An Anti-Symmetric Key Algorithm for Signal Encryption

Y. WU and A. C. VOSLER
Department of Mathematical Sciences
Georgia Southern University
P. O. Box 8093
Statesboro, GA 30460
United States of America

Abstract: - In this paper, we propose an encryption algorithm with anti-symmetric keys. The algorithm is assumed to be known to public, but the keys are kept private. The primary key (at the encryption site) is designed based on matrix transformation with the requirement that all arithmetic operations are strictly over the integer field. Therefore, there are no round off errors when the signal is deciphered at the receiver site. The anti-symmetric key algorithm is designed for constructing such a primary key in the proposed encryption scheme, which is shown to be robust against attacks.

Key-Words: Adjunct matrix, Linear Diophantine equation, nonlinear Diophantine equation, cryptosystem, Encryption key, Kerckhoffs' principle, integral matrix, matrix encryption

1 Introduction

Information security and integrity are protected by encryption techniques. Encryption is the process of scrambling the original form of information by mathematical algorithms. This process results in enciphered information that will not give away its content to any unauthorized person. With the rapid progress of internet and advances in communication technology, there is a great need of more efficient and faster cryptography algorithms than those classical algorithms such as IDEA-International Data Encryption Algorithm or DES-Data Encryption Standard [1]. In the past ten years or so, a number of innovative encryption techniques have been developed for network security and signal encryption [2-5]. Few of those algorithms satisfy both efficient and secure encryption.

In this paper, we present an anti-symmetric encryption scheme based on matrix masking. The proposed technique is efficient and satisfies Kerckhoffs' principle in cryptography. This encryption technique is developed within the realm of Digital Signal Processing. As in DSP, analog signals are converted into digital signals through sampling and quantization. Each sample of a digital signal is presented in the form of an integer or an array of binary bits. Therefore, matrix encryption becomes a natural resort for scrambling the digital signal data via multiplications between the transformation matrix and the data. The main idea is to assemble the data into multiple matrices with an associate key. The data matrix is then multiplied with another matrix, the primary key. Decryption is done by rearranging the numbers into matrices (the inverse of the associate key), and then multiplied by the inverse of the primary key matrix. Yang *et. al.* [6] used matrix transformation to develop an asymmetric block encryption scheme. Their method was shown to be effective for the encryption of large amounts of data, such as digital images, in terms of less computation complexity and better security. However, a significant setback of their scheme is the inevitable round off errors in and out of the DCT (Discrete Cosine Transform) domain, which could lead to imperfect decryption, distortion, and loss information. This problem can be magnified when large dimensional key matrices are adopted. The anti-symmetric key algorithm presented in this paper is developed under the restriction that all arithmetic operations are done over the integer field. This is made possible with a special design on the primary key integral matrix such that its inverse is also an integral matrix. Therefore, it is guaranteed that there are no round off errors at both the encryption site and the decryption site, making the proposed encryption system a lossless system in addition to its security. Details of the key algorithms are found in section 2, followed by attack analysis and an application to speech coding in sections 3 and 4 respectively.

2 Anti-symmetric key algorithm

There are two keys at the encryption site and decryption site respectively. The two keys, the primary key and the associate key, at the receiver end are the inverse of the two keys at the sender site.

They are not identical, but closely related. That is why this is called an anti-symmetric key encryption system.

The associate key, T_α, defined as an operator taking matrices from one vector space to another vector space, i.e. $T_\alpha : Z^{k\times L} \to Z^{M\times N}$, where $Z^{k\times L}$ is the subspace of all K by L matrices over the integer field Z, same for $Z^{M\times N}$. The associate key functions as an assembling operator that shuffles the data of the original signal into a new set of data (as a matrix). The operator, T_α, is required to be homeomorphic, i.e. the inverse operator, T_α^{-1} exists. It is also required that such an operator is nontrivial in a sense that the original signal cannot be easily recovered from the new set of data. For example, given a speech signal, which is usually stored as a one-dimensional array, one only needs to use the corresponding cardinal indices associated with the original signal. The associate key T_α is characterized by a random or a special protocol shuffling of the indices. Then, the elements in the sequence of the original signal are rearranged accordingly. At the receiver site, the order of the elements in the original sequence can be recovered by sorting as denoted by T_α^{-1}. Same process can be applied to two-dimensional signals such as images. One technical point for the output of T_α is that, if $MN > KL$, one simply pad zeros to the end of the data so that the new set of data can be stored as a matrix with appropriate dimensions.

The design of the primary key is the center piece of this paper. Let T_p denote the primary key, the kernel of which is in essence a matrix multiplication, $T_p : Z^{M\times N} \to Z^{M\times N}$. Therefore, at the sender site, a message $s \in Z^{K\times L}$ is encrypted as $S = T_\alpha T_p T_\alpha \{s\}$, and, at the receiver site, the message is decrypted as $s = T_\alpha^{-1} T_p^{-1} T_\alpha^{-1}\{S\}$. As mentioned earlier, the kernel matrix of T_p^{-1} is required to be an integral matrix so that no roundoff errors take place during the process of decryption. The following theorem provides theoretical support of the proposed technique.

Theorem 1. *Suppose* $A \in Z^{N\times N}$ *and* A *is a nonsingular matrix, then* $A^{-1} \in Z^{N\times N}$ *if and only if* $\det(A) = \pm 1$.

Proof: Since A is invertible, and with the fact that $\det(A^{-1}) = \dfrac{1}{\det(A)}$, then $\det(A^{-1})\det(A) = 1$. Since $A \in Z^{N\times N}$, $\det(A) \in Z$. If $A^{-1} \in Z^{N\times N}$, then $\det(A^{-1}) \in Z$. Therefore, $\det(A) = \pm 1$. Conversely, if $\det(A) = \pm 1$, use the inverse formula, $A^{-1} = \dfrac{A^{*\mathrm{T}}}{\det(A)}$, where A^* is the adjunct matrix of A, let $A^* = (a_{ij}^*), a_{ij}^* = (-1)^{i+j}\det(A_{ij})$, where A_{ij} is a submatrix of A by eliminating the i_{th} row and j_{th} column of A. Hence $\det(A_{ij}) \in Z$, which leads to the result $A^{-1} \in Z^{N\times N}$.

We seek a primary key, an integral matrix, in the proposed algorithm under the condition that the inverse of the matrix is also an integral matrix so that there will be no roundoff errors during the decryption process. Theorem 1 provides a necessary and sufficient condition for constructing such integral matrices, i.e. the determinant of the matrix has to be one. Consider the integer field as well the complexity of evaluating a determinant, the chance for the determinant of an arbitrary integral matrix to equal plus or minus one is very slim. It is the goal of this paper, however, to develop algorithms for systematic construction of such matrices.

Suppose we have at hand an arbitrary integral matrix, the main idea is to allow one or two (maximum two in this paper) free parameters in the entries of the matrix. We will solve a number of Diophantine equations for the undetermined free parameters so that the solutions will make the determinant of the new matrix equal plus or minus one. The general form of the Diophantine equation [7] is given by

$$axy + bx + cy + d = \pm 1 \qquad (1)$$

where the left side of eqn.(1) equals the determinant of the matrix A, i.e. $\det(A) = axy + bx + cy + d$, and $a, b, c, d, x, y \in Z$. We consider the following three cases:

(i) One free parameter x, corresponding linear Diophantine equation $bx + d = \pm 1, b \neq 0$.

(ii) Two free parameters, x and y, corresponding linear Diophantine equation $bx + cy + d = \pm 1$, $b^2 + c^2 \neq 0$.

(iii) Two free parameters, x and y, corresponding nonlinear Diophantine equation $axy + bx + cy + d = \pm 1,\ a \neq 0$.

See (2) for an interpretation of the above three cases. It is worth noticing that case (ii) emerges whenever the two free parameters are on the same row or column. Otherwise, the Diophantine equation will be nonlinear (1).

$$\text{(i) } A(x)=\begin{bmatrix} 1 & -2 & 3 & 5 \\ 0 & x & 1 & 2 \\ -4 & 11 & 6 & 8 \\ 1 & 0 & 1 & 2 \end{bmatrix}$$

$$\text{(ii) } A(x,y)=\begin{bmatrix} 1 & -2 & 3 & 5 \\ 0 & 3 & x & 2 \\ -4 & 11 & y & 8 \\ 1 & 0 & 1 & 2 \end{bmatrix} \qquad (2)$$

$$\text{(iii) } A(x,y)=\begin{bmatrix} 1 & x & 3 & 5 \\ 0 & 3 & 1 & y \\ -4 & 11 & 6 & 8 \\ 1 & 0 & 1 & 2 \end{bmatrix}$$

Next, we need to determine the coefficients in the Diophantine equation, which is done as follows:

(i) Set $x=0$, then $d=\det(A(0))$; set $x=1$,
$b=\det(A(1))-d=\det(A(1))-\det(A(0))$.

(ii) Set $x=y=0$, then $d=\det(A(0,0))$; set $x=0$ and $y=1$, $c=\det(A(0,1))-d=\det(A(0,1))-\det(A(0,0))$; set $x=1$ and $y=0$, $b=\det(A(1,0))-\det(A(0,0))$.

(iii) Repeat (ii), and set $x=1$ and $y=1$,
$a=\det(A(1,1))-b-c-d=$
$\det(A(1,1))-\det(A(1,0))-\det(A(0,1))+\det(A(0,0))$.

Now, the coefficients in Diophantine equation (1) are totally determined, we proceed to solve these three cases respectively. Case (i) is the easiest, because the solution exists as long as the coefficient b is a factor of $-d\pm 1$. In general, such a solution is hard to find. It can be considered as a low probability event. That is why we introduce two free parameters in (ii) and (iii).

For case (ii), the linear Diophantine equation is rewritten as

$$bx+cy=e. \qquad (3)$$

Without loss of generality, we assume $b^2+c^2\neq 0$. The Diophantine equation can be solved via the Euclidean algorithm as follows:

Step 1. Find the greatest common divisor between *b* and *c*, i.e. gcd(*b*,*c*), as follows:

$$\begin{aligned} & b=q_1c+r_1 \\ & c=q_2r_1+r_2 \\ & r_1=q_3r_2+r_3 \\ & \vdots \\ & \text{stop if } r_i=0, \gcd(b,c)=r_{i-1} \end{aligned} \qquad (4)$$

Divide both sides of (3) by r_{i-1} to get

$$\frac{b}{r_{i-1}}x+\frac{c}{r_{i-1}}y=\frac{e}{r_{i-1}}.$$

If $\frac{e}{r_{i-1}}$ is not an integer, there is no integer solution to (3). Otherwise, proceed to step 2 with a new Diophantine equation

$$b'x+c'y=1 \qquad (5)$$

where $b'=\frac{b}{r_{i-1}}$ and $c'=\frac{c}{r_{i-1}}$. The term $\frac{e}{r_{i-1}}$ will be used to multiply the solution of (5) to obtain the solution of (3).

Step 2. Use the matrix form of the Euclidean algorithm to solve for *x* and *y* in (5), begin with

$$\begin{bmatrix} 1 & 0 & b' \\ 0 & 1 & c' \end{bmatrix}$$

Let $b'\equiv n_1 \bmod c'$, and $\frac{\mathrm{b}'-n_1}{c'}=q_a$

$$\begin{bmatrix} 1 & -q_a \\ 0 & 1 \end{bmatrix}\begin{bmatrix} 1 & 0 & b' \\ 0 & 1 & c' \end{bmatrix}=\begin{bmatrix} 1 & -q_a & b'-c'q_a \\ 0 & 1 & c' \end{bmatrix} \qquad (6)$$

Let $c'\equiv n_2 \bmod \mathrm{b}'-c'q_a$, and $\frac{\mathrm{c}'-n_2}{b'-c'q_a}=q_b$

$$\begin{bmatrix} 1 & 0 \\ -q_b & 1 \end{bmatrix}\begin{bmatrix} 1 & -q_a & b'-c'q_a \\ 0 & 1 & c' \end{bmatrix}=$$

$$\begin{bmatrix} 1 & -q_a & b'-c'q_a \\ -q_b & 1+q_aq_b & c'-b'q_b+c'q_aq_b \end{bmatrix}$$

Let $b'-c'q_a\equiv n_3 \bmod c'-b'q_b+c'q_aq_b$, and

$$\frac{b'-c'q_a-n_3}{c'-b'q_b+c'q_aq_b}=q_c.$$

Note: The *q*-matrix in (6) alternates with -*q* in position (1,2) and (2,1) in the matrix with -*q* starting in (1,2) position during the iterations. When -*q* is in (1,2) position, the new divisor will be in (1,3) position in the new matrix as a result of multiplication in (6). When -*q* is in (2,1) position of the *q*-matrix, the new divisor will be in (2,3) position in the new matrix. Stop the iteration when zero appears in (1,3) or (2,3) position of the new matrix, but not both, and the nonzero number must be 1. The other two numbers on the same row are solutions for *x* and *y*, multiply *x* and *y* by $\frac{e}{r_{i-1}}$ to get the solution for (3).

For case (iii), referring to the nonlinear Diophantine equation (1), we first rewrite it as

$$axy + bx + cy + e = 0 \qquad (7)$$

where $e = -d \pm 1$ from (1). Use the following algorithm to solve for *x* and *y*:

Step 1. Factor (7) into

$$(ax + c)(ay + b) = bc - ae \qquad (8)$$

Step 2a. If $bc - ae = 0$, solutions are

$$x = -\frac{c}{a} \text{ if } \frac{c}{a} \in Z,\ y = \text{any integers}$$

or

$$x = \text{any integers},\ y = -\frac{b}{a} \text{ if } \frac{b}{a} \in Z.$$

Step 2b. If $bc - ae \neq 0$, use prime factorizations to find all possible divisors of $bc - ae$, such divisors d_i should satisfy

$$-|bc - ae| \le d_i \le |bc - ae|,\ i = 1, 2, 3, \ldots$$

and Eqn. (8) yields the following possible solutions:

(i) If $x = \frac{d_i - c}{a} \in Z$ and $y = \frac{bc - ae - bd_i}{ad_i} \in Z$, then $\left\{ \frac{d_i - c}{a}, \frac{bc - ae - bd_i}{ad_i} \right\}$ are the integer solutions.

(ii) If $x = \frac{bc - ae - cd_i}{ad_i} \in Z$ and $y = \frac{d_i - b}{a} \in Z$, then $\left\{ \frac{bc - ae - cd_i}{ad_i}, \frac{d_i - b}{a} \right\}$ are the integer solutions.

Otherwise, use new divisor d_j and repeat the step (i) or (ii).

3 Performance analysis and attack analysis

As shown in the above discussions of the algorithms for constructing the integral matrix with an integral inverse, it is possible that there might be no solutions to a certain Diophantine equation. However, it only implies that the choices for the locations of the free parameters in a matrix are unsuitable. One can always pick different locations in the matrix for the free parameters and seek solutions. According to our experience, the possibility of finding an integral matrix with an integral inverse is much better with the proposed algorithm and the computational complexity is considerably low, examined from MATLAB's built in function FLOPS for counting float point operations.

As a demonstrative example, we apply the above algorithms to a random 3 by 3 integral matrix,

$$A = \begin{bmatrix} 1 & 3 & 2 \\ -2 & 6 & 11 \\ 1 & -9 & 13 \end{bmatrix},\ \det(A) = 312.$$

Let the two free parameters take $a_{12}(x)$ and $a_{33}(y)$ positions in *A*, via the proposed algorithms, the modified matrix is found to be

$$B = \begin{bmatrix} 1 & -4 & 2 \\ -2 & 6 & 11 \\ 1 & -9 & 39 \end{bmatrix},\ \det(B) = 1, \text{and}$$

$$B^{-1} = \begin{bmatrix} 333 & 138 & -56 \\ 89 & 37 & -15 \\ 12 & 5 & -2 \end{bmatrix}.$$

In order to demonstrate the proposed encryption method, we encrypt a speech signal with an associate key based on a normally distributed random seed generator and a primary key matrix, a ten by ten integral matrix with integral inverse. The encrypted and decrypted signals are shown in Fig. 1. As discussed earlier, the decrypted signal is a prefect match of the original signal because of the specially designed primary key matrix.

Kerckhoffs' principle states that a cryptosystem should be secure even if everything about the system, except the key, is public knowledge; it was reformulated by Claude Shannon as ``the enemy knows the system''. We assume that the proposed algorithm is available to the public. However, the keys are kept secret. In order to break the cryptosystem, the attackers need to obtain the private keys in their own exact forms. This is difficult not only those keys can be renewed at any moment, but any attempt to reconstruct the keys would be in vain. First off, the associate key is constructed via a random seed generator. It is totally random. Therefore, it is impossible to predict or reconstruct. On top of that associate key, the primary key matrix is designed at will. To a certain degree, it is a random matrix too. Only one or two entries of the matrix are altered to make sure that the matrix has an integral inverse. Even for a small matrix such as an 8 by 8 matrix, there are 64 entries in the matrix, and there are infinitely many ways of assigning integers to those 64 spots simply because there are infinitely many integers. It is even more difficult to estimate the inverse of the primary key directly because the inverse of an integral matrix (with a unit determinant) is usually a matrix with big numbers, see the 3 by 3 example earlier. What if the attackers are lucky to fabricate a matrix that is close to the private primary key? Will it post a serious threat? Not at all. It is well known in matrix theory that inverting a matrix is the most unstable matrix operation, which means that a small perturbation to the original matrix usually leads to a totally different inverse matrix comparing to the inverse of the original matrix. In order to demonstrate this property, we use the same speech signal as earlier. A small perturbation to the diagonal elements of the primary key matrix (other elements of the perturbed matrix are exactly the same as the primary key matrix) and the inverse of the resulted matrix is used to decrypt the coded signal. The reconstructed signal is plotted in the same window as the original signal, see Fig.2, the discrepancies are so significant that the end result is unacceptable.

4 Conclusion

The main objective of this paper is to precisely reconstruct the signal at the decryption site with the proposed cryptosystem without sacrificing the security of the system. The proposed algorithms are heuristic and easy to implement. This scheme satisfies the characters of convenient realization, low computational complexity, and sound security. The existence of such integral matrices with integral inverses for the design of a primary key can be significantly improved if one allows more than two free parameters in the algorithm. The authors have made good progress toward generalizing the algorithms to more than two free parameters. The results will be reported in a future paper.

References:

[1] B. Schneier, *Applied Cryptography*, 2_{nd} edition, John Wiley & Sons, Inc. 1996.

[2] K. X. Yi and S. Xing, An image encryption algorithm based on chaotic sequences, *Journal of Computer Aided Design and Computer Graphics*}, v. 6, 2000, pp. 672-676.

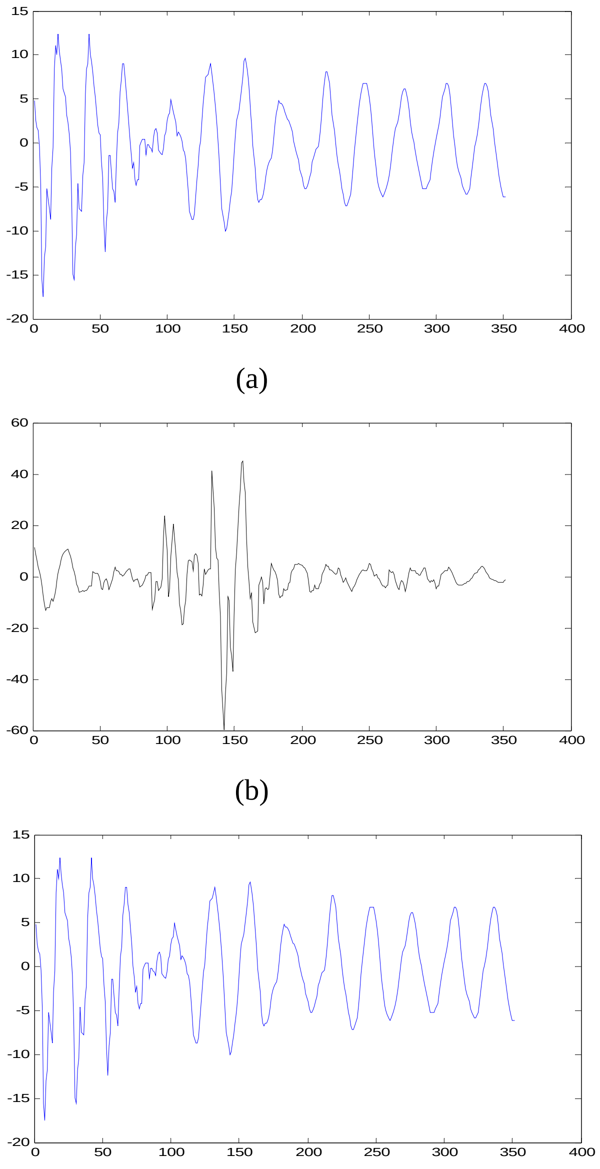
[3] X. H. Zhang, F. Liu, and L. C. Jiao, An encryption arithmetic based on chaotic sequence, *Image and Graphics*, v. 8, no. 4, 2003, pp.374-378.

[4] C.C. Chang, A new encryption algorithm for image cryptosystems, *Journal of Systems and Software*, v. 5, no. 7, 2001, pp. 83-91.

[5] C. J. Kou, Novel image encryption technique and its application in progressive transmission, *J. Electron. Imaging*, v. 2, no. 4, 1994, pp.345-351.

[6] S. Y. Yang, Z. D. Lu, and S. H. Han, An asymmetric image encryption based on matrix transformation, Proceedings of International Symposium on Communications and Information Technology, 2004, pp. 66-69.

[7] D. Redmond, *Number Theory*, Marcel Dekker, Inc., New York, 1996.



(a)

(b)

(c)

Figure 1. (a) A speech signal; (b) Encrypted signal with an 8 by 8 primary key matrix; (c) Decrypted signal with the inverse of the primary key matrix, a perfect match with the original signal.

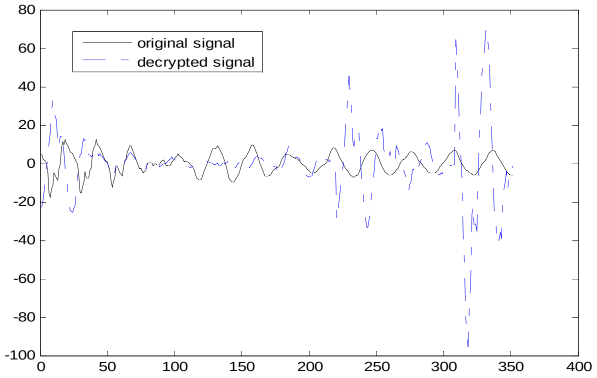


Figure 2. Comparison between the original signal and the decrypted signal with a slightly perturbed primary key matrix