Cryptographic Assumptions: A Position Paper

Shafi Goldwasser[1,2(✉)] and Yael Tauman Kalai[3]

[1] MIT, Cambridge, USA
shafi@theory.csail.mit.edu
[2] Weizmann Institute, Rehovot, Israel
[3] Microsoft Research, Cambridge, USA
yael@microsoft.com

Abstract. The mission of theoretical cryptography is to define and construct *provably secure* cryptographic protocols and schemes. Without proofs of security, cryptographic constructs offer no guarantees whatsoever and no basis for evaluation and comparison. As most security proofs necessarily come in the form of a reduction between the security claim and an intractability assumption, such proofs are ultimately only as good as the assumptions they are based on. Thus, the complexity implications of every assumption we utilize should be of significant substance, and serve as the yard stick for the value of our proposals.

Lately, the field of cryptography has seen a sharp increase in the number of new assumptions that are often complex to define and difficult to interpret. At times, these assumptions are hard to untangle from the constructions which utilize them.

We believe that the lack of standards of what is accepted as a reasonable cryptographic assumption can be harmful to the credibility of our field. Therefore, there is a great need for *measures* according to which we classify and compare assumptions, as to which are *safe* and which are not. In this paper, we propose such a classification and review recently suggested assumptions in this light. This follows the footsteps of Naor (Crypto 2003).

Our governing principle is relying on hardness assumptions that are independent of the cryptographic constructions.

1 Introduction

Conjectures and assumptions are instrumental for the advancement of science. This is true in physics, mathematics, computer science, and almost any other discipline. In mathematics, for example, the Riemann hypothesis (and its extensions) have far reaching applications to the distribution of prime numbers. In computer science, the assumption that $\mathsf{P} \neq \mathsf{NP}$ lies in the foundations of complexity theory. The more recent Unique Games Conjecture [40] has been instrumental to our ability to obtain tighter bounds on the hardness of approximation of several problems. Often, such assumptions contribute tremendously to our understanding of certain topics and are the force moving research forward.

Assumptions are paramount to cryptography. A typical result constructs schemes for which breaking the scheme is an NP computation. As we do not

E. Kushilevitz and T. Malkin (Eds.): TCC 2016-A, Part I, LNCS 9562, pp. 505–522, 2016.
DOI: 10.1007/978-3-662-49096-9_21

know that $\mathsf{P} \neq \mathsf{NP}$, an assumption to that effect (and often much more) must be made. Thus, essentially any cryptographic security proof is a reduction from the existence of an adversary that violates the security definition to dispelling an underlying conjecture about the intractability of some computation. Such reductions present a "win-win" situation which gives provable cryptography its beauty and its power: either we have designed a scheme which resists all polynomial time adversaries or an adversary exists which contradicts an existing mathematical conjecture. Put most eloquently, "Science wins either way[1]".

Naturally, this is the case *only if* we rely on mathematical conjectures whose statement is scientifically interesting independently of the cryptographic application itself. Most definitely, the quality of the assumption determines the value of the proof.

Traditionally, there were a few well-studied computational assumptions under which cryptographic schemes were proven secure. These assumptions can be partitioned into two groups: *generic* and *concrete*. Generic assumptions include the existence of one-way functions, the existence of one-way permutations, the existence of a trapdoor functions, and so on. We view generic assumptions as postulating the existence of a cryptographic primitive. Concrete assumptions include the universal one-way function assumption [31],[2] the assumption that Goldreich's expander-based function is one-way [32], the Factoring and RSA assumptions [47,49], the Discrete Log assumption over various groups [24], the Quadratic Residuosity assumption [37], the DDH assumption [24], the parity with Noise (LPN) assumption [2,10], the Learning with Error (LWE) assumption [48], and a few others.

A construction which depends on a generic assumption is generally viewed as superior to that of a construction from a concrete assumption, since the former can be viewed as an unconditional result showing how abstract cryptographic primitives are reducible to one another, setting aside the question of whether a concrete implementation of the generic assumption exists. And yet, a generic assumption which is not accompanied by at least one proposed instantiation by a concrete assumption is often regarded as useless. Thus, most of the discussion in this paper is restricted to concrete assumptions, with the exception of Sect. 2.5, which discusses generic assumptions.

Recently, the field of cryptography has been overrun by numerous assumptions of radically different nature than the ones preceding. These assumptions are often nearly impossible to untangle from the constructions which utilize them. The differences are striking. Severe restrictions are now assumed on the class of algorithms at the disposal of any adversary, from assuming that the adversary is only allowed a restricted class of operations (such as the Random Oracle Model restriction, or generic group restrictions), to assuming that any adversary who breaks the cryptosystem must do so in a particular way (this includes various

[1] Silvio Micali, private communication.

[2] A universal one-way function is a candidate one-way function f such that if one-way functions exist then f itself is one-way [31]. The universal one-way function assumption asserts that this universal f is indeed one-way.

knowledge assumptions). The assumptions often make mention of the cryptographic application itself and thus are not of independent interest. Often the assumptions come in the form of an exponential number of assumptions, one assumption for every input, or one assumption for every size parameter. Overall, whereas the constructions underlied by the new assumptions are ingenious, their existence distinctly lacks a "win-win" consequence.

Obviously, in order to make progress and move a field forward, we should occasionally embrace papers whose constructions rely on newly formed assumptions and conjectures. This approach marks the birth of modern cryptography itself, in the landmark papers of [24,49]. However, any conjecture and any new assumption must be an open invitation to refute or simplify, which necessitates a clear understanding of what is being assumed in the first place. The latter has been distinctly lacking in recent years.

Our Thesis. We believe that the lack of standards in what is accepted as a reasonable cryptographic assumption is harmful to our field. Whereas in the past, a break to a provably secure scheme would lead to a mathematical breakthrough, there is a danger that in the future the proclaimed guarantee of provable security will lose its meaning. We may reach an absurdum, where the underlying assumption is that the scheme itself is secure, which will eventually endanger the mere existence of our field.

We are in great need of *measures* which will capture which assumptions are "safe", and which assumptions are "dangerous". Obviously, safe does not mean correct, but rather captures that regardless of whether a safe assumption is true or false, it is of interest. Dangerous assumptions may be false and yet of no independent interest, thus using such assumptions in abundance poses the danger that provable security will lose its meaning.

One such measure was previously given by Naor [43], who classified assumptions based on the complexity of falsifying them. Loosely speaking,[3] an assumption is said to be *falsifiable*, if one can efficiently check whether an adversary is successful in breaking it.

We argue that the classification based on falsifiability alone has proved to be too inclusive. In particular, assumptions whose mere statement refers to the cryptographic scheme they support can be (and have been) made falsifiable. Thus, falsifiability is an important feature but not sufficient as a basis for evaluating current assumptions,[4] and in particular, it does not exclude assumptions that are construction dependent.

In this position paper, we propose a stricter classification. Our governing principle is the goal of relying on hardness assumptions that are independent of the constructions.

[3] We refer here to the notion of falsifiability as formalized by Gentry and Wichs [30], which is slightly different from the original notions proposed by Naor. We elaborate on these notions, and on the difference between them, in Sect. 2.6 and in Appendix A.

[4] We note that this was also explicitly pointed out by Naor who advocated falsifiability as an important feature, not as a sufficient one.

2 Our Classification

We formalize the notion of a *complexity assumption*, and argue that such assumptions is what we should aim for.

Intuitively, complexity assumptions are non-interactive assumptions that postulate that given an input, distributed according to an efficiently sampleable distribution $\mathcal{D}$, it is hard to compute a valid "answer" (with non-negligible advantage), where checking the validity of the answers can be done in polynomial time.

More specifically, we distinguish between two types of complexity assumptions:

1. *Search* complexity assumptions, and
2. *Decision* complexity assumptions.

Convention: Throughout this manuscript, for the sake of brevity, we refer to a family of poly-size circuits $\mathcal{M} = \{\mathcal{M}_n\}$ as a polynomial time non-uniform algorithm $\mathcal{M}$.

2.1 Search Complexity Assumptions

Each assumption in the class of search complexity assumptions consists of a pair of probabilistic polynomial-time algorithms $(\mathcal{D}, \mathcal{R})$, and asserts that there does not exist an efficient algorithm $\mathcal{M}$ that on input a random challenge x, distributed according $\mathcal{D}$, computes any value y such that $\mathcal{R}(x, y) = 1$, with non-negligible probability. Formally:

Definition 1. *An assumption is a* search complexity assumption *if it consists of a pair of probabilistic polynomial-time algorithms $(\mathcal{D}, \mathcal{R})$, and it asserts that for any efficient*[5] *algorithm $\mathcal{M}$ there exists a negligible function μ such that for every $n \in \mathbb{N}$,*

$$\Pr_{x \leftarrow \mathcal{D}(1^n)}[\mathcal{M}(x) = y \text{ s.t. } \mathcal{R}(x, y) = 1] \leq \mu(n). \tag{1}$$

Note that in Definition 1 above, we require that there is an efficient algorithm $\mathcal{R}$ that takes as input a pair (x, y) and outputs 0 or 1. One could consider a more liberal definition, of a *privately-verifiable* search complexity assumption, which is similar to the definition above, except that algorithm $\mathcal{R}$ is given not only the pair (x, y) but also the randomness r used by $\mathcal{D}$ to generate x.

Definition 2. *An assumption is a* privately-verifiable search complexity assumption *if it consists of a pair of probabilistic polynomial-time algorithms $(\mathcal{D}, \mathcal{R})$, and it asserts that for any efficient algorithm $\mathcal{M}$ there exists a negligible function μ such that for every $n \in \mathbb{N}$,*

$$\Pr_{r \leftarrow \{0,1\}^n}[\mathcal{M}(x) = y \text{ s.t. } \mathcal{R}(x, y, r) = 1 \mid x = \mathcal{D}(r)] \leq \mu(n). \tag{2}$$

[5] "Efficient" can be interpreted in several ways. We elaborate on the various interpretations below.

The class of privately-verifiable search complexity assumptions is clearly more inclusive.

What is an Efficient Algorithm? Note that in Definitions 1 and 2 above, we restricted the adversary $\mathcal{M}$ to be an *efficient algorithm*. One can interpret the class of efficient algorithms in various ways. The most common interpretation is that it consists of all non-uniform polynomial time algorithms. However, one can interpret this class as the class of all *uniform* probabilistic polynomial time algorithms, or parallel NC algorithms, leading to the notions of search complexity assumption with *uniform* security or with *parallel* security, respectively. One can also strengthen the power of the adversary $\mathcal{M}$ and allow it to be a *quantum* algorithm.

More generally, one can define a (t, ϵ) search complexity assumption exactly as above, except that we allow $\mathcal{M}$ to run in time $t(n)$ (non-uniform or uniform, unbounded depth or bounded depth, with quantum power or without) and require that it cannot succeed with probability $\epsilon(n)$ on a random challenge $x \leftarrow \mathcal{D}(1^n)$. For example, $t(n)$ may be sub-exponentially large, and $\epsilon(n)$ may be sub-exponentially small. Clearly the smaller t is, and the larger ϵ is, the weaker (and thus more reasonable) the assumption is.

Uniformity of $(\mathcal{D}, \mathcal{R})$. In Definition 1 above, we require that the algorithms $\mathcal{D}$ and $\mathcal{R}$ are *uniform* probabilistic polynomial-time algorithms. We could have considered the more general class of *non-uniform* search complexity assumptions, where we allow $\mathcal{D}$ and $\mathcal{R}$ to be *non-uniform* probabilistic polynomial-time algorithms. We chose to restrict to uniform assumptions for two reasons. First, we are not aware of any complexity assumption in the cryptographic literature that consists of *non-uniform* $\mathcal{D}$ or $\mathcal{R}$. Second, allowing these algorithms to be non-uniform makes room for assumptions whose description size grows with the size of the security parameter, which enables them to be construction specific and not of independent interest. We would like to avoid such dependence. We note that one could also consider search complexity assumptions where $\mathcal{D}$ and $\mathcal{R}$ are allowed to be quantum algorithms, or algorithms resulting from any biological process.

Examples. The class of (publicly-verifiable) search complexity assumptions includes almost all traditional search-based cryptographic assumptions, including the Factoring and RSA assumptions [47,49], the strong RSA assumption [6,26], the Discrete Log assumption (in various groups) [24], the Learning Parity with Noise (LPN) assumption [10], and the Learning with Error (LWE) assumption [48]. An exception is the computational Diffie-Hellman assumption (in various groups) [24], which is a *privately-verifiable* search complexity assumption, since given (g^x, g^y, z) it is hard to test whether $z = g^{xy}$, unless we are given x and y, which constitutes the randomness used to generate (g^x, g^y).

We note that the LPN assumption and the LWE assumption each consists of a family of complexity assumptions,[6] one assumption for each m, where m is the number of examples of noisy equations given to the adversary. However, as was observed by [29], there is a reduction between the LPN (repectively LWE) assumption with a fixed m to the LPN (repectively LWE) assumption with an arbitrary m, that incurs essentially no loss in security.

***t*-Search Complexity Assumptions.** The efficient algorithm $\mathcal{R}$ associated with a search complexity assumption can be thought of as an NP relation algorithm. We believe that it is worth distinguishing between search complexity assumptions for which with overwhelming probability, $x \leftarrow \mathcal{D}(1^n)$ has at most polynomially many witnesses, and assumptions for which with non-negligible probability, $x \leftarrow \mathcal{D}(1^n)$ has exponentially many witnesses. We caution that the latter may be too inclusive, and lead to an absurdum where the assumption assumes the security of the cryptographic scheme itself, as exemplified below.

Definition 3. *For any function $t = t(n)$, a search complexity assumption $(\mathcal{D}, \mathcal{R})$ is said to be a* t-search complexity assumption *if there exists a negligible function μ such that*

$$\Pr_{x \leftarrow \mathcal{D}(1^n)} [|\{y : (x, y) \in \mathcal{R}\}| > t] \leq \mu(n) \tag{3}$$

Most traditional search-based cryptographic assumptions are 1-search complexity assumptions; i.e., they are associated with a relation $\mathcal{R}$ for which every x has a *unique* witness. Examples include the Factoring assumption, the RSA assumption, the Discrete Log assumption (in various groups), the LPN assumption, and the LWE assumption. The square-root assumption in composite order group is an example of a 4-search complexity assumption, since each element has at most 4 square roots modulo $N = pq$.

An example of a traditional search complexity assumption that is a t-search assumption only for an exponentially large t, is the strong RSA assumption. Recall that this assumption assumes that given an RSA modulus N and a random element $y \leftarrow \mathbb{Z}_N^*$, it is hard to find *any* exponent $e \in \mathbb{Z}_N^*$ together with the e'th root $y^{e^{-1}} \bmod N$. Indeed, in some sense, the strong RSA assumption is "exponentially" stronger, since the standard RSA assumption assumes that it is

[6] Loosely speaking, the LPN assumption with error parameter $p \in (0, 1)$ (where p is a constant), asserts that for any poly-size adversary that observes polynomially many noisy linear equations of the form $\{(a_i, a_i \cdot x + e_i)\}_{i=1}^{\mathsf{poly}(n)}$, outputs x with at most negligible probability, where $x \in_R \{0, 1\}^n$ is random, all the linear equations $a_i \in_R \{0, 1\}^n$ are independent and random, and each e_i is an independent Bernoulli random variable, where $e_i = 1$ with probability p and $e_i = 0$ otherwise. The LWE assumption is similar to the LPN assumption, except that it is associated with a (possibly large) field $\mathbb{F}$. It assumes that as above, given noisy linear equations $\{(a_i, a_i \cdot x + e_i)\}_{i=1}^{\mathsf{poly}(n)}$ it is hard to find x, where now the equations are over the field $\mathbb{F}$, and $x \in_R \mathbb{F}$, each $a_i \in_R \mathbb{F}^n$, and each error e_i is independently distributed according to a discrete Gaussian distribution. We refer the reader to [48] for the precise definition.

hard to find the e'th root, for a single e, whereas the strong RSA assumption assumes that this is hard for exponentially many e's.

Whereas the strong RSA assumption is considered quite reasonable in our community, the existence of exponentially many witnesses allows for assumptions that are overly tailored to cryptographic primitives, as exemplified below.

Consider for example the assumption that a given concrete candidate two-message delegation scheme for a polynomial-time computable language L is *adaptively* sound. This asserts that there does not exist an efficient non-uniform algorithm $\mathcal{M}$ that given a random challenge from the verifier, produces an instance $x \notin L$ together with an accepting answer to the challenge. By our definition, this is a t-complexity assumption for an exponential t, which is publicly verifiable if the underlying delegation scheme is publicly verifiable, and is privately verifiable if the underlying delegation scheme is privately verifiable. Yet, this complexity assumption is an example of an absurdum where the assumption assumes the security of the scheme itself. This absurdum stems from the fact that t is exponential. If we restricted t to be polynomial this would be avoided.

We emphasize that we are not claiming that 1-search assumptions are necessarily superior to t-search assumptions for exponential t. This is illustrated in the following example pointed out to us by Micciancio and Ducas. Contrast the Shortest Integer Solution (SIS) assumption [41], which is a t-search assumption for an exponential t, with the Learning with Error (LWE) assumption, which is 1-complexity assumption. It is well known that the LWE assumption is reducible to the SIS assumption [48]. Loosely speaking, given an LWE instance one can use an SIS breaker to find short vectors in the dual lattice, and then use these vectors to solve the LWE instance. We note that a reduction in the other direction is only known via a quantum reduction [53].

More generally, clearly if Assumption A possesses properties that we consider desirable, such as being 1-search, falsifiable, robust against quantum adversaries, etc., and Assumption A is reducible to Assumption B, then the latter should be considered at least as reasonable as the former.

2.2 Decisional Complexity Assumptions

Each assumption in the class of decisional complexity assumptions consists of two probabilistic polynomial-time algorithms $\mathcal{D}_0$ and $\mathcal{D}_1$, and asserts that there does not exist an efficient algorithm $\mathcal{M}$ that on input a random challenge $x \leftarrow \mathcal{D}_b$ for a random $b \leftarrow \{0,1\}$, outputs b with non-negligible advantage.

Definition 4. *An assumption is a* decisional complexity assumption *if it is associated with two probabilistic polynomial-time distributions* $(\mathcal{D}_0, \mathcal{D}_1)$*, such that for any efficient*[7] *algorithm* $\mathcal{M}$ *there exists a negligible function* μ *such that for any* $n \in \mathbb{N}$*,*

$$\Pr_{b \leftarrow \{0,1\}, x \leftarrow \mathcal{D}_b(1^n)}[\mathcal{M}(x) = b] \leq \frac{1}{2} + \mu(n). \tag{4}$$

[7] "Efficient algorithms" can be interpreted in several ways, as we elaborated on in Sect. 2.1.

Example 1. This class includes all traditional decisional assumptions, such as the DDH assumption [24], the Quadratic Residuosity (QR) assumption [37], the N'th Residuosity assumption [44], the decisional LPN assumption [2], the decisional LWE assumption [48], the decisional linear assumption over bilinear groups [11], and the Φ-Hiding assumption [15]. Thus, this class is quite expressive. The Multilinear Subgroup Elimination assumption, which was recently proposed and used to construct IO obfuscation in [28], is another member of this class. To date, however, this assumption has been refuted in all proposed candidate (multilinear) groups [18,19,42].

An example of a decisional assumption that *does not* belong to this class is the strong DDH assumption over a prime order group G [16]. This assumption asserts that *for every* distribution $\mathcal{D}$ with min-entropy $k = \omega(\log n)$, it holds that

$$(g^r, g^x, g^{rx}) \approx (g^r, g^x, g^u),$$

where $x \leftarrow \mathcal{D}$ and $r, u \leftarrow \mathbb{Z}_p$, where p is the cardinality of G, and g is a generator of G.

This assumption was introduced by Canetti [16], who used it to prove the security of his point function obfuscation construction. Since for point function obfuscation the requirement is to get security for *every* point x, it is impossible to base security under a polynomial complexity assumption. This was shown by Wee [54], who constructed a point function obfuscation scheme under a complexity assumption with an extremely small ϵ. We note that if instead of requiring security to hold for *every* point x, we require security to hold for every distribution on inputs with min-entropy n^ϵ, for some constant $\epsilon > 0$, then we can rely on standard (polynomial) complexity assumptions, such as the LWE assumption [36], and a distributional assumption as above is not necessary.

Many versus two distributions. One can consider an "extended" decision complexity assumption which is associated with polynomially many distributions, as opposed to only two distributions. Specifically, one can consider the decision complexity assumption that is associated with a probabilistic polynomial-time distribution $\mathcal{D}$ that encodes $t = \mathsf{poly}(n)$ distributions, and the assumption is that for any efficient algorithm $\mathcal{M}$ there exists a negligible function μ such that for any $n \in \mathbb{N}$,

$$\Pr_{i \leftarrow [t], x \leftarrow \mathcal{D}(1^n, i)}[\mathcal{M}(x) = i] \leq \frac{1}{t} + \mu(n). \tag{5}$$

We note however that such an assumption can be converted into an equivalent decision assumption with two distributions $\mathcal{D}_0$ and $\mathcal{D}_1$, using the Goldreich-Levin hard-core predicate theorem [34], as follows: The distribution $\mathcal{D}_0$ will sample at random $i \leftarrow [t]$, sample at random $x \leftarrow \mathcal{D}(1^n, i)$, sample at random $r \leftarrow [t]$, and output $(x, r, r \cdot i)$. The algorithm $\mathcal{D}_1$ will similarly sample i, x, r but will output (x, r, b) for a random bit $b \leftarrow \{0, 1\}$.

2.3 Worst-Case vs. Average-Case Hardness

Note that both Definitions 1 and 4 capture *average-case* hardness assumptions, as opposed to *worst-case* hardness assumptions. Indeed, at first sight, relying on average-case hardness in order to prove the security of cryptographic schemes seems to be necessary, since the security requirements for cryptographic schemes require adversary attacks to fail with *high probability*, rather than in the worst case.

One could have considered the stricter class of *worse-case* (search or decision) complexity assumptions. A worst-case search assumption, is associated with a polynomial time computable relation $\mathcal{R}$, and requires that no polynomial-time non-uniform algorithm $\mathcal{M}$ satisfies that *for every* $x \in \{0,1\}^*$, $\mathcal{R}(x, \mathcal{M}(x)) = 1$. A worst-case decisional assumption is a promise assumption which is associated with two sets of inputs S_0 and S_1, and requires there is no polynomial-time non-uniform algorithm $\mathcal{M}$, that *for every* $x \in \{0,1\}^*$, given the promise that it is in $S_0 \cup S_1$, guesses correctly whether $x \in S_0$ or $x \in S_1$.

There are several cryptographic assumptions for which there are random self-reductions from worst-case to average-case for *fixed-parameter problems*[8]. Examples include the Quadratic-Residuosity assumption, the Discrete Logarithm assumption, and the RSA assumption [37]. In fact, the Discrete Log assumption over fields of size 2^n has a (full) worst-case to average case reduction [7].[9] Yet, we note that the Discrete Log assumption over fields of small characteristic (such as fields of size 2^n) have been recently shown to be solvable in quasi-polynomial time [5], and as such are highly vulnerable.

There are several lattice based assumptions that have a worst-case to average-case reduction [1,13,46,48]. Such worst-case assumptions are usable for cryptography, and include the GapSVP assumption [33] and the assumption that it is hard to approximate the Shortest Independent Vector Problem (SIVP) within polynomial approximation factors [41].

Whereas being a worst-case complexity assumption is a desirable property and average to worst case reductions are a goal in itself, we believe that at this point in the life-time of our field establishing the security of novel cryptographic schemes (e.g., IO obfuscation) based on an average case complexity assumption would be a triumph. We note that traditionally cryptographic hardness assumptions were average-case assumptions (as exemplified above).

2.4 Search versus Decision Complexity Assumptions

An interesting question is whether search complexity assumptions can always be converted to decision complexity assumptions and vice versa.

[8] By a "worst-case to average-case reduction for a fixed-parameter problem", we think of a problem instance as a pair (n, x) and a reduction which holds per fixed n.

[9] More generally, such a worst-case to average case reduction exists if the security parameter determines the field, its representation, and a generator of the field. As was shown by Shoup in [50,51], finding a representation (i.e., an irreducible polynomial) and a generator for fields of small characteristic can be done in polynomial time.

We note that any decision complexity assumption can be converted into a *privately-verifiable* search complexity assumption that is sound assuming the decision assumption is sound, but not necessarily into a *publicly verifiable* search complexity assumption. Consider, for example, the DDH assumption. Let f_{DDH} be the function that takes as input n tuples (where n is the security parameter), each tuple is either a DDH tuple or a random tuple, and outputs n bits, predicting for each tuple whether it is a DDH tuple or a random tuple. The direct product theorem [39] implies that if the DDH assumption is sound then it is hard to predict f_{DDH} except with negligible probability. The resulting search complexity assumption is privately-verifiable, since in order to verify whether a pair $((x_1,\ldots,x_n),(b_1,\ldots,b_n))$ satisfies that $(b_1,\ldots,b_n) = f_{\mathrm{DDH}}(x_1,\ldots,x_n)$, one needs the private randomness used to generate $(x_1,\ldots,x_n)$.

In the other direction, it would seem at first that one can map any (privately-verifiable or publicly verifiable) search complexity assumption into an equivalent decision assumption, using the hard-core predicate theorem of Goldreich and Levin [34]. Specifically, given any (privately-verifiable) search complexity assumption $(\mathcal{D},\mathcal{R})$, consider the following decision assumption: The assumption is associated with two distributions $\mathcal{D}_0$ and $\mathcal{D}_1$. The distribution $\mathcal{D}_b$ generates (x,y), where $x \leftarrow \mathcal{D}(1^n)$ and where $\mathcal{R}(x,y) = 1$, and outputs a triplet (x,r,u) where r is a random string, and if $b = 0$ then $u = r \cdot y (\mathrm{mod}\ 2)$ and if $b = 1$ then $u \leftarrow \{0,1\}$. The Goldreich-Levin hard-core predicate theorem states that the underlying search assumption is sound if and only if $x \leftarrow \mathcal{D}_0$ is computationally indistinguishable from $x \leftarrow \mathcal{D}_1$. However, $\mathcal{D}_0$ and $\mathcal{D}_1$ are efficiently sampleable only if generating a pair (x,y), such that $x \leftarrow \mathcal{D}(1^n)$ and $\mathcal{R}(x,y) = 1$, can be done efficiently. Since the definition of search complexity assumptions only assures that $\mathcal{D}$ is efficiently sampleable and does not mandate that the pair (x,y) is efficiently sampleable, the above transformation from search to decision complexity assumption does not always hold.

2.5 Concrete versus Generic Assumptions

The examples of assumptions we mentioned above are concrete assumptions. Another type of assumption made in cryptography is a *generic* assumption, such as the assumption that one-way functions exist, collision resistant hash families exist, or IO secure obfuscation schemes exist.

We view generic assumptions as *cryptographic primitives* in themselves, as opposed to cryptographic assumptions. We take this view for several reasons. First, in order to ever make use of a cryptographic protocol based on a generic assumption, we must first instantiate it with a concrete assumption. Thus, in a sense, a generic assumption is only as good as the concrete assumptions it can be based on. Second, generic assumptions are *not falsifiable*. The reason is that in order to falsify a generic assumption one needs to falsify *all* the candidates.

The one-way function primitive has the unique feature that it has a *universal* concrete instantiation, and hence is falsifiable. Namely, there exists a (universal) concrete one-way function candidate f such that if one-way functions exist then f itself is one-way [31]. This state of affairs would be the gold standard for any

generic assumption; see discussion in Sect. 2.7. Moreover, one-way functions can be constructed based on any complexity assumption, search or decision.

In the other extreme, there are generic assumptions that have no instantiation under any (search or decisional) complexity assumption. Examples include the generic assumption that there exists a 2-message delegation scheme for NP, the assumption that P-certificates exist [20], the assumption that extractable collision resistant hash functions exist [8,21,23], and the generic assumption that IO obfuscation exists.[10]

2.6 Falsifiability of Complexity Assumptions

Naor [43] defined the class of falsifiable assumptions. Intuitively, this class includes all the assumptions for which there is a constructive way to demonstrate that it is false, if this is the case. Naor defined three notions of falsifiability: *efficiently falsifiable*, *falsifiable*, and *somewhat falsifiable*. We refer the reader to Appendix A for the precise definitions.

Gentry and Wichs [30] re-formalized the notion of a falsifiable assumption. They provide a single formulation, that arguably more closely resembles the intuitive notion of falsifiability. According to [30] an assumption is falsifiable if it can be modeled as an interactive game between an efficient challenger and an adversary, at the conclusion of which the challenger can efficiently decide whether the adversary won the game. Almost all followup work that use the term of falsifiable assumptions use the falsifiability notion of [30], which captures the intuition that one can efficiently check (using randomness and interaction) whether an attacker can indeed break the assumption. By now, when researchers say that an assumption is falsifiable they most often refer to the falsifiability notion of [30]. In this paper we follow this convention.

Definition 5. *[30] A falsifiable cryptographic assumption consists of a probabilistic polynomial-time interactive challenger C. On security parameter n, the challenger $C(1^n)$ interacts with a non-uniform machine $\mathcal{M}(1^n)$ and may output a special symbol win. If this occurs, we say that $\mathcal{M}(1^n)$ wins $C(1^n)$. The assumption states that for any efficient non-uniform $\mathcal{M}$,*

$$\Pr[\mathcal{M}(1^n) \textit{ wins } C(1^n)] = \mathsf{negl}(n),$$

where the probability is over the random coins of C. For any $t = t(n)$ and $\epsilon = \epsilon(n)$, an (t, ϵ) assumption is falsifiable if it is associated with a probabilistic polynomial-time C as above, and for every $\mathcal{M}$ of size at most $t(n)$, and for every $n \in \mathbb{N}$,

$$\Pr[\mathcal{M}(1^n) \textit{ wins } C(1^n)] \leq \epsilon(n).$$

The following claim is straightforward.

Claim 1. Any (search or decision) complexity assumption is also a falsifiable assumption (according to Definition 5), but not vice versa.

[10] We note that this assumption was recently reduced to the subgroup elimination assumption [28], which is a new decisional complexity assumptions. To date, however, this assumption has been refuted in all proposed candidate (multi-linear) groups.

2.7 Desirable Properties of Complexity Assumptions

We emphasize that our classification described above is minimal and does not take into account various measures of how "robust" the assumption is. We mention two such robustness measures below.

Robustness to auxiliary inputs. One notion of robustness that was considered for *search* assumptions is that of robustness to *auxiliary inputs*.

Let us consider which auxiliary inputs may be available to an adversary of a complexity assumption. Recall that search complexity assumptions are associated with a pair of probabilistic polynomial time algorithms $(\mathcal{D}, \mathcal{R})$ where the algorithm $\mathcal{D}$ generates instances $x \leftarrow \mathcal{D}$ and the assumption is that given $x \leftarrow \mathcal{D}$ it is computationally hard to find y such that $(x, y) \in \mathcal{R}$. As it turns out however, for all known search assumptions that are useful in cryptography, it is further the case that one can efficiently generate not only an instance $x \leftarrow \mathcal{D}$, but pairs (x, y) such that $(x, y) \in \mathcal{R}$. Indeed, it is what most often makes the assumption useful in a cryptographic context. Typically, in a classical adversarial model, y is part of the secret key, whereas x is known to the adversary. Yet due to extensive evidence a more realistic adversarial model allows the adversary access to partial knowledge about y which can be viewed generally as access to an auxiliary input.

Thus, one could have defined a search complexity assumption as a pair $(\mathcal{D}, \mathcal{R})$ as above, but where the algorithm $\mathcal{D}$ generates pairs (x, y) (as opposed to only x), such that $(x, y) \in \mathcal{R}$ and the requirement is that any polynomial-size adversary who is given only x, outputs some y' such that $(x, y') \in \mathcal{R}$, only with negligible probability. This definition is appropriate when considering robustness to auxiliary information. Informally, such a search assumption is said to be resilient to auxiliary inputs if given an instance x sampled according to $\mathcal{D}$, and given some auxiliary information about the randomness used by $\mathcal{D}$ (and in particular, given some auxiliary information about y), it remains computationally hard to find y' such that $(x, y') \in \mathcal{R}$.

Definition 6. *A search complexity assumption $(\mathcal{D}, \mathcal{R})$ as above is said to be resilient to $t(n)$-hard-to-invert auxiliary inputs if for any $t(n)$-hard-to-invert function $L : \{0,1\}^n \to \{0,1\}^*$,*

$$\Pr_{r \leftarrow \{0,1\}^n, (x,y) \leftarrow \mathcal{D}(r)} [\mathcal{M}(x, L(r)) = y' \text{ s.t. } \mathcal{R}(x, y') = 1] \leq \mu(n), \tag{6}$$

where L is said to be $t(n)$-hard-to-invert if for every $t(n)$-time non-unform algorithm $\mathcal{M}$ there exists a negligible μ such that for every $n \in \mathbb{N}$,

$$\Pr_{z \leftarrow L(U_n)} [\mathcal{M}(z) = r : L(r) = z] = \mu(n). \tag{7}$$

It was shown in [36] that the decisional version of the LWE assumption is resilient to $t(n)$-hard-to-invert auxiliary inputs for $t(n) = 2^{n^\delta}$, for any constant $\delta > 0$. In particular, this implies that the LWE assumption is robust to leakage attacks. In contrast, the RSA assumptions is known to be completely broken even if only 0.27 fraction of random bits of the secret key are leaked [38].

Universal assumptions. We say that a (concrete) complexity assumption A is *universal* with respect to a generic assumption if the following holds: If A is false then the generic assumption is false. In other words, if the generic assumption has a concrete sound instantiation then A is it. Today, the only generic assumption for which we know a universal instantiation is one-way functions [31].

Open Problem: We pose the open problem of finding a universal instantiations for other generic assumptions, in particular for IO obfuscation, witness encryption, or 2-message delegation for NP.

3 Recently Proposed Cryptographic Assumptions

Recently, there has been a proliferation of cryptographic assumptions. We next argue that many of the recent assumptions proposed in the literature, *even the falsifiable* ones, are *not* complexity assumptions.

IO Obfuscation constructions. Recently, several constructions of IO obfuscation have been proposed. These were proved under ad-hoc assumptions [27], meta assumptions [45], and ideal-group assumptions [4,14]. These assumptions are not complexity assumptions, for several reasons: They are either overly tailored to the construction, or artificially restrict the adversaries.

The recent result of [28] constructed IO obfuscation under a new complexity assumption, called Subgroup Elimination assumption. This is a significant step towards constructing IO under a standard assumption. However, to date, this assumption is known to be false in all candidate (multi-linear) groups.

Assuming IO obfuscation exists. A large body of work which emerged since the construction of [27], constructs various cryptographic primitives assuming IO obfuscation exists. Some of these results require only the existence of IO obfuscation for circuits with only polynomially many inputs (eg., [9]). Note that any instantiation of this assumption is falsifiable. Namely, the assumption that a given obfuscation candidate $\mathcal{O}$ (for circuits with polynomially many inputs) is IO secure, is falsifiable. The reason is that to falsify it one needs to exhibit two circuits C_0 and C_1 in the family such that $C_0 \equiv C_1$, and show that it can distinguish between $\mathcal{O}(C_0)$ and $\mathcal{O}(C_1)$. Note that since the domain of C_0 and C_1 consists of polynomially many elements one can efficiently test whether indeed $C_0 \equiv C_1$, and of course the falsifier can efficiently prove that $\mathcal{O}(C_0) \not\approx \mathcal{O}(C_1)$ by showing that one can distinguish between these two distributions. On the other hand, this is not a complexity assumption. Rather, such an assumption consists of many (often exponentially many) decision complexity assumptions: For every $C_0 \equiv C_1$ in the family $\mathcal{C}_n$ (there are often exponentially many such pairs), the corresponding decision complexity assumption is that $\mathcal{O}(C_0) \approx \mathcal{O}(C_1)$. Thus, intuitively, such an assumption is exponentially weaker than a decisional complexity assumption.

Artificially restricted adversaries assumptions. We next consider the class of assumptions that make some "artificial" restriction on the adversary. Examples include the Random Oracle Model (ROM) [25] and various generic group models [12,52]. The ROM restricts the adversary to use a given hash function only in a black-box manner. Similarly, generic group assumptions assume the adversary uses the group structure only in an "ideal" way. Another family of assumptions that belongs to this class is the family knowledge assumptions. Knowledge assumptions artificially restrict the adversaries to compute things in a certain way. For example, the Knowledge-of-Exponent assumption [22] assumes that any adversary that given (g, h) computes (g^z, h^z), must do so by "first" computing z and then computing (g^z, h^z).

We note that such assumptions cannot be written even as exponentially many complexity assumptions. Moreover, for the ROM and the generic group assumptions, we know of several examples of *insecure* schemes that are proven secure under these assumptions [3,17,35].

We thus believe that results that are based on such assumption should be viewed as *intermediate* results, towards the goal of removing such artificial constraints and constructing schemes that are provably secure under complexity assumptions.

4 Summary

Theoretical cryptography is in great need for a methodology for classifying assumptions. In this paper, we define the class of search and decision *complexity assumptions*. An overall guiding principle in the choices we made was to rule out hardness assumptions which are construction dependent.

We believe that complexity assumptions as we defined them are general enough to capture all "desirable" assumptions, and we are hopeful that they will suffice in expressive power to enable proofs of security for sound constructions. In particular, all traditional cryptographic assumptions fall into this class.

We emphasize, that we do not claim that all complexity-based complexity assumptions are necessarily desirable or reasonable. For example, false complexity assumptions are clearly not reasonable. In addition, our classification does not incorporate various measures of how "robust" an assumption is, such as: how well studied the assumption is, whether it is known to be broken by quantum attacks, whether it has a worst-case to average-case reduction, or whether it is known to be robust to auxiliary information.

Acknowledgements. We would like to thank many colleagues for their comments on an earlier draft of this work, including Ducas, Goldreich Micianccio, Peikert, Regev, Sahai, and Vaikuntanathan. In particular, we are grateful to Micciancio for extensive illuminating discussions on many aspects of hardness assumptions on lattices, and to Sahai and Vaikuntanathan for clarifying discussions on the strength of the sub-group elimination assumption used for IO obfuscation. Thanks to Bernstein for pointing out a long overlooked worst-case to average-case reduction for discrete-log in fields of small characteristic.

A Falsifiable Assumptions

Naor [43] defined three notions of falsifiability: *efficiently falsifiable*, *falsifiable*, and *somewhat falsifiable*.

Definition 7. *A (t, ϵ) assumption is* efficiently falsifiable *if there exists a family of distributions $\{\mathcal{D}_n\}_{n\in\mathbb{N}}$, a verifier $V : \{0,1\}^* \times \{0,1\}^* \to \{0,1\}$, such that the following holds for any parameter $\delta > 0$:*

1. *If the assumption is false then there exists a falsifier $\mathcal{B}$ that satisfies*

$$\Pr_{x\to\mathcal{D}_n}[\mathcal{B}(x) = y \text{ s.t. } V(x,y) = 1] \geq 1 - \delta. \tag{A.1}$$

   Moreover, the runtime of $\mathcal{B}$ is polynomial in the runtime of the adversary that breaks the assumption and polynomial in $n, \log 1/\epsilon, \log 1/\delta$.
2. *The runtime of V and the time it takes to sample an element from $\mathcal{D}_n$ is $\mathsf{poly}(n, \log 1/\epsilon, \log 1/\delta)$.*
3. *If there exists a falsifier $\mathcal{B}$ that runs in time t and solves random challenges $x \leftarrow \mathcal{D}_n$ with probability γ, then there exists an adversary $\mathcal{A}$ that runs in time $\mathsf{poly}(t)$ and breaks the original assumption with probability $\mathsf{poly}(\gamma)$.*

Definition 8. *A (t, ϵ) assumption is* falsifiable *if everything is as in Definition 7 except that the runtime of V and of sampling $\mathcal{D}_n$ may depend on $1/\epsilon$ (as opposed to $\log 1/\epsilon$).*

Definition 9. *A (t, ϵ) assumption is* somewhat falsifiable *if everything is as in Definition 7 except that the runtime of V and of sampling $\mathcal{D}_n$ may depend on $1/\epsilon$ (as opposed to $\log 1/\epsilon$), and on the runtime of $\mathcal{B}$. In particular, this means that V may simulate $\mathcal{B}$.*

Remark 1. We note that any efficiently falsifiable assumption is also a relation-based complexity assumption. However, we find the notion of efficiently falsifiable to be very restrictive, since intuitively it only includes assumptions that are random self reducible. The definition of falsifiable is less restrictive, however a falsifiable assumption is not necessarily a complexity assumption, since in order to verify a break of the assumption one needs to run in time $1/\epsilon$ which is super-polynomial. We view the notion of somewhat falsifiable to be too weak. Allowing the runtime of the verifier to depend on the runtime of the falsifier $\mathcal{B}$ makes this class very inclusive, and it includes many interactive assumptions (we refer the reader to [43] for details).

References

1. Ajtai, M.: Generating hard instances of lattice problems (extended abstract). In: Proceedings of the Twenty-Eighth Annual ACM Symposium on the Theory of Computing, 22–24 May 1996, Philadelphia, Pennsylvania, USA, pp. 99–108 (1996)

2. Alekhnovich, M.: More on average case vs approximation complexity. In: Proceedings of the 44th Symposium on Foundations of Computer Science (FOCS 2003), 11–14 October 2003, Cambridge, MA, USA, pp. 298–307 (2003)
3. Barak, B.: How to go beyond the black-box simulation barrier. In: 42nd Annual Symposium on Foundations of Computer Science, FOCS 2001, 14–17 October 2001, Las Vegas, Nevada, USA, pp. 106–115 (2001)
4. Barak, B., Garg, S., Kalai, Y.T., Paneth, O., Sahai, A.: Protecting obfuscation against algebraic attacks. In: Nguyen, P.Q., Oswald, E. (eds.) EUROCRYPT 2014. LNCS, vol. 8441, pp. 221–238. Springer, Heidelberg (2014)
5. Barbulescu, R., Gaudry, P., Joux, A., Thomé, E.: A heuristic quasi-polynomial algorithm for discrete logarithm in finite fields of small characteristic. In: Nguyen, P.Q., Oswald, E. (eds.) EUROCRYPT 2014. LNCS, vol. 8441, pp. 1–16. Springer, Heidelberg (2014)
6. Barić, N., Pfitzmann, B.: Collision-free accumulators and fail-stop signature schemes without trees. In: Fumy, W. (ed.) EUROCRYPT 1997. LNCS, vol. 1233, pp. 480–494. Springer, Heidelberg (1997)
7. Bernstein, D.J.: Private communication (2015)
8. Bitansky, N., Canetti, R., Chiesa, A., Goldwasser, S., Lin, H., Rubinstein, A., Tromer, E.: The hunting of the SNARK. IACR Cryptology ePrint Archive 2014, p. 580 (2014)
9. Bitansky, N., Garg, S., Lin, H., Pass, R., Telang, S.: Succinct randomized encodings and their applications. In: Proceedings of the Forty-Seventh Annual ACM on Symposium on Theory of Computing, STOC 2015, 14–17 June 2015, Portland, OR, USA, pp. 439–448 (2015)
10. Blum, A., Kalai, A., Wasserman, H.: Noise-tolerant learning, the parity problem, and the statistical query model. In: Proceedings of the Thirty-Second Annual ACM Symposium on Theory of Computing, May 21–23, 2000, Portland, OR, USA, pp. 435–440 (2000)
11. Boneh, D., Boyen, X., Shacham, H.: Short group signatures. In: Franklin, M. (ed.) CRYPTO 2004. LNCS, vol. 3152, pp. 41–55. Springer, Heidelberg (2004)
12. Boneh, D., Lipton, R.J.: Algorithms for black-box fields and their application to cryptography. In: Koblitz, N. (ed.) CRYPTO 1996. LNCS, vol. 1109, pp. 283–297. Springer, Heidelberg (1996)
13. Brakerski, Z., Langlois, A., Peikert, C., Regev, O., Stehlé, D.: Classical hardness of learning with errors. In: Symposium on Theory of Computing Conference, STOC 2013, 1–4 June 2013, Palo Alto, CA, USA, pp. 575–584 (2013)
14. Brakerski, Z., Rothblum, G.N.: Virtual black-box obfuscation for all circuits via generic graded encoding. In: Lindell, Y. (ed.) TCC 2014. LNCS, vol. 8349, pp. 1–25. Springer, Heidelberg (2014)
15. Cachin, C., Micali, S., Stadler, M.A.: Computationally private information retrieval with polylogarithmic communication. In: Stern, J. (ed.) EUROCRYPT 1999. LNCS, vol. 1592, pp. 402–414. Springer, Heidelberg (1999)
16. Canetti, R.: Towards realizing random oracles: hash functions that hide all partial information. In: Kaliski Jr., B.S. (ed.) CRYPTO 1997. LNCS, vol. 1294, pp. 455–469. Springer, Heidelberg (1997)
17. Canetti, R., Goldreich, O., Halevi, S.: The random oracle methodology, revisited. J. ACM **51**(4), 557–594 (2004). http://doi.acm.org/10.1145/1008731.1008734
18. Cheon, J.H., Han, K., Lee, C., Ryu, H., Stehlé, D.: Cryptanalysis of the multilinear map over the integers. In: Oswald, E., Fischlin, M. (eds.) EUROCRYPT 2015. LNCS, vol. 9056, pp. 3–12. Springer, Heidelberg (2015)

19. Cheon, J.H., Lee, C., Ryu, H.: Cryptanalysis of the new CLT multilinear maps. IACR Cryptology ePrint Archive 2015, p. 934 (2015)
20. Chung, K., Lin, H., Pass, R.: Constant-round concurrent zero knowledge from p-certificates. In: 54th Annual IEEE Symposium on Foundations of Computer Science, FOCS 2013, 26–29 October, 2013, Berkeley, CA, USA, pp. 50–59 (2013)
21. Di Crescenzo, G., Lipmaa, H.: Succinct NP proofs from an extractability assumption. In: Beckmann, A., Dimitracopoulos, C., Löwe, B. (eds.) CiE 2008. LNCS, vol. 5028, pp. 175–185. Springer, Heidelberg (2008)
22. Damgård, I.B.: Towards practical public key systems secure against chosen ciphertext attacks. In: Feigenbaum, J. (ed.) CRYPTO 1991. LNCS, vol. 576, pp. 445–456. Springer, Heidelberg (1992)
23. Damgård, I., Faust, S., Hazay, C.: Secure two-party computation with low communication. In: Cramer, R. (ed.) TCC 2012. LNCS, vol. 7194, pp. 54–74. Springer, Heidelberg (2012)
24. Diffie, W., Hellman, M.E.: New directions in cryptography. IEEE Trans. Inf. Theory **22**(6), 644–654 (1976)
25. Fiat, A., Shamir, A.: How to prove yourself: practical solutions to identification and signature problems. In: Odlyzko, A.M. (ed.) CRYPTO 1986. LNCS, vol. 263, pp. 186–194. Springer, Heidelberg (1987)
26. Fujisaki, E., Okamoto, T.: Statistical zero knowledge protocols to prove modular polynomial relations. In: Kaliski Jr., B.S. (ed.) CRYPTO 1997. LNCS, vol. 1294, pp. 16–30. Springer, Heidelberg (1997)
27. Garg, S., Gentry, C., Halevi, S., Raykova, M., Sahai, A., Waters, B.: Candidate indistinguishability obfuscation and functional encryption for all circuits. In: 54th Annual IEEE Symposium on Foundations of Computer Science, FOCS 2013, 26–29 October, 2013, Berkeley, CA, USA, pp. 40–49 (2013)
28. Gentry, C., Lewko, A.B., Sahai, A., Waters, B.: Indistinguishability obfuscation from the multilinear subgroup elimination assumption. IACR Cryptology ePrint Archive 2014, p. 309 (2014)
29. Gentry, C., Peikert, C., Vaikuntanathan, V.: Trapdoors for hard lattices and new cryptographic constructions. In: Proceedings of the 40th Annual ACM Symposium on Theory of Computing, 17–20 May 2008, Victoria, British Columbia, Canada, pp. 197–206 (2008)
30. Gentry, C., Wichs, D.: Separating succinct non-interactive arguments from all falsifiable assumptions. In: STOC, pp. 99–108 (2011)
31. Goldreich, O.: The Foundations of Cryptography: Basic Techniques, vol. 1. Cambridge University Press, Cambridge (2001)
32. Goldreich, O.: Randomness and computation. In: Goldreich, O. (ed.) Studies in Complexity and Cryptography. LNCS, vol. 6650, pp. 507–539. Springer, Heidelberg (2011)
33. Goldreich, O., Goldwasser, S.: On the limits of nonapproximability of lattice problems. J. Comput. Syst. Sci. **60**(3), 540–563 (2000)
34. Goldreich, O., Levin, L.A.: A hard-core predicate for all one-way functions. In: Proceedings of the 21st Annual ACM Symposium on Theory of Computing, 14–17 May 1989, Seattle, Washigton, USA, pp. 25–32 (1989)
35. Goldwasser, S., Kalai, Y.T.: On the (in)security of the Fiat-Shamir paradigm. In: Proceedings of the 44th Symposium on Foundations of Computer Science (FOCS 2003), 11–14 October 2003, Cambridge, MA, USA, pp. 102–113 (2003)

36. Goldwasser, S., Kalai, Y.T., Peikert, C., Vaikuntanathan, V.: Robustness of the-learning with errors assumption. In: Proceedings of the Innovations in Computer Science, ICS 2010, 5–7 January 2010, Tsinghua University, Beijing, China, pp. 230–240 (2010). http://conference.itcs.tsinghua.edu.cn/ICS2010/content/papers/19.html
37. Goldwasser, S., Micali, S.: Probabilistic encryption. J. Comput. Syst. Sci. **28**(2), 270–299 (1984)
38. Heninger, N., Shacham, H.: Reconstructing RSA private keys from random key bits. In: Halevi, S. (ed.) CRYPTO 2009. LNCS, vol. 5677, pp. 1–17. Springer, Heidelberg (2009)
39. Impagliazzo, R., Jaiswal, R., Kabanets, V., Wigderson, A.: Uniform direct product theorems: simplified, optimized, and derandomized. SIAM J. Comput. **39**(4), 1637–1665 (2010)
40. Khot, S.: On the unique games conjecture. In: Proceedings of the 46th Annual IEEE Symposium on Foundations of Computer Science (FOCS 2005), 23–25 October 2005, Pittsburgh, PA, USA, p. 3 (2005)
41. Micciancio, D., Goldwasser, S.: Complexity of Lattice Problems: A Cryptographic Perspective, vol. 671. Springer Science & Business Media, New York (2012)
42. Minaud, B., Fouque, P.: Cryptanalysis of the new multilinear map over the integers. IACR Cryptology ePrint Archive 2015, p. 941 (2015)
43. Naor, M.: On cryptographic assumptions and challenges. In: Boneh, D. (ed.) CRYPTO 2003. LNCS, vol. 2729, pp. 96–109. Springer, Heidelberg (2003)
44. Paillier, P.: Public-key cryptosystems based on composite degree residuosity classes. In: Stern, J. (ed.) EUROCRYPT 1999. LNCS, vol. 1592, pp. 223–238. Springer, Heidelberg (1999)
45. Pass, R., Seth, K., Telang, S.: Indistinguishability obfuscation from semantically-secure multilinear encodings. In: Garay, J.A., Gennaro, R. (eds.) CRYPTO 2014, Part I. LNCS, vol. 8616, pp. 500–517. Springer, Heidelberg (2014)
46. Peikert, C.: Public-key cryptosystems from the worst-case shortest vector problem: extended abstract. In: Proceedings of the 41st Annual ACM Symposium on Theory of Computing, STOC 2009, May 31 - June 2, 2009, Bethesda, MD, USA, pp. 333–342 (2009)
47. Rabin, M.O.: Digitalized signatures and public-key functions as intractable as factorization. Technical report, MIT Laboratory for Computer Science (1979)
48. Regev, O.: On lattices, learning with errors, random linear codes, and cryptography. In: Proceedings of the 37th Annual ACM Symposium on Theory of Computing, 22–24 May 2005, Baltimore, MD, USA, pp. 84–93 (2005)
49. Rivest, R.L., Shamir, A., Adleman, L.M.: A method for obtaining digital signatures and public-key cryptosystems. Commun. ACM **21**(2), 120–126 (1978)
50. Shoup, V.: New algorithms for finding irreducible polynomials over finite fields. In: 29th Annual Symposium on Foundations of Computer Science, 24–26 October 1988, White Plains, New York, USA, pp. 283–290 (1988)
51. Shoup, V.: Searching for primitive roots in finite fields. In: Proceedings of the 22nd Annual ACM Symposium on Theory of Computing, 13–17 May 1990, Baltimore, Maryland, USA, pp. 546–554 (1990)
52. Shoup, V.: Lower bounds for discrete logarithms and related problems. In: Fumy, W. (ed.) EUROCRYPT 1997. LNCS, vol. 1233, pp. 256–266. Springer, Heidelberg (1997)
53. Stehlé, D., Steinfeld, R., Tanaka, K., Xagawa, K.: Efficient public key encryption based on ideal lattices. In: Matsui, M. (ed.) ASIACRYPT 2009. LNCS, vol. 5912, pp. 617–635. Springer, Heidelberg (2009)
54. Wee, H.: On obfuscating point functions. In: Proceedings of the 37th Annual ACM Symposium on Theory of Computing, 22–24 May 2005, Baltimore, MD, USA, pp. 523–532 (2005). http://doi.acm.org/10.1145/1060590.1060669