Public Key Cryptosystems and Line Pictures

P.J. ABISHA, D.G. THOMAS AND D. JAYASEELAN SAMUEL
Department of Mathematics
Madras Christian College
Tambaram, Chennai- 600 059, India
e-mail: *jayaseelansamuel@gmail.com*

Abstract

Public Key Cryptosystems (PKC) based on formal language theory and semi groups have been of interest and study. A PKC based on free group has been presented in [7]. Subsequently another PKC using free partially commutative monoids and groups is studied in [1]. In this paper, we propose a PKC for chain code pictures that uses a finitely presented group for encryption and free group for decryption. Also, we present another PKC for line pictures in the hexagonal grid, which uses a finitely presented group for encryption and finitely presented free partially commutative group for decryption.

Keywords. Public key cryptosystems, finitely presented groups, word problem, line pictures, chain code pictures

1 Introduction

Cryptography is the science of keeping secrets secret. It is the study of sending messages in disguised form so that the intended recipient can remove the disguise and read the message. Diffie and Hellman introduced the concept of Public Key Cryptosystem (PKC) [3]. In public key cryptography the encryption key is available to everyone and the decryption key is kept secret by the owner. In public key cryptosystems we are looking for family of functions such that each function f is computable by an efficient algorithm but it is infeasible to compute the pre-images. Such functions are called one-way functions. For each function in that family there is some secret information which enables an efficient computation of the inverse of f. This secret information is called trapdoor information. One-way functions with this property are called trapdoor functions.

In [6], Salomaa has very elegantly formulated the general technique to construct public key cryptosystems based on formal language theory.

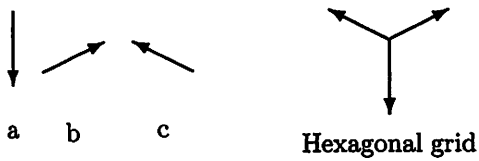


Figure 1:

Choose a difficult (undecidable or intractable) problem Q and a sub problem P, which is solvable in linear time. Shuffle P to obtain $P1$, which looks like Q. The manner of shuffling is the trapdoor, which is secret. Use $P1$ to encrypt and P to decrypt. In [1], a PKC based on the finitely presented partially commutative group is presented. In this paper, we present PKCs for line pictures based on finitely presented groups and finitely presented partially commutative groups.

A picture whose structure is described by unit lines is called the line picture. A line picture in the cartesian plane considered as a square grid described by a word over the alphabet $\{l, r, u, d\}$ where l means go and draw one unit line left from the current position and r, u, d are interpreted analogously with right, up, down is called a Chain Code Picture (CCP) [5].

Such line pictures can also be developed on a hexagonal grid by fixing three basic directions and three reverse directions in the following manner. Let a, b, c stand for the lines in the directions $\downarrow, \nearrow, \nwarrow$ of a hexagonal grid in Figure 1 and a^{-1}, b^{-1}, c^{-1} denote the reverse directions. We propose a PKC for certain type of chain code pictures where the decryption is done using the free group. The encryption is done using a finitely presented group. We present another PKC for certain type of line pictures over the hexagonal grid, which uses a free partially commutative group for decryption and finitely presented group for encryption.

2 Preliminaries

In this section, we give some preliminary definitions ([1], [7]). For basic definitions in formal languages theory, we refer to [6].

Definition 2.1. *Given an alphabet Σ we consider $\Sigma^{-1} = \{a^{-1}/a \in \Sigma\}$ and $\Sigma^{\pm 1} = \Sigma \cup \Sigma^{-1}$. A word x in $\Sigma^{\pm 1}$ is called reduced if it does not contain any subword of the form $a^{\sigma}a^{-\sigma}, a \in \Sigma, \sigma = \pm 1$.*

Definition 2.2. *Let Σ be an alphabet and $\theta = \{(a, b)/$ for some $a, b \in \Sigma\}$*

be a concurrency relation on Σ. It means that a and b can be commuted. In other words, an occurrence of ab can be replaced by ba and ba by ab. If a word $u \in \Sigma^*$ is obtained from a word $v \in \Sigma^*$ by such a sequence of replacements then we say that u and v are equivalent with respect to the relation θ and it is denoted by $u \equiv_\theta v$ or $u \equiv v \bmod \theta$.

Definition 2.3. *A group G, or more precisely, a presentation of the group G, denoted by $G =< \Sigma, R >$, is given by an alphabet Σ and a set R of pairs in $(\Sigma^{\pm 1})^* \times \{\lambda\}$ called defining relators of G. If Σ and R are finite we say that G is a finitely presented group. If $R = \phi$, G is called a free group generated by Σ and is denoted by $F(\Sigma)$. Given a group $G =< \Sigma, R >$, we consider the binary relation on $(\Sigma^{\pm 1})^*$ denoted by $\underset{G}{\Longleftrightarrow}$ or simply $\Leftrightarrow$ and defined as follows: For any $x, y \in (\Sigma^{\pm 1})^*$, $x \underset{G}{\Longleftrightarrow} y$ if and only if one of the following cases hold.*

1. $x = urv, y = uv$ *with* $(r, \lambda) \in R$ *and* $u, v \in (\Sigma^{\pm 1})^*$
2. $x = uv, y = urv$ *with* $(r, \lambda) \in R$ *and* $u, v \in (\Sigma^{\pm 1})^*$
3. $x = ua^\sigma a^{-\sigma} v, y = uv$ *with* $a \in \Sigma, \sigma = \pm 1, u, v \in (\Sigma^{\pm 1})^*$
4. $x = uv, y = ua^\sigma a^{-\sigma} v$ *with* $a \in \Sigma, \sigma = \pm 1, u, v \in (\Sigma^{\pm 1})^*$

Then we define $\overset{*}{\underset{G}{\Longleftrightarrow}}$ to be the reflexive, transitive closure of $\underset{G}{\Longleftrightarrow}$. It is easy to see that $\overset{*}{\underset{G}{\Longleftrightarrow}}$ is a congruence relation and the quotient $(\Sigma^{\pm 1})^* / \overset{*}{\underset{G}{\Longleftrightarrow}}$ is a group also denoted by G. The congruence class of a word x is denoted by $[x]_G$ or simply $[x]$. Evidently $x \overset{*}{\underset{G}{\Longleftrightarrow}} y$ if and only if $[x]_G \overset{*}{\underset{G}{\Longleftrightarrow}} [y]_G$. We usually write $x =_G y$ instead of $[x]_G = [y]_G$ and say that the words x and y are equal in G.

The word problem for a group G consists in deciding for any two words x, y in $(\Sigma^{\pm 1})^*$, whether $x =_G y$ or equivalently in deciding, for any given word x whether $x =_G \lambda$ where λ is the empty word.

Definition 2.4. *A Thue system T on Σ is a finite subset of $\Sigma^* \times \Sigma^*$. Each member of T is called a rule. The Thue congruence $\overset{*}{\underset{T}{\longleftrightarrow}}$ generated by T is the reflexive, transitive closure of the symmetric relation $\underset{T}{\leftrightarrow}$, defined as follows: For any u, v such that $(u, v) \in T$ or $(v, u) \in T$ and any $x, y \in \Sigma^*, xuy \leftrightarrow xvy$. Two strings $w, z \in \Sigma^*$ are congruent with respect to T if and only if $w \overset{*}{\underset{T}{\leftrightarrow}} z$ where $\overset{*}{\underset{T}{\leftrightarrow}}$ is the reflexive, transitive closure of $\underset{T}{\leftrightarrow}$.*

The word problem for the Thue system on Σ is as follows: given any two words x and y in Σ, is $x \overset{*}{\underset{T}{\leftrightarrow}} y$?. The word problem is in general undecidable for Thue systems.

Definition 2.5. *Let Σ be an alphabet and θ_0 be a partially commutative (concurrency) relation on Σ. Let $\theta \subseteq \Sigma^{\pm 1} \times \Sigma^{\pm 1}$ be the extension of θ_0 to $\Sigma^{\pm 1}$:*

$$\theta = \left\{(a,b), \left(a^{-1},b\right), \left(a,b^{-1}\right), \left(a^{-1},b^{-1}\right) : (a,b) \in \theta_0\right\}.$$

This develops the following Thue system T on $\Sigma^{\pm 1}$, where $T = \left\{(aa^{-1},\lambda), (a^{-1}a,\lambda) : a \in \Sigma\right\} \cup \{(cd,dc) : c,d \in \theta\}$. This Thue system T presents the free partially commutative group $G(\theta_0)$. This $G(\theta_0)$ is the finitely generated group $< \Sigma, R >$ where the set of defining relators $R = \{(cdc^{-1}d^{-1},\lambda) : (c,d) \in \theta\}$. If θ_0 is empty then $G(\theta_0)$ is just the free group on Σ and if θ_0 contains every pair of distinct letters, then $G(\theta_0)$ is the free abelian group on Σ.

A very well known result, due to Novikov [5], says that the word problem for finitely presented group is undecidable. But the algorithm solving word problem for free group is quite simple because for any free group $F(\Sigma)$ and any $x, y \in \left(\Sigma^{\pm 1}\right)^*$, $x =_F y$ if and only if x and y can be reduced to the same word. Wrathall [8] proved that the word problem for finitely presented free partially commutative group is decidable in linear time.

Definition 2.6. *Let Σ be an alphabet and $X \subseteq \Sigma^*$. Then X is said to be a code if whenever $x_1x_2\ldots x_n = y_1y_2\ldots y_m$ where $x_i, y_i \in X$, $i = 1,2,3,\ldots,n$, $j = 1,2,\ldots,m$ then $m = n$ and $x_i = y_i$ for all i.*

3 PKC for Chain Code Pictures

In this section, we propose a PKC for Chain Code Pictures without over writing (retracing) that uses a free group for decryption and a finitely presented group for encryption.

3.1 Construction of PKC

Consider a free group $F(\Sigma)$, which is non-empty and finite. Fix four different reduced words x_l, x_r, x_u, x_d in $\left(\Sigma^{\pm 1}\right)^*$ such that

1. $\{x_l, x_r, x_u, x_d\}$ is a code
2. The word x_ix_j is reduced without cancellation where $i,j \in \{l,r,u,d\}$ Let Δ be an alphabet of cardinality much greater than that of Σ. Let $g : \left(\Delta^{\pm 1}\right)^* \to \left(\Sigma^{\pm 1}\right)^*$ be a morphism, mapping every letter to a letter or to the non-empty word such that
3. $g\left(c^{\sigma_1}\right) = a^{\sigma_2}$ implies $g\left(c^{-\sigma_1}\right) = a^{-\sigma_2}$, $c \in \Delta$, $a \in \Sigma$, $\sigma_1, \sigma_2 = \pm 1$
4. $g\left(c^{\sigma}\right) = \lambda$ implies $g\left(c^{-\sigma}\right) = \lambda$, $c \in \Delta$, $a \in \Sigma$, $\sigma = \pm 1$

g is called the trapdoor morphism.

Fix a finite subset R of $g^{-1}([\lambda])$ and define $G = \langle \Delta, R \rangle$. Select four different words $\{w_l, w_r, w_u, w_d\}$ from $\left(\Delta^{\pm 1}\right)^*$ such that $g(w_i) \in [x_i]_F$ where $i \in \{l, r, u, d\}$.

3.2 Encryption

The public encryption key is (G, w_l, w_r, w_u, w_d).

1. Replace each occurrence of l, r, u, d by w_l, w_r, w_u, w_d respectively.
2. Insert relators from R and rewrite to obtain arbitrary cryptotext c.

3.3 Decryption

The secret decryption key is $(F, x_l, x_r, x_u, x_d, g)$

1. Calculate $g(c)$
2. Reduce $g(c)$ to obtain a reduced word z.
3. Factorize z over elements $\{x_l, x_r, x_u, x_d\}$ say $z = x_{i_1} x_{i_2} \dots x_{i_m}$ (This factorization is unique as $\{x_l, x_r, x_u, x_d\}$ is a code)
4. $i_1, i_2 \dots, i_m$ is the required plaintext.

Clearly each of these steps can be done in linear time.

Example

Let us consider the chain code alphabet $\{l, r, u, d\}$ as $\Sigma^{\pm 1} = \Sigma \cup \Sigma^{-1}$ where $\Sigma = \{l, d\}$ and $\Sigma^{-1} = \{r, u\}$.

Let $\Delta = \{c_1, c_2, c_3, c_4, c_5, c_6\}$ and we define

$$g(c_1) = g\left(c_2^{-1}\right) = l, \quad g\left(c_1^{-1}\right) = g(c_2) = r$$

$$g\left(c_4^{-1}\right) = g(c_6) = d, \quad g(c_4) = g\left(c_6^{-1}\right) = u$$

$$\text{and} \quad g(c_3) = g\left(c_3^{-1}\right) = g(c_5) = g\left(c_5^{-1}\right) = \lambda$$

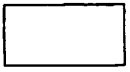
Let $R = \left\{c_6 c_3^{-1} c_4, c_1^{-1} c_5 c_2^{-1}, c_6^{-1} c_5 c_4^{-1}, c_2 c_3 c_2^{-1} c_5\right\}$ and define $G = \langle \Delta, R \rangle$.

Let us choose $x_l = ld, x_d = lul, x_r = ll, x_u = dld$ in $F(\Sigma)$.

Choose $w_l = c_1 c_4^{-1}, w_d = c_2^{-1} c_4 c_1, w_r = c_1 c_5 c_2^{-1}$ and $w_u = c_4^{-1} c_2^{-1} c_6$.

To encrypt the rectangle in Figure 2, if we consider the picture from the bottom most vertex on the right, in the clockwise direction we obtain the word $p = llurrd$ which is taken to be the corresponding plaintext. Replacing l, r, u, d by w_l, w_r, w_u, w_d respectively, we get

$$w_l w_l w_u w_r w_r w_d = c_1 c_4^{-1} c_1 c_4^{-1} c_4^{-1} c_2^{-1} c_6 c_1 c_5 c_2^{-1} c_1 c_5 c_2^{-1} c_2^{-1} c_4 c_1$$



Rectangle

Triangle

Figure 2: Line Pictures

Inserting relators from R, one possible cryptotext is

$$c = c_1c_6c_3^{-1}c_4c_4^{-1}c_1c_4^{-1}c_6c_3^{-1}c_4c_4^{-1}c_2^{-1}c_1^{-1}c_5c_2^{-1}c_6c_1c_5c_2^{-1}c_1c_5c_2^{-1}c_2^{-1}c_4c_1$$

$$= c_1c_6c_3^{-1}c_1c_4^{-1}c_6c_3^{-1}c_2^{-1}c_1^{-1}c_5c_2^{-1}c_6c_1c_5c_2^{-1}c_1c_5c_2^{-1}c_2^{-1}c_4c_1$$

To decrypt, apply g on c.

$$\begin{aligned} g(c) &= ld\lambda lddd\lambda lr\lambda ldl\lambda ll\lambda llul \\ &=_F ldlddlrldllllul \\ &=_F ldlddldllllul \\ &=_F x_l x_l x_u x_r x_r x_d \end{aligned}$$

Thus the message is *llurrd*.

4 PKC for Line Pictures using Finitely Presented Free Partially Commutative Group

In this section we present a PKC for line pictures over the hexagonal grid, which uses finitely presented free partially commutative group for decryption and finitely presented group for encryption.

4.1 Construction of PKC

Let Σ be an alphabet and θ_0 be a partially commutative relation on Σ such that $G(\theta_0)$ is the finitely presented free partially commutative group.

Fix six different reduced words $x_a, x_b, x_c, x_{a^{-1}}, x_{b^{-1}}, x_{c^{-1}}$ in $\left(\Sigma^{\pm 1}\right)^*$ such that

1. $\{x_a, x_b, x_c, x_{a^{-1}}, x_{b^{-1}}, x_{c^{-1}}\}$ is a code.

2. The word $x_i x_j$ is reduced without cancellation where $i, j \in \{a, b, c, a^{-1}, b^{-1}, c^{-1}\}$.

   Let Δ be an alphabet of cardinality much greater than that of Σ. Let g be a morphism from $\Delta^{\pm 1}$ to $\Sigma^{\pm 1} \cup \{\lambda\}$ such that

3. $g(c^{\sigma_1}) = a^{\sigma_2}$ implies $g(c^{-\sigma_1}) = a^{-\sigma_2}$, $c \in \Delta$, $\sigma_1, \sigma_2 = \pm 1$, $a \in \Sigma$

4. If $g(c^{\sigma}) = \lambda$ implies $g(c^{-\sigma}) = \lambda$, $c \in \Delta$, $\sigma = \pm 1$

5. If $g(c) = a, g(d) = b$ and $(a, b) \in \theta$ then c and d commute where $c, d \in \Delta$

Fix a finite subset R of $g^{-1}([\lambda])$ and define $G = \langle \Delta, R \rangle$. Six words $w_a, w_b, w_c, w_{a^{-1}}, w_{b^{-1}}, w_{c^{-1}}$ from $(\Delta^{\pm 1})^*$ is selected such that $g(w_i) \in [x_i]_G$ where $i \in \{a, b, c, a^{-1}, b^{-1}, c^{-1}\}$. Fix a finite subset $\bar{R}$ of $(\Delta^{\pm 1})^* \times \{\lambda\}$ such that if $(uv^{-1}, \lambda) \in \bar{R}$ then one of the following is true

1. $\left(g(u)\left(g(v)^{-1}\right), \lambda\right) \in R$

2. $g(u) =_{G(\theta_0)} \lambda$ and $g(v) =_{G(\theta_0)} \lambda$

Then $\bar{G} = \langle \Delta, \bar{R} \rangle$ is a finitely presented group.

4.2 Encryption

The public encryption key is $(\bar{G}, w_a, w_b, w_c, w_{a^{-1}}, w_{b^{-1}}, w_{c^{-1}})$

1. Replace each occurrence of $a, b, c, a^{-1}, b^{-1}, c^{-1}$ by $w_a, w_b, w_c, w_{a^{-1}}, w_{b^{-1}}, w_{c^{-1}}$ respectively.

2. Insert relators from $\bar{R}$ and reduce to obtain arbitrary cryptotext c.

4.3 Decryption

The secret decryption key is $(G, x_a, x_b, x_c, x_{a^{-1}}, x_{b^{-1}}, x_{c^{-1}}, g)$

1. Calculate $g(c)$

2. Reduce $g(c)$ to obtain a reduced word z.

3. Factorize z over elements $x_a, x_b, x_c, x_{a^{-1}}, x_{b^{-1}}, x_{c^{-1}}$ say $z = x_{i_1} x_{i_2} \ldots x_{i_m}$ (This factorization is unique as $\{x_a, x_b, x_c, x_{a^{-1}}, x_{b^{-1}}, x_{c^{-1}}\}$ is a code)

4. $i_1 i_2 \ldots i_m$ is the required plaintext.

Clearly each of these steps can be done in linear time.

Example Let us consider the alphabet $\{a, b, c, a^{-1}, b^{-1}, c^{-1}\}$ as $\Sigma^{\pm 1}$ where $\Sigma = \{a, b, c\}$ and $\Sigma^{-1} = \{a^{-1}, b^{-1}, c^{-1}\}$

Let $\theta_0 = \{(a, c)\}$ and $\theta = \{(a, c), (a^{-1}, c), (a, c^{-1}), (a^{-1}, c^{-1})\}$

$$R = \{(aca^{-1}c^{-1}, \lambda), (a^{-1}cac^{-1}, \lambda), (ac^{-1}a^{-1}c, \lambda), (a^{-1}c^{-1}ac, \lambda)\}$$

Choose $x_a = bcc$, $x_b = abc$, $x_c = bbc$ and $x_{a^{-1}} = b^{-1}ac$, $x_{b^{-1}} = aab$, $x_{c^{-1}} = bc^{-1}a$

Let $\Delta = \{c_1, c_2, c_3, c_4, c_5, c_6\}$

Define $g : (\Delta^{\pm 1})^* \to (\Sigma^{\pm 1})^*$ by

$$\begin{aligned} g(c_1) &= g(c_2^{-1}) = a, & g(c_1^{-1}) &= g(c_2) = a^{-1}, \\ g(c_4^{-1}) &= b, & g(c_4) &= b^{-1}, \\ g(c_6^{-1}) &= c^{-1}, & g(c_6) &= c, \\ g(c_3) &= g(c_3^{-1}) = g(c_5) = g(c_5^{-1}) = \lambda \end{aligned}$$

and choose $w_a = c_3^{-1}c_4^{-1}c_6c_6$, $w_b = c_2^{-1}c_4^{-1}c_6$, $w_c = c_4^{-1}c_5c_4^{-1}c_6$ and

$$w_{a^{-1}} = c_4c_3c_1c_6, \quad w_{b^{-1}} = c_1c_2^{-1}c_4^{-1}c_5^{-1}, \quad w_{c^{-1}} = c_4^{-1}c_6^{-1}c_1$$

$$\bar{R} = \{(c_2^{-1}c_6c_3c_1^{-1}c_6^{-1}, \lambda), (c_1^{-1}c_5c_2^{-1}c_3^{-1}, \lambda), (c_3c_5, \lambda), (c_4c_5c_4^{-1}c_5c_3^{-1}, \lambda), (c_2c_3c_2^{-1}c_5, \lambda)\}$$

$$\bar{G} = \langle \Delta, \bar{R} \rangle.$$

Consider the triangle in Figure 2, in the hexagonal grid. If we consider the picture from the bottom most vertex in the anticlockwise direction we obtain the word $p = bca$ which is taken to be the corresponding plaintext.

To encrypt, replace each occurrence of a, b, c by w_a, w_b, w_c respectively, we get

$$\begin{aligned} w_bw_cw_a &= c_2^{-1}c_4^{-1}c_6c_4^{-1}c_5c_4^{-1}c_6c_3^{-1}c_4^{-1}c_6c_6 \\ c &= c_2^{-1}c_4^{-1}c_2^{-1}c_6c_3c_1^{-1}c_6^{-1}c_6c_4^{-1}c_3c_5c_5c_4^{-1}c_6c_3^{-1}c_3c_5c_4^{-1}c_6c_6 \\ &= c_2^{-1}c_4^{-1}c_2^{-1}c_6c_3c_1^{-1}c_4^{-1}c_3c_5c_5c_4^{-1}c_6c_5c_4^{-1}c_6c_6 \end{aligned}$$

To decrypt, apply g on c.

$$\begin{aligned} g(c) &= abac\lambda a^{-1}b\lambda\lambda\lambda bc\lambda bcc \\ &=_G abaca^{-1}bbcbcc \\ &=_G abcaa^{-1}bbcbcc \\ &=_G abcbbcbcc \\ &= x_bx_cx_a \end{aligned}$$

Thus the message is bca.

Remark In Figure 3, we give some line pictures which can be communicated by the above PKC.

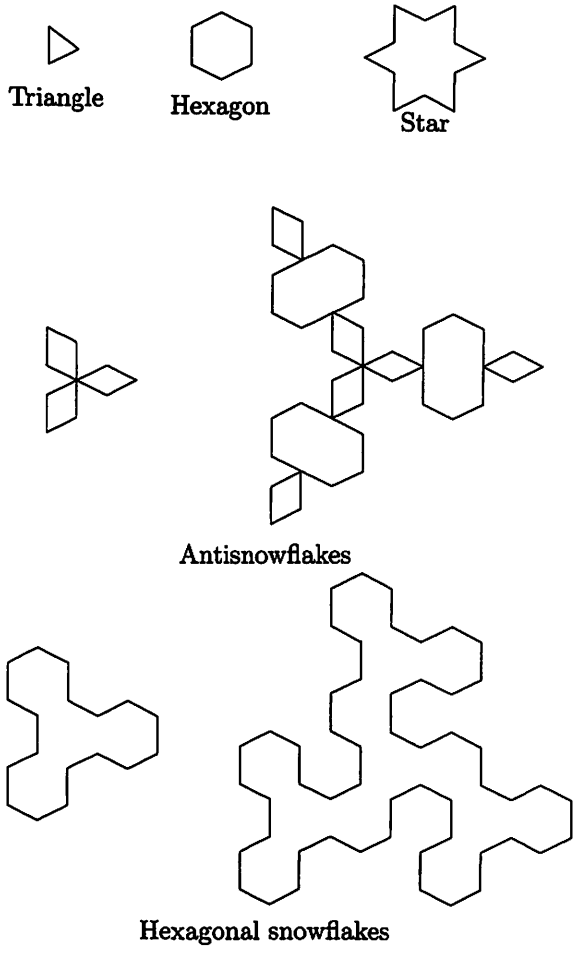


Figure 3: Line pictures over the hexagonal grid.

5 Security and Safety

The security of the above mentioned systems relies on the difficulty of the word problem. R.V. Book proved that the word problem is undecidable for Thue systems [2]. In [5], Novikov proved that the word problem for finitely presented group is undecidable. But in [8], Wrathall proved that the word problem for free partially commutative group is decidable in linear time.

In [4], an attack on partially commutative monoids and groups is presented. To avoid such attack, in this system we have introduced the code conditions. The zero knowledge protocol presented in [1] can be used to convince a Verifier that there is a morphism which maps the public finitely presented group in to a free partially commutative group.

References

[1] P.J. Abisha, D.G. Thomas, K.G. Subramanian, Public key cryptosystems based on free partially commutative monoids, *Proceeding of INDOCRYPT 2003, LNCS 2904,* (2003), 218 - 227.

[2] R.V. Book, Confluent and other type of Thue systems, *Journals of the ACM,* **29**(1987), 171 - 182.

[3] W. Diffie and M. Hellman, New direction in cryptography, *IEEE Transactions and Information Theory,* **IT-22**(6)(1976), 644 - 654.

[4] F. Levy-dit-vehal and L. Perret, Attacks on public key cryptosystems based on free partially commutative monoids and groups, *INDOCRYPT 2004,* LNCS 3348, (2004), 275 - 289.

[5] P.S. Novikov, On the algorithmic unsolvability of the word problem in group theory, *Trudy Mat. Inst. Steklov,* **44** (1955), 1 - 143.

[6] A. Salomaa, *Computation and Automata,* Cambridge University Press, 1986.

[7] G. Siromoney, R. Siromoney, K.G. Subramanian, V.R. Dare and P.J. Abisha, Generalized Parish Vector and public key cryptosystems, *in A perspective in Theoretical Computer Science - Commemorative Volume for Gift Siromoney,* Ed. R. Narasimhan, World Scientific, (1689), 301 - 323.

[8] C. Wrathall, The word problem for free partially commutative groups, *J. Symbolic Computations* **6** (1988), 99 - 104.