Zero-Knowledge for Multivariate Polynomials

Valérie Nachef[1], Jacques Patarin[2], Emmanuel Volte[1]

[1] Department of Mathematics, University of Cergy-Pontoise, CNRS UMR 8088
2 avenue Adolphe Chauvin, 95011 Cergy-Pontoise Cedex, France
[2] PRISM, University of Versailles
45 avenue des Etats-Unis, 78035 Versailles Cedex, France
valerie.nachef@u-cergy.fr, emmanuel.volte@u-cergy.fr
jacques.patarin@prism.uvsq.fr

Abstract. In [13] a Zero-Knowledge scheme $ZK(2)$ was designed from a solution of a set of multivariate quadratic equations over a finite field. In this paper we will give two methods to generalize this construction for polynomials of any degree d, i.e. we will design two Zero-Knowledge schemes $ZK(d)$ and $\tilde{ZK}(d)$ from a set of polynomial equations of degree d. We will show that $\tilde{ZK}(d)$ is optimal in term of the number of computations to be performed and that $ZK(d)$ is optimal in term of the number of bits to be send. Moreover this property is still true for all kinds of polynomials: for example if the polynomials are sparse or dense. Finally, we will present two examples of applications: with Brent equations, or with morphisms of polynomials.

Key words: Authentication scheme, Zero-Knowledge, Multivariate polynomials.

1 Introduction

The first Zero-Knowledge schemes were based on the factorization problem (for example Fischer-Micali-Rackoff in 1984, or Fiat-Shamir in 1986) or the Graph Isomorphism Problem. However the factorization problem is not expected to be a NP complete problem (since it is in NP and Co NP) and it has sub-exponential algorithms (such as NFS) and even polynomial algorithms on quantum computers (Shor algorithm). Then, it was proved in 1991 by O. Goldreich, S. Micali and A. Widgerson that any problem of NP has a Zero-Knowledge proof ([4]). But the general construction (cf [4]) of Zero-Knowledge proofs from any problem of NP is usually not very efficient. This is why various Zero-Knowledge schemes have been specifically designed from some well suited and well chosen NP complete schemes based on simple combinatorial problems expected to be exponentially difficult, such as PKP of Adi Shamir [14], PP of David Pointcheval [11] or CLE [16] or SD [15] of Jacques Stern for example. Recently ([13]) such a scheme was designed from the MQ problem, i.e. the problem of finding a solution from a set of multivariate quadratic equations over a finite field. This MQ problem is related to various primitives in cryptography [2, 7–9], and is NP-complete over any finite field ([3, 10]).

In this paper, we will generalize the construction of [13] in order to design a Zero-Knowledge authentication from a solution of any set of multivariate polynomials of degree d over a finite field (i.e. not only $d = 2$). We will describe two schemes extending the results of [13]. The $ZK(d)$ scheme is optimal if we focus on the number of bits to be sent and the $\tilde{ZK}(d)$ scheme is optimal if we consider the number of computations. This is true for any kind of polynomials (dense or sparse). For practical applications the case $d = 3$ (i.e. cubic equations) is particularly important, since from these polynomials we will be able to design Zero-Knowledge schemes based on the (NP-complete) Morphism problem (MP) or from the Brent equations related to the optimal way to solve sets of linear equations (i.e. improvements of the Gauss elimination). We will explain in this paper why these two problems are really interesting for cryptography. We can notice that MP (morphism of Polynomial) is NP hard while IP (isomorphism of Polynomials) is expected not to be NP hard (since it has an Arthur-Merlin game for yes or no answers). We will detail the case $d = 3$. The general schemes $ZK(d)$ and $\tilde{ZK}(d)$ are studied in Appendix A.

2 Zero-Knowledge Protocols and Commitments

In an interactive Protocol, there are two entities: the prover and the verifier. The Prover wants to convince the verifier that she knows a secret. Both interact and at the end, the verifier accepts or refuses. In Zero-Knowledge Protocols there is a possibility of fraud. A cheater will be able to answer some of the questions (but not all of them). The protocol must be designed such that an answer to one of the questions does not give any indication on the secret but if someone is able to answer all the questions then this will reveal the Prover's secret. We will use the following definitions in order to describe the properties that we want to be satisfied by our protocols:

1. The protocol has **perfect correctness** if a legitimate prover is always accepted.
2. The protocol is **statistically zero knowledge** if there exists an efficient simulating algorithm U such that for every feasible Verifier strategy V, the distributions produced by the simulator and the proof protocol are statistically indistinguishable.
3. The protocol is **proof of zero knowledge with error knowledge** α if there is a knowledge extractor K and a polynomial Q such that if p denotes the probability that K finds a valid witness for x using its access to a prover P^* and p_x denotes the probability that P^* convinces the honest verifier on x, and $p_x > \alpha$, then we have $p \geq Q(p_x - \alpha)$.

In our protocols, we will need string commitment schemes. A string commitment function is denoted by Com. The commitment scheme runs in two phases. In the first phase, the sender computes a commitment value $c = Com(s; \rho)$ and sends c to the receiver, where s is the committed string and ρ is a random string. In the second phase, the sender gives (s, ρ) and the receiver verifies if $c = Com(s; \rho)$. we require the two following properties of Com.

1. The commitment scheme is **statistically hiding** if for uniform (x, ρ) and (x', ρ') the distributions $Com(x, \rho)$ and $Com(x', \rho')$ are statistically indistinguishable. This means that the commitment to x reveals (almost) no information on x even to an infinitely powerful Verifier.
2. The commitment scheme is **computationally binding** if the probability to that two different values (x, ρ) and (x', ρ') produce the same $c = Com(x, \rho) = Com(x', \rho')$ is negligible in polynomial time, i.e. the chances to change the committed value after the first phase are very small.

A practical construction of such a commitment is given in [5].

3 Systems of Multivariate equations of degree d

We consider the following function of degree d from $\mathbb{F}_q^n$ to $\mathbb{F}_q^m$:

$$F(x) = (f_1(x), f_2(x), \ldots, f_m(x))$$

where $\forall \ell,\ 1 \leq \ell \leq m$, and $x = (x_1, \ldots, x_n)$:

$$f_\ell(x) = \sum_{1 \leq i_1 \leq \ldots \leq i_d \leq n} \gamma^\ell_{i_1 \ldots i_d} x_{i_1} x_{i_2} \ldots x_{i_d} + \sum_{1 \leq i_1 \leq \ldots \leq i_{d-1} \leq n} \gamma^\ell_{i_1 \ldots i_{d-1}} x_{i_1} x_{i_2} \ldots x_{i_{d-1}}$$

$$+ \ldots + \sum_{1 \leq i_1 \leq i_2 \leq n} \gamma^\ell_{i_1 i_2} x_{i_1} x_{i_2} + \sum_{1 \leq i_1 \leq n} \gamma^\ell_{i_1} x_{i_1}$$

We omit the constant term since we are going to deal with a system of the form $F(v) = s$. Let the function G defined from $(\mathbb{F}_q^n)^d$ to $\mathbb{F}_q^m$ by:

$$G(r_0, r_1, \ldots, r_{d-1}) = \sum_{i=1}^{d} (-1)^{d-i} \sum_{\substack{S \subset \{0, \ldots, d-1\} \\ |S| = i}} F(\sum_{j \in S} r_j)$$

Then G is d-linear (see [18] for example), i.e. we have for all i, $0 \leq i \leq d-1$:

$$G(r_0, \ldots, r_i + r'_i, \ldots r_{d-1}) = G(r_0, \ldots, r_i, \ldots r_{d-1}) + G(r_0, \ldots, r'_i, \ldots r_{d-1})$$

For F, a multivariate function of degree d, we define a binary relation $R_F = \{(v, x) \in \mathbb{F}_q^m \times \mathbb{F}_q^n;\ v = F(x)\}$. The problem is: Given F and $v \in \mathbb{F}_q^m$ find $s \in \mathbb{F}_q^n$ such that $F(s) = v$, i.e. $s \in R_F(v)$.

4 $ZK(3)$ Schemes

We introduce our notation when $d = 3$. We consider the following cubic functions: $F(x) = (f_1(x), f_2(x), \ldots, f_m(x))$ where $\forall \ell,\ 1 \leq \ell \leq m$ and $x = (x_1, \ldots, x_n)$

$$f_\ell(x) = \sum_{1 \leq i \leq j \leq k \leq n} \gamma^\ell_{ijk} x_i x_j x_k + \sum_{1 \leq i \leq j \leq n} \gamma^\ell_{ij} x_i x_j + \sum_{1 \leq i \leq n} \gamma^\ell_i x_i$$

Let

$$G(x,y,z) = F(x+y+z) - F(x+y) - F(x+z) - F(y+z) + F(x) + F(y) + F(z)$$

We have: $G(x,y,z) = (g_1(x,y,z), g_2(x,y,z), \ldots, g_m(x,y,z))$
where $\forall \ell,\ 1 \leq \ell \leq m$, $x = (x_1, \ldots, x_n)$, $y = (y_1, \ldots, y_n)$ and $z = (z_1, \ldots, z_n)$

$$g_\ell(x,y,z) = \sum_{1 \leq i \leq j \leq k \leq n} \gamma_{ijk}^{\ell}(x_i y_j z_k + x_i y_k z_j + x_j y_i z_k + x_j y_k z_i + x_k y_i z_j + x_k y_j z_i)$$

Then G is trilinear.
The problem is: Given F and $v \in \mathbb{F}_q^m$ find $s \in \mathbb{F}_q^n$ such that $F(s) = v$.
The public key is (F, v). The secret is s such that $F(s) = v$. The Prover is going to convince the Verifier of his knowledge of s.

4.1 3-pass scheme

For simplicity, the random string in Com is not written explicitly. If X is a set, $x \in_R X$ means that x is randomly chosen in X with the uniform distribution.

1. The Prover picks up $r_0, r_1, t_0 \in_R \mathbb{F}_q^n$ and $e_0, f_0, h_0 \in_R \mathbb{F}_q^m$. Then she computes

   $$r_2 = s - r_1 - r_0, \quad t_1 = r_0 - t_0$$
   $$e_1 = F(r_0) - e_0, \quad f_1 = F(r_0 + r_1) - f_0, \quad h_1 = F(r_0 + r_2) - h_0$$

   The Prover sends to the Verifier

   $$c_0 = Com(r_1, r_2, G(t_0, r_1, r_2) - e_0 + f_0 + h_0)$$
   $$c_1 = Com(r_1, t_0, e_0, f_0), \quad c_2 = Com(r_1, t_1, e_1, f_1)$$
   $$c_3 = Com(r_2, t_0, e_0, h_0), \quad c_4 = Com(r_2, t_1, e_1, h_1)$$

2. The verifier chooses a query $\mathcal{Q} \in_R \{0, 1, 2, 3\}$ and sends $\mathcal{Q}$ to the prover.
3. (a) If $\mathcal{Q} = 0$ then the Prover sends $(r_0, r_1, t_1, e_1, f_1)$. The Verifier checks if $c_1 = Com(r_1, r_0 - t_1, F(r_0) - e_1, F(r_0 + r_1) - f_1)$, $c_2 = Com(r_1, t_1, e_1, f_1)$
   (b) If $\mathcal{Q} = 1$ then the Prover sends $(r_0, r_2, t_1, e_1, h_1)$. The Verifier checks if $c_3 = Com(r_2, r_0 - t_1, F(r_0) - e_1, F(r_0 + r_2) - h_1)$, $c_4 = Com(r_2, t_1, e_1, h_1)$
   (c) If $\mathcal{Q} = 2$ then the Prover sends $(r_1, r_2, t_1, e_1, f_1, h_1)$. The Verifier checks if $c_0 = Com(r_1, r_2, v - G(t_1, r_1, r_2) + e_1 - f_1 - h_1 - F(r_1 + r_2) + F(r_1) + F(r_2))$, $c_2 = Com(r_1, t_1, e_1, f_1)$, $c_4 = Com(r_2, t_1, e_1, h_1)$
   (d) If $\mathcal{Q} = 3$ then the Prover sends $(r_1, r_2, t_0, e_0, f_0, h_0)$. The Verifier checks if $c_0 = Com(r_1, r_2, G(t_0, r_1, r_2) - e_0 + f_0 + h_0)$, $c_1 = Com(r_1, t_0, e_0, f_0)$, $c_3 = Com(r_2, t_0, e_0, h_0)$

The verifier outputs 1 if the she gets the correct value in the commitments, 0 otherwise.

4.2 Properties of the 3-pass scheme

It is easy to see that the verifier always accepts an interaction with the honest prover. Thus the 3-pass scheme has perfect correctness.

Theorem 1 *The 3-pass protocol is statistically zero knowledge when the commitment scheme Com is statistically hiding.*

Proof. We construct a black-box simulator $\mathcal{S}$ which have oracle access to a cheating verifier $\mathcal{CV}$ takes F and v, and outputs a simulated transcripts with probability 3/4 as follows. The simulator randomly chooses a value $\mathcal{Q}^* \in_R \{0,1,2,3\}$ and vectors $s', r'_0, r'_1, t'_0 \in_R \mathbb{F}_q^n$ and $e'_0, f'_0, h'_0 \in_R \mathbb{F}_q^m$, where $\mathcal{Q}^*$ is a prediction what value the cheating verifier $\mathcal{CV}$ will not choose. Then it computes

$$r'_2 = s' - (r'_0 + r'_1),\ t'_1 \leftarrow r'_0 - t'_0,\ e'_1 \leftarrow F(r'_0) - e'_0$$

Moreover it sets:

1. If $\mathcal{Q}^* = 0$, $f'_1 = v - F(s') + F(r'_0 + r'_1) - f'_0$, else $f'_1 = F(r'_0 + r'_1) - f'_0$.
2. If $\mathcal{Q}^* = 1$, $h'_1 = v - F(s') + F(r'_0 + r'_2) - f'_0$, else $h'_1 = F(r'_0 + r'_2) - h'_0$.
3. If $\mathcal{Q}^* = 3$, $c'_0 = Com(r'_1, r'_2, v - G(t'_1, r'_1, r'_2) - f'_1 - h'_1 + e'_1 - F(r'_1 + r'_2) + F(r'_1) + F(r'_2))$, else $c'_0 = Com(r'_1, r'_2, G(t'_0, r'_1, r'_2) - f'_0 - h_0 + e'_0)$

It also computes:

$$c'_1 = Com(r'_1, t'_0, e'_0, f'_0),\ \ c'_2 = Com(r'_1, t'_1, e'_1, f'_1)$$
$$c'_3 = Com(r'_2, t'_0, e'_0, h'_0),\ \ c'_4 = Com(r'_2, t'_1, e'_1, h'_1)$$

and sends $(c'_0, c'_1, c'_2, c'_3, c'_4)$ to $\mathcal{CV}$.
Receiving a query $\mathcal{Q}$ from $\mathcal{CV}$ the simulator outputs $\perp$ if $\mathcal{Q} = \mathcal{Q}^*$ and stops. If $\mathcal{S}$ does not output $\perp$, it produces a transcript as follows:

- If $\mathcal{Q} = 0$, it outputs $((c'_0, c'_1, c'_2, c'_3, c'_4), 0, (r'_0, r'_1, t'_1, e'_1, f'_1))$
- If $\mathcal{Q} = 1$, it outputs $((c'_0, c'_1, c'_2, c'_3, c'_4), 1, (r'_0, r'_2, t'_1, e'_1, h'_1))$
- If $\mathcal{Q} = 2$, it outputs $((c'_0, c'_1, c'_2, c'_3, c'_4), 2, (r'_1, r'_2, t'_1, e'_1, f'_1, h'_1))$
- If $\mathcal{Q} = 3$, it outputs $((c'_0, c'_1, c'_2, c'_3, c'_4), 3, (r'_1, r'_2, t'_0, e'_0, f'_0, h'_0))$

We can check that if $\mathcal{S}$ does not output $\perp$, the transcript is accepted. For example, we consider the case where $\mathcal{Q}^* = 0$ and $\mathcal{Q} = 2$. The output is $((c'_0, c'_1, c'_2, c'_3, c'_4), 2, (r'_1, r'_2, t'_1, e'_1, f'_1, h'_1))$. Thus, we have the right values for c'_2 and c'_4. Now, c'_0 is computed as follows: $c'_0 = Com(r'_1, r'_2, v - G(t'_1, r'_1, r'_2) + e'_1 - f'_1 - h'_1 - F(r'_1 + r'_2) + F(r'_1) + F(r'_2))$. Here $f'_1 = v - F(s') + F(r'_0 + r'_1) - f'_0$. Thus we obtain $c'_0 = Com(r'_1, r'_2, G(t'_0, r'_1, r'_2) - f'_0 - h_0 + e'_0)$ and the transcript is accepted. The other cases are checked similarly.
We now show that the distribution of the output $\mathcal{S}$ is statistically close to the distribution of a real transcript since the commitment is statistically hiding. A real transcript between the the legitimate prover P and a cheating verifier $\mathcal{CV}$ on (F, v, s) is denoted by $\langle P(s), \mathcal{CV} \rangle (F, v)$. The simulator output is denoted by

$\langle \mathcal{S}, \mathcal{CV} \rangle(F, v)$. We analyze the output distribution.
First first we consider the case where $\mathcal{Q} = 0$. Then

$$\langle P(s), \mathcal{CV} \rangle(F, v) = ((c_0, c_1, c_2, c_3, c_4), 0, (r_0, r_1, t_1, e_1, f_1))$$

$$\langle \mathcal{S}, \mathcal{CV} \rangle(F, v) = ((c'_0, c'_1, c'_2, c'_3, c'_4), 0, (r'_0, r'_1, t'_1, e'_1, f'_1))$$

Assume that $(r'_0, r'_1, t'_0, e'_0, f'_0) = (r_0, r_1, t_0, e_0, f_0)$. Then we obtain $t'_1 = t_1$, $e'_1 = e_1$, $f'_1 = f_1$ and $c'_1 = c_1$, $c'_2 = c_2$ in all cases $\mathcal{Q}^* = 1, 2, 3$.
The second case is when $\mathcal{Q} = 1$. Then

$$\langle P(s), \mathcal{CV} \rangle(F, v) = ((c_0, c_1, c_2, c_3, c_4), 1, (r_0, r_2, t_1, e_1, h_1))$$

$$\langle \mathcal{S}, \mathcal{CV} \rangle(F, v) = ((c'_0, c'_1, c'_1, c'_3, c'_4), 1, (r'_0, r'_2, t'_1, e'_1, h'_1))$$

This case is very similar to the previous one. We get $t'_1 = t_1$, $e'_1 = e_1$, $h'_1 = h_1$ and $c'_3 = c_3$, $c'_4 = c_4$ in all cases $\mathcal{Q}^* = 0, 2, 3$.
The third case is $\mathcal{Q} = 2$. Then

$$\langle P(s), \mathcal{CV} \rangle(F, v) = ((c_0, c_1, c_2, c_3, c_4), 2, (r_1, r_2, t_1, e_1, f_1, h_1))$$

$$\langle \mathcal{S}, \mathcal{CV} \rangle(F, v) = ((c'_0, c'_1, c'_2, c'_3, c'_4), 2, (r'_1, r'_2, t'_1, e'_1, f'_1, h'_1))$$

When $\mathcal{Q}^* = 0$, assume that $(r'_0, r'_1, t'_0, e'_0, f'_0, h'_0) = (r_0 + s' - s, r_1, t_0 + s' - s, e_0 - F(r_0) + F(r_0 + s' - s), f_0 - F(r_0 + r_1) + v - F(s') + F(r_0 + r_1 + s' - s), h_0 - F(r_0 + r_2) + F(r_0 + r_2 + s' - s))$. When $\mathcal{Q}^* = 1$, assume that $(r'_0, r'_1, t'_0, e'_0, f'_0, h'_0) = (r_0 + s' - s, r_1, t_0 + s' - s, e_0 - F(r_0) + F(r_0 + s' - s), f_0 - F(r_0 + r_1) + F(r_0 + r_1 + s' - s), h_0 - F(r_0 + r_2) + v - F(s') + F(r_0 + r_2 + s' - s))$. When $\mathcal{Q}^* = 1$, assume that $(r'_0, r'_1, t'_0, e'_0, f'_0, h'_0) = (r_0 + s' - s, r_1, t_0 + s' - s, e_0 - F(r_0) + F(r_0 + s' - s), f_0 - F(r_0 + r_1) + F(r_0 + r_1 + s' - s), h_0 - F(r_0 + r_2) + F(r_0 + r_2 + s' - s))$. We can check that for all these cases we obtain: $r'_2 = r_2$, $t_1 = t'_1$, $e'_1 = e_1$, $f'_1 = f_1$, $h'_1 = h_1$ and then $c'_0 = c_0$, $c'_2 = c_2$, $c'_4 = c_4$.
The last case is $\mathcal{Q} = 3$. Then

$$\langle P(s), \mathcal{CV} \rangle(F, v) = ((c_0, c_1, c_2, c_3, c_4), 3, (r_1, r_2, t_0, e_0, f_0, h_3))$$

$$\langle \mathcal{S}, \mathcal{CV} \rangle(F, v) = ((c'_0, c'_1, c'_2, c'_3, c'_4), 3, (r'_1, r'_2, t'_0, e'_0, f'_0, h'_0))$$

Assume that $(r'_0, r'_1, t'_0, e'_0, f'_0, h'_0) = (r_0 s' - s, r_1, t_0, e_0, f_0, h_0)$, then we obtain $r'_2 = r_2$, $c'_0 = c_0$, $c'_1 = c_1$ and $c'_3 = c_3$. Since the commitment is statistically hiding, we get that when $\mathcal{S}$ does not output $\perp$, the distribution of the output of $\mathcal{S}$ is statistically close to the distribution of the real transcript. □

Theorem 2 *The 3-pass protocol is proof of zero knowledge with zero knowledge error* $3/4$ *when the commitment scheme Com is computationally binding.*

Proof. Suppose that there exists a false prover C that can answer all the questions. Then either C will compute a collision for Com or will extract a solution for (F, v). Let $((c_0, c_1, c_2, c_3, c_4), \mathcal{Q}_0, Rsp_0)$, $((c_0, c_1, c_2, c_3, c_4), \mathcal{Q}_1, Rsp_1)$, $((c_0, c_1,$

$c_2, c_3, c_4), \mathcal{Q}_2, Rsp_2)$, $((c_0, c_1, c_2, c_3, c_4), \mathcal{Q}_3, Rsp_3)$, be four transcripts such that $\mathcal{Q}_i = i$ and all the responses are accepted. Consider the situation where the responses are parsed as $Rsp_0 = (r_0^{(0)}, r_1^{(0)}, t_1^{(0)}, e_1^{(0)}, f_1^{(0)})$, $Rsp_1 = (r_0^{(1)}, r_2^{(1)}, t_1^{(1)}, e_1^{(1)}, h_1^{(1)})$, $Rsp_2 = (r_1^{(2)}, r_2^{(2)}, t_1^{(2)}, e_1^{(2)}, f_1^{(2)}, h_1^{(2)})$, $Rsp_3 = (r_1^{(3)}, r_2^{(3)}, t_0, e_0, f_0, h_0)$. We obtain:

$$\begin{aligned} c_0 &= Com(\, r_1^{(2)}, r_2^{(2)}, v - G(t_1^{(2)}, r_1^{(2)}, r_2^{(2)}) - f_1^{(2)} - h_1^{(2)} + e_1^{(2)} \\ &\qquad\qquad -F(r_1^{(2)} + r_2^{(0)}) + F(r_1^{(2)}) + F(r_2^{(2)}) \quad) \\ &= Com(\, r_1^{(3)}, r_2^{(3)}, G(t_0, r_1^{(3)}, r_2^{(3)}) + f_0 + h_0 - e_0 \quad) \end{aligned} \tag{1}$$

$$\begin{aligned} c_1 &= Com(\, r_1^{(0)}, r_0^{(0)} - t_1^{(0)}, F(r_0^{(0)}) - e_1^{(0)}, F(r_0^{(0)} + r_1^{(0)}) - f_1^{(0)} \,) \\ &= Com(\, r_1^{(3)}, t_0, \quad e_0, \quad f_0 \quad) \end{aligned} \tag{2}$$

$$c_2 = Com(r_1^{(0)}, t_1^{(0)}, e_1^{(0)}, f_1^{(0)}) = Com(r_1^{(2)}, t_1^{(2)}, e_1^{(2)}, f_1^{(2)}) \tag{3}$$

$$\begin{aligned} c_3 &= Com(\, r_2^{(1)}, r_0^{(1)} - t_1^{(1)}, F(r_0^{(1)}) - e_1^{(1)}, F(r_0^{(1)} + r_2^{(1)}) - h_1^{(1)} \,) \\ &= Com(\, r_2^{(3)}, t_0, \quad e_0, \quad h_0 \quad) \end{aligned} \tag{4}$$

$$c_4 = Com(r_2^{(1)}, t_1^{(1)}, e_1^{(1)}, h_1^{(1)}) = Com(r_2^{(2)}, t_1^{(2)}, e_1^{(2)}, h_1^{(2)}) \tag{5}$$

If two tuples of the arguments of Com are are distinct on either of the above equations, then we have a collision for Com. Otherwise, these equalities give:

$$h_1^{(1)} \overset{(5)}{=} h_1^{(2)}, r_1^{(0)} \overset{(2)}{=} r_1^{(3)} \overset{(1)}{=} r_1^{(2)}, t_1^{(0)} \overset{(3)}{=} t_1^{(1)} \overset{(5)}{=} t_1^{(2)}, r_0^{(0)} \overset{(3,4)}{=} r_0^{(1)}$$

$$\text{and} \quad r_2^{(2)} \overset{(1)}{=} r_2^{(3)} \overset{(4)}{=} r_2^{(1)}, e_1^{(2)} \overset{(3)}{=} e_1^{(0)} \overset{(2,4)}{=} e_1^{(1)}, f_1^{(0)} \overset{(3)}{=} f_1^{(2)}$$

So, all upper scripts are useless and from (1) we have:

$$v = G(t_1 + t_0, r_1, r_2) + f_1 + h_1 - e_1 + F(r_1 + r_2) - F(r_1) - F(r_2) + f_0 + h_0 - e_0$$

Then, from (2) and (4) we have $r_0 = t_0 + t_1$, $F(r_0) = e_0 + e_1$, $F(r_0 + r_1) = f_0 + f_1$ and $F(r_0 + r_2) = h_0 + h_1$, so if we replace these values in the previous equality, we obtain: $v = G(r_0, r_1, r_2) + F(r_0 + r_1) + F(r_0 + r_2) + F(r_1 + r_2) - F(r_0) - F(r_1) - F(r_2) = F(r_0 + r_1 + r_2)$ This means that a solution $r_0 + r_1 + r_2$ for v is extracted.

Let p_s be the probability that P^* convinces the honest verifier on s and p the probability that all the 3 transcripts are accepted. Suppose that $p_s > \frac{3}{4}$. Depending on the fact that the 4 transcripts are accepted or not, we get $p_s \leq 1 \cdot p + \frac{3}{4}(1 - p)$. This implies that $p \geq 4(p_s - \frac{3}{4})$. Thus the proof of Theorem 2 is complete. □

4.3 Computations in the 3-pass scheme

We give the maximum number of computations that have to be done either by the prover or by the receiver in the case of $\mathbb{F}_2$. We must calculate the number of computations for F and for G. Moreover we see that F is computed at most 3 times and G is computed just one time. We only count multiplications. In $\mathbb{F}_2$, we have: $x_i^3 = x_i^2 = x_i$. We can write:

$$f_\ell(x) = \sum_{i=1}^{n} x_i \big[\gamma_i^\ell + \gamma_{ii}^\ell + \gamma_{iii}^\ell + \sum_{j=i+1}^{n} x_j \big(\gamma_{ij}^\ell + \gamma_{ijj}^\ell + \sum_{k=j+1}^{n} \gamma_{ijk}^\ell x_k\big)\big]$$

Let M denotes the number of multiplications needed to compute F. Using the above expression for F, we obtain $M \simeq \frac{n^3}{6} m$. In the following table, we give the characteristics of the scheme $ZK(3)$ and the values that we obtain when we choose $n = 84$, $m = 80$, in order to have 80-bit security (cf. [2]). Moreover if we want an impersonation probability less than 2^{-30}, we need to perform at least 73 rounds. R stands for the number of rounds and C for the maximum number of computations that have to be done either by the prover or by the receiver. The values are given in Table 1.

Remarks.

Table 1. $ZK(3)$ Scheme

	Formulas	Parameters for 2^{80} security
Public key (bit)	m	80
Secret key (bit)	n	84
M	$\frac{n^3}{6} \times m$	7902720
Communication (bit)	$(4 \times 160 + 2 + 3n + 2m) \times R$ or $(3 \times 160 + 2 + 3n + 3m) \times R$	76942 or 71102
Number of multiplications	C=9MR	2^{33}

We may need less computations. It depends on the number of non zero coefficients. This is the case for Brent equations as explain in Section 8.
It is also possible to design a 5-pass scheme. This in done in the extended version of this paper.

5 The $\tilde{ZK}(3)$ Scheme.

In this section, we propose another scheme inspired from [13]. The idea is to transform the cubic system into a quadratic one and the to use the scheme given in [13]. As we will see, the number of computations is smaller, but the number of communication bits is more important.

We investigate the transformation of a system with cubic equations to a system with quadratic equations. We will introduce new variables. Once we have obtained a system of equations with quadratic polynomials, we can apply the identification scheme of [13]. We will calculate the number of multiplications in this case. In our system, we have $F(x) = (f_1(x), f_2(x), \ldots, f_m(x))$ where $\forall \ell,\ 1 \leq \ell \leq m$,

$$f_\ell(x) = \sum_{1 \leq i \leq j \leq k \leq n} \gamma_{ijk}^\ell x_i x_j x_k + \sum_{1 \leq i \leq j \leq n} \gamma_{ij}^\ell x_i x_j + \sum_{1 \leq i \leq n} \gamma_i^\ell x_i$$

and $x = (x_1, \ldots, x_n)$. We introduce the new variables $\forall i, j,\ 1 \leq i \leq j \leq n,\ X_{ij} = x_i x_j$. The number of new variables is $\frac{n(n-1)}{2}$, if $q = 2$, and $\frac{n(n+1)}{2}$, if $q \neq 2$. In our new system, we have

$$\tilde{F} = (\tilde{f}_1, \ldots, \tilde{f}_m, (\tilde{f}_{ij})_{1 \leq i \leq j \leq n})$$

where for $\tilde{x} = (x_1, \ldots, x_n, (X_{ij})_{1 \leq i \leq j \leq n})$ and $1 \leq \ell \leq m$,

$$f_\ell(\tilde{x}) = \sum_{1 \leq i \leq j \leq k \leq n} \gamma_{ijk}^\ell X_{ij} x_k + \sum_{1 \leq i \leq j \leq n} \gamma_{ij}^\ell X_{ij} + \sum_{1 \leq i \leq n} \gamma_i^\ell x_i$$

and for $1 \leq i \leq j \leq n$, $f_{ij}(\tilde{x}) = X_{ij} - x_i x_j$. Here the number of variables is $\tilde{n} \simeq n + \frac{n^2}{2}$ and the number of equations is $\tilde{m} = m + \frac{n^2}{2}$. As before, M denotes the number of multiplications needed to compute F. $\tilde{M}$ denotes the number of multiplications for the computation of $\tilde{F}$. We choose $q = 2$. Then $\tilde{M} = M + \frac{n(n-1)}{2}$. Thus $\tilde{M} \simeq M \simeq \frac{n^3}{6} m$. $\tilde{C}$ stands for the maximum number of computations that have to be done either by the prover or by the receiver. If $\tilde{R}$ denotes the number of rounds performed in order to have an impersonation probability less than 2^{-30}, then $\tilde{R} = 52$ (cf. [13]). The following table gives the characteristics of the $\tilde{ZK}(3)$ Scheme and the values we get when $n = 84$ and $m = 80$.

Table 2. $\tilde{ZK}(3)$ Scheme

	Formulas	Parameters for 2^{80} security
Public key (bit)	$\tilde{m}$	3483
Secret key (bit)	n	84
M	$\frac{n^3}{6} \times m$	7902720
Communication (bit)	$(2 \times 160 + 2 + 2\tilde{n} + \tilde{m}) \times \tilde{R}$	560508
Number of multiplications	$\tilde{C} = 3\tilde{M}\tilde{R}$	2^{31}

6 $ZK(d)$ and $\tilde{ZK}(d)$ Schemes for any d

The schemes described in Section 4 and Section 5 can be generalized for any d. We obtain the $ZK(d)$ and $\tilde{ZK}(d)$ schemes which are described in Appendix A. In

Table 3 we provide the features of both schemes. Here we have an impersonation probability less than 2^{-30}.

Table 3. $ZK(d)$ and $\tilde{ZK}(d)$ Schemes

	$ZK(d)$ scheme	$\tilde{ZK}(d)$ scheme
Public key (bits)	m	$\tilde{m} \simeq m + n^{\lceil \frac{d}{2} \rceil}/\lceil \frac{d}{2} \rceil !$
Secret key (bits)	n	n
M	$\frac{n^d}{d!} \times m$	$\frac{n^d}{d!} \times m$
Rounds	$R = \lceil \frac{30 \ln(2)}{\ln(1+1/d)} \rceil$	$\tilde{R} = 52$
Number of Communication bits	$(5dn + \lceil \frac{\ln d}{2} \rceil + 2^{d-1} - 1) \cdot R$ or $(3dn + \lceil \frac{\ln d}{2} \rceil + 2^d - 2) \cdot R$	$(322 + 2n + m + 3n^{\lceil \frac{d}{2} \rceil}/\lceil \frac{d}{2} \rceil !) \cdot \tilde{R}$
Multiplications	$C = 9(2^{d-1} - 1 + d!)MR$	$\tilde{C} = 3\tilde{M}\tilde{R}$

There are more multiplications in $ZK(d)$ and more communication bits in $\tilde{ZK}(d)$.

7 Relations between the number of computations and the number of coefficients

In the previous computations, we have supposed that we have the maximum number of coefficients. Then we obtained that in both cases, $M \simeq \tilde{M} \simeq \frac{n^d}{d!}$. Then the total number of multiplications is a function of M or $\tilde{M}$ and the number of rounds. For sparse systems, M or $\tilde{M}$ will be smaller but we will still have the same relations between C, M and R (and similarly $\tilde{C}$, $\tilde{M}$ and $\tilde{R}$). Here again, we can see that there are more variables and more communications bits in the $\tilde{ZK}(d)$ schemes and more computations in the $ZK(d)$ schemes.

More precisely with the $ZK(d)$ scheme, we have:

$$f_\ell((x_1, x_2, \ldots, x_n) = \sum_{(i_1,\ldots,i_d) \in S_d^\ell} \gamma^\ell_{i_1 \ldots i_d} x_{i_1} x_{i_2} \ldots x_{i_d} +$$

$$\sum_{i_i,\ldots,i_{d-1} \in S_{d-1}^\ell} \gamma^\ell_{i_1 \ldots i_{d-1}} x_{i_1} x_{i_2} \ldots x_{i_{d-1}}$$

$$+ \ldots + \sum_{i_i, i_2 \in S_2^\ell} \gamma^\ell_{i_1 i_2} x_{i_1} x_{i_2} + \sum_{i_i \in S_2^\ell} \gamma^\ell_{i_1} x_{i_1}$$

The number of multiplications for f_ℓ is given by

$$d|S_d^\ell| + (d-1)|S_{d-1}^\ell| + (d-2)|S_{d-2}^\ell| + \ldots + 2|S_2^\ell| + |S_1^\ell|$$

and for F the number M of multiplications is

$$M = \sum_{\ell=1}^{m} \Big[d|S_d^\ell| + (d-1)|S_{d-1}^\ell| + (d-2)|S_{d-2}^\ell| + \ldots + 2|S_2^\ell| + |S_1^\ell| \Big]$$

Moreover, F is computed at most $2^{d-1} - 1$ times during the process. For g_l we have $\big(d!(d-1)+1\big)|S_d^\ell|$ multiplications and for G, $\sum_{\ell=1}^{m} \big(d!(d-1)+1\big)|S_d^\ell|$ multiplications and G is computed one time. Finally, for one round, the number of multiplications is given by

$$\Big(\sum_{\ell=1}^{m} \Big[\Big(d(2^{d-1}-1) + d!(d-1) + 1 \Big) |S_d^\ell| + (d-1)|S_{d-1}^\ell| + (d-2)|S_{d-2}^\ell| + \ldots + 2|S_2^\ell| + |S_1^\ell| \Big] \Big) \quad (\sharp)$$

and then we have to multiply by the number of rounds R to get C.

For the $\tilde{Z}K(d)$ scheme, we have $\tilde{M} = M + n^{\lceil \frac{d}{2} \rceil} / \lceil \frac{d}{2} \rceil !$.
Then $\tilde{C} = 3\tilde{M}\tilde{R}$.

8 The particular case of Brent equations

In this section, we introduce the Brent equations. Suppose we want to multiply two $N \times N$ matrices. The naive method will use N^2 multiplications. In fact for $N = 2$ Strassen's algorithm ([17]) requires 7 multiplications instead of 8 multiplications and Laderman showed that when $N = 3$ it is possible to use 23 multiplications instead of $3^3 = 27$ ([6]). For $N = 2$, 7 is the least number we can obtain. For $N = 3$, it is not known if 23 is the least number in the non-commutative case. In [1], it is shown that obtaining the product of two matrices $N \times N$ can be done using s multiplications is equivalent to solve the following system of cubic equations: $\sum_{k=1}^{s} \gamma_{ijk}\alpha_{abk}\beta_{cdk} = \delta_{bc}\delta_{ia}\delta_{jd} \qquad a, b, c, d, i, j \in \{1, 2, \ldots, n\}$

Here we have $n = 3sN^2$, $m = N^6$. If we use formula $(\sharp)$, we obtain that the number of multiplications is $22 \times s \times N^6$. It is also interesting to design an authentication public key cryptographic scheme as close as possible to Brent equations. In order to do this, we choose a system similar to Brent equations but for which we know a particular solution. It is possible to proceed as follows:

1. We consider the finite field $\mathbb{Z}/2\mathbb{Z}$
2. We take the Brent equations with $s = 22$ in order obtain an open problem in the non-commutative case
3. We pick randomly variables α, β, γ in $\mathbb{Z}/2\mathbb{Z}$
4. We deduce the corresponding constants
5. We then use either $ZK(3)$ or $\tilde{ZK}(3)$ to have a zero-knowledge protocol.

9 Morphisms of polynomials and systems of cubic equations

9.1 The MP Problem

The IP problem (Isomorphism of Polynomials) has been used to construct pubic key schemes (cf [9]). On one hand, this is not a NP-complete problem since it admits an Arthur-Merlin game when the answer is yes and when the answer is no). On the other hand, the MP problem(morphisms of polynomials) where matrices are not supposed to be invertible is proved to be NP-complete ([3, 10]) and thus is much more difficult. So it is interesting to design a public key authentication scheme based on MP. We explain briefly below how it is possible to construct such a scheme by transforming MP very efficiently into a system of equations of degree 3 and then applying our $ZK(3)$ or $\tilde{ZK}(3)$ protocols.

9.2 From MP to polynomials of degree 3

We consider the two following systems:

$$(\mathcal{A}) \quad c_k = \sum_{1\leq i\leq n,\, 1\leq j\leq n} \gamma_{ij}^k a_i a_j + \sum_{i=1}^{n} \mu_i^k a_i, \quad 1 \leq k \leq u$$

$$(\mathcal{B}) \quad z_t = \sum_{1\leq i\leq p,\, 1\leq j\leq p} \alpha_{ij}^t x_i x_j + \sum_{i=1}^{p} \beta_i^t x_i, \quad 1 \leq t \leq v$$

We want to find 2 matrices $M = (m_{rs})_{\substack{1\leq r\leq v\\ 1\leq s\leq u}}$ and $H = (h_{df})_{\substack{1\leq d\leq n\\ 1\leq f\leq p}}$ such that

$$M \begin{pmatrix} c_1 \\ \vdots \\ c_u \end{pmatrix} = \begin{pmatrix} z_1 \\ \vdots \\ z_v \end{pmatrix} \quad \text{and} \quad H \begin{pmatrix} x_1 \\ \vdots \\ x_p \end{pmatrix} = \begin{pmatrix} a_1 \\ \vdots \\ a_n \end{pmatrix}$$

For all $t,\ 1 \leq t \leq v$, on one hand, we have:

$z_t = \sum_{s=1}^{u} m_{ts} \Big(\sum_{\substack{1\leq i\leq n\\ 1\leq j\leq n}} \gamma_{ij}^s a_i a_j + \sum_{i=1}^{n} \mu_i^s a_i \Big)$

$z_t = \sum_{s=1}^{u} m_{ts} \Big(\sum_{\substack{1\leq i\leq n\\ 1\leq j\leq n}} \gamma_{ij}^s \big(\sum_{f=1}^{p} h_{if} x_f\big)\big(\sum_{b=1}^{p} h_{jb} x_b\big) + \sum_{i=1}^{n} \mu_i^s \big(\sum_{f=1}^{p} h_{if} x_f\big)\Big)$

$z_t = \sum_{f=1}^{p}\sum_{b=1}^{p} \Big[\sum_{s=1}^{u} \sum_{\substack{1\leq i\leq n\\ 1\leq j\leq n}} \gamma_{ij}^s m_{ts} h_{if} h_{jb}\Big] x_f x_b + \sum_{f=1}^{p} \Big[\sum_{s=1}^{u}\sum_{i=1}^{n} \mu_i^s m_{ts} h_{if}\Big] x_f$. On the other hand, we have: $z_t = \sum_{\substack{1\leq i\leq p\\ 1\leq j\leq p}} \alpha_{ij}^t x_i x_j + \sum_{i=1}^{p} \beta_i^t x_i$. This gives $\forall t\, 1 \leq t \leq$

$v,\ \forall f, 1 \leq f \leq p,\ \forall b, 1 \leq b \leq p,\ \alpha_{fb}^t + \beta_f^t = \sum_{s=1}^{u} \sum_{\substack{1 \leq i \leq n \\ 1 \leq j \leq n}} \gamma_{ij}^s m_{ts} h_{if} h_{jb} + \sum_{s=1}^{u} \sum_{i=1}^{n} \mu_i^s m_{ts} h_{if}$. Thus we obtain vp^2 cubic equations and $np + vu$ unknowns.

10 Conclusion

In [13], a very efficient zero-knowledge proof based on the MQ problem (multivariate quadratic polynomials) is given. In this paper we proved that this construction can be generalized to polynomials of degree d for any $d \geq 3$. We studied several constructions and we presented here the two most efficient ones denoted by $ZK(d)$ and $\tilde{ZK}(d)$. $ZK(d)$ is quasi optimal in term of communication bits and $\tilde{ZK}(d)$ is quasi optimal in term of number of computations. This result is true for dense or sparse systems. We also presented two important specific problems (Brent equations and morphisms of polynomials) that can be transformed into efficient public key schemes using $ZK(d)$ and $\tilde{ZK}(d)$.

References

1. Gregory V. Bard. New Practical Strassen-like Approximate Matrix-multiplication Algorithms found via solving a system of cubic equations. *http://www.users.math.umd.edu/~bardg/*.
2. Come Berbain, Henri Gilbert, and Jacques Patarin. QUAD: A Practical Cipher with Provable Security . In Serge Vaudenay, editor, *Advances in Cryptology – EUROCRYPT 2006*, volume 4404 of *Lecture Notes in Computer Science*, pages 109–128. Springer-Verlag, 2006.
3. Michael R. Garey and David S. Johnson. *Computers and Intractability; A Guide to the Theory of NP-Completness*. W.H. Freeman and Co, 1979.
4. Oded Goldreich, Silvio Micali, and Avi Wigderson. Proofs that yield nothing but their validity or all languages in np have zero-knowledge proof systems. *J. ACM*, 38:690–728, July 1991.
5. Shai Halevi and Silvio Micali. Practical and Provably-Secure Commitment Schemes from Collision-Free Hashing . In Neal Koblitz, editor, *Advances in Cryptology – CRYPTO 1996*, volume 1109 of *Lecture Notes in Computer Science*, pages 201–215. Springer-Verlag, 1996.
6. J.Laderman. A noncomuutative algorithm for multiplying 3 x3 matrices using 23 multiplication. *Bulletin of the American Mathematical Society*, 82:126–128, 1976.
7. Aviad Kipnis, Jacques Patarin, and Louis Goubin. Unbalanced Oil and Vinegar Signature Schemes. In Jacques Stern, editor, *Advances in Cryptology – EUROCRYPT 1999*, volume 1492 of *Lecture Notes in Computer Science*, pages 206–222. Springer-Verlag, 1999.
8. Tsutomu Matsumo and Hideki Imai. Public Quadratic polynomial-Tuples for Efficient Signature-Verification and Message-Encryption . In C.G. Gunther, editor, *Advances in Cryptology – EUROCRYPT 1988*, volume 330 of *Lecture Notes in Computer Science*, pages 419–453. Springer-Verlag, 1988.

9. Jacques Patarin. Hidden Fields Equations (HFE) and Isomorphims of Polynomials (IP): Two New Families of Asymmetric Algorithms. In Ueli Maurer, editor, *Advances in Cryptology – EUROCRYPT 1996*, volume 1070 of *Lecture Notes in Computer Science*, pages 33–48. Springer-Verlag, 1996.
10. Jacques Patarin and Louis Goubin. Trapdoor one-way permutations and multivariate polynomials. In Yongfei Han, Tatsuaki Okamoto, and Sihan Qing, editors, *ICICS*, volume 1334 of *LNCS*, pages 356–368. Springer, 1997.
11. David Poincheval. A New Identification Scheme based on the Perceptrons Problem. In Alfredo de Santis, editor, *Advances in Cryptology – EUROCRYPT 1995*, volume 950 of *Lecture Notes in Computer Science*, pages 319–328. Springer-Verlag, 1995.
12. Koichi Sakumuto. Public-Key Identification Schemes based on Multivariate Cubic Polynomials. In Marc Fischlin, Johannes Buchmann, and Mark Manulis, editors, *Public Key Cryptography– PKC 2012*, volume 7293 of *Lecture Notes in Computer Science*, pages 172–189. Springer-Verlag, 2012.
13. Koichi Sakumuto, Taizo Shirai, and Harunaga Hiwatari. Public-Key Identification Schemes based on Multivariate Quadratic Polynomials. In Philip Rogaway, editor, *Advances in Cryptology – CRYPTO 2011*, volume 6841 of *Lecture Notes in Computer Science*, pages 706–723. Springer-Verlag, 2011.
14. Adi Shamir. An Efficient Identification Scheme Based on Permuted Kernels (Extended Abstract). In Gilles Brassard, editor, *Advances in Cryptology – CRYPTO 1989*, volume 435 of *LNCS*, pages 606–609. Springer-Verlag, 1989.
15. Jacques Stern. A New Identification Scheme based on Syndrome Decoding. In Douglas R. Stinson, editor, *Advances in Cryptology – CRYPTO 1993*, volume 773 of *Lecture Notes in Computer Science*, pages 13–21. Springer, 1993.
16. Jacques Stern. Designing Identification Schemes with Keys of Short Size. In Yvo G. Desmedt, editor, *Advances in Cryptology – CRYPTO 1994*, volume 839 of *Lecture Notes in Computer Science*, pages 164–173. Springer, 1994.
17. Volker Strassen. Gaussion Elimination is not optimal. *Numerische Mathematik*, 13(3):354–356, 1969.
18. Erik G.F. Thomas. A Polarizatin Identity for Multilinear Maps. *University of Groningen - Preprint*, (1997).

A $ZK(d)$ and $Z\tilde{K}(d)$ Schemes for any d

A.1 The $ZK(d)$ Scheme

We will design a 3-pass scheme
We consider the following function of degree d from $\mathbb{F}_q^n$ to $\mathbb{F}_q^m$:

$$F(x) = (f_1(x), f_2(x), \ldots, f_m(x))$$

where $\forall \ell,\ 1 \leq \ell \leq m$,

$$f_\ell(x) = \sum_{1 \leq i_i \leq \ldots \leq i_d \leq n} \gamma^\ell_{i_1 \ldots i_d} x_{i_1} x_{i_2} \ldots x_{i_d} +$$

$$\sum_{1 \leq i_i \leq \ldots \leq i_{d-1} \leq n} \gamma^\ell_{i_1 \ldots i_{d-1}} x_{i_1} x_{i_2} \ldots x_{i_{d-1}} + \ldots + \sum_{1 \leq i_i \leq i_2 \leq n} \gamma^\ell_{i_1 i_2} x_{i_1} x_{i_2} + \sum_{1 \leq i_i \leq n} \gamma^\ell_{i_1} x_{i_1}$$

and $x = (x_1, \ldots, x_n)$. We omit the constant term. Let

$$G(r_0, r_1, \ldots, r_{d-1}) = \sum_{i=1}^{d} (-1)^{d-i} \sum_{\substack{S \subset \{0,\ldots,d-1\} \\ |S|=i}} F(\sum_{j \in S} r_j)$$

Then G is d-linear.

The problem is: Given F and $v \in \mathbb{F}_q^m$ find $s \in \mathbb{F}_q^n$ such that $F(s) = v$
The public key is (F, v). The secret is s such that $F(s) = v$.

1. The Prover picks up at random $r_0, r_1, \ldots, r_{d-2}, t_0 \in_R \mathbb{F}_q^n$, $f_0 \in_R \mathbb{F}_q^m$, and $\forall p,\ 1 \leq p \leq d-2,\ \forall i_1, \ldots i_p,\ 1 \leq i_1 < i_2 < \ldots i_p \leq d-1,\ f_0^{i_1 \ldots i_p},\ \in_R \mathbb{F}_q^m$. Then she computes

$$\begin{array}{ll} r_{d-1} & = s - \sum_{i=1}^{d-2} r_i \\ t_1 & = r_0 - t_0 \\ f_1 & = F(r_0) - f_0 \end{array}$$

and

$$\forall p,\ 1 \leq p \leq d-2,\ \forall i_1, \ldots i_p,\ 1 \leq i_1 < i_2 < \ldots i_p \leq d-1$$

$$f_1^{i_1 \ldots i_p} = F(r_0 + r_{i_1} + \ldots + r_{i_p}) - f_0^{i_1 \ldots i_p}$$

Then the Prover sends to the Verifier

$$c_0 \leftarrow Com\Big(r_1, \ldots, r_{d-1}, G(t_0, r_1, \ldots, r_{d-1}) +$$

$$\sum_{p=1}^{d-2} (-1)^{d-p} \sum_{1 \leq i_1 < \ldots < i_p \leq d-1} f_0^{i_1 \ldots i_p} + (-1)^d f_0\Big)$$

$$\forall i,\ 1 \leq i \leq d-1$$

$$c_{2i-1} \leftarrow Com\Big(r_1, \ldots, r_{i-1}, r_{i+1}, \ldots, r_{d-1}, t_0, f_0,$$

$$\forall p,\ 1 \leq p \leq d-2,\ \forall i_1, \ldots i_p,\ 1 \leq i_1 < i_2 < \ldots i_p \leq d-1,$$

$$\text{such that } \forall j,\ i_j \neq i,\ \ f_0^{i_1 \ldots i_p}\Big)$$

$$c_{2i} \leftarrow Com\Big(r_1, \ldots, r_{i-1}, r_{i+1}, \ldots, r_{d-1}, t_1, f_1,$$

$$\forall p,\ 1 \leq p \leq d-2,\ \forall i_1, \ldots i_p,\ 1 \leq i_1 < i_2 < \ldots i_p \leq d-1,$$

$$\text{such that } \forall j\ i_j \neq i,\ \ f_1^{i_1 \ldots i_p}\Big)$$

The verifier chooses a query $\mathcal{Q} \in_R \{0, 1, \ldots, d\}$ and sends $\mathcal{Q}$ to the prover.

2. (a) If $\mathcal{Q} = 0$, then the Prover sends $(r_1, r_2, \ldots, r_{d-1}, t_0, f_0,$
$\forall p,\ 1 \leq p \leq d-2,\ \forall i_1, \ldots i_p,\ 1 \leq i_1 < i_2 < \ldots i_p \leq d-1,\ f_0^{i_1 \ldots i_p})$.
The Verifier checks if

$$c_0 = Com\Big(r_1, \ldots, r_{d-1}, G(t_0, r_1, \ldots, r_{d-1}) + \sum_{p=1}^{d-2}(-1)^{d-p}$$

$$\sum_{1 \leq i_1 < \ldots < i_p \leq d-1} f_0^{i_1 \ldots i_p} + (-1)^d f_0\Big)$$

and $\forall i,\ 1 \leq i \leq d-1$,

$$c_{2i-1} = Com\Big(r_1, \ldots, r_{i-1}, r_{i+1}, \ldots, r_{d-1}, t_0, f_0,$$

$$\forall p,\ 1 \leq p \leq d-2,\ \forall i_1, \ldots i_p,\ 1 \leq i_1 < i_2 < \ldots i_p \leq d-1,$$

$$\text{such that } \forall j,\ i_j \neq i,\ f_0^{i_1 \ldots i_p}\Big)$$

(b) If $\mathcal{Q} = d$, then the Prover sends $(r_1, r_2, \ldots, r_{d-1}, t_1, f_1,$
$\forall p,\ 1 \leq p \leq d-2,\ \forall i_1, \ldots i_p,\ 1 \leq i_1 < i_2 < \ldots i_p \leq d-1,\ f_1^{i_1 \ldots i_p})$.
$\forall i,\ 1 \leq i \leq d-1$,
The Verifier checks if

$$c_0 = Com\Big(r_1, \ldots, r_{d-1}, v - G(t_1, r_1, \ldots, r_{d-1}) - \sum_{p=1}^{d-2}(-1)^{d-p}$$

$$\sum_{1 \leq i_1 < \ldots < i_p \leq d-1} f_1^{i_1 \ldots i_p} -$$

$$(-1)^d f_1 + \sum_{i=1}^{d}(-1)^{d-i} \sum_{\substack{S \subset \{1,\ldots,d-1\} \\ |S|=i}} F(\sum_{j \in S} r_j)\Big)$$

and $\forall i,\ 1 \leq i \leq d-1$,

$$c_{2i} = Com\Big(r_1, \ldots, r_{i-1}, r_{i+1}, \ldots, r_{d-1}, t_1, f_1,$$

$$\forall p,\ 1 \leq p \leq d-2,\ \forall i_1, \ldots i_p,\ 1 \leq i_1 < i_2 < \ldots i_p \leq d-1, \text{ such}$$

$$\text{that } \forall j,\ i_j \neq i,\ f_1^{i_1 \ldots i_p}\Big)$$

(c) if $\mathcal{Q} = i$, then the prover sends $(r_0, r_1, \ldots, r_{i-1}, r_{i+1}, \ldots, r_{d-1}, t_1, f_1,$
$\forall p,\ 1 \leq p \leq d-2,\ \forall i_1, \ldots i_p,\ 1 \leq i_1 < i_2 < \ldots i_p \leq d-1,$
such that $\forall j,\ i_j \neq i,\ f_1^{i_1 \ldots i_p})$
The Verifier checks if

$$c_{2i-1} = Com\Big(r_1, \ldots, r_{i-1}, r_{i+1}, \ldots, r_{d-1}, r_0 - t_1, F(r_0) - f_1,$$

$$\forall p,\ 1 \leq p \leq d-2,\ \forall i_1, \ldots i_p,\ 1 \leq i_1 < i_2 < \ldots i_p \leq d-1,$$
$$\text{such that } \forall j,\ i_j \neq i,\ \ F(r_0 + r_{i_1} + \ldots + r_{i_p}) - f_1^{i_1 \ldots i_p}\Big)$$

and

$$c_{2i} = Com\Big(r_1, \ldots, r_{i-1}, r_{i+1}, \ldots, r_{d-1}, t_1, f_1,$$
$$\forall p,\ 1 \leq p \leq d-2,\ \forall i_1, \ldots i_p,\ 1 \leq i_1 < i_2 < \ldots i_p \leq d-1,$$
$$\text{such that } \forall j,\ i_j \neq i,\ \ f_1^{i_1 \ldots i_p}\Big)$$

We now give the properties of the $ZK(d)$ Scheme.

Theorem 3 *The 3-pass protocol is statistically zero knowledge when the commitment scheme Com is statistically hiding.*

Proof sketch. Let $\mathcal{S}$ be a simulator which takes F and v without knowing s, and interacts with a cheating verifier $\mathcal{CV}$. We show that the simulator can impersonate the honest prover with probability $\frac{d}{d+1}$. The simulator randomly chooses a value $Ch^* \in_R \{0, 1, \ldots, d\}$ and vectors $s', r'_0, r'_1, \ldots, r'_{d-2}, t'_0 \in_R \mathbb{F}_q^n$, $f'_0 \in_R \mathbb{F}_q^m$, and $\forall p,\ 1 \leq p \leq d-2,\ \forall i_1, \ldots i_p,\ 1 \leq i_1 < i_2 < \ldots i_p \leq d-1,\ f_0^{'1 \ldots i_p},\ \in_R \mathbb{F}_q^m$, where Ch^* is a prediction what value the cheating verifier $\mathcal{CV}$ will not choose. Then it computes

$$r'_{d-1} \leftarrow s' - \sum_{i=1}^{d-2} r'_i$$
$$t'_1 \leftarrow r'_0 - t'_0$$
$$f'_1 \leftarrow F(r'_0) - f'_0$$
$$\forall p,\ 1 \leq p \leq d-2,\ \forall i_1, \ldots i_p,\ 1 \leq i_1 < i_2 < \ldots i_p \leq d-1,$$
$$f_1^{'i_1 \ldots i_p} \leftarrow F(r'_0 + r'_{i_1} + \ldots + r'_{i_p}) - f_0^{'i_1 \ldots i_p}$$

Moreover it sets:

1. If $Ch^* = 0$,
   $c'_0 = v - G(t'_1, r'_1, \ldots, r'_{d-1}) - \sum_{p=1}^{d-2}(-1)^{d-p} \sum_{1 \leq i_1 < \ldots < i_p \leq d-1} f_1^{'i_1 \ldots i_p} -$
   $(-1)^d f'_1 \sum_{i=1}^{d}(-1)^{d-i} \sum_{\substack{S \subset \{1,\ldots,d-1\} \\ |S|=i}} F(\sum_{j \in S} r'_j)$,
   else $c_0 = G(t'_0, r'_1, \ldots, r'_{d-1}) + \sum_{p=1}^{d-2}(-1)^{d-p} \sum_{1 \leq i_1 < \ldots < i_p \leq d-1} f_0^{'i_1 \ldots i_p}$
   $+ (-1)^d f'_0$
2. If $Ch^* = i$, $1 \leq i \leq d-1$, $f_1^{'12 \ldots (i-1)(i+1) \ldots (d-1)} =$
   $v - F(s') + F(r'_0 + r_1 + \ldots + r'_{i-1} + r_{i+1} + \ldots + r'_{d-1})$
   $- f_0^{'12 \ldots (i-1)(i+1) \ldots (d-1)}$,
   else $f_1^{'12 \ldots (i-1)(i+1) \ldots (d-1)} = F(r'_0 + r_1 + \ldots + r'_{i-1} + r_{i+1} + \ldots + r'_{d-1})$
   $- f_0^{'12 \ldots (i-1)(i+1) \ldots (d-1)}$,

It also computes:

$$\forall i,\ 1 \leq i \leq d-1$$

$$c'_{2i-1} \leftarrow Com\Big(r'_1, \ldots, r'_{i-1}, r'_{i+1}, \ldots, r'_{d-1}, t'_0, f'_0,$$

$$\forall p,\ 1 \leq p \leq d-2,\ \forall i_1, \ldots i_p,\ 1 \leq i_1 < i_2 < \ldots i_p \leq d-1,$$

$$\text{such that } \forall j,\ i_j \neq i,\ \ {f_0}^{'i_1 \ldots i_p}\Big)$$

$$c_{2i} \leftarrow Com\Big(r'_1, \ldots, r'_{i-1}, r'_{i+1}, \ldots, r'_{d-1}, t'_1, f'_1,$$

$$\forall p,\ 1 \leq p \leq d-2,\ \forall i_1, \ldots i_p,\ 1 \leq i_1 < i_2 < \ldots i_p \leq d-1,$$

$$\text{such that } \forall j\ i_j \neq i,\ \ {f_1}^{'i_1 \ldots i_p}\Big)$$

and sends $c'_0, c'_1, c'_2, \ldots, c'_{2d-2}$ to t$\mathcal{CV}$.
Due to the statistically hiding property of Com, a challenge Ch from $\mathcal{CV}$ is different from Ch^* with probability $\frac{d}{d+1}$. If $Ch \neq Ch^*$, then $(r_1, r_2, \ldots, r_{d-1}, t_0, f_0, \forall p,\ 1 \leq p \leq d-2,\ \forall i_1, \ldots i_p,\ 1 \leq i_1 < i_2 < \ldots i_p \leq d-1,\ f_0^{i_1 \ldots i_p})$, and $\forall i,\ 1 \leq i \leq d-1,\ (r_0, r_1, \ldots, r_{i-1}, r_{i+1}, \ldots, r_{d-1}, t_1, f_1, \forall p,\ 1 \leq p \leq d-2,\ \forall i_1, \ldots i_p,\ 1 \leq i_1 < i_2 < \ldots i_p \leq d-1,$ such that $\forall j\ i_j \neq i,\ f_1^{i_1 \ldots i_p})$ are accepted responses to $Ch = 0, 1, \ldots (d-1)$. □

Theorem 4 *The 3-pass protocol is proof of zero knowledge with zero knowledge error $\frac{d}{d+1}$ when the commitment scheme Com is computationnaly binding.*

Proof sketch. Let $\forall i,\ 0 \leq i \leq d$, let $((c_0, c_1, c_2, \ldots, c_{2d-2}), Ch_i, Rsp_i)$ be $d+1$ transcripts such that $Ch_i = i$ and $Dec(F, v; (c_0, c_1, c_2, \ldots, c_{2d-2}), Ch_i, Rsp_i) = 1$ for $i \in \{0, 1, 2, \ldots, d\}$. Then, by using the four transcripts, we show that we are able either to break the binding property of Com or extract a solution for v. Consider the situation where the responses are parsed as

$$Rsp_0 = (\tilde{r}_1^{(0)}, \tilde{r}_2^{(0)}, \ldots, \tilde{r}_{d-1}^{(0)}, \tilde{t}_0^{(0)}, \tilde{f}_0^{(0)},$$

$$\forall p,\ 1 \leq p \leq d-2,\ \forall i_1, \ldots i_p,\ 1 \leq i_1 < i_2 < \ldots i_p \leq d-1,\ \tilde{f}_0^{(0), i_1 \ldots i_p})$$

$$\forall i,\ 1 \leq i \leq d-1,$$

$$Rsp_i = (\tilde{r}_0^{(i)}, \tilde{r}_1^{(i)}, \ldots, \tilde{r}_{i-1}^{(i)}, \tilde{r}_{i+1}^{(i)}, \ldots, \tilde{r}_{d-1}^{(i)}, \tilde{t}_1^{(i)}, \tilde{f}_1^{(i)},$$

$$\forall p,\ 1 \leq p \leq d-2,\ \forall i_1, \ldots i_p,\ 1 \leq i_1 < i_2 < \ldots i_p \leq d-1,$$

$$\text{such that } \forall j,\ i_j \neq i,\ \tilde{f}_1^{(i), i_1 \ldots i_p})$$

$$Rsp_d = (\tilde{r}_1^{(d)}, \tilde{r}_2^{(d)}, \ldots, \tilde{r}_{d-1}^{(d)}, \tilde{t}_1^{(d)}, \tilde{f}_1^{(0)},$$

$$\forall p,\ 1 \leq p \leq d-2,\ \forall i_1, \ldots i_p,\ 1 \leq i_1 < i_2 < \ldots i_p \leq d-1,\ \tilde{f}_1^{(d), i_1 \ldots i_p})$$

We obtain:

$$c_0 = Com\Big(\tilde{r}_1^{(d)}, \ldots, \tilde{r}_{d-1}^{(d)}, v - G(\tilde{t}_1^{(d)}, \tilde{r}_1^{(d)}, \ldots, \tilde{r}_{d-1}^{(d)}) - \sum_{p=1}^{d-2}(-1)^{d-p}$$

$$\sum_{1 \le i_1 < \ldots < i_p \le d-1} \tilde{f}_1^{(d),i_1 \ldots i_p} -$$

$$(-1)^d \tilde{f}_1^{(d)} + \sum_{i=1}^{d}(-1)^{d-i} \sum_{\substack{S \subset \{1,\ldots,d-1\} \\ |S|=i}} F(\sum_{j \in S} \tilde{r}_j^{(d)})\Big)$$

$$= Com\Big(\tilde{r}_1^{(0)}, \ldots, \tilde{r}_{d-1}^{(0)}, G(\tilde{t}_0^{(0)}, \tilde{r}_1^{(0)}, \ldots, \tilde{r}_{d-1}^{(0)}) + \sum_{p=1}^{d-2}(-1)^{d-p}$$

$$\sum_{1 \le i_1 < \ldots < i_p \le d-1} \tilde{f}_0^{(0),i_1 \ldots i_p} + (-1)^d \tilde{f}_0^{(0)}\Big)$$

$$\forall i,\ 1 \le i \le d-1,$$

$$c_{2i-1} = Com\Big(\tilde{r}_1^{(i)}, \ldots, \tilde{r}_{i-1}^{(i)}, \tilde{r}_{i+1}^{(i)}, \ldots, \tilde{r}_{d-1}^{(i)}, \tilde{r}_0^{(i)} - \tilde{t}_1^{(i)}, F(\tilde{r}_0^{(i)}) - \tilde{f}_1^{(i)},$$

$$\forall p,\ 1 \le p \le d-2,\ \forall i_1, \ldots i_p,\ 1 \le i_1 < i_2 < \ldots i_p \le d-1,$$

$$\text{such that } \forall j,\ i_j \ne i,\ F(\tilde{r}_0^{(i)} + \tilde{r}_{i_1}^{(i)} + \ldots + \tilde{r}_{i_p}^{(i)}) - \tilde{f}_1^{(i),i_1 \ldots i_p}\Big)$$

$$= Com\Big(\tilde{r}_1^{(0)}, \ldots, \tilde{r}_{i-1}^{(0)}, \tilde{r}_{i+1}^{(0)}, \ldots, \tilde{r}_{d-1}^{(0)}, \tilde{t}_0^{(0)}, \tilde{f}_0^{(0)},$$

$$\forall p,\ 1 \le p \le d-2,\ \forall i_1, \ldots i_p,\ 1 \le i_1 < i_2 < \ldots i_p \le d-1,$$

$$\text{such that } \forall j,\ i_j \ne i,\ \tilde{f}_0^{(0),i_1 \ldots i_p}\Big)$$

$$c_{2i} = Com\Big(\tilde{r}_1^{(i)}, \ldots, \tilde{r}_{i-1}^{(i)}, \tilde{r}_{i+1}^{(i)}, \ldots, \tilde{r}_{d-1}^{(i)}, \tilde{t}_1^{(i)}, \tilde{f}_1^{(i)},$$

$$\forall p,\ 1 \le p \le d-2,\ \forall i_1, \ldots i_p,\ 1 \le i_1 < i_2 < \ldots i_p \le d-1,$$

$$\text{such that } \forall j,\ i_j \ne i,\ \tilde{f}_1^{(i),i_1 \ldots i_p}\Big)$$

$$= Com\Big(\tilde{r}_1^{(d)}, \ldots, \tilde{r}_{i-1}^{(d)}, \tilde{r}_{i+1}^{(d)}, \ldots, \tilde{r}_{d-1}^{(d)}, \tilde{t}_1^{(d)}, \tilde{f}_1^{(d)},$$

$$\forall p,\ 1 \le p \le d-2,\ \forall i_1, \ldots i_p,\ 1 \le i_1 < i_2 < \ldots i_p \le d-1,$$

$$\text{such that } \forall j,\ i_j \ne i,\ \tilde{f}_1^{(d),i_1 \ldots i_p}\Big)$$

If two tuples of the arguments of Com are are distinct on either of the above equations, the binding property of Com is broken.
Otherwise the conditions on c_0 give:

$$\tilde{r}_1^{(0)} = \tilde{r}_1^{(d)}, \ldots, \tilde{r}_{d-1}^{(0)} = \tilde{r}_{d-1}^{(d)}$$

$$v = G(\tilde{t}_0^{(0)} + \tilde{t}_1^{(d)}, \tilde{r}_1^{(d)}, \ldots, \tilde{r}_{d-1}^{(d)}) + \sum_{p=1}^{d-2}(-1)^{d-p}$$

$$\sum_{1 \le i_1 < \ldots < i_p \le d-1} (\tilde{f}_0^{(0),i_1 \ldots i_p} + \tilde{f}_1^{(d),i_1 \ldots i_p})$$

$$+(-1)^d(\tilde{f}_0^{(0)} + \tilde{f}_1^{(d)}) - \sum_{i=1}^{d}(-1)^{d-i} \sum_{\substack{S \subset \{1,\ldots,d-1\} \\ |S|=i}} F(\sum_{j \in S} \tilde{r}_j^{(d)})\Big)$$

The conditions on c_{2i-1} give:

$$\tilde{r}_1^{(i)} = \tilde{r}_1^{(0)}, \ldots, \tilde{r}_{i-1}^{(i)} = \tilde{r}_{i-1}^{(0)}, \tilde{r}_{i+1}^{(i)} = \tilde{r}_{i+1}^{(0)}, \ldots \tilde{r}_{d-1}^{(i)} = \tilde{r}_{d-1}^{(0)}$$

$$\tilde{t}_0^{(i)} = \tilde{t}_0^{(0)},\ \tilde{f}_0^{(i)} = \tilde{f}_0^{(0)}$$

$$\forall p,\ 1 \le p \le d-2,\ \forall i_1, \ldots i_p,\ 1 \le i_1 < i_2 < \ldots i_p \le d-1,$$

$$\text{such that } \forall j,\ i_j \ne i,\ \tilde{f}_0^{(i),i_1 \ldots i_p} = \tilde{f}_0^{(0),i_1 \ldots i_p}$$

The conditions on c_{2i} give:

$$\tilde{r}_1^{(i)} = \tilde{r}_1^{(d)}, \ldots, \tilde{r}_{i-1}^{(i)} = \tilde{r}_{i-1}^{(d)}, \tilde{r}_{i+1}^{(i)} = \tilde{r}_{i+1}^{(d)}, \ldots \tilde{r}_{d-1}^{(i)} = \tilde{r}_{d-1}^{(d)}$$

$$\tilde{t}_1^{(i)} = \tilde{t}_1^{(d)},\ \tilde{f}_1^{(i)} = \tilde{f}_1^{(d)}$$

$$\forall p,\ 1 \le p \le d-2,\ \forall i_1, \ldots i_p,\ 1 \le i_1 < i_2 < \ldots i_p \le d-1,$$

$$\text{such that } \forall j,\ i_j \ne i,\ \tilde{f}_1^{(i),i_1 \ldots i_p} = \tilde{f}_1^{(d),i_1 \ldots i_p}$$

We obtain:

$$\tilde{t}_0^{(0)} + \tilde{t}_1^{(d)} = \tilde{t}_0^{(1)} + \tilde{t}_1^{(1)} = \tilde{r}_0^{(1)}$$

$$F(\tilde{r}_0^{(1)}) = \tilde{f}_0^{(1)} + \tilde{f}_1^{(1)} = \tilde{f}_0^{(0)} + \tilde{f}_1^{(d)}$$

$$\forall i,\ 1 \le i \le d-1,\ F(\tilde{r}_0^{(1)} + \tilde{r}_i^{(d)}) = F(\tilde{r}_0^{(1)} + \tilde{r}_i^{(1)}) = \tilde{f}_0^{(1),i} + \tilde{f}_1^{(1),i} = \tilde{f}_0^{(0),i} + \tilde{f}_1^{(d),i}$$

More generally

$$F(\tilde{r}_0^{(1)} + \tilde{r}_{i_1}^{(d)} + \ldots + \tilde{r}_{i_p}^{(d)}) = F(\tilde{r}_0^{(1)} + \tilde{r}_{i_1}^{(1)} + \ldots + \tilde{r}_{i_p}^{(1)}) =$$

$$\tilde{f}_0^{(1),i_1 \ldots i_p} + \tilde{f}_1^{(1),i_1 \ldots i_p} = \tilde{f}_0^{(0),i_1 \ldots i_p} + \tilde{f}_1^{(d),i_1 \ldots i_p}$$

Finally, we obtain

$$v = F(\tilde{r}_0^{(1)} + \tilde{r}_1^{(d)} + \ldots + \tilde{r}_{d-1}^{(d)})$$

This means that a solution $\tilde{r}_0^{(1)} + \tilde{r}_1^{(d)} + \ldots + \tilde{r}_{d-1}^{(d)}$ for v is extracted. □

Remark. As for the case $d = 3$, it is possible to design a 5-pass scheme.

A.2 The $\tilde{ZK}(d)$ Scheme

Here we explain how it is possible to design the $\tilde{ZK}(d)$ Scheme. Again there are two functions F and $\tilde{F}$. If M (resp. $\tilde{M}$) denotes the number of computations for F (resp. $\tilde{F}$), then $M \simeq n^d/d!$ and $\tilde{M} \simeq \frac{n^d}{d!} + n^{\lceil \frac{d}{2} \rceil}/\lceil \frac{d}{2} \rceil! \simeq M$. For $ZK(d)$, there are n variables and m equations. For $\tilde{ZK}(d)$, there are $\tilde{n} \simeq n + n^{\lceil \frac{d}{2} \rceil}/\lceil \frac{d}{2} \rceil!$ variables and $\tilde{m} \simeq m + n^{\lceil \frac{d}{2} \rceil}/\lceil \frac{d}{2} \rceil!$ equations.