Authors: N. Alexandris, M. Burmester , V. Chrissikopoulos

Title: The Structure and complexity of interactive zero-knowledge proofs

Creator: HDML

BULLETIN THE
GREEK MATHEMATICAL SOCIETY
VOLUME 30, 1989

THE STRUCTURE AND COMPLEXITY OF INTERACTIVE ZERO-KNOWLEDGE PROOFS

BY

N. Alexandris, M. Burmester and V. Chrissikopoulos

Abstract

We discuss the algebraic structure and the computational complexity of 3-Step interactive zero-knowledge proofs and describe a general class of protocols for which the number of iterations has an optimal lower bound. These protocols are practical and are the most efficient presently available, as far as we know.

1. Introduction

There are many ways in which a prover A can convince a verifier B that he possesses a short proof of a theorem. The simplest way is to give B the short proof. B can then check it, and accept it if it is valid. This however is not a zero-knowledge proof: the verifier learns *all* the information. Zero-knowledge proofs reveal no more than is strictly necessary. In this case, the validity of a theorem.

The proofs which concern us are proofs of theorems x in **NP** languages [7]. Theorems in such languages posses short proofs or witnesses. Given a witness s it is 'easy' to check that x is valid, *i.e.*, that x belongs to some language L. Finding a witness however (if it exists) is usually 'hard'. In our first model the prover A has unlimited computational power (so A can find a witness s if $x \in L$), and B is polynomially bounded (B only needs to check that s is a witness of x). This is a natural setting for proofs of membership in a language [8]. When $x \in L$, A can always find a witness s for x and prove to B that such a witness exits. B is not in a position to check for himself that $x \in L$ because he is polynomially bounded. Such proofs involve randomness (both A and B toss coins) and interaction.

The protocol which we will use has 3 interactive steps. In the first step the prover A selects a random number r, which he uses to 'disguise' the witness s. He commits himself to r by sending the verifier B: $Z = f(r)$, where f is a suitable (polynomial time) function. In the second step B tosses a coin and sends A a query $q = 0$ or $q = 1$. Finally in the third step the prover answers by revealing either the disguise $Y = r$ when $q = 0$, or the disguised witness $Y = rs$ when $q = 1$. This completes one interactive round. The verifier checks that the answer is correct. That is, (a) that it is of the 'proper' form, say a unit (an invertible element), and (b) that it satisfies an appropriate algebraic relation involving x, Z, q and Y. If the answer is not correct then the protocol is halted. Otherwise the round is repeated, altogether t times. After t successful rounds B accepts that $x \in L$. The protocol is general and allows for different algebraic interpretations. For example r might be a permutation or an element of a monoid. The set of queries might be much larger. We have only described the simplest case. We remark that for cryptographic applications f must be a 'one-way' function: *i.e.*, it must be 'easy' to compute $f(r)$ given r but 'hard' to compute r given $f(r)$. This is because the verifier can compute s from r and $Y = rs$ (when r is a unit: as we shall see r is almost always a unit).

The prover A and the verifier B may cheat. In the first case A may wish to use the protocol to prove to B that there exists a short proof s for a false theorem x (so $x \notin L$). With zero-knowledge proofs, for any q, the prover can compute a Y and a Z such that Y is a correct answer if Z is the commitment and q is the query. If A guesses q and sends the verifier Z in the first step and Y in the third step, then the round will be valid if the verifier's query is q. The probability that B will accept is at least $|Q|^{-t}$, were Q is the set of queries and t is the number of iterations. So t must be sufficiently large to make the probability of cheating negligible. (If a round is valid only when $x \in L$, then B can check membership in L by simulating the first round without any interaction. We are not interested in such proofs. So for some $x \notin L$ there are queries q which the prover can answer correctly. It is in this sense that A cheats). The verifier may cheat because wants to find out more about x, not merely that $x \in L$. So instead of tossing a fair coin to get q he uses a biased coin. The protocol should prevent B from

getting any information about x beyond the fact that $x \in L$.

A zero-knowledge proof (A, B) for a language L must satisfy the following three conditions:

(i) *Completeness:* If $x \in L$ then B will accept with overwhelming probability.

(ii) *Soundness:* If $x \notin L$ then B will accept with negligible probability (no matter how A deviates from the protocol).

(iii) *Zero-knowledge:* If $x \in L$ then any B, whatever his previous knowledge, learns no more than is strictly necessary, *e.g.*, only that $x \in L$ (one bit of information).

The probabilities are taken over the coin tosses of A and B. The last condition makes use of a simulation: when $x \in L$, B's view of the communication can be simulated by a polynomial time Turing machine, the *simulator*. So B cannot expect to learn more than the validity of the proof (which is secured from the first two conditions).

We will also consider proofs of knowledge [4]. With such proofs both the prover A and the verifier B are polynomially bounded, but A has access to an oracle. The language L belongs to the class **P** of polynomial time problems [7]. So B can check membership in L for himself. The prover obtains from the oracle a witness s for x which, as in the case of proofs of membership, is hard to compute but easy to check. The protocol proves knowledge of s, or more generally, knowledge of the predicate (or relation) $P(s, x)$ which relates s and x.

As with proofs of membership, a zero-knowledge proof (A, B) of knowledge of a predicate $P(\cdot, \cdot)$ must satisfy three conditions [4]:

(i) *Completeness*: If A knows a witness s such that $P(s, x)$ then B accepts with overwhelming probability.

(ii) *Soundness*: If B accepts with non-negligible probability then A must know a witness s. The knowledge of A is checked by a polynomial time probabilistic Turing machine, the *interrogator*, which by 'probing' and 'resetting' A succeed in 'extracting' the witness s.

(iii) *Unrestricted input zero-knowledge*: The view of the communication of (A, B) that any B has, whatever his previous knowledge, can be simulated for any x (even when $x \notin L$).

We shall see that the completeness condition for zero-knowledge

proofs is determined, to a large extent, by the algebraic structure of the setting, whereas the soundness condition is determined by the computational aspects of the protocol , *e.g.*, the number of rounds t of the protocol and the number of queries $|Q|$. Zero-knowledge makes use of both. The simulator (i) computes an algebraic expression for each query q and, (ii) probes and resets the verifier to obtain the appropriate distribution.

Our main objective in this paper is to show that there exist 3-step zero-knowledge proofs for which the number of iterations is optimal. We only consider protocols with polynomially bounded 'query sets'. In the remainder of the section we introduce a general 3-step interactive protocol. This will be used to specify particular cases. In §2 we consider proofs of membership. We start with $\mathrm{P}_{Z_2}(\mathrm{f})$ which is easiest to grasp, and build up to more complex protocols. This makes the proofs simpler. There is a small amount of overlap which is offset by a clearer presentation. Our main results are at the end of this section. In §3 we consider proofs of knowledge. We conclude by restating our results.

The protocol

To start with we consider the Goldwasser–Micali–Rackoff quadratic residuosity proof [8], which in many respects is typical of 3-step zero-knowledge proofs.

The protocol $\mathrm{P}_{\mathrm{GMR}}$: input $x = x_0 x_1$, where x_1 is a quadratic residue mod x_0

First B checks that $x_1 \in Z^*_{x_0}$, where $Z^*_{x_0}$ is the ring of units mod x_0. If so, the following 3 steps are repeated $t = \lceil \log_2 x_0 \rceil$ times:

Step 1 A sends B: $Z \equiv r^2 (\mathrm{mod}\, x_0)$, where $r \in_R Z^*_{x_0}$,[1]

Step 2 B sends A: a bit $q \in_R \{0,1\}$.

Step 3 A sends B: $Y \equiv r s_1^q (\mathrm{mod}\, x_0)$, where $s_1 \in Z^*_{x_0}$ is such that $s_1^2 x_1 \equiv 1 (\mathrm{mod}\, x_0)$.

B verifies that Y is a unit and that $Z \equiv Y^2 x_1^q (\mathrm{mod}\, x_0)$. If the verifi-

[1] $r \in_R Z^*_x$ means that r is selected randomly with uniform distribution from Z^*_x.

cation fails then the protocol is halted. B accepts the proof of A after t complete rounds.

Observe that s_1 is a witness to the quadratic residuosity of x_1, since $x_1 \equiv s_1^{-2} (\mathrm{mod}\, x_0)$. Furthermore $Z \equiv Y^2 x_1^q (\mathrm{mod}\, x_0)$ reduces to either $r^2 \equiv r^2 (\mathrm{mod}\, x_0)$ or $r^2 \equiv (rs_1)^2 x_1 (\mathrm{mod}\, x_0)$, and $(rs_1)^2 x_1 \equiv r^2 (s_1^2 x_1) \equiv r^2 (\mathrm{mod}\, x_0)$ since $s_1^2 x_1 \equiv 1 (\mathrm{mod}\, x_0)$.

The protocol $\mathrm{P}_{\mathrm{GEN}}$: input x

First B checks that the input x is 'valid'. If so, the following 3 steps are repeated t times:

Step 1 A sends B: Z (*A's commitment*),

Step 2 B sends A: q (*B's query*),

Step 3 A sends B: $Y = Y(q, Z, x)$ (*A's answer*).

B verifies that the answer Y is 'correct'. If it is not correct then the round is halted. B accepts after t rounds.

This describes in general terms a 3-step interactive protocol [3]. It is intentionally vague. The terms 'valid' and 'correct' have not been defined, t has not been specified, and it is not clear how A and B use their randomness in Step 1 and Step 2. In the following sections we consider particular cases of this protocol and define these terms appropriately.

2. Proofs of membership

Let L be a language over $\{0,1\}^*$. The elements x of L are encodings of 'particular instances'. For example when L is QR, the language of quadratic residues, then $x = x_0 x_1 \in L$ if the number represented by x_1 is a quadratic residue modulo the number represented by x_0. So $x_0 x_1$ represents a quadratic residue. A set of numbers $x_0 x_1, x_0 x_2, \ldots, x_0 x_k$ belongs to the same 'class of instances' when $x_1, x_2, \ldots, x_k$ are all quadratic residues mod x_0. Such sets are represented by $x = x_0 x_1 \cdots x_k$. x_0 specifies the class of instances. We denote this class by L_{x_0}. So $L = \cup L_{x_0}$. When there is no ambiguity we will drop x_0, and write $x_1 x_2 \cdots x_k \in L$, $x_j \in L$ or $x_j \in L_{x_0}$.

The security parameter for all protocols is the length $|x|$ of the binary representation of the input x. The computational complexity, the asymptotic bounds and all probabilities are expressed in terms of $|x|$.

Definition An interactive protocol (A, B) is a proof for L [8] if:

(i) *Completeness* (For any constant c) when the input $x = x_0x_1 \cdots x_k$ belongs to L ($|x|$ sufficiently large) then B accepts with overwhelming probability ($> 1 - |x|^{-c}$).

(ii) *Soundness* (For any constant c) when the input $x = x_0x_1 \cdots x_k$ does not belong to L ($|x|$ sufficiently large) then B accepts only with negligible probability ($< |x|^{-c}$).

Observe that $x_0x_1 \cdots x_k \notin L$ means that some $x_i \notin L_{x_0}$.

Particular cases of $\mathrm{P}_{\mathrm{GEN}}$

The protocol $\mathrm{P}_{Z_2}(\mathrm{f})$: input $x = x_0x_1$

Step 1 A sends B: $Z = f(r)$, $r \in_R G_{x_0}$, G_{x_0} a monoid,

Step 2 B sends A: $q \in_R \{0,1\}$,

Step 3 A sends B: $Y = rs_1^q$, where $s_1 \in G^*_{x_0}$ is such that $f(s_1)x_1 = 1$.

For this protocol x_0 specifies the monoids G_{x_0} and H_{x_0}. x_1 is a valid if it is a unit in H_{x_0} and Y is correct if it is a unit in G_{x_0} and if $Z = f(Y)\,x_1^q$. $G^*_{x_0}, H^*_{x_0}$ are the groups of units of G_{x_0}, H_{x_0}. The protocol accepts the language $L = \cup L_{x_0}$ for which $L_{x_0} = \{x_0x_1 \mid x_1 \in H^*_{x_0} : \exists s_1 \in G^*_{x_0}$ with $f(s_1)x_1 = 1\}$.

The protocol $\mathrm{P}_{Z_2}(\mathrm{f})$ is a complete proof for L if:

(i) almost all elements of G_{x_0} are units, and

(ii) when $x = x_0x_1 \in L$ and $f(s_1)x_1 = 1$, $s_1 \in G^*_{x_0}$, then $f(r) = Z = f(rs_1^q)\,x_1^q$ for almost all $r \in G_{x_0}$.

Since $x_1^q f(s_1)^q = 1$, (ii) is equivalent to $f(r)f(s_1)^q = f(rs_1^q)$. We shall assume condition (i) and take f to be a homomorphism of G, that is: $f(a \cdot b) = f(a) \cdot f(b)$ for all $a, b \in G$. With these assumptions $\mathrm{P}_{Z_2}(\mathrm{f})$ is complete. Observe that for the proof of the quadratic residuosity $\mathrm{P}_{\mathrm{GMR}}$ we have $G = G^* = Z^*_{x_0}$, where $Z^*_{x_0}$ is the group of units of Z_{x_0}, and $f : Z^*_{x_0} \to Z^*_{x_0}$; $x \to x^2 \,(\mathrm{mod}\, x_0)$ is a homomorphism of Z_{x_0}.

The protocol $P_{Z_2}(f)$ is sound if and only if

$$|x|^c \succeq t \succ \log|x|, \; c \text{ a (positive) constant.}^2$$

Necessity: Suppose that $t \leq d\log|x|$ for some constant d, and let $x \notin L$. A cheating prover can convince B that $x \in L$ by sending either $Z_0 = f(r)$ or $Z_1 = f(r)x_1$ in Step 1, with equal probability, and $Y = r$ in Step 3. Half the times the answer is correct (when $q = 0$ or $q = 1$ respectively). So B will accept with non-negligible probability $2^{-t} \geq 2^{-d\log|x|} = |x|^{-d}$, d a constant. Consequently $P_{Z_2}(f)$ is not sound. The upper limit $|x|^c \succeq t$ is necessary because B is polynomially bounded.

Sufficiency: The proof is similar to the one in [4]. Suppose that the condition holds, that $x \notin L$ and that B accepts with non-negligible probability. We shall derive a contradiction. Consider the 'execution tree' T of $P_{Z_2}(f)$. This is a truncated tree which describes the responses of A to the queries of B. Its vertices correspond to the states of A and its edges to the queries of B. If A cannot answer correctly a particular query of B then the corresponding edge and all its descendants are eliminated. T has height t, the number of iterations of $P_{Z_2}(f)$. The fraction of leaves that survive the truncation is the equal to the probability that B accepts. Since $t \succ \log|x|$, at least two leaves must survive otherwise this fraction (2^{-t}) is negligible. We trace these leaves back to the root to find a vertex at which A answers both queries. So there exist Z, Y_0, Y_1 such that $Z = f(Y_0) = f(Y_1)x_1$. Then $f(Y_0^{-1}Y_1)x_1 = 1$ (Y_0 is a unit and f is a homomorphism) and therefore $s_1 = Y_0^{-1}Y_1$ is a witness for x_1. Thus $x = x_0x_1 \in L$. This is a contradiction and proves that B accepts with negligible probability.

Remark The convention with zero-knowledge protocols is to specify the rules for honest provers and honest verifiers. When the prover cheats he does not adhere to the strict rules of the protocol and the description of P_{GEN} is more appropriate. For example in P_{GMR} when s_1 does not exist, then Step 3 in which A sends B: $Y \equiv rs_1^{q_1} (\mathrm{mod}\ x_0)$ makes little

[2]These are asymptotic bounds. $|x|^c \succeq t$ means that $|x|^c \geq dt$, d a positive constant, $|x|$ large; $t \succ \log|x|$ means that $t > d\log|x|$ for all constants d, $|x|$ large. The first is equivalent to $t = O(|x|^c)$ and the second to $\lim_{x\to\infty} \log|x|/t = 0$.

sense. Similarly in Step 1, the prover may send $Z \equiv a^2 x_1 \pmod{x_0}$ which is not of the form $r^2 \pmod{x_0}$ when x_1 is a quadratic non-residue. If the prover sends $Z \equiv a^2 x_1^q \pmod{x_0}$ in Step 1 and $Y = a$, a unit, in Step 3, then the verifier *must* accept when his query is q. Thus a cheating prover who guesses the queries will succeed half the times.

We can extend this protocol to one with 2^k queries.

The protocol $\mathrm{P}_{Z_2^k}(\mathrm{f})$: input $x = x_0 x_1 \cdots x_k$, $k = O(\log |x|)$

Step 1 A sends B: $Z = f(r)$, $r \in_R G_{x_0}$, G_{x_0} a monoid,

Step 2 B sends A: $q = (q_1, \ldots, q_k) \in_R \{0,1\}^k$,

Step 3 A sends B: $Y = r s_1^{q_1} \cdots s_k^{q_k}$, where $s_j \in G_{x_0}^*$ are such that $f(s_j) x_j = 1$ for all $j > 0$.

For this protocol the query set is $\{0,1\}^k$. The input x is valid if all the x_j, $j > 0$, are units, and the answer Y is correct if it is a unit and if $Z = f(Y)\, x_1^{q_1} \cdots x_k^{q_k}$.

As with $\mathrm{P}_{Z_2}(\mathrm{f})$ this protocol is complete if, (i) almost all elements of G_{x_0} are units and (ii) f is a homomorphism of G_{x_0}. Observe that when $G_{x_0} = Z_{x_0}$, x_0 a product of two distinct primes and $f : Z_{x_0} \to Z_{x_0};\ x \to x^2 \pmod{x_0}$, then we have the Fiat–Shamir scheme [5]. The protocol is sound if and only if,

$$|x|^c \succeq t \succ \log |x|, \quad c \text{ a constant.}$$

The proof is an extension of the previous one. The verifier B should not accept unless all the $x_j \in L$, $j > 0$. When $x_1, \ldots, x_{k-1} \in L$ but $x_k \notin L$, then a cheating prover can convince B that all $x_j \in L$ with probability $\frac{1}{2}$ in each round (by guessing the bit q_k of q). For this to be negligible we must have $t \succ \log |x|$. This proves that the condition is necessary. The proof for sufficiency is similar to the one in [4]. We consider the execution tree T. A vertex of T is called 'heavy' if its degree is $> \frac{1}{2} 2^k = 2^{k-1}$. At least one vertex of the execution tree must be heavy otherwise the fraction of leaves that survive the truncation is $< 2^{(k-1)t}/2^{kt} = 2^{-t}$, which is negligible. So for some commitment Z the prover must answer correctly more than half of all possible queries. Then for every $j = 1, 2, \ldots, k$ there exist two queries $(q_i), (q_i')$ with $q_i = q_i'$ for all i except $q_j = 0$, $q_j' = 1$, which the prover answers correctly with

the same commitment is Z. If Y_0, Y_1 are the corresponding answers then,

$$Z = f(Y_0)\, x_1^{q_1} \cdots x_j^0 \cdots x_k^{q_k} = f(Y_1)\, x_1^{q_1} \cdots x_j^1 \cdots x_k^{q_k}.$$

Take $j = 1$. Then $f(Y_0) = f(Y_1)x_1$, by cancelling the x_i. Since f is a homomorphism and Y_0 is a unit this is equivalent to $f(Y_0^{-1}Y_1)x_1 = 1$. So $s_1 = Y_0^{-1}Y_1$ is a witness. We get the other witnesses $s_2, s_3, \ldots$ in a similar way, each time substituting $x_1, x_2, \ldots$ by $f(s_1), f(s_2), \ldots$ in the above equation. Thus $x_j \in L$ for all j. (Observe that we do not need to assume that multiplication is commutative.)

The protocol $\mathrm{P}_{Q^k}(\mathrm{f})$: input $x = x_0 x_1 \cdots x_k$, $|Q|^k = O(|x|^c)$, c a constant.

Step 1 A sends B: $Z = f(r)$, $r \in_R G$, G a monoid,

Step 2 B sends A: $q = (q_1, \ldots, q_k) \in_R Q^k$,

Step 3 A sends B: $Y \doteq r s_1^{q_1} \cdots s_k^{q_k}$, where $s_j \in G^*$ are such that $f(s_j)x_j = 1$ for all $j > 0$.

The query set is $Q^k = \{0, 1, \ldots, v-1\}^k$. The conditions for validity, correctness and completeness are the same as those for the protocol $\mathrm{P}_{Z_2{}^k}(\mathrm{f})$. We take f to be a homomorphism. The protocol is sound if and only if,

$$|x|^c \succeq t \succ \log|x| \,/\, \log(v/\hat{v}),\ c \text{ a constant.}$$

Here $\hat{v} = |\hat{Q}|$, where $\hat{Q}$ is a subset of Q of maximal order such that for some $x_a \in H^*_{x_0}$: $x_a^{\hat{q}} \in L$ for all $\hat{q} \in \hat{Q}$ but $x_a \notin L$ ($\hat{Q}$ may not be unique but its order $\hat{v}$ is unique). The proof is an extension of the previous one. When $x_1, \ldots, x_{k-1} \in L$ and $x_k = x_a$ ($\notin L$) then the prover will convince the verifier with probability $(\hat{v}/v)^t$ which is negligible only when $t \succ \log|x| \,/\, \log(v/\hat{v})$. So the condition is necessary. For sufficiency we assume that the condition holds, that $x \notin L$, and that B accepts with non-negligible probability. In this case we consider 'prolific' vertices, that is vertices of degree $> \frac{\hat{v}}{v} v^k = \hat{v}\, v^{k-1}$. It is easy to see that for every $j = 1, 2, \ldots, k$ we can find more than $\hat{v}$ queries (q_i) for which all the entries q_i are fixed except the j-th entry and which the

prover answers correctly with the same commitment Z. If (q_i), (q_i') are two such queries and Y, Y' are the corresponding answers then,

$$Z = f(Y)\, x_1^{q_1} \cdots x_j^{q_j} \cdots x_k^{q_k} = f(Y')\, x_1^{q_1} \cdots x_j^{q_j'} \cdots x_k^{q_k}.$$

Take $j=1$. Then $f(Y^{-1}Y')x_1^{q_1'-q_1}=1$, so that $x_1^{q_1'-q_1} \in L$. Assume that q_1 is minimum and keep it fixed. By letting q_1' vary we get a set of exponents $Q' \subset Q$ with $|Q'| > \hat{v}$ and $x_1^{q'} \in L$ for all $q' \in Q'$. Since $\hat{v}$ is maximal we must have $x_1 \in L$. Therefore there exists an s_1 such that $f(s_1)x_1 = 1$. Proceed for $j = 2, \ldots, k$ each time substituting $x_1, x_2, \ldots$ by $f(s_1^{-1}), f(s_2^{-1}), \ldots$. Then $x_j \in L$ for all j (again multiplication need not be commutative).

Remarks

1. $1 \leq \hat{v} \leq \frac{1}{2}v$. Indeed $\hat{Q} \supseteq \{0\}$ always, and if $\hat{v} > \frac{1}{2}v$ then $\hat{Q}$ contains two successive exponents of x_a, say $m, m+1$. So $x_a^m, x_a^{m+1} \in L$. Let s_1, s_2 be the corresponding witnesses. Then $s_a = s_2 s_1^{-1}$ is a witness for x_a since $f(s_2 s_1^{-1})x_a = f(s_2)(f(s_1)^{-1}x_a^{-m})x_a^{m+1} = 1$. Therefore $x_a \in L$, which is a contradiction.

2. $\log(v/\hat{v})$ ranges from 1 to $\log v$, so the lower soundness bound ranges from $\log|x|$ to $\log|x|/\log v$.

3. As observed earlier, with zero-knowledge proofs when $x \notin L$ the prover can guess the query and the verifier will accept with probability at least $1/|Q|^k$ in each round (for proofs of membership we assume that $x \in L$ cannot be checked in polynomial time). If $|Q|^k$ is polynomially bounded then, for soundness, t should not be constant. So $t \succ 1$ is an optimal lower bound.

4. Let $g : G_{x_0} \to g(G_{x_0}) = H_{x_0}$ be a homomorphism and $f : w \to g(w)^v$. The set $L_{x_0} = \{x_0 x_i \mid x_i \in H_{x_0}^* : \exists s_i \in G_{x_0}^* \text{ with } g(s_i)^v x_i = 1\} = g(G_{x_0}^*)^v = H_{x_0}^{*v}$ which $\mathrm{P}_{Z_*}(g^v)$ accepts is a normal subgroup of $H_{x_0}^*$. Indeed $f = g^v$ is a homomorphism, and if $z = g(t) \in H_{x_0}^*$ and $x = g(w)^v \in H_{x_0}^{*v}$, then $z^{-1}xz = g(t^{-1}wt)^v \in H_{x_0}^{*v}$. In this case the elements of $\hat{Q}$ are multiples of the least positive number $q_0 \in \hat{Q}$. Indeed if $\hat{q} \in \hat{Q}$ and $\hat{q} = q_0 q_1 + r$, $q_0 > r \geq 0$, then $x_a^r = x_a^{\hat{q}} x_a^{-q_0 q_1} \in L_{x_0}$ since L_{x_0} is a group. This implies that $r = 0$ because q_0 is minimal in $\hat{Q}$. A similar

argument shows that q_0 is a factor of v. Finally $q_0 > 1$ otherwise $\hat{Q} = Q$ and $x_a \in L_{x_0}$.

5. Let $f = g^v$ as in 4. Suppose that v is a prime with $|x|^a \leq v \leq |x|^b$, a, b constants. Then $\hat{v} = 1$ and the lower soundness bound is $t \succ 1$. In general we get $t \succ 1$ when the least factor $p \neq 1$ of v is $\geq |x|^c$, c a constant, since $v/\hat{v} = q_0 \geq p$ and therefore $\log v/\hat{v} \succeq \log|x|$. Observe that k must be constant since v^k is polynomially bounded.

Extensions of the Guillou–Quisquater protocol [9] (g is the identity mapping) and the Beth [1] scheme (g is the mapping $x \to \alpha^x$, $\alpha \in GF(q)^*$) are of this type. For these protocols when v is chosen as above we get the optimal lower soundness bound $t \succ 1$. The soundness bound for the Fiat–Shamir protocol [5] is at the top end since $v = 2$ and $\log(v/\hat{v}) = 1$. For this scheme we must have $t \succ \log|x|$ for soundness.

6. The efficiency e of a protocol is related to its computational complexity, that is the number of computations that are required for its execution. For $\mathrm{P}_{Z_v}(g^v)$ the number of operations for t rounds is $O(t \log v) = O(t \log|x|)$ (we only count multiplications). So the lower efficiency bound is $e \succ \log|x|$. With the Fiat–Shamir scheme the number of operations required is $O(kt) = O(t \log|x|)$. So the efficiency bound is $e \succ \log|x| \cdot \log|x| = (\log|x|)^2$. Furthermore the lower bound for the number of bits communicated with $\mathrm{P}_{Z_v}(g^v)$ is $|x| \log|x|$ (taking the input length to be $|x| \log|x|$) compared to $|x|(\log|x|)^2$ for the Fiat–Shamir scheme.

The essential difference between the two protocols is that when $x \notin L$, in the former $x_1^q \notin L$ for all $q \in Q \backslash \{0\}$, whereas in the latter some of the products $x_1^{q_1} \cdots x_k^{q_k}$, $q \in Q^k \backslash \{0\}$ may belong to L (but not more than $2^{(k-1)}$). In both cases the order of the query set is polynomial in $|x|$.

Zero-knowledge

Definition An interactive proof is *zero-knowledge* if for any verifier B, history h and input $x = x_0 x_1 \cdots x_k \in L$, there is an expected polynomial time probabilistic Turing machine M_B, which can simulate the view V

of the communication that B has with an indistinguishable probability distribution.

With zero-knowledge the prover A is honest whereas the verifier B may cheat. B will cheat in the hope of extracting additional information about the input x: a zero-knowledge protocol should prevent this. For 3-step protocols the only contribution that B makes is to send a query q in Step 2. When B is honest this is selected with uniform distribution, but when B cheats a different distribution is used. This distribution must be completely determined be the state of B at that particular stage (B is a Turing machine). We assume that B has kept a history h from previous interactions with A (h must be polynomially bounded). So the i-th query is

$$q_i = F(x; h; \{Z_j, q_j, Y_j\}_{j=1}^{i-1}; Z_i; R), \tag{1}$$

where F is a deterministic function, $\{Z_j, q_j, Y_j\}_{j=1}^{i-1}$ are the transcripts of the completed iterations, Z_i is the current commitment and R is a bit string from the random tape of B. R is long enough to provide the randomness needed for a complete round of t iterations. The query q_i is a random variable whose value is given by the function F. The view of B is random variable

$$V = (R; \{Z_j, q_j, Y_j\}_{j=1}^{t}).$$

We first consider the quadratic residuosity proof and show that it is zero-knowledge. Then we shall show how this proof can be extended to the protocols described earlier.

The protocol P_{GMR}

When $x_1 \in L$ then it is easy for the simulator to construct valid transcripts $\{Z, q, Y\}$ which are indistinguishable from the real ones. For example when $X \in_R Z_{x_0}^*$:

If $q=0$ then $\{Z \equiv X^2 (\mathrm{mod}\, x_0), q=0, Y=X\}$ is a valid,
If $q=1$ then $\{Z \equiv X^2 x_1 (\mathrm{mod}\, x_0), q=1, Y=X\}$ is a valid.

In both cases, since $x_1 \in L$, Z has uniform distribution in QR_n. So M_B only needs to select q' with a distribution which is indistinguishable from the distribution of the verifier's query q. To find the correct

distribution M_B tosses a random coin and 'matches' its distribution to the distribution of q by probing B and resetting B whenever the outcome is different. First M_B selects from its random tape a bit string R' of the appropriate length, which it outputs. Then it proceeds to output the transcripts. Suppose that M_B has output the transcripts $\{Z'_1, q'_1, Y'_1\}, \ldots, \{Z'_{i-1}, q'_{i-1}, Y'_{i-1}\}$. The next transcript is obtained as follows. M_B selects a random bit q'_i and probes B by sending in Step 1 the commitment $Z'_i = X'^2_i x_1^{q'_i}$, where $X'_i \in_R Z^*_{x_0}$. If the query of B, q_i, is the same as q'_i, or equivalently if q'_i is the value of the deterministic function (1) for the given arguments, then M_B will output $\{Z'_i, q'_i, Y'_i{=}X'_i\}$. Otherwise it resets B to the previous state and tosses a coin again. The expected number of probes per iteration is 2, so for a complete round the expected number of probes is $2t$. Since we are assuming that the number of iterations t is polynomially bounded, M_B, which is an expected polynomial time machine, will output a random variable V' which is indistinguishable from V by any polynomial time computation. In this case it is easy to see that V and V' are actually identical. We say that $\mathrm{P}_{\mathrm{GMR}}$ is a *perfect* zero-knowledge proof.

The protocol $\mathrm{P}_{Q^k}(\mathrm{f})$

The proof that this protocol is zero-knowledge is essentially the same as that for $\mathrm{P}_{\mathrm{GMR}}$. Given $(q_i) = (q_1, \ldots, q_k)$, M_B prepares the transcripts $\{Z'_i = f(X'_i)\, x_1^{q_{i1}} \cdots x_k^{q_{ik}}, (q_{ij}), Y{=}X'_i\}$, where $X'_i \in_R G$. For this protocol the expected number of probes for a complete round is $t\,|Q|^k$. Provided that $|Q|^k$ and t are polynomially bounded, M_B can obtain the appropriate distribution. Again $\mathrm{P}_{Q^k}(\mathrm{f})$ is perfect zero-knowledge.

We now state the main results of this section.

Theorem *The protocols* $\mathrm{P}_{Q^k}(\mathrm{f})$ *contain a class of zero-knowledge proofs* $\mathrm{P}_{Z_v{}^k}(g^v)$ *for which,*

(i) g *is a homomorphism,* $v = |Q|$ *is a prime,* $|x|^a \leq v \leq |x|^b$, a, b *constants* ($|x|$ *is the length of the input*), k *is a constant, and the number of iterations* t *is bounded by* $t \succ 1$;

(ii) *the efficiency* e *is bounded by* $e \succ \log |x|$;

(iii) *on input $x \in L$, the protocols induce the groups $L_{x_0} \subset L$, $G^*_{x_0}$ and $H^*_{x_0} = g(G^*_{x_0})$. L_{x_0} is a normal subgroup of $H^*_{x_0}$, the quotient group $H^*_{x_0}/L_{x_0}$ is an elementary v-group and $L_{x_0} = H^{*v}_{x_0}$.*

Proof This follows from the Remarks 1 – 6. Observe that the bounds are asymptotic and exclusive.

3. Proofs of knowledge

Let s, x be such that $|s|$ is polynomially related to $|x|$ and let $P(s, x)$ be a relation (or predicate) whose truth value can be checked in polynomial time in $|x|$. With proofs of knowledge the prover wants to prove that he knows a witness s which is such that $P(s, x)$ (is true). Let L be the language of all the x for which such witnesses exist. We will assume that membership in L can be checked in polynomial time but that (for cryptographic applications) given x it is 'hard' to find an s such that $P(s, x)$. The verifier can check for himself that $x \in L$ and is not interested in membership, but he cannot find s. With proofs of knowledge the prover A proves that he *knows* an s such that $P(s, x)$, not merely that such an s exists [4]. We will assume that both A and B are polynomially bounded and that A has an additional 'private knowledge' tape.

Definition The interactive protocol (A, B) is a proof of knowledge of the relation $P(\cdot, \cdot)$ [4] if,

(i) *Completeness* For any set $x = x_0 x_1 \cdots x_k$ ($|x|$ sufficiently large) for which $P(\cdot, x_0 x_j)$ is satisfiable for all $j > 0$ we have: when the input is x and A knows elements s_j (*e.g.*, the s_j are given on his knowledge tape) such that $P(s_j, x_0 x_j)$ for all $j > 0$ then B accepts with overwhelming probability.

(ii) *Soundness* There exists a polynomial time probabilistic Turing machine, the *interrogator* M_A, which is such that, for any input $x = x_0 x_1 \cdots x_k$ ($|x|$ sufficiently large): if B accepts the proof of A with non-negligible probability then M_A will succeed with overwhelming probability in extracting from A elements s_j for which $P(s_j, x_0 x_j)$ for all $j > 0$.

As Feige–Fiat–Shamir point out this definition captures the notion of

knowledge in a general sense. It may be that the prover A is not aware of this knowledge. For example A may succeed in convincing B with the help of an oracle. It should be possible for the interrogator to obtain from A (and thus indirectly from the oracle) the relevant knowledge. M_A achieves this by probing and resetting A. This forces A to answer all the queries *while remaining in the same state.*

The Feige–Fiat–Shamir scheme

Let $x_0 = pq$, where p, q are distinct primes such that $p \equiv q \equiv 3 \,(\mathrm{mod} 4)$. Then -1 is a quadratic non-residue mod x_0 with Jacobi symbol $(-1 \,|\, x_0) = +1$. (The Jacobi symbol $(y \,|\, x_0)$ of a number y mod x_0 is defined as the product $\prod_{i=1}^{s}(y \,|\, p_i)$, where $x_0 = \prod_{i=1}^{s} p_i$, p_i primes; $(y \,|\, p_i)$ is $+1$ if y is a quadratic residue mod p_i, -1 if y is a quadratic non-residue and 0 otherwise. It is a well known fact that the Jacobi symbol $(y \,|\, x_0)$ can be computed in polynomial time.)

The Feige–Fiat–Shamir proof [4] is a proof of knowledge of s_j such that $s_j^2 x_j \equiv \pm 1 \,(\mathrm{mod}\, x_0)$. Such an s_j exists if and only if $(x_j \,|\, x_0) = 1$. Since checking the Jacobi symbol is a polynomial time operation, there is no point for A to prove that s_j exists. Proving however knowledge of s_j is a different matter: it is 'hard' to find s_j. With zero-knowledge proofs of knowledge, A wants to prove that he knows s_j without revealing any information *at all* [4]. For the Feige–Fiat–Shamir protocol we have:

Protocol $\mathrm{P_{FFS}}$: input $x = x_0 x_1 \cdots x_k$, $k = \log |x_0|$.

Step 1 A sends B: $Z \equiv \pm r^2 (\mathrm{mod}\, x_0)$, $r \in_R Z_{x_0}^*$,

Step 2 B sends A: $q = (q_i) \in_R \{0,1\}^k$,

Step 3 A sends B: $Y \equiv r \prod_{j=1}^{k} s_j^{q_j} (\mathrm{mod}\, x_0)$, where $s_j \in Z_{x_0}$ are such that $s_j^2 x_j \equiv \pm 1 \,(\mathrm{mod} x_0)$ for all $j > 0$.

The input x is valid when all the x_i, $i > 0$, are units mod x_0, and Y is correct when $Y \not\equiv 0 \,(\mathrm{mod} x_0)$ and $Z \equiv \pm Y^2 \prod_{j=1}^{k} x_j^{q_j} (\mathrm{mod}\, x_0)$. Observe that the natural equality '$\equiv (\mathrm{mod}\, x_0)$' of Z_{x_0} has been replaced by the equivalence $\sim$ for which $a \sim b$ if and only if $a \equiv \pm b \,(\mathrm{mod}\, x_0)$. The equivalence classes of $\sim$ are the elements of Z_m, where $m = \frac{1}{2}(x_0 + 1)$,

and multiplication is given by $a \cdot b = ab$ if $ab < m$ and $a \cdot b = x_0 - ab$ otherwise. This is a particular case of the protocol $\mathrm{P}_{Z_2{}^k}(\mathrm{f})$ for which membership in the language L can be checked in polynomial time and for which the condition for correctness of answers is slightly weaker.

The protocol $\mathrm{P}_{Z_2{}^k}(\mathrm{f})$

As in the case of proofs of membership, completeness of $\mathrm{P}_{Z_2{}^k}(\mathrm{f})$ is related to the algebraic structure of the setting. We must have (i) almost all elements of G_{x_0} are units, and (ii) $f(r\prod s_j^{q_j}) \cdot \prod x_j^{q_j} = f(r)$, where $f(s_j)x_j = 1$, $s_j \in G_{x_0}$, $j = 1, \ldots, k$ for (almost) all $r \in G_{x_0}$ and $(q_i) \in Q^k$. We clearly have (ii) when f is a homomorphism of (the monoid) G_{x_0}. Furthermore $\mathrm{P}_{2^k}(\mathrm{f})$ is sound if and only if,

$$|x|^c \succeq t \succ \log|x|, \quad c \text{ a constant.}$$

The proof is similar to that in §2. We assume that B accepts with non-negligible probability ϵ and show that the execution tree T of $\mathrm{P}_{Z_2{}^k}(\mathrm{f})$ has at least one heavy vertex. From this we obtain for every j, elements Y_0, Y_1 such that $f(Y_0^{-1}Y_1)x_j = 1$. The witnesses are the $s_j = Y_0^{-1}Y_1$. With proofs of knowledge the mere existence of witnesses is not sufficient: we must also show that the interrogator can obtain these witnesses in polynomial time. So we have to show how M_A will find such vertices. We use the technique described in [4] for $\mathrm{P}_{\mathrm{FFS}}$ and extend it when necessary. We claim that at least one half of all vertices of T at some level i are heavy. Indeed if this were not the case then the ratio of the number of vertices at any two successive levels of T would be $\leq \frac{1}{2}2^k + \frac{1}{2}2^{k-1} = \frac{3}{4}2^k$, and the fraction of leaves that survive the truncation would be $\leq (\frac{3}{4})^t$, which is negligible. This contradicts our earlier assumption that B accepts with non-negligible probability. To find a heavy vertex M_A searches 'blindly' along random branches of the untruncated tree. At each vertex M_A probes and resets the prover 2^k times to obtain the degree of the vertex. Since a non-negligible fraction ϵ of branches survive the truncation and since half of the surviving branches contain a heavy vertex, M_A will find such a vertex in polynomial time. More specifically, if M_A checks $2|x|/\epsilon$ branches then it will find a suitable branch with overwhelming probability $1 - (1 - \frac{\epsilon}{2})^{2|x|/\epsilon} \approx 1 - e^{-|x|}$. Thus after at most $2|x|\,t\,2^k/\epsilon$ probes

of A, M_A will find elements s_j such that $f(s_j)x_j = 1$ for all j. Since t, 2^k and $\frac{1}{\epsilon}$ are polynomially bounded, the number of probes required is polynomially bounded.

The protocol $P_{Z_v^k}(g^v)$

We assume that $H_{x_0} = g(G_{x_0})$ and that for any $x \in H_{x_0}^*$ one can compute in polynomial time a $y \in G_{x_0}$ such that $x = g(y)$ (so g is not a one-way function).

The conditions for completeness are as for $P_{Z_2^k}(f)$. The protocol is a sound proof of knowledge if and only if,

$$|x|^c \succeq t \succ \log|x| / \log(v/\hat{v}), \quad c \text{ a constant.}$$

Here $\hat{v} = v/p$, where p is the least prime factor of v. As in §2 we consider prolific vertices (of degree $> \hat{v}v^{k-1}$). We assume that B accepts with non-negligible probability ϵ and, in this case, show that at some level of the execution tree the fraction of such vertices is non-negligible (rather than $\frac{1}{2}$). To prove this we first observe that the ratio of the number of vertices $\frac{v_{i+1}}{v_i}$ at two successive levels of the execution tree T is $\geq (\hat{v}v^{-1} + |x|^{-d})\, v^k$, for some i and for some constant d. The proof is by contradiction. Suppose not, and that for all i and d, $\frac{v_{i+1}}{v_i} < (\hat{v}v^{-1} + |x|^{-d})\, v^k$. Then the fraction of leaves that survive the truncation $\prod \frac{v_{i+1}}{v_i}\, v^{-kt}$ is bounded by,

$$\begin{aligned}(\hat{v}v^{-1} + |x|^{-d})^t &< (\hat{v}v^{-1})^t + t\,|x|^{-d} + t^2|x|^{-2d} + \ldots \\ &< |x|^{-b} + |x|^{c-d+1} + |x|^{2(c-d+1)} + \ldots,\end{aligned}$$

b any constant, since $|x|^c \succeq t \succ \log|x| / \log(v/\hat{v})$. This expression is bounded by,

$$\begin{aligned}|x|^{-b} + |x|^{c-d+1} + |x|^{c-d+1} + \ldots &= |x|^{-b} + t\,|x|^{c-d+1} \\ &\leq |x|^{-b} + |x|^{2c-d+2},\end{aligned}$$

when d is large. Since b, d are *any* constants this is negligible and we have a contradiction. Next let m be the fraction of prolific vertices at level i. Then,

$$(\hat{v}v^{-1} + |x|^{-d})\, v^k v_i \;\leq\; v_{i+1} \;<\; (mv^k + (1-m)\hat{v}v^{k-1})\, v_i\,,$$

so that

$$|x|^{-d} \;<\; m(1-\hat{v}v^{-1}) \quad \text{or} \quad m \;> |x|^{-d}\, v(v-\hat{v})^{-1},$$

and therefore m is non-negligible. So if the interrogator checks blindly $|x|\,(|x|^d(v-\hat{v})\,v^{-1})/\epsilon$ branches of the untruncated execution tree it will find with overwhelming probability a prolific vertex. Again the number of probes is polynomial.

It remains to show that M_A can obtain a witness s_j for x_j, given the witnesses s_j^q for $x_j^q \in L_{x_0} = H_{x_0}^{*v}$, $q \in Q'$, $|Q'| > \hat{v}$. Observe that $\hat{v} = v/p$ is the largest proper factor of v, so the (additive) group $\bar{Q}$ generated by Q' must be the whole of $Q = Z_v$. Thus $1 \in \bar{Q}$. Since we have witnesses for all x_j^q, $q \in Q'$, we can obtain a witness s_j for x_j^1.

Unrestricted input zero-knowledge

An interactive proof of knowledge is *unrestricted input zero-knowledge* if it is zero-knowledge for any input x, *i.e.*, if the simulator can recreate the view of the verifier even when $x \notin L$ [4]. When $x \notin L$ an honest prover will halt the protocol. So the communication tapes are blank. The simulator recreates the communication that B has with honest provers. Since it can check in polynomial time if $x \in L$, it has no difficulty to recreate this communication: when $x \in L$ it proceeds as in §2 and when $x \notin L$ its tapes are empty.

The theorem in §2 applies here as well, provided that membership in L can be checked in polynomial time. So for the Beth scheme for which $L_q = GF(q)^*$ we assume that all the factors of $q-1$ are public. Then B can check that x has a v-th root in polynomial time.

Conclusion

We have shown that there is a class of interactive 3-step zero-knowledge protocols $\mathrm{P}_{Z_v}(g^v)$ for which the number of queries $v = |x|^b$ is a (large) prime, and for which the number of iterations has an optimal lower bound. The associated groups $L_{x_0}, H_{x_0}^*$ are such that $H_{x_0}^*/L_{x_0}$ is an elementary v-group. These protocols are the most efficient protocols presently available.

For cryptographic applications $f = g^v$ must be a one-way function. One such function is $f : Z_{x_0} \rightarrow Z_{x_0}; x \rightarrow x^v \,(\mathrm{mod}\, x_0)$, where x_0 is the product of two distinct primes. For random values, inversion of this function is equivalent to factoring x_0. This function has been used in the Guillou-Quisquater identification scheme. Another such function is $f : GF(q) \rightarrow GF(q)^*; x \rightarrow \alpha^{xv}$, where $\alpha \in GF(q)^*$. For random values, inversion of this function is equivalent to solving the discrete logarithm problem. This function has been used in the Beth identification scheme.

Acknowledgement: We wish to thank Yvo Desmedt whose enthusiastic encouragement and helpful discussions made this research possible.

References

[1] T. Beth. A Fiat–Shamir–like authentication protocol for the El–Gamal–scheme. In C. G. Günther, editor, *Advances in Cryptology, Proc. of Eurocrypt'88 (Lecture Notes in Computer Science 330)*, pp. 77–84, Springer–Verlag, May 1988. Davos, Switzerland.

[2] M. Burmester, Y. Desmedt, F. Piper and M. Walker. A general zero-knowledge scheme. In *Advances in Cryptology, Proc. of Eurocrypt'89 (Lecture Notes in Computer Science)*, Springer-Verlag, May 1989. Houthalen, Belgium. *To appear.*

[3] M. Burmester, Y. Desmedt, F. Piper and M. Walker. A Meta Zero-Knowledge Scheme. *To appear.*

[4] U. Feige, A. Fiat, and A. Shamir. Zero knowledge proofs of identity. In *Journal of Cryptology*, 1(2), pp. 77–94, May 25–27, 1988.

[5] A. Fiat and A. Shamir. How to prove yourself: Practical solutions to identification and signature problems. In A. Odlyzko, editor, *Advances in Cryptology, Proc. of Crypto'86 (Lecture Notes in Computer Science 263)*, pp. 186–194, Springer–Verlag, 1987. Santa Barbara, California, U. S. A., August 11–15.

[6] A. Fiat and A. Shamir. Unforgeable proofs of identity. In *Securicom 87*, pp. 147 – 153, March 4–6, 1987. Paris, France.

[7] M.R. Garey and D.S Johnson. Computers and Intractability, A guide to the theory of NP-Completeness. W.H.Freeman and Co, New York, 1979.

[8] S. Goldwasser, S. Micali, and C. Rackoff. The knowledge complexity of interactive proof systems. *Siam J. Comput.*, 18(1), pp. 186–208, February 1989.

[9] L.C. Guillou and J.–J. Quisquater. A practical zero-knowledge protocol fitted to security microprocessor minimizing both transmission and memory. In C. G. Günther, editor, *Advances in Cryptology, Proc. of Eurocrypt'88 (Lecture Notes in Computer Science 330)*, pp. 123–128, Springer–Verlag, May 1988. Davos, Switzerland.

[10] A. Shamir. The search for provably secure identification schemes. In *Proceedings of the International Congress of Mathematicians*, pp. 1488–1495, August 3–11, 1986.

University of Piraeus
Piraeus, Greece

Department of Mathematics,
RHBNC, University of London
Egham, Surrey TW20
OEX, England