks [8]. Key dependent S-boxes are stronger, since the strength of S-box depends on the generated symmetric key [8]. This research work proposes a cryptographically strong AES-128 bit algorithm with improved S-box characteristics using hybrid combination of GA and PSO. The key dependent S-boxes are generated and the optimized S-box is generated in comparison with the previous generation S-box [11][13][17]. The main intention of using GA and PSO in designing the S-box is to generate finite number of S-boxes through the mutation, crossover and selection operators. The S-box with high non-linearity, maximum bit independence, avalanche criteria and bijection is selected for encryption process.

The remaining part of this research paper is organized as follows. Section 2 discusses the evaluation criteria for the S-box. Section 3 gives an overview of proposed method using GA and PSO. Section 4 explains the method of generating key dependent S-box using GA and PSO followed by section 5 with performance evaluation of the generated S-box with respect to the existing AES-128. Conclusion and references are charted out in section 6 and 7 respectively.
2. Cryptographic Properties of S-boxes

Substitution Box (S-BOX)

In AES-128 algorithm [9], S-box performs the key non-linear transformation which substitutes the set of input vector with a different set of output vector. If S-Box is denoted by $\pi$, then:

$$\pi_s: \{0,1\}^n \rightarrow \{0,1\}^m$$

where $n$ is number of input bits for S-Box and $m$ is the number of output bits to have $n \times m$ S-box [4][13].

The strength of AES-128 bit algorithm depends on the strong cryptographic symmetric key and the S-box with desirable cryptographic characteristics [8]. This section presents an overview of the desired characteristics of S-box such as Non-linearity, bijectivity, strict avalanche criteria and bits independence which has an impact on the strength of cryptosystem itself. These characteristics are explained as follows [2]:

**Bijectivity Property**

An S-box of $n \times n$ size is said to be bijective in nature, if it has all possible output values from interval $[0,2^n-1]$. If $f_i (1 \leq i \leq n)$ is a Boolean function of an S-box, it satisfy

$$wt \left( \sum_{i=1}^{n} a_i f_i \right) = 2^{n-1}$$ (1)

Where $a_i \in \{0,1\}$, $\{a_1,a_2,..a_n\} \neq (0,0,...0)$ and $wt$ is the Hamming weight which indicates number of 1’s in a given vector. $f_i$ requires to be either 0 or 1.

**Non-Linearity**

An S-box is strong if it has the Boolean function with high non-linearity value[15]. The non-linearity $N_f$ of a Boolean function $f(x)$ is calculated as

$$N_f = 2^{n-1} \left( 1 - 2^{-n} \max |S_f(\omega)| \right)$$ (2)

Where $f(x)$ if defined as

$$S_f(\omega) = \sum_{\omega \in GF(2^n)} (-1)^{f(x) + x \cdot \omega}$$ (3)

Where $\omega \in GF(2^n)$ and $x$, $\omega$ denotes the dot product of $x$ and $\omega$

**Strict Avalanche Criteria**

If a Boolean function satisfies the Strict Avalanche Criteria (SAC), half of the output bits should change when there is a change in one bit. Any change in the input vector will have a significant change in the output vector with a probability of $\frac{1}{2}$.

**Bit Independence**

Bit Independence criteria state that the output bit should be independent from each other without any statistical dependencies among bits in the output vector.
The avalanche variables should be pair-wise independent for a given set of avalanche vectors generated by complementing a single plaintext bit. The degree of independence between the paired variables can be calculated using correlation coefficient. If the variables are taken as A and B

\[ \rho \{A,B\} = \frac{\text{cov} \{A,B\}}{\sigma_A \sigma_B} \]  

where \( \rho \{A,B\} \) is the correlation coefficient of A and B, the numerator is the covariance of A and B. If the Boolean functions in the S-box are \( f_1, f_2, \ldots \) and if the S-box meets the criteria,

\[ f_j \text{ XOR } f_k \; (j \neq k, 1 \leq j, k \leq n) \]  

is considered as highly non-linear.

The integral part of the block cipher is a Substitution box [10]. To protect the cipher against the linear and differential cryptanalysis, the confusion of bits should be higher which is provided by the non-linear transformation. This substitution converts the plaintext into cipher text using the symmetrical key in AES-128. A strong S-box with high non-linearity and more avalanche effects is needed to have an enhanced cryptosystem [1][11].

3. The Proposed Method for S-box Design

The proposed research work to construct a key dependent S-box is based on the Evolutionary Algorithms: Genetic Algorithm (GA) and Particle Swarm Optimization (PSO). The basic working principles of GA and PSO are explained in this section [3].

**Genetic Algorithm**

Genetic algorithm (GA) is a well-known and frequently used evolutionary computation technique. The idea was inspired from Darwin’s natural selection theorem which is based on the idea of the survival of the fittest. The GA is inspired by the principles of genetics and evolution, and mimics the reproduction behavior observed in biological populations. In GA, a candidate solution for a specific problem is called as individual or a chromosome and consists of a linear list of genes. Genetic algorithm begins its search from a randomly generated population of designs that evolve over successive iterations, eliminating the need for a user-supplied starting point [16].

To perform its key generation-like process, the Genetic algorithm employs three operators to propagate its population from one generation to another. Selection operator in which the GA considers the principle of “survival of the fittest” to select and generate individual design solutions that are adapted to their environment. The crossover operator mimics mating in the initial population. The crossover operator propagates features of good design solutions from the current population into the future population, which will have a better fitness value on average. Mutation, promotes diversity in population characteristics. The mutation operator allows for global search of the design space and prevents...
the algorithm from getting trapped in local minima.

**Particle Swarm Optimization**

PSO and evolutionary computation techniques such as GA have much functionality resemblance between them. The optimization is achieved by initializing the system by generating a population of random solutions. But like GA, PSO does not have evolution operators such as crossover and mutation. The candidate solutions called as particles move through the problem space by following the current particles. Each particle is considered as the potential solution to the optimization problem. The best position and the best particle in that position are determined by the particles in the neighborhood. Each time the particle visits a new position, the previous position is remembered and compared with that of the new position. Each particle movement is determined by its initial random velocity and two random weights namely: local best position and the best position in the search space.

The fitness of the particles is evaluated using the fitness function to optimize the value. Each particle may change its position, and consequently may explore the problem space, simply varying its associated velocity. The main PSO operator is the velocity update, which considers the best position, in terms of fitness value reached by all the particles during their paths, and the best position that the agent itself has reached during its search, resulting in a migration of the entire swarm towards the global optimum. Each iteration, the particle moves around according to its velocity and position; the cost function to be optimized is evaluated for each particle to rank the current S-Box Generation. PSO search for the optimal solutions by learning from the individual particle and the group in the swarm. The optimal solution is updated by regulating the parameters and positioning the random variables.

**Design of S-BOX Using GA and PSO in AES-128**

The proposed methodology exploits the features of both GA and PSO to design S-boxes which are key dependent [14]. The structure of AES-128 is considered for the cryptographic operations. The well-defined characteristic of S-box confusion and diffusion are performed by the operators of GA and PSO[7][9][11]. The symmetric key is generated using GA [5] which is an optimized key as the result of Gap test, Frequency test and Autocorrelation test. The block diagram of an enhanced cryptosystem AES-128 bit using GA and PSO is shown in the Figure:1 [2][3][16].
This algorithm comprises of effective function such as initial population with random number generation, selection, crossover, and mutation. An elitist scheme has been employed in this research work, where the best solution gives the chance to be directly carried over to the next generation. In each generation, the fitness values of all the individuals in the population are estimated subsequently, the top-half best executed ones are noticed. These individuals are considered elites. Alternatively reproducing the elites directly to the next generation as elite GAs does, we first increase the elites. The improvement that makes elites more agreeable operation, which tries to mimic the grow phenomenon in nature, where individuals will become desirable to the environment after gain knowledge from the society. Moreover, by using these improved elites as parents, the generated offspring will achieve better performance than those cover by master elites. The improvement of the elites is executed by the velocity and position update procedures in PSO. The stability condition in PSO is defined by

\[
\frac{C_1 r_1 + C_2 r_2}{2} - \phi < 1 \\
\phi < 1
\]

(5)
Knowing that $\theta \in \{0, 1\}$, the specified parameter selection generalization was derived

$$0 < C_1 + C_2 < 4$$
$$\frac{C_1 + C_2}{2} - 1 < \theta < 1$$

(6)

Parameters $C_1$, $C_2$ and $\omega$ are chosen considering the above conditions and according to the finding ability needed to solve each of the problems. The elites in every considered aspect, individualistic generation can be from PSO and GA groups of the parameter selection generalization generation, i.e., the improved elites.

Selection

The GA functioning is executed on the improved elites accomplished by PSO. With the objective of parents’ selection in the crossover process, the s-Box selection scheme is used. Two enhanced elites are selected indiscriminately, and their fitness values are examined and note the similarities or differences. Finally, select the one with superior fitness as a parent and place it in the mating pool.

Crossover

Parents are selected in a random manner from the mating pool in groups of two and two offspring are created by performing crossover on the parent solutions. Parent solutions can be controlled using a distributed index, $c_{\eta}$. With accompanying distribution index operator any arbitrary contiguous region can be explored, provided there is as much as necessary diversity maintained among the feasible parent solutions.

$$c_1 = x - \frac{1}{2} \beta \left[ x - p_1 \right]$$
$$c_2 = x + \frac{1}{2} \beta \left[ x - p_1 \right]$$

(7)

where $x = \frac{1}{2} \beta (p_2 + p_1)$, $p > p_1, c = x \beta_{ss}$ output of the hidden node

Mutation

A mutation can produce a new genetic material in the population to maintain the population’s diversity. Mutation is not applied to all of the population, and a mutation probability $P_m$ is assigned to every individual according to its fitness value:

$$P_{m_i} = \begin{cases} 0.5 \left[ \frac{F_{\max} - F_i}{F_{\max} - F_{\text{ave}}} \right] & \text{if } F_i \geq F_{\text{ave}} \\ \frac{F_{\text{ave}} - F_i}{F_{\max} - F_{\text{ave}}} & \text{if } F_i < F_{\text{ave}} \end{cases}$$

(8)

$F_i$ is the fitness value of the individual $i$, $F_{\max}$ and $F_{\text{ave}}$ are the maximum and average fitness values of the population in each generation. After assigning the
$P_m$, a random number in the range $[0, 1]$ is created for each generation. The individuals having $aP_m$ greater than this number are mutated. The mutation operator employed in accomplishing a variable dependent random mutation. In the random mutation operator, a solution is produced in the vicinity of the parent solution with a uniform probability distribution

$$x_{i+1}^{(k)} = x_{i}^{(k)} + (r - 0.5)\Delta_i$$

(9)

$r_i$ is a random number in $[0, 1]$. $\Delta_i$ is the user defined maximum perturbation allowed in the $i$th decision variable ($x_i$). At each generation $\Delta_i$ for a variable $x_i$ is calculated using the average of that variable or the difference between its maximum and minimum in the population, i.e.

$$\Delta_i = 0.5 \times \left( \max_{x_i} - \min_{x_i} \right)$$

$$\Delta_i = 0.025 - 0.075 \frac{\text{ave}_{x_i}}{2}$$

(10)

After applying the GA operators, the offspring and the improved elites from PSO, form the new population and their fitness is evaluated and compared in order to select the elites for the next generation.

4. **Proposed S-box Generation Algorithm Using GA and PSO for AES-128**

A novel cryptographic encryption algorithm has been proposed, considering the base as AES-128 encryption algorithm, an enhanced AES symmetric algorithm is integrated with the Genetic Algorithm to improve the complexity of the cryptosystem. The proposed algorithm is implemented by taking combination of keys from which a key-dependent S-box is generated dynamically. A set of key is generated by the Genetic algorithm based key Generation Algorithm [5] and the best keys are chosen by the application of the fitness function. The best fitting key is identified by the implementation of the Gap test and Frequency Test to the set of keys. The chosen best key is used to create an S-Box for AES encryption [7][9][11]. Here the S-Box creation is done by the proposed Particle Swarm intelligence based Genetic Algorithm. The generated S-box and the inverse S-box will be used for the encryption and the decryption operation. The pseudo code for the same is given as follows.[3][14].

/*AES symmetric cryptosystem:Particle Swarm optimization based Genetic Algorithm*/

Input: Random S Box SBox_initial , Fittest Key Fit_key

Output: Best S Box SBox_final

Procedure:

Step1: Initialize particle size, particle position, velocity Local Best.

Step2: Calculate objective function,
Index=find (SBox_initial==Fit_key )
Update_sbox (Index_shu)=SBox_initial (Index)
Step3: Local Best L_best= Update_sbox and Global Best G_best=L_best
Step4: while k<=Max_iteration
   Velocity_particle=(W* Velocity_particle )+(c*(L_best Position_particle )) +(c*( G_best-Position_particle ))
   Position_particle= Position_particle+ Velocity_particle
Step5: Calculate objective function by performing step2, obtained Update_sbox
Step6: Estimate local best and global best,
Step7: Apply crossover,
Step8: Calculate and Compare general characteristics for S-Box,
   Bijection:
   Nonlinearity:
   Strict Avalanche Criteria:
   Bit Independence:
   SBox_final= Current_SBOX; Else
   SBox_final= SBox_initial

The proposed S-box is constructed with the desirable properties such as Bijectivity, Nonlinearity, Strict Avalanche Criteria and Bit independence distribution.

**Key Dependent S-Boxes**

Initialize the value of particle size, particle position and calculate the objective function such as local best, global best [14]. The fittest key generated [6] by genetic algorithm has been used for the encryption process for AES-128 symmetric cryptosystem. The S-boxes are generated in random numbers until the generated population converges and the global best is achieved. Sample S-boxes which are generated for the key size of 16 bits is shown in the Figure 2 to Figure 7. The initial S-box is generated by iterating the GA process. The S-boxes are generated for each iterations and compared with the previously generated population [2]. The local best is obtained. This process repeats until the global best value is achieved and the final S-box is generated as shown in Figure 7.
5. Performance Analysis of the Generated S-BOX

The initial parameters to generate the key dependent S-boxes are set as probability of crossover $P_c=0.8$, probability of mutation $P_m=0.3$, number of iterations as 100 and the population size as 1000 [5]. Based on the fittest key generated [6], the proposed hybrid method of GA and PSO generates the S-boxes which are compared subsequently with the previously generated...
population. As an instance the sample S-boxes are given in Figure:2 to Figure:7. The final S-box generated has the auto correlation which lies between -1.96 and 1.96, to detect the randomness in data. Frequency test value is greater than 0.01 and the Gap test value is less than or equal to 3. The secured S-box characteristics are determined by bijectivity, non-linearity, strict avalanche criteria and bits distribution[10]. The literature study explains that a strong S-box should have the bijectivity equal to 1, non-linearity equal to 1 [15], avalanche criterion less than 0.5 and bits distribution less than or equal to 10 which is achieved in this proposed work. Table:1 depicts the key characteristics comparison values with respect to the different validation tests considered as fitness function such as Auto Correlation Test, Frequency Test, and Gap-Test. Figure: 8 shows the comparison graph for the above mentioned test.

Table 1: Key Characteristics Comparison of Proposed GA-PSO AES with Conventional AES

<table>
<thead>
<tr>
<th>Key Characteristics</th>
<th>Conventional AES</th>
<th>Proposed GA-PSO AES</th>
</tr>
</thead>
<tbody>
<tr>
<td>Auto Correlation Test</td>
<td>-3.5764</td>
<td>-1.7074</td>
</tr>
<tr>
<td>Frequency Test</td>
<td>1.9953</td>
<td>0.8184</td>
</tr>
<tr>
<td>Gap Test</td>
<td>17</td>
<td>2</td>
</tr>
</tbody>
</table>

Figure 8: Key Characteristics of S-box

Table 2: S-Box Characteristics Comparison of Proposed GA-PSO AES with Conventional AES

<table>
<thead>
<tr>
<th>S-Box Characteristics</th>
<th>Conventional AES</th>
<th>Proposed GA-PSO AES</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bijectivity</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Non-Linearity</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Strict Avalanche value</td>
<td>0.0313</td>
<td>0.0365</td>
</tr>
<tr>
<td>XOR Distribution</td>
<td>65.2600</td>
<td>7.3958</td>
</tr>
</tbody>
</table>

The cryptographic properties of S-box such as Non-linearity, bijectivity, strict avalanche criteria and bits independence that are analyzed with the existing S-box are shown in Table: 2. The resultant comparison is shown as graph in Figure: 9.
Figure 9: S-Box Characteristics for Conventional AES with Proposed GA-PSO AES

6. Conclusion

S-box is the key mechanism used in any block ciphers. Many novel techniques have been introduced for designing and generating the S-box in the recent past. Evolutionary Algorithms such as Genetic Algorithms and Particle Swarm optimization are being suggested in generating the strong S-box which exploits the randomness of these algorithms. The strength of these S-boxes depends on non-linearity, Strict Avalanche criteria, Bits distribution and Bijectivity property of the input and output bit vector. The proposed method of generating key dependent S-box is based on the hybrid combination of EAs, GA and PSO. The initial S-box is generated by iterating the GA process or can be an existing S-box generated from conventional AES-128. Genetic algorithm is applied on the basic S-box to search for a high performance S-box. The operators of GA and PSO are used as the control parameters to achieve the expected performance criteria of a better S-box. The evaluation criteria comparison shows that the key dependent S-box generated by this approach is high in performance. This research work generates a strong S-box, since the security of a cryptosystem can be increased by strengthening the performance of S-box used for cryptography. This research work is developed for the encryption and decryption of Text and Audio files. The work can be further extended to accept video and image files for encryption. Further the usage of the evolutionary algorithms can be fine-tuned and improved to generate optimized S-boxes for cryptography.
References


