УДК 510.52

ZERO-KNOWLEDGE PROOFS OF THE CONJUGACY FOR PERMUTATION GROUPS

Oleg VERBITSKY

Kyiv Taras Shevchenko National University,
64 Volodymyrska Str. 01033 Kyiv, Ukraine

We design a perfect zero-knowledge proof system for recognition if two permutation groups are conjugate. It follows, answering a question posed by O. G. Ganyushkin, that this recognition problem is not NP-complete unless the polynomial-time hierarchy collapses.

Key words: interactive proof system, zero-knowledge proof, NP-completeness, permutation group, conjugacy.

1. Let S_m be a symmetric group of order m. We suppose that an element of S_m, a permutation of the set $\{1, 2, \ldots, m\}$, is encoded by a binary string of length $l = \lceil \log_2 m! \rceil$, $m(\log_2 m - O(1)) \leqslant l \leqslant m \log_2 m$. Given $v \in S_m$, $y \in S_m$, and $Y \subseteq S_m$, we denote $y^v = v^{-1}yv$ and $Y^v = \{y^v : y \in Y\}$. Two subgroups G and H of S_m are *similar* if their actions on $\{1, 2, \ldots, m\}$ are isomorphic or, equivalently, if $G = H^v$ for some $v \in S_m$. If $X \subseteq S_m$, let $\langle X \rangle$ denote the group generated by elements of X.

We address the following algorithmic problem.

SIMILITUDE OF PERMUTATION GROUPS
Given: $A_0, A_1 \subseteq S_m$.
Recognize if: A_0 and A_1 are similar.

Note that the EQUALITY OF PERMUTATION GROUPS problem, that is, recognition if $\langle A_0 \rangle = \langle A_1 \rangle$ reduces to recognition, given $X \subseteq S_m$ and $y \in S_m$, if $y \in \langle X \rangle$. Since the latter problem is known to be solvable in time bounded by a polynomial of the input length [20, 10], so is EQUALITY OF PERMUTATION GROUPS. As a consequence, SIMILITUDE OF PERMUTATION GROUPS belongs to NP, the class of decision problems whose yes-instances have polynomial-time verifiable certificates. The similitude of $\langle A_0 \rangle$ and $\langle A_1 \rangle$ is certified by a permutation v such that $\langle A_1 \rangle = \langle A_0^v \rangle$.

Another problem, ISOMORPHISM OF PERMUTATION GROUPS, is to recognize if $\langle A_0 \rangle$ and $\langle A_1 \rangle$ are isomorphic. This problem also belongs to NP (E. Luks, see [5, Corollary 4.11]). Furthermore, it is announced [7] that ISOMORPHISM OF PERMUTATION GROUPS belongs to the complexity class coAM (see Section 2 for the definition). By [8] this implies that ISOMORPHISM OF PERMUTATION GROUPS is not NP-complete unless the polynomial-time hierarchy collapses to its second level (for the background on computational complexity theory the reader is referred to [12]).

O. G. Ganyushkin [11] posed a question if a similar non-completeness result can be obtained for SIMILITUDE OF PERMUTATION GROUPS. In this paper we answer this question in affirmative. We actually prove a stronger result of independent interest, namely, that SIMILITUDE OF PERMUTATION GROUPS has a perfect zero-knowledge interactive proof system. It follows from [1] that SIMILITUDE OF PERMUTATION GROUPS belongs to coAM and is therefore not NP-complete unless the polynomial-time hierarchy collapses.

Informally speaking, a zero-knowledge proof system for a recognition problem of a language L is a protocol for two parties, the prover and the verifier, that allows the prover to convince the verifier that a given input belongs to L, with high confidence but without communicating the verifier any information (the rigorous definitions are in Section 2). Our zero-knowledge proof system for SIMILITUDE OF PERMUTATION GROUPS uses the underlying ideas of the zero-knowledge proof systems designed in [16] for the QUADRATIC RESIDUOSITY and in [14] for the GRAPH ISOMORPHISM problem. In particular, instead of direct proving something about the input groups $\langle A_0 \rangle$ and $\langle A_1 \rangle$, the prover prefers to deal with their conjugates $\langle A_0 \rangle^w$ and $\langle A_1 \rangle^w$ via a random permutation w. The crucial point is that these random groups are indistinguishable by the verifier because they are identically distributed, provided $\langle A_0 \rangle$ and $\langle A_1 \rangle$ are similar. However, we here encounter a complication: the verifier may actually be able to distinguish between $\langle A_0 \rangle^w$ and $\langle A_1 \rangle^w$ based on particular representations of these groups by their generators. Overcoming this complication, which does not arise in [16, 14], is a novel ingredient of our proof system.

Our result holds true even for a more general problem of recognizing if $\langle A_0 \rangle$ and $\langle A_1 \rangle$ are conjugated via an element of the group generated by a given set $U \subseteq S_m$. We furthermore observe that a similar perfect zero-knowledge proof system works also for the ELEMENT CONJUGACY problem of recognizing, given $a_0, a_1 \in S_m$ and $U \subseteq S_m$, if $a_1 = a_0^v$ for some $v \in \langle U \rangle$. A version of this problem where $a_0, a_1 \in \langle U \rangle$ was proved to be in coAM in [5, Corollary 12.3 (i)]. Note that the proof system developed in [5] uses different techniques and is not zero-knowledge.

2. Preliminaries. Every decision problem under consideration can be represented through a suitable encoding as a recognition problem for a language L over the binary alphabet. We denote the *length* of a binary word w by $|w|$.

An *interactive proof system* $\{V, P\}$, further on abbreviated as IPS, consists of two probabilistic Turing machines, a polynomial-time *verifier* V and a computationally unlimited *prover* P. The input tape is common for the verifier and the prover. The verifier and the prover also share a communication tape which allows message exchange between them. The system works as follows. First both the machines V and P are given an input w and each of them is given an individual random string, r_V for V and r_P for P. Then P and V alternatingly write messages to one another in the communication tape. V computes its i-th message a_i to P based on the input w, the random string r_V, and all previous messages from P to V. P computes its i-th message b_i to V based on the input w, the random string r_P, and all previous messages from V to P. After a number of message exchanges V terminates interaction and computes an output based on w, r_V, and all b_i. The output is denoted by $\{V, P\}(w)$. Note that, for a fixed w, $\{V, P\}(w)$ is a random variable depending on both random strings r_V and r_P.

Let $\epsilon(n)$ be a function of a natural argument taking on positive real values. We say

that $\{V, P\}$ is an *IPS for a language L with error $\epsilon(n)$* if the following two conditions are fulfilled.

Completeness. If $w \in L$, then $\{V, P\}(w) = 1$ with probability at least $1 - \epsilon(|w|)$.

Soundness. If $w \notin L$, then, for an arbitrary interacting probabilistic Turing machine P^*, $\{V, P^*\}(w) = 1$ with probability at most $\epsilon(|w|)$.

We will call any prover P^* interacting with P on input $w \notin L$ *cheating*. If in the completeness condition we have $\{V, P\}(w) = 1$ with probability 1, we say that $\{V, P\}$ has *one-sided error $\epsilon(n)$*.

An IPS is *public-coin* if the concatenation $a_1 \ldots a_k$ of the verifier's messages is a prefix of his random string r_V. A *round* is sending one message from the verifier to the prover or from the prover to the verifier. The class AM consists of those languages having IPSs with error $1/3$ and with number of rounds bounded by a constant for all inputs. A language L belongs to the class coAM iff its complement $\{0, 1\}^* \setminus L$ belongs to AM.

2.1. Proposition (Goldwasser-Sipser [17]). *Every IPS for a language L can be converted into a public-coin IPS for L with the same error at cost of increasing the number of rounds in 2.*

Given an IPS $\{V, P\}$ and an input w, let $\mathrm{view}_{V,P}(w) = (r'_V, a_1, b_1, \ldots, a_k, b_k)$ where r'_V is a part of r_V scanned by V during work on w and $a_1, b_1, \ldots, a_k, b_k$ are all messages from P to V and from V to P (a_1 may be empty if the first message is sent by P). Note that the verifier's messages $a_1, \ldots, a_k$ could be excluded because they are efficiently computable from the other components. For a fixed w, $\mathrm{view}_{V,P}(w)$ is a random variable depending on r_V and r_P.

An IPS $\{V, P\}$ is *perfect zero-knowledge on L* if for every interacting polynomial-time probabilistic Turing machine V^* there is a probabilistic Turing machine M_{V^*}, called a *simulator*, that on every input $w \in L$ runs in expected polynomial time and produces output $M_{V^*}(w)$ which, if considered as a random variable depending on a random string of M_{V^*}, is distributed identically with $\mathrm{view}_{V^*,P}(w)$. This notion formalizes the claim that the verifier gets no information during interaction with the prover: everything that the verifier gets he can get without the prover by running the simulator. According to the definition, the verifier learns nothing even if he deviates from the original program and follows an arbitrary probabilistic polynomial-time program V^*. We will call the verifier V *honest* and all other verifiers V^* *cheating*. If, for all V^*, M_{V^*} is implemented by the same simulator M running V^* as a subroutine, we say that $\{V, P\}$ is *black-box simulation* zero-knowledge.

We call $\epsilon(n)$ *negligible* if $\epsilon(n) < n^{-c}$ for every c and all n starting from some $n_0(c)$. The class of languages L having IPSs that are perfect zero-knowledge on L and have negligible error is denoted by PZK.

2.2. Proposition (Aiello-Håstad [1]). $PZK \subseteq coAM$.

The $k(n)$-*fold sequential composition* of an IPS $\{V, P\}$ is the IPS $\{V', P'\}$ in which V' and P' on input w execute the programs of V and P sequentially $k(|w|)$ times, each time with independent choice of random strings r_V and r_P. At the end of interaction V' outputs 1 iff $\{V, P\}(w) = 1$ in all $k(|w|)$ executions. The initial system $\{V, P\}$ is called *atomic*.

2.3. Proposition.

1)If $\{V', P'\}$ is the $k(n)$-fold sequential composition of $\{V, P\}$, then

$$\max_{P^*} \mathbf{P}\left[\{V', P^*\}(w) = 1\right] = \left(\max_{P^*} \mathbf{P}\left[\{V, P^*\}(w) = 1\right]\right)^{k(|w|)}.$$

Consequently, if $\{V, P\}$ is an IPS for a language L with one-sided constant error ϵ, then $\{V', P'\}$ is an IPS for L with one-sided error $\epsilon^{k(n)}$.

2) (Goldreich-Oren [15], see also [13, Lemma 6.19]) If in addition $\{V, P\}$ is black-box simulation perfect zero-knowledge on L, then $\{V', P'\}$ is perfect zero-knowledge on L.

In the $k(n)$-fold parallel composition $\{V'', P''\}$ of $\{V, P\}$, the program of $\{V, P\}$ is executed $k(|w|)$ times in parallel, that is, in each round all $k(|w|)$ versions of a message are sent from one machine to another at once as a long single message. In every parallel execution V'' and P'' use independent copies of r_V and r_P. At the end of interaction V'' outputs 1 iff $\{V, P\}(w) = 1$ in all $k(|w|)$ executions.

2.4. Proposition. *If $\{V'', P''\}$ is the $k(n)$-fold parallel composition of $\{V, P\}$, then*

$$\max_{P^*} \mathbf{P}\left[\{V'', P^*\}(w) = 1\right] = \left(\max_{P^*} \mathbf{P}\left[\{V, P^*\}(w) = 1\right]\right)^{k(|w|)}.$$

3. Group Conjugacy.

We consider the following extension of SIMILITUDE OF PERMUTATION GROUPS.

GROUP CONJUGACY
Given: $A_0, A_1, U \subseteq S_m$.
Recognize if: $\langle A_1 \rangle = \langle A_0 \rangle^v$ for some $v \in \langle U \rangle$.

3.1. Theorem. GROUP CONJUGACY *is in PZK.*

Designing a perfect zero-knowledge interactive proof system for GROUP CONJUGACY, we will make use of the following facts due to Sims [20,10].

1) There is a polynomial-time algorithm that, given $X \subseteq S_m$ and $y \in S_m$, recognizes if $y \in \langle X \rangle$. As a consequence, there is a polynomial-time algorithm that, given $X \subseteq S_m$ and $Y \subseteq S_m$, recognizes if $\langle X \rangle = \langle Y \rangle$.

2) There is a probabilistic polynomial-time algorithm that, given $X \subseteq S_m$, outputs a random element of $\langle X \rangle$. Here and further on, by a *random element* of a finite set Z we mean a random variable uniformly distributed over Z.

Given $A \subseteq S_m$ and a number k, define

$$G(A, k) = \left\{ (x_1, \ldots, x_k) : x_i \in S_m, \langle x_1, \ldots, x_k \rangle = \langle A \rangle \right\}.$$

In the sequel, the length of the binary encoding of an input $A_0, A_1, U \subseteq S_m$ will be denoted by n. We set $k = 4m$. On input (A_0, A_1, U), the IPS we design is the n-fold sequential repetition of the following 3-round system. We will say that the verifier V *accepts* if $\{V, P\}(A_0, A_1, U) = 1$ and *rejects* otherwise.

If (A_0, A_1, U) is yes-instance of GROUP CONJUGACY, P finds an element $v \in \langle U \rangle$ such that $\langle A_1 \rangle = \langle A_0 \rangle^v$.

1st round.

P generates a random element $u \in \langle U \rangle$, computes $A = A_1^u$, chooses a random element $(a_1, \ldots, a_k)$ in $G(A, k)$, and sends $(a_1, \ldots, a_k)$ to V. V checks if all $a_i \in S_m$ and, if not (this is possible in the case of a cheating prover), halts and rejects.

2nd round.
V chooses a random bit $\beta \in \{0, 1\}$ and sends it to P.

3rd round.
Case $\beta = 1$. P sends V the permutation $w = u$. V checks if $w \in \langle U \rangle$ and if $\langle a_1, \ldots, a_k \rangle = \langle A_1^w \rangle$.
Case $\beta \neq 1$ (this includes the possibility of a message $\beta \notin \{0, 1\}$ produced by a cheating verifier). P computes $w = vu$ and sends w to V. V checks if $w \in \langle U \rangle$ and if $\langle a_1, \ldots, a_k \rangle = \langle A_0^w \rangle$.

V halts and accepts if the conditions are checked successfully and rejects otherwise.

We now need to prove that this system is indeed an IPS for GROUP CONJUGACY and, moreover, that it is perfect zero-knowledge.

Completeness. To show that the prover is able to follow the above protocol, we have to check that $G(A, k) \neq \emptyset$ for $k = 4m$. The latter is true by the fact that every subgroup of S_m can be generated by at most $m - 1$ elements [18]. If $\langle A_0 \rangle$ and $\langle A_1 \rangle$ are conjugate via an element of $\langle U \rangle$ and the prover and the verifier follow the protocol, then $\langle a_1, \ldots, a_k \rangle = \langle A \rangle = \langle A_1^u \rangle = \langle A_0^{vu} \rangle$. Therefore, the verifier accepts with probability 1 both in the atomic and the composed systems.

Soundness. Assume that $\langle A_0 \rangle$ and $\langle A_1 \rangle$ are not conjugate via an element of $\langle U \rangle$ and consider an arbitrary cheating prover P^*. Observe that if both $\langle a_1, \ldots, a_k \rangle = \langle A_1^u \rangle$ and $\langle a_1, \ldots, a_k \rangle = \langle A_0^w \rangle$ with $u, w \in \langle U \rangle$, then $\langle A_1 \rangle = \langle A_0 \rangle^{wu^{-1}}$. It follows that V rejects for at least one value of β and, therefore, in the atomic system V accepts with probability at most $1/2$. By Proposition 2.3 (1), in the composed system V accepts with probability at most 2^{-n}.

Zero-knowledge. We will need the following fact.

3.2. Lemma. *Let G be a subgroup of S_m and $a_1, \ldots, a_k$ be random independent elements of G.*
1) If $k = 4m$, then $\langle a_1, \ldots, a_k \rangle = G$ with probability more than $1/2$.
2) If $k = 8m$, then $\langle a_1, \ldots, a_k \rangle = G$ with probability more than $1 - 2^{-m}$.

Proof. We will estimate from above the probability that $\langle a_1, \ldots, a_k \rangle \neq G$. This inequality is equivalent with the condition that all $\langle a_1 \rangle$, $\langle a_1, a_2 \rangle$, $\ldots$, $\langle a_1, \ldots, a_k \rangle$ are proper subgroups of G. Assume that this condition is true. Since every subgroup chain in S_m has length less than $2m$ (see [3, 9]), less than $2m - 1$ inclusions among $\langle a_1 \rangle \subseteq \langle a_1, a_2 \rangle \subseteq \cdots \subseteq \langle a_1, \ldots, a_k \rangle$ are proper. In other words, less than $2m - 1$ of the events $a_2 \notin \langle a_1 \rangle$, $a_3 \notin \langle a_1, a_2 \rangle$, $\ldots$, $a_k \notin \langle a_1, \ldots, a_{k-1} \rangle$ occur. Equivalently, there occur more than $k - 2m$ of the events $a_2 \in \langle a_1 \rangle$, $a_3 \in \langle a_1, a_2 \rangle$, $\ldots$, $a_k \in \langle a_1, \ldots, a_{k-1} \rangle$.

Let $p = |H|/|G|$ be the maximum density of a proper subgroup H of G. Given $a_1, \ldots, a_i \in G$, define $E(a_1, \ldots, a_i)$ to be an arbitrary subset of G fixed so that

    (i) $E(a_1, \ldots, a_i)$ has density p in G, and
    (ii) $E(a_1, \ldots, a_i)$ contains $\langle a_1, \ldots, a_i \rangle$ if the latter is a proper subgroup of G.

If $\langle a_1, \ldots, a_k \rangle \neq G$, there must occur more than $k - 2m$ of the events

$$a_2 \in E(a_1), \ a_3 \in E(a_1, a_2), \ldots, a_k \in E(a_1, \ldots, a_{k-1}). \qquad (1)$$

It suffices to show that the probability of so many occurrences in (1) is small enough. Set $X_i(a_1, \ldots, a_k)$ to be equal to 1 if $a_{i+1} \in E(a_1, \ldots, a_i)$ and to 0 otherwise. In these terms, we have to estimate the probability that

$$\sum_{i=1}^{k-1} X_i > k - 2m. \qquad (2)$$

It is easy to calculate that an arbitrary set of l events in (1) occurs with probability p^l. Hence events (1) as well as the random variables $X_1, \ldots, X_{k-1}$ are mutually independent, and $X_1, \ldots, X_{k-1}$ are successive Bernoulli trails with success probability p.

If $k = 4m$, inequality (2) implies that strictly more than a half of all the trails are successful. Since $p \leqslant 1/2$, this happens with probability less than $1/2$ and item 1 of the lemma follows.

If $k = 8m$, inequality (2) implies

$$\frac{1}{k-1} \sum_{i=1}^{k-1} X_i > p + \epsilon$$

with deviation $\epsilon = 1/4$ from the mean value $p = \mathbf{E}\left[\frac{1}{k-1} \sum_{i=1}^{k-1} X_i\right]$. By the Chernoff bound [2, Theorem A.4], this happens with probability less than $\exp\left(-2\epsilon^2(k-1)\right) = \exp(-m + \frac{1}{8}) < 2^{-m}$. This proves item 2 of the lemma. $\square$

By Proposition 2.3 (2) it suffices to show that the atomic system is black-box simulation perfect zero-knowledge. We describe a probabilistic simulator M that uses the program of V^* as a subroutine and, for each V^*, runs in expected polynomial time. Assume that the running time of V^* is bounded by a polynomial q in the input size. On input (A_0, A_1, U) of length n, M will run the program of V^* on the same input with random string r, where r is the prefix of M's random string of length $q(n)$. In all other cases of randomization, M will use the remaining part of its random string.

Having received an input (A_0, A_1, U), the simulator M chooses a random element $w \in \langle U \rangle$ and a random bit $\alpha \in \{0, 1\}$. Then M randomly and independently chooses elements $a_1, \ldots, a_k$ in $\langle A_\alpha^w \rangle$ and checks if

$$\langle a_1, \ldots, a_k \rangle = \langle A_\alpha^w \rangle. \qquad (3)$$

If (3) is not true, M repeats the choice of $a_1, \ldots, a_k$ again and again until (3) is fulfilled. By Lemma 3.2 (1), M succeeds in at most 2 attempts on average. The resulting sequence $(a_1, \ldots, a_k)$ is uniformly distributed on $G(A_\alpha^w, k)$. Then M computes $\beta = V^*(A_0, A_1, U, r, a_1, \ldots, a_k)$, the message that V^* sends P in the 2-nd round after receiving P's message $a_1, \ldots, a_k$. If β and α are simultaneously equal to or different from 1, M halts and outputs $(r', a_1, \ldots, a_k, \beta, w)$, where r' is the prefix of r that V^* actually uses after reading the input (A_0, A_1, U) and the prover's message $a_1, \ldots, a_k$. If exactly one of β and α is equal to 1, then M restarts the same program from the

very beginning with another independent choice of w, α, and $a_1, \ldots, a_k$. Notice that it might happen that in unsuccessful attempts V^* used a prefix of r longer than r'.

We first check that, for each V^*, the simulator M terminates in expected polynomial time whenever A_0 and A_1 are conjugated via an element of $\langle U \rangle$. Since V^* is polynomial-time, one attempt to pass the body of M's program takes time bounded by a polynomial of n. Observe that α and $(r, a_1, \ldots, a_k)$ are independent. Really, independently of whether $\alpha = 0$ or $\alpha = 1$, r is a random string of length $q(n)$ and $(a_1, \ldots, a_k)$ is a random element of $G(A, k)$, where A itself is a random element of the orbit $\{A_0^w : w \in \langle U \rangle\} = \{A_1^w : w \in \langle U \rangle\}$ under the conjugating action of $\langle U \rangle$ on subsets of S_m. It follows that α and β are independent and therefore an execution of the body of M's program is successful with probability $1/2$. We conclude that on average M's program is executed twice and this takes expected polynomial time.

We finally need to check that, whenever A_0 and A_1 are conjugated via an element of $\langle U \rangle$, for each V^* the output $M(A_0, A_1, U)$ is distributed identically with $\text{view}_{V^*, P}(A_0, A_1, U)$. Notice that both the random variables depend on V^*'s random string r. It therefore suffices to show that the distributions are identical when conditioned on an arbitrary fixed r. Denote these conditional distributions by $D_M(A_0, A_1, U, r)$ and $D_{V^*, P}(A_0, A_1, U, r)$. We will show that they are both uniform on the set

$$S = \Big\{ (a_1, \ldots, a_k, \beta, w) \; : \; w \in \langle U \rangle, \; \beta = V^*(A_0, A_1, U, r, a_1, \ldots, a_k), $$
$$(a_1, \ldots, a_k) \in G(A_{\delta(\beta)}^w, k) \Big\},$$

where $\delta(\beta)$ is equal to 1 if $\beta = 1$ and to 0 otherwise.

Let $v \in \langle U \rangle$, such that $\langle A_1 \rangle = \langle A_0 \rangle^v$, be chosen by the prover P on input (A_0, A_1, U). Given $x_1, \ldots, x_k \in G(A_1, k)$ and $u \in \langle U \rangle$, define $\phi(x_1, \ldots, x_k, u) = (a_1, \ldots, a_k, \beta, w)$ by $a_i = x_i^u$ for all $i \leqslant k$, $\beta = V^*(A_0, A_1, U, r, a_1, \ldots, a_k)$, and $w = v^{1-\delta(\beta)}u$. As easily seen, $\phi(x_1, \ldots, x_k, u) \in S$.

Claim. The map $\phi : G(A_1, k) \times \langle U \rangle \to S$ is one-to-one.

Proof. Define $\psi(a_1, \ldots, a_k, \beta, w) = (x_1, \ldots, x_k, u)$ by $u = v^{\delta(\beta)-1}w$ and $x_i = a_i^{u^{-1}}$ for all $i \leqslant k$. It is not hard to check that the map ψ is the inverse of ϕ. $\square$

Observe now that if $(x_1, \ldots, x_k, u)$ is chosen at random uniformly in $G(A_1, k) \times \langle U \rangle$, then $\phi(x_1, \ldots, x_k, u)$ has distribution $D_{V^*, P}(A_0, A_1, U, r)$. By Claim we conclude that $D_{V^*, P}(A_0, A_1, U, r)$ is uniform on S.

As a yet another consequence of Claim, observe that if a random tuple $(a_1, \ldots, a_k, \beta, w)$ is uniformly distributed on S, then its prefix $(a_1, \ldots, a_k)$ is a random element of $G(A, k)$, where A is a random element of the orbit $\{A_0^w : w \in \langle U \rangle\} = \{A_1^w : w \in \langle U \rangle\}$ under the conjugating action of $\langle U \rangle$ on subsets of S_m. This suggests the following way of generating a random element of S. Choose uniformly at random $\alpha \in \{0, 1\}$, $w \in \langle U \rangle$, $(a_1, \ldots, a_k) \in G(A_\alpha^w, k)$ and, if

$$\delta\left(V^*(A_0, A_1, U, r, a_1, \ldots, a_k)\right) = \alpha, \tag{4}$$

output $(a_1, \ldots, a_k, V^*(A_0, A_1, U, r, a_1, \ldots, a_k), w)$; otherwise repeat the same procedure once again independently. Under the condition that (4) is fulfilled for the first time in the i-th repetition, the output is uniformly distributed on S. Notice now that

this sampling procedure coincides with the description of $D_M(A_0, A_1, U, r)$. It follows that $D_M(A_0, A_1, U, r)$ is uniform on S. The proof of the perfect zero-knowledge property of our proof system for GROUP CONJUGACY is complete.

The following corollary immediately follows from Theorem 3.1 by Proposition 2.2 and the result of [8].

3.3. Corollary. GROUP CONJUGACY *is in coAM and is therefore not NP-complete unless the polynomial-time hierarchy collapses.*

We also give an alternative proof of this corollary that consists in direct designing a two-round IPS $\{V, P\}$ with error $1/4$ for the complement of GROUP CONJUGACY and applying Proposition 2.1. More precisely, we deal with the GROUP NON-CONJUGACY problem of recognizing, given $A_0, A_1, U \subseteq S_m$, if there is no $v \in \langle U \rangle$ such that $\langle A_1 \rangle = \langle A_0 \rangle^v$.

Set $k = 8m$. The below IPS is composed twice in parallel.

1st round.
V chooses a random bit $\alpha \in \{0, 1\}$, a random element $u \in \langle U \rangle$, and a sequence of random independent elements $a_1, \ldots, a_k \in \langle A_\alpha^u \rangle$. Then V sends $(a_1, \ldots, a_k)$ to P.

2nd round.
P determines β such that $\langle a_1, \ldots, a_k \rangle$ and $\langle A_\beta \rangle$ are conjugate via an element of $\langle U \rangle$ and sends β to V.

V accepts if $\beta = \alpha$ and rejects otherwise.

Completeness. By Lemma 3.2 (2), $\langle a_1, \ldots, a_k \rangle = \langle A_\alpha^u \rangle$ with probability at least $1 - 2^{-m}$. If this happens and if $\langle A_0 \rangle$ and $\langle A_1 \rangle$ are not conjugated via $\langle U \rangle$, the group $\langle a_1, \ldots, a_k \rangle$ is conjugated via $\langle U \rangle$ with precisely one of $\langle A_0 \rangle$ and $\langle A_1 \rangle$. In this case P is able to determine α correctly. Therefore V accepts with probability at least $1 - 2^{-m}$ in the atomic system and with probability at least $1 - 2^{-m+1}$ in the composed system.

Soundness. If $\langle A_0 \rangle$ and $\langle A_1 \rangle$ are conjugated via $\langle U \rangle$, then for both values $\alpha = 0$ and $\alpha = 1$, the vector $(a_1, \ldots, a_k)$ has the same distribution, namely, it is a random element of A^k, where A is a random element of the orbit $\{ A_0^w : w \in \langle U \rangle \} = \{ A_1^w : w \in \langle U \rangle \}$ under the conjugating action of $\langle U \rangle$ on subsets of S_m. It follows that, irrespective of his program, P guesses the true value of α with probability $1/2$. With the same probability V accepts in the atomic system. By Proposition 2.4, in the composed system V accepts with probability $1/4$.

Note that $\{V, P\}$ is perfect zero-knowledge only for the honest verifier but may reveal a non-trivial information for a cheating verifier.

4. Element Conjugacy. This section is devoted to the following problem.

ELEMENT CONJUGACY
Given: $a_0, a_1 \in S_m$, $U \subseteq S_m$.
Recognize if: $a_1 = a_0^v$ for some $v \in \langle U \rangle$.

L. Babai [5] considers a version of this problem with $a_0, a_1 \in \langle U \rangle$ and proves that it belongs to coAM. His result holds true not only for permutation groups but also for arbitrary finite groups with efficiently performable group operations, in particular, for matrix groups over finite fields. It is easy to see that Theorem 3.1 carries over to ELEMENT CONJUGACY.

4.1. Theorem. ELEMENT CONJUGACY *is in PZK.*

The proof system designed in the preceding section for GROUP CONJUGACY applies to ELEMENT CONJUGACY as well. Moreover, the proof system for ELEMENT CONJUGACY is considerably simpler. In place of groups $\langle A_0^u \rangle$ and $\langle A_1^u \rangle$ we now deal with single elements a_0^u and a_1^u and there is no complication with representation of $\langle A_0^u \rangle$ and $\langle A_1^u \rangle$ by generating sets.

We now notice relations of ELEMENT CONJUGACY with the following problem considered by E. Luks [19] (see also [6, Section 6.5]). Given $x \in S_m$, let $C(x)$ denote the centralizer of x in S_m.

CENTRALIZER AND COSET INTERSECTION
Given: $x, y \in S_m$, $U \subseteq S_m$.
Recognize if: $C(x) \cap \langle U \rangle y \neq \emptyset$.

Since, given a permutation x, one can efficiently find a list of generators for $C(x)$, this is a particular case of the COSET INTERSECTION problem of recognizing, given $A, B \subseteq S_m$ and $s, t \in S_m$, if the cosets $\langle A \rangle s$ and $\langle B \rangle t$ intersect.

4.2. Proposition. ELEMENT CONJUGACY *and* CENTRALIZER AND COSET INTERSECTION *are equivalent with respect to the polynomial-time many-one reducibility.*

Proof. We first reduce ELEMENT CONJUGACY to CENTRALIZER AND COSET INTERSECTION. Given permutations a_0 and a_1, it is easy to recognize if they are conjugate in S_m and, if so, to find an s such that $a_1 = a_0^s$. The set of all $z \in S_m$ such that $a_1 = a_0^z$ is the coset $C(a_0)s$. It follows that $\langle U \rangle$ contains v such that $a_1 = a_0^v$ iff $C(a_0)$ and $\langle U \rangle s^{-1}$ intersect.

A reduction from CENTRALIZER AND COSET INTERSECTION to ELEMENT CONJUGACY is based on the fact that $C(x)$ and $\langle U \rangle y$ intersect iff x and yxy^{-1} are conjugated via an element of $\langle U \rangle$. $\square$

Note that, while the reduction we described from ELEMENT CONJUGACY to CENTRALIZER AND COSET INTERSECTION works only for permutation groups, the reduction in the other direction works equally well for arbitrary finite groups with efficiently performable group operations, in particular, for matrix groups over finite fields.

We now have three different ways to prove that ELEMENT CONJUGACY is in coAM and is therefore not NP-complete unless the polynomial-time hierarchy collapses. First, this fact follows from Theorem 4.1 by Proposition 2.2. Second, one can use Proposition 4.2 and the result of [5, Corollary 12.2 (d)] that COSET INTERSECTION is in coAM. Finally, one can design a constant-round IPS for the complement of ELEMENT CONJUGACY as it is done in the preceding section for the complement of GROUP CONJUGACY.

We conclude with two questions.

4.3. Question. Is there any reduction between GROUP CONJUGACY and COSET INTERSECTION? We are not able to prove an analog of Proposition 4.2 for groups because, given $A_0, A_1 \subseteq S_m$ such that $\langle A_1 \rangle = \langle A_0 \rangle^v$ for some $v \in S_m$, we cannot efficiently find any v with this property (otherwise we could efficiently recognize the SIMILITUDE OF PERMUTATION GROUPS).

4.4. Question. Does ELEMENT CONJUGACY reduce to GROUP CONJUGACY? Whereas Corollary 3.3 gives us an evidence that GROUP CONJUGACY is not NP-complete, we have no formal evidence supporting our feeling that GROUP CONJUGACY is not solvable

efficiently. A reduction from ELEMENT CONJUGACY could be considered such an evidence as ELEMENT CONJUGACY is not expected to be solvable in polynomial time [4, page 1483].

Note that the conjugacy of permutations a_0 and a_1 via an element of a group $\langle U \rangle$ does not reduce to the conjugacy of the cyclic groups $\langle a_0 \rangle$ and $\langle a_1 \rangle$ via $\langle U \rangle$ because $\langle a_0 \rangle$ and $\langle a_1 \rangle$ can be conjugated by conjugation of another pair of their generators, while such a new conjugation may be not necessary via $\langle U \rangle$. For example, despite the groups $\langle (123) \rangle$ and $\langle (456) \rangle$ are conjugated via $\langle (14)(26)(35) \rangle$, the permutations (123) and (456) are not.

Acknowledgement. I appreciate useful discussions with O. G. Ganyushkin.

1. *Aiello B., Håstad J.* Perfect zero-knowledge languages can be recognized in two rounds // Proc. of the 28-th IEEE Ann. Symp. on Foundations of Computer Science (FOCS). – 1987. – P. 439-448.

2. *Alon A., Spencer J. H.* The probabilistic method. – John Wiley & Sons, 1992.

3. *Babai L.* On the length of chains of subgroups in the symmetric group // Comm. Algebra. – 1986. – Vol. 14. – P. 1729-1736.

4. *Babai L.* Computational complexity in finite groups // Proc. of the Int. Congr. of Mathematicians. Japan, 1990. – P. 1479-1489.

5. *Babai L.* Bounded round interactive proofs in finite groups // SIAM Journal of Discrete Mathematics. – 1992. – Vol. 5. – № 1. – P. 88-111.

6. *Babai L.* Automorphism groups, isomorphism, reconstruction // Handbook of Combinatorics, Ch. 27. – Elsevier Publ. – 1995. – P. 1447-1540.

7. *Babai L., Kannan S., Luks E. M.* Bounded round interactive proofs for nonisomorphism of permutation groups // Quoted in [6] and [5].

8. *Boppana R. B., Håstad J., Zachos S.* Does co-NP have short interactive proofs? // Information Processing Letters. – 1987. – Vol. 25. – P. 127-132.

9. *Cameron P. J., Solomon R., Turull A.* Chains of subgroups in symmetric groups // J. Algebra. – 1989. – Vol. 127. – P. 340-352.

10. *Furst M. L., Hopcroft J., Luks E. M.* Polynomial-time algorithms for permutation groups // Proc. of the 21-st IEEE Ann. Symp. on Foundations of Computer Science (FOCS). – 1980. – P. 36-41.

11. *Ganyushkin O. G.* Personal communication.

12. *Garey M. R., Johnson D. S.* Computers and Intractability. A guide to the theory of NP-completeness. – W. H. Freeman. – 1979 (a Russian translation available).

13. *Goldreich O. Foundations of cryptography (fragments of a book).* – Weizmann Institute of Science. – 1995 (available from **www.eccc.uni-trier.de/eccc/**).

14. *Goldreich O., Micali S., Wigderson A.* Proofs that yield nothing but their validity or all languages in NP have zero-knowledge proof systems // J. Assoc. Comput. Math. – 1991. – Vol. 38. – № 3. – P. 691-729.

15. *Goldreich O., Oren Y.* Definitions and properties of zero-knowledge proof systems // Journal of Cryptology. – 1994. – Vol. 7. – № 1. – P. 1-32.

16. *Goldwasser S., Micali S., Rackoff C.* The knowledge complexity of interactive proof systems // SIAM Journal on Computing. – 1989. – Vol. 18. – № 1. – P. 186-208.

17. *Goldwasser S., Sipser M.* Private coins versus public coins in interactive proof systems // Proc. of the 18-th ACM Ann. Symp. on the Theory of Computing (STOC). – 1986. – P. 59-68.

18. *Jerrum M. R.* A compact representation for permutation groups // Proc. of the 23-rd IEEE Ann. Symp. on Foundations of Computer Science (FOCS). – 1982. – P. 126-133.

19. *Luks E. M.* Isomorphism of graphs of bounded valence can be tested in polynomial time // Journal of Computer and System Sciences. – 1982. – Vol. 25. – P. 42-65.

20. *Sims C. C.* Some group theoretic algorithms. – Vol. 697 of *Lecture Notes in Computer Science*. – Springer Verlag, Berlin. – 1978. – P. 108-124.

ДОВЕДЕННЯ БЕЗ РОЗГОЛОШЕННЯ ДЛЯ СПРЯЖЕНОСТІ ГРУП ПІДСТАНОВОК

О. Вербіцький

Київський національний університет імені Тараса Шевченка,
вул. Володимирська, 64 01033 Київ, Україна

Описано досконалу систему доведення без розголошення для задачі розпізнавання спряженості двох груп підстановок. Звідси випливає, відповідаючи на запитання О. Г. Ганюшкіна, що ця задача розпізнавання не є NP-повною за умови невиродженості поліноміальної ієрархії.

Ключові слова: подібність груп перестановок, алгоритмічні задачі розпізнавання, NP-повнота, системи доведень без розголошення.