# Proposal for Two Enhanced NTRU

Ahmed Tariq Sadiq
Computer Science Department
University of Technology
Baghdad, Iraq

Najlaa Mohammad Hussein
Computer Science Department
Baghdad University
Baghdad, Iraq

Suha Abdul Raheem Khoja
Electronic and communication Engineering Department
Baghdad University
Baghdad, Iraq

***Abstract*—Sound is very widely used in communication. In order to ensure secure communication a cryptographic data scheme is used. Secure sound is needed in many fields such as military, business, banking and electronic commerce. There is also an increasing demand for secured sound in network communication. Several symmetric and asymmetric algorithms are used for sound encryption. In this work, NTRU, the last in line public key cryptosystem is enhanced in two methods and used for encrypting sound files after converting the sound into text. In the proposed methods the message is encrypted one character at a time, since NTRU encrypts only prime numbers, thus 7 bits of each character is encrypted and the eighth bit is left without encryption. In method I NTRU algorithm is enhanced by adding the result obtained from calculating a mathematical equation of one variable to the message and then the resulted encrypted bit is fed-back and added to the next bit of the message in the next step; this procedure is repeated for the subsequent bits of the message. In method II NTRU algorithm is enhanced by adding the subsequent states of LFSR (Linear Feedback Shift Register) to the subsequent bytes of the message. The proposed methods are tested on several sound files; the results show that the proposed methods I and II maintain approximately the same original method encryption and decryption time while generating more complex encryption.**

***Keywords*—NTRU; security; sound**

## I. Introduction

NTRU (Number theory Research unit) algorithm is a public key cryptosystem invented by three professors of mathematics from brown university of America Jeffrey Hoffstin, Jill Pipher and Joseph H. Silverman in 1996. [1] NTRU is built on polynomial algebra. The basic objects are truncated polynomials in the ring $R=Z[X] / (X^N-1)$ and the basic tool is the reduction of polynomials with respect to two relatively prime modulo. The security of the system is (hoped to be) based on the difficulty of finding a "short" factorization for such polynomials. This latter problem is equivalent to finding a short vector in a certain 2N dimensional lattice, a commonly known and also widely studied hard problem. [2] Since NTRU is a ring based public key cryptosystem and is therefore quite different from the group based cryptosystems whose security relies on the integer factorization problem or the discrete logarithm problem. This extra structure can be exploited to obtain a very fast cryptosystem; to encrypt/decrypt a message block of length N, NTRU only requires $O(N^2)$ time, whereas the group based schemes like RSA etc. requires $O(N^3)$ time. Furthermore, NTRU also has a very short key size of O(N) and very low memory requirements, which makes it ideal for constrained devises such as smart cards. [3]

The rest of this paper is organized as follows: related work is given in section II, section III provides a brief description of NTRU algorithm, the proposed methods are described in section IV, section V presents the experimental results and finally conclusions and future work are given in section VI.

## II. Related Work

Jaspreet Kaur and Er. Kanwal preet Singh [4] use three different kinds of algorithms NTRU, RSA and RINGDAEL for speech encryption and decryption by first converting the speech into text then further the text is converted into cipher text. The performances are analyzed of these three approaches respectively the parameters calculated are encryption, decryption, delay time, complexity, packet lost and security levels. In these three approaches, encryption decryption and delay time are varied according to the number of bits per second.

On the other hand, complexity and packet lost are approximately the same. There is no packet lost during transmitting and receiving the data. Also, Jaspreet Kaur and Er. Kanwal preet Singh [5] use three different kind of techniques i.e. MD-5, SHA-2 and RINGDAEL for speech encryption, where the speech is first converted into text then the text is converted into cipher text. At the end, the performances of these three approaches are analyzed, respectively.

## III. Brief Description of Ntru Algorithm

### A. Prameters

NTRU has three integer parameters N, p and q. N represents the degree of the polynomials at most N-1, p and q are used to reduce the coefficients of the polynomials, p is smaller than q and they have no common divisor. [6, 7, 8]

### B. Key generation

Sending a secret message from Bob to Alice requires the generation of a public and private key. The public key is known by both Alice and Bob and the private key is only known by Alice. To generate the key pair two polynomials f and g with coefficients much smaller than q, with degree at most N-1 and with coefficients in {-1, 0, 1} are required.

The polynomial f must satisfy the requirement that the inverses modulo q and modulo p exist, which means that $f*fp = 1 \pmod p$ and $f*fq = 1 \pmod q$ must hold. So when the chosen f is not invertible Alice has to go back and try another f. Both f and fp is Alice private key. The public key h is generated by computing $h = fq * g \pmod q$. [9]

### C. Encryption

When Bob wants to send a secret message to Alice, he puts his message in the form of a polynomial m with coefficients between $-1/2p$ and $1/2p$ .Next Bob randomly chooses another small polynomial r. This is the blinding value which is used to obscure the message. Bob uses the message m; randomly chosen polynomial r and Alice's public key h to compute the polynomial e = r * h + m (mod q). The polynomial e is the encrypted message which Bob sends to Alice. [10, 11]

### D. Decryption

In addition to the publically available information Alice knows her own private key, on receiving Bob's cipher text, Alice start the decryption process by computing the polynomial a = f * e (mod q). [12] She then shift the coefficient of polynomial a to the range $\left(-q/2, q/2\right)$[13] and does a mod p computation to obtain: d = fp * a (mod p). Assuming that the parameters have been chosen properly then the polynomial d must be equal to Bob plain text m. [12]

## IV. Proposed Methods

### A. methodI

In this method Alice and Bob agree on a mathematical equation of one variable say (x), the value of this variable is send via one of the key establishment protocols. [14]

Bob start the encryption process by calculating the result of the mathematical equation and assigning it to a variable say (v), the value of this variable is added to the message (m), then for each bit of the encrypted message (e) the value of the previous e is assigned to the variable v, this means for each bit the value of the encrypted message is fed-back and added to the new value of m. Adding the mathematical equation to the message makes the encryption process more complex especially if the degree of this equation is high. The pseudo code of encryption process in the original NTRU algorithm [15, 13] is enhanced in this method and listed in pseudo code (1) as follows:

**Pseudo code 1 Encode (N, q, r, m, h, e, x)**

```
Require: N, q, Public Key h, message m, and random polynomial r.
1:  v = calculate the result of mathematical equation of one
        variable
2:  Star Multiply (r, h, e, N, q)
3:  for i = 0 to N - 1 do
4:    e[i] = e[i] + m[i] + v mod q
5:    v = e[i]
6:  end for
7:  {Encode returns the encrypted message, e, through the argument list.}
```

When Alice starts the decryption process, she also calculates the result of the mathematical equation and assigns it to a variable, but instead of adding the result of the mathematical equation to the message she subtracted it form the encrypted message, then for each bit the value of the encrypted message is fed-back and subtracted from the new value of the message. The pseudo code of the decryption process in the original NTRU algorithm is enhanced in this method and listed in pseudo code (2) as follows:

**Pseudo code 2 Decode (N, q, p, f, fp, e, d, x)**

```
Require: N, q, p, secret key f, inverse polynomial fp, and encrypted message e.
1:  v = calculate the result of mathematical equation of one
        variable
2:  for i = 0 to N - 1 do
3:    vv = e[i]
4:    e[i] = e[i] – v mod q
5:    v= vv
6:  end
7:  Star Multiply (f, e, a, N, q)
8:  for i = 0 to N - 1 do
9:    if a[i] < 0 then
10:     a[i] = a[i] + q {Make all coefficients positive}
11:   end if
12:   if a[i] > q/2 then
13:     a[i] = a[i] - q {Shift coefficients of a into range (-q/2,
          q/2)}
14:   end if
15: end for
16: Star Multiply (a, fp, d, N, p)
17: {Decode returns the decrypted message, d, through the argument list.}
```

### B. methodII

In this method pseudo random bits are generated with a LFSR [16], Alice and Bob agree on the initial state of the LFSR. The generation of LFSR is shown in pseudo code (3)

**Pseudo code 3 LFSR**

```
1:  for i = N down to 2 do
2:    lfsr (i) = lfsr (i-1)
3:  end for
4:  lfsr (1) = xor (lfsr (3), lfsr (5))
```

To encrypt the message Bob adds the initial state of the LFSR to the first byte of the message. The subsequence states of the LFSR are then added to the subsequent bytes of the message. The changes that are made to the pseudo code of encryption process in the original NTRU algorithm are shown in pseudo code (4) as follows:

**Pseudo code 4 Encode (N, q, r, m, h, e, lfsr)**

```
Require: N, q, Public Key h, message m, and random polynomial r.
1:  Star Multiply (r, h, e, N, q)
2:  for i = 0 to N - 1 do
3:    e[i] = e[i] + m[i] + lfsr[i] mod q
4:  end for
5:    {Encode returns the encrypted message, e, through the argument list.}
```

To decrypt the message Alice repeats the same steps that are followed by Bob but instead of adding the states of the LFSR to the message, she subtracted the states from the subsequent bytes of the message. The changes that are made to the pseudo code of the decryption process in the original NTRU are shown in pseudo code (5) as follows:

**Pseudo Code 5 Decode (N, q, p, f, fp, e, d, lfsr)**

```
Require: N, q, p, secret key f, inverse polynomial fp, and encrypted message e
1:  for i = 0 to N - 1 do
2:    e[i] = e[i] – lfsr[i] mod q
3:  end
4:  Star Multiply (f, e, a, N, q)
5:  for i = 0 to N - 1 do
```

```
6:      if a[i] < 0 then
7:        a[i] = a[i] + q {Make all coefficients positive}
8:      end if
9:      if a[i] > q/2 then
10:       a[i] = a[i] - q {Shift coefficients of a into range (-
          q/2, q/2)}
11:     end if
12:   end for
13:   Star Multiply (a, fp, d, N, p)
14:   {Decode returns the decrypted message, d, through the argument list.}
```

## V. Experimental Results

In this work the sound is first converted into text then further the text is converted into cipher text. This method can be applied to any kind of sound files after storing the file in a text editor such as note pad. This method is applied to the original NTRU algorithm (namely original method) and to the two proposed methods. The sound is converted into text via ISO-8859-1: 8-bit single-byte coded graphic character sets - Part 1: Latin alphabet No. 1, is part of the ISO/IEC 8859 series of ASCII-based standard character encodings. It is generally intended for "Western European" languages [17].

In the original method and proposed methods I and II the message is partitioned into characters, each character is encrypted separately. Since N must be a prime number, 7 bit of the character is encrypted and the eighth bit is left without encryption. The method is tested for different values of N, the results show that the maximum value for N in this method is 47; it is also shown that if the value of N is increased, encryption and decryption time will also increase. The original method and the proposed methods are tested on 25 wave sound files of sizes ranging from 10 KB to 1MB. The encryption and decryption time in seconds is computed for each one of the 25 files 25 times and then average of computation is taken to increase the accuracy of calculation.

Fig. 1, Fig. 2 and Fig. 3 respectively, displays the effect of file size on the time of encryption and decryption of the original method proposed method I and proposed method II respectively. Fig. 4 displays a comparison for the effect of file size on the time of encryption of the original method, proposed method I and proposed method II. Fig. 5 displays a comparison for the effect of file size on time of decryption of the original method proposed method I and proposed method II.

## VI. Conclusions and Future Work

Proposed method I enhanced the original NTRU algorithm by adding the results obtained from calculating a mathematical equation of one variable to the message. This led to a more complex encryption while maintaining approximately the same original algorithm encryption and decryption time. Proposed method II enhanced the original NTRU algorithm by adding the states of the LFSR to the bytes of the message. This maintains approximately the same original method encryption and decryption time while generating more complex cipher. The time needed for encryption and decryption in proposed method I and II is approximately the same in spite of the different values that are added to the message in each method. In future research, Apply the proposed methods on Field Programmable Gate Array (FPGA) which has higher speed when compared to the standard processors.

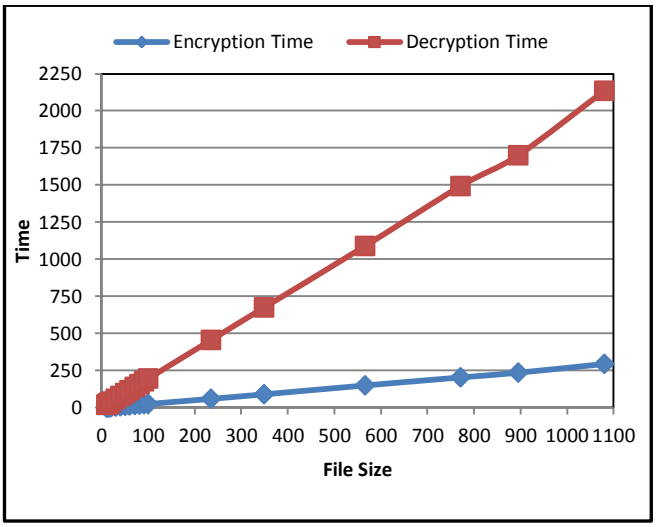


Fig. 1. The Effect Of File Size On The Time Of Encryption And Decryption Of The Original Method

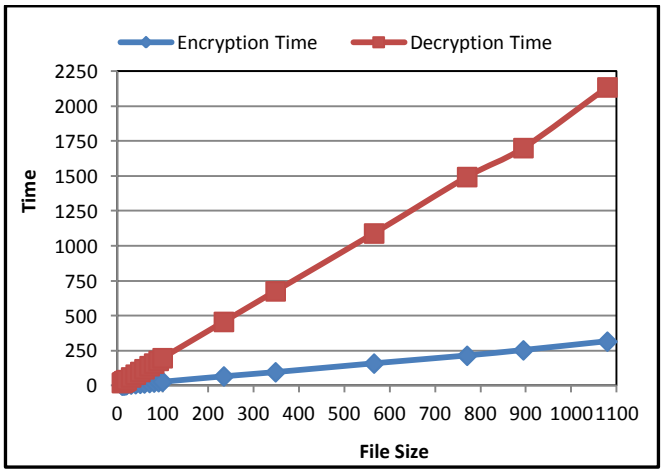


Fig. 2. The Effect Of File Size On The Time Of Encryption And Decryption Of Proposed Method I

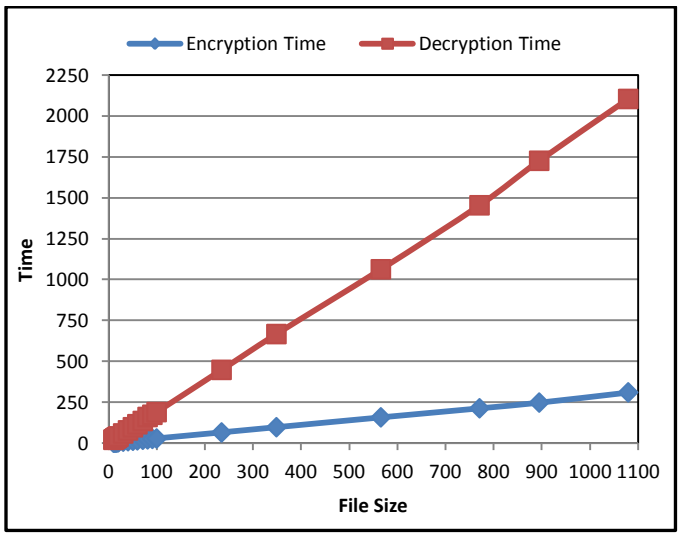

Fig. 3. The Effect Of File Size On The Time Of Encryption And Decryption Of Proposed Method Ii

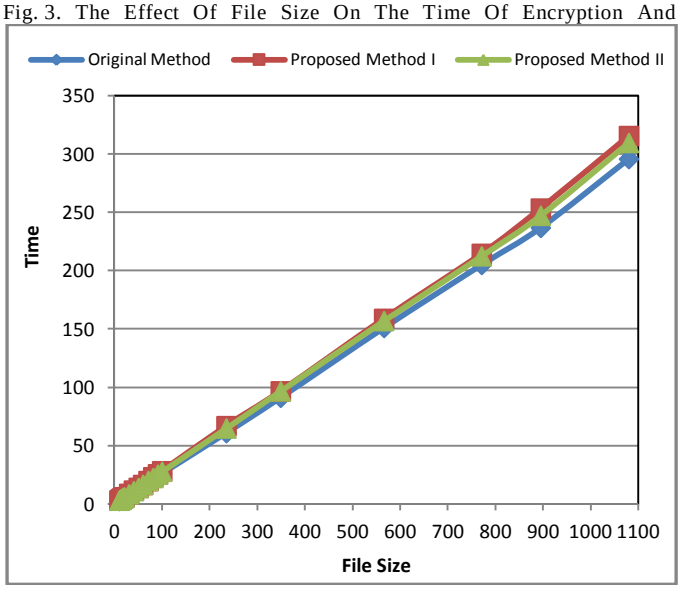


Fig. 4. Compression For The Effect Of File Size On Time Of Encryption Of The Original Method And The Two Proposed Methods

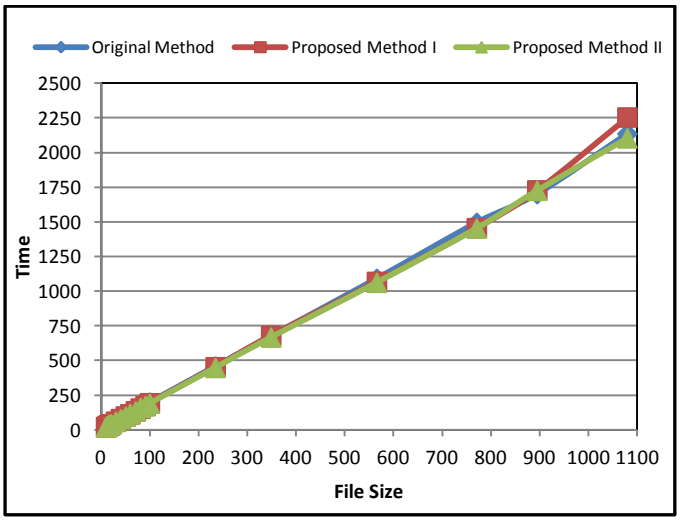


Fig. 5. Compression For The Effect Of File Size On The Time Of Decryption Of The Original Method And The Two Proposed Methods

## REFERENCES

[1] Yue, B. S., Yan, Z. H., Rruchuan, W., "A algorithm of dynamic patterns for NTRU", IEEE, 2008.

[2] Nayak, R., Pradhan, J., Sastry, C. V."A matrix formulation for NTRU cryptosystem", IEEE 2008.

[3] Jha, R., Saimi, Anil, K., "A comparative analysis and enhancement of NTRU algorithm for network security and performance improvement" International Conference on Communication Systems and Network Technologies, 2011.

[4] Kaur, J., Singh, K. p. "Comparative study of speech encryption algorithms using mobile applications", International Journal of Computer Trends and Technology, Vol.4, No. 7, Jully 2013, pp. 2346 - 2350.

[5] Kaur, J., Singh, K. p. "Speech to text encryption using cryptography techniques", International Journal of Innovative Research and Development, Volume 2, N0. 4, pp. 274 - 283, April 2013.

[6] Shen, X., Du, Z., Chen, R., "Research on NTRU algorithm for mobile java security" IEEE 2009, pp.366 - 369.

[7] Ramajaneyulu, S., Nayak, R., "Secure mobile system using NTRU encrypt", International Journal of Computer Trends and Technology, Volume 4, Issue 2, 2013.

[8] Manasa, C., Masheswar, M.V.S.N. "Secure mobile IM system using NTRU", Intrernation Jornal of Engineering Research and Technology, Volume 1, No.9, November 2012.

[9] Kumar, R. G. V. S., Jumar, N. K., Sekhar, C. P., Numma, B. V. V. S., Kumar, V. B., "Modified mutual authentication and key agreement protocol based on NTRU cryptography for wireless communications", International Journal of Computer Science and Network (IJCSN), Volume 1, Issue 4, August, 2012.

[10] Reddy, A. N., Nayak, R., Baboo, S., "Analysis and performance characteristics of cryptosystem using image files", International Journal of Computer Applications , pp.0975 – 8887, Volume 51 – No. 22, August 2012.

[11] Narasimham, C., Pradhan, A., "Evaluation of performance characteristics of cryptosystem using text files" Journal of Theoretical and Applied Information Technology, JATIT, 2008.

[12] Jeffrey, H., Pipher, J., Silverman, J. H., "An introduction to mathematical cryptography", Springer, New York, 2008.

[13] O'Rourke, C. M., "Efficient NTRU implementations" Thesis, April, 2002.

[14] Paar, C., Pelzl, J., "Understanding cryptography: a textbook for student and pracittionners", Springer-Verlag Berlin Heidelberg, 2010.

[15] Yadav, S. K., Bhardwaj, K.,"On NTRU implementation: an algorithmic approach",Proceedings of the 4th National Conference; INDIA, 2010.

[16] Beker, H., Piper, F., "Cipher systems the protection of communications", Northwood Books: London, 1982.

[17] ISO /IEC JTC 1/SC 2/WG 3 7bit and 8bit codes and their extension SECRETARIAT: ELOT, 1998.