Cubic Root Extractors of Gaussian Integers and Their Application in Fast Encryption for Time-Constrained Secure Communication

Boris Verkhovsky
Computer Science Department, New Jersey Institute of Technology, Newark, USA
E-mail: verb73@gmail.com
Received March 2, 2011; revised April 12, 2011; accepted April 15, 2011

Abstract

There are settings where encryption must be performed by a sender under a time constraint. This paper describes an encryption/decryption algorithm based on modular arithmetic of complex integers called Gaussians. It is shown how cubic extractors operate and how to find all cubic roots of the Gaussian. All validations (proofs) are provided in the Appendix. Detailed numeric illustrations explain how to use the method of digital isotopes to avoid ambiguity in recovery of the original plaintext by the receiver.

Keywords: Cryptographic Protocol, Secure Communication, Time-Constrained Encryption, Cubic Root Extractor, Gaussian Integers, Modular Arithmetic, Prefix/Suffix Positioning, Digital Isotope, Quadratic Residue, Jacoby Symbol

1. Introduction

This paper describes a cryptographic algorithm based on the extraction of cubic roots from complex numbers \( a + bi \) with integer components \( a \) and \( b \). Such complex integers are called Gaussian integers (Gaussians, for short) [1]. Let’s denote \( (a, b) := a + bi \) and \( N := a^2 + b^2 \), where \( N \) is called a norm of \((a, b)\). In modular arithmetic based on Gaussians, if \( p \) is a prime and \( N \) mod \( p \neq 0 \), then for every integer \( a \) and \( b \) holds an equivalent of the Fermat identity [2]:

\[
(a, b)^{p^2 - 1} \text{ mod } p = (1, 0) = 1. 
\]

This means that the cycles in Gaussian modular arithmetic have order \( O\left(p^2\right) \), while the cycles in modular arithmetic based on real integers have order \( O(p) \). Application of Gaussians for ElGamal cryptosystem is considered in [3]; and the RSA digital signature is described in [4]. Public key cryptography based on cubic roots of real integers is provided in [5] and in [6].

Definition1: A Gaussian integer \((x, y)\) is called the cubic root of \((a, b)\) modulo integer \( n \), and defined as \( \sqrt[3]{(a, b)} \) mod \( n \), if

\[
(x, y)^3 = (a, b) \text{ mod } n. \tag{2} 
\]

Proposition1: If \( p \) is a prime, \( p \text{ mod } 12 = 1 \) and

\[
V := (a, b)^{(p-1)/3} \text{ mod } p = (1, 0) = 1, \tag{3} 
\]

then there exists a cubic root of \((a, b)\) modulo \( p \).

Proposition2: If \( p \text{ mod } 12 = 5 \), then for every integer \( a \) and \( b \) there exists an unique cubic root of \((a, b)\) modulo \( p \).

Proposition3: If \( p^2 \text{ mod } 9 \neq 1 \) and

\[
W := V^p \text{ mod } p = (1, 0) = 1, \tag{4} 
\]

then there exists a cubic root of \((a, b)\) modulo prime \( p \).

Remark1: Here are examples, where \( p^2 \text{ mod } 9 = 1 \): \( p = 17, 19, 53, 71, 89, 107, 109, 179, 197, 199, 269, 271 \).

The following two algorithms are constructive proofs of these propositions.

2. Algorithm-1

Step 1.1: Compute

\[
W := (a, b)^{(p^2 - 1)/3} \text{ mod } p; \tag{5} 
\]

Step 2.1: if \( W \neq (1, 0) \), then cubic root of \((a, b)\) modulo \( p \) does not exist;

Step 3.1: Compute

\[
s := p \text{ mod } 9; \tag{6} 
\]

there are six possibilities \( s = \pm 1; \pm 2; \pm 4 \).
Step 4.1: if \( s \neq \pm 1 \), then
\[
m := 4/|s|,
\]
otherwise apply Algorithm-2;
Step 5.1: Compute
\[
E_p := \left\lfloor m \left( p^6 - 1 \right) + 3 \right\rfloor / 9,
\]
{where \( m = 1 \) or 2, see Table 1};
Step 6.1: Compute
\[
(pE)^{(x,y)} := (a,b)^{E_p} \mod p.
\]

Example 1: Let \( p = 23 \); \((a,b) = (19,4)\);
\[W := (19,4)^{(23^2-1)/3} = (19,4)^{76} \mod 23 = (1,0); \]
hence \((19,4)\) is a cubic residue. \( E_p = 59 \);
\[
(x,y) := \sqrt[3]{(19,4)} = (19,4)^{59} \mod 23 = (16,16).\] Indeed,
\[(16,16)^3 \mod 23 = (19,4).\] It is easy to verify that \((5,2)\)
and \((2,5)\) are also cubic roots of \((19,4)\). Hence, algorithm
(5)-(9) computes only one of three cubic roots of \((a,b)\). How to compute the two other cubic roots is discussed in sections 5 and A3.

3. Algorithm-2

If \( q \mod 12 = 5 \), then a cubic root exists for every \((a,b)\); and each Gaussian has a unique cubic root. The following algorithm computes such a cubic root (1).

Step 1.2: Compute cubic extractor
\[
E_q := (2q-1)/3;
\]
Step 2.2: Compute \( R := (a,b)^{E_q} \mod q \)
Step 3.2: Output \((x,y) := R.\)
Three examples are provided in Table 2.

4. Multiplicity of Cubic Roots

Proposition 4: Suppose \( C_1, C_2 \) and \( C_3 \) are three cubic roots of \( L := (a,b) \) modulo \( p \), each satisfying the equation
\[
(C^3 - L) \mod p = 0;
\]
then for every \( i = 1, 2, 3 \) the following identities hold:
\[
C_i^3 - L \mod p = 0;
\]
\[
(C_i + C_j + C_k) \mod p = L; \]
\[
(C_i C_j + C_i C_k + C_j C_k) \mod p = 0.
\]
where \( \{i,j,k\} \) is every permutation of \( \{1,2,3\} \).

5. Cubic Roots of \((1,0)\) and Gaussians

In order to find two other roots of \((a,b)\), consider cubic roots of unity:
\[
(u,w) := \sqrt[3]{(1,0)} \mod n.\]
If \((x,y)\) is a cubic root of \((a,b)\), then \((u,w)(x,y)\) and \((u,w)^2 (x,y) \mod p\) are also its cubic roots modulo \(n\).

Proposition 5: If \( p \) is a Blum prime, then either \(\sqrt{3}\) or \(\sqrt{-3}\) modulo \(p\) exists, but not both; if \(\sqrt{3} \mod p\) exists, then
\[
u = \left( (p-1)/2 \right) \mod p;\]
and
\[
w = \left( (p+1)/2 \right) \mod p.\]
If \(\sqrt{-3} \mod p\) exists, then
\[
u = (1 \pm \sqrt{-3}) \mod p.\]
Proof is provided in the Appendix.

6. Existence of \(\sqrt{3} \mod p\) or \(\sqrt{-3} \mod p\)

Jacoby symbols [8,9] analyze whether a specified integer is quadratic residue (QR). If \( p \) is a Blum prime, then
\[
\left( \frac{3}{p} \right) = \left\{ \frac{p \mod 3}{3} = \left\{ \begin{array}{ll} 1, & p \equiv 1 \mod 3 \\ -1, & p \equiv 2 \mod 3 \end{array} \right. \right.\]
or
\[
\left( \frac{2}{3} \right) = \left\{ \frac{2}{3} \right\} = \left\{ \begin{array}{ll} 1, & 2^2 \equiv 1 \mod 3 \\ -1, & 2^2 \equiv 4 \mod 3 \end{array} \right.\]
Therefore, if \( p \mod 12 = 11 \), then 3 is QR. Seven examples are listed in Table 3.

Table 1. Cubic extractors \( E_p \) and \( m.\)

<table>
<thead>
<tr>
<th>( p )</th>
<th>( 7 )</th>
<th>( 11 )</th>
<th>( 13 )</th>
<th>( 23 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( E_p ); ( m )</td>
<td>5; 2</td>
<td>27; 2</td>
<td>19; 1</td>
<td>59; 1</td>
</tr>
<tr>
<td>( p )</td>
<td>29</td>
<td>31</td>
<td>41</td>
<td>43</td>
</tr>
<tr>
<td>( E_p ); ( m )</td>
<td>187; 2</td>
<td>107; 1</td>
<td>187; 1</td>
<td>411; 2</td>
</tr>
</tbody>
</table>

Table 2. Illustrations of cubic root extraction; \( q \mod 12 = 5.\)

<table>
<thead>
<tr>
<th>( q ); ( a,b )</th>
<th>( 53; (19,13) )</th>
<th>( 89; (17,77) )</th>
<th>( 269; (19,73) )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( E_p )</td>
<td>35</td>
<td>59</td>
<td>179</td>
</tr>
<tr>
<td>( (x,y) )</td>
<td>(45,28)</td>
<td>(6,85)</td>
<td>(112,124)</td>
</tr>
</tbody>
</table>

Table 3. \( \sqrt{3} \mod p \) if \( p \mod 12 = 11.\)

<table>
<thead>
<tr>
<th>( p )</th>
<th>11</th>
<th>23</th>
<th>47</th>
<th>59</th>
<th>71</th>
<th>83</th>
<th>107</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \sqrt{3} )</td>
<td>5</td>
<td>16</td>
<td>12</td>
<td>48</td>
<td>43</td>
<td>70</td>
<td>89</td>
</tr>
</tbody>
</table>

Copyright © 2011 SciRes.
7. Properties of Gaussian Cubes

Consider
\[(t,v) := (u,w)^3 = \left(u\left(u^2 - 3w^2\right),w\left(3u^2 - w^2\right)\right) \mod p \tag{21}\]

**Property 1:**
\[(\pm u,\pm w)^3 = \left(\pm u\left(u^2 - 3w^2\right),\pm w\left(3u^2 - w^2\right)\right) = (\pm t,\pm v)\; ; \tag{22}\]

**Property 2:**
\[(w,u)^3 = \left(w\left(w^2 - 3u^2\right),u\left(3w^2 - u^2\right)\right) = -(v,t) \mod p; \tag{23}\]

**Property 3:** If \(u + w = p\), then
\[(u,w)^3 = (u,w)^3 = u^3 (1,1) \mod p. \tag{24}\]

8. Cryptographic Protocol

**System design** (each user’s actions):

- **Step 1.3:** Selects two large distinct primes \(p\) and \(q\), where \(p \mod 12 = 11; \ p^2 \mod 9 \neq 1\); and \(q \mod 12 = 5; \tag{20}\)
- **Step 2.3:** Computes \(n = pq; \ {n \text{ is user’s public key; } p\text{ and } q\text{ are his private keys}}\};
- **Step 3.3:** Finds cubic root \((u,w)\) of \((1,0)\) modulo \(p\):
  \[u = (p - 1)/2 \mod p; \ w = \left(u\sqrt[3]{3}\right) \mod p. \]
- **Step 4.3:** Pre-computes
  \[P = q \left(q^{-1} \mod p\right); \ Q = p \left(p^{-1} \mod q\right) \mod n; \]

**Protocol implementation:** Suppose a sender (Sam) wants to securely transmit a plaintext \(G\) to receiver (Regina);

- Sam divides \(G\) into an array of blocks \(\{(g_1,h_1); (g_2,h_2);\cdots; (g_{33},h_{33});\cdots\}\) in such a way that every \(g_k < n\) and \(h_k < n\);
- **Encryption** {Sam’s actions}:
  - **Step 5.3:** He gets Regina’s public key \(n; \) computes ciphertext
    \[(a,b) := (g,h)^3 \mod n; \]
  and sends \(a, b\) to her;
- **Decryption** {Regina’s actions}:
  - **Step 6.3:** She, using her private keys \(p\) and \(q\), extracts cubic roots
    \[M_1 := \sqrt[3]{(a,b)} \mod p; \text{ and } R := \sqrt[3]{(a,b)} \mod q; \]
  - **Step 7.3:** She computes
    \[M_2 := M_1 \times (u,w) \mod p; \text{ and } M_3 := M_2 \times (u,w) \mod p; \]
  - **Step 8.3:** {Using Chinese Remainder Theorem [10], Regina computes all 3 roots of \((a, b)\)} \(D := \frac{1}{3}\sqrt[3]{(a,b)} \mod n; \)
  for \(k = 1, 2, 3\) \(D_k := \left(M_k P + RQ\right) \mod n; \)

**Step 9.3:** {The original plaintext is recovered via digital isotopes—see sections 10 and 11} \(D = G.\)

9. Efficient Encryption of Gaussians

Squaring of a Gaussian requires two multiplications of real integers (MoRI); and multiplication of two Gaussians requires three MoRI [11]. Therefore, the cubic power of Gaussian requires five MoRI. Yet, encryption
\[(a,b) := (g,h)^3 = \left(g^3 - 3gh^2,3g^2h - h^3\right) \mod p \]
in **Step 5.3** requires only four MoRI:

\[P_1 := g^2 \mod p; \quad P_2 := h^2 \mod p; \]
\[S := P_1 - P_2; \quad A_1 := S - 2P_1; \quad A_2 := S + 2P_1; \]
\[P_3 := gA_2 \mod p; \quad P_4 := hA_4 \mod p; \]
{there are no \(A_1\) and \(A_2\); where the doublings \(2P_1\) and \(2P_2\) are achieved by binary shifting; then \((a,b) := (P_3, P_4)\).}

10. Asymmetric Tagging of Digital Isotopes

In cryptographic algorithms based on extraction of square roots of real integers [12] or Gaussians [6] there are four pairs of solutions, and only one of them is the original plaintext. To distinguish the original solution from the other three, the authors use methods of tails, which is an analogue of using isotopes to tag various chemical components.

If the digital isotopes repeat \(r\) rightmost digits in each component of plaintext \((g, h)\), then the probability of erroneous recovery of the “plaintext” is of order \(O\left(1/10^r\right)\). For instance, if the length of isotope \(r = 3\), then the probability of error is one in one million.

As shown below, a more elaborate strategy must be used to avoid ambiguity in the recovery of the original plaintext.

**Definition 2:** If there exist Gaussians with distinct components \(x\) and \(y\) such that
\[(x,y)^3 = (y,x)^3 \mod p, \tag{25}\]
then such cubic roots are called Gaussian twins (or CT, for short).

**Proposition 6:** If the square root of 3 modulo prime \(p\) exists, then there exists the CT; {see Table 4 for examples}.

| Table 4. Examples of cubic roots twins (CT) for \(p = 83\). |
|-----------------|---------------------|---------------------|---------------------|---------------------|---------------------|---------------------|---------------------|
| \((a,b)\)       | \((27, 56)\)         | \((31, 52)\)         | \((26, 57)\)         | \((2, 81)\)          | \((78, 5)\)          | \((53, 30)\)         | \((76, 7)\)          |
| \(\sqrt[3]{(a,b)}\) | \((22, 76)\)         | \((15, 24)\)         | \((46, 8)\)          | \((77, 7)\)          | \((8, 5)\)           | \((1, 11)\)          | \((8, 5)\)           |

Copyright © 2011 SciRes.
Proof: since \((x - y) \mod p \neq 0\), then (25) implies the following relationships:
\[
(x - y)(1, -1) [(x, x)^2 + (x, y)(y, x) + (y, x)^2] = 0; \\
(x^2 - y^2, 2xy) + (0, x^2 + y^2) + (y^2 - x^2, 2xy) \\
= (0, 2xy + (x + y)^2) \mod p = 0; \\
i.e., [(x + y)^2 + 2xy] \mod p = 0.
\]
Let \(y := T \mod p\), then \(x^2 [(1 + T)^2 + 2T] \mod p = 0\), i.e., \((T^2 + 4T + 1) \mod p = 0\); which implies that \(T = -2 \pm \sqrt{3} \mod p\).

For instance, if \(p = 83\), then \(\sqrt{3} = 70\), i.e., \(T = -2 \pm 70 \mod 83 = \{11 \text{ or } 68\}\).

If \(x = 1\), then \(y^2 = \{11 \text{ or } 68\}; \) hence \(1, (68)^3 = (68.1)^3 \mod 83 = (73,10)\); and \((1, 11)^3 = (11,1)^3 \mod 83 = (53,30)\).

It means that \((53,30) \mod 83\) equals either \((1,1)\) or \((11,1)\). Yet, if in both components the rightmost digit is “1”, it is not clear whether the original plaintext is \((0,1)\) or \((1,0)\). For every \(p \mod 12 = 11\) there exist 4 \((p - 1)\) CTS that satisfy (25) \{examples are provided in Table 4\}.

11. Numeric Illustration

**Algorithm in a nutshell**

| System design | Select \(p, q\); Compute \(n, P, Q, u, w, s, m, E_p, E_q\); |
| Encryption | Create plaintext \(Z\) with isotopes; compute ciphertext \((a, b)\); |
| Decryption | \(R; RQ; M_i; Z\); |

**System design:** Let Regina’s \(p = 227\) and \(q = 1109\), where \(p^2 \mod 9 = 4 \neq 1\), \(p \mod 12 = 11\); and \(q \mod 12 = 5\); she computes \(n = pq = 251,743\); \(P\) and \(Q\):

\[
P := q \left( q^{-1} \mod p \right) = 1109 \times \left( 1109^{-1} \mod 227 \right) = 1109 \times 96 = 106464;
\]

\[
Q := p \left( p^{-1} \mod q \right) = 227 \times \left( 227^{-1} \mod 1109 \right) = 227 \times 640 = 145280;
\]

and a cubic root \((u, w)\) of \((1,0)\) modulo \(p\):

\[
u = (227 - 1)/2 \mod p = 113;
\]

\[
w = u \sqrt{3} = 113 \times 3^{37} \mod 227 = 25;
\]

**Remark 2:** \((u, w)^3 = (u, p - w) \mod p\); \{indeed, \((113, 25)^3 \mod 227 = (1,0)\); and \((113, 25)^2 \mod 227 = (113, 202)\); \(s := 227 \mod 9 = 2\); \(m = 4[2] = 2\); and Gaussian cubic extractor \{Step 5.1\}.

\[
E_p := \left[ 2 \times (227^2 - 1) + 3 \right]/9 = 11451;
\]

finally, Regina pre-computes another cubic extractor \(E_q := (2q - 1)/3 = 739\).

**Encryption:** Suppose the sender (Sam) wants to securely transmit message \(G = (1941, 2487)\) to Regina with two-digit isotopes:

\[
Z := 100G + G \mod 100 = (1941, 2487);
\]

In this case the probability of erroneous recovery of the original message will not exceed 1/10,000, i.e., it equals 0.01%.

Sam computes ciphertext \(a, b) := Z^3 \mod 251743 = (227258, 195067)\)

**Decryption:** Regina computes

\[
M_i := (227258, 195067) \mod 227 = (31, 74) \mod 227 = (74, 78)
\]

and two other cubic roots:

\[
M_2 := M_i \times (u, w) = (74, 78) \times (113, 25) = (56, 222);
\]

\[
M_3 := M_i \times (u, w) = (56, 222) \times (113, 25) = (97, 154);
\]

and then unique cubic root \(R\) modulo \(q\):

\[
R := (a, b)^k \mod q = (227258, 195067)^{739} = (66, 371) \mod 1109\)

Using the Chinese Remainder Theorem [10], Regina computes (28):

\[
(x, y)_k = \left[ M_k \times 1109 \times (1109^{-1} \mod 227) \right] + R \times 227 \times (227^{-1} \mod 1109) \mod 251743 = (106464M_k + 145280R) \mod 251743;
\]

until she detects isotopes

\[
(x, y)_k = 106464M_1 + 145280 \times (66, 371) = (106464 \times (74, 78) + (22246, 25878) \mod n = (96549, 274294); \text{ where } n = 251743.
\]

\[
(x, y)_k = 106464M_2 + (22246, 25878) \mod n = (106464 \times (56, 222) + (22246, 25878) \mod n = (1941, 2487).
\]

Therefore, Regina recovers the original Gaussian block of information; and it is not necessary to compute \((x, y)_j\).

12. Optimized Recovery of Information

Let \(M_k = (M_{k1}, M_{k2})\); \(R = (R_1, R_2)\);
then \( x_k = M_{11}P + R_1Q; y_k = M_{12}P + R_2Q \).

\( M_f = (M_{11}, M_{12}) \) is computed by cubic root extraction; if the isotopes in \( Z \) are detected, then the original information is recovered; otherwise Regina needs to compute four components of two other cubic roots of \((a, b)\):

\[
M_2 = (M_{11}, M_{12})(u, w) = (M_{11}u - M_{12}w, M_{11}w + M_{12}u); \tag{27}
\]

\[
M_3 = (M_{11}, M_{12})(u, -w) = (M_{11}u + M_{12}w, -M_{11}w + M_{12}u). \tag{28}
\]

Yet, to minimize computational burden, instead of computing \( M_2 \) and \( M_3 \), she finds

\[
N_i := M_{12}wP; \tag{29}
\]

and then computes

\[
x_2 = (M_{11}uP + R_1Q) - N_i. \tag{30}
\]

If the isotopes are detected, then she computes \( y_2 \), otherwise Regina computes

\[
x_3 = x_2 + 2N_i
\]

\[
= (M_{11}uP + R_1Q) + M_{12}wP \text{ and } y_3. \tag{31}
\]

13. Elimination of Ambiguity in Recovery of Original Information

The probability of erroneous recovery can be decreased if, instead of repeating \( r \) rightmost digits of \( g \) and \( h \), the following procedure is applied:

1) Consider \( r \) leftmost digits (prefix \( P_r \)) of the first component \( g \) in plaintext \((g, h)\) and repeat it as its digital isotope;

2) Consider \( r \) rightmost digits (suffix \( S_r \)) of the second component \( h \) of plaintext \((g, h)\) and repeat it as its digital isotope.

**Example 2:** if \((g, h) = (31415926, 27182845)\) and \( r = 2 \), then \((3131415926, 27182845455)\).

**NB:** if \( n \) is \( r \)-digits long and the number of digits in \( g \) is smaller than \( t \), then the prefix \( P_r = \ldots \cdot 0 \). To avoid ambiguity, the sender must attach both digital isotopes \( P_r \) and \( S_r \) as suffixes. Below is a simple mnemonic/schematic rule for constructing the digital isotopes:

\[
\text{priority, cocktail} \Rightarrow \text{ priorityprio, cocktailTAIL).}
\]

**Remark 3:** Therefore, there can be two types of cubic roots with isotopes:

\((PUP, VSS)\) and \((USS, PVP)\). Only the former one is authentic. Hence, a receiver (Regina) searches for the cubic root with isotopes in format \((PUP, VSS)\), where \( P \) and \( S \) are prefix and suffix respectively. In this case \((PU, VSS)\) is acceptable as the genuine plaintext.

**Example 3:** if \( t = 8; r = 2 \); and \((g, h) = (00415926, 07182845)\), then

\[
Z = (g - g \mod 10^6 \times 10^8 + g,
\]

\[
(h - h \mod 10^2 \times 10^2 + h \mod 10^2) \quad (0041592600, 07182845454).
\]

14. Second Numeric Illustration

**System design:** Let \( p = 227; q = 1109, n = pq = 251,743;\)

\( P = 106464; Q = 145280; (u, w) = (113,25); s = 2; \)

\( m = 2; E_x = 11451; E_y = 739.\)

**Encryption:** Plaintext \( G = (1756, 2011)\); plaintext with isotopes

\[
Z := (175617, 201111);
\]

and ciphertext \((a, b) = (57971, 209989)\);

**Decryption:** \( R := (395, 382); M := (202, 137); \)

\( QR \mod n = 239939; \quad N_i := M_{12}wP = 115336; \)

\[
x_2 = M_{11}wP + QR_i - N_i = 196688; \]

since there is no isotope in \( x_2 \), then \((x_3, y_3)\) is the original Gaussian.

Indeed, \( x_3 = x_2 + 2N_i = 175617.\)

15. Algorithm Analysis

The cryptographic algorithm described above is neither a generalization nor a special case of the RSA protocol [13].

First of all, the following identity holds:

\[
(a, b)^{(p-1)(q-1)} \mod n = (1, 0) = 1. \tag{32}
\]

In the RSA algorithm, if \( z \) is the length of group cycle [13], then each user selects a public key \( e \) that is co-prime with \( z \). In the proposed algorithm the length of cycle \( c \) is equal

\[
c := (p^2 - 1)(q - 1). \tag{33}
\]

Therefore in the RSA extension it would have been necessary to compute a multiplicative inverse \( d \) of \( e \) modulo \( c \). Yet, in the algorithm described above the encryption key \( e = 3 \). Hence, the decryption key \( d \) cannot be computed as a modular multiplicative inverse, since \( \gcd(3, z) = 3 \), which implies that such an inverse does not exist [14].

16. Communication Speed-Up

Suppose it is necessary to transmit an \( H \) digit-long plain-
text, where the size of each block must not exceed sixteen digits; in addition, suppose that we want to ensure that the probability of erroneous recovery does not exceed one in one million. There are two options:

Option 1 is to select the size of each block equal to ten digits and the size of each tail equal to six digits; Option 2 is to select the size of each block equal to thirteen digits and the size of each tail equal to three digits.

In the 1st option we will treat each block individually as a real integer; which implies that we need to transmit $H/10$ real integers. In the 2nd option we will treat a pair of blocks as a Gaussian; which implies that we need to transmit $H/26$ Gaussians, i.e., $H/13$ real integers. Therefore, the first option requires $13/10 = 1.3$ times more bandwidth, than the second option. In other words, the bandwidth can be reduced by 30% if Gaussian integers are considered.

17. Possible Applications and Conclusions

The proposed cryptosystem has significant specifics: the encryption is substantially faster than the decryption. There are certain settings where the sender has limited time to transmit the message: visual images or video, and receiver does not have such restriction. For instance, the sender is a system that urgently needs to transmit information prior to either collision with a target or before it is destroyed by a hostile action [15]. Another example is if the sender (say, an interplanetary or interstellar space station) detects an impending collision with an asteroid and is programmed to report about such collision and transmit visual and other details about the asteroid.

In this case it is paramount to ensure the reliability of message delivery [15,16]. Yet another example is of a security camera that has detected an imminent explosion (audio, pictures and/or video) [17] prior to its own destruction from the explosion.

18. Acknowledgements

I express my appreciation to J. Jones, and R. Rubino for corrections, and to E. A. Verkhovsky as well as anonymous reviewers for several suggestions that improved this paper.

19. References


Appendix

A1. Validation of Algorithm-1

If condition (5) holds; then

\[(x, y)^3 = (a, b)^3 \equiv (a, b)^{m(p^2-1)/9}\]

\[(a, b) \mod p\] \hspace{1cm} (A1)

If \((a, b)^{m(p^2-1)/9} \mod p = (1, 0)\), then by Definition (2) \((x, y)\) is a cubic root of \((a, b)\) modulo prime \(p\). Hence, if \((p^2-1)/9\) is not an integer, then there exists an integer \(m\) such that \(E_p\) is an integer, i.e., there exists an integer solution of equation

\[m(p^2-1)/9 \mod 9 = 0.\] \hspace{1cm} (A2)

Indeed, observe that

1) Every integer greater than 3 can be expressed either as \(p = 6k + 1\) or as \(p = 6k - 1\);
2) \((p^2-1)/3\) is an integer for every prime greater than 3;
3) If \((p^2-1)/9\) is not an integer, then \(k \mod 3 = 0\); and \((p^2-1)/3\) is not co-prime with 3.

Therefore, (A2) can be rewritten as

\[m(p^2-1)/3 \mod 3 = 2.\] \hspace{1cm} (A3)

If there is no integer solution of Equation (A3); then Algorithm-1 is not applicable for these cases. In other terms, if either \((p-1)/9\) or \((p+1)/9\) is an integer, then \(E_p\) is not an integer.

A2. Validation of Algorithm-2

Let

\[R^3 = \left[(a, b)^{(p-1)/2}\right]^2 \times (a, b) \mod q.\] \hspace{1cm} (A4)

Since \(q \mod 12 = 5\), then \((q-1)/2\) is even, hence by Euler criterion of quadratic residuosity \((-1)^{(q-1)/2} \mod q = 1\), i.e.,

\[i = \sqrt{p-1} \mod q\] \hspace{1cm} (A5)

is a real integer; and hence \((a, b) \mod q\) is also a real integer. Therefore, by the Fermat identity

\[R^3 \mod q = (a, b).\] Q. E. D.

A3. More on Identities for Cubic Roots

By the Vieta theorem [7], equation

\[(C^3 - L) \mod p = 0;\] \hspace{1cm} (A6)

implies that

\[(C - C_1)(C - C_2)(C - C_3) \mod p = 0.\] \hspace{1cm} (A7)

Hence, (A6) and (A7) imply

\[(C_1 + C_2 + C_3) \mod p = 0;\] \hspace{1cm} (A8)

and for every permutation \(\{i, j, k\}\) of

\[\{1, 2, 3\}; \quad \{C_iC_j-C_i^2\} \mod p = 0.\] \hspace{1cm} (A9)

On the other hand, (A9) implies

\[\left[(C_1 + C_j)^2 - C_iC_j\right] \mod p = 0.\] \hspace{1cm} (A10)

Yet, neither (A9) nor (A10) are instrumental in recovery of all cubic roots.

A4. Proof of Proposition 5

Algebraic approach:

\[(u, w)^3 = (u^3 - 3uw^2, 3u^2w - w^3) \mod p = (1, 0)\] \hspace{1cm} (A11)

Therefore from (A6) we deduce two equations with unknown \(u\) and \(w\):

\[u(u^2 - 3w^2) \mod p = 1;\] \hspace{1cm} (A12)

and

\[w(3u^2 - w^2) \mod p = 0.\] \hspace{1cm} (A13)

Since in (A13) \(w \mod p \neq 0\), then

\[3u^2 = w^2 (\mod p).\] \hspace{1cm} (A14)

Hence, (A12) and (A14) imply that

\[8u^3 + 1 = 0 (\mod p);\] \hspace{1cm} or

\[(2u+1)(4u^2 - 2u + 1) \mod p = 0.\] \hspace{1cm} (A15)

Equation (A15) holds if either

\[u = (p-1)/2 (\mod p);\] \hspace{1cm} (A16)

or

\[4u^2 - 2u + 1 \mod p = 0.\] \hspace{1cm} (A17)

Thus in (A16) case, if there exists square root of 3 modulo \(p\), then from (A14)

\[w = \pm u\sqrt{3} = \pm(p-1)\sqrt{3}/2.\] \hspace{1cm} (A18)

Otherwise, Equation (A17) implies that

\[u = 1 \pm \sqrt{-3} \mod p;\] \hspace{1cm} (A19)

and finally from (A12) we deduce \(w\).

Trigonometric approach: Consider

\[(u, w) = \sqrt{(1, 0)} = \sqrt{(\cos \pi + i\sin \pi)}\]

\[= (\cos \pi/3 + i\sin \pi/3) = (1, \sqrt{3})/2 (\mod p)\] \hspace{1cm} (A20)
Proposition 7: If \((x, y)\) is a cubic root of \((a, b)\) modulo \(p\), then \((u, w)(x, y) \mod p\) and \((u, w)^{(k)}(x, y) \mod p\) are also cubic roots of \((a, b)\).

Proof: From Definition 1 (2) for \(k = 1, 2\)
\[
\left[(u, w)^{(k)}(x, y)\right]^3 = (a, b) \mod p, \text{Q.E.D.}
\]