Notions of Reducibility between Cryptographic Primitives*

Omer Reingold[1**], Luca Trevisan[2***], and Salil Vadhan[3†]

[1] AT&T Labs - Research.
Room A201, 180 Park Avenue, Bldg. 103
Florham Park, NJ, 07932.
omer@research.att.com

[2] Computer Science Division
U.C. Berkeley
615 Soda Hall
Berkeley, CA 94720
luca@cs.berkeley.edu

[3] Division of Engineering & Applied Sciences
Harvard University
33 Oxford Street
Cambridge, MA 02138
salil@eecs.harvard.edu

Abstract. Starting with the seminal paper of Impagliazzo and Rudich [18], there has been a large body of work showing that various cryptographic primitives cannot be reduced to each other via "black-box" reductions. The common interpretation of these results is that there are inherent limitations in using a primitive as a black box, and that these impossibility results can be overcome only by explicitly using the *code of the primitive* in the *construction*.

In this paper we revisit these negative results, we give a more careful taxonomy of the ways in which "black-box reductions" can be formalized, we strengthen some previous results (in particular we give unconditional proofs of results that were previously proved only assuming $P = NP$), and we offer a new interpretation of them: that, in many cases, there is no limitation in using a primitive as a black box, but there is a limitation in treating *adversaries* as such. In particular, these negative results may be overcome by using the *code of the adversary* in the *analysis*.

* Research was supported in part by US-Israel Binational Science Foundation Grant 2002246.

** Part of this research was performed while visiting the Institute for Advanced Study, Princeton, NJ.

*** Supported by NSF grant CCR-9984703, a Sloan Research Fellowship and an Okawa Foundation Grant.

† Supported by NSF Grant CCR-0205423 and a Sloan Research Fellowship.

1 Introduction

In most of the current body of work in the foundations of cryptography, cryptographic protocols are not shown to be unconditionally secure, but, rather, their security is reduced to the security of seemingly weaker or simpler primitives. We now know that, if one way functions exist, then private-key encryption and authentication are possible, as well as (public-key) digital signatures and zero-knowledge proofs [14, 12, 23, 21, 13]. On the other hand, if one-way functions do not exist then most interesting cryptographic problems, including all of the above, have no solution [15, 22].

Some cryptographic primitives, however, such as public-key encryption, key agreement, oblivious transfer, collision-resistant hash functions, and non-interactive zero knowledge, are not known to be equivalent to the existence of one-way functions. Furthermore, several of the known constructions based on one-way functions (in particular, the construction of pseudorandom generators from one-way functions [14], which is a component of several other constructions) run in polynomial time but are extremely inefficient. Since these are some of the main gaps in our systematization of the foundations of cryptography, it is natural to ask whether additional primitives, such as public-key encryption, can be constructed from one-way functions, and whether known constructions can be made more efficient. One has to be careful in formalizing such questions. It is commonly believed that one-way functions exist and that public-key encryption is possible, which would mean that the existence of one-way functions *implies* the existence of public key encryption in a trivial logical sense. The question is whether *the techniques that we typically use to prove implications of one-way functions in cryptography* have some inherent limitation that prevents us from deriving the existence of public-key encryption.

Impagliazzo and Rudich [18] were the first to give a formal treatment of such issues. They observed that most implications in cryptography are proved using a reduction, where the primitive is treated as an oracle, or a "black box," and the analysis shows that if the primitive is secure in a black-box sense then the construction is also secure. Impagliazzo and Rudich consider various black-box settings (where there are some additional constraints beyond the primitive being treated as a black box) and show that, in one model, a black-box construction of key agreement based on one-way functions implies a proof that $P \neq NP$; in a more constrained model such a construction is unconditionally impossible. The formal framework of Impagliazzo and Rudich has subsequently been used to address other "implication" questions, such as one-way functions versus one-way permutations [17, 19], one-way functions versus collision-resistant hash functions [25], between key agreement, oblivious transfer, public-key encryption and trapdoor functions and permutations [9, 10]. Variations of the framework have also been used to address the issue of the number of rounds in KA protocols [24], of the efficiency of constructions of universal one-way hash functions based on one-way permutations [20, 8], of pseudorandom generators based on one-way permutations [8] and of public-key encryption based on trapdoor permutations [7].

The common interpretation of these results is that there are inherent limitations in using a primitive as a black box, and that these impossibility results can be overcome only by explicitly using the *code of the primitive* in the *construction*.

In this paper we revisit these negative results, we give a more careful taxonomy of the ways in which "black-box reductions" can be formalized, we strengthen some previous results (in particular we give unconditional proofs of results that were previously proved only assuming $P = NP$), and we offer a new interpretation of them: that, in many cases, there is no limitation in using a primitive as a black box, but there is a limitation in treating *adversaries* as such. In particular, these negative results may be overcome by using the *code of the adversary* in the *analysis*.

1.1 Impossibility Results for Reductions

The starting point of the work of Impagliazzo-Rudich is the observation that most known cryptographic constructions based on one-way functions treat the one-way function as a "black box." (Exceptions are discussed in Section 1.5.) Roughly speaking, a *black-box (BB) reduction* of a primitive Q to one-way functions (OWF) is a construction that uses oracle access to a function f, and guarantees that if f is one-way then the construction is secure. In particular:

- The construction does not use the code of the function f;
- The construction is well defined and efficient even if f is not efficiently computable (as long as it is given as an oracle);
- There is a proof security that shows that an adversary breaking the protocol yields an adversary that inverts f.

There are various ways to formalize the third condition (which we make precise in Section 2. One possibility considered in [18], which we call *fully-BB*, is that there is an algorithm that converts every adversary that supposedly breaks the construction (according to the definition of security for Q) into a procedure that inverts f. This algorithm is efficient and it is given oracle access to the adversary and to f. In this setting, both the *construction* and the *analysis* are black box. Another way to look at it is that both the *primitive* and the *adversary* are treated as black boxes. Most reductions in the cryptography literature are fully-BB.

Impagliazzo and Rudich [18] prove that there can be no fully-BB reduction of key agreement (KA) to OWF. Since public-key encryption, trapdoor permutations and oblivious transfer all imply KA (by fully-BB reductions), it then follows that there are no fully-BB transformations of OWF into these other primitives as well. One may (and should) wonder whether the impossibility is due to the fact both the primitive and the adversaries are treated as oracles, or if it is enough that just the primitive is.

Impagliazzo and Rudich also consider a weaker form a BB reduction of KA to OWF, a form that we call *semi-BB* in this paper. In a semi-BB reduction, we have a BB construction of KA based on a function f given as an oracle. The analysis proves that for every efficient adversary with oracle to f that breaks the construction, there is an efficient adversary that inverts f if given oracle access to f. This seems to formalize the notion of a BB construction with an arbitrary analysis, but we argue that it does not. If f is a one-way function in the black-box sense,[4] then the construction has to be secure not only against efficient adversaries, but also against adversaries that have oracle access to f. A proof technique that makes use of the code of the adversary is not BB in this sense.

Impagliazzo and Rudich prove that, if $P = NP$, there is no semi-BB reduction of KA to OWF. This means that, in order to come up with a proof that OWF implies KA, one must either avoid semi-BB reductions or find, along the way, a proof that $P \neq NP$. Impagliazzo and Rudich prove their result by establishing the stronger (and independently interesting) statement that if $P = NP$, then there is no secure KA in the random oracle model. (Note that a random oracle is one-way in the black-box sense even if P=NP.)

1.2 The Limitations of Semi-BB Reductions

In this paper we prove, unconditionally, that there is no semi-BB reduction of OWF to KA. We prove this unconditional result by embedding a PSPACE oracle into a small part of the random oracle used in the Impagliazzo–Rudich result, and use the fact that $P^{PSPACE} = NP^{PSPACE}$. This embedding technique is due to Simon [25].

Following the lead of Impagliazzo and Rudich, several other works explored the limitations of black-box reductions with examples being [24, 25, 20, 8–10]. Most results ruled out fully-BB reductions unconditionally, and semi-BB reductions if P=NP. An exception is the work of Gertner et al [10], which involves a model that is slightly different from the one of [18], and which only rules out fully-BB reductions. The embedding technique allows us to prove that semi-BB reductions are unconditionally impossible in all case where semi-BB reductions were previously conditionally ruled out.

More generally, we show that, under mild conditions satisfied by most natural primitives, semi-BB reductions are equivalent to *relativizing reductions* (proofs that the implication holds relative to any oracle). Since the above works rule out relativizing reductions unconditionally, we obtain unconditional impossibility of semi-BB reductions.

1.3 The Power of Weakly-BB Reductions

Semi-BB reductions have typically been considered to be BB constructions with arbitrary proofs, and negative results about semi-BB reductions have typically

[4] Meaning that no efficient procedure with oracle access to f can invert f on a non-negligible fraction of inputs.

been interpreted as limitations for constructions that do not use the code of the primitive. In this paper, we present a different perspective.

We first formalize the notion of a BB construction with an arbitrary proof, which we call a weakly-BB reduction. In a weakly-BB reduction of, say, KA to OWF, the construction refers to an oracle function, and it is secure whenever the oracle function is one-way in a black-box sense, but the *analysis* of the construction may be arbitrary. This means that for every oracle f and for every efficient adversary that breaks the KA protocol constructed from f, there is an efficient procedure that inverts f when given oracle access to f. The difference with semi-BB is that we do not care about KA adversaries that require oracle access to f to be efficiently realized.

A first observation is that if we had a provably secure KA scheme, then it would also be a weakly-BB reduction of OWF to KA: just let the parties ignore the oracle, and then the security of the construction in the real world implies that it is also secure as a weakly-BB reduction. This means that it is unrealistic to look for an unconditional proof that weakly-BB reductions of OWF to KA do not exist; indeed, most likely, such a weakly-BB reduction exists. However one can still wonder whether the only way to come up with a weakly-BB reduction is to "cheat" in this manner, and have the analysis of the construction contain the proof of a strong lower bound (so that the intractability comes not from the primitive used as an oracle but from the proof of correctness of the reduction).

A similar situation arises in the random oracle model studied by Impagliazzo and Rudich: a secure KA protocol in the real world would also be secure in the random oracle model. However, Impagliazzo and Rudich show that if $P = NP$ then there can be no secure construction of KA in the random oracle model. That is, the only way to construct a secure KA in the random oracle model is to come up with a proof that $P \neq NP$ along the way.

One may conjecture that, similarly to the Impagliazzo–Rudich result, if $P = NP$ then there is no weakly-BB reduction of KA to OWF. Perhaps surprisingly, we prove that the opposite is true: if $P = NP$ then *there is* a weakly-BB reduction of KA to OWF. Indeed, such a reduction exists even under the weaker assumption that OWFs do not exist.[5] In other words, if KA is possible, then there is weakly-BB reduction of OWF to ioKA, and if OWF do not exist then there is also a weakly-BB reduction of OWF to KA. That is, if OWF imply KA in the logical sense (i.e., unless OWF exist but KA is impossible) then the implication can be proved using weakly-BB reductions.[6] We feel that this result is important because it shows that there is no inherent limitation (at least in KA versus OWF) in ignoring the code of the primitive, although there are limitations in ignoring the code of the adversary as well.

[5] Actually, the reduction only provides "infinitely-often KA" (ioKA) from one-way functions; see Section 4.

[6] To be precise, our result leaves out the case in which ioKA exist but KA do not exist. Even in such a case, it is possible to argue that for every input length, if OWF imply KA in the logical sense for that input length, then the implication can be established with a weakly-BB reduction.

We similarly show that weakly-BB reductions are as powerful as arbitrary reductions in transforming OWF to one-way permutations, to collision-resistant hash functions, to trapdoor permutations, and other primitives.

1.4 Efficiency of Reductions

We next turn our attention to another line of research about the limitations of black-box reductions, namely, the *efficiency* of reductions. The issue of efficiency was first raised by Rudich [24], who investigated the round complexity of KA schemes. Rudich proved that one cannot use a fully-BB reduction to transform a k-round KA scheme into a $(k-1)$-round one. Later, Kim, Simon and Tetali [20] considered the question of efficiency of constructions of universal one-way hash functions (UOWHFs) based on one-way permutations (OWPs). The known reduction is fully black box and invokes the OWP a number of times that is roughly linear in the compression of the UOWHF. Kim et al. [20] show that every fully-BB construction must invoke the OWP a number of times that is about the square root of the expansion.

Gennaro and Trevisan [8] considered again the question of reductions of OWPs to UOWHF, as well as the question of constructions of pseudorandom generators (PRGs) based on OWPs. The Blum-Micali-Yao construction [3, 26, 11] invokes the OWP a number of times that is roughly linear in the expansion of the generator. Gennaro and Trevisan proved that if OWF do not exist, then there is no weakly-BB transformation of OWP to PRG and no weakly-BB transformation of OWP to UOWHF where the OWP is invoked a sub-linear number of times (sub-linear in the expansion and in the compression, respectively). On the other hand, if OWF do exist, then there are zero-query weakly-BB constructions. This means that the only way of improving current constructions, even with a weakly-BB reduction, is to come up with an unconditional construction and disregard the oracle.[7] Gennaro, Gertner and Katz [7] gave similar results for constructions of public-key encryption and signature schemes. These results by Gennaro et al. [8, 7] about the efficiency of reductions are the only ones that rule out even *weakly*-BB reductions.

Regarding the efficiency of known reductions in cryptography, perhaps the most pressing open question is whether the construction of PRG based on OWF by Håstad et al. [14] can be made more efficient. It was conjectured in [8] that black-box transformations of OWF into PRG have to invoke the OWF a super-linear number of times. In this paper, we show that there is a weakly-BB construction of PRG based on OWF that invokes the one-way function *only once*. This sounds like a great improvement over [14] but, unfortunately, we use [14] as part of our construction. The idea is that if OWFs exist, then we can use [14] to obtain a PRG that is secure in the real world, and then it will also be a weakly-BB construction of PRG from OWF (which makes zero oracle queries).

[7] Gennaro and Trevisan also show unconditionally that there can be no fully-BB sublinear construction and, using our results in Section 3.2, we get an unconditional result for semi-BB constructions.

On the other hand, if OWFs do not exist, then we describe a weakly-BB construction.[8] How should we interpret such a result? It seems to say that we should not stop looking for more efficient constructions than the one in [14] and that, in this search, we may restrict ourselves to constructions that treat the one-way function as a black box.

1.5 Perspective

It should be stressed that not all reductions in the cryptographic literature are black box. Many of the examples are constructions that make use of the general construction of zero-knowledge proofs (and variants) for arbitrary NP languages [13], as the [13] protocol makes use of the code of the algorithm that verifies witnesses for the NP algorithm. For example, when using this result to construct identification schemes from any one-way function [5], the identification scheme makes use of the code of the one-way function and thus this is not a black-box reduction. In a similar fashion, there are a number of other results in cryptography that make non-black-box use of the starting primitive. Only recently, however, have we seen reductions making non-black-box use of *adversary* in the proof of security, in the exciting works of Barak [1, 2].

Given the fact that non-black-box reductions exist in the literature, one might wonder how to interpret black-box reductions and impossibility results. For this, it is useful to consider an analogy with the role of reductions in complexity theory. The first motivation for introducing polynomial-time reducibilities (e.g. Karp reductions and Cook reductions) was to relate the existence of polynomial-time algorithms for various problems: if problem A reduces to problem B, then $B \in P \Rightarrow A \in P$. Note that here the polynomial-time algorithm for B is used in a *black-box* manner. The constructed polynomial-time algorithm for A only uses the B-algorithm as a subroutine and its correctness doesn't make use of the fact that the B-algorithm is efficient.[9] One can envision non-black-box ways of proving implications of the form $B \in P \Rightarrow A \in P$, and there are examples in the literature (one is mentioned below). Still we find reductions to be an extremely useful concept. First, they provide a natural way of comparing the "complexity" of problems (even when we believe neither problem has a polynomial-time algorithm). For example, SAT trivially reduces to QBF_2 (quantified boolean formulae with two alternating quantifiers) and it is known that QBF_2 does not (Cook-)reduce to SAT unless the polynomial-time hierarchy collapses. Nevertheless, the implication $\mathrm{SAT} \in P \Rightarrow \mathrm{QBF}_2 \in P$ is known to hold, and indeed it (necessarily) makes non-black-box use of the polynomial-time algorithm for SAT. Still we interpret the lack of a Cook-reduction from QBF_2 to SAT saying that QBF_2 as a more "complex" problem than SAT. Second, results showing that certain reductions are unlikely to exist provide a guide for attempts to

[8] There are again some technical issues about infinitely many versus all input lengths.

[9] Note that the black-box use of the B-algorithm is particularly acute when B is a promise problem, since A must work for all oracles that are correct on inputs that satisfy the promise, even undecidable ones.

prove the corresponding implication. For example, it is known that for any NP-complete problem L, there is no *nonadaptive* reduction from deciding L in the worst case to deciding L in the average case (with respect to any samplable distribution) unless the polynomial-time hierarchy collapses [6, 4]. Thus, in future attempts to establish a worst-case/average-case equivalence for NP, it is natural to start by looking for *adaptive* reductions.

Both of these uses of reductions also seem relevant in cryptography. It is scientifically interesting to have notions for formalizing the idea that, say, public-key cryptography is a "more complex" primitive than private-key cryptography (even when we believe both to exist). And results on the non-existence of black-box reductions help guide attempts to establish new implications. For example, our results highlight the significance of making non-black-box use of the *adversary*, as in [1, 2], and suggest that it may enable us to overcome some previous barriers. We note that when using non-existence of reductions as a guide for future work, it is important to make the notions of reduction precise and carefully interpret their meaning. Indeed, these are some of the goals of the taxonomy and results presented in this paper.

2 Black-Box Constructions and Analyses

2.1 Cryptographic Primitives

In order to define the various notions of reduction between cryptographic primitives we first need to clarify what constitutes a primitive. The definition we use is quite general. Still, for the sake of readability, we do not state our definitions and results in the most general setting possible. In particular, our notion of efficiency will be that of probabilistic polynomial-time Turing machines (PPTMs) and we assume that all parties involved in the definition of a primitive (including the adversaries) are efficient. Therefore, our results are *stated* in a way that does not apply to non-uniform or information theoretic notions of security.

Definition 1. *A primitive $\mathcal{P}$ is a pair $\langle F_\mathcal{P}, R_\mathcal{P}\rangle$, where $F_\mathcal{P}$ is a set of functions $f : \{0,1\}^* \mapsto \{0,1\}^*$, and $R_\mathcal{P}$ is a relation over pairs $\langle f, M\rangle$ of a function $f \in F_\mathcal{P}$ and a machine M. The set $F_\mathcal{P}$ is required to contain at least one function which is computable by a PPTM.*

A function $f : \{0,1\}^* \mapsto \{0,1\}^*$ **implements** *$\mathcal{P}$ or is an implementation of $\mathcal{P}$ if $f \in F_\mathcal{P}$. An* **efficient implementation** *of $\mathcal{P}$ is an implementation of $\mathcal{P}$ which is computable by a PPTM. A machine M* **$\mathcal{P}$-breaks** *$f \in F_\mathcal{P}$ if $\langle f, M\rangle \in R_\mathcal{P}$. A* **secure implementation** *of $\mathcal{P}$ is an implementation of $\mathcal{P}$ such that no PPTM $\mathcal{P}$-breaks f. The primitive $\mathcal{P}$* **exists** *if there exists an efficient and secure implementation of $\mathcal{P}$.*

Let us elaborate on the semantics of the above definition. It is natural that an implementation of a primitive can be represented as a function $f : \{0,1\}^* \mapsto \{0,1\}^*$. For example, in the case of one-way function, f is simply the one-way function itself. In the case of encryption schemes, f represents three functions:

the key generation, the encryption and the decryption functions. In the case of key-agreement and protocols in general, f represents the message function (the function that determines the message a party should send given its inputs and the previous messages). The set $F_{\mathcal{P}}$ in the definition of a primitive $\mathcal{P}$ captures various structural requirements for an implementation of $\mathcal{P}$. For example, in the case of one-way permutations we require that an implementation f will be a length-preserving permutation. The set $F_{\mathcal{P}}$ also captures correctness requirements (when they are separated from the security of the primitive). For example, for encryption schemes, we require that the decryption of an encryption of a plaintext m will recover m. For key agreement, we require that the two parties output the same key. The structural and correctness requirements of a primitive are usually easy to obtain when we do not insist on security. Therefore, it is not very restrictive to require the set $F_{\mathcal{P}}$ to contain at least one efficiently computable function. Finally, the security requirement of a primitive is specified through the definition of breaking an implementation of this primitive. This is captured by the relation $R_{\mathcal{P}}$. For example, for one-way functions, we would define $\langle f, M\rangle \in R_{\mathcal{P}}$ if there is a polynomial p such that $\Pr[M(f(U_n)) \in f^{-1}(f(U_n))] > 1/p(n)$ for infinitely many n. Sometimes, we will need to work with "infinitely often" (io) analogues of primitives, where the security is only required to hold for infinitely many input lengths, i.e. to "break" the primitive, an adversary must succeed on all but finitely many input lengths. For example, if $\mathcal{P}$ is the primitive ioOWF, then we would define $\langle f, M\rangle \in R_{\mathcal{P}}$ if there is a polynomial p such that $\Pr[M(f(U_n)) \in f^{-1}(f(U_n))] > 1/p(n)$ for all but finitely many n.

We will also need to define the existence of a primitive relative to an oracle.

Definition 2. *A primitive $\mathcal{P}$* **exists relative to an oracle** *Π if there exists an implementation f of $\mathcal{P}$ which is computable by a probabilistic polynomial time oracle machine with access to Π and such that no probabilistic polynomial time oracle machine with access to Π $\mathcal{P}$-breaks f.*

2.2 Notions of Reducibility

A reduction from a primitive $\mathcal{P}$ to a primitive $\mathcal{Q}$ means that the existence of $\mathcal{Q}$ implies the existence of $\mathcal{P}$. In other words, it means that either $\mathcal{P}$ exists or $\mathcal{Q}$ does not exist. Reductions in the literature usually entail much more than that. For example, a reduction from $\mathcal{P}$ to $\mathcal{Q}$ usually gives a constructive way of obtaining a secure and efficient implementation of $\mathcal{P}$ from one of $\mathcal{Q}$. We now define various such types of more restricted and structured reductions. For comparison we refer to an arbitrary reduction as a **free reduction**.

The most restricted form of reduction considered in this paper is what we call a *fully black-box (BB) reduction*, where the construction and analysis (showing that the construction produces a secure implementation of $\mathcal{P}$ given a secure implementation of $\mathcal{Q}$) are both BB. Most, but not all, reductions in the literature are fully BB.

Definition 3. *There exists a* **fully-BB reduction** *from a primitive* $\mathcal{P} = \langle F_{\mathcal{P}}, R_{\mathcal{P}} \rangle$ *to a primitive* $\mathcal{Q} = \langle F_{\mathcal{Q}}, R_{\mathcal{Q}} \rangle$*, if there exist probabilistic polynomial-time oracle machines* G *and* S *such that:*

Correctness *For every implementation* $f \in F_{\mathcal{Q}}$ *we have that* $G^f \in F_{\mathcal{P}}$*.*
Security *For every implementation* $f \in F_{\mathcal{Q}}$ *and every machine* A*, if* A $\mathcal{P}$*-breaks* G^f *then* $S^{A,f}$ $\mathcal{Q}$*-breaks* f*.*

The next, less restricted, notion of reduction is a reduction that works even if *all parties* get an oracle access to a, possibly *inefficient*, implementation of $\mathcal{Q}$.

Definition 4. *There exists a* **semi-BB reduction** *from a primitive* $\mathcal{P} = \langle F_{\mathcal{P}}, R_{\mathcal{P}} \rangle$ *to a primitive* $\mathcal{Q} = \langle F_{\mathcal{Q}}, R_{\mathcal{Q}} \rangle$ *if there exists a probabilistic polynomial-time oracle machine* G *such that:*

Correctness *For every implementation* $f \in F_{\mathcal{Q}}$ *we have that* $G^f \in F_{\mathcal{P}}$*.*
Security *For every implementation* $f \in F_{\mathcal{Q}}$*, if there exists a probabilistic polynomial-time oracle machine* A *such that* A^f $\mathcal{P}$*-breaks* G^f*, then there exists a probabilistic polynomial-time oracle machine* S *such that* S^f $\mathcal{Q}$*-breaks* f*.*

It is tempting to view a semi-BB reduction as a BB-construction with an arbitrary analysis, since only f is treated as a black box. However, as we try to argue in Section 3, the analysis in semi-BB reduction is still very much black box. In essence, this is due to the oracle access that A gets to (the computationally unbounded) f. Since f may be the heart of the adversary A^f that breaks $\mathcal{P}$, the access S has to this adversary is in large part black box. Following is our attempt to formalize what we view as a BB construction with arbitrary analysis.

Definition 5. *There exists a* **weakly-BB reduction** *from a primitive* $\mathcal{P} = \langle F_{\mathcal{P}}, R_{\mathcal{P}} \rangle$ *to a primitive* $\mathcal{Q} = \langle F_{\mathcal{Q}}, R_{\mathcal{Q}} \rangle$ *if there exists a probabilistic polynomial-time oracle machine* G *such that:*

Correctness *For every implementation* $f \in F_{\mathcal{Q}}$ *we have that* $G^f \in F_{\mathcal{P}}$*.*
Security *For every implementation* $f \in F_{\mathcal{Q}}$*, if there exists a PPTM* A *that* $\mathcal{P}$*-breaks* G^f*, then there exists a probabilistic polynomial-time oracle machine* S *such that* S^f $\mathcal{Q}$*-breaks* f*.*

Remark 1. Arguably, a definition that would capture the phrase "a BB construction with arbitrary analysis" even better is one where S is also denied access to f. For the sake of this discussion, let us refer to such reductions as weakly $'$-BB. One problematic aspect of weakly $'$-BB reductions is that not only such reductions are more restricted that weakly-BB they even seem incomparable to fully-BB reductions. In particular, for many fundamental BB-reductions known in cryptography, it is not clear if the corresponding implications can also be proven via weakly $'$-BB reductions. In Section 4, we show that in many settings ruling out the existence of weakly-BB reductions is essentially as hard as ruling out the existence of any kind of reduction (i.e. a free reduction). Thus, these results also apply to the more restricted weakly $'$-BB reductions.

Fig. 1. Simple relations between notions of reduction. An arrow goes from a more restricted form of reduction to a less restricted one.

Related to BB-reductions are relativizing reduction which turn out very useful in the context of BB separations.

Definition 6. *There exists a* **relativizing reduction** *from a primitive* $\mathcal{P} = \langle F_{\mathcal{P}}, R_{\mathcal{P}} \rangle$ *to a primitive* $\mathcal{Q} = \langle F_{\mathcal{Q}}, R_{\mathcal{Q}} \rangle$*, if for every oracle* Π*, if* $\mathcal{Q}$ *exists relative to* Π *then so does* $\mathcal{P}$*.*

Finally, we consider two additional notions of reductions that are obtained from semi and weak BB reductions by a switch of quantifiers. Previously we asked for a "universal" procedure G that reduces all secure implementations f of $\mathcal{Q}$ to secure implementations G^f of $\mathcal{P}$. But this may not be necessary if we are only trying to show that $\mathcal{P}$ reduces to $\mathcal{Q}$. In the following definitions we are satisfied with the existence of a (possibly different) G for every f (hence the name $\forall\exists$).

Definition 7. *There exists a* $\forall\exists$**semi-BB reduction** *from a primitive* $\mathcal{P} = \langle F_{\mathcal{P}}, R_{\mathcal{P}} \rangle$ *to a primitive* $\mathcal{Q} = \langle F_{\mathcal{Q}}, R_{\mathcal{Q}} \rangle$*, if for every implementation* $f \in F_{\mathcal{Q}}$ *there exist a probabilistic polynomial time oracle machine* G *such that:*

Correctness $G^f \in F_{\mathcal{P}}$.

Security *If there exists a probabilistic polynomial-time oracle machine* A *such that* A^f $\mathcal{P}$*-breaks* G^f*, then there exist a probabilistic polynomial time oracle machines* S *such that* S^f $\mathcal{Q}$*-breaks* f*.*

Definition 8. *There exists a* $\forall\exists$**weakly-BB reduction** *from a primitive* $\mathcal{P} = \langle F_{\mathcal{P}}, R_{\mathcal{P}} \rangle$ *to a primitive* $\mathcal{Q} = \langle F_{\mathcal{Q}}, R_{\mathcal{Q}} \rangle$*, if for every implementation* $f \in F_{\mathcal{Q}}$ *there exist a probabilistic polynomial time oracle machine* G *such that:*

Correctness $G^f \in F_{\mathcal{P}}$.

Security *If there exists a PPTM* A *that* $\mathcal{P}$*-breaks* G^f*, then there exists a probabilistic polynomial-time oracle machine* S *such that* S^f $\mathcal{Q}$*-breaks* f*.*

Some simple relations between the various notions of reductions are given by the following lemma (and are illustrated in Figure 1).

Lemma 1. *For any two primitives $\mathcal{P}$ and $\mathcal{Q}$, we have the following:*

1. *If there exists a fully-BB reduction from $\mathcal{P}$ to $\mathcal{Q}$ then there exists a semi-BB reduction from $\mathcal{P}$ to $\mathcal{Q}$ as well.*
2. *If there exists a semi-BB reduction from $\mathcal{P}$ to $\mathcal{Q}$ then there exists a weakly-BB reduction from $\mathcal{P}$ to $\mathcal{Q}$ as well.*
3. *If there exists a semi-BB reduction from $\mathcal{P}$ to $\mathcal{Q}$ then there exists a $\forall\exists$semi-BB reduction from $\mathcal{P}$ to $\mathcal{Q}$ as well.*
4. *If there exists a weakly-BB reduction from $\mathcal{P}$ to $\mathcal{Q}$ then there exists a $\forall\exists$weakly-BB reduction from $\mathcal{P}$ to $\mathcal{Q}$ as well.*
5. *If there exists a $\forall\exists$semi-BB reduction from $\mathcal{P}$ to $\mathcal{Q}$ then there exists a $\forall\exists$weakly-BB reduction from $\mathcal{P}$ to $\mathcal{Q}$ as well.*
6. *If there exists a $\forall\exists$weakly-BB reduction from $\mathcal{P}$ to $\mathcal{Q}$ then there exists a free reduction from $\mathcal{P}$ to $\mathcal{Q}$ as well.*
7. *If there exists a fully-BB reduction from $\mathcal{P}$ to $\mathcal{Q}$ then there exists a relativizing reduction from $\mathcal{P}$ to $\mathcal{Q}$ as well.*
8. *If there exists a relativizing reduction from $\mathcal{P}$ to $\mathcal{Q}$ then there exists a $\forall\exists$semi-BB reduction from $\mathcal{P}$ to $\mathcal{Q}$ as well.*

Proof. All relations follow quite easily from the definitions. We concentrate on the two relations connecting relativizing reductions to BB reductions.

For Item 7, assume that there exists a fully-BB reduction from a primitive $\mathcal{P} = \langle F_{\mathcal{P}}, R_{\mathcal{P}} \rangle$ to a primitive $\mathcal{Q} = \langle F_{\mathcal{Q}}, R_{\mathcal{Q}} \rangle$, given by the probabilistic oracle machines G and S (see the definition of a fully-BB reduction). Let Π be an arbitrary oracle and further assume that $\mathcal{Q}$ exists relative to Π. We need to prove that $\mathcal{P}$ also exists relative to Π.

Let $f \in F_{\mathcal{Q}}$ be (an implementation of $\mathcal{Q}$) such that no probabilistic polynomial time oracle machine with access to Π $\mathcal{Q}$-breaks f and such that f is computable by M^{Π} where M is some probabilistic polynomial time oracle machine. By definition, $G^f \in F_{\mathcal{P}}$. Its also immediate that $G^f \equiv G^{M^{\Pi}}$ is also computable by a probabilistic polynomial time oracle machine with access to Π. Assume for the sake of contradiction that there exists a probabilistic polynomial time oracle machine A such that A^{Π} $\mathcal{P}$-breaks G^f. Then, by definition $S^{A^{\Pi},f}$ $\mathcal{Q}$-breaks f. Since $S^{A^{\Pi},f} \equiv S^{A^{\Pi},M^{\Pi}}$ is computable by a probabilistic polynomial time oracle machine with access to Π, we arrive at a contradiction and can conclude Item 7.

For Item 8 let us assume that there is a relativizing reduction from $\mathcal{P}$ to $\mathcal{Q}$ and prove that there also exists a $\forall\exists$semi-BB reduction from $\mathcal{P}$ to $\mathcal{Q}$. Let f be any implementation of $\mathcal{Q}$. One possibility is that there exists a probabilistic polynomial-time oracle machine S such that S^f $\mathcal{Q}$-breaks f. Define a probabilistic polynomial-time oracle machine G that ignores its oracle and computes some (not necessarily secure) implementation of $\mathcal{P}$. Its follows immediately that G and S satisfy the conditions in the definition of $\forall\exists$semi-BB reduction. We can therefore assume that S as above does not exist. This means that relative to the oracle f the primitive $\mathcal{Q}$ exists (its efficient and secure implementation is f itself). By the definition of relativizing reduction, the primitive $\mathcal{Q}$ exists relative to f as well. Therefore, there exists a probabilistic polynomial time oracle machine G such that $G^f \in F_{\mathcal{P}}$ and there is no probabilistic polynomial time oracle machine A such that A^f $\mathcal{P}$-breaks G^f. We can therefore conclude that there exists a $\forall\exists$semi-BB reduction from $\mathcal{P}$ to $\mathcal{Q}$ which completes the proof of the item.

3 Semi-BB versus Relativization

The study of BB separations in cryptography started with the seminal work of Impagliazzo and Rudich [18]. Previously it was known that the existence of many cryptographic primitives, such as various private-key primitives and digital signatures, reduces to the existence of one-way functions (OWF), which in turn are essentially necessary for all computational aspects of security in Cryptography. Other primitives however such as key-agreement (KA), and thus also various fundamental primitives that imply KA, resisted attempts to be reduced to OWF. Noting that almost all reductions in cryptography are black-box, [18] turned to showing that such reductions are simply not sufficiently powerful to reduce KA to OWF or even to one way permutations (OWP).

Theorem 1. *[18] There is no relativizing reduction from KA to OWP.*

An immediate consequence of Theorem 1 is that there is no fully-BB reduction from KA to OWP. At the core of the proof of Theorem 1 stands a lemma which states that, relative to a random oracle (which is in some sense a "perfect OWP"), there are no KA unless $P \neq NP$. In particular, constructing KA in the random-oracle model is at least as hard as proving $P \neq NP$. In addition, [18] pointed that this lemma seems to "rule out" even less restrictive forms of BB reductions from KA to OWP. Using the taxonomy of this paper, we can state the results of [18] with respect to BB reductions as follows.

Theorem 2. *(implicit in [18]) There is no fully-BB reduction from key-agreement to one-way permutations. Furthermore, there is no $\forall\exists$semi-BB reduction from KA to OWP unless $P \neq NP$.*

In this section we prove an unconditional version of Theorem 2. We generalize this by showing that "usually" $\forall\exists$semi-BB reductions are equivalent to relativizing reductions. This implies unconditional proofs of various results that were previously only known to hold conditionally. Finally, based on the new equivalence between reduction types, we reinterpret the notion of semi-BB reductions.

3.1 Impagliazzo-Rudich Revisited

Based on Theorem 1 and using an "embedding technique" due to Simon [25], we are able to strengthen Theorem 2 as follows.

Theorem 3. *There is no $\forall\exists$semi-BB reduction from KA to OWP.*

Proof. Theorem 1 implies that there exists an oracle $\Pi : \{0,1\}^* \mapsto \{0,1\}$ such that relative to Π, OWP exists and KA does not. Let f' be the secure and efficient OWP which exist relative to Π. We define a permutation f such that (1) f is computable by a probabilistic polynomial-time oracle machine with access to Π, (2) f is one-way relative to Π, and (3) There exists a probabilistic polynomial-time oracle machine with access to f that evaluates Π. Let us first assume that such an f exist and see how it implies the theorem.

Properties (1) and (2) of f imply that f is one-way relative to itself (since an oracle machine that OWP-breaks f relative to f can be efficiently simulated relative to Π). Properties (1) and (3) of f imply that there is no KA relative to f. This is because an efficient implementation of KA relative to f is also an efficient implementation of KA relative to Π which implies that it can be broken relative to Π and thus also relative to f. Now assume for the sake of contradiction that there exist a ∀∃semi-BB reduction from KA to OWP. Let G be the probabilistic polynomial-time oracle machine which corresponds to f as guaranteed by the definition of ∀∃semi-BB reduction. It is now straight forward from the definition of G that G^f is a secure KA relative to f which contradicts our assumption that there exist a ∀∃semi-BB reduction from KA to OWP.

It remains to define f with the desired properties. Intuitively Π is "embedded" into a small part of f and on the rest of the inputs, f evaluates f'. On a $2n+1$-bit long input $\langle r, x, \sigma \rangle$ where r and x are n-bit long each and σ is a bit, the function f is defined as follows: If r is the all-zero string then $f(r, x, \sigma) = r, x, \Pi(x) \oplus \sigma$. Otherwise, $f(r, x, \sigma) = r, f'(x), \sigma$. (The definition can be naturally extended to even-length inputs.) That f is a permutation follows trivially from f' being a permutation. Property (2) (the one-wayness of f relative to Π) is also easy as on all but a negligible fraction of its inputs (those with r being the all-zero string), inverting f on a random input is equivalent to inverting f' on a random input. Finally, properties (1) and (3) follows immediately from the definition. □

3.2 The General Condition for Equivalence

The equivalence between the existence of a relativizing reduction and a ∀∃semi-BB reduction, is not limited to the reduction from KA to OWP. In fact, essentially the same argument was used by Simon [25], in regard of the reduction of collision-resistant hash functions to OWP. In general, the two notions of reduction are equivalent for showing a reduction from a primitive $\mathcal{P}$ to a primitive $\mathcal{Q}$, if it is possible to "embed" an arbitrary oracle into $\mathcal{Q}$ as in the proof of Theorem 3.

Definition 9. *We say that a primitive $\mathcal{Q} = \langle F_{\mathcal{Q}}, R_{\mathcal{Q}} \rangle$* allows embedding *if for any $f' \in F_{\mathcal{Q}}$ and any oracle $\Pi : \{0,1\}^* \mapsto \{0,1\}$ there exist $f \in F_{\mathcal{Q}}$ such that the following hold:*

1. *f is computable by a probabilistic polynomial-time oracle machine with access to Π,.*

Fig. 2. In addition to the simple relations already shown in Figure 1, the dashed arrow indicates that "usually" relativizing reduction are equivalent to $\forall\exists$semi-BB reduction.

2. *If there exists a probabilistic polynomial-time oracle machine M that $\mathcal{Q}$-breaks f then there exists such M that $\mathcal{Q}$-breaks f'.*
3. *There exists a probabilistic polynomial-time oracle machine with access to f that evaluates Π.*

The following equivalence is proven in exactly the same way as Theorem 3.

Theorem 4. *Let $\mathcal{P} = \langle F_{\mathcal{P}}, R_{\mathcal{P}} \rangle$ be any primitive and $\mathcal{Q} = \langle F_{\mathcal{Q}}, R_{\mathcal{Q}} \rangle$ be any primitive that allows embedding. Then there exist a relativizing reduction from $\mathcal{P}$ to $\mathcal{Q}$ if and only if there exist a $\forall\exists$semi-BB reduction from $\mathcal{P}$ to $\mathcal{Q}$.*

We note that its hard to think of a natural primitive that does not allow embedding. In fact, the case of OWP is relatively difficult compared to others we could think of (because of the need to preserve the permutation). Therefore, we can informally say that "usually" the above equivalence holds (see Figure 2 for an updated picture which takes this "equivalence" into account). The embedding technique allows us to prove that $\forall\exists$semi-BB reductions are unconditionally impossible in all case where $\forall\exists$semi-BB reductions were previously only conditionally ruled out. Two examples are [24] on reducing the number of rounds in KA and [9] on the relationships among KA, oblivious transfer, public-key encryptions, and trapdoor functions and permutations. In fact, this also holds for the results of [8, 7] regarding the *efficiency* of known constructions. In this setting however, it is important to take into account the efficiency of the embedding technique itself. Usually however the embedding is extremely efficient. For example, in the definition of f above evaluating it requires a single oracle query (either to f' or to Π) and similarly evaluating Π requires a single oracle call to f.

3.3 Discussion

It is typical to view semi-BB reductions and certainly $\forall\exists$semi-BB as a BB-construction with arbitrary analysis. However, we feel that the equivalence to relativizing reductions and more particularly the embedding technique implies

Fig. 3. In addition to the picture given by Figure 2, the dotted arrow indicates that in some interesting cases weakly-BB reductions are equivalent to free (arbitrary) reductions.

that the analysis in semi-BB reduction is still very much black box. Recall that in a semi-BB reduction from $\mathcal{P}$ to $\mathcal{Q}$ we only consider polynomial time machines A such A^f $\mathcal{P}$-breaks G^f and the requirement is that if such a machine A exists then there also exists an efficient S such that S^f $\mathcal{Q}$-breaks f. This looks less BB than the analysis in fully-BB reductions since S does not get oracle access to A but rather only to f and since we only consider efficient machines A. The reason that this analysis is still very much BB is that the adversary for $\mathcal{P}$ is A^f (which may be very inefficient) rather than f. Meaning in particular that the reduction does not have access to a small decryption of this adversary (yet alone a small circuit that evaluates it). What the embedding technique contributes to this argument is that usually f may be viewed as the major part of the adversary A^f. In fact, in many cases we can assume that A is embedded into f. Therefore, getting oracle access may effectively give S access to A^f.

4 Weakly-BB Versus Arbitrary Reductions

In this section we show various settings for which weakly-BB reductions exist iff free (arbitrary) reductions exist (this is illustrated in Figure 3). In other words, in some settings weakly-BB are as powerful as free reductions. We could therefore concentrate on finding such reductions *which treat the primitive as a black box.* These results also indicate that it is unlikely we could strengthen some previous BB separations that previously ruled out semi-BB reductions so that they also rule out weakly-BB reductions in the same settings.

4.1 Distributional One-Way Functions

Part of each proof of equivalence between weakly-BB reductions and free reductions given in this section is a construction which works under the assumption that one-way functions do not exist. As shown by Impagliazzo and Luby [16], the non-existence of one-way functions imply the non-existence of *distributional* one-way functions, a fact that we will use in our constructions.

Lemma 2 (Impagliazzo and Luby). *Suppose that one-way functions do not exist, and let g be a (possibly probabilistic) polynomial time computable function and $q(n)$ be a polynomial. Then there is a polynomial time probabilistic algorithm S such that on, infinitely many input lengths n, the distributions $(x, g(x))$ and $(S(g(x)), g(x))$ have statistical distance at most $1/q(n)$, where x is uniform in $\{0,1\}^n$.*

In other words, the result says that if it is possible to compute preimages of polynomial time computable functions, then it is also possible to sample almost uniformly from the set of preimages.

4.2 Weakly-BB Reductions from KA to OWF

We now show that if the statement "the existence of OWF implies the existence of ioKA" is true then it can be proved via a weakly-BB construction of KA based on OWF. We note that this means that it is unlikely that we could rule out a weakly-BB reduction from ioKA to OWF whereas [18] (implicitly) rule out such semi-BB reductions. The equivalence between free reductions and weakly-BB reductions in this context follows from the next two lemmas.

Lemma 3. *Suppose that ioKA exists. Then there is a weakly-BB reduction from ioKA to OWF.*

Proof. The efficient oracle machine G needed by the definition of weakly-BB reductions simply ignores the oracle f and evaluates from scratch the ioKA which independently exist. The reduction is secure as there is no PPTM A that ioKA-breaks G^f. □

Lemma 4. *Suppose that OWF do not exist. Then there is a weakly-BB reduction from ioKA to OWF.*

Proof (Proof sketch). Consider the following construction: given security parameter n and oracle f

- Alice picks at random $x, r \in \{0,1\}^n$, and sends x, r to Bob.
- Alice and Bob agree on the bit $f(x) \cdot r$.

The protocol does not make much sense in the "real world," but the reader should be reminded that the protocol is only meant to work in case OWFs do not exist, a case in which no KA protocol can exist in the real world.

To prove the Lemma, we will show that if f is a black-box one-way function, then the protocol cannot be broken by an efficient adversary. Intuitively, the reason is that if f is a black-box one-way function, and OWFs do not exist, then f must be a function that cannot be computed efficiently. Using Goldreich-Levin, we can then infer that $f(x) \cdot r$ is hard to predict.

More precisely, let E be a polynomial-time adversary that has a noticeable advantage in guessing the generated key $f(x) \cdot r$ on all but finitely many input lengths. Using Goldreich-Levin, we can then find a probabilistic polynomial-time algorithm M that computes f correctly with noticeable probability, i.e. $\Pr[M(U_n) = f(U_n)] \geq 1/p(n)$ for some polynomial p. Since the nonexistence of one-way functions implies the nonexistence of distributional one-way functions [16], there is a probabilistic polynomial-time algorithm I that generates almost-uniformly distributed preimages under M. That is, $(U_n, M(U_n))$ and $(I(M(U_n)), M(U_n))$ have statistical difference at most $1/q(n)$ for any desired polynomial q and infinitely many n. We will now argue that, by choosing $q = 4p$, it follows that I inverts f with nonnegligible probability.

It is not hard to show that $\Pr_{y \in M(U_n)}[I(y) \in f^{-1}(y)]$ is non-negligible. However, this in itself does not imply that $\Pr_{y \in f(U_n)}[I(y) \in f^{-1}(y)]$ is non-negligible as well. For that we need to first argue about the likelihood of obtaining a particular output y under f and under M. Define the set O of outputs $\{y : \Pr[M(U_n) = y] \leq 2p(n) \Pr[f(U_n) = y]\}$. Since $\Pr[M(U_n) = f(U_n)] \geq 1/p(n)$ it follows that $\Pr[M(U_n) = f(U_n) \in O] \geq 1/2p(n)$, as otherwise the probability mass that $M(U_n)$ gives to strings outside of O is strictly larger than 1 which is of course impossible.

Consider the statistical test T defined as follows: $T(x, y) = 1$ iff. $f(x) = M(x) = y \in O$. We now have that $\Pr[T(U_n, M(U_n)) = 1] = \Pr[M(U_n) = f(U_n) \in O] \geq 1/2p(n)$. Since (for infinitely many values of n), $(U_n, M(U_n))$ and $(I(M(U_n)), M(U_n))$ have statistical difference at most $1/4p(n)$ we can conclude that $\Pr[T((I(M(U_n)), M(U_n))) = 1] \geq 1/4p(n)$. Equivalently,

$$\Pr_{y \in M(U_n)}[I(y) \in \left(f^{-1}(y) \cap M^{-1}(y)\right) \text{ and } y \in O] \geq 1/4p(n).$$

By the definition of O we get that for infinitely many values of n,

$$\begin{aligned}
&1/4p(n) \\
&\leq \Pr_{y \in M(U_n)}[I(y) \in \left(f^{-1}(y) \cap M^{-1}(y)\right) \text{ and } y \in O] \\
&= \sum_{y \in O} \Pr[M(U_n) = y] \Pr[I(y) \in \left(f^{-1}(y) \cap M^{-1}(y)\right)] \\
&\leq 2p(n) \sum_{y \in O} \Pr[f(U_n) = y] \Pr[I(y) \in \left(f^{-1}(y) \cap M^{-1}(y)\right)] \\
&= 2p(n) \Pr_{y \in f(U_n)}[I(y) \in \left(f^{-1}(y) \cap M^{-1}(y)\right) \text{ and } y \in O] \\
&\leq 2p(n) \Pr_{y \in f(U_n)}[I(y) \in f^{-1}(y)]
\end{aligned}$$

We conclude that, for infinitely many values of n, $\Pr_{y \in f(U_n)}[I(y) \in f^{-1}(y)] \geq 1/8p^2(n)$.

□

From the above two lemmas, we conclude that weakly-BB reductions are as powerful as free reductions for this problem:

Theorem 5. *There is weakly-BB reduction from ioKA to OWF if and only if there is a free reduction from ioKA to OWF.*

Next we give a similar result for reducing trapdoor permutations to OWF.

Theorem 6. *There is a weakly-BB reduction of io-trapdoor permutations to one-way functions if and only if there is a free reduction of io-trapdoor permutations to one-way functions.*

Proof (Proof sketch). The proof has the same structure as that of Theorem 5. The case that io-trapdoor permutations exist is immediate. For the case that one-way functions don't exist, we consider the following construction of a family of trapdoor permutations from a black-box OWF f. On security parameter n and oracle access to f: the key generation algorithm outputs the empty string for both the index of the function and the trapdoor information. The trapdoor permutation g^f is defined as $g^f(x, r) = (x, r \oplus f(x))$. It is clear that this defines a permutation. The inversion algorithm uses oracle access to f to invert g^f.

The proof that g^f is hard to invert by PPT algorithms uses the fact that one-way functions don't exist, and is similar to the proof of Lemma 4. As in that lemma, this construction does not make much sense in the "real world," but recall that the protocol is only meant to work in case OWFs do not exist, a case in which no trapdoor permutation can exist in the real world. □

4.3 A Weakly-BB Construction More Efficient than HILL

As mentioned in the introduction, a long-standing open question is to reduce or explain the inefficiency of the construction of pseudorandom generators from general one-way functions [14]. The construction of [14] is a fully black-box reduction that seems to require polynomially many queries to the one-way function even to obtain a pseudorandom generator that stretches by one bit (in contrast to the construction of pseudorandom generators from one-way permutations [3, 26, 11], which requires only one query to stretch by one bit).

Theorem 7. *There is weakly-BB construction of ioPRGs from OWFs that makes only one query.*

Thus to show that the inefficiency of [14] is inherent, one must consider more constrained reductions than weakly-BB reductions. In particular, one cannot directly use the approach of [8], which gives lower bounds on the efficiency of weakly-BB reductions. Alternatively, this theorem says that, in attempting to improve the efficiency of [14], there is no loss in treating the OWF as a black box.

Lemma 5. *Suppose that OWF exist. Then there is a weakly-BB construction of ioPRG based on OWF, where the construction makes zero oracle queries.*

Lemma 6. *Suppose that OWF do not exist. Then there is a weakly-BB construction of ioPRG based on OWF, where the construction makes one oracle queries.*

Proof (Proof sketch). The construction is $G^f(x, r) = (x, r, f(x) \cdot r)$, for $|x| = |r|$. The proof that this is a weakly BB construction is analogous to the proof of Lemma 4.

Acknowledgments

We thank Cynthia Dwork, Russell Impagliazzo, Tal Malkin, Moni Naor, and Steven Rudich for helpful discussions.

References

1. B. Barak. How to go beyond the black-box simulation barrier. In *Proc. of 42nd IEEE Symposium on Foundations of Computer Science (FOCS'01)*, pages 106–115, 2001.
2. Boaz Barak. Constant-round coin-tossing with a man in the middle or realizing the shared random string model. *Cryptology ePrint Archive*, 2002.
3. Manuel Blum and Silvio Micali. How to generate cryptographically strong sequences of pseudorandom bits. *SIAM J. Comput.*, 13(4):850–864, 1984.
4. Andrej Bogdanov and Luca Trevisan. On worst-case to average-case reductions for NP problems. In *Proc. of 44th IEEE Symposium on Foundations of Computer Science (FOCS'03)*, pages 308–317, 2003.
5. Uriel Feige, Amos Fiat, and Adi Shamir. Zero-knowledge proofs of identity. *Journal of Cryptology*, 1(2):77–94, 1988.
6. Joan Feigenbaum and Lance Fortnow. Random-self-reducibility of complete sets. *SIAM J. Comput.*, 22(5):994–1005, 1993.
7. R. Gennaro, Y. Gertner, and J. Katz. Lower bounds on the efficiency of encryption and digital signature schemes. In *STOC 2003*, pages 417–425, 2003.
8. R. Gennaro and L. Trevisan. Lower bounds on the efficiency of generic cryptographic constructions. In *Proceedings of the IEEE Symposium on Foundations of Computer Science*, 2000.
9. Y. Gertner, S. Kannan, T. Malkin, O. Reingold, and M. Viswanathan. The relationship between public key encryption and oblivious transfer. In *Proceedings of the IEEE Symposium on Foundations of Computer Science*, 2000.
10. Yael Gertner, Tal Malkin, and Omer Reingold. Lower bounds on the efficiency of generic cryptographic constructions. In *Proceedings of the IEEE Symposium on Foundations of Computer Science*, pages 126–135, 2001.
11. O. Goldreich and L. Levin. A hard predicate for all one-way functions. In *Proceedings of the ACM Symposium on the Theory of Computing*, 1989.
12. Oded Goldreich, Shafi Goldwasser, and Silvio Micali. How to construct random functions. *Journal of the Association for Computing Machinery*, 33(4):792–807, 1986.
13. Oded Goldreich, Silvio Micali, and Avi Wigderson. Proofs that yield nothing but their validity or all languages in NP have zero-knowledge proof systems. *Journal of the ACM*, 38(3):691–729, July 1991.
14. Johan Håstad, Russell Impagliazzo, Leonid A. Levin, and Michael Luby. A pseudorandom generator from any one-way function. *SIAM Journal on Computing*, 28(4):1364–1396, 1999.
15. R. Impagliazzo and M. Luby. One-way functions are essencial for complexity-based cryptography. In *Proceedings of the 30th Symposium on Foundations of Computer Science, IEEE*, 1989.
16. R. Impagliazzo and M. Luby. One-way functions are essential for complexity based cryptography. In *Proc. of 30th IEEE Symp. on Foun. of Comp. Sci. (FOCS'89)*, pages 230–235, 1989.

17. R. Impagliazzo and S. Rudich. private communication.
18. R. Impagliazzo and S. Rudich. Limits on the provable consequences of one-way permutations. In *Proceedings of the 21st ACM Symposium on the Theory of Computing*, 1989.
19. Jeff Kahn, Michael Saks, and Cliff Smyth. A dual version of reimer's inequality and a proof of rudich's conjecture. In *Proceedings of the 15th Annual IEEE Conference on Computational Complexity*, 2000.
20. Jeong Han Kim, Danial Simon, and Prasad Tetali. Limits on the efficiency of one-way permuation-based hash functions. In *Proceedings of the IEEE Symposium on Foundations of Computer Science*, 1999.
21. Moni Naor. Bit commitment using pseudorandomness. *Journal of Cryptology*, 4(2):151–158, 1991.
22. Rafail Ostrovsky and Avi Wigderson. One-way functions are essential for non-trivial zero-knowledge. Technical Report TR-93-073, International Computer Science Institute, Berkeley, CA, November 1993. Preliminary version in Proc. 2nd Israeli Symp. on Theory of Computing and Systems, 1993, pp. 3–17.
23. John Rompel. One-way functions are necessary and sufficient for secure signatures. In *Proceedings of the Twenty Second Annual ACM Symposium on Theory of Computing*, pages 387–394, 1990.
24. S. Rudich. The use of interaction in public cryptosysytems. In *Advances in Cryptology – Crypto '91 Proceedings*, pages 242–251, 1991.
25. Dan Simon. Finding collisions on a one-way street: Can secure hash functions be based on general assumptions. In *Proceedings of EUROCRYPT*, 1998.
26. A. Yao. Theory and applications of trapdoor functions. In *Proceedings of the 23rd Symposium on Foundations of Computer Science, IEEE*, 1982.