 doi:10.20944/preprints201805.0126.v1

Article

A Zero-Knowledge Proof Based on a Multivariate Polynomial Reduction of the Graph Isomorphism Problem

Edgar González Fernández [1,3,†,*], Guillermo Morales-Luna [1,†] and Feliú Sagols Troncoso [2,†]

[1] Department of Computer Science, CINVESTAV-IPN, Av. IPN 2508, Gustavo A. Madero, San Pedro Zacatenco, 07360 Mexico City, Mexico; egonzalez@computacion.cs.cinvestav.mx (E.G.F.), gmorales@cs.cinvestav.mx (G.M.L.)
[2] Department of Mathematics, CINVESTAV-IPN, Av. IPN 2508, Gustavo A. Madero, San Pedro Zacatenco, 07360 Mexico City, Mexico; fsagols@math.cinvestav.edu.mx
[3] Group of Analysis, Security and Systems (GASS), Department of Software Engineering and Artificial Intelligence (DISIA), Faculty of Computer Science and Engineering, Universidad Complutense de Madrid (UCM), Calle Profesor José García Santesmases, 9, Ciudad Universitaria, Madrid 28040, Spain; edggonza@ucm.es
* Correspondence: edggonza@ucm.es; Tel.: +34 913 947 649
† The authors contributed equally to this work.

Abstract: Zero-Knowledge Proofs ZKP provide a reliable option to verify that a claim is true without giving detailed information other than the answer. A classical example is provided by the ZKP based in the Graph Isomorphism problem (GI), where a prover must convince the verifier that he knows an isomorphism between two isomorphic graphs without publishing the bijection. We design a novel ZKP exploiting the NP-hard problem of finding the algebraic ideal of a multivariate polynomial set, and consequently resistant to quantum computer attacks. Since this polynomial set is obtained considering instances of GI, we guarantee that the protocol is at least as secure as the GI based protocol.

Keywords: Graph Isomorphism Problem, Multivariate Polynomial System, Zero-Knowledge Proof

1. Introduction

First presented in [1] by Goldwasser, Micali and Rackoff, interactive proof systems are introduced as a method that allows an entity (the prover) to prove the truth of a proposition to a second party (the verifier) without releasing additional information. The parties involved interact in a challenge-response process until the verifier is convinced that the prover's claim is correct, or concludes that the claim is false. This kind of proofs are commonly used in authentication and identification systems, allowing an entity to prove ownership of a valid credential (ie, credit card number or password) without transmitting or storing this information.

As for now, many of the authentication schemes used in the industry make use of protocols based on PKI by means of digital certificates. A vast majority of these schemes are based on either the factorization problem (RSA) or the DLP, both susceptible to quantum computer attacks. To address this issue, we propose a ZKP whose security relies on $\mathcal{MQ}$, known to be NP-hard, and GI, both resistant against quantum computer attacks up to now.

Recently, suitable instances of $\mathcal{MQ}$ have been used for proposing novel PKC schemes, since they are considered resistant to quantum computers attacks [2], a feature that popular cryptographic algorithms, such as RSA, DSA and ECDSA, do not share. Frequently, algorithms for key generation in MPKC involve two major phases:

- Private key generation. The private key consists of a set of polynomials $F = \{f_1, \ldots, f_m\}$ such that the problem of finding a common root is easy.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 8 May 2018 doi:10.20944/preprints201805.0126.v1

- Public key derivation. Starting with the private key F we create another set of polynomials $\overline{F} = \{\overline{f}_1, \ldots, \overline{f}_n\}$. Finding a common root of the set $\overline{F}$ must be a computationally difficult task since this set must be publicly exhibited without weakening the cryptosystem.

Public key is usually derived from the private key by performing compositions with affine bijective transformations, say S_1, S_2, by performing $\overline{F} = S_2 \circ F \circ S_1$. Consequently, S_1 and S_2 are also kept secret, since they can be easily inverted, and are considered part of the private key. Many other methods for public key generation are explained with detail in [3, 1.2].

One of the first attempts to exploit multivariate polynomials in cryptography can be found in [4,5], where a cipher system, known as the *Matsumoto-Imai cryptosystem*, is proposed but unfortunately broken shortly after being published [6]. However, this effort set the basis for a number of other families of cryptographic schemes, such as the *Unbalanced Oil-Vinegar* (UOV) [7], the Faugere's *Hidden Field Equations* (HFE) [8] and the *Rainbow* signature schemes [9]. A list of the most promising post-quantum cryptographic algorithms can be found in [10].

We may distinguish cryptanalytic attacks on multivariate schemes (and in public cryptosystems in general) according to two main purposes:

- Attacks on ciphertext, where the primary goal is to get the plaintext from the ciphertext. These attacks make use of polynomial system solvers such as the *Buchberger Algorithm* [11] to compute Groebner bases. The algorithm must be executed each time a ciphertext is gathered.
- Attacks to recover the private key, consisting of the private set F and the affine transformations S_1, S_2. Example of this algorithms are: *High Rank*, *MinRank* and Separation of Oil and Vinegar [12](see Section [VI.5.4]).

The method we define in this work produces key pairs from an associated isomorphism between a pair of graphs. The public key will consist of a system of polynomial equations. The private key will consist of a solution to this system. We will show that finding this solution is at least as difficult as finding an isomorphism between the associated graphs. At present, the fastest algorithm for solving the GI problem runs in quasi-polynomial time [13], but an authentic prover will be able to provide a solution efficiently.

The general layout of this paper is as follows. In Section 2 some basic concepts as well as notation necessary for the development of the zero-knowledge proof is introduced. Next, Section 3 is appointed to the construction of the polynomial sets arising from the GI problem as a reduction exercise. The construction of the ZKP will be explained in 4. Finally, in Section 5 we exhibit evidence supporting the viability of the algorithm by estimating the theoretic complexity of the polynomial set construction.

2. Mathematical Background

We recall the basic concepts needed to develop the translation from instances of the GI problem in instances of the $\mathcal{MQ}$ problem.

2.1. Graphs

A *graph* consists of a set $V = \{v_1, \ldots, v_n\}$, the *vertices* and a subset E of $V^{(2)} = \{e \subset V | \, \#e = 2\}$, the *edges*. The number of elements of V and E are known as the *order* and the *size* of G respectively. We say that two vertices u_1, u_2 with $u_1 \neq u_2$, are *adjacent* if they are joined by an edge. Similarly, two different edges $e_1, e_2 \in E$ are said to be *adjacent* if they are joined by a vertex. The *complementary graph* $\overline{G}$ of G is defined as $\overline{G} = (V', E')$ with $V' = V$ and $E' = \{v_i v_j \in V^{(2)} | v_i v_j \notin E\}$.

Whenever there exist two disjoint subsets V_1, V_2 of V such that $V_1 \bigcup V_2 = V$ and every edge has vertices in both sets V_1 and V_2 the graph is called *bipartite*. Additionally, we say G is *complete bipartite* provided that every vertex in V_1 is connected to every vertex in V_2 and vice versa.

Given a pair of graphs $G = (U, D)$ an $H = (V, E)$ a bijection $\phi : U \to V$ that preserves edges is an *isomorphism* between G and H. If such a bijection exists between G and H, they are said to be

 doi:10.20944/preprints201805.0126.v1

isomorphic and we denote it $G \approx H$. Thus, we can define the GI problem as the task of finding one of the possibly many isomorphisms between G and H or deciding that this bijection does not exist.

Finally, we define a *matching* as a subset $M \subseteq E$ where no to edges $e_1, e_2 \in M$ share a common vertex. If every vertex of G is an extreme of some edge in M, then the matching is *perfect*.

2.2. Polynomial Idelas and Algebraic Sets

Let $\mathbb{F}_q$ be the finite field of q elements and R the ring of polynomials in n variables over $\mathbb{F}_q$. An *ideal* is a subset $I \subset R$ such that for every $f, g \in I$ the sum $f + g \in I$ and for every $f \in I, h \in R$ the product $hf \in I$. Then by considering a finite set $F = \{f_1, \ldots, f_m\} \subset R$ we can define the *ideal generated by* F as follows

$$\langle F \rangle = \{h_1 f_1 + \ldots + h_m f_m | h_i \in R, i = 1, \ldots, m\}.$$

It can be seen without too much effort that a common root for the polynomials $f_i, i = 1, \ldots, m$ is also a root for any $f \in \langle F \rangle$. The *zero-set* of the ideal I consists of all the points $(x_1, \ldots, x_n) \in \mathbb{F}_q^n$ such that $f(x_1, \ldots, x_n) = 0$ for all $f \in I$, denoted $V_I(\mathbb{F}_q)$. If we consider any algebraic extension of $\mathbb{F}_q$ then the zero-set is known as the *algebraic set* of I.

We can now formalize $\mathcal{MQ}$ as a decision problem. Additionally, we state the related search problem.

DECISION PROBLEM
Instance: An ideal $I \subset \mathbb{F}_q[X_1, \ldots, X_n]$
Solution: $\begin{cases} 1 & \text{if } V_I(\mathbb{F}_q) \neq \emptyset \\ 0 & \text{otherwise} \end{cases}$

SEARCH PROBLEM
Instance: An ideal $I \subset \mathbb{F}_q[X_1, \ldots, X_n]$.
Solution: Either a proof that $V_I(\mathbb{F}_q) = \emptyset$ or a point $x \in \mathbb{F}_q^n$ such that $x \in V_I(\mathbb{F}_q)$.

A solution of the search problem gives an immediate solution for the decision problem. If we are able to find a solution for the polynomial system $f_1 = 0, \ldots, f_m = 0$ we conclude that $V_I(\mathbb{F}_q) \neq \emptyset$ and the value 1 is returned. On the other hand, if we can show that no solution exists then we return 0. This implies that the search problem is at least as difficult as the decision problem, which is known to be NP-complete.

We have seen that a solution of a polynomial system is also a solution for any element in its generated ideal. The idea behind the most common system solvers is to provide a new set of representatives (generators) of the same ideal, but with nicer properties, making it easier to find such a solution. This is the case for solvers based on the problem of finding *Gröbner Basis*. We can mention improved versions of the Buchberger Algorithm, such as F4 and F5. They have been successful to attack cryptographic schemes such as the HFE and the Matsumoto-Imai [14] an some variations of UOV [15]. Despite this efforts, the complexity of these algorithms on random instances of $\mathcal{MQ}$ is fully exponential [16].

2.3. Zero-Knowledge Proof Systems

A very useful cryptographic tool to provide identification services is the zero-knowledge proofs. In the most basic scenario, it consists of two parties: the *verifier* performs a series of questions to the *prover*, who has to answer correctly in each step to convince the verifier. The prover will be capable of answering correctly on each trial only if he has legitimate information.

For this process to be implemented successfully, some characteristics are desirable. The whole verification process should be computationally efficient for an authentic verifier, whereas it must be infeasible for a malicious entity to impersonate the authentic prover. Furthermore, no information that

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 8 May 2018 doi:10.20944/preprints201805.0126.v1

permits a tricky verifier to reveal the prover's information is gathered, though this is commonly relaxed to "no statistically significant information". Additionally, we require the following characteristics:

- *Completeness*. An authentic prover will always be accepted by an honest verifier.
- *Soundness*. If the prover is not authentic the verifier rejects with high probability.

This is, a verifier always accepts an authentic prover, but a malicious prover can impersonate an authentic one with a very small probability.

3. Construction of the Polynomial System

We exhibit the construction of the polynomial ideal from a graph and an isomorphism between them.

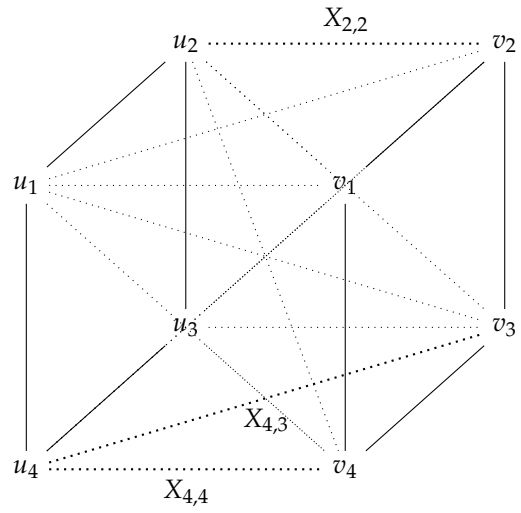
Let G and H be two isomorphic graphs of size e and order n with vertex sets $U = \{u_1, \ldots, u_n\}$ and $V = \{v_1, \ldots, v_n\}$ and edge sets D and E respectively. Let $K_{U,V}$ denote the complete bipartite graph with bipartition U, V. We get a perfect matching M in $K_{U,V}$ by selecting u_iv_k, u_jv_l into M if and only if and only if $u_iu_j \in D$ and $v_kv_l \in E$. In other words:

(i) if u_iu_j is an edge in G but v_kv_l is not an edge in H, then the edges u_iv_k and u_jv_l do not lie simultaneously in M,
(ii) if v_kv_l is an edge in H but u_iu_j is not an edge in G, then the edges u_iv_k and u_jv_l do not lie simultaneously in M.

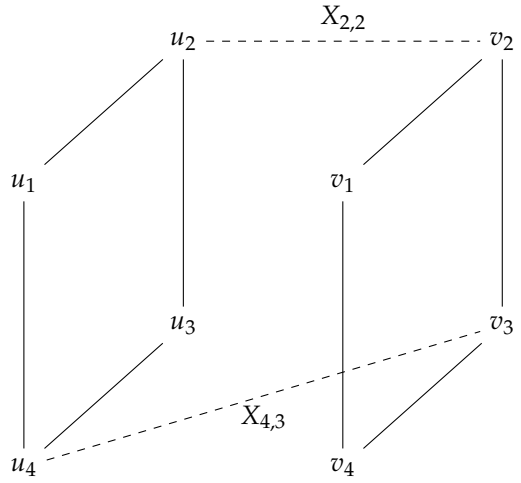
We can identify any perfect matching M built in this way with a bijection ϕ that defines the isomorphism of graphs. From a set-theoretic point of view, ϕ is treated as a collection of pairs being their first coordinate elements that belong to the domain of the function, while the second ones belong to the co-domain [17]. Conditions (i) and (ii) aforementioned constitute an alternative way to assert:

$$u_iu_j \in D \iff \phi(u_i)\phi(u_j) \in E.$$

We illustrate what we just explained in figure 1.



(a) An isomorphism between G and H can be seen as a perfect matching in the graph $K_{U,V}$, preserving adjacencies between G and H.



(b) The edges u_2v_2 and u_3v_4 cannot belong simultaneously to M because $u_2u_3 \in D$, but $v_2v_4 \notin E$. We add the equation $X_{2,2}X_{3,4} = 0$ in I.

Figure 1. Process to generate the polynomials set associated to the graph isomorphism.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 8 May 2018 doi:10.20944/preprints201805.0126.v1

Now, we perform a suitable reduction from an instance of GI to an instance of $\mathcal{MQ}$ following the same ideas exposed in reductions of several other problems in graphs to Boolean quadratic polynomials [18,19].

First we will consider the set of n^2 variables $\{X_{i,k}\}$ for $i,k = 1,\ldots,n$. We restrict any possible solution to the binary set $\{0,1\}$ by introducing the following polynomials:

$$X_{i,k}^2 - X_{i,k} \text{ for } i,k \in \{1,\ldots,n\} \tag{1}$$

Now, the following polynomials are introduced to require that one and only one vertex v_i from U is connected to one vertex of V and vice versa. This links solutions to the fact that we have a perfect matching in M.

$$\begin{aligned} &\sum_{j=1}^{n} X_{i,k} - 1 \qquad \text{for } i = 1,\ldots n \\ &\sum_{i=1}^{n} X_{i,k} - 1 \qquad \text{for } j = 1,\ldots n \end{aligned} \tag{2}$$

Finally, to guarantee that the set of polynomials has a solution related to the chosen isomorphism, we introduce a third set of polynomials:

$$\begin{aligned} &X_{i,k}X_{j,l} \text{ for any } i,j,k,l \text{ satisfying} \\ &\qquad \left(u_iu_j \in D \wedge v_kv_l \notin E\right) \vee \\ &\qquad \left(u_iu_j \notin D \wedge v_kv_l \in E\right) \end{aligned} \tag{3}$$

This completes the construction of the polynomial set related to the given GI instance.

4. Zero-Knowledge Protocol

We are ready to explain how we use the theory developed in Section 3 to perform the zero-knowledge proof.

Let us start by generating a graph G and a random bijection ϕ of its vertices. We create a second graph H which is isomorphic to G with isomorphism ϕ. Now let F_1 be the polynomial system resulting from the process of construction shown in Section 3. A solution $\mathbf{x_1}$ for the system F_1 is found by setting $X_{i,k} = 1$ if $u_iv_k \in M$ and $X_{i,k} = 0$ otherwise. The polynomial set F_1 will be public and is used as the public key. The private key will be the solution $\mathbf{x_1}$.

Next, we create a second random bijection ψ and the graph K isomorphic to G defined by this isomorphism. We get a chain of isomorphisms as follows:

$$G \xrightarrow{\phi} H \xrightarrow{\psi} K, \qquad \psi \circ \phi : G \to K.$$

We apply the same process to generate a second set F_2 of polynomials and find a solution in the exact same way as we did for the first set. We can avoid the process of graph generation by applying the permutation directly into the public system. We note that from the bijection $\psi : U \to V$ we can derive a permutation σ_ψ of the set $\{1,\ldots,n\}$ defined by $\sigma_\psi(i) = k$ if $\psi(u_i) = v_k$. This creates a mapping of variables by sending $X_{i,k}$ to $X_{i,\sigma_\psi(k)}$. We write the polynomials of F_2 satisfying condition (3) as

$$X_{i,\sigma_\psi(k)}X_{j,\sigma_\psi(l)}. \tag{4}$$

A solution for the system F_2 is provided by applying the permutation σ_ψ to reorder $\mathbf{x_1}$. A third set of polynomials F_3 can be obtained if we consider $\gamma = \psi \circ \phi$.

 doi:10.20944/preprints201805.0126.v1

Authentication protocol. The following steps are performed between Alice (the prover) and Bob (the verifier):

Key Generation:

- Alice picks a graph G and randomly generates a permutation of the set $\{1,\ldots,n\}$. This permutation is used to create the isomorphic graph H together with its isomorphism ψ. Then the public key F_1 is computed as we have established. The private key is a solution to the public system F_1.

Authentication:

1. Alice generates a permutation σ for the set $\{1,\ldots,n\}$ at random and computes the polynomial system F_2, which is sent to Bob as a *compromise*.
2. Bob creates a challenge by selecting at random $b \in \{0,1\}$. Bob sends b to Alice.
3. Once Alice has received b she must answer accordingly:
   - if $b = 0$, she sends the solution $\mathbf{x_2}$ of the system F_2 to Bob,
   - if $b = 1$, she sends σ.
4. According to the value of b Bob performs the following to authenticate Alice:
   - if $b = 0$, he checks whether $\mathbf{x_2}$ is a solution for F_2 or not,
   - if $b = 1$, he computes the system F_2' applying σ to F_1 and checks if he obtains the system F_2.

4.1. Possible attacks

We will consider that a malicious entity (Eve) wants to play the role of Alice. Then she can try the following strategy.

Eve flips a coin to decide which value b will send Bob as a challenge. If the result is $b = 0$, then she randomly generates a system F_2' with a known solution for her. Then Eve sends the system F_2' and waits for the challenge. If Bob selects $b = 0$ the Eve is able to provide an answer to the challenge. Otherwise, if $b = 1$ she will fail to provide the permutation. Now if the result of the flip is $b = 1$, the she selects a permutation at random to transform the system $F1$ into F_2'. Now she will have the answer for the challenge if Bob chooses to send $b = 1$, but she fails if this is not the case.

Now we suppose that Eve wants to obtain information about the secret key, so she plays the role of Bob. She can try asking several times and hope that she can get the same set of polynomials twice. The first time she challenges Alicia with $r = 0$ so she can get the permutation. The second one she sends $r = 1$ and gets the solution. Applying the inverse permutation to the solution she can get the private key. However, there are $n!$ different elements, and since $n! > 2^n$ this strategy is not a good one, since the running time will be exponential.

We can try to solve these problems to break the protocol with more sophisticate tools:

- $\mathcal{MQ}$: An efficient polynomial system solver to find a solution for F_1 would break completely the scheme by exhibiting the private key (even a different solution $\mathbf{x_1'}$ would work).
- Solving the *Isomorphism of Polynomials Problem* (IP), which consists of finding two affine transformations S_1, S_2 such that, for two quadratic transformations $\overline{F}, F$, we have $\overline{F} = S_2 \circ F \circ S_1$. In our case, the variable permutation can be regarded as a special case of IP where S_2 is the identity and S_1 a permutation matrix.
- Solving the GI Problem. For this approach we need to retrieve the initial isomorphic graphs from the polynomial set and find an isomorphism.

For the time being, there is no quantum algorithm that solves efficiently any of the aforementioned problems.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 8 May 2018 doi:10.20944/preprints201805.0126.v1

5. Computational complexity

We analyse the cost of creating the sets of polynomials, which is the main step in the key generation process. For the first and second sets of polynomials given in (1) and (2) we have to consider the pairs (i,k) for $i,k \in \{1,\ldots,n\}$. The asymptotic time complexity for these is $O(n^2)$.

We include now the polynomials of the form (3). We need also the solution for this system, we complete the construction with these steps:

- For every edge $u_i u_j \in D$, we look for every edge $v_k v_l$ in the complementary graph $\overline{H}$. We add the corresponding polynomials $X_{i,k}X_{j,l}$ to the system.
- For every edge $v_k v_l \in D$, we walk over every edge $v_k v_l$ in the complement $\overline{G}$. We add the corresponding polynomials $X_{i,k}X_{j,l}$ to the system.
- With the chosen isomorphism $\phi : G \to H$ we create the complete bipartite graph $K_{U,V}$ an the matching $M = \{u_i\phi(u_i) | u_i \in U\}$.

These equations comprise a total number bounded by n^2e, where e is size of G. Then we can build the complete system in time $O(n^2e)$, which is polynomial on the order of G.

5.1. Toy example

We will show the construction of a polynomial set with a small example. Let us consider the graph $G = (U, D)$ with $U = \{1,2,3,4\}, D = \{(1,2),(1,4),(2,3),(3,4)\}$ and the permutation

$$\sigma = \begin{pmatrix} 1 & 2 & 3 & 4 \\ 1 & 3 & 2 & 4 \end{pmatrix}$$

By applying σ to the vertex set U we get the isomorphic graph $H = \{V, E\}$ where $V = U$ and $E = \{(1,3),(1,4),(2,3),(2,4)\}$. The graphs G, H are shown in Fig. 2. The dashed lines are the edges in the corresponding complementary graphs.

(a) Graph G **(b)** Graph H

Figure 2. Isomorphic graphs. Dashed lines correspond to the complementary graphs

To build the polynomial set, we start with the edges in G and $\overline{H}$. For instance, considering $(1,2) \in G$ and $(3,4) \in \overline{H}$, we get the polynomials $X_{1,2}X_{3,4}$. Once we walk over all the edges of G in this fashion, we get the polynomials

$$X_{1,1}X_{2,2},\ X_{1,1}X_{4,2},\ X_{2,1}X_{3,2},\ X_{3,1}X_{4,2}$$
$$X_{1,3}X_{2,4},\ X_{1,3}X_{4,4},\ X_{2,3}X_{3,4};\ X_{3,3}X_{4,4}.$$

Now, by considering the edges $\overline{G}$ and H, we get another set of 8 polynomials:

$$X_{1,1}X_{4,3},\ X_{1,1}X_{4,2},\ X_{2,1}X_{3,2},\ X_{3,1}X_{4,2}$$
$$X_{2,1}X_{3,3},\ X_{2,3}X_{3,4},\ X_{2,3}X_{3,4};\ X_{3,3}X_{4,4}.$$

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 8 May 2018 doi:10.20944/preprints201805.0126.v1

The roots of these polynomials related to the isomorphism between these graphs can be computed by letting $x_{i,\sigma(i)} = 1$ for $i = 1, 2, 3, 4$ and 0 in other case. Then $x_{1,1} = x_{1,4} = x_{2,3} = x_{2,4} = 1$ and$x_{i,j} = 0$ for the rest of the elements.

6. Conclusions

We have built an alternative zero-knowledge authentication protocol whose security relies in the difficulty of solving $\mathcal{MQ}$. A solution for this set of polynomials represents an isomorphism between graphs. Then we guarantee that the protocol is at least as secure as the classical ZKP based solely in the GI problem. We have also shown that the construction is feasible in terms of time complexity, and since only a permutation of length n or a binary vector of size n^2 is sent in response at every step, most of the information interchanged on every interaction consists of the set of polynomials, which is a bit string in the order of $O(n^4)$. We leave as a future work to verify the possibility of reducing the number of polynomials in the system without weakening the proof system, as well as a complete implementation of the authentication protocol.

Acknowledgments: The authors acknowledge the support of Mexican CONACYT. The first author has a grant of Conacyt's Scholarship Program, the last two authors have been partially supported from Conacyt's National System of Researchers. Finally, the support from ABACUS-CINVESTAV (Conacyt, EDOMEX-2011-C01-165873) is gratefully acknowledged.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

DLP	Discrete Logarithm Problem
DSA	Digital Signature Algorithm
GI	Graph Isomorphism Problem
ECDSA	Elliptic Curve Digital Signature Algorithm
$\mathcal{MQ}$	Multivariate Quadratic Problem
NP	Non-deterministic Polynomial Time
PKI	Public Key Infrastructure
PKC	Public Key Cryptography
MPKC	Multivariate Public Key Cryptography
RSA	Rivest, Shamir and Adleman (a public key cryptographic scheme)
ZKP	Zero-Knowledge Proof

References

1. Goldwasser, S.; Micali, S.; Rackoff, C. The Knowledge Complexity of Interactive Proof-systems. Proceedings of the Seventeenth Annual ACM Symposium on Theory of Computing, Providence, Rhode Island, USA, May 6-8, 1985; ACM: New York, USA, 1985; pp. 291–304.
2. Sakalauskas, E. The multivariate quadratic power problem over $\mathbb{Z}_n$ is NP-complete. *Inf. Tech. Control* **2012**, *4*, 33–39.
3. Ding, J.; Gower, J.E.; Schmidt, D.S. *Multivariate Public Key Cryptosystems*, 1 ed.; Vol. 25, Springer-Verlag: Berlin, Heidelberg, 2006.
4. Imai, H.; Matsumoto, T. Algebraic methods for constructing asymmetric cryptosystems. Proceedings of the 3rd International Conference on Applied Algebra, Algebraic Algorithms, and Error-Correcting Codes, Grenoble, France, July 15-19, 1985; Springer: Berlin/Heidelberg, Germany, 1986; pp. 108–119.
5. Matsumoto, T.; Imai, H. Public Quadratic Polynomial-Tuples for Efficient Signature-Verification and Message-Encryption. Proceedings of the Workshop on the Theory and Application of Cryptographic Techniques — EUROCRYPT '88, Davos, Switzerland, May 25–27, 1988; Springer: Berlin/Heidelberg, Germany, 1988; pp. 419–453.

 doi:10.20944/preprints201805.0126.v1

6. Patarin, J. Cryptanalysis of the Matsumoto and Imai public key scheme of Eurocrypt'88. Proceedings of the 37th Annual International Cryptology Conference — CRYPTO 1995, Santa Barbara, CA, USA, August 27-31, 1995; Springer: Berlin/Heidelberg, Germany, 1995; pp. 248–261.
7. Kipnis, A.; Patarin, J.; Goubin, L. Unbalanced Oil and Vinegar Signature Schemes. Proceedings of the International Conference on the Theory and Application of Cryptographic Techniques — EUROCRYPT '99, Prague, Czech Republic, May 2-May 6, 1999; Springer: Berlin/Heidelberg, Germany, 1999; pp. 206–222.
8. Patarin, J. Hidden Fields Equations (HFE) and Isomorphisms of Polynomials (IP): Two New Families of Asymmetric Algorithms. Proceedings of the International Conference on the Theory and Application of Cryptographic Techniques — EUROCRYPT '96, Saragossa, Spain, May 12–16, 1996; Springer: Berlin/Heidelberg, Germany, 1996; pp. 33–48.
9. Ding, J.; Schmidt, D. Rainbow, a New Multivariable Polynomial Signature Scheme. Proceedings of the Third International Conference on Applied Cryptography and Network Security, New York, NY, USA, June 7-10, 2005; Springer: Berlin/Heidelberg, Germany, 2005; pp. 164–175.
10. NIST News Service. Candidate Quantum-Resistant Cryptographic Algorithms Publicly Available, December 28, 2017. https://www.nist.gov/news-events/news/2017/12/candidate-quantum-resistant-cryptographic-algorithms-publicly-available.
11. Buchberger, B., An Algorithmic Criterion for the Solvability of a System of Algebraic Equations. In *Gröbner Bases and Applications*; Number 251 in London Math. Soc. Lecture Note Ser., Cambridge University Press: Cambridge, UK, 1998; pp. 535–545.
12. Bernstein, D.J.; Buchmann, J.; Dahmen, E. *Post-Quantum Cryptography*, 1 ed.; Springer: Berlin/Heidelberg, Germany, 2009.
13. Babai, L. Graph isomorphism in quasipolynomial time. Available online: http://arxiv.org/abs/1512.03547, 2015, [1512.03547].
14. Faugère, J.C.; Joux, A. Algebraic cryptanalysis of Hidden Field Equations (HFE) using Gröbner bases. Proceedings of the 23rd Annual International Cryptology Conference — CRYPTO 2003, Santa Barbara, CA, USA, August 17-21, 2003; Springer: Berlin/Heidelberg, Germany, 2003; pp. 44–60.
15. Braeken, A.; Wolf, C.; Preneel, B. A Study of the Security of Unbalanced Oil and Vinegar Signature Schemes. Proceedings of the Cryptographers' Track at the RSA Conference 2005 — CT-RSA 2005, San Francisco, CA, USA, February 14-18, 2005; Springer: Berlin/Heidelberg, Germany, 2005; pp. 29–43.
16. Bard, G. *Algebraic Cryptanalysis*, 1 ed.; Springer: Berlin/Heidelberg, 2009.
17. Moschovakis, Y. *Notes on Set Theory*; Undergrad. Texts Math., Springer-Verlag: New York, USA, 2005.
18. Goldreich, O. *Computational Complexity: A Conceptual Perspective*; Cambridge University Press: Cambridge, UK, 2008.
19. Nemhauser, G.L.; Wolsey, L.A. *Integer and Combinatorial Optimization*; Wiley-Interscience: New York, NY, USA, 1988.