Zero Knowledge Authentication Protocols With Algebraic Geometry Techniques

Edgar González*[1], Guillermo Morales†[1], and Feliú Sagols‡[2]

[1]Department of Computer Science, Cinvestav-IPN, Mexico City
[2]Department of Mathematics, Cinvestav-IPN, Mexico City

July 26, 2016

Abstract

Several cryptographic methods have been developed based on the difficulty to determine the set of solutions of a polynomial system over a given field. We build a polynomial ideal whose algebraic set is related to the set of isomorphisms between two graphs. The problem ISOMORPHISM, posed in the context of Graph Theory, has been extensively used in zero knowledge authentication protocols. Thus, any cryptographic method based on ISOMORPHISM can be translated into an equivalent method based on the problem of finding rational points in algebraic sets associated to polynomial ideals.

Keywords— zero knowledge procedures, graph isomorphism problem, multivariate polynomial system.

1 Introduction

Public Key Cryptography (PKC) is based on *one-way maps.* Public Key Encryption and Digital Signatures are two of the most relevant applications of PKC. In order to ensure secure communication, *public keys* are used for message encryption or signature verification, while *private keys* are used for message decryption or signature creation.

Recently, the NP-hard problem to solve a multivariate quadratic system, denoted $\mathcal{MQ}$, has been exploited in PKC, since it is believed to resist quantum computers attacks [1]. This is an advantage with respect to more popular methods as RSA, DSA and ECDSA. Most of the PKC systems based on $\mathcal{MQ}$ consist of the following. An easily solvable quadratic map $Q : \mathbb{F}_q^n \to \mathbb{F}_q^m$, two affine bijective transformations $S : \mathbb{F}_q^n \to \mathbb{F}_q^n, T : \mathbb{F}_q^m \to \mathbb{F}_q^m$, and a quadratic system P obtained as $P := T \circ Q \circ S$, which in turn must be difficult to solve. The system P is used as the public key, while Q, S and T conform the private key. The main purpose of S and T is to hide the algebraic structure that makes Q easy to solve. In [2], [3], $\mathcal{MQ}$ was used for cryptographic purposes, producing the so-called *Matsumoto-Imai crypto-system.* Similar schemes have appeared later, as *Unbalanced Oil-Vinegar* (UOV) [4], *Hidden Field Equations* (HFE) [5], QUARTZ [6], and several variations intended to repair security weakness problems. Many of the aforementioned schemes have shown vulnerabilities against cryptographic analysis, mainly because the proposed constructions generate weak and easily solvable instances of $\mathcal{MQ}$.

In a general way, attacks on schemes based on $\mathcal{MQ}$ can be classified in two categories:

- *General attacks.* The solutions to the public system P are searched directly on the system itself.
- *Structure based attacks.* The construction of P is used to obtain information about the resulting system algebraic structure.

Some successful algorithms to attack $\mathcal{MQ}$ protocols use *Buchberger Algorithm* [7] to compute Gröbner bases. For instance, the algorithms F4 [8] and F5 [9] are based on *linearization*, which was used as well in the procedure XL [10] (*eXtended Linearization*), and *Zhuang-Zi Algorithm* [11].

*egonzalez@computacion.cs.cinvestav.mx
†gmorales@cs.cinvestav.mx
‡fsagols@math.cinvestav.edu.mx

The algorithms *Rank Attack*, (*High Rank*, *MinRank* and Separation of Oil and Vinegar) [12](see Section [VI.5.4]), are structure based attacks and succeeded in breaking the schemes UOV and HFE [13, 14]. Other successful attacks used *linearization equations* [15] to solve instances produced by the Matsumoto and Imai scheme in its original form.

Our approach reduces instances of the *Graph Isomorphism Problem* (ISOMORPHISM) to instances of $\mathcal{MQ}$. Thus, the most direct attack to our method may consist in either to calculate a solution to the translated polynomial system or to locate a graph isomorphism. It is worth to mention that recently a procedure claiming to solve ISOMORPHISM instances in quasi-polynomial time was published [16]. Thus the problem to find appropriate instances for cryptographic purposes is more relevant.

The key idea in our reduction is the difficulty to find rational points in the algebraic set determined by polynomial ideals. In practice, we propose and test authentication schemes to verify its security through currently available effective computational tools, mainly the computer program `PoliBoRy` [17], even though the known algorithms to solve $\mathcal{MQ}$ have super-polynomial time complexity.

This paper is constituted as follows. In Section 2 we introduce basic concepts about graph theory and polynomial ideals. Section 3 recalls zero-knowledge protocols. We continue with the reduction of ISOMORPHISM instances to the $\mathcal{MQ}$ problem in Section 4. In Section 5 some cryptographic applications are developed. Finally, in Section 6 we give a complexity estimation of the given procedures.

2 Preliminaries

Let us recall elementary graph theory and polynomial ideals concepts.

2.1 Graphs

A *graph* G is a pair (V, E), where V is the set of *vertices*, and $E \subseteq V^{(2)} = \{a \subset V| \,\#a = 2\}$ is a set of *edges*. The cardinalities of V and E are called, respectively, the *order* and the *size* of the graph. Two vertices v_i, v_j, $v_i \neq v_j$, are *adjacent* if $v_iv_j \in E$. Two edges $e_1, e_2 \in E$ are *adjacent* if they share a common vertex. For a given graph G, $V(G)$ denotes its set of vertices, and $E(G)$ its set of edges. If $E(G) = V^{(2)}$ the graph is *complete*, it is unique up to isomorphism. We denote the complete graph on n vertices by K_n .

We say that a graph G is *bipartite* if there exists a partition $\{V_1, V_2\}$ of V such that no edge has both vertices in the same set V_i, $i \in \{1, 2\}$. Equivalently, every edge has an extreme in V_1 and the other in V_2. The graph is *complete bipartite* if any vertex of V_1 is adjacent to every vertex of V_2 and vice versa.

Two graphs $G_1 = (V_1, E_1), G_2 = (V_2, E_2)$ are *isomorphic* if there exists a bijective map $\phi : V_1 \to V_2$ such that two vertices $v_i, v_j \in V$ are adjacent in G_1 if and only if $\phi(v_i), \phi(v_j)$ are adjacent in G_2. The bijective map ϕ is an *isomorphism* from G_1 to G_2. The *Graph Isomorphism Problem* (ISOMORPHISM) of two graphs G_1, G_2 consists in finding an isomorphism $\phi : G_1 \to G_2$ provided that G_1 and G_2 are isomorphic, or proving that they are not isomorphic otherwise.

A *matching* is a subset $M \subseteq E$ such that no pair of edges $e_1, e_2 \in M$ are adjacent. We say that the matching is *complete* if every vertex of G is an extreme of an edge in M.

2.2 Algebraic sets and rational points

We consider an arbitrary field $\mathbb{K}$ and $R = \mathbb{K}[X_1, \ldots, X_n]$, the *ring of polynomials* in n variables $X_1, \ldots, X_n$ with coefficients in $\mathbb{K}$. An additive subgroup $I \subseteq R$ is an *ideal* if for every $f \in I$, $gf \in I$ for any $g \in R$. Let $f \in R$. The *ideal generated by* f is $\langle f \rangle = \{gf | g \in R\}$. Similarly, for an arbitrary set of m polynomials $F \subseteq R$, the *ideal generated by* F is

$$\langle F \rangle = \{g_1f_1 + \ldots + g_mf_m | g_i \in R, f_i \in F, i = 1, \ldots, m\}.$$

Let $\overline{\mathbb{K}}$ denote the algebraic closure of $\mathbb{K}$, then the *algebraic set* defined by I is given by

$$V_I = \{\mathbf{x} \in \overline{\mathbb{K}}^n | f(\mathbf{x}) = 0 \text{ for every } f \in I\}.$$

The set $V_I(\mathbb{K}) = V_I \bigcap \mathbb{K}^n$ is known as the set of $\mathbb{K}$-*rational points of* V_I. In the following, we will focus on finite fields $\mathbb{F}_q$, where q is the power of a prime p. Then we may refer to the set of $\mathbb{F}_q$-rational points as rational points for simplicity.

It arises naturally the problem of characterising the set of rational points $V_I(\mathbb{F}_q)$ given an ideal I. This problem can be stated in two versions: as a *decision* problem, or as a *search* problem.

Decision problem
Instance: An ideal $I \subset \mathbb{F}_q[X_1, \ldots, X_n]$ and a point $x \in \mathbb{F}_q^n$.
Solution: $\begin{cases} 1 & \text{if } x \in V_I(\mathbb{F}_q) \\ 0 & \text{if } x \notin V_I(\mathbb{F}_q) \end{cases}$

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 $x \in V_I(\mathbb{F}_q)$.

If the ideal I is provided by a finite set of generators $\{f_1, \ldots, f_m\}$, the Decision Problem is trivial: it suffices to check whether $f_i(x) = 0$ for $i = 1, \ldots, m$. However, the Search Problem implies finding a solution $x \in \mathbb{F}_q^n$ of the system of simultaneous equations $\{f_i(X) = 0 |\ i = 1, \ldots, m\}$, which can be a very complex task if the degrees of the generators are high.

Today, the most useful methods to solve systems of polynomial equations are based on *Buchberger Algorithm* to find *Gröbner Basis* of the ideals generated by the polynomials in the system. Improved versions of Buchberger Algorithm, such as F4 and F5 are useful in solving $\mathcal{MQ}$ directly. These algorithms have provided effective cryptographic attacks against some schemes, such as the HFE, or against some particular selection of parameters in other schemes. This is the case of UOV [18], when the number of vinegar variables exceeds greatly the quantity of oil variables. Even though, the worst case time complexity of these algorithms is doubly exponential [19].

3 Zero knowledge protocols

Informally, a zero knowledge proof allows an entity to show the possession of certain information without exposing details. The idea is to consider the verifier as a potential adversary who tries to acquire information.

A zero knowledge protocol consists of a verification process between two entities. A *verifier*, who executes the process. And a *prover*, who tries to convince the verifier that he or she possesses valid identity credentials.

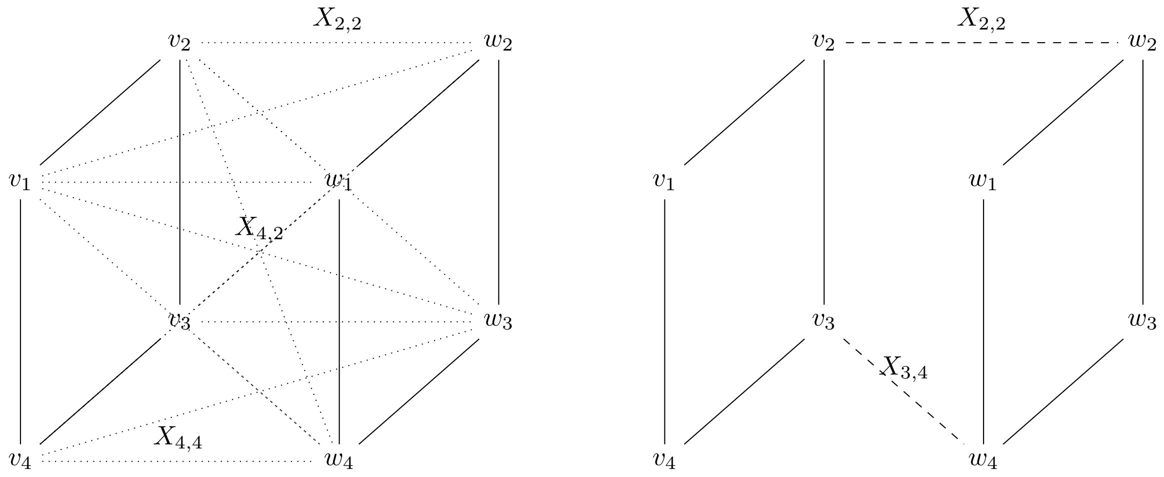
This verification process has to follow certain rules. It must be computed efficiently, whereas finding the required proof has to be a computationally difficult task for any other entity distinct to the prover.

A *strategy* describes the next action to be performed by each entity at any stage of the process. This could be regarded as a game. Assuming that the verifier takes the first move, then he asks a question to the prover involving the assertion to be verified. If the prover's claim is true, then an authentic verifier can be convinced by an authentic prover. If the assertion is false, then it should be impossible for a non-authentic prover to convince the verifier, or at most, the non-authentic prover should have a very small success probability. These properties characterise the *interactive proof systems*. In order to formalise the interaction between prover and verifier, let us refer to a strategy as a function depending on a common input and the interactions made so far. For a pair of strategies A and B, we denote by r_A, r_B their respective *randomness*. Assuming that A takes the first move, the interaction of A and B after t rounds on a common input x is denoted by $A(x, r_A, \beta_1, \ldots, \beta_t)$ and $B(x, r_B, \alpha_1, \ldots, \alpha_t)$ where $\alpha_i = A(x, r_A, \beta_1, \ldots, \beta_{i-1})$ and $\beta_i = B(x, r_B, \alpha_1, \ldots, \alpha_i)$. A *probabilistic strategy* is then a probability distribution over a set of deterministic strategies, defined by its randomness. An entity uses a *probabilistic polynomial time strategy* if its next step can be computed a number of steps that is polynomial in the size of x.

Definition 1 *An* interactive proof system *for a set S consists of an interaction between two entities, a* verifier *who runs a strategy V in probabilistic polynomial time, and a* prover*, who runs a computationally unbounded strategy P with the following characteristics:*

- Completeness. *For all $x \in S$, the verifier V always accepts after interacting with the authentic prover P on a common input x.*
- Soundness. *For all $x \notin S$ and every strategy P^*, the verifier V rejects with probability at least $\frac{1}{2}$ after interacting with P^* on a common input x.*

Definition 2 *The strategy of a prover P is* perfect zero knowledge *over a set S if for any probabilistic polynomial time strategy V^* of a verifier, there exists a probabilistic polynomial time algorithm A^* such that $(P, V^*)(x) \equiv A^*(x)$ for all $x \in S$, where $(P, V^*)(x)$ is a random variable that represents the output of the verifier V^* after interacting with the prover P on a common input x, and $A^*(x)$ is a random variable representing the output if algorithm A^* in the input x.*



(a) To decide if G_1 y G_2 are isomorphic, we must find a perfect matching M using the edges in dashed lines, preserving adjacencies between G_1 and G_2.

(b) Both edges v_2w_2 and v_3w_4 cannot belong simultaneously to M because $v_2v_3 \in E(G_1)$, but $w_2w_4 \notin E(G_2)$. So, we may define the equation $X_{2,2}X_{3,4} = 0$ in I.

Figure 1: Process to identify isomorphic graphs.

The symbol $\equiv$ in the last definition represents an equality. If we allow this equality to change to a bounded statistical proximity, then the definition of *quasi-perfect* or *statistical zero knowledge* arises. Generally, this is the meaning given to the phrase zero knowledge, the proximity thus meaning *computationally indistinguishable*. More detailed information about Zero Knowledge Proofs can be found in [20] Ch 9.

4 Ideal associated to ISOMORPHISM

Let G_1 and G_2 be two graphs, both having order n and size e. Henceforth, we will write $V(G_1) = \{v_1, \ldots, v_n\}$ and $V(G_2) = \{w_1, \ldots, w_n\}$. Let $K_{V(G_1),V(G_2)}$ denote the complete bipartite graph with bipartition $V(G_1)$, $V(G_2)$.

If G_1 and G_2 are isomorphic, then there exists a perfect matching M in $K_{V(G_1),V(G_2)}$ such that $v_iw_k, v_jw_l \in M$ if and only if and only if $v_iv_j \in E(G_1)$ and $w_kw_l \in E(G_2)$. In other words:

1. if v_iv_j is an edge in G_1 but w_kw_l is not an edge in G_2, then both edges v_iw_k and v_jw_l cannot lie simultaneously in the matching M,

2. if w_kw_l is an edge in G_2 but v_iv_j is not an edge in G_1, then both edges v_iw_k and v_jw_l cannot lie simultaneously in the matching M.

The perfect matching M plays the role of the bijection ϕ in the isomorphism definition, as stated in Section 2. From the point of view of Set Theory, a function is just a collection of pairs whose first elements belong to the domain of the function, and the second ones are elements of the co-domain [21]. Conditions 1) and 2) constitute an alternative way to assert:

$$v_iv_j \in E(G_1) \iff \phi(v_i)\phi(v_j) \in E(G_2).$$

In Figure 1 we illustrate the above notions.

Now, we translate the notion of isomorphism between graphs to a strictly algebraic language. The idea is to perform a proper reduction from ISOMORPHISM to $\mathcal{MQ}$ motivated by conventional reductions of several problems in graphs to Boolean quadratic polynomials [20, 22].

Let G_1 and G_2 be a pair of graphs of order n such that there exists a perfect matching M in $K_{V(G_1),V(G_2)}$. In addition, $v_iw_k, v_jw_l \in M$ if and only if $v_iv_j \in E(G_1)$ and $w_kw_l \in E(G_2)$. We consider the polynomial ring over a fixed field $\mathbb{K}$ with a set of variables $\mathbf{X} = \{X_{i,j}\}$ for $i, j \in \{1, \ldots, n\}$, and we will consider the polynomial ring $\mathbb{K}[\mathbf{X}]$. We restrict the values for each variable to $\{0, 1\}$, so the solutions will be obtained as elements of the vector space $\mathbb{F}_2^{n^2}$. A variable $X_{i,j}$ takes the value 1 if and only if the edge $v_i w_j$ is an element of the matching M. We proceed as follows.

In order to restrict the values of every variable to the set $\{0,1\}$ we introduce the set of polynomials:

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

Finally, we include the following polynomials to

Now, to restrict the set of one valued variables to only those that represent a perfect matching in M the following polynomials must be satisfied:

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

Here, the first set of polynomials force each vertex of G_1 to have an incident edge in M. The second one works in the same way for G_2.

In order to ensure that only matchings coming from isomorphisms between G_1 and G_2 can be obtained as a zero of the ideal, we introduce a further set of polynomials:

$$\begin{aligned} X_{i,k}X_{j,l} \text{ for any } & i,j,k,l \text{ such that} \\ & (v_i v_j \in E(G_1) \wedge w_k w_l \notin E(G_2)) \vee \\ & (v_i v_j \notin E(G_1) \wedge w_k w_l \in E(G_2)) \end{aligned} \tag{3}$$

Hence, if the set of rational points defined by the ideal I generated by the polynomials in (1), (2) and (3) is not empty, then G_1 and G_2 are isomorphic.

5 Cryptographic applications

Now, we apply the theory developed in Section 4 to construct a zero knowledge authentication protocol based on the ISOMORPHISM and $\mathcal{MQ}$ problems.

Let G_1 and G_2 be a pair of isomorphic graphs. Let P_1 be the set of polynomials associated to the set of isomorphisms between G_1 and G_2, and let $\mathbf{x}$ be a solution of system P_1. Then $\mathbf{x}$ could be considered as a matching M on $K_{V(G_1),V(G_2)}$, or an isomorphism ϕ between G_1 and G_2 as it was explained in Section 4. In order to create a zero knowledge authentication protocol, we obtain a new set of polynomials as follows:

Consider a new graph G_3 such that there exists an isomorphism $\psi : G_2 \to G_3$:

$$\begin{tikzcd} G_1 \arrow[r, "\phi"] & G_2 \arrow[r, "\psi"] & G_3 \end{tikzcd}$$

Let us find a new set P_2 of polynomials with its set of solutions associated to the perfect matchings of $K_{V(G_1),V(G_3)}$, but using a more direct construction. For each vertex $u_r \in V(G_3)$ there exists $w_k \in V(G_2)$ such that $\psi(w_k) = u_r$. This defines a permutation σ_ψ on $\{1,\ldots,n\}$ such that $\psi(w_k) = u_{\sigma_\psi(k)}$, $\forall w_k \in V(G_2)$. Using the fact that

$$w_k w_l \in E(G_2) \iff \psi(w_k)\psi(w_l) \in E(G_3),$$

we write the polynomials of P_2 satisfying condition (3) as

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

The other polynomials in P_2 are built directly from expresions (1) and (2).

A solution for P_2 can be found easily from $\mathbf{x}$ by applying the permutation σ_ψ. Analogously, we can obtain another set of polynomials P_3 generating the isomorphisms between G_2 and G_3 by considering $\gamma = \psi \circ \phi$.

Authentication protocol. Alice is the prover and Bob the verifier:

Key Generation:

- Alice generates a pair of isomorphic graphs G_1, G_2 along with an isomorphism ϕ between them. The public key is the associated system P_1 with the characteristics aforementioned. The private key is a solution to the system P_1, which can be obtained from the matching M representing the isomorphism ϕ.

Authentication Protocol:

1. Alice selects at random a permutation σ of the set of vertices $\{1, \ldots, n\}$ and generates a new set P_2 of polynomials as seen on (4). She sends the system P_2 to Bob as a *compromise.*
2. Bob chooses a random bit $b \in \{0, 1\}$ and sends it to Alice.
3. Alice receives b as a challenge:
    - if $b = 0$, she must send a solution $\mathbf{x}'$ of the system P_2 to Bob,
    - if $b = 1$, she sends the permutation σ used to generate P_2.
4. Bob performs the following steps to authenticate Alice:
    - if $b = 0$, he checks that $\mathbf{x}'$ is a solution of P_2,
    - if $b = 1$, he applies the permutation σ to P_1 and checks if he obtains the system P_2.

There are many other authentication protocol schemes, which are based directly on the problem *Isomorphism of Polynomials* (IP) and $\mathcal{MQ}$ [23], [24] and even in a more general case [25] than the one presented here. The last generalisation allows to generate a multivariate system Q at random, and to create new systems by applying an invertible affine map S to get $P = Q \circ S$. In our case, the polynomial is not selected at random, and the affine map can be regarded as a permutation of the rows from the identity matrix, which is of course invertible. However, some advantages can be listed.

- Since the algebraic set is related to the set of isomorphisms of two graphs, trying to solve the system is at least as difficult as ISOMORPHISM.
- The amount of data sent in each iteration of the process can be reduced, since we do not need to send an affine transformation, but only the permutation required to get the new system.
- Furthermore, in each iteration a new system comes up by applying a simple permutation on the indices of the variables, which is faster than applying an affine transformation.

Thus, we can at least rely on the difficulty of solving ISOMORPHISM as a lower bound for the security of this protocol.

5.1 Security notes

An entity, say Eve, who tries to impersonate Bob may proceed in two different ways.

If Eve predicts that Bob will challenge with $b = 0$, then she can generate an arbitrary system P_2' and create a solution to this challenge. In this case, Eve sends P_2' to Bob and has the chance to provide a solution to P_2'. However, Eve will not be capable of providing a permutation that transforms P_1 into P_2'. Otherwise, if Eve believes that Bob will challenge with the bit $b = 1$, then she can generate a random permutation and create a system P_2' that will be sent to Bob. In this way, Eve can provide the permutation used to obtain the new system. But, since she does not have a solution to P_1, it will be difficult to provide a solution for P_2'.

Now, if Bob tries to obtain supplementary information while interacting with Alice, then, in the same fashion, he will receive only a piece of information in any of the following two cases.

If Bob challenges Alice with $b = 0$, then he will know a solution of the associated system P_2. Not knowing how the system P_1 has been transformed into P_2, he will not be able to recover the original solution $\mathbf{x}$. If Bob challenges with $b = 1$, then he will have the permutation σ applied to P_1, but as he does not know a solution for P_2, he will not be able to recover $\mathbf{x}$.

Thus, the scheme can be broken if any of the following problems can be solved efficiently for the instances that come up from this technique:

- The $\mathcal{MQ}$ problem: if we can find a solution $\mathbf{x}'$ to the polynomial P_1, then we can forge a valid private key.

- The *Isomorphism of Polynomials Problem*: The permutation applied to the system P_1 can be regarded as an invertible linear transformation S such that $P_2 = P_1 \circ S$. In our case, S is a permutation matrix.
- The Graph Isomorphism Problem, as has been stated before.

6 Complexity analysis

In order to create the set of polynomials associated to two isomorphic graphs G_1 and G_2, we need to consider all pairs (i, j) with $i, j \in \{1, \ldots, n\}$ to generate the polynomials that satisfies conditions (1) and (2). This will take $O(n^2)$ steps.

Next, to add the set of polynomials that correspond to condition (3) and provide a solution for the resulting system, we proceed as follows:

- For each edge $v_i v_j \in E(G_1)$, we look for every edge $w_k w_l$ in the complement $\overline{G_2}$. We add the corresponding polynomials $X_{i,k}X_{j,l}$ to the system.
- For each edge $w_k w_l \in E(G_1)$, we go over al the edges $w_k w_l$ in the complement $\overline{G_1}$. We add the corresponding polynomials $X_{i,k}X_{j,l}$ to the system.
- Finally, given a fixed isomorphism $\phi : G_1 \to G_2$ we build the matching M given by the set of edges of the form $v_i \phi(v_i)$, and proceed as we have argued before.

If e is the size of G, then the quantity of equations generated in step 3 is bounded by $n^2 e$, so the resulting system can be accomplished in polynomial time in the order and size of G.

7 Conclusions

By means of the reductions presented in this paper, the conventional procedures of zero knowledge authentication protocols can be settled as procedures based on the problem of finding solutions to a system of quadratic equations. Equivalently, the procedures for key generation, which consist on the generation of difficult instances of the Graph Isomorphism Problem, can be reduced to procedures for key generation for the corresponding reductions to zero knowledge authentication based on the problem of finding solutions to the system.

Since every system of polynomials equations can be regarded as a system of polynomial equations of degree 2, the authentication protocols presented are comparable to the $\mathcal{MQ}$ problem.

References

[1] E. Sakalauskas, “The multivariate quadratic power problem over $\digamma_n$ is NP-complete,” *Information Technology and Control*, vol. 4, no. 1, pp. 33–39, 2012.

[2] T. Matsumoto, “A public key cryptosystem based on the difficulty of solving a system of nonlinear equations,” *ICICE Transactions*, vol. J69-D, no. 12.

[3] T. Matsumoto and H. Imai, “Public quadratic polynomial-tuples for efficient signature-verification and message-encryption,” *Advances in Cryptology — EUROCRYPT ’88*, pp. 419–453, 1988. [Online]. Available: http://dx.doi.org/10.1007/3-540-45961-8_39

[4] A. Kipnis, J. Patarin, and L. Goubin, “Unbalanced Oil and Vinegar Signature Schemes,” in *Advances in Cryptology — EUROCRYPT ’99*, ser. Lecture Notes in Computer Science, J. Stern, Ed. Berlin, Heidelberg: Springer Berlin / Heidelberg, 1999, vol. 1592, ch. 15, pp. 206–222. [Online]. Available: http://dx.doi.org/10.1007/3-540-48910-x_15

[5] J. Faugère and A. Joux, “Algebraic cryptanalysis of Hidden Field Equations (HFE) using Gröbner bases,” *Advances in Cryptology — CRYPTO 2003*, vol. 2729, pp. 44–60, 2003.

[6] N. T. Courtois, L. Goubin, and J. Patarin, “Quartz, an asymmetric signature scheme for short signatures on pc – primitive specification and supporting documentation,” 2001.

[7] B. Buchberger, “Ein algorithmisches Kriterium fuer die Loesbarkeit eines algebraischen Gleichungssystems (An Algorithmic Criterion for the Solvability of Algebraic Systems of Equations),” *Aequationes mathematicae*, vol. 3, pp. 374–383, 1970, (english transl.: B. Buchberger, F. Winkler: Groebner Bases and Applications, Proc. of the International Conference ”33 Years of Groebner Bases”, 1998, RISC, Austria, London Math. Society Lecture Note Series 251, Cambridge Univ. Press, 1998, pp.535 -545).

[8] J. Faugère, “A new efficient algorithm for computing Gröbner bases (F4),” *Journal of Pure and Applied Algebra*, no. 139, pp. 61–88, 1999.

[9] ——, “A new efficient algorithm for computing Gröbner bases without reduction to zero (F5),” *International Symposium on Symbolic and Algebraic Computation — ISSAC 2002*, pp. 75–83, 2002.

[10] N. Courtis, A. Shamir, J. Patarin, and A. Klimov, “Efficient Algorithms for solving Overdefined Systems of Multivariate Polynomial Equations,” *Advances in Cryptology — EUROCRYPT 2000*, pp. 392–407, 2000.

[11] J. Ding, J. E. Gower, and D. S. Schmidt, “Zhuang-zi: A new algorithm for solving multivariate polynomial equations over a finite field,” 2006.

[12] D. J. Bernstein, J. Buchmann, and E. Dahmen, *Post Quantum Cryptography*, 1st ed. Springer Publishing Company, Incorporated, 2008.

[13] A. Kipnis and A. Shamir, “Cryptanalysis of the oil and vinegar signature scheme,” *Advances in Cryptology — CRYPTO ’98*, pp. 257–266, 1998. [Online]. Available: http://dx.doi.org/10.1007/BFb0055733

[14] ——, “Cryptanalysis of the hfe public key cryptosystem by relinearization,” *Advances in Cryptology — CRYPTO’ 99*, pp. 19–30, 1999.

[15] J. Patarin, “Cryptanalysis of the Matsumoto and Imai public key scheme of Eurocrypt’88,” *Advances in Cryptology — CRYPTO 1995*, vol. 963, pp. 248–261, 1995.

[16] L. Babai, “Graph isomorphism in quasipolynomial time,” *CoRR*, vol. abs/1512.03547, 2015. [Online]. Available: http://arxiv.org/abs/1512.03547

[17] M. Brickenstein and A. Dreyer, “Polybori: A framework for Gröbner-basis computations with boolean polynomials,” *Journal of Symbolic Computation*, vol. 44, no. 9, pp. 1326 – 1345, 2009, effective Methods in Algebraic Geometry. [Online]. Available: http://dx.doi.org/10.1016/j.jsc.2008.02.017

[18] A. Braeken, C. Wolf, and B. Preneel, “A study of the security of unbalanced oil and vinegar signature schemes,” Cryptology ePrint Archive, Report 2004/222, 2004. [Online]. Available: http://eprint.iacr.org/2004/222

[19] G. V. Bard, *Algebraic Cryptanalysis*, 1st ed. Springer Publishing Company, Incorporated, 2009.

[20] O. Goldreich, *Computational Complexity: A Conceptual Perspective.* Cambridge University Press, June 2008.

[21] Y. Moschovakis, *Notes on Set Theory*, ser. Undergraduate Texts in Mathematics. Springer, 2005.

[22] L. Wolsey and G. Nemhauser, *Integer and Combinatorial Optimization*, ser. Wiley Series in Discrete Mathematics and Optimization. Wiley, 2014. [Online]. Available: https://books.google.com.mx/books?id=MvBjBAAAQBAJ

[23] K. Sakumoto, T. Shirai, and H. Hiwatari, “Public-key identification schemes based on multivariate quadratic polynomials,” *Advances in Cryptology — CRYPTO 2011*, pp. 706–723, 2011.

[24] V. Nachef, J. Patarin, and E. Volte, “Zero-knowledge for multivariate polynomials,” *Progress in Cryptology — LATINCRYPT 2012*, pp. 194–213, 2012. [Online]. Available: http://dx.doi.org/10.1007/978-3-642-33481-8_11

[25] C. Wolf and B. Preneel, “$\mathcal{MQ}^* - IP$: An identity-based identification scheme without number-theoretic assumptions,” Cryptology ePrint Archive, Report 2010/087, 2010. [Online]. Available: http://eprint.iacr.org/2010/087