# Secure Knapsack Problem Based on Continued Fraction

**Rifaat Z. Khalaf[1], Ahmed A. Muhsin[2], Taha A.Shalfon[3]**

[1]University of Diyala, Iraq
[2]University of Diyala, Iraq
[3]University of Diyala, Iraq

**Article History**: Received: 10 January 2021; Revised: 12 February 2021; Accepted: 27 March 2021; Published online: 10 May 2021

**Abstract:** Merkle–Hellman knapsack cryptosystem is a public key cryptosystem, which entails the use of two keys: public and private, the fisrt one used for encryption, while the second one used for decryption. Unfortunately, it is not secure against cryptosystems attacks,where it is broken by **Lenstra, Lenstra, and Lovasz** ( LLL Algorithm), Adi Shamir. In this paper, we propose a Knapsack–type public key cryptosystems by using a continued fraction, where the continued fraction is used to reduce the coding of plain text into two numbers, regardless of the length of the plain text. We will show that in this paper the Knapsack cryptosystems is secure against the orthogonal lattice attack(LLL Algorithm). Also, the proposed cryptosystems are secured against some attacks (brute–force attack, some known key–recovery attack, frequency attack and quantum attacks). It shows that the continued Fraction provides short plaintext and ciphertext, in which the encrypted data volume is noticed to be decreased by 60 precent, and this in turn reduces the delay time.

**Keywords:** Continued fraction, Merkle – Hellman knapsack, LLL Algorithm.

## 1. Introduction

A public key cryptosystem (PKC), which is a connotation first made known by Diffie and Hellman in their salient study [1, 2], is an essential cryptographic principle in the security domain of information and network. Conventionally, PKCs, such as RSA [3,4], and ElGamal [5,6] bear the relatively low speed obstacle which affects other applications cryptography of the public key. Because of this, designing faster PKCs has become a challenge for cryptographers. Consequently, invention of fast PKCs, like cryptosystems of knapsack-type has become one of the first schemes of the public key

The evolution of Knapsack system was first done by Merkle and Hellman [7], though several other cryptosystems of Knapsack-type are there, the considered secure ones are few, like the Chor-Rivest Knapsack system [8,9]. In the previous studies, there have been many evolved ways and there are several trapdoors for information hiding. For example, the use 0f the problems of 0-1 Knapsack [7], compact knapsack[10], multiplicative knapsack [11,12], modular knapsack [13,14], matrix cover [15], group factorization [16,17], and polynomials over GF(2) [18], Diophantine equations[19], complementing sets[20], ect. Yet, nearlly the whole used cryptosystems of Knapsack-type are subject to the attacks of low–density subset-sum [21,22,23], GCD [24], simultaneous Diophantine approximation [25] or orthogonal lattice [17].

For designing a safe knapsack–type PKC which cannot be attacked by LLL algorithm, we must ensure that in the system, we encode the message first using continued fraction, and then we encrypt it with the knapsack problem to disguise the easy knapsack problem, then through the theory of continued fraction, the output of the ciphertext is much less than the input of the plaintext, sometimes it researches about 20 percent of the plaintext. In this case, the attacker cannot obtain a loophole that enables him to attack the ciphertext. The ciphertext of the proposed method ensures the resulting encryption scheme meets strong security.

The study paper is divided into six sections; section 1 intdroduces the study, section 2 is devoted to discuss Merkle-Hellman Knapsack cryptosystem, section 3 grapples with the theory of continued fraction, section 4 presents the proposed method, security analysis is provided in section 5, and finally section 6 which sums up the conclusions of the study.

## 2. Merkle-Hellman Knapsack cryptosystem

The Merkle -Hellman Knapsack cryptosystem [26] was one of the first proposed public-key cryptosystems. A super increasing knapsack [27,28], which is a set $S$ that satisfied the condition

$s_j > \sum_{i=1}^{j-1} s_i$ , $2 \leq j \leq n$

**key generation**

choosing a super increasing knapsack $S = (s_1, s_2, \dots \dots, s_j)$, also chooseing a conversion factor $a$ and modulus $n$, where $n > \sum_{j=1}^{n} s_j$ , $gcd(n, a) = 1$

$T = s_j a \ (mod\ n) \quad for\ all\ \ j$

The private key consists of the $S$ and $a^{-1}\ (mod\ n)$.

**Encryption**

$C = M.T$ ,where M message, C ciphertext, T publickey

**Decryption**

$$C.a^{-1} = K \quad where\ K \in N\ (natural\ number)\ , \ \ K = \sum_{j=1}^{n} S_j x_j \quad , \quad x_j \in \{0,1\}$$

$we\ get\ encoded\ message \quad m = x_j$

Then by private key(S) and CF and Table 1 we ge the message "M".

## 3. Continued fraction

Continued fractions (CF) are number theory tools. The number theory is employed for providing a powerful and helpful mode to express numbers [29]. The is an infinite continued fraction in each irrational number, whereas there is a finite continued fraction in each rational number.

Simple continued fraction is the focus of this paper, common definitions and feartures of continued fraction will be discussed. A simple continued fraction might be represented in numerous forms, among which is demonstared below:

**Definition 1:** A simple (infinite) continued fraction is an expression of the form:

$$a_0 + \cfrac{1}{a_1 + \cfrac{1}{a_2 + \cfrac{1}{a_3 + \cdots}}}$$

where $a_0, a_1, a_2, \cdots$ are integers, $a_i > 0$ for all $i = 1, 2, \cdots$ , and the number $a_i$ are called partial quotients of the continued fraction.

The continued fraction can be written as $[a_1, a_2, a_3 \cdots]$.

**Theorem 1:**

(a) Every rational number can represent a finite continued fraction.

(b) Every finite continued fraction stands for a rational number.

(c) Every irrational number can singularly be repressed by an infinite continued fraction.

(d) Every infinite continued fraction represents an irrational number.

## 4. Proposal Method

As known, a knapsack cryptosystem can be broken by the LLL algorithm [30,31,32], however, in the present study, we present a novel knapsack cryptosystem procedure based on using continued fraction. The proposed procedure increases the security of knapsack cryptosystem and it makes it unattackable by the LLL algorithm. We can illustrate the proposed algorithm as in the following stepes:

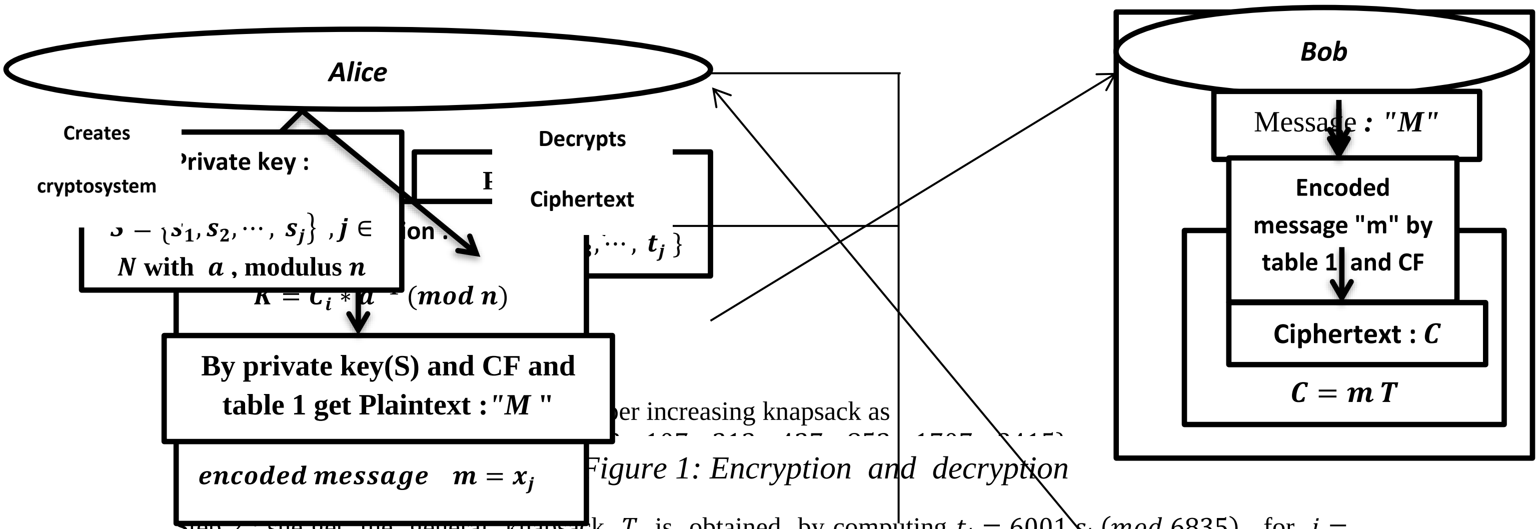
Step 1 : suppose Alice constructs her super increasing knapsack as $S = \{s_1, s_2, \cdots, s_j\}$ , $j \in N(natural\ number)$, with $a$ , modulus $n$ and $a^{-1}$.

Step 2 : Alice gets the general knapsack $T$ is obtained by computing $t_i = a\ s_i\ (mod\ n)$, for $i = 1, 2, 3, \cdots, j$. Then, $T = \{\ t_1, t_2, t_3, \cdots, t_j\}$,therefore, $T$ is the public key, whereas the private key is $S$ and $a^{-1}(mod\ n)$ .

Step 3 : suppose Bob wants encrypting the message “M” and send it to Alice, Bob uses the table of Corresponding integers to letters (Table 1) ,and he writes it in the form of the continued fraction and then he converts it to binary to get encoded message "m".

Step 4 : he computes cipher text $C$ as $C = m\ T$ , and it is sent to Alice as in figure 1.

Step 5 : Alice computes $a^{-1}$ using Euclidean Algorithm, then, she computes, $k = C * a^{-1}\ (mod\ n)\ where\ K \in N\ (natural\ number)$ and using the private key, Alice gets the encoded message and then, by continued fraction, she gets the original text as in figure 1.




...er increasing knapsack as

*Figure 1: Encryption and decryption*

Step 2 : she get the general knapsack $T$ is obtained by computing $t_i = 6001\, s_i\ (mod\ 6835)$, for $i = 0, 1, 2, \cdots, 11$. Then,

$$T = \{5167,\ 4333,\ 997,\ 2828,\ 4822,\ 3643, 6452, 68, 6137, 6273,\ 4877,\ 2085\}.$$

Step 3 : suppose Bob wants encrypting the message "Ahmed" and send it to Alice, Bob uses Table 1 to get the values A=1, H=8, M=13, E=5, D=4.

*Table 1: Corresponding integers to letters.*

| letters | [illegible] | [illegible] | [illegible] | [illegible] | [illegible] | [illegible] | [illegible] | [illegible] | [illegible] | [illegible] | [illegible] | [illegible] | [illegible] | [illegible] | [illegible] | [illegible] |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Integers Corresponding to letters | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 0 | 1 | 2 | 3 | 4 | 5 |

Step 4 : he writes them in the form of the continued fraction as below:

$$1 + \cfrac{1}{8 + \cfrac{1}{13 + \cfrac{1}{5 + \cfrac{1}{4}}}}$$

which will be equal to $\frac{2514}{2237}$, then he converts it to binary as

| Decimal | Binary |
|---|---|
| 2514 | 100111010010 |
| 2237 | 100010111101 |

Step 5 : Now, Bob computes cipher text $C$ as $\boldsymbol{C = m\,T}$

$$C_1 = 1 * t_0 + 0 * t_1 + 0 * t_2 + 1 * t_3 + 1 * t_4 + 1 * t_5 + 0 * t_6 + 1 * t_7 + 0 * t_8 + 0 * t_9 + 1 * t_{10} + 0 * t_{11}$$

Then,

$$1 * 5167 + 0 * 4333 + 0 * 997 + 1 * 2828 + 1 * 4822 + 1 * 3643 + 0 * 6452 + 1 * 68 + 0 * 6137 + 0 * 6273 + 1 * 4877 + 0 * 2085 = 21405$$

And,

$$C_2 = 1 * t_0 + 0 * t_1 + 0 * t_2 + 0 * t_3 + 1 * t_4 + 0 * t_5 + 1 * t_6 + 1 * t_7 + 1 * t_8 + 1 * t_9 + 0 * t_{10} + 1 * t_{11}$$

Then,

$$1*5167+0*4333+0*997+0*\ 2828+1*\ 4822+\ 0*\ 3643+1*6452$$
$$+1*\ 68+1*\ 6137+1*\ \ 6273\ +\ 0*4877+1*\ 2085=31004$$

Then, the ciphertext $C\ =$ {21405, 31004}, and it will be sent to Alice.

Ste 6: Alice computes $a^{-1}$ using Euclidean Algorithm, then $a^{-1}=2516\ (mod\ 6835)$.

Then, she computes, K = $C_i*a^{-1}\ (mod\ n),\ i=1,2$ as

$$21405*2516\ (mod\ 6835)=2015$$

And

$$31004*2516\ (mod\ 6835)=5044$$

Step 7 : Now, for 2015 and using the private key, Alice gets 100111010010 = 2514, and she gets 100010111101= 2237 for 5044. These values are written as $\frac{2514}{2237}$ , and she uses Euclidean Algorithm to get the following continued fraction:

$$\frac{2514}{2237}=1+\cfrac{1}{8+\cfrac{1}{13+\cfrac{1}{5+\cfrac{1}{4}}}}$$

That is, $\frac{2514}{2237}=$ [1;8,13,5,4] and using Table 1, the same original plaintext "Ahmed" .

Step 8 : assume that Miro endeavours recovering the plaintext which matches with the ciphertext $C$. As Miro knows the public key $T$ and ciphertext $C$, she needs to find a set of $u_i\ for\ i=0,1,2,\dots\dots,11$ with the restriction that each $u_i\in\{\,0\,,1\}$. Then,

$$5167u_0+4333\,u_1+997\,u_2+2828\,u_3+4822\,u_4+\ 3643u_5+6452\,u_6$$
$$+68\,u_7+6137\,u_8+6273\,u_9 4877u_{10}+2085\,u_{11}=21405$$

and,

$$5167u_0+4333\,u_1+997\,u_2+2828\,u_3+4822\,u_4+\ 3643u_5+6452\,u_6$$
$$+68\,u_7+6137\,u_8+6273\,u_9+4877u_{10}+2085\,u_{11}=31004$$

The matrix equation can be written as follows:

$$T\cdot U\ =\ C$$

Then, Miro rewrites the matrix equation as:

$$M\cdot V\ =\begin{bmatrix} I_{n\times n} & 0_{n\times 1} \\ A_{m\times n} & -B_{m\times 1} \end{bmatrix}\begin{bmatrix} U_{n\times 1} \\ 1_{1\times 1}\end{bmatrix}=\begin{bmatrix} U_{n\times 1} \\ 0_{1\times 1}\end{bmatrix}=W$$

applying the LLL algorithm to $M$. Hence, Miro detects

$$M=\begin{bmatrix} I_{12\times 12} & 0_{12\times 1} \\ T_{1\times 12} & -C_{1\times 1} \end{bmatrix}$$

$$=\begin{bmatrix}
1&0&0&0&0&0&0&0&0&0&0&0&0\\
0&1&0&0&0&0&0&0&0&0&0&0&0\\
0&0&1&0&0&0&0&0&0&0&0&0&0\\
0&0&0&1&0&0&0&0&0&0&0&0&0\\
0&0&0&0&1&0&0&0&0&0&0&0&0\\
0&0&0&0&0&1&0&0&0&0&0&0&0\\
0&0&0&0&0&0&1&0&0&0&0&0&0\\
0&0&0&0&0&0&0&1&0&0&0&0&0\\
0&0&0&0&0&0&0&0&1&0&0&0&0\\
0&0&0&0&0&0&0&0&0&1&0&0&0\\
0&0&0&0&0&0&0&0&0&0&1&0&0\\
0&0&0&0&0&0&0&0&0&0&0&1&0\\
5167&4333&997&2828&4822&3643&6452&68&6137&6273&4877&2085&-C
\end{bmatrix}$$

where, $C=\{21405,\ 31004\}$. The output of LLL algorithm is a matrix $M'$, made of short vectors in the lattice extended by the matrix M columns.

step 9 : Now for the case: $-\,C\ =\ -21405$, Miro obtains

$$M' = \begin{bmatrix} -1 & 0 & 1 & 2 & -2 & 0 & -2 & 1 & 0 & -2 & 2 & -1 & 1 \\ 1 & 0 & -1 & -1 & 1 & 0 & 0 & 1 & -1 & 1 & -1 & 1 & -1 \\ -2 & -2 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & -1 \\ 1 & -1 & 0 & 0 & 0 & 0 & 1 & -1 & 0 & -1 & -1 & 1 & 0 \\ 0 & 1 & 0 & -2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & -2 & 1 & 0 & 0 & -1 & -2 & 1 & 1 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -2 & -2 & 0 & -1 & 0 & 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 & 1 & -1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & -1 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 1 & -2 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 & -2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & -1 & -1 & 0 & 0 & -1 \end{bmatrix}$$

Therefore, Miro failed to get the solution $U = 100111010010$.

And for the case : $-C = -31004$.

$$M' = \begin{bmatrix} -1 & 0 & 1 & 2 & -2 & 0 & -2 & 1 & 0 & -1 & -1 & -1 & -1 \\ 1 & 0 & -1 & -1 & 1 & 0 & 0 & 1 & -1 & 1 & 1 & -1 & 2 \\ -2 & -2 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & -1 & -1 & -1 & -3 \\ 1 & -1 & 0 & 0 & 0 & 0 & 1 & -1 & 0 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & -2 & 0 & 0 & 0 & 0 & 0 & 0 & 2 & 0 & 3 \\ 0 & 0 & -2 & 1 & 0 & 1 & -1 & -2 & 1 & 1 & -1 & 0 & -1 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & -2 & -2 & 0 & -1 & 0 & 2 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & -1 & 0 & 1 & 0 & -1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & -1 & 1 & 0 & 2 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 & -2 & 0 & -1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & -1 & -1 & 0 & 0 & 0 \end{bmatrix}$$

Also, in this case, Miro failed to get the solution $U = 100010111101$. Therefore, She failed to obtain the plaintext.

## 5. Security analysis

- Through the above example, we proved that the proposed algorithm cannot be attacked by LLL algorithm.
- In this paragraph we are trying to show if there were other attacks against the proposed Algorithm. If the attacker tries to obtain the encrypted text C = U T, he cannot obtain the plaintext of the message, because it represents a coded message and does not represent the original text of the message. but if the attacker tries to find the value of C = U. T, C = U. M. S, he cannot get the plaintext of the message because the values (M, S, U) are unknown.
- Continued fraction should reduce some attacks effectiveness. A well-known process of cryptanalysis is the analysis of frequency, which counts on detecting repeated data. Wanton force attacks run by attempting to take several keys and decrypting the data and making sure if the data of the output is of any significance. By CF first, an attacker has to go through decrypting the data, thereafter decoding it before checking if the output data make any sense. He will go through a longer highly demanding process, and if he ignorant of the coding of the data at all, he most probably never break the encryption.
- Quantum attack cannot attack our proposed system for three reasons [33][34]:

**First:** If the enemy is able to obtain the ciphertext, then he cannot obtain the plaintext because the message was encoded and then encrypted even using quantum computers.

**Second:** The ciphertext resulted from the encoding process is like data compression, for example if the letters of the plaintext of the message is 60 letters, and the block size is 5, then the output of the ciphertext is about 20 letters, meaning about a third of the message information is hidden on the attacker, and the greater the block size, the greater the amount of hidden information is.

**Third:** the length of the block must be divided (60); so, the number of letters in a block could is: 1, 2, 3, 4, 5, 6, 10, 12, 15, 20, 30, 60, there is no evident padding.

## 6. Simulation and Results

In the beginning we used the knapsack algorithm for data in the range of (10 k.B - 410 k.B) through Table 2 and figure 2, we noticed that the more encrypted data, the greater the delay time is. While when using the algorithm of knapsack whith CF for data, we noticed that the encrypted data volume decreased by 60 precent, and this in turn reduces the delay time, as is evident from Table 2 and figure 2, [hint using computer specification : cpu:intel core i5 g1 2.27GH7, RAM:8GB, Windows7 64bihs, simulation: omnet++5].

Table 2 : Knapsack vs Knapsack whith CF Latency

| Delay Link | Data size(B) | Delay process | Delay transmit | Latency-Knapsack(s) | Latency -Knapsack-CF(s) |
|---|---|---|---|---|---|
| 0 | 10000 | 4.5 | 0 | 4.5 | 4.31872 |
| 0.07872 | 20000 | 9 | 1.6 | 10.67872 | 6.415105 |
| 0.055105 | 30000 | 13.5 | 2.4 | 15.955105 | 8.583816 |
| 0.103816 | 40000 | 18 | 3.2 | 21.303816 | 10.657553 |
| 0.057552 | 50000 | 22.5 | 4 | 26.557552 | 12.942352 |
| 0.222352 | 60000 | 27 | 4.8 | 32.022354 | 15.171492 |
| 0.331491 | 70000 | 31.5 | 5.6 | 37.431492 | 17.00836 |
| 0.04836 | 80000 | 36 | 6.4 | 42.44836 | 19.236889 |
| 0.15689 | 90000 | 40.5 | 7.2 | 47.856892 | 21.275269 |
| 0.075267 | 100000 | 45 | 8 | 53.075268 | 23.403942 |
| 0.083943 | 110000 | 49.5 | 8.8 | 58.383942 | 25.699825 |
| 0.259825 | 120000 | 54 | 9.6 | 63.859825 | 27.567368 |
| 0.007369 | 130000 | 58.5 | 10.4 | 68.907372 | 29.689116 |
| 0.009116 | 140000 | 63 | 11.2 | 74.209114 | 31.802044 |
| 0.002043 | 150000 | 67.5 | 12 | 79.502045 | 34.098747 |
| 0.178749 | 160000 | 72 | 12.8 | 84.978752 | 36.190578 |
| 0.150578 | 170000 | 76.5 | 13.6 | 90.25058 | 38.364037 |
| 0.204031 | 180000 | 81 | 14.4 | 95.604034 | 40.66452 |
| 0.384522 | 190000 | 85.5 | 15.2 | 101.084518 | 42.560524 |
| 0.160524 | 200000 | 90 | 16 | 106.160522 | 44.581894 |
| 0.061893 | 210000 | 94.5 | 16.799999 | 111.361893 | 46.791653 |
| 0.151654 | 220000 | 99 | 17.6 | 116.751656 | 48.772591 |
| 0.012587 | 230000 | 103.5 | 18.4 | 121.91259 | 50.982143 |
| 0.102143 | 240000 | 108 | 19.200001 | 127.302139 | 53.015472 |
| 0.0154 | 250000 | 112.5 | 20 | 132.515472 | 55.4094 |

| | | | | | |
|---|---|---|---|---|---|
| 73 | | | | | 39 |
| 0.289442 | 260000 | 117 | 20.799999 | 138.089447 | 57.313782 |
| 0.073783 | 270000 | 121.5 | 21.6 | 143.173782 | 59.413559 |
| 0.053557 | 280000 | 126 | 22.4 | 148.453552 | 61.510727 |
| 0.030728 | 290000 | 130.5 | 23.200001 | 153.730728 | 63.748825 |
| 0.148825 | 300000 | 135 | 24 | 159.148819 | 65.780907 |
| 0.060908 | 310000 | 139.5 | 24.799999 | 164.360916 | 67.924034 |
| 0.084033 | 320000 | 144 | 25.6 | 169.684036 | 69.961899 |
| 0.001897 | 330000 | 148.5 | 26.4 | 174.901886 | 72.17614 |
| 0.096138 | 340000 | 153 | 27.200001 | 180.296143 | 74.294701 |
| 0.0947 | 350000 | 157.5 | 28 | 185.594696 | 76.415955 |
| 0.095955 | 360000 | 162 | 28.799999 | 190.89595 | 78.727783 |
| 0.287791 | 370000 | 166.5 | 29.6 | 196.387802 | 80.674515 |
| 0.114514 | 380000 | 171 | 30.4 | 201.514511 | 82.724548 |
| 0.044552 | 390000 | 175.5 | 31.200001 | 206.744553 | 84.85746 |
| 0.057453 | 400000 | 180 | 32 | 212.057449 | 87.039619 |
| 0.119611 | 410000 | 184.5 | 32.799999 | 217.419617 | 92.518771 |

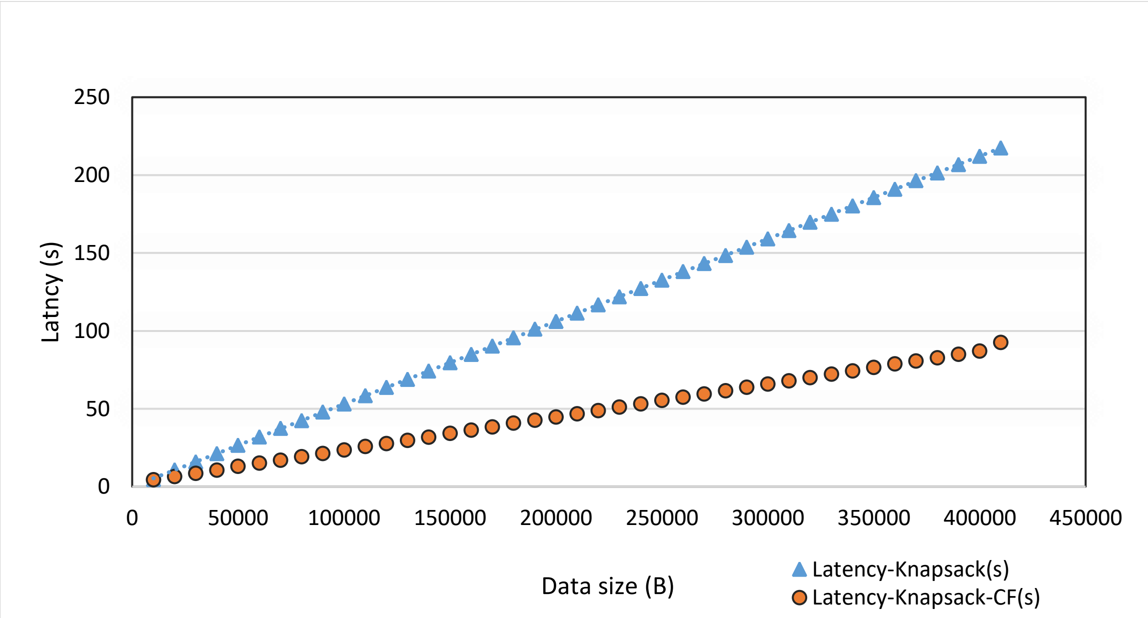



Figure 2: Data size (B) vs Latency(S)

## 7. Conclusion

In our proposed algorithm, the concept of Continued Fraction was used to increase the security of Merkle – Hellman Knapsack cryptosystem, so that it cannot be attacked by LLL algorithm.

Another benefit of using Continued Fraction is offering shorter ciphertext and plaintext, thus decreasing the amount of time required for encrypting, decrypting, and transmiting data. The decreased redundancy in the plaintext can potentially inhibit certain cryptanalysis attacks.

**References**

1. Cherowitzo, William (2002-03-02). "Merkle-Hellman Knapsack Cryptosystem". Math 5410 - Modern Cryptology. Retrieved 2019-08-18.
2. Diffie, W.; Hellman, M.E. New Directions in Cryptography .IEEE Trans. Inf. Theory1976, IT-22, 644–654
3. Rivest, R.L.; Shamir, A.; Adleman, L.M. A Method for Obtaining Digital Signature and Public Key Cryptosystems. Commun. ACM1978, 21, 120–126.
4. Smart, Nigel (February 19, 2008). "Dr Clifford Cocks CB". Bristol University. Retrieved August 14, 2011.
5. ElGamal, T. A Public Key Cryptosystem and a Signature Scheme Based on Discrete Logarithms.IEEE Trans.Inf. Theory1985, IT-31, 469–472.
6. Mike Rosulek (2008-12-13). "Elgamal encryption scheme". University of Illinois at Urbana-Champaign. Archived from the original on 2016-07-22.
7. Merkle, R.C.; Hellman, M.E. Hiding Information and Signatures in Trapdoor Knapsacks.IEEE Trans. Inf. Theory1978, IT-24, 525–530.
8. Chor, B.; Rivest, R.L. A Knapsack-Type Public Key Cryptosystem Based on Arithmetic in Finite Fields.IEEE Trans.Inf. Theory1988, IT-34, 901–909.
9. Vaudenay, S. Cryptanalysis of the Chor–Rivest Cryptosystem. J. Cryptol.2001, 14, 87–100.
10. Orton, G. A Multiple-Iterated Trapdoor for Dense Compact Knapsacks. In Advances in Cryptology–Eurocrypt 1994(LNCS); Springer-Verlag: Perugia, Italy, 1995; Volume 950, pp. 112–130.
11. Morii, M.; Kasahara, M. New Public Key Cryptosystem Using Discrete Logarithm OverGF(p). IEICE Trans. Fund.1988, J71-D, 448–453.
12. Naccache, D.; Stern, J. A New Public-Key Cryptosystem. In Advances in Cryptology–Eurocrypt 1997 (LNCS); Springer-Verlag: Konstanz, Germany, 1997; Volume 1233, pp. 27–36.
13. Goodman, R.M.F.; McAuley, A.J. New Trapdoor-Knapsack Public-Key Cryptosystem.IEE Proc.1985, 132 Pt E, 282–292.
14. Niemi, V. A New Trapdoor in Knapsacks. In Advances in Cryptology–Eurocrypt 1990 (LNCS); Springer-Verlag: Aarhus, Denmark, 1990; Volume 473, pp. 405–411.
15. Janardan, R.; Lakshmanan, K.B. A Public-Key Cryptosystem based on The Matrix Cover NP-Complete Problem. In Advances in Cryptology–Crypto 1982; Plenum: New York, NY, USA, 1983; pp. 21–37.
16. Blackburn, S.R.; Murphy, S.; Stern, J. Weaknesses of A Public Key Cryptosystem based on Factorization of FiniteGroups. InAdvances in Cryptology–Eurocrypt 1993 (LNCS); Springer-Verlag: Lofthus, Norway, 1994; Volume 765, pp. 50–54.
17. Nguyen, P.; Stern, J. Merkle-Hellman Revisited: A cryptanalysis of The Qu-Vanstone Cryptosystem based onGroup Factorizations. InAdvances in Cryptology–Crypto 1997 (LNCS); Springer-Verlag: Santa Barbara, CA, USA, 1997; Volume 1294, pp. 198–212. Information2019, 10, 7526 of 27.
18. Pieprzyk, J.P. On Public-Key Cryptosystems, Built Using Polynomial Rings. In Advances in Cryptology–Eurocrypt 1985(LNCS); Springer-Verlag: Linz, Austria, 1985; Volume 219, pp. 73–80.
19. Lin, C.H.; Chang, C.C.; Lee, R.C.T. A New Public-Key Cipher System based upon The Diophantine Equations. IEEE Trans. Comput.1995, 44, 13–19.
20. Webb, W.A. A Public Key Cryptosystem based on Complementing Sets.Cryptologia1992, XVI, 177–181.
21. Brickell, E.F. Solving Low Density Knapsacks. In Advances in Cryptology–Crypto 1983; Plenum: New York, NY, USA, 1984; pp. 24–37.
22. Lagarias, J.C.; Odlyzko, A.M. Solving Low-Density Subset Sum Problems.J. ACM1985,32, 229–246.
23. Coster, M.J.; LaMacchia, B.A.; Odlyzko, A.M.; Schnorr, C.P. An Improved Low-Density Subset Sum Algorithm. In Advances in Cryptology–Eurocrypt 1991 (LNCS); Springer-Verlag: Brighton, UK, 1991; Volume 547, pp. 54–67.

24. Brickell, E.F.; Odlyzko, A.M. Cryptanalysis: A Survey of Recent Results. In Contemporary Cryptology, the Science of Information Integrity; IEEE Press: New York, NY, USA, 1992; pp. 501–540.
25. Lagarias, J.C. Knapsack Public Key Cryptosystems and Diophantine Approximation. In Advances in Cryptology–Crypto 1983; Plenum: New York, NY, USA, 1984; pp. 3–23.
26. R.Merkl and M.Hellman, Hiding information and signatures in trapdoor knapsacks, IEEE Transactions on Information theory, Vol. IT-24, No.5, 1978, pp. 525-530 .
27. Richard A. Mollin, An Introduction to Cryptography (Discrete Mathematical & Applications), Chapman & Hall/CRC; 1 edition (August 10, 2000), ISBN 1-58488-127-5
28. Bruce Schneier, Applied Cryptography: Protocols, Algorithms, and Source Code in C, pages 463-464, Wiley; 2nd edition (October 18, 1996), ISBN 0-471-11709-9
29. Rosen K. Elementary Number Theory and its Applications. Addison-Wesley : New York, 2005**.**
30. Nguyen, Phong Q.; Stehlè, Damien (September 2009). "An LLL Algorithm with Quadratic Complexity". SIAM J. Comput. 39 (3): 874–903. doi:10.1137/070705702. Retrieved 3 June 2019
31. Divasón, Jose. "A Formalization of the LLL Basis Reduction Algorithm". Conference paper. Retrieved 3 May 2020.
32. Regev, Oded. "Lattices in Computer Science: LLL Algorithm" (PDF). New York University. Retrieved 1 February 2019.
33. Rifaat Z. Khalaf "Quantum enecryption algorithim based on modified BB84 and authentication DH algorithm ": August 2015**.**
34. Alharith A. Abdullah "Modified Quantum Three Pass Protocol Based on Hybrid Cryptosystem." 2015.