On Diophantine Complexity and Statistical Zero-Knowledge Arguments

Helger Lipmaa

Laboratory for Theoretical CS, Department of CS&E
Helsinki University of Technology, P.O.Box 5400, FIN-02015 HUT, Espoo, Finland
helger@tcs.hut.fi

Abstract. We show how to construct practical honest-verifier statistical zero-knowledge *Diophantine* arguments of knowledge (HVSZK AoK) that a committed tuple of integers belongs to an arbitrary language in bounded arithmetic. While doing this, we propose a new algorithm for computing the Lagrange representation of nonnegative integers and a new efficient representing polynomial for the exponential relation. We apply our results by constructing the most efficient known HVSZK AoK for non-negativity and the first constant-round practical HVSZK AoK for exponential relation. Finally, we propose the outsourcing model for cryptographic protocols and design communication-efficient versions of the Damgård-Jurik multi-candidate voting scheme and of the Lipmaa-Asokan-Niemi $(b+1)$st-price auction scheme that work in this model.
Keywords. Arguments of knowledge, Diophantine complexity, integer commitment scheme, statistical zero knowledge.

1 Introduction

A set $S \subset \mathbb{Z}^n$ is called *Diophantine* [Mat93], if it has a *representing polynomial* $\mathfrak{R}_S \in \mathbb{Z}[X;Y]$, $X = (X_1, \dots, X_n)$ and $Y = (Y_1, \dots, Y_m)$, such that $\mu \in S$ iff for some witness $\omega \in \mathbb{Z}^m$, $\mathfrak{R}_S(\mu;\omega) = 0$. A seminal result of Matiyasevich from 1970 states that every recursively enumerable set is Diophantine. It has been an open question since [AM76], whether $\mathbf{D} \stackrel{?}{=} \mathbf{NP}$, where $\mathbf{D}$ is the class of sets S that have representing polynomials $\mathfrak{R}_S$, such that $\mu \in S$ iff for some polynomially long witness $\omega \in \mathbb{Z}^m$, $\mathfrak{R}_S(\mu;\omega) = 0$. One is also tempted to ask a similar question $\mathbf{PD} \stackrel{?}{=} \mathbf{P}$ about the "deterministic" version of class $\mathbf{D}$, the class $\mathbf{PD}$ that contains such languages for which the corresponding polynomially-long witnesses can be found in polynomial time. The gap in our knowledge in such questions is quite surprising; this is maybe best demonstrated by the recent proof of Pollett that if $\mathbf{D} \subseteq$ **co-NLOGTIME** then $\mathbf{D} = \mathbf{NP}$ [Pol03].

In this paper we take a more practice oriented approach. Namely, we are interested in the sets S with sub-quadratic (i.e., with length, sub-quadratic in the length of the inputs) witnesses. We propose representing polynomials with sub-quadratic , polynomial-time computable, witnesses for a practically important, although relatively small, class L_2 of languages of bounded arithmetic. (This class of languages includes many arithmetic and number-theoretic relations like $[\mu_3 = \max(\mu_1, \mu_2)]$, but also relations like [μ_2 is the ith bit of μ_1].) For this, we demonstrate that the exponential relation has a representing polynomial with polynomial-time computable sub-quadratic-length witnesses. This improves somewhat on the previous best result of [AM76]; differently from the latter, we will also give a self-contained proof of this result, and provide a precise complexity analysis. Our next contribution is a new algorithm for finding, given a positive integer μ, such integers $(\omega_1, \dots, \omega_4)$ that $\mu = \omega_1^2 + \dots + \omega_4^2$. This algorithm improves on the Rabin-Shallit algorithm [RS86].

While representing polynomials with short witnesses have independent interest in complexity theory [AM76], our work on this topic was motivated by cryptographic applications. Given an integer commitment scheme [FO99,DF02] with efficient arguments of knowledge for additive and multiplicative relations, one can argue (by using the methodology from [FO99]) in honest-verifier statistical zero-knowledge (HVSZK) that $f(\mu) = 0$, where μ is a tuple of committed integers. By following this methodology, one can design efficient argument systems for several important cryptographic problems. However, there has been no previous formal treatment of what happens if one extends this methodology (at least not when coupled with an *integer* commitment scheme) so as to enable the demonstration of knowledge of an auxiliary witness ω, for which $f(\mu; \omega) = 0$. A natural requirement here is that if the arguer convinces the verifier that she knows such an ω, the verifier will also be convinced that $\mu \in S$ where $f = \mathfrak{R}_S$ is the representing polynomial of S.

Thus, by using well-known cryptographic tools, one can construct polynomial-length three-round HVSZK arguments of knowledge that $\mu \in S$ for any $S \in \mathbf{D}$. However, these arguments can only be executed if the arguer knows the corresponding witness. If there is a polynomial-time algorithm to compute the witness from μ (that is, $S \in \mathbf{PD}$), then one will be able to argue that $\mu \in S$ for an arbitrary $\mu \in S$. If, additionally, the corresponding witnesses are sub-quadratic (as they are when $S \in L_2$) then by using the described methodology one can often improve upon previously known arguments of knowledge—either in efficiency, or by basing the arguments on weaker security requirements: namely, it is sufficient to require that the underlying integer commitment scheme is statistically hiding and computationally binding [FO99]. In particular, we use our new algorithm for finding the representation $\mu = \omega_1^2 + \dots + \omega_4^2$ to propose a new argument of knowledge for non-negativity of the committed integer. Compared to Boudot's protocol for the same problem [Bou00], this argument is conceptually much simpler, somewhat shorter, and offers perfect completeness.

After that, we propose a general model for cryptographic protocols that involve social or financial choices (e.g., voting or auctions). In this model one can implement any function from the class L_2 (e.g., maximum-finding in the case of auctions) by using sub-quadratic-length interaction. As [CGS97,DJ01,LAN02], our model uses a certain encoding function enc of the social choices together with a homomorphic public-key cryptosystem. As an example, in this model we can construct an efficient minimal-disclosure voting protocol where the talliers will only get to know the winning candidate.

Finally, we propose a few alternative constructions for the encoding function. Until now, one has mostly used the function $\mathsf{enc}(n) = a^n$, where a is an a priori fixed upper limit on the number of participants [CGS97,DJ01,LAN02]. We show that instead, one can use the function $\mathsf{enc}(n) = Z_a(n)$, where $Z_a(n)$ is the nth member of a certain Lucas sequence, to achieve otherwise exactly the same properties as in [DJ01,LAN02] but with correctness arguments of length $\Theta(\max(k, m \log a))$, where k is the security parameter, a is the maximal number of participants, and m is the number of possible social of financial choices (e.g., the number of different bids). This is $\Theta(\log m)$ times more efficient than the protocols from [DJ01,LAN02]. We also propose an efficient algorithm for computing $Z_a(n)$. Lucas sequences have definitely more applications in zero-knowledge proofs or arguments than described in this paper. We also demonstrate another approach that uses exponentiation as the encoding function.

Road-map. We introduce necessary preliminaries in Section 2. In Section 3, we prove that languages in L_2 have representing polynomials with sub-quadratic-length witnesses. In Section 4, we present a methodology that allows to apply our HVSZK arguments-of-knowledge together with homomorphic cryptosystems to a variety of cryptographic protocols. Finally, the appendix describes our simplifications and extensions to the Damgård-Fujisaki commitment scheme together with a new and efficient argument system for nonnegativity.

2 Preliminaries and Notation

We say that an algorithm f is *efficient* when f works in the probabilistic polynomial time with respect to the summatory length of its parameters; we denote the set of efficient algorithms by $\mathcal{EA}$. Let $\mathsf{bit}(x, i)$ denote the ith bit of x, i.e., $x = \sum_{i \geq 0} \mathsf{bit}(x, i) \cdot 2^i$. When D is a distribution (including the output distribution of some probabilistic algorithm) then $x \leftarrow D$ denotes the choice of a random element x according to D. We denote the uniform distribution over a set S also by S; that is, $x \leftarrow S$ means that x is chosen uniformly and randomly from S.

Bounded arithmetic. Bounded arithmetic is a first-order theory of the natural numbers with non-logical symbols 0, σ, $+$, $\cdot$, $\leq$, $\dot{-}$, $\lfloor x/2 \rfloor$, $|x|$, $\mathsf{MSP}(x, i)$ and $\sharp$. The symbols 0, $\sigma(x) := x + 1$, $+$, $\cdot$, and $\leq$ have their usual meaning. Other operations are defined as $x \dot{-} y := \max(x - y, 0)$, $|x| := \lfloor \log_2(x + 1) \rfloor$, $\mathsf{MSP}(x, i) := \lfloor x/2^i \rfloor$ and $x \sharp y := 2^{|x| \cdot |y|}$. For our purposes we adapt a slightly modified definition of bounded arithmetic where the underlying domain is $\mathbb{Z}$ instead of $\mathbb{N}$. We denote by L_2 the set of terms of the quantifier-free bounded arithmetic (over $\mathbb{Z}$).

One can express a large number of relations in L_2. Many familiar predicates (like $[\mu_1 > \mu_2]$, $[\mu$ is a perfect square$]$, $[\mu_2 = \mathsf{bit}(\mu_1, i)]$) are known to belong to L_2. They can be readily found from the literature.

Lucas sequences. All nonnegative integral solutions (x, y) of the equation $x^2 - axy - y^2 = 1$ are either equal to $(Z_a(n+1), Z_a(n))$ or $(Z_a(n), Z_a(n+1))$, $n \geq 0$, where $Z_a(n)$ (that we mostly denote by $a^{[\![n]\!]}$) can be computed by using the next recurrent identities [Mat93]: $Z_a(0) := 0$, $Z_a(1) := 1$, and $Z_a(n+2) := aZ_a(n+1) - Z_a(n)$ for $n \geq 0$. Thus, $\{Z_a(n)\}_{n \in \mathbb{N}}$ is a Lucas sequence. Another important property of $Z_a(n)$ is that when $a > 2$ and $n > 0$ then $(a-1)^n \leq Z_a(n+1) \leq a^n$. The next variant of the Russian peasant algorithm can be used to efficiently compute the pair $(Z_a(n+1), Z_a(n))$:

Lemma 1. *The next algorithm computes $(Z_a(n+1), Z_a(n))$ from (a, n) by doing $\approx 3 \cdot \log_2 n$ two-variable multiplications in average:*

1. $\ell := \lfloor \log_2 n \rfloor; z := 1; z' := 0$
2. **for** $i := \ell$ **downto** 0 **do**
   - $t := z$; **if** $\mathsf{bit}(n, i) = 1$ **then** $z := z(at - 2z'); z' = t^2 - z'^2$
     else $z := t^2 - z'^2; z' = z'(2t - az')$;
3. Return (z, z').

Proof. Follows from the identities $Z_a(2n) = Z_a(n)(2Z_a(n+1) - aZ_a(n)) = Z_a(n)(aZ_a(n) - 2Z_a(n-1))$ and $Z_a(2n+1) = Z_a^2(n+1) - Z_a^2(n)$. $\square$

While a similar $O(\log n)$-time algorithm for Lucas sequences is described, for example, in [JQ96], the algorithm presented there works for somewhat different sequences and requires $4.5(\log_2 n + O(1))$ multiplications. Log-time algorithms for Lucas sequences have been known at least since [Wil82].

Arguments of knowledge. For bit-strings a and μ, and predicate $Q(\cdot)$, we denote by $\mathrm{AK}(Q(a, \mu))$ a three-round honest-verifier statistical zero-knowledge (HVSZK) two-party argument of knowledge (AoK) that given a value a (known to both parties), the arguer knows an integer parameter μ, such that the predicate $Q(a, \mu)$ is true. We always denote the values, knowledge of which has to be proved, by Greek letters; the scope of such variables lies within a single AoK. The symbol ω will always denote an auxiliary witness. As an example, $\mathrm{AK}(y = E_K(\mu; \rho) \wedge \mu^2 = \omega)$ denotes a HVSZK AoK that given a ciphertext y and a public key K, the arguer knows a plaintext μ and a randomness ρ such that $y = E_K(\mu; \rho)$ and μ is a perfect square. Our protocols will be AoK-s in the model of Damgård and Fujisaki [DF02]. An important property of the zero-knowledge arguments is that the verifier cannot extract (significant) additional information even if he is given infinite time. This makes AoK-s more attractive than proofs of knowledge in applications where privacy of the arguer is paramount. A HVSZK argument system can be made non-interactive by using the Fiat-Shamir heuristic [FS86] in the random-oracle model. The converted argument is also secure against malicious verifiers. There exist alternative methods for converting a HVSZK argument into a a full interactive zero-knowledge argument that do not use random oracles. For the purpose of Fiat-Shamir heuristic, we introduce a random oracle $H : \{0,1\}^* \rightarrow \{0,1\}^{2k}$.

Integer commitment schemes. A *secure* (in the sense of being statistically hiding and computationally binding) *integer commitment scheme* C allows the arguer $A \in \mathcal{EA}$ to commit to an integer $m \in \mathbb{Z}$, so that (1) for uniform and random r_1, r_2 and any $m_1, m_2 \in \mathbb{Z}$, the distributions $C_K(m_1; r_1)$ and $C_K(m_2; r_2)$

are statistically close; and (2) it is intractable for A to find m_1, m_2, r_1 and r_2, such that $m_1 \neq m_2$ but $C_K(m_1; r_1) = C_K(m_2; r_2)$. Known integer commitment schemes include [FO99,DF02]; the security of both integer commitment schemes bases on some reasonable security assumptions that seem to be satisfied by class groups and a large variety of the RSA groups. We will give a description of a simplified Damgård-Fujisaki scheme in Appendix A. The main simplifications are: (a) In revealing phase, it is sufficient for the committer to send the pair (m, r) instead of the triple (m, r, b) and (b) The underlying root assumption is modified to have the following, simpler, form: given random y, it is hard to produce such (x, d, e) that $y^e = x^{de}$ and e is reasonably small.

By using a secure integer commitment scheme, one can build an HVSZK argument system for different relations between committed integers μ_i. In all such argument systems, arguer and verifier have to fix, for every i, an a priori upper bound M_i to input μ_i [FO99,DF02]. The argument system is guaranteed to have the statistical zero-knowledge property only if $|\mu_i| < M_i$. Therefore, in such protocols the interaction length depends on $\log_2 M_i$, and thus it is beneficial to precompute as precise values of M_i as feasible. Certainly it must be the case that $\log_2 M_i = k^{O(1)}$. Additionally, we will describe in Appendix A how to commit to an integer tuple (and not just to an integer). The resulting *integer tuple commitment scheme* can be used to construct more efficient arguments of knowledge than the Damgård-Fujisaki commitment scheme by itself.

Diophantine complexity. Based on the earlier work of Davis, Putnam and Robinson, Matiyasevich proved in 1970 [Mat70] that every recursively enumerable set is Diophantine (this important result is known as the DPRM theorem), solving thus negatively Hilbert's tenth problem from year 1900. This work on the Hilbert's tenth problem has had many interesting consequences. See [Mat93] for a representation of main results of this work and related history. In 1976, Adleman and Manders [AM76] proposed the next complexity-theoretic class $\mathbf{D}$ of sets: $S \in \mathbf{D}$ iff there exists a *representing polynomial* $\mathfrak{R}_S$, such that $\mu \in S \iff (\exists \omega)[|\sum_i \omega_j| = |\sum_i \mu_i|^{O(1)} \wedge \mathfrak{R}_S(\mu; \omega) = 0]$. Obviously, $\mathbf{D} \subseteq \mathbf{NP}$. On the other hand, Adleman and Manders showed that several $\mathbf{NP}$-complete problems belong to the class $\mathbf{D}$ and, based on that, conjectured that $\mathbf{D} = \mathbf{NP}$. Their conjecture was later implicitly supported by Jones and Matiyasevich [JM84] who proved that $\mathbf{D} = \mathbf{NP}$ iff the set $\{(\mu_1, \mu_2) : \mu_1 \leq_2 \mu_2\}$ belongs to $\mathbf{D}$ (Here, $\mu_1 \leq_2 \mu_2$ iff $\mathsf{bit}(\mu_1, i) \leq \mathsf{bit}(\mu_2, i)$ for every i.) and by Pollet [Pol03], who recently showed that when $\mathbf{co\text{-}NLOGTIME} \subseteq \mathbf{D}$ then $\mathbf{D} = \mathbf{NP}$. The gap between $\mathbf{co\text{-}NLOGTIME}$ and $\mathbf{NP}$ is wide and thus, as expected, not much is known about the actual power of the class $\mathbf{D}$.

In the following, let M_i be some a priori upper bound on the length of the input μ_i and let W_j be a similar upper bound on the witness ω_j that holds when the lengths of the input μ_i never exceed the values M_i. Let $M := \max_i M_i$ and $W := \max_j W_j$; note that the value W is a function of M and $\mathfrak{R}_S$. Since the number of witnesses m and the degree of the polynomial $\mathfrak{R}_S$ do not depend on the input size M, the total size of inputs to the representing polynomial will be $\Theta(M + W)$. Now, $S \in \mathbf{D}$ if for some representing polynomial $\mathfrak{R}_S$, $W = M^{O(1)}$

and therefore, the Adleman-Manders conjecture says that $S \in \mathbf{NP}$ iff for some polynomial $\mathfrak{R}_S$, $\mu \in S \iff (\exists\omega)[\mathfrak{R}_S(\mu;\omega) = 0 \wedge W = M^{O(1)}]$.

In the standard definition of Diophantine sets [Mat93] only nonnegative witnesses are admitted. The classes $\mathbf{D}$ and $\mathbf{PD}$ do not change when we modify their definitions to allow negative integer witnesses, since $\mu \in S \iff (\exists\omega,\omega' \in \mathbb{N}_0^m)[\mathfrak{R}_S(\mu;\omega_1-\omega'_1,\dots,\omega_m-\omega'_m) = 0]$. On the other hand, if S has a representing polynomial $\mathfrak{R}'_S(\mu;\omega)$ with nonnegative witnesses, then S can be represented by $\mathfrak{R}'_S(\sum_{i=1}^4 \mu_{1i}^2,\dots,\sum_{i=1}^4 \mu_{ni}^2;\sum_{i=1}^4 \omega_{1i}^2,\dots,\sum_{i=1}^4 \mu_{mi}^2)$; the latter follows from a classical theorem of Lagrange (see also Thm. 2). For convenience, we will implicitly assume that all the variables belong to $\mathbb{Z}$ (and not to $\mathbb{N}_0$).

3 Bounded Arithmetic is in PD

First, let us introduce a new complexity class $\mathbf{PD}$ that is a Diophantine analogue of $\mathbf{P}$. Namely, we say that $S \in \mathbf{PD}$ iff there is a polynomial $\mathfrak{R}_S \in \mathbb{Z}[X]$, such that (1) there exists an efficient *witness algorithm* $\mathfrak{P}_S \in \mathcal{EA}$, such that if $\mu \in S$ then $\mathfrak{R}_S(\mu;\mathfrak{P}_S(\mu)) = 0$; (2) if $\mu \notin S$ then for any ω with $|\omega| = |\mu|^{O(1)}$, $\mathfrak{R}_S(\mu;\omega) \neq 0$. Recently, Pollett proved that all sets in L_2 belong to $\mathbf{D}$ [Pol03]. We extend this to a proof that all sets in L_2 belong to $\mathbf{PD}$.

Theorem 1. *All L_2-terms belong to $\mathbf{PD}$, with $W = M^{2-\varepsilon}$ for $\varepsilon > 0$.*

Proof. To show that L_2-terms belong to $\mathbf{PD}$, we will first show that all non-logical basic relations of bounded arithmetic belong to $\mathbf{PD}$. Thereafter, we show how to implement the Boolean operators that connect them by using induction on the structure of formulas. Clearly, the first four basic non-logical symbols (0, σ, $+$, $\cdot$) have representing polynomials with no auxiliary witnesses. (For example, the predicate $[\mu_2 = \sigma(\mu_1)]$ is represented by the polynomial $\mathfrak{R}_S(\mu_1,\mu_2) = \mu_2 - \mu_1 - 1$.) The representing polynomial for $\leq$ can be constructed by using the representing polynomial for non-negativity, see Thm. 2.

The Boolean operators $\wedge$, $\vee$ and $\neg$ can be dealt with as follows. Let $S, S' \in \mathbf{PD}$ have representing polynomials $\mathfrak{R}_S$ and $\mathfrak{R}_{S'}$ and witness algorithms $\mathfrak{P}_S$ and $\mathfrak{P}_{S'}$. Then $\mathfrak{R}_{S\cup S'}(\mu;\omega,\omega') = \mathfrak{R}_S(\mu;\omega)\cdot\mathfrak{R}_{S'}(\mu;\omega')$, $\mathfrak{R}_{S\cap S'}(\mu;\omega,\omega') = \mathfrak{R}_S(\mu;\omega)^2 + \mathfrak{R}_{S'}(\mu;\omega')^2$ and $\mathfrak{P}_{S\cup S'}(\mu) = \mathfrak{P}_{S\cap S'}(\mu) = (\mathfrak{P}_S(\mu),\mathfrak{P}_{S'}(\mu))$. Therefore, if $S_1 \in \mathbf{X}$ then also $S_1 \cup S_2, S_1 \cap S_2 \in \mathbf{X}$ for $\mathbf{X} \in \{\mathbf{D},\mathbf{PD}\}$. One can establish that $\neg P(\cdot)$ belongs to $\mathbf{PD}$ by induction, assuming that $P(\cdot)$ belongs to $\mathbf{PD}$ and then studying the case of every possible main connective of P separately. (This can introduce some new witnesses.) As an example, $[\mu_1 \neq \mu_2] \equiv [(\mu_1 < \mu_2) \vee (\mu_2 > \mu_1)]$.

Three of the remaining operations can now be defined as $[\mu_3 = \mu_1 \dot{-} \mu_2] \equiv [((\mu_1 - \mu_2 = \mu_3) \wedge (\mu_1 \geq \mu_2)) \vee (\mu_3 = 0 \wedge \mu_1 < \mu_2)]$, $[\mu_2 = \lfloor \mu_1/2 \rfloor] \equiv [(\mu_1 = 2\omega_1) \vee (\mu_1 = 2\omega_1 + 1)]$ and $[\mu_2 = \mathsf{MSP}(\mu_1,i)] \equiv [(\mu_1 = 2^i \cdot \mu_2 + \omega \wedge \omega \in [0, 2^i - 1])]$. Note that only the last three operations need a nonempty witness ω, with $W = O(M)$. That $[\mu_3 = \mu_1^{\mu_3}]$ is in $\mathbf{PD}$ follows from Thm. 3. Finally, $[\mu_2 = |\mu_1|] \equiv [\omega_1 = 2^{\mu_2} \wedge \omega_1 \leq 2(\mu_1+1) \wedge (\mu_1+1) < \omega_1]$. Thus, $[\mu_3 = \mu_1 \sharp \mu_2] \equiv [(\omega_1 = |\mu_1|) \wedge (\omega_2 = |\mu_2|) \wedge (\mu_3 = 2^{\omega_1\cdot\omega_2})]$. The theorem follows from Thm. 3,

Algorithm 1 Algorithm for computing an Lagrange representation $\mu = \omega_1^2 + \omega_2^2 + \omega_3^2 + \omega_4^2$, $\omega \leftarrow \mathsf{Lagrange}(\mu)$

1. Write μ in the form $\mu = 2^t(2k+1)$, where $t, k \geq 0$.
2. If $t = 1$, then
   (a) Choose random $\omega_1 \leq \sqrt{\mu}$, $\omega_2 \leq \sqrt{\mu - \omega_1^2}$, such that exactly one of ω_1, ω_2 is even. Let $p \leftarrow \mu - \omega_1^2 - \omega_2^2$. Now $p \equiv 1 \pmod 4$.
   (b) Hoping that p is prime, try to express $p = \omega_3^2 + \omega_4^2$ as follows: First, find a solution u to the equation $u^2 \equiv -1 \pmod p$. Apply the Euclidean algorithm to (u, p), take the first two remainders that are less than $\sqrt{p}$ to be ω_3 and ω_4. If $p \neq \omega_3^2 + \omega_4^2$, p was not prime, so go back to step 2a.
   (c) Return $(\omega_1, \dots, \omega_4,)$ as the representation.
3. If t is odd but not 1, find a representation $(\omega_1, \dots, \omega_4)$. Return $(s\omega_1, \dots, s\omega_4)$, where $s = 2^{(t-1)/2}$.
4. If t is even, find a representation $\omega_1^2 + \omega_2^2 + \omega_3^2 + \omega_4^2$ for $2(2k+1)$ by step 2. Then convert this to a representation for $(2k+1)$ as follows: Group ω_1, ω_2, ω_3, ω_4 so that $\omega_1 \equiv \omega_2 \pmod 2$ and $\omega_3 \equiv \omega_4 \pmod 2$. Return $(s(\omega_1 + \omega_2), s(\omega_1 - \omega_2), s(\omega_3 + \omega_4), s(\omega_3 - \omega_4))$, where $s = 2^{t/2-1}$.

that, together with Thm. 2, will finish this proof when we note that by induction on the length of formulas, all terms of L_2 have witnesses of sub-quadratic length, $W = M^{2-o(1)}$. □

Next, we show that non-negativity and exponential relation have representing polynomials with sub-quadratic W. These results are novel in the following sense. First, in the proof of non-negativity we propose a slightly more efficient witness algorithm, compared to the prior art. Our system of Diophantine equations for the exponential relation, on the other hand, has substantially shorter witnesses compared to what was known previously for this relation [AM76].

Theorem 2. *An integer μ can be represented as $\mu = \omega_1^2 + \omega_2^2 + \omega_3^2 + \omega_4^2$ with integer ω_i iff $\mu \geq 0$. Moreover, if $\mu \geq 0$ then the corresponding representation $(\omega_1, \omega_2, \omega_3, \omega_4)$ can be computed efficiently by using Algorithm 1.*

Proof. First, no negative integer is a sum of four squares. Second, if $\mu \geq 0$, μ can decomposed as $\sum_{i=1}^4 \omega_i^2$ by a well-known result of Lagrange from 1770. Rabin and Shallit [RS86] proposed a probabilistic polynomial-time algorithm for computing the witnesses ω_i. The new Algorithm 1 is somewhat more efficient, due to the pairing of the Rabin-Shallit algorithm with the well-known Cornacchia algorithm from 1908 [Coh95, Section 1.5.2] that, given a prime $p \equiv 1 \pmod 4$, finds a pair (ω_3, ω_4), such that $p = \omega_3^2 + \omega_4^2$. (To compare, the original Rabin-Shallit algorithm used the full Euclidean algorithm over Gaussian integers, while Cornacchia's algorithm uses the partial Euclidean algorithm over integers). Finally, square root of -1 modulo p can be found efficiently. □

Exponential Relation is in PD. For a long time, finding a representing polynomial for the exponential relation was the last open issue in the solution of the Hilbert's 10th problem [Mat93]. Matiyasevich was the first to describe an explicit representing polynomial for the exponential relation. Alternative polynomial were later found in [Dav73,JSWW76], but none of these polynomials is really practical for our purposes due to at least cubic-length witnesses. However, Adleman and Manders showed in 1976 [AM76] that when one allows exponentially long witnesses when $x \notin S$ then the polynomial proposed in [MR75] can be modified to have sub-quadratic-length witnesses when $x \in S$.

Next, we construct a new representing polynomial that is slightly more efficient than the one in [AM76]. Our proof bases on ideas from [AM76,Mat93,Rob52]. To prove our result, we use crucially the next lemma that is an analogue of Lemma VII from [AM76]. ([AM76, Lemma VII] was stated for a different Lucas sequence, worked only when $c < 2d$, and guaranteed only that either $a < (2c)^d$ or $a \geq c^c$.)

Lemma 2. *Let (a, b, c, d) be any integers with $c > d + 2 \geq 2$. If $[(a^2 - cab - b^2 = 1) \wedge (0 \leq a < b) \wedge (a \equiv d \pmod{c-2})]$, then either $(a, b) = (c^{[\![d]\!]}, c^{[\![d+1]\!]})$ and $a \leq c^{d-1}$, or $(a, b) \neq (c^{[\![d]\!]}, c^{[\![d+1]\!]})$ and $a \geq (c-1)^{d+c-3}$.*

Proof. Let (a, b, c, d) be such integers. Since $[(a^2 - cab - b^2 = 1) \wedge (0 \leq a < b)]$, then $(a, b) = (c^{[\![x]\!]}, c^{[\![x+1]\!]})$ for some $x \in \mathbb{N}_0$. Since $e^{[\![f]\!]} \equiv f \pmod{e-2}$ for any e, f [Mat93], $[a \equiv d \pmod{c-2}]$ guarantees that $x \equiv d \pmod{c-2}$. Since $c > d+2$, then $(a, b) = (c^{[\![d+k(c-2)]\!]}, c^{[\![d+k(c-2)+1]\!]})$ for some $k \geq 0$. If $x = d$ then $a = c^{[\![d]\!]} \leq c^{d-1}$. On the other hand, if $x \neq d$ then $a \geq c^{[\![d+(c-2)]\!]} \geq (c-1)^{d+c-3}$. □

Theorem 3. *Assume $\mu_1 > 1$, $\mu_3 > 0$ and $\mu_2 > 2$. The exponential relation $[\mu_3 = \mu_1^{\mu_2}]$ belongs to* **PD**. *More precisely, let $E(\mu_1, \mu_2, \mu_3)$ be the next equation:*

$$
\begin{aligned}
&[(\exists \omega_1, \omega_2, \omega_3, \omega_4, \omega_5, \omega_6)(\exists_b \omega_7, \omega_8)] \\
&\quad [(\omega_2 = \omega_1\mu_1 - \mu_1^2 - 1) \wedge (\omega_2 - \mu_3 - 1 \geq 0) \wedge && (E1 - E2) \\
&\quad (\mu_3 - (\mu_1 - \omega_1)\omega_7 - \omega_8 = \omega_2\omega_3)) \wedge (\omega_1 - 2 \geq 0) \wedge && (E3 - E4) \\
&\quad ((\omega_1 - 2)^2 - (\mu_1 + 2)(\omega_1 - 2)\omega_5 - \omega_5^2 = 1) \wedge && (E5) \\
&\quad (\omega_1 - 2 = \mu_2 + \omega_6(\mu_1 + 2)) \wedge (\omega_7 \geq 0) \wedge (\omega_7 < \omega_8) \wedge && (E6 - E8) \\
&\quad (\omega_7^2 - \omega_1\omega_7\omega_8 - \omega_8^2 = 1) \wedge (\omega_7 = \mu_2 + \omega_4(\omega_1 - 2)] \ , && (E9 - E10)
\end{aligned}
$$

where "$\exists_b$" signifies a bounded quantifier in the following sense: if $\mu_3 = \mu_1^{\mu_2}$ then $E(\mu_1, \mu_2, \mu_3)$ is true with $W = \Theta(\mu_2^2 \log \mu_1) = o(M^2)$. On the other hand, if $\mu_3 \neq \mu_1^{\mu_2}$ then either $E(\mu_1, \mu_2, \mu_3)$ is false, or it is true but the intermediate witnesses ω_7 and ω_8 have length $\Omega(\mu_3 \log \mu_3)$, which is equal to $\Omega(2^M \cdot M)$ in the worst case.

(Note that 16 additional witnesses are needed in four inequalities. For the sake of simplicity we will not enlist all of them.)

Proof. Denote the ith conjuctive subformula of E by Ei. We will proceed by showing that the required witnesses are $\omega_1 \leftarrow (\mu_1 + 2)^{[\![\mu_2+1]\!]} + 2$, $\omega_2 \leftarrow \omega_1\mu_1 -$

$\mu_1^2 - 1$, $\omega_3 \leftarrow (\mu_3 - (\mu_1 - \omega_1)\omega_1^{[\![\mu_2]\!]} - \omega_1^{[\![\mu_2+1]\!]})/\omega_2$, $\omega_4 \leftarrow (\omega_8 - \mu_2)/(\omega_1 - 2)$, $\omega_5 \leftarrow (\mu_1+2)^{[\![\mu_2+2]\!]}$, $\omega_6 \leftarrow (\omega_1 - 2 - \mu_2)/(\mu_1+2)$, $\omega_7 \leftarrow \omega_1^{[\![\mu_2]\!]}$ and $\omega_8 \leftarrow \omega_1^{[\![\mu_2+1]\!]}$.

Really, let

$$B_a := \begin{pmatrix} a & -1 \\ 1 & 0 \end{pmatrix} \quad , \quad \text{then} \quad B_a^r = \begin{pmatrix} a^{[\![r+1]\!]} & -a^{[\![r]\!]} \\ a^{[\![r]\!]} & -a^{[\![r-1]\!]} \end{pmatrix}$$

for any a and r. For an ω_1 that we will fix later, let $\omega_2 := \omega_1\mu_1 - \mu_1^2 - 1$, i.e., assume that $E1$ holds. Then, $(\mu_1, 1)^\top$ is an eigenvector of B_{ω_1} modulo ω_2, with eigenvalue μ_1, since $B_{\omega_1} \cdot (\mu_1, 1)^\top = (\omega_1\mu_1 - 1, \mu_1)^\top \equiv (\mu_1^2, \mu_1)^\top = \mu_1 \cdot (\mu_1, 1)^\top \pmod{\omega_2}$. Therefore,

$$\begin{pmatrix} \omega_1^{[\![\mu_2+1]\!]} & -\omega_1^{[\![\mu_2]\!]} \\ \omega_1^{[\![\mu_2]\!]} & -\omega_1^{[\![\mu_2-1]\!]} \end{pmatrix} \cdot \begin{pmatrix} \mu_1 \\ 1 \end{pmatrix} = B_{\omega_1}^{\mu_2} \cdot \begin{pmatrix} \mu_1 \\ 1 \end{pmatrix} \equiv \mu_1^{\mu_2} \cdot \begin{pmatrix} \mu_1 \\ 1 \end{pmatrix} \pmod{\omega_2} \ .$$

In particular, $\mu_1\omega_1^{[\![\mu_2]\!]} - \omega_1^{[\![\mu_2-1]\!]} \equiv \mu_1^{\mu_2} \pmod{\omega_2}$. Now, as soon as $\mu_1^{\mu_2} < \omega_2$, we can write $[\mu_3 = \mu_1^{\mu_2}] \iff [E2 \wedge (\mu_1\omega_1^{[\![\mu_2]\!]} - \omega_1^{[\![\mu_2-1]\!]} \equiv \mu_1^{\mu_2} \pmod{\omega_2})]$.

One can guarantee that $\mu_1^{\mu_2} < \omega_2$ by selecting ω_1, so that $\omega_1 \geq \mu_1^{\mu_2-1} + \mu_1 + 2$. To be able later to apply Lemma 2, it also must be the case that $\omega_2 > \mu_2 + 2$. Since $\mu_1 > 1$, we can choose $\omega_1 \leftarrow (\mu_1+2)^{[\![\mu_2]\!]} + 2 \geq (\mu_1+1)^{\mu_2-1} + 2 \geq \mu_1^{\mu_2-1} + \mu_1 + 2$. Since $\mu_1 > 0$, we can invoke Lemma 2 with $(a, b, c, d) = (\omega_1 - 2, \omega_5, \mu_1 + 2, \mu_2)$. Since here it suffices to show that $\omega_1 - 2 = (\mu_1+2)^{[\![\mu_2+k\mu_1]\!]}$ and $\omega_5 = (\mu_1 + 2)^{[\![\mu_2+k\mu_1+1]\!]}$ for some $k > 0$, we are done by adding two verifications ($E5$ and $E6$) from Lemma 2. (More precisely, here we one does not have to verify that $\omega_1 - 2 < \omega_5$.)

Now, due to the choice of ω_1, $\omega_1 > (\mu_1+1)^{\mu_2-1} + 2 \geq \mu_2 + 2$. Therefore, Lemma 2 with inputs $(a, b, c, d) = (\omega_7, \omega_8, \omega_1, \mu_2)$ guarantees that after doing the verifications ($E7 - E10$), one can be assured that one of the next two cases is true. First, $(\omega_7, \omega_8) = (\omega_1^{[\![\mu_2]\!]}, \omega_1^{[\![\mu_2+1]\!]})$. Then $|\omega_7| \approx |\omega_8| \approx \mu_2 \cdot |\omega_1| \approx \mu_2^2 \cdot |\mu_1| \leq \mu_2^2 \cdot |\mu_1| < |M_3|^2 < 2|M|^2$. (Note that $M \approx \mu_2|\mu_1|$.) Second, $(\omega_7, \omega_8) \neq (\omega_1^{[\![\mu_2]\!]}, \omega_1^{[\![\mu_2+1]\!]})$, but then $|\omega_7| \geq |(\omega_1 - 1)^{\omega_1-2}| \approx \omega_1|\omega_1| \approx \mu_1^{\mu_2-1} \cdot \log_2 \mu_1^{\mu_2-1} \geq \mu_3 \cdot \log_2 \mu_3 \approx 2^M \cdot M$, which is exponential in the input size. □

The largest Z-function occurring in this lemma is $\omega_8 = \omega_1^{[\![\mu_2+1]\!]} = Z_{(\mu_1+2)^{[\![\mu_2]\!]}+2}(\mu_2+1) \leq Z_{(\mu_1+2)^{\mu_2-1}}(\mu_2+1) \leq (\mu_1+2)^{\mu_2^2-\mu_2}$. For comparison, [AM76] used an equation system from [MR75], where the largest ψ-function (for a different Lucas sequence ψ) is $\psi_{4\mu_2\mu_1(\mu_3+1)+\mu_1^2+2\mu_1}(\mu_2+1)$.

The cases $\mu_1 \in [0, 1]$, $\mu_3 = 0$ and $\mu_2 \in [0, 1, 2]$ can be handled trivially, and therefore the exponential relation belongs to **PD** for any μ_1, μ_2, μ_3. One application of this theorem is that an arbitrary Turing machine can be emulated by a slightly more efficient Diophantine Turing machine than it was known before [AM76].

4 Cryptographic Applications

Diophantine Membership Arguments. Given a secure integer commitment scheme with efficient HVSZK AoK-s for additive and multiplicative relations, one can argue in HVSZK that any polynomial relation holds between a tuple of committed integers [FO99]. That is, one can argue in HVSZK that $p(\mu) = 0$ for some fixed $p \in \mathbb{Z}[X]$, and a committed $\mu \in \mathbb{Z}^n$.

We will expand the [FO99]-methodology as follows. When $S \in \mathbf{D}$ and the arguer knows the witness, then by using an integer commitment scheme, she can argue in HVSZK that she knows an auxiliary (suitably chosen) witness ω, such that $\mathfrak{R}_S(\mu; \omega) = 0$, where $\mathfrak{R}_S$ is again the representing polynomial of S. This results in a what we call a *Diophantine argument system* $\mathrm{AK}(c_1 = C_K(\mu_1, \dots, \mu_n; \rho_1) \wedge (\mu_1, \dots, \mu_n) \in S)$.

The asymptotical communication complexity of the resulting Diophantine argument system is $\Theta(W + M)$, where the constant depends on the number of parameters and witnesses, but also on the degree of $\mathfrak{R}_S$ and on the internal structure of $\mathfrak{R}_S$. (For example, a Diophantine argument system for $\mu_1 + \mu_2 = \omega_1^4 + \omega_2^2$ requires a constant times more interaction than the one for $\mu_1 = \omega_1^2$.) Thus, Diophantine argument systems with interaction $M^{O(1)}$ exist for all $S \in \mathbf{D}$. In particular, an immediate corollary of the positive solution to the Adleman-Manders conjecture $\mathbf{NP} = \mathbf{D}$ is that every set $S \in \mathbf{NP}$ has a Diophantine HVSZK argument system with communication complexity $M^{O(1)}$. However, there are two practical considerations.

First, if (say) $W = M^{\Omega(2)}$ then the resulting argument systems are asymptotically too long to have immediate applications in cryptography. As we also so in this paper, finding representing polynomials $\mathfrak{R}_S$ with small W is a nontrivial task, and it often needs breakthroughs in number theory.

Note also that quadratic length seems to be a reasonable metering point, since for many interesting predicates one can build trivial quadratic-length zero-knowledge arguments (here and in the following, assume for the sake of simplicity that the input length M is larger than the security parameter k). In such AoK-s, one separately commits to every bit of the input, and then shows that the committed bits satisfy some Boolean formula. An immediate corollary of Theorem 1 is that one can build *sub-quadratic-length* HVSZK AoK-s for all languages from L_2. Therefore, our AoK-s are an improvement upon such argument systems.

Second, if $S \in \mathbf{D} \setminus \mathbf{PD}$, the arguer cannot efficiently find the witness ω for every relevant input μ. In such a case, the witness ω can be seen as a trapdoor information. However, this case is still relevant in certain cryptographic applications. For example, the relation ["μ is composite"] $\equiv [(\exists y_1, y_2 \leq \mu)[\mu = y_1 y_2 \wedge y_1 > 1 \wedge y_2 > 1]]$ does not have a witness algorithm, given that factoring is hard. (The resulting argument system that a committed number is composite can be compared to a more complex protocol by Poupard and Stern [PS00].) In particular this means that $\mathbf{D} \neq \mathbf{PD}$, unless factoring is easy.

Note that to apply the previously described methodology, one needs to both encrypt and commit all messages. Additionally, one needs to argue that that encrypted and committed messages are equal. This can be done straightforwardly by using standard cryptographic tools. We finish the paper with concrete applications and protocols. There are definitely more applications than we mention in the following. In particular, our methodology is not limited to the outsourcing model.

Example: Efficient Range Proofs. A cryptographically important argument system for $\mathbb{N}_0$ (a partial list of potential applications to this argument system can be found in [Bou00], it includes electronic cash systems, verifiable encryption, group signatures, publicly verifiable secret sharing schemes and other zero-knowledge protocols; more applications can be found in [LAN02] and in the current paper) can be based on Theorem 2. Briefly, during this argument system, the arguer first represents μ as $\mu = \omega_1^2 + \omega_2^2 + \omega_3^2 + \omega_4^2$ (here, $\omega = (\omega_1, \dots, \omega_4)$ is the witness). After that, she argues in HVSZK that she knows such a representation. Our argument system bases on the new integer tuple commitment scheme. The full argument system is described in Appendix B. A non-interactive version of such argument system is ≈ 1700 bytes long for realistic security parameters. This is slightly shorter than Boudot's argument system [Bou00] for the same problem. Additionally, our argument system is perfectly complete, while Boudot's argument system is not. A nice demonstration of the usefulness of the new integer tuple commitment scheme (presented in Appendix A) is the fact that this argument system has only ≈ 1.9 times larger non-interactive argument than the original multiplication proof of Damgård and Fujisaki; this is achieved by doing four squarings in parallel.

Outsourcing Model. A general setting in many cryptographic protocols (like voting and auctions [LAN02]) involves a set of participants, an authority and possibly an impartial third party. The participants make social or financial choices $\{v_i\}$, encode them as $\{\mathsf{enc}(v_i)\}$ by using some encoding function enc, and then encrypt the resulting encodings by using a homomorphic public-key cryptosystem and third party's public key, and send the results, together with an HVSZK argument of correctness, to the authority. (Of course, we assume that all the steps are authenticated.) The authority multiplies the ciphertexts and sends the product $\prod_i E_K(\mathsf{enc}_i(v_i)) = E_K(\sum_i \mathsf{enc}(v_i))$ to the third party. The third party decrypts the result, obtains the sum $\sum_i \mathsf{enc}(v_i)$ and applies a decoding function dec to obtain the vector $e = (\dots, e_j, \dots)$, where e_j can for example be the number of voters whose choice was j. The third party applies some function final to e, and sends $\mathsf{final}(e)$ to the authority together with an zero-knowledge argument of correctness that $\mathsf{final}(e)$ was correctly computed. The authority then broadcasts $\mathsf{final}(e)$ and the argument of correctness to all participants.

As an example, final could be an identity function. Then this model will implement a common voting process with an accountable third party. If $\mathsf{final}(e) = j_0$ where $e_{j_0} = \max e_j$, one could implement voting with minimal information disclosure. Namely, the authority would only get to know the name of the winner. To the best of our knowledge, there are no such efficient prior art voting schemes. One can also implement the $(b+1)$st-price auctions by choosing $\mathsf{final}(e) = j_0$, where j_0 is the $(b+1)$st largest social choice [LAN02]. (This includes Vickrey auctions, for example.)

In general, the "outsourcing" model enables one to construct secure and extremely efficient voting (or auction) schemes with the only drawback that the third party (but only she) will get to know the value of e. In particular, this enables one to avoid threshold trust. See [LAN02] for a discussion why at least in the auction scenario, the information leakage to the authority does not matter but the property of not using threshold trust does. In the most common in the real-world voting scenario, the vector e is meant to be leaked. Moreover, even in the nation-wide elections, one does not really want to have threshold trust between computers. Instead, it seems to be desirable—as show discussions with the members of electorate committees—that the encoded and encrypted vector e can be decrypted by using a single hardware-protected private key that can be used only by the presence of several trusted entities and independent experts, and will be destroyed as soon as some allocated period at the end of elections (and all election-related legal discussions) have ended.

Now, final can be any function for which the predicate $[y = \mathsf{final}(x)]$ belongs to **PD**. As we have shown, extremely efficient arguments are available when $\mathsf{final} \in L_2$. It is not known how to implement as efficiently so many different schemes for such a broad variety of functions final in the model that involves threshold trust but no third party like in [CGS97,DJ01]. In particular, no really efficient $(b+1)$st-price auctions are known in the threshold trust scenario.

Efficient Range Arguments in Exponents. The costliest part of the otherwise efficient Damgård-Jurik voting protocol from [DJ01] involves an argument for $\mathrm{AK}(y = E_K(\mathsf{enc}(\mu)) \wedge \mu \in [0, h])$ that is necessary to show that the votes were encoded properly. We call this argument a *range argument in exponents* (RAIE). An RAIE is also necessary in the auction protocol of [LAN02], both to show that the bids were encoded correctly, and that the authority returns the correct value of $\mathsf{final}(e)$. The proposed AoK-s from [DJ01,LAN02] have interaction $\Theta(\max(k, m \cdot \log a) \cdot \log m)\Theta(m \cdot \log a \cdot \log m)$, where a is an a priori fixed upper bound to the number of participants, and m is the number of possible social choices. (This follows from [LAN02, Section 8], when we assume that the security parameter is approximately equal to $m \log a$.)

The most efficient known RAIE [LAN02] has $\mathsf{enc}(\mu) := (\mathsf{nextprime}(a))^{\mu}$ (where $\mathsf{nextprime}(a)$ is the smallest prime $\geq a$) and results in a HVSZK AoK with interaction length $\Theta(m \cdot \log a)$. We propose two different RAIE-s that do not require computing the $\mathsf{nextprime}$ function. The first approach sets $\mathsf{enc}(\mu) := Z_a(\mu + 1)$, where $Z_a(\mu)$ is the μth element in the familiar Lucas sequence, and results in a HVSZK AoK with interaction length $\Theta(m \cdot \log a)$. Application of Z instead of the exponentiation enables us to improve over the communication efficiency of the Damgård-Jurik multi-candidate voting scheme [DJ01] and over the Lipmaa-Asokan-Niemi $(b+1)$st-price auction scheme [LAN02] by a factor of $\Theta(\log m)$. Finally, we propose a *Diophantine* RAIE with $\mathsf{enc}(\mu) := a^{\mu}$ and interaction $\Theta(W + M) = \Theta(M^{2-\varepsilon}) = \Theta((m \cdot \log a)^{2-\varepsilon})$.

First approach: Lucas sequences. The function $a^{[\![n]\!]} = Z_a(n)$ is a suitable replacement for exponentiation in the sense, intended in [DJ01,LAN02], since $(a-1)^n \leq Z_a(n) \leq a^n$ whenever $a > 2$ (This makes the constants e_j in the sum $\sum_{i=1}^{a} Z_{a+1}(v_i) = \sum_j e_j Z_{a+1}(j)$ unambiguous whenever $v_i \in [1, h]$, and thus makes it possible to uniquely recover the vector e from $\sum \mathsf{enc}(e_i)$. However,

we must make the plausible assumption that $a > 2$, for $a = 2$ one has to use another approach.), and that $Z_a(n)$ can be computed in time $O(\log n)$. Most importantly, one can very efficiently argue that the committed number μ belongs to the set $\{a^{[\![n]\!]} : n \geq 0 \wedge n = k^{O(1)}\}$ by using the representing polynomial $\Re_S(\mu; \omega) = \omega^2 - a\mu\omega - \mu^2 - 1$. This must be accompanied by an AoK that $\mu \in [l, h]$. The length of a non-interactive version of this argument is ≈ 1200 bytes for realistic security parameters. A minor drawback of this solution is that computing $Z_a(n)$ requires about twice more resources than computing of a^n without the function $\mathsf{nextprime}$. (Also, in some solutions one cannot readily substitute exponentiation with the function Z.) Note also that $Z_a(n)$ is not the unique Lucas sequence that satisfies all these conditions.
Second approach. Here, one would have $\mathsf{enc}(n) = a^n$, as in [DJ01,LAN02]. The argument system from Thm. 2 is usually not more communication-efficient than the protocols from [DJ01,LAN02], however, it is constant-round, which may have advantages in some concrete applications. (Precise analysis omitted due to the space constraints. Note that here we have the relation $[\mu_2 = a^{\mu_1}]$ for a *constant* a, that allows us to improve on Thm. 3.)

Acknowledgements and Further Work

This work was partially supported by the Finnish Defense Forces Research Institute of Technology. We would like to thank Yuri Matiyasevich, Jeffrey Shallit and anonymous referees for useful comments. This paper obsoletes an earlier technical report [Lip01].

Efficient Diophantine membership arguments can be given for many interesting sets $S \subset \mathbb{Z}$. We did certainly not mention all cryptographically relevant sets S that have such arguments, and the class L_2 can be certainly broadened. We hope that this paper stimulates the research both in finding more efficient representing polynomials for concrete sets S but also in giving a (positive or negative) answer to the conjecture $\mathbf{NP} = \mathbf{D}$.

References

[AM76] Leonard M. Adleman and Kenneth L. Manders. Diophantine Complexity. In *17th Annual Symposium on Foundations of Computer Science*, pages 81–88, Houston, Texas, USA, 25–27 October 1976. IEEE Computer Society Press.

[Bou00] Fabrice Boudot. Efficient Proofs that a Committed Number Lies in an Interval. In Bart Preneel, editor, *Advances in Cryptology — EUROCRYPT 2000*, volume 1807 of *Lecture Notes in Computer Science*, pages 431–444, Bruges, Belgium, May 14–18 2000. Springer-Verlag. ISBN 3-540-67517-5.

[Bra97] Stefan Brands. Rapid Demonstration of Linear Relations Connected by Boolean Operators. In Fumy [Fum97], pages 318–333.

[CGS97] Ronald Cramer, Rosario Gennaro, and Berry Schoenmakers. A Secure and Optimally Efficient Multi-Authority Election Scheme. In Fumy [Fum97], pages 103–118.

[Coh95] Henri Cohen. *A Course in Computational Algebraic Number Theory*, volume 138 of *Graduate Texts in Mathematics*. Springer-Verlag, 1995.

[Dav73] Martin Davis. Hilbert's Tenth Problem is Unsolvable. *American Mathematical Monthly*, 80(3):233–269, March 1973.

[DF02] Ivan Damgård and Eiichiro Fujisaki. An Integer Commitment Scheme Based on Groups with Hidden Order. In Yuliang Zheng, editor, *Advances on Cryptology — ASIACRYPT 2002*, volume 2501 of *Lecture Notes in Computer Science*, pages 125–142, Queenstown, New Zealand, December 1–5 2002. Springer-Verlag.

[DJ01] Ivan Damgård and Mads Jurik. A Generalisation, a Simplification and Some Applications of Paillier's Probabilistic Public-Key System. In Kwangjo Kim, editor, *Public Key Cryptography '2001*, volume 1992 of *Lecture Notes in Computer Science*, pages 119–136, Cheju Island, Korea, 13–15 February 2001. Springer-Verlag.

[FO99] Eiichiro Fujisaki and Tatsuaki Okamoto. Statistical Zero-Knowledge Protocols to Prove Modular Polynomial Relations. *IEICE Transaction of Fundamentals of Electronic Communications and Computer Science*, E82-A(1):81–92, January 1999.

[FS86] Amos Fiat and Adi Shamir. How to Prove Yourself: Practical Solutions to Identification and Signature Problems. In Andrew M. Odlyzko, editor, *Advances in Cryptology—CRYPTO '86*, volume 263 of *Lecture Notes in Computer Science*, pages 186–194, Santa Barbara, California, USA, 11–15 August 1986. Springer-Verlag, 1987.

[Fum97] Walter Fumy, editor. *Advances in Cryptology — EUROCRYPT '97*, volume 1233 of *Lecture Notes in Computer Science*, Konstanz, Germany, 11–15 May 1997. Springer-Verlag.

[JM84] J. P. Jones and Yuri Matiyasevich. Register Machine Proof of the Theorem on Exponential Diophantine Representation of Enumerable Sets. *Journal of Symbolic Logic*, 49:818–829, 1984.

[JQ96] Marc Joye and Jean-Jacques Quisquater. Efficient Computation of Full Lucas Sequences. *Electronics Letters*, 32(6):537–538, March 1996.

[JSWW76] James P. Jones, Daihachiro Sato, Hideo Wada, and Douglas Wiens. Diophantine Representation of the Set of Prime Numbers. *American Mathematical Monthly*, 83(6):449–464, June–July 1976.

[LAN02] Helger Lipmaa, N. Asokan, and Valtteri Niemi. Secure Vickrey Auctions without Threshold Trust. In Matt Blaze, editor, *Financial Cryptography — Sixth International Conference*, volume 2357 of *Lecture Notes in Computer Science*, pages 87–101, Southhampton Beach, Bermuda, March 11–14 2002. Springer-Verlag.

[Lip01] Helger Lipmaa. Statistical Zero-Knowledge Proofs from Diophantine Equations. Cryptology ePrint Archive, Report 2001/086, November 20 2001. http://eprint.iacr.org/.

[Mat70] Yuri Matiyasevich. Enumerable Sets are Diophantine. *Soviet Math., Doklady*, 11:354–358, 1970. English translation.

[Mat93] Yuri Matiyasevich. *Hilbert's Tenth Problem*. Foundations of Computing. MIT Press, October 1993. ISBN 0-262-13295-8.

[MR75] Yuri Matiyasevich and Julia Robinson. Reduction of an Arbitrary Diophantine Equation to One in 13 Unknowns. *Acta Arithmetica*, 27:521–553, 1975.

[Pol03] Chris Pollett. On the Bounded Version of Hilbert's Tenth Problem. *Archive for Mathematical Logic*, 42(5):469–488, 2003.

[PS00] Guillaume Poupard and Jacques Stern. Short Proofs of Knowledge for Factoring. In Hideki Imai and Yuliang Zheng, editors, *Public Key Cryptography '2000*, volume 1751 of *Lecture Notes in Computer Science*, pages 147–166, Melbourne, Victoria, Australia, 18–20 January 2000. Springer-Verlag.

[Rob52] Julia Robinson. Existential Definability in Arithmetic. *Transactions of the American Mathematical Society*, 72(3):437–449, May 1952.

[RS86] Michael O. Rabin and Jeffrey O. Shallit. Randomized Algorithms in Number Theory. *Communications in Pure and Applied Mathematics*, 39:239–256, 1986.

[Wil82] Hugh C. Williams. A $p+1$ Method of Factoring. *Mathematics of Computation*, 39:225–234, 1982.

A Extensions to Damgård-Fujisaki Integer Commitment Scheme

Let $Gen \in \mathcal{EA}$ be a group generation algorithm that on the input 1^k outputs the description $\mathsf{descr}(\mathcal{G})$ of a finite Abelian group $\mathcal{G}$. Apart from the usual assumptions (given $D \in \Sigma^*$, it is easy to verify that $D \in Gen(1^k)$, easy to verify whether some μ belongs to $\mathcal{G}$ for which $D = \mathsf{descr}(\mathcal{G})$, and easy to perform group operations in $\mathcal{G}$ for which $D = \mathsf{descr}(\mathcal{G})$), we require a few additional assumptions.

First, one assumes that while the arguer knows a reasonably close upper bound $2^B > \mathrm{ord}(\mathcal{G})$ to the order of $\mathcal{G}$, $B = B_{\mathcal{G}}$, he does *not* know the order itself. Let $\ell(k)$ be polynomial in k. Another large number $F = F(k)$ is chosen, such that it is still feasible to factor numbers that are smaller than $F(k)$. Say, $F(k) = O(k^{\log k})$. (In our calculations we will take $F(k) = 2^{80}$ when $k = 1024$.) Based on the fundamental theorem of finite Abelian groups, one can write $\mathcal{G}$ as $\mathcal{G} = \mathcal{U} \times \mathcal{H}$, where the order of $\mathcal{U}$ has only prime factors at most $F(k)$ (we call such numbers $F(k)$-smooth) and the order of $\mathcal{H}$ has prime factors larger than $F(k)$ (we call such numbers $F(k)$-rough).

Let $\ell(\mathcal{G}) := |\mathcal{U}|$. Then $\ell(\mathcal{G})$ is $F(k)$-smooth. It is assumed that (1) $\ell(\mathcal{G}) \leq \ell(k)$ and that $\mathsf{descr}(\mathcal{G})$ includes $\ell(\mathcal{G})$; (2) for any string μ it can be decided on polynomial time, based on $(x, \mathsf{descr}(\mathcal{G}))$, whether x represents an element in $\mathcal{G}$. Finally, it is assumed that the next *strong divisible root assumption* holds: given a random $\mathcal{G} \leftarrow Gen(1^k)$ and $y \leftarrow \mathcal{G}$, it is hard to produce such (x, d, e) that $y^e = x^{de}$ and $e \leq \ell(\mathcal{G})$. The probability is taken over the coin tosses of Gen and of the adversary. Note that this assumption is an equivalent but simpler version of the root assumption from [DF02].

It was shown in [DF02] that $\mathcal{G}$ can be chosen as $\mathbb{Z}_n$ for RSA modulus $n = pq$, such that $\gcd(p-1, q-1) = 2$, $p-1$ and $q-1$ do not have too many small factors, and the strong RSA assumption holds. However, when the RSA group $\mathbb{Z}_n^*$ is used, one must additionally assume that the arguer does not know the value $\varphi(n)$. This may achieved, for example, when the verifier creates n and keeps its factorisation secret.

Commitment scheme. During the setup phase of Damgård-Fujisaki integer commitment scheme, A and V agree on the group $\mathcal{G}$ and on a large integer $F(k)$.

Verifier V chooses a random element $h \in \mathcal{G}$ (which by the group assumptions has a $F(k)$-rough order [DF02] with an overwhelming probability. To make the order certainly $F(k)$-rough, one might raise a random element to the power $\ell(\mathcal{G})$.) and a random secret key $s \in \mathbb{Z}_{2^{B+k}}$. V sets $g \leftarrow h^s$. Verifier V sends the public key $K = (g; h)$ to A and then proves in SZK that $g \in \langle h \rangle$. Let $\mathcal{C}_{Com}$ denote the commitment space of the used integer commitment scheme (in this concrete case, $\mathcal{C}_{Com} = \mathcal{G}$). When committing to $m \in \mathbb{Z}$, A chooses a random $r \leftarrow \mathbb{Z}_{2^{B+k}}$ and sends $C_K(m; r) := g^m h^r$ to V. To open a commitment c, A sends to V a triple (m, r, b), such that $c = C_K(m; r) \cdot b$ and $b^{\ell(\mathcal{G})} = 1$. (For an explanation of the role of b in the opening phase, see [DF02].) Alternatively, A can send only (m, r) to V who then verifies that $c^{\ell(\mathcal{G})} = C_K(m; r)^{\ell(\mathcal{G})}$. Clearly, this alternative is equivalent to the Damgård-Fujisaki commitment scheme in security. (The proof of this is trivial: if $c^{\ell(\mathcal{G})} = C_K(m; r)^{\ell(\mathcal{G})}$ then V can compute b as $b \leftarrow c \cdot C_K(m; r)^{-1}$. Clearly, $b^{\ell(\mathcal{G})} = 1$ and $c = C_K(m; r) \cdot b$. On the other hand, given b with $b^{\ell(\mathcal{G})} = 1$ and $c = C_K(m; r) \cdot b$, clearly $c^{\ell(\mathcal{G})} = C_K(m; r))^{\ell(\mathcal{G})}$.

Integer Tuple Commitment Scheme. We now sketch an extension to the Damgård-Fujisaki commitment scheme that allows to simultaneously commit to a tuple of integers. As in the Damgård-Fujisaki commitment scheme, the arguer and verifier initially agree on a group $\mathcal{G}$, and then verifier creates a random element $h \in \mathcal{G}$. Additionally, the verifier will choose m random elements $s_i \leftarrow [0, 2^{B+k}]$, where B is a security parameter [DF02], set $g_i \leftarrow h^{s_i}$ and send the values g_i to the verifier. Apart from that, arguer A and verifier V follow the same initialisation rules as in the Damgård-Fujisaki scheme. A tuple $(\mu_1, \dots, \mu_n) \in \mathbb{Z}^n$ is committed by drawing a random integer $\rho \leftarrow [0, 2^{B+k}]$ and then setting the commitment to $C_K(\mu_1, \dots, \mu_n; \rho) := (\prod_{i=1}^n g_i^{\mu_i}) \cdot h^\rho$. During the opening phase, A sends the tuple $(\mu_1, \dots, \mu_n; \rho)$ to V, and the verifier checks that $c^{\ell(\mathcal{G})} = C_K(\mu_1, \dots, \mu_n; \rho)^{\ell(\mathcal{G})}$, where $\ell(\mathcal{G})$ is another security parameter [DF02]. (Equivalently, A can send the tuple $(\mu_1, \dots, \mu_n; \rho; b)$, and the verifier checks that $c = C_K(\mu_1, \dots, \mu_n; \rho) \cdot b$ and that $b^{\ell(\mathcal{G})} = 1$.)

It is straightforward to show that the security of the Damgård-Fujisaki integer commitment scheme and the security of the the sketched extension (that we call the RDF integer commitment scheme) are equivalent, given that the arguer does not know the mutual discrete logarithms of elements g_i. As a simple corollary, we can use the RDF integer commitment scheme C to build HVSZK AoK-s of type $\mathrm{AK}(\dots \wedge y = C_K(\mu_1, \dots, \mu_n; \rho) \wedge \dots)$.

The RDF integer commitment scheme can be used to speed up the efficiency of many argument systems, by enabling one to prove several multiplicative or additive relations at once [Bra97]. (In contrast, without using the RDF scheme, a separate protocol must be used for every polynomial relation.) That is, such combined arguments enable one to argue in parallel that $\bigwedge_i y_i = p(\mu_{i1}, \dots, \mu_{in})$ for polynomially many instances of any polynomial p.

As an example, one can construct an argument for the multiplicative relation $\mathrm{AK}(y = C_K(\mu_1, \mu_2, \mu_1\mu_2; \rho))$, $K = (g_1, g_2, g_3; h)$, that is approximately 20% shorter than the argument from [DF02] when using the same security parameters.

Protocol 1 Computationally sound HVSZK argument system for the set of nonnegative integers.

1. Arguer A represents μ as $\omega_1^2+\omega_2^2+\omega_3^2+\omega_4^2$, using the algorithm from Theorem 2. For $i \in [1,4]$, A chooses random $r_{1i} \leftarrow \mathbb{Z}_{2^{B+k}}$ such that $\sum_i r_{1i} = \rho$; A chooses random $m_{1i} \leftarrow \mathbb{Z}_{2^k F(k) M^{1/2}}$, $r_{2i} \leftarrow \mathbb{Z}_{2^{B+2k} F(k)}$ and lets $c_{1i} \leftarrow C_{K_i}(\omega_i; r_{1i})$. She also chooses a random $r_3 \leftarrow \mathbb{Z}_{2^{B+2k} F(k) M^{1/2}}$ and lets $c_2 \leftarrow C_{K'}(m_{11}, \ldots, m_{14}; \sum_i r_{2i})$, $c_3 \leftarrow C_{(c_{11},\ldots,c_{14};h)}(m_{11},\ldots,m_{14}; r_3)$. Arguer sends $(c_{11}, c_{12}, c_{13}, c_{14}, c_2, c_3)$ to V.
2. V generates a random $e \leftarrow \mathbb{Z}_{F(k)}$ and sends it to A.
3. A computes $m_{2i} = m_{1i} + e\omega_i$, $r_{4i} \leftarrow r_{2i} + e\sum r_{1i}$, $i \in [1,4]$, and $r_5 \leftarrow r_3 + e\sum_i (1-\omega_i) r_{1i}$. A sends $(m_{21}, m_{22}, m_{23}, m_{24}, r_{41}, r_{42}, r_{43}, r_{44}, r_5)$ to V.
4. V checks that $\prod_i (C_K(m_{2i}; r_{4i}) \cdot c_{1i}^{-e}) = c_2$ and $\left(\prod_{i=1}^4 c_{1i}^{m_{2i}}\right) \cdot h^{r_5} c^{-e} = c_3$.

The argument is based on the idea that $y = C_K(\mu_1, \mu_2, \mu_3; \rho)$ with $\mu_3 = \mu_1 \mu_2$ iff A knows such a c_1 that $c_1 = C_{K_1}(\mu_1; \rho_2)$ and $y = C_{K_2}(\mu_1, \mu_2; \rho_3)$, where $K_2 = (g_1, g_2 c_1; h)$. (This holds except with a negligible probability.)

The RDF integer tuple commitment scheme exhibits the next *public-key homomorphicity property*, the use of which makes many AoK-s more efficient: if $K = (g_1, \ldots, g_n; h)$ and $K' = (\prod_i g_i^{a_{1i}} \cdot h^{r_1}, \ldots, \prod_i g_i^{a_{ni}} \cdot h^{r_n}; h)$ then

$$C_{K'}(\beta_1, \ldots, \beta_n; r) = C_K(\sum_i \beta_i a_{i1}, \ldots, \sum_i \beta_i a_{in}; \sum_i \beta_i r_i + r) \ .$$

B Argument System for Non-negativity

Theorem 4. *Let C be the RDF integer tuple commitment scheme, let k be the security parameter and let $\log_2 M = k^{O(1)}$. Let $K = (g; h)$ be the public key. Protocol 1 is a perfectly complete AoK for $\mathrm{AK}(c = C_K(\sum_{i=1}^4 \omega_i^2; \rho))$, or equivalently, for $\mathrm{AK}(c = C_K(\mu) \wedge \mu \geq 0)$. If $\mu \leq M$ then Protocol 1 is HVSZK.*

Proof. Proof idea: show that $y = C_K(\sum \nu_i) \wedge \bigwedge \left(c_i = C_K(\omega_i) \wedge \nu_i = \omega_i^2\right)$, where all four AoK-s $c_i = C_K(\omega_i) \wedge \nu_i = \omega_i^2$ are done in parallel.

COMPLETENESS. $c^{-e} \cdot \prod_{i=1}^4 C_K(m_{2i}; r_{4i}) = \prod_{i=1}^4 (C_K(m_{1i} + e\omega_i; r_{2i} + er_{1i}) \cdot C_K(-e\omega_i; -er_{1i})) = \prod_{i=1}^4 C_K(m_{1i}; r_{2i}) = c_2$ and $\prod_i c_{1i}^{m_{2i}} \cdot h^{r_5} c^{-e} = \prod_i c_{1i}^{m_{1i}} \cdot \prod_i (C_K(\omega_i; r_{1i}))^{e\omega_i} \cdot h^{r_3 + e\sum_i (1-\omega_i) r_{1i}} \cdot C_K(-e\sum_i \omega_i^2; -e\rho) = \prod_i c_{1i}^{m_{1i}} \cdot h^{r_3} = c_3$.

HVSZK. The simulator acts as follows. For $i \in [1,4]$, generate $\tilde{c}_{1i} \leftarrow \mathcal{C}_{Com}$, $\tilde{m}_{2i} \leftarrow \mathbb{Z}_{2^F(k) M}$. For $i \in [1,4]$, generate $\tilde{r}_{4i} \leftarrow \mathbb{Z}_{2^{B+2k} F(k)}$. Generate $\tilde{e} \leftarrow \mathbb{Z}_{F(k)}$, $\tilde{r}_5 \leftarrow \mathbb{Z}_{2^{B+2k} F(k) M}$. Let $\tilde{c}_2 \leftarrow \prod_{i=1}^4 C_K(\tilde{m}_{2i}; \tilde{r}_{4i}) \tilde{c}_{1i}^{-\tilde{e}}$. Let $\tilde{c_3} \leftarrow \prod_i \tilde{c}_{1i}^{\tilde{m}_{2i}} \cdot h^{\tilde{r}_5} c^{-\tilde{e}}$. The resulting view $((\tilde{c}_{1i})_i, \tilde{c}_2, \tilde{c}_3; \tilde{e}; (\tilde{m}_{2i})_i, (\tilde{r}_{4i})_i, \tilde{r}_5)$ is accepting and has a distribution, statistically close to the distribution of views in a real execution.

To prove that this protocol is specially sound, we must show that from two accepting views, $((c_1)_i, c_2, c_3; e; (m_{2i})_i, (r_{4i})_i, r_5)$ and $((c_{1i})_i, c_2, c_3; e'; (m'_{2i})_i, (r'_{4i})_i, r'_5)$ with $e \neq e'$, one can efficiently find a tuple $((\omega_i)_i, \rho)$, such that $c = C_K(\sum \omega_i^2; \rho))$. This can be proven as follows. Given such views, $\prod_{i=1}^4 C_K(m_{2i} - m'_{2i}; r_{4i} - r'_{4i}) = \prod_{i=1}^4 c_{1i}^{e-e'}$ and $\prod_i c_{1i}^{(m_{2i}-m'_{2i})} \cdot h^{r_5 - r'_5} = c^{e-e'}$. Assuming $K' = (c_{11}, \ldots, c_{14}; h)$, this is equivalent to $C_{K'}(m_{21} - m'_{21}, \ldots, m_{24} - m'_{24}; r_5 - r'_5) = c^{e-e'}$. By the generalisation of Lemma 1 from [DF02] and by $|e - e'| \in \mathbb{Z}_{F(k)}$, there exists a verifier V^* who together with the arguer A can break the strong divisible root problem with a high probability. $\square$

Non-interactive version of this argument system is $((c_{1i})_i; e \bmod k; (m_{2i}, r_{4i})_{i=1}^4, r_5)$, where the verifier checks that $e \equiv H(c_{11}, \ldots, c_{14}, (C_K(m_{2i}; r_{2i}) c_{1i}^{-e})_{i=1}^4, c^{-e} \prod_i c_{1i}^{m_{2i}} \cdot h^{r_5}) \pmod{2^k}$. The length of non-interactive argument system is $4|\mathcal{C}_{Com}| + k + 4(B + 3k + 2\log_2 F(k) + \frac{1}{2}\log_2 M) + B + 2k + \log_2 F(k) + \frac{1}{2}\log_2 M = 4096 + 80 + 4 \cdot (1024 + 240 + 160) + 1024 + 160 + 80 + \frac{5}{2}\log_2 M = 11136 + \frac{5}{2}\log_2 M$ bits or $1392 + \frac{5}{16}\log_2 M$ bytes.

One can parallelise this argument system even more. Namely, to prove that $y = C_K(\mu; \rho)$, it suffices to prove that $c_i = C_K(\omega_i; r_{1i})$ and $y = \left(\prod (c_i)^{\omega_i}\right)(g_i)\, h^{r_{10}}$, where $r_{10} \leftarrow \rho - r_{11}^2 - \cdots - r_{14}^2$.