Commun. Korean Math. Soc. **24** (2009), No. 4, pp. 501–508
DOI 10.4134/CKMS.2009.24.4.501

A NEW REPRESENTATION ALGORITHM IN A FREE GROUP

Su-Jeong Choi

Abstract. This paper presents a new representation algorithm which computes the representation for elements of a free group generated by two linear fractional transformations and also the justification of the algorithm in order to show how it operates correctly and efficiently according to inputs.

1. Introduction

Let $n \in \mathbb{N}$ with $n \geq 3$ and Γ_n a free group [5] generated by two linear fractional transformations $A_n = \left[\begin{smallmatrix} 1 & n \\ 0 & 1 \end{smallmatrix}\right]$ and $B_n = \left[\begin{smallmatrix} 1 & 0 \\ n & 1 \end{smallmatrix}\right]$. Then every element of Γ_n can be represented by a reduced word in $\{A_n, B_n\}^{\pm}$, called the X_n-representation with $X_n = \{A_n, B_n\}$. Since A_n and B_n are the cases of ${A_1}^n$ and ${B_1}^n$, every element of Γ_n can also be represented by a reduced word in $\{A_1, B_1\}^{\pm}$, called the X_1-representation with $X_1 = \{A_1, B_1\}$. Namely, the X_n-representation of each element of Γ_n enables computation of the X_1-representation of the element. The X_1-representation of each element of Γ_n is one of the following forms

$$\begin{array}{l} {A_1}^{nu_1}{B_1}^{nu_2}\cdots{B_1}^{nu_{m-1}}{A_1}^{nu_m}, \\ {A_1}^{nu_1}{B_1}^{nu_2}\cdots{A_1}^{nu_{m-1}}{B_1}^{nu_m}, \\ {B_1}^{nu_1}{A_1}^{nu_2}\cdots{A_1}^{nu_{m-1}}{B_1}^{nu_m}, \\ {B_1}^{nu_1}{A_1}^{nu_2}\cdots{B_1}^{nu_{m-1}}{A_1}^{nu_m}, \end{array}$$

where u_i is a nonzero integer and $m \in \mathbb{N}$.

In 2004, Grigoriev and Ponomarenko introduced the homomorphic public-key cryptosystem [4]. Then the secret key n is hidden into the decryption scheme including the X_n-representation algorithm which outputs the X_n-repres entation of the ciphertext in the process of the decryption. Later on, this X_n-representation algorithm is modified to make it more clear and efficient in [2]. On the other hand, the attacker must strive to find the secret key n of the

Received April 3, 2009.
2000 *Mathematics Subject Classification.* Primary 20C20, 20F05.
Key words and phrases. linear fractional transformation, free group, representation, algorithm.

cryptosystem. So the motivation of the design of the X_1-representation algorithm derives from cryptanalysis of the homomorphic public-key cryptosystem. Consequently the X_1-representation for elements of Γ_n can be used to break the homomorphic public-key cryptosystem in [3].

In this paper, the X_1-representation algorithm is presented with the justification of the algorithm, which is stated explicitly with some properties of the two linear fractional transformations in order to show how it operates correctly and efficiently according to inputs. Note that the material of this paper is extracted from the PhD thesis of the author [1].

2. Representation algorithm in a free group Γ_n

In this section we introduce a new representation algorithm to compute the X_1-representation for elements of Γ_n and also prove correctness of the algorithm with some properties of two linear fractional transformations ${A_n}^u$ and ${B_n}^u$ for a nonzero integer u. Let D be a unit open disk in the complex plane $\mathbb{C}$ with the center 0, i.e., $D = \{z \in \mathbb{C} \mid |z| < 1\}$ and a complement of the closure of D, $D^c = \mathbb{C} - \bar{D} = \{z \in \mathbb{C} \mid |z| > 1\}$. 1_{X_1} denotes the empty word.

Assume that n is unknown and $M \in \Gamma_n$. The following representation algorithm outputs the X_1-representation of M either for $z = \frac{1}{2}$ or for $z = 2$. Otherwise, it does the error message ϵ. Once the algorithm first outputs the X_1-representation of M for one of two z values, it does not need to run for the other z value as the goal is achieved.

Algorithm

Step 0
$w \leftarrow 1_{X_1}$
$L \leftarrow M$
Step 1
(1) $L(z) = 0, |L(z)| = 1, L(z) = \infty \Rightarrow$ output ϵ.
(2) $|L(z)| > 1 \Rightarrow$ compute e, μ s.t. $L(z) = e + \mu$, $e \in \mathbb{Z}$, $-\frac{1}{2} < \mu \le \frac{1}{2}$ and go to Step 2.
(3) $|L(z)| < 1 \Rightarrow$ compute e, μ s.t. $\frac{1}{L(z)} = e + \mu$, $e \in \mathbb{Z}$, $-\frac{1}{2} < \mu \le \frac{1}{2}$ and go to Step 3.
Step 2
(1) $C \leftarrow {A_1}^e$ and $w \leftarrow wC$.
(2) $C = I \Rightarrow$ output ϵ.
(3) $L \leftarrow C^{-1}L$
(4) $L = I \Rightarrow$ output w. Otherwise, return Step 1.
Step 3
(1) $C \leftarrow {B_1}^e$ and $w \leftarrow wC$.
(2) $C = I \Rightarrow$ output ϵ.
(3) $L \leftarrow C^{-1}L$
(4) $L = I \Rightarrow$ output w. Otherwise, return Step 1.

Next, some properties of the two linear fractional transformations are as follows.

Lemma 2.1 ([2]). *For $z \in D$, ${A_n}^u(z) \in D^c$.*

Lemma 2.2 ([2]). *For $z \in D^c$, ${B_n}^u(z) \in D$.*

Theorem 2.3 ([2]). *The following properties hold*:

(1) ${A_n}^{u_1}{B_n}^{u_2} \cdots {B_n}^{u_{m-1}}{A_n}^{u_m} \in D^c$ *for* $z \in D$.
(2) ${A_n}^{u_1}{B_n}^{u_2} \cdots {A_n}^{u_{m-1}}{B_n}^{u_m} \in D^c$ *for* $z \in D^c$.
(3) ${B_n}^{u_1}{A_n}^{u_2} \cdots {A_n}^{u_{m-1}}{B_n}^{u_m} \in D$ *for* $z \in D^c$.
(4) ${B_n}^{u_1}{A_n}^{u_2} \cdots {B_n}^{u_{m-1}}{A_n}^{u_m} \in D$ *for* $z \in D$.

Theorem 2.4. *Let $n \geq 3$ and $z \in \mathbb{R}$ such that $|z| < \frac{1}{2}$. Then $|{A_n}^u(z)| > \frac{5}{2}$.*

Proof. For $n \geq 3$ and $z \in \mathbb{R}$ s.t. $|z| < \frac{1}{2}$, by Lemma 2.1 ${A_n}^u(z) = nu + z \in D^c$. If $u \geq 1$, then $\frac{5}{2} \leq nu - \frac{1}{2} < nu + z < nu + \frac{1}{2}$ and so ${A_n}^u(z) > \frac{5}{2}$. If $u < -1$, then $nu - \frac{1}{2} < nu + z < nu + \frac{1}{2} < -\frac{5}{2}$ and so ${A_n}^u(z) < -\frac{5}{2}$. □

Theorem 2.5. *For $n \geq 3$ and $z \in \mathbb{R} \cap D^c$, $|{B_n}^u(z)| < \frac{1}{2}$.*

Proof. Let $n \geq 3$ and $z \in \mathbb{R} \cap D^c$. If $u \geq 1$, then $0 < \frac{1}{nu+\frac{1}{z}} < \frac{1}{nu-1} \leq \frac{1}{2}$ and so $0 < {B_n}^u(z) < \frac{1}{2}$. If $u < -1$, then $-\frac{1}{2} < \frac{1}{nu+1} < \frac{1}{nu+\frac{1}{z}} < 0$ and so $-\frac{1}{2} < {B_n}^u(z) < 0$. □

It should be noticed that both properties above are not guaranteed for $n \geq 2$. In other words, let $n \geq 2$ and $z \in \mathbb{R} \cap D^c$, then $|{B_n}^u(z)| < 1$, but for $n \geq 3$, $|{B_n}^u(z)| < \frac{1}{2}$. Now the justification of the algorithm is being carried out with some characteristics of the two linear fractional transformations. So it shows how the algorithm works correctly and efficiently according to inputs. For the sake of avoiding the similarity of proofs, one of the X_1-representation forms is taken for the verification of the algorithm.

Theorem 2.6. *If a matrix $M = {A_n}^{u_1}{B_n}^{u_2} \cdots {A_n}^{u_{m-1}}{B_n}^{u_m}$ is input to the algorithm ($z = \frac{1}{2}$) with even $m \geq 2$, then it outputs ϵ as the error message.*

Proof. Let

$$M = {A_n}^{u_1}{B_n}^{u_2} \cdots {A_n}^{u_{m-1}}{B_n}^{u_m} \in \Gamma_n \text{ and } \beta_1 = {B_n}^{u_2} \cdots {A_n}^{u_{m-1}}{B_n}^{u_m}(\frac{1}{2}).$$

Then $L(\frac{1}{2}) = {A_n}^{u_1}{B_n}^{u_2} \cdots {A_n}^{u_{m-1}}{B_n}^{u_m}(\frac{1}{2}) = nu_1 + \beta_1$.
Case 1 : If $n = 3$ and $u_m = -1$, then $|{B_n}^{u_m}(\frac{1}{2})| = 1$ and in Step 1 of the first iteration, by Theorem 2.3(2),

$$L(\tfrac{1}{2}) = {A_n}^{u_1}{B_n}^{u_2} \cdots {B_n}^{u_{m-2}}(nu_{m-1} - 1) \in D^c.$$

As ${A_n}^{u_{m-1}}{B_n}^{u_m}(\frac{1}{2}) = {A_n}^{u_{m-1}}(-1) \in D^c$, by Theorem 2.5, $|\beta_1| < \frac{1}{2}$, so that $e = nu_1$ and $\mu = \beta_1$.
Case 2 : If $n \neq 3$ or $u_m \neq -1$, then $|{B_n}^{u_m}(\frac{1}{2})| < 1$. By Theorem 2.3(1),

$$L(\tfrac{1}{2}) = {A_n}^{u_1}{B_n}^{u_2} \cdots {A_n}^{u_{m-1}}(\tfrac{1}{nu_m+2}) \in D^c$$

and by Theorem 2.5, $|\beta_1| < \frac{1}{2}$, so that $e = nu_1$ and $\mu = \beta_1$. In Step 2 of the first iteration of those cases, $C = {A_1}^e = {A_1}^{nu_1}$, $w = wC = {A_1}^{nu_1}$ and $L = C^{-1}L = {B_n}^{u_2} \cdots {A_n}^{u_{m-1}} {B_n}^{u_m} \neq I$. So return Step 1.

Suppose that for $1 \leq i-1 < m-2$, in Step 2 of the $i-1$th iteration, $L = C^{-1}L = {B_n}^{u_i} {A_n}^{u_{i+1}} \cdots {A_n}^{u_{m-1}} {B_n}^{u_m}$ or in Step 3 of the $i-1$th iteration $L = C^{-1}L = {A_n}^{u_i} {B_n}^{u_{i+1}} \cdots {A_n}^{u_{m-1}} {B_n}^{u_m}$ according as $i-1$ is odd or even.

For even i, let $L = {B_n}^{u_i} {A_n}^{u_{i+1}} \cdots {A_n}^{u_{m-1}} {B_n}^{u_m}$ in Step 1 of the ith iteration and $\alpha_i = {A_n}^{u_{i+1}} {B_n}^{u_{i+2}} \cdots {A_n}^{u_{m-1}} {B_n}^{u_m}(\frac{1}{2})$. Then

$$L(\tfrac{1}{2}) = {B_n}^{u_i}(\alpha_i) = \frac{1}{nu_i + \frac{1}{\alpha_i}}.$$

Case 1 : If $n = 3$ and $u_m = -1$, then $|{B_n}^{u_m}(\frac{1}{2})| = 1$ and by Theorem 2.3(3),

$$L(\tfrac{1}{2}) = {B_n}^{u_i} {A_n}^{u_{i+1}} \cdots {B_n}^{u_{m-2}}(nu_{m-1} - 1) \in D.$$

As ${A_n}^{u_{m-1}} {B_n}^{u_m}(\frac{1}{2}) = {A_n}^{u_{m-1}}(-1) \in D^c$, by Theorem 2.5,

$$|{B_n}^{u_{i+2}} \cdots {A_n}^{u_{m-1}} {B_n}^{u_m}(\tfrac{1}{2})| < \tfrac{1}{2}.$$

By Theorem 2.4, $\frac{1}{|\alpha_i|} < \frac{2}{5}$ and as $\frac{1}{L(\frac{1}{2})} = nu_i + \frac{1}{\alpha_i}$,

$$e = nu_i \text{ and } \mu = \frac{1}{\alpha_i}.$$

Case 2 : If $n \neq 3$ or $u_m \neq -1$, then $|{B_n}^{u_m}(\frac{1}{2})| < 1$ and by Theorem 2.3(4),

$$L(\tfrac{1}{2}) = {B_n}^{u_i} {A_n}^{u_{i+1}} \cdots {A_n}^{u_{m-1}}(\tfrac{1}{nu_m + 2}) \in D.$$

By Lemma 2.1, ${A_n}^{u_{m-1}}(\frac{1}{nu_m+2}) \in D^c$ and by Theorem 2.5,

$$|{B_n}^{u_{i+2}} \cdots {A_n}^{u_{m-1}} {B_n}^{u_m}(\tfrac{1}{2})| < \tfrac{1}{2}.$$

By Theorem 2.4, $\frac{1}{|\alpha_i|} < \frac{2}{5}$ and $\frac{1}{L(\frac{1}{2})} = nu_i + \frac{1}{\alpha_i}$, so that $e = nu_i$ and $\mu = \frac{1}{\alpha_i}$. In Step 3 of the ith iteration of those cases, $C = {B_1}^e = {B_1}^{nu_i}$, $w = wC = {A_1}^{nu_1} {B_1}^{nu_2} \cdots {A_1}^{nu_{i-1}} {B_1}^{nu_i}$ and $L = C^{-1}L = {A_n}^{u_{i+1}} \cdots {A_n}^{u_{m-1}} {B_n}^{u_m} \neq I$. So return Step 1.

For odd i, let $L = {A_n}^{u_i} {B_n}^{u_{i+1}} \cdots {A_n}^{u_{m-1}} {B_n}^{u_m}$ and $\beta_i = {B_n}^{u_{i+1}} \cdots {A_n}^{u_{m-1}} {B_n}^{u_m}(\frac{1}{2})$. Then $L(\frac{1}{2}) = nu_i + \beta_i$.

Case 1 : If $n = 3$ and $u_m = -1$, then $|{B_n}^{u_m}(\frac{1}{2})| = 1$ and by Theorem 2.3(2),

$$L(\tfrac{1}{2}) = {A_n}^{u_i} {B_n}^{u_{i+1}} \cdots {B_n}^{u_{m-2}}(nu_{m-1} - 1) \in D^c.$$

As ${A_n}^{u_{m-1}} {B_n}^{u_m}(\frac{1}{2}) = {A_n}^{u_{m-1}}(-1) \in D^c$, by Theorem 2.5, $|\beta_i| < \frac{1}{2}$. Hence

$$e = nu_i \text{ and } \mu = \beta_i.$$

Case 2 : If $n \neq 3$ or $u_m \neq -1$, then $|{B_n}^{u_m}(\frac{1}{2})| < 1$. By Theorem 2.3(1),

$$L(\tfrac{1}{2}) = {A_n}^{u_i} {B_n}^{u_{i+1}} \cdots {A_n}^{u_{m-1}}(\tfrac{1}{nu_m + 2}) \in D^c$$

and by Theorem 2.5, $|\beta_i| < \frac{1}{2}$, so that $e = nu_i$ and $\mu = \beta_i$. In Step 2 of the ith iteration of those cases,

$$C = {A_1}^e = {A_1}^{nu_i}, w = wC = {A_1}^{nu_1}{B_1}^{nu_2}\cdots{B_1}^{u_{i-1}}{A_1}^{nu_i}$$
$$\text{and } L = C^{-1}L = {B_n}^{u_{i+1}}\cdots{A_n}^{u_{m-1}}{B_n}^{u_m} \neq I.$$

So return Step 1.

For $i = m-1$, the algorithm runs with $L = C^{-1}L = {A_n}^{u_{m-1}}{B_n}^{u_m}$ in Step 3 of the $m-2$th iteration.

Case 1 : If $n = 3$ and $u_m = -1$, then $|{B_n}^{u_m}(\frac{1}{2})| = 1$ and

$$L(\tfrac{1}{2}) = {A_n}^{u_{m-1}}{B_n}^{u_m}(\tfrac{1}{2}) = {A_n}^{u_{m-1}}(-1) \in D^c.$$

In Step 1 of the $m-1$th iteration,

$$e = nu_{m-1} - 1 \text{ and } \mu = 0.$$

In Step 2 of the $m-1$th iteration,

$$C = {A_1}^e = {A_1}^{nu_{m-1}-1},$$
$$w = wC = {A_1}^{nu_1}{B_1}^{nu_2}\cdots{B_1}^{nu_{m-2}}{A_1}^{nu_{m-1}-1}$$

and

$$L = C^{-1}L = A_1{B_n}^{u_m} \neq I.$$

So return Step 1.

For $i = m$ of case 1, the algorithm runs with $L = A_1{B_n}^{u_m}$ in Step 1 of the mth iteration and then

$$L(\tfrac{1}{2}) = A_1{B_n}^{u_m}(\tfrac{1}{2}) = A_1(-1) = 0.$$

Therefore the algorithm outputs ϵ as the error message and then it terminates.

Case 2 : If $n \neq 3$ or $u_m \neq -1$, then $|{B_n}^{u_m}(\frac{1}{2})| < 1$ and

$$L(\tfrac{1}{2}) = {A_n}^{u_{m-1}}{B_n}^{u_m}(\tfrac{1}{2}) = {A_n}^{u_{m-1}}(\tfrac{1}{nu_m+2}) \in D^c.$$

Since $|{B_n}^{u_m}(\frac{1}{2})| \leq \frac{1}{2}$,

$$e = nu_{m-1} \text{ and } \mu = \tfrac{1}{nu_m+2}.$$

In Step 2 of the $m-1$th iteration,

$$C = {A_1}^e = {A_1}^{nu_{m-1}},$$
$$w = wC = {A_1}^{nu_1}{B_1}^{nu_2}\cdots{B_1}^{nu_{m-2}}{A_1}^{nu_{m-1}}$$
$$\text{and } L = C^{-1}L = {B_n}^{u_m} \neq I.$$

So return Step 1.

For $i = m$ of case 2, the algorithm runs with $L = {B_n}^{u_m}$ in Step 1 of the mth iteration.

Case 1 : If $n = 3$ and $u = -1$, then in Step 1 of the mth iteration, $|L(\frac{1}{2})| = 1$. So the algorithm outputs ϵ as the error message and then it terminates.

Case 2 : If $n \neq 3$ or $u_m \neq -1$, then in Step 1 of the mth iteration, $|L(\frac{1}{2})| \leq \frac{1}{2}$ and so

$$e = \frac{1}{L(\frac{1}{2})} = nu_m + 2 \text{ and } \mu = 0.$$

In Step 3 of the mth iteration,

$$C = {B_1}^e = {B_1}^{nu_m+2},$$

$$w = wC = {A_1}^{nu_1}{B_1}^{nu_2}\cdots{B_1}^{nu_{m-2}}{A_1}^{nu_{m-1}}{B_1}^{nu_m+2}$$

and

$$L = C^{-1}L = {B_1}^{-nu_m-2}{B_n}^{u_m} = {B_1}^{-2} \neq I.$$

So return Step 1.

For $i = m+1$ of case 2, the algorithm runs with $L = C^{-1}L = {B_1}^{-2}$ in Step 3 of the mth iteration and then

$$L(\tfrac{1}{2}) = {B_1}^{-2}(\tfrac{1}{2}) = \infty$$

in Step 1 of the $m+1$th iteration. Therefore the algorithm outputs ϵ as the error message and then it terminates. □

Theorem 2.7. *If a matrix $M = {A_n}^{u_1}{B_n}^{u_2}\cdots{A_n}^{u_{m-1}}{B_n}^{u_m}$ is input to the algorithm ($z = 2$) with even $m \geq 2$, then it outputs ${A_1}^{nu_1}{B_1}^{nu_2}\cdots{A_1}^{nu_{m-1}}{B_1}^{nu_m}$ as the X_1-representation of M.*

Proof. Let

$$M = {A_n}^{u_1}{B_n}^{u_2}\cdots{A_n}^{u_{m-1}}{B_n}^{u_m} \in \Gamma_n \text{ and } \beta_1 = {B_n}^{u_2}\cdots{A_n}^{u_{m-1}}{B_n}^{u_m}(2).$$

Then

$$L(2) = {A_n}^{u_1}{B_n}^{u_2}\cdots{A_n}^{u_{m-1}}{B_n}^{u_m}(2) = nu_1 + \beta_1.$$

By Theorem 2.3(2), $L(2) \in D^c$ and by Theorem 2.5, $|\beta_1| < \frac{1}{2}$, so that $e = nu_1$ and $\mu = \beta_1$. In Step 2 of the first iteration,

$$C = {A_1}^e = {A_1}^{nu_1}, w = wC = {A_1}^{nu_1}$$
$$\text{and } L = C^{-1}L = {B_n}^{u_2}\cdots{A_n}^{u_{m-1}}{B_n}^{u_m} \neq I.$$

So return Step 1.

Assume that for $1 \leq i-1 < m-1$, in Step 3 of the $i-1$th iteration $L = C^{-1}L = {A_n}^{u_i}{B_n}^{u_{i+1}}\cdots{A_n}^{u_{m-1}}{B_n}^{u_m}$ or in Step 2 of the $i-1$th iteration $L = C^{-1}L = {B_n}^{u_i}{A_n}^{u_{i+1}}\cdots{A_n}^{u_{m-1}}{B_n}^{u_m}$ according as $i-1$ is even or odd.

For even i, in Step 1 of the ith iteration, let $L = {B_n}^{u_i}{A_n}^{u_{i+1}}\cdots{A_n}^{u_{m-1}}{B_n}^{u_m} \in \Gamma_n$ and $\alpha_i = {A_n}^{u_{i+1}}\cdots{A_n}^{u_{m-1}}{B_n}^{u_m}(2)$. Then

$$L(2) = {B_n}^{u_i}(\alpha_i) = \frac{1}{nu_i + \frac{1}{\alpha_i}}.$$

By Theorem 2.5, $|L(2)| < \frac{1}{2}$ and by Theorem 2.4, $\frac{1}{|\alpha_i|} < \frac{2}{5}$, so that

$$e = nu_i \text{ and } \mu = \tfrac{1}{\alpha_i}.$$

In Step 3 of the ith iteration, $C = {B_1}^e = {B_1}^{nu_i}$, $w = wC = {A_1}^{nu_1}{B_1}^{nu_2}\cdots {A_1}^{nu_{i-1}}{B_1}^{nu_i}$ and $L = C^{-1}L = {A_n}^{u_{i+1}}\cdots {A_n}^{u_{m-1}}{B_n}^{u_m} \neq I$. So return Step 1.

For odd i, let $L = {A_n}^{u_i}{B_n}^{u_{i+1}}\cdots {A_n}^{u_{m-1}}{B_n}^{u_m} \in \Gamma_n$ in Step 1 of the ith iteration and $\beta_i = {B_n}^{u_{i+1}}\cdots {A_n}^{u_{m-1}}{B_n}^{u_m}(2)$. Then

$$L(2) = {A_n}^{u_i}{B_n}^{u_{i+1}}\cdots {A_n}^{u_{m-1}}{B_n}^{u_m} = nu_i + \beta_i$$

and by Theorem 2.5, $|\beta_i| < \frac{1}{2}$, so that $e = nu_i$ and $\mu = \beta_i$. In Step 2 of the ith iteration, $C = {A_1}^e = {A_1}^{nu_i}$, $w = wC = {A_1}^{nu_1}{B_1}^{nu_2}\cdots {B_1}^{nu_{i-1}}{A_1}^{nu_i}$ and $L = C^{-1}L = {B_n}^{u_{i+1}}\cdots {A_n}^{u_{m-1}}{B_n}^{u_m} \neq I$. So return Step 1.

If $i = m$, then the algorithm runs with $L = {B_n}^{u_m}$ in Step 1 of the mth iteration and by Theorem 2.5, $|L(2)| < \frac{1}{2}$. As $\frac{1}{L(2)} = nu_m + \frac{1}{2}$,

$$e = nu_m \text{ and } \mu = \tfrac{1}{2}.$$

In Step 3 of the mth iteration,

$$C = {B_1}^e = {B_1}^{nu_m}, w = wC = {A_1}^{nu_1}{B_1}^{nu_2}\cdots {A_1}^{nu_{m-1}}{B_1}^{nu_m}$$

and

$$L = C^{-1}L = {B_1}^{-nu_m}{B_n}^{u_m} = I.$$

Hence the algorithm outputs

$${A_1}^{nu_1}{B_1}^{nu_2}\cdots {A_1}^{nu_{m-1}}{B_1}^{nu_m}$$

as the X_1-representation of M and then it terminates. □

Theorem 2.8.

(1) *If a matrix* $M = {A_n}^{u_1}{B_n}^{u_2}\cdots {B_n}^{u_{m-1}}{A_n}^{u_m}$ *is input to the algorithm* $(z=\frac{1}{2})$ *with odd* $m \geq 3$, *then it outputs* ${A_1}^{nu_1}{B_1}^{nu_2}\cdots {B_1}^{nu_{m-1}}{A_1}^{nu_m}$ *as the* X_1*-representation of* M.

(2) *If a matrix* $M = {A_n}^{u_1}{B_n}^{u_2}\cdots {B_n}^{u_{m-1}}{A_n}^{u_m}$ *is input to the algorithm* $(z=2)$ *with odd* $m \geq 3$, *then it outputs* ϵ *as the error message.*

(3) *If a matrix* $M = {B_n}^{u_1}{A_n}^{u_2}\cdots {B_n}^{u_{m-1}}{A_n}^{u_m}$ *is input to the algorithm* $(z=\frac{1}{2})$ *with even* $m \geq 2$, *then it outputs* ${B_1}^{nu_1}{A_1}^{nu_2}\cdots {B_1}^{nu_{m-1}}{A_1}^{nu_m}$ *as the* X_1*-representation of* M.

(4) *If a matrix* $M = {B_n}^{u_1}{A_n}^{u_2}\cdots {B_n}^{u_{m-1}}{A_n}^{u_m}$ *is input to the algorithm* $(z=2)$ *with even* $m \geq 2$, *then it outputs* ϵ *as the error message.*

(5) *If a matrix* $M = {B_n}^{u_1}{A_n}^{u_2}\cdots {A_n}^{u_{m-1}}{B_n}^{u_m}$ *is input to the algorithm* $(z=\frac{1}{2})$ *with odd* $m \geq 3$, *then it outputs* ϵ *as the error message.*

(6) *If a matrix* $M = {B_n}^{u_1}{A_n}^{u_2}\cdots {A_n}^{u_{m-1}}{B_n}^{u_m}$ *is input to the algorithm* $(z=2)$ *with odd* $m \geq 3$, *then it outputs* ${B_1}^{nu_1}{A_1}^{nu_2}\cdots {A_1}^{nu_{m-1}}{B_1}^{nu_m}$ *as the* X_1*-representation of* M.

Proof. It is similar to the proofs of Theorem 2.6 and Theorem 2.7. □

3. Conclusions

The purpose of this paper is to design a new representation algorithm for elements of a free group generated by the two linear fractional transformations and also show proofs of correctness of the algorithm, which are dominant in this note. This work seemingly looks more or less straightforward, but indeed, it clarifies even subtle cases in which the algorithm may not work properly. Subsequently the algorithm comes to have computational efficiency. Moreover some theoretical background of the algorithm is apparently shown with the properties of the two linear fractional transformations. Further from combinatorial group theoretical point of view, the X_1-representation algorithm might give an insight to design algorithms for other groups such as symplectic group $\mathrm{Sp}(2,1)$, special linear group $\mathrm{SL}(2,\mathbb{Z})$ or general linear group $\mathrm{GL}(2,\mathbb{Z})$. In practice programming of the algorithm and demonstrations with experiments appear in [1].

References

[1] S. J. Choi, *Cryptanalysis of a homomorphic public-key cryptosystem over a finite group*, Ph.D. thesis, University of London, 2006.

[2] ______, *Representation algorithms in some free groups*, J. Korea Soc. Math. Educ. Ser. B : Pure Appl. Math. **15** (2008), no. 3, 229–243.

[3] S. J. Choi, S. R. Blackburn, and P. R. Wild, *Cryptanalysis of a homomorphic public-key cryptosystem over a finite group*, J. Math. Crypt. **1** (2007), no. 4, 351–358.

[4] D. Grigoriev and I. Ponomarenko, *Homomorphic public key cryptosystems over groups and rings*, Complexity of computations and proofs, Quad. Math. **13** (2004), 305–325.

[5] R. C. Lyndon and P. E. Schupp, *Combinatorial Group Theory*, Springer, Berlin, New York, 1977.

Department of Mathematics
Dong-A University
Busan 604-714, Korea
E-mail address: sjchoi09@donga.ac.kr