Malaysian Journal of Computing and Applied Mathematics 2019, Vol 2(1): 16-24
©Universiti Sultan Zainal Abidin
(Online)

MALAYSIAN JOURNAL OF COMPUTING AND APPLIED MATHEMATICS

Production of a Super-Increasing Sequence based on the Fibonaci Sequence

Assia Merzoug[a], *Adda Ali-Pacha[b], Naima Hadj-Said[b], Hana Ali-Pacha[b]

[a]University Hadj Lakhdar Batna, 05 Avenue Chahid Boukhlouf Batna 05000, Algeria
[b]University Science and Technology of Oran Mohamed Boudiaf, BP 1505 El M'Naouer Oran 31000, Algeria

*Corresponding author: a.alipacha@gmail.com

Received: 07/01/2019, Accepted: 22/05/2019

Abstract

One shows that we can build a long recurring sequence super-increasing and use it in the cryptographic system based on the Knapsack problem, for example the Merkle-Hellman cipher. Thus, we reduce the size of the key with the same system safety. For this, we have modified the generalized Fibonacci sequence to produce of a super-increasing sequence. This modification was based essentially on the use of real coefficients in the main recursive equation. The result of this, it is the transformation of the public key of the Merkle-Hellman crypto system into a secret algorithm and, for an equivalent complexity.

Keywords: Fibonacci sequence; Merkle-Hellman; super-increasing sequence; Knapsack problem.

Introduction

The knapsack problem is one of 21 NP-complete problems of Richard Karp, set out in his Article in 1972 (Karp, 1972; Clark et al., 1996). The formulation of the problem is simple, but its resolution is more complex. However, the singularly structure of the problem and, the fact that it to be present as a sub-problem in other more general problems, make it a subject for the research. It consists to stack objects into a bag, to achieve (if possible) a total fixed weight. More formally, given the integers weights P_1... P_n and the goal T, it is to find b_1... b_n, being worth 0 or 1, such as:

$$T = b_1P_1 + b_2P_2 + ... + b_nP_n$$

If the sequence of P_k weight is a super-increasing sequence (each weight is strictly greater than the sum of all previous), then there exist a simple resolution method (greedy algorithm):

Greedy Algorithm

For i = n à 1 Do

Si T ≥ Pi Then
T = T - Pi
bi = 1
If not
bi = 0
If T = 0 alors {b_1, ..., b_n} is a solution, otherwise there is no solution

Make sure that with this super-increasing sequence P_1=2, P_2=3, P_3=6, P_4=12 and T=15, one obtained the solution b_1=0, b_2=1, b_3=0, b_4=1.

On the contrary, if the sequence of the weight is not a super-increasing sequence, the only known algorithm consists to try successively all the solutions (b_1, b_2... b_n) possible. If the sequence is sufficiently long, it is an impractical algorithm. The knapsack problem is another example of one-way function (for fixed x, the calculation of f(x) is very easy, but the reverse is not possible). It is also used in cryptography as a basis for different encryption schemes. It should be noted that most of these encryption schemes are currently not considered safe. In 1978 (Merkle and Hellman, 1978), Ralph Merkle and Martin Hellman proposed their public key crypto system based on this famous problem. In this work, we try to propose a method based on the generalized Fibonaci sequence (Schneier, 1996) with real coefficients, for the construction of the super-increasing sequence and, use it in the Merkle-Helman cryptographic system. The novelty in this approach is the transformation of this algorithm into a secret-key cryptographic system, with a diminution of the length of the encryption key.

The organization of this work is as follows, Section 2 involves the study of the Merkle-Hellman algorithm and Section 3 provides an overview on the Fibonaci sequence. In Section 4, we propose a method to generate the super-increasing sequence, the originality of this improvement is the use of real coefficients in the generalized Fibonaci sequence, some results are presented in this section and ending in the next section with a conclusion.

Merkle-Helman Cryptographic System

The cryptosystem Merkle and Hellman (Merkle and Hellman, 1978; Schneier, 1996; Stinson, 2001) uses the knapsack problem described above as follows:

Generation of keys of crypto system

1. A positive integer n, large enough (Merkle and Hellman recommend taking n in the order of 100).
2. Choose a sequence {b_1, b_2,..., b_n} of positive integers checking the following property:
$$\forall\, i \in [2,n], b_i > \sum_{j=1}^{i-1} b_j$$
3. Choose an integer M, called module, such as:
$$M > \sum_{i=1}^{n} b_i$$
4. Choose an integer $W \in [1, M\text{-}1]$ prime with M, as the gcd (W, M) = 1.
5. Calculate:
$$a_i' = W \times b_i \bmod M \text{ pour } i \in [1, n]$$
6. Find the permutation π from {1, 2,..., n} such that {$a_{\pi(1)}'$, $a_{\pi(2)}'$,..., $a_{\pi(n)}'$} is an increasing sequence.
7. The public key is:
$$(a_1, a_2, ..., a_n) = (a_{\pi(1)}', a_{\pi(2)}', ..., a_{\pi(n)}')$$
This key may be freely distributed to all potential correspondents or stored in a directory comparable to those used for phone numbers.
8. The private key consists of M, W, (b_1, b_2,..., b_n) and the permutation π.

This private key must always be kept confidential and must not be provided to anyone because it is not needed to encrypt a message. However, It is indispensable (if the system is safe) in order to decrypt a message.

Principle of Encryption

1. The message to be encrypted is written in a sequence in the binary form: $m_1 m_2 ... m_n$, with $m_i \in \{0,1\}$, (if the message is too long, it is cut into blocks of n bits or less).
2. Calculate the encrypted C as:

$$C = \sum_{i=1}^{n} m_i . a_i$$

3. Transmit C.

Decryption Algorithm

1. Calculate $d = W^{-1} \times (c \bmod M)$, using the extended Euclidean algorithm,
2. Calculate $\varepsilon_1, \varepsilon_2, ..., \varepsilon_n$ such that d is given by

$$d = \sum_{i=1}^{n} \varepsilon_i . b_i$$

It is very simple to solve this knapsack problem using the following property of b_i:

$$b_i > \sum_{j=1}^{i-1} b_j$$

3. Calculate $m_i = \varepsilon_{\pi(i)}$ for $i \in [1, n]$
4. The decrypted message is written in a sequence in the binary form, as $m_1 m_2 ... m_n$.

Numerical Example

1. For a small artificially n, we take n = 10
2. Choose the b_i : 4, 9, 30, 70, 185, 451, 1306, 3534, 6517, 17486
3. Choose M : 50349 ($> \sum_{i=1}^{n} b_i$)
4. Choose W : 36334 (prime with M)
5. Calculate the a_i : 44638, 24912, 32691, 25930, 25373, 23209, 23446, 14406, 47680, 32642
6. Find the permutation π : π(1)=8, π(2)=6, π(3)=7, π(4)=5, π(5)=2, π(6)=4, π(7)=10, π(8)=3, π(9)=1, π(10)=9.
7. The public key is: (14406, 23206, 23446, 25373, 24912, 25930, 32642, 32691, 44638, 47680)
8. The private key consists of M, W, (b_1, b_2,..., b_{10}) and the permutation π.
   Encryption of the message (1,0,0,1,0,0,0,1,1,0), we will have C=14406+25373+32691+44638=117108
   The decryption of C is as follows, applying the extended Euclidean algorithm to W and M : 7864× W-5675× M=1 thus W^{-1}=7864 mod M, we calculate $d = W^{-1} \times c \bmod M$ = 7865× 117108 mod 50349=3753
9. Resolution of the knapsack problem with b_i and the target value 3753 : 3534+185+ 30+4=3753=b_1+b_3+b_5+b_8, whether : (ε_1,ε_2,ε_3, ε_4, ε_5, ε_6, ε_7, ε_8, ε_9, ε_{10})= (1,0,1,0,1,0,0,1,0,0)
10. Calculation of $m_1 = \varepsilon_{\pi(1)} = \varepsilon_8 = 1$, $m_2 = \varepsilon_{\pi(2)} = \varepsilon_6 = 0$, $m_3 = \varepsilon_7 = 0$, $m_4 = \varepsilon_5 = 1$, $m_5 = 0$, $m_6 = 0$, $m_7 = 0$, $m_8 = 1$, $m_9 = 1$, $m_{10} = 0$
11. We recover the original message (1,0,0,1,0,0,0,1,1,0).

Other Example

1. For n = 9, Alice chooses S = (1, 3, 5, 11, 25, 53, 101, 205, 512), m = 960 and w = 143. The inverse d of 143 mod 960 is 47.
2. For each element ai of S, Alice computes $b_i = a_i * e \bmod m$, to give (143, 429, 715, 613, 695, 859, 43, 515, 256). By ordering b_i, it gets the public key 'knapsack' S '= (43, 143, 256, 429, 515, 613, 695, 715, 859).

3. To express the message "RAS" in binary code, Bernard can, for example, use the 8-bit ASCII code. R corresponds to 01010010, A to 01000001 and S to 01010011. The message to be encryption is “RAS” = 0101001 0010000 0101010 011.
   It breaks down into blocks of length L, agreed (7 for example) and encrypts each block: 0101001 to be coded into 43 + 429 + 613 = 1085, 0010000 to be coded into 515, 0101010 to be coded into 143 + 429 + 613 = 1185, 011 $\rightarrow$ 0110000 and, to be coded into 515 + 613 = 1128.
   It transmits the following message to Alice 108-515-1185-1128
4. Alice will decrypt the message element by element, calculating (M*d mod m), and determining the solution of the knapsack problem.

- ✓ 1085 * 47 mod 960 = 115 = 101+11+3 correspond to 0000001 + 0100000 + 0001000 = 0101001
- ✓ 515 * 47 mod 960 = 205 = 205 correspond to 0010000
- ✓ 1185 * 47 mod 960 = 15 = 11+3+1 correspond to 0100000 + 0001000 + 0000010 = 0101010
- ✓ 1128 * 47 mod 960 = 216 = 205+11 correspond to 001000 + 0100000 = 0110000
  Alice finds the original message: 0101 0010 0100 0001 0101 0011 0000 : RAS.

Encryption of Text

1. n = 9
2. Choose a super-increasing sequence S (Schneier 1996) containing at least nine elements (separate your numbers with commas): 2, 5, 9, 21, 45, 103, 215, 450, 946
3. $\sum_{i=1}^{n} a_i = 1796$
4. Choose a number M greater than ($\sum_{i=1}^{n} a_i$) and a number W prime with M:
   - M = 2003
   - W = 1289
5. Your public key is: {436, 569, 575, 721, 1030, 1183, 1570, 1586, 1921}.
6. The inverse of (W mod m) is : 317
7. Choose the length L of the encryption block, L should be less than or equal to 9
   - L = 5
   - L = 8

The text to encrypt is in French, its translation is as follows: “The first public key of cryptosystem, which was proposed by Ralph Merkle and Martin Hellman in 1978, is based on the knapsack problem. There is currently not used, as well as many variations, has been broken by Adi Shamir in the early 80.”

Plaintext in French : « *Le premier cryptosystème à clef publique, qui fut proposé par Ralph Merkle et Martin Hellman en 1978, est basé sur le problème du sac à dos (Knapsack problem en anglais). Il n'est plus utilisé actuellement puisque ce chiffre, ainsi que de nombreuses variantes, a été cassé au début des années 80 par Adi Shamir.* »

Ciphertext With L= 5 : « 1157,*1466,1599,1599,0,2326,1005,1599,1296,2041,2174,2174,1599,2187,1726,1599,575,436,1466,2610,575,2895,1726,1030,1865,1466,2610,2610,1144,2895,1726,2035,1865,2035,1605,1144,2320,2187,1157,0,2326,0,1030,1144,1144,2756,1005,1011,1296,1751,1030,1580,0,2762,1726,569,1732,1466,1605,2610,569,2762,1726,1011,1011,1030,1030,1580,569,2762,1726,1157,575,436,2035,1580,1290,2762,436,0,1865,436,2187,1144,2895,2326,1005,2301,1865,2301,1605,1599,0,2326,1005,436,1865,721,1030,1011,575,1751,1726,1296,1865,436,1605,569,0,2035,1726,1011,1865,1157,2041,2174,1751,2187,1157,0,1296,2041,2320,569,0,2035,1726,436,1865,1157,2320,1144,1599,2756,1466,0,1157,436,1599,2174,1751,2756,1005,1732,1296,1011,2610,569,0,2187,1726,1865,575,0,1751,2035,1599,1732,2187,1751,1011,1030,1030,1144,1290,2326,2756,1605,575,436,1466,1144,569,2326,3331,1157,575,436,2187,2610,1290,2326,1466,0,1732,1466,1599,1599,0,2326,1005,1599,1732,2762,1466,1144,2187,2320,1005,1732,1296,1605,1030,1144,721,2762,1157,0,1865,1732,1030,2174,1144,721,1580,0,575,436,1599,1144,2895,2326,2187,0,1011,436,1011,2174,2326,1751,1726,1030,1865,1732,1030,2174,1144,2320,2187,0,1865,436,2187,1144,2895,1751,2035,1296,1296,2041,2174,1599,0,2187,1726,1865,575,436,1030,2174,2326,2187,2756,1296,1296,1011,1605,2610,1144,1290,1157,1865,575,436,575,2174,1751,721,1005,1865,575,2762,1599,2610,1144,2762,436,0,1865,436,2174,1580,1290,2326,21*

87,0,1865,2041,2320,1144,1599,2756,1005,1157,1865,2301,1605,1599,0,1751,1726,1005,1865,1466,2320,2174,12 90,2756,1005,1296,1296,2041,2174,2174,1290,2756,2035,1605,575,436,1751,1580,1290,2320,1726,2035,1865,10 11,2320,2174,1290,721,1005,1005,1296,1605,1030,1144,1144,2320,1005,1157,1296,2187,2035,1580,575,2187,11 57,1296,575,436,1030,2174,1599,2756,2035,2035,1732,575,1030,1580,569,2762,1726,1011,575,436,1599,1144,12 90,721,1005,1865,1732,2762,2174,2174,575,2326,2035,1011,1865,2041,2187,2174,1290,2326,2187,0,1865,2187,1 030,2610,575,2320,1726,436,1732,2187,2320,1144,1290,2326,2187,1296,575,436,1030,1599,436,2320,1726,1605, 2762,575,1030,1144,1144,1751,1726,2035,1865,2301,1605,1599,0,1751,1726,2041,575,436,1599,1865,1599,1751, 2035,2041,1865,1030,1030,1144,721,2187,1726,2035,575,436,1030,2174,2326,2756,2610,1157,1296,2041,2187,1 599,0,1865,436,1030,575,436,1751,1144,569,2326,1466,0,721,1011,1599,1144,1599,721,569,2035,1732,436,1030, 2174,2320,2320,1726,1599,1011,175. ».

Ciphertext With L= 8 : « *2866,3764,1183,3783, 4352,3764,4485,3910,3764,4352,1183,3758,4352,4940,3783,4358,5054,4788,4940,4788,4358,5060,4485,3764,11 83,4339,1183,3758,4049,3764,3897,1183,3783,4794,3322,4049,3910,4219,4794,3764,2479,1183,4219,4794,3910, 1183,3897,4794,4358,1183,3783,4352,5054,3783,5054,4788,5496,1183,3783,3189,4352,1183,3169,3189,4049,37 83,3474,1183,3302,3764,4352,4479,4049,3764,1183,3764,4358,1183,3302,3189,4352,4358,3910,4618,1183,2291, 3764,4049,4049,4485,3189,4618,1183,3764,4618,1183,2649,3370,3793,2934,2479,1183,3764,4788,4358,1183,33 22,3189,4788,5496,1183,4788,4794,4352,1183,4049,3764,1183,3783,4352,5054,3322,4049,5060,4485,3764,1183, 3328,4794,1183,4788,3189,3758,1183,4339,1183,3328,5054,4788,1183,1904,3296,4618,3189,3783,4788,3189,37 58,4479,1183,3783,4352,5054,3322,4049,3764,4485,1183,3764,4618,1183,3189,4618,4333,4049,3189,3910,4788, 2340,3048,1183,2727,4049,1183,4618,2763,3764,4788,4358,1183,3783,4049,4794,4788,1183,4794,4358,3910,40 49,3910,4788,5496,1183,3189,3758,4358,4794,3764,4049,4049,3764,4485,3764,4618,4358,1183,3783,4794,3910, 4788,4219,4794,3764,1183,3758,3764,1183,3758,3474,3910,3897,3897,4352,3764,2479,1183,3189,3910,4618,47 88,3910,1183,4219,4794,3764,1183,3328,3764,1183,4618,5054,4485,3322,4352,3764,4794,4788,3764,4788,1183, 4927,3189,4352,3910,3189,4618,4358,3764,4788,2479,1183,3189,1183,5496,4358,5496,1183,3758,3189,4788,47 88,5496,1183,3189,4794,1183,3328,5496,3322,4794,4358,1183,3328,3764,4788,1183,3189,4618,4618,5496,3764, 4788,1183,2934,2213,1183,3783,3189,4352,1183,2006,3328,3910,1183,3605,3474,3189,4485,3910,4352,3048* ».

Several remarks must be made:

1. If the length of the encryption block is equal to the characters in 8-bit ASCII code, each letter will be encoded by the same number. Then the system is vulnerable to an attack by the frequency analysis. It is therefore appropriate to choose the length of encryption blocks lower than that of the key.
2. Only the knowledge of the public key is required to decrypt a message. By cons, it is necessary to have all of the private key elements to be able to encrypt, using the proposed algorithm. We are therefore in a public key encryption scenario.
3. Find the plaintext message associated with an encrypted requires solving a knapsack problem as described above. Thus, the security of the crypto-system therefore relies heavily on the assumption that this problem is impossible to solve without knowing the private key.
4. Knowing the private key, it will be easy to calculate the public key. Nevertheless, find the private key, including b_i, from the public key, i.e. a_i, is a seemingly difficult problem.

Generalized Fibonaci Sequence

The Fibonacci sequence (Schneier, 1996; Allaire and Kaber, 2002; Schatzmann, 2002) satisfies the following recurrence relation:

$$u_{n+1}=u_n+u_{n-1}.$$

This recurrence relation is initiated by the first two terms, which are $u_0=0$, and $u_1=1$. The first terms of the Fibonacci sequence are:

1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89,...

What characterizes this sequence of numbers and make it universal is the universal fact that, if we take two consecutive numbers and that, we divide the smallest of them by the largest, we always get an approximate value of quotient 1/1.618, or 0.618, that is to say, the golden ratio ($\varphi = \frac{1}{1.618} = 0.618$).

Sometimes we meet the definition of generalized Fibonacci sequences. These are the sequences that satisfy the following recurrence equation, with a and b integers:

$$u_{n+1}=a*u_n+b*u_{n-1}.$$

The name of this sequence "Fibonacci sequence" was given by the French arithmetician, Edouard Lucas in 1817, when he studied what is now called the "generalized Fibonacci sequences" obtained by changing the first two terms of the sequence Fibonacci and following the same process of construction.
The simplest of them, is the sequence which has the first two terms are 1 and 3, now called the Lucas sequence (It begins with 1, 3, 4, 7, 11, 18, 29, 47 ...).

Proposal for the Construction of the Super-Increasing Sequence

The crypto systems based on the use of a Knapsack, use a considerable data contained in the latter (Bournon, 1991; Petit, 1982; Karnin and Hellman, 1983; Chor and Rivest, 1988). The generation of these data is tricky.
We propose a method to generate a super-increasing sequence (each element is greater than the sum of the above). This generation is based on the generalized Fibonaci sequence.
It is only necessary to have four elements to calculate the sequence A super-increasing for any value of N. For this, we take two initial values of the sequence A (starting values A1and A2) with two other real factors α, ß as follows:

$$A_i=\lceil A_{i-2}*\alpha + A_{i-1}*ß\rceil \qquad \forall i\geq 3 \quad (*)$$

$\lceil . \rceil$ (Floor: Integer part of A_i)+1)

Condition:
We must have: $A_1 < A_2$ and $\alpha \leq ß$
Otherwise, the super-increasing sequence would not be possible.

Examples of the generation of the super-increasing sequence A
1) For this first example we have taken the following four values:

A_1	A_2	α	ß
1	3	1.1045	1.852

The first sequence is:
1 – 3 – 6- 14 – 32 – 74 – 172 – 400 – 930 - 2164
We can check easily that this sequence is super-increasing sequence.

2) Here is a second example that we have changed the two starting coefficients. For this example, we changed the value of both coefficients α and ß

A_1	A_2	α	ß
1	3	1.0045	1.952

We found the sequence:
1 – 3 – 6 - 14 – 33 – 78 – 185 – 439 – 1042 – 2474.
We can check that this sequence is super-increasing sequence.

3) We'll now take a third example, by putting α=ß

A_1	A_2	α	ß
1	3	1.952	1.952

We found the following sequence: 1 – 3 – 7 – 19 – 50 – 134 – 359 – 962 – 2578 – 6910

This sequence is super-increasing sequence for these values.

4) We'll now take a fourth example, by putting α > ß

A_1	A_2	α	ß
1	3	1.952	1.0045

We found the following sequence: 1 – 3 – 4 – 9 – 16 – 33 – 64 – 128 – 253 - 503
This sequence is not a super-increasing sequence because the condition ($\alpha \leq$ ß) was not respected, for this example we have α> ß

5) We will now take a fifth example, by putting $A_1 > A_2$

A_1	A_2	α	ß
3	1	1.0045	1.952

We found the following sequence:
3 – 1 – 4 – 8 – 19 – 45 – 106 – 252 – 598 – 1420
At first, this sequence is not a super-increasing sequence, but after a few iterations, it checks the condition of the super-increasing sequence.

6) In this sixth example we will talk about the deviation rate between the two factors α and ß, so that the sequence A either a super-increasing sequence.

A_1	A_2	α	ß
1	3	1.0045	1.5

For these values, and after trying several times, we have successfully get this super-increasing sequence:

1 – 3 – 5 – 10 – 20 – 40 – 80 – 160 – 320 – 640

The deviation rate T between s α and ß is:

$$T = \frac{\alpha}{\beta} = \frac{1.0045}{1.5} = 0.669$$

Deduction:
If we Take ß> α so that, the sequence A of n = 10 either a super-increasing sequence, we will require that the following equation be achievable:

$$ß = \frac{\alpha}{T} \quad \text{and} \quad T < 66.9\% \quad \text{(eq.*)}$$

7) In this seventh example, we will choose α and T, one way to obtain ß (eq.*) with T=52% so that, the sequence A either a super-increasing sequence.

A_1	A_2	α	ß
1	3	1.03	1.980

We found the following sequence: 1 – 3 – 6 – 14 – 33 – 79 – 190 – 457 – 1100 – 2648
We can check that this sequence is super-increasing sequence.

8) We will change the value of T = 72% and we will check if the sequence A is a super - increasing sequence.

A_1	A_2	α	ß
1	3	1.03	1.43

We found the following sequence: 1 – 3 – 5 – 10 – 19 – 37 – 72 – 141 – 275 – 538

This sequence is not super-increasing sequence for T=72%, because it did not verify the equation (eq. *).

Remarks: Recent research dealing with this crypto system (Agarwal, 2011; Lokeshwari et al., 2011), they use systematically of any super-increasing sequences, without any details of their origin.

This proposal for the generation of a super-increasing sequence A of n elements, allowed us to gain in the fields of encryption key: we have 4 values for the A generation, plus 3 values for M, W and N. We have only 7 values for our proposal compared to the original method which requires (n+3) values.

The proposed method produces elements of the super-increasing sequence, in a manner recurring and adds the random characteristic to the latter.

Conclusion

We have transformed the public key cryptosystem of Merkle and Hellman in a secret key algorithm and for an equivalent complexity.

In the proposed algorithm, the size of the secret key encryption and the number of different values that can be used in the encryption process are set as follows:

$$ST = \{\alpha, ß, A, B, D, N\}.$$

where α and ß are double precision numbers. D (the starting index value, of the generalized Fibonaci sequence in the previous examples we took D = 1). N= (the number of samples of the super-increasing sequence), A (= A1) and B (= A2) are integer constants.
If the accuracy of calculation of α, ß (irrationals numbers), is 10^{-16} and, $A \in [1, 128]$, $B \in [1, 128]$, $D \in [1, 64]$ and $N \in [1, 64]$.

Therefore, the key space is greater than:
$10^{16} \times 10^{16} \times 128 \times 128 \times 64 \times 64$, (with $10^3 \approx 2^{10}$), in this case there will be a key field of the order of 2^{132}**,**
Thus, the encryption key length is 132 bits, and it is huge.

Therefore, the encryption algorithm has a very large key space to withstand all kinds of brute force attacks.

Also, by this process we reduced the number of bits used in the old cryptosystem because a practical implementation must been contain at least 200 terms and each term should be 200 bits.

Conflict of Interests

There is no conflict of interest regarding the publication of this paper.

Reference

Karp, R M. (1972). Reducibility Among Combinatorial Problems. In Complexity of Computer Computations. Proc. Sympos. IBM Thomas J. Watson Res. Center, Yorktown Heights, N.Y. New York: Plenum, p.85-103.

Clark, A, Dawson, E, and Bergen, H. (1996). Combinatorial Optimization And The Knapsack Cipher, Cryptologia, 20(1), 85-93.

Merkle, R, and Hellman, (1978). M. Hiding information and signatures in trapdoor knapsacks. IEEE Trans. Inform. Theory, IT-24, pp. 525- 530.

Schneier, B. (1996). Applied Cryptography-Protocols, Algorithms and Source Code in C. John Wiley & Sounds, Inc, New York, Second Edition.

Stinson, D. (2001). Cryptography – Theory and practice. Vuibert Informatics, Paris.

Allaire, G and Kaber, S M. (2002). Linear digital Algebra. Ellipses.

Schatzmann, M. (2002). Numerical Analysis, A Mathematical Introduction. Oxford University Press.

Bournon, L. (1991). Comparative study of several families of Knapsack», DEA thesis. Université d'Aix Marseille II.

Petit, M. (1982). Mathematical study of some cryptosystems: the Knapsack. Thesis presented to Rennes University.

Karnin, E, and Hellman, M. (1983). The Largest Super-Increasing Subset of a Random Set. IEEE Trans. Inform. Theory, IT-29, N°1, 146-148.

Chor, B and Rivest,R L. (1988). A knapsack type public-key cryptosystem based on arithmetic in finite fields. IEEE Trans. Inform Theory N°34/5, 901-909.

Agarwal, A. (2011). Encrypting Messages using the Merkle-Hellman Knapsack Cryptosystem. International Journal of Computer Science and Network Security, 11(5), 12-14.

Lokeshwari, G, Aparna, G, Udaya Kumar, S. (2011). A Novel Scheme for Image Encryption using Merkle-Hellman Knapsack Cryptosystem-Approach, Evaluation and Experimentation. International Journal of Computer Science & Technology, 2(4), 336-339.