Tatra Mt. Math. Publ. **33** (2006), 137–148

**TATRA MOUNTAINS**
*Mathematical Publications*

# ON PUBLIC-KEY CRYPTOSYSTEMS BASED ON COMBINATORIAL GROUP THEORY

Jean-Camille Birget* — Spyros S. Magliveras† — Michal Sramka†

ABSTRACT. We analyze and critique the public-key cryptosystem, based on combinatorial group theory, that was proposed by Wagner and Magyarik in 1984. This idea is actually not based on the word problem but on another, generally easier, premise problem. Moreover, the idea of the Wagner–Magyarik system is vague, and it is difficult to find a secure realization of this idea. We describe a public-key cryptosystem inspired in part by the Wagner–Magyarik idea, but we also use group actions on words.

## 1. Introduction

A number of public-key cryptosystems based on combinatorial group theory have been proposed since the early 1980s, the first of which was probably the outline of Wagner and Magyarik [13]. A good overview of various other group-based systems is given in the dissertation of M. I. González Vasco [6]; see also [8].

In this paper we present a critique of the Wagner–Magyarik system, and propose a public-key cryptosystem based on finitely presented groups with hard word problem, which are also transformation groups.

In order to make the paper more self-contained, we give some basic definitions from the combinatorial group theory. More details and rigor can be found in texts like [10] or [11].

Let $G$ be a group, defined by a presentation $(X, R)$, where $X = \{x_1, x_2, \dots\}$ is a set of generators and $R = \{r_1, r_2, \dots\}$ is a set of relators. When the sets $X$

2000 Mathematics Subject Classification: 20F10, 94A62, 68Q15.
Keywords: public-key cryptosystem, combinatorial group theory, Richard Thompson groups, (NP ∩ coNP)-complete premise problems.

* JCB is supported in part by NSF grant CCR-0310793.
† Work supported in part by a Federal Earmark grant for *Research in Secure Telecommunication Networks* (2004–05).

and $R$ are both finite we say that the group $G$ is *finitely-presented.* A *word* $w$ over $X$ is a finite sequence of elements of the set $X \cup X^{-1}$. The *empty word* is the empty sequence, of length $0$. A word which defines the identity element in the group $G$ is called a *relator*. We say that two words $w$ and $w'$ are *equivalent* for the presentation $(X, R)$ iff the following operations, applied a finite number of times, transform $w$ into $w'$:

(T1) Insertion of one of the relators $r_1, r_1^{-1}, r_2, r_2^{-1}, \dots \in R \cup R^{-1}$, or of a trivial relator (of the form $x_i x_i^{-1}$ or $x_i^{-1} x_i$ with $x_i \in X$) at the beginning of a word, at the end of a word, or between any two consecutive symbols of a word.

(T2) Deletion of one of the relators $r_1, r_1^{-1}, r_2, r_2^{-1}, \dots$, or of a trivial relator, if it forms a block of consecutive symbols in a word.

An application of one transformation of the form (T1) or (T2) is called a *rewrite step.* Every element $g$ of $G = (X, R)$ can be described by a word over $X \cup X^{-1}$ usually in many ways; the length of the shortest word that describes $g$ is called the *word length of* $g$. For a word $w$ over some fixed alphabet we denote the length of $w$ by $|w|$; also, for $g \in G = (X, R)$ we denote the word length of $g$ by $|g|$.

The *word problem* of a group with generating set $X$, as introduced by Max Dehn in 1911, is the following decision problem: For an arbitrary word $w$ over $X \cup X^{-1}$, is $w$ equivalent to the empty word?

In the 1950's, Novikov and Boone independently showed that there are finite group presentations whose word problem is undecidable. It is an important fact that the decidability and the complexity of the word problem of a finitely generated group depend only on the group, and not on the generators or the presentation chosen (provided that one sticks to finite generating sets). In other words, if $G$ has a decidable word problem for some finite generating set $X$, then $G$ has a decidable word problem for every finite generating set. Concerning complexity, a change of the finite generating set changes the complexity only linearly (see [12]). Therefore we are allowed to talk about "the word problem of a group $G$" without referring to a specific presentation.

It was proved more recently that there are finitely presented groups whose word problem is NP-complete [14], [4], or whose word problem is coNP-complete [1].

By a group with an *easy* word problem we will understand a group whose word problem is decidable in deterministic polynomial time. The other groups are said to have a *hard* word problem.

We will also deal with the following variant of the word problem, which we call *the word choice problem.* Let us fix a group $G$ with a finite generating set $X$, and let us fix two words $w_0$ and $w_1$ over $X \cup X^{-1}$.

INPUT: A word $w$ over $X \cup X^{-1}$.

PREMISE: $w$ is either equivalent to $w_0$ or to $w_1$.

QUESTION: Is $w$ equivalent to $w_0$?

Note that this is a "premise problem"[1], i.e., a problem with restrictions (pre-condition) on the input; an algorithm for solving a premise problem can assume that the pre-condition holds, and is not required to give correct answers (or any answer at all) on inputs that violate the pre-condition.

The word choice problem is rather different from the word problem. E.g., for a finitely presented group, the word choice problem is always decidable; and for a group with word problem in NP or in coNP, the word choice problem is in NP $\cap$ coNP. One sees from these examples that the word choice problem can be much easier than the word problem.

## 2. Critique of the Wagner–Magyarik system

In 1984 Wagner and Magyarik [13] proposed a public-key cryptosystem "based on the word problem". The general scheme follows.

**Setup:** Let $X$ be a finite set of generators, and let $R$ and $S$ be finite sets of relators such that the group $G = (X, R)$ has a hard word problem, and the group $G' = (X, R \cup S)$ has an easy word problem. Choose two words $w_0$ and $w_1$, which are not equivalent in $G'$ (and hence not equivalent in $G$ either). *Public key:* The presentation $(X, R)$ and the words $w_0$ and $w_1$.

**Encryption:** To encrypt a single bit $i \in \{0, 1\}$, pick $w_i$ and transform it into a ciphertext word $w$ by repeatedly and randomly applying the transformations (T1) and (T2) for the presentation $(X, R)$.

**Decryption:** To decrypt a word $w$, run the algorithm for the word problem of $G'$ in order to decide which of $ww_0^{-1}$ and $ww_1^{-1}$ is equivalent to the empty word for the presentation $(X, R \cup S)$.

The *private key* is the set $S$. Actually, this is not sufficient (and [13] is not very precise at this point): the public key should be a deterministic polynomial-time algorithm for the word problem of $G' = (X, R \cup S)$; indeed, just knowing $S$ does not automatically and explicitly give us an efficient algorithm (even if such an algorithm exists).

To make their system concrete, Wagner and Magyarik introduce the following collection of finitely-presented groups: The set of generators is $X =$

[1]In the complexity literature, premise problems are usually called promise problems; however, the word 'premise' is the appropriate logical term; look up 'premise' in the Merriam-Webster Dictionary `http://www.m-w.com/home.htm`

$\{x_1, x_2, \ldots, x_m\}$, and the set of relators $R$ is any set of words of the following three types:

(R1) $y_i y_j y_k y_\ell y_i^{-1} y_k^{-1} y_j^{-1} y_\ell^{-1}$,

(R2) $y_i y_j y_k y_i^{-1} y_j^{-1} y_k^{-1}$,

(R3) $y_i y_j y_k y_i^{-1} y_k^{-1} y_j^{-1}$,

where $y_i$, $y_j$, $y_k$, and $y_\ell$ stand for generators or inverses of generators, not necessarily distinct. We will call such presentations *Wagner–Magyarik presentations.*

For the private key $S$ they propose any set of words of the following three types:

(S1) $x_i$ (elimination of a generator),

(S2) $x_i x_j^{-1}$ (collapse of two generators into one),

(S3) $x_i x_j x_i^{-1} x_j^{-1}$ (commutator of two generators),

where $x_i$ and $x_j$ are any generators. A requirement on $S$ is that it should contain enough relators, so that the group $G' = (X, R \cup S)$ is isomorphic to a *partially commutative free group*, i.e., a group generated by a subset of $X$ and presented by a few commutation relations between generators. This will guarantee that the word problem of $G'$ can be decided in polynomial time [16]. The words $w_0, w_1$ need to be chosen so that they are not equivalent in $G'$.

**Critique**

1. *Vagueness of the general scheme:* In its general form the Wagner–Magyarik cryptosystem is far too vague. To turn their idea into an actual cryptosystem, design questions would need to be answered:

(1) How do we find appropriate presentations $(X, R)$ and $(X, R \cup S)$ as well as a polynomial-time algorithm for the word problem of $(X, R \cup S)$?

(2) How do we find appropriate words $w_0$ and $w_1$?

(3) How is the random application of the transformations (T1) and (T2) carried out, and when does it stop?

(4) Finally, once all these design choices have been specified, how secure is this cryptosystem?

2. *Vagueness and insecurity of the concrete specification:* In their specific example Wagner and Magyarik give an answer to design question (1), albeit an unsatisfactory one. Design questions (2), (3) and (4) are left open. Concerning (1), it is an open problem whether the word problem of the Wagner–Magyarik presentations is hard. It is certainly not hard for every choice of (R1), (R2), (R3); e.g., some of the choices lead to commutative groups. This means that in

the Wagner–Magyarik system key generation is problematic: making sure that the chosen $R$ makes the word problem of $(X, R)$ hard is itself apparently a hard problem. Concerning (4), a reaction attack [7] and a chosen-ciphertext attack are possible, both of complexity $O(m^2)$.

3. *Spurious keys:* Another problem (already mentioned in [13]) is the existence of spurious keys. More precisely, in order to decrypt one does not explicitly need the presentation $(X, R \cup S)$. Any homomorphic image of $G$ with an easy word problem will decrypt, as long as it separates $w_0$ and $w_1$. So, even if $S$ might be hard to find, one also has to prove that any homomorphic image of $G$ with an easy word problem is hard to find; this adds to the difficulty of proving the security of any concrete cryptosystem that follows the Wagner–Magyarik approach.

4. *Word choice problem:* An analytical flaw in the Wagner–Magyarik paper (and subsequent papers that comment on their paper) is the claim that the system is based on the word problem. In reality, it is based on the *word choice problem*, which we introduced earlier. We have already pointed out that the word choice problem can be much easier than the word problem. In particular, it seems unlikely that this system could ever lead to NP-completeness. Instead, (NP $\cap$ coNP)-completeness is more likely to be the highest difficulty that we can hope for, regarding robustness to attack. It is generally believed that NP $\cap$ coNP is a strict subclass of NP. Although no (NP $\cap$ coNP)-complete decision problem is known (see e.g., [5], page 116), it is not hard to see that for every NP-complete decision problem one can construct a (NP $\cap$ coNP)-complete *premise* problem. See the Appendix for details.

5. *In summary:* The Wagner–Magyarik cryptosystem is not a cryptosystem, but an approach towards finding new public-key cryptosystems. As a research approach it is worthwhile, however, leading to interesting (yet unsolved) problems.

## 3. A public-key cryptosystem based on finitely presented transformation groups

We describe a public-key cryptosystem that has some similarity with the Wagner–Magyarik system, as far as the encryption is concerned. However, we use a group $G$ whose word problem is known to be coNP-complete. The main difference is that for decryption we use the action of the group on words (instead of Wagner and Magyarik's homomorphic image $G'$).

Our contribution is that (referring to point 1 in our critique of the Wagner-Magyarik system) we answer the design questions (1) and (2). Design question (3) is addressed, but our method needs further study, and probably further improvements. Regarding question (4), the security of our scheme is much better motivated than the security of the original Wagner–Magyarik system, but it is necessarily limited (due to the multitude of hard open problems in complexity, combinatorial group theory, and cryptography).

We pick a finitely presented group $G = (X, R)$ together with a faithful transitive action of $G$ on $\{0,1,2\}^*$ (the set of all strings over the alphabet $\{0,1,2\}$). We can assume that the word problem of $G$ is coNP-complete. We conjecture that the word choice problem of $G$ is $(\mathrm{NP} \cap \mathrm{coNP})$-complete. The Appendix deals with a semigroup version of this question.

An example of such a group is constructed in [1], where it is called $G = \langle G_{3,1}^{\bmod 3}(0,1;\#) \cup \{\kappa_{321}\}\rangle$; it is closely related to the Higman-Thompson group $G_{3,1}$ (generalizing Richard Thompson's infinite finitely presented simple group $G_{2,1}$). This group has the property that if two elements $g_0, g_1 \in G$ of the word length $\leq n$ are different, then there exists a word $z \in \{0,1,2\}^*$ of length $O(n)$ on which $g_0$ and $g_1$ act differently. Moreover, given a word $z \in \{0,1,2\}^*$ and a word $w$ over a finite generating set of $G$, the word $(z)w \in \{0,1,2\}^*$ (resulting from the action of $w$ on $z$) can be computed in deterministic time $O\big(|z| + |w|\big)$. For a definition of the Higman–Thompson groups see also [2], [15] and [9].

**Key selection:** We first pick a word $x \in \{0,1,2\}^*$. For encrypting and decrypting 0 we choose a word $z \in \{0,1,2\}^*$ and, similarly, for 1 we choose a word $u \in \{0,1,2\}^*$; the three words $x$, $z$, $u$ should be long enough so that it is impossible to guess them. For 0, we also choose $m-1$ "intermediary words" $z_i \in \{0,1,2\}^*$ (with $i = 1, \dots, m-1$); similarly, for 1 we choose $m-1$ "intermediary words" $u_i \in \{0,1,2\}^*$ (with $i = 1, \dots, m-1$). Here, $m$ is a security parameter chosen so that $2^m$ or $4^m$ is very large; e.g., we could have $m = 100$ or $m = 200$. The two sets $\{z\} \cup \{z_i : i = 1, \dots, m-1\}$ and $\{u\} \cup \{u_i : i = 1, \dots, m-1\}$ are required to be disjoint.

Next, we choose a "system of words" over $X \cup X^{-1}$ for encrypting a bit 0, and a system of words over $X \cup X^{-1}$ for encrypting a bit 1. A system of words (say for encrypting 0) is a sequence of $m$ finite sets $(Z_1, \dots, Z_m)$. Each set $Z_j$ is a small set of words over $X \cup X^{-1}$ (with, e.g., 4 elements). Each element $w \in Z_j$ has the property that $(z_{j-1})w = z_j$, for $j = 2, \dots, m-1$; also, for each element $w \in Z_1$, $(x)w = z_1$, and for each element $w \in Z_m$, $(z_{m-1})w = z$. For 1 a similar system $(U_1, \dots, U_m)$ of sets of words is chosen with similar properties regarding $x$, $u_j$ $(j = 1, \dots, m-1)$ and $u$. The action diagram below shows the role of the intermediate words $z_i \in \{0,1,2\}^*$ and the action of the words in $Z_j$ on the intermediate words:

$$x \xrightarrow{Z_1} z_1 \xrightarrow{Z_2} z_2 \xrightarrow{Z_3} \dots \xrightarrow{Z_{i-1}} z_{i-1} \xrightarrow{Z_i} z_i \xrightarrow{Z_{i+1}} \dots \xrightarrow{Z_{m-1}} z_{m-1} \xrightarrow{Z_m} z\,.$$

The **private key** is $(x, z, u)$. (The words $z_i$ and $u_i$ are required to remain secret, but are not needed after key selection, i.e., they are not used in encryption or decryption.)

The **public key** consists of the presentation $(X, R)$, as well as the two set systems $(Z_1, \ldots, Z_m)$ (for 0), and $(U_1, \ldots, U_m)$ (for 1).

**Encryption:** To encrypt a bit 0, randomly choose an element $w_j$ in each set $Z_j$ $(j = 1, \ldots, m)$, and concatenate these elements to form the word $w_1 w_2 \ldots w_m$. Next, as in the Wagner–Magyarik system, we rewrite $w_1 w_2 \ldots w_m$ by applying the relators of $G = (X, R)$ (as well as the trivial relators) randomly a "sufficiently large" number of times; see the discussion below concerning this rewriting. This yields some word $W_0$, encrypting 0. To encrypt a bit 1, the procedure is similar, but now the set system $(U_1, \ldots, U_m)$ is used.

**Decryption:** With a ciphertext $w$, compute $(x)w$. If $(x)w = z$, decrypt as a 0; if $(x)w = u$, decrypt as a 1.

**Some design issues:**

1. The words $x, z, u \in \{0, 1, 2\}^*$ are selected uniformly at random among words of length between $n$ and $2n$. Here $n$ is a security parameter; e.g., $n = 100$ or $n = 200$. Similarly, the intermediary words are selected uniformly at random among words of length between $n/2$ and $4n$.
Another security parameter is $m$; e.g., $m = 100$ or $m = 200$.

2. How is the "system of words" $(Z_1, \ldots, Z_m)$ (and similarly $(U_1, \ldots, U_m)$) determined? For each pair of intermediary words $(z_j, z_{j+1})$ (for 0) we design a boolean circuit that maps $z_j$ to $z_{j+1}$; similarly, we design a boolean circuit that maps $u_j$ to $u_{j+1}$. These two circuits should be as similar as possible (in fact, when $z_j \neq u_j$, the same circuit could be used for both; we then can make them different in random details). If we want $Z_{j+1}$ (and $U_{j+1}$) to have 4 elements, we repeat this four times. Next, we use the correspondence between circuits and elements of the Higman-Thompson group $G_{3,1}$ (see [1]) to construct elements of $G$ that simulate these circuits.

3. *Random rewriting:* The rewriting of an element from $Z_1 \times \cdots \times Z_m$ (respectively from $U_1 \times \cdots \times U_m$) could be done as follows. First enlarge the presentation $G = (X, R)$ by including $R^{-1}$ (the set of inverses of the words in $R$) into the set of relators, and by adding all cyclic permutations of words in $R \cup R^{-1}$ as well; this gives us the "symmetrized presentation" $(X, R_s)$ of $G$. Next, we turn $(X, R_s)$ into a string rewriting system by taking all rules of the form $u \to v$ for any (possibly empty) strings $u, v$ over $X \cup X^{-1}$ such that $u^{-1}v$ is a relator in $R_s$. We also add the rules $1 \to a^{-1}a$ and $a^{-1}a \to 1$ for any $a \in X \cup X^{-1}$; here, 1 is the empty string. For rewriting a word $w$ of the length $n$ we do the following:

Procedure A: 1. choose a position in the word obtained so far; 2. choose a rule, and apply it at the chosen position (if the rule doesn't apply at this position, go back to step 1.).

After $n$ repetitions of procedure A we check whether every letter of $w$ has been rewritten (this assumes that we marked the original letters of $w$); if not all letters have been rewritten, run procedure A another $n$ times; keep repeating $n$ runs of procedure A until all letters of $w$ have been rewritten. At this point, most positions of $w$ will have been rewritten many times.

Now we could mark all the letters in the word $w'$ obtained so far, and start over with the rewriting until all positions in $w'$ have been rewritten. All this could be repeated a few more times.

The encryption of 0 first chooses one out of $4^m$ elements from $Z_1 \times \cdots \times Z_m$ (or from $U_1 \times \cdots \times U_m$ for 1). The rewriting process then makes it hard to recognize what system of sets the chosen element $w_1 \dots w_m$ originally came from. The rewrite rules are applied everywhere in the word, so that no local pattern from a set $Z_j$ or $U_j$ ($j = 1, \dots, m$) remains. Because of the exponential number of choices for $w_1 \dots w_m$, the role of the rewriting is less important than in the original Wagner–Magyarik idea. The role of the systems of words $(Z_1, \dots, Z_m)$ and $(U_1, \dots, U_m)$ is precisely to (exponentially) strengthen the confusion caused by rewriting, and this is one of the contributions of our paper. But the rewriting is nevertheless important, and research is needed to determine how (and how much of) the random rewriting should be done.

4. *Security, open problems:* A *spurious key* is any triple $(x', z', u')$ of words over the alphabet $\{0, 1, 2\}$, with such properties that $(x')v = z'$ for any word $v$ that encrypts 0, and $(x')w = u'$ for any word $w$ that encrypts 1. For a known-plaintext or a chosen-ciphertext attack, suppose the attacker has a collection of plaintext-ciphertext pairs $(0, v_i)$, $(1, w_j)$ for $i = 1, \dots, m$, and $n = 1, \dots, n$. Finding (spurious) keys is the search version of the *common action problem* of groups elements, which we conjecture to be NP-hard; see Appendix 2.

Our complexity analysis in this paper refers to the worst case complexity. For security, almost-all case complexity, or at least average case complexity is needed. Unfortunately, almost-all case and average case complexity are still relatively poorly explored, and still have definitional problems.

Other open problems:

- Is the word choice problem of the group $G = \langle G_{3,1}^{\bmod 3}(0, 1; \#) \cup \{\kappa_{321}\}\rangle$ an (NP $\cap$ coNP)-complete premise problem? (Appendix 1 gives a result for semigroups.)
- Is the common action problem of the group $G = \langle G_{3,1}^{\bmod 3}(0, 1; \#) \cup \{\kappa_{321}\}\rangle$ NP-complete? (Appendix 2 gives a result for circuits and a connection with $G$.)

5. *Other groups that could be used in our public-key cryptosystem:*

The Higman–Thompson group $G_{3,1}$ with infinite generating set $\Delta_{3,1} \cup \{\tau_{0,i}\colon i > 0\}$, as studied in [1], could be used. This group has a finite presentation, and over this finite presentation the word problem is easy. However, over the infinite generating set $\Delta_{3,1} \cup \{\tau_{0,i}\colon i > 0\}$ the word problem of $G_{3,1}$ is coNP-hard. This group can be used directly to simulate circuits.

The finite symmetric group $\mathfrak{S}_N$ could be used; here $N = 2^n$, and $n$ is a security parameter, e.g., $n = 100$. Although this group is finite, its size is exponential in the security parameter. It is an open problem whether $\mathfrak{S}_N$ has presentations of size linear in $n$. We think of $\mathfrak{S}_N$ as acting on bit-strings of length $n$, hence it is natural to use elements of $\mathfrak{S}_N$ for representing circuits.

# 4. Appendix

### Appendix 1: (NP$\cap$coNP)-complete premise problems

We obtain an (NP$\cap$coNP)-complete word choice problem for a finitely presented *semigroup*. For groups it is an open problem whether there are (NP$\cap$coNP)-complete word choice problems.

Let $S_{np} = (X, R)$ be a finitely presented *semigroup* with NP-complete word problem, as constructed in [3]; this presentation was derived from any nondeterministic polynomial-time Turing machine that recognizes an NP-complete language.

**Proposition.** *The word choice problem of the finitely presented semigroup $S_{np}$ above is an (NP$\cap$ coNP)-complete premise problem.*

Proof. Let $L$ be any problem in NP$\cap$coNP. Consider a nondeterministic polynomial-time Turing machine that recognizes $L$ and consider also a nondeterministic polynomial-time Turing machine that recognizes the complement $\overline{L}$. Without loss of generality we can assume that these two Turing machines are actually the same Turing machine (let's call it $M$), except for the accept states: $L$ is accepted by $M$ using accept state $q_1$, and $\overline{L}$ is accepted by $M$ using accept state $q_2$. In [3] the acceptance problem "does $M$ accept a word $w$ using accept state $q_i$?" (for $i = 1, 2$) is reduced to the word problem "$F(q_0 w) =_{S_{np}} F(q_i)$?"; here, $q_0$ is the start state of $M$, and $F$ is a linear-time computable function from the words over the symbol set of $M$ to the words over $X$; $F$ is the function that reduces the decision problem of $M$ to the word problem of $S_{np}$. Observe that the same word $F(q_0 w)$ is used for both $L$ and $\overline{L}$. Therefore $w \in L$ iff $F(q_0 w) =_{S_{np}} F(q_1)$, and $w \notin L$ iff $F(q_0 w) =_{S_{np}} F(q_2)$; hence also, $F(q_1) \neq_{S_{np}} F(q_2)$. So, $F$ reduces the language $L$ to the word choice problem of the semigroup $S_{np}$, relative to the two words $F(q_1)$ and $F(q_2)$. $\square$

**Appendix 2: The common action problem**

Let $G$ be a group generated by a finite set $X \subset G$, and acting faithfully (by total or partial injective or bijective maps) on the set $A^*$ (the set of all words over a finite alphabet $A$). The *common action problem* of $G$ (with generating set $X$, acting on $A^*$) is specified as follows:

INPUT: words $w_1, \dots, w_n$ over $X \cup X^{-1}$;

QUESTION: does there exist $(x, y) \in A^* \times A^*$ such that for each $i = 1, \dots, n$: $(x)w_i = y$ ?

The *search* version of this problem consists of outputting any such pair $(x, y)$, rather than just finding out whether there is one.

The *circuit common action problem* is specified as follows:

INPUT: combinational circuits $C_i$ (with I/O function $f_i \colon \{0,1\}^n \to \{0,1\}^n$), for $i = 1, \dots, k$;

QUESTION: is there $(x, y) \in \{0,1\}^n \times \{0,1\}^n$ such that for each $i = 1, \dots, k$: $f_i(x) = y$?

**PROPOSITION.** *The common action problem for combinational circuits is NP-complete.*

Proof. We will reduce the circuit satisfiability problem (which is NP-complete) to the circuit common action problem. In the circuit satisfiability problem the input is a combinational circuit, and the question is whether there is a circuit input $x \in \{0,1\}^n$ for which the circuit produces the all 1s output $1^n$. A circuit $C$ has an input $x$ that produces the output $1^n$ iff the following two circuits $C_1'$, $C_2'$ have a common action pair: $C_1'$ on input $x$ first uses $C$, and then checks whether the output of $C$ (on input $x$) is $1^n$; if it is, $C_1'$ outputs $1^n$, otherwise $C_1'$ outputs $0^n$. The circuit $C_2'$ always outputs $1^n$. So, $x$ is a satisfying input of $C$ iff $(x, 1^n)$ is a common action pair of $C_1'$ and $C_2'$, which is iff $C_1'$ and $C_2'$ have a common action pair at all. □

We would like to reduce the common action problem of circuits to the common action problem of the group $G = \langle G_{3,1}^{\mathrm{mod}\ 3}(0,1;\#) \cup \{\kappa_{321}\} \rangle$ by using methods similar to those of [1]. However, those methods only show that the common action problem of $G$ is NP-complete when we restrict the question to pairs $(x, y)$ with $x \in 0\{0,1\}^* \cup 0\{0,1\}^*2$. It seems likely that the common action problem of $G$ is NP-complete, but this remains a conjecture.

## 5. Conclusion

The general idea for a public-key cryptosystem proposed by Wagner and Magyarik in 1984 is an interesting subject for research. The original idea is too vague to be called a cryptosystem, and it is an interesting challenge to make the idea precise in such a way as to obtain a secure system. Also, the idea needs a better analysis; in particular, it is not based on the word problem (as has been claimed so far) but on the word choice problem, which is a less difficult problem, and which is related to (NP $\cap$ coNP)-completeness of premise problems. It seems possible to construct public-key cryptosystems based on a combination of finite presentations and transformation groups. We describe such a system, based on groups related to the Higman–Thompson groups. The security evaluation of these schemes leads to interesting new complexity problems in combinatorial group theory.

## REFERENCES


[1] BIRGET, J. C.: *Circuits, coNP-completeness, and the groups of Richard Thompson*, Internat. J. Algebra Comput. **16** (2006), 35–90.
Preprint (2003), `http://arXiv.org/abs/math.GR/0310335`.

[2] BIRGET, J. C.: *The groups of Richard Thompson and complexity*, Internat. J. Algebra Comput. **14** (2004), 569–626, `http://arXiv.org/abs/math.GR/0204292`.

[3] BIRGET, J. C.: *Time-complexity of the word problem for semigroups and the Higman Embedding Theorem*, Internat. J. Algebra Comput. **8** (1998), 235–294.

[4] BIRGET, J. C.—OL'SHANSKII, A.—RIPS, E.—SAPIR, M. V.: *Isoperimetric functions of groups and computational complexity of the word problem*, Ann. Math. **156** (2002), 467–518.

[5] DING-ZHU DU, KER-I KO: *Theory of Computational Complexity*, Wiley, New York, 2000.

[6] GONZÁLEZ VASCO, M. I.: *Criptosistemas Basados en Teoría de Grupos*, Tesis Doctoral, Universidad de Oviedo (Julio, 2003).
`http://www.criptored.upm.es/paginas/investigacion.htm`.

[7] GONZÁLEZ VASCO, M. I.—STEINWANDT, R.: *Reaction attacks on public key cryptosystems based on the word problem*, Appl. Algebra Eng. Commun. Comput **14** (2002), 215–226, `http://eprint.iacr.org/2002/139`.

[8] GONZÁLEZ VASCO, M. I.—MARTÍNEZ, C.—STEINWANDT, R.: *Towards a uniform description of several group based cryptographic primitives*, Cryptography ePrint Archive, Report 2002/048, Preprint (2002), `http://eprint.iacr.org/2002.048`.

[9] HIGMAN, G.: *Finitely Presented Infinite Simple Groups*, Notes on Pure Math. 8, I. A. S. Australian National University, Canberra, 1974.

[10] LYNDON, R. C.—SCHUPP, P. E.: *Combinatorial Group Theory*, Ergeb. Math. Grenzgeb., Springer-Verlag, Berlin, 1977.

[11] MAGNUS, W.—KARRASS, A.—SOLITAR, D.: *Combinatorial Group Theory*, Presentations of Groups in Terms of Generators and Relations, Interscience Publishers, a Division of John Wiley and Sons, New York, 1966.

[12] MADLENER, K.—OTTO, F.: *Pseudo-natural algorithms for the word problem for finitely presented monoids and groups*, J. Symbolic Comput. **1** (1985), 383–418.

[13] WAGNER, N. R.—MAGYARIK, M. R.: *A public-key cryptosystem based on the word problem*, in: Proc. Advances in Cryptology, CRYPTO '84, Lecture Notes in Comput. Sci., Vol. 196, Springer-Verlag, Berlin, 1985, pp. 19–36.

[14] SAPIR, M. V.—BIRGET, J. C.—RIPS, E.: *Isoperimetric and isodiametric functions of groups*, Ann. Math. **156** (2002), 345–466.

[15] SCOTT, E. A.: *A construction which can be used to produce finitely presented infinite simple groups*, J. Algebra **90** (1984), 294–322.

[16] WRATHALL, C.: *The word problem for free partially commutative groups*, J. Symbolic Comput. **6** (1988), 99–104.

Received October 1, 2004

*Jean-Camille Birget*
*Department of Computer Science*
*Rutgers University–Camden*
*Camden, NJ 08102*
*U. S. A.*
*E-mail*: birget@camden.rutgers.edu

*Spyros S. Magliveras*
*Michal Sramka*
*Center for Cryptology and Information Security*
*Department of Mathematical Sciences*
*Florida Atlantic University*
*777 Glades Road*
*Boca Raton*
*FL–33431*
*U. S. A.*
*E-mail*: spyros@fau.edu
sramka@math.fau.edu