Universal Reductions: Reductions Relative to Stateful Oracles

Benjamin Chan$^{1[0000-0003-1406-9845]}$, Cody Freitag$^{1[0000-0002-6307-204X]}$, and Rafael Pass^{1,2}

[1] Cornell Tech, New York, USA
{byc,cfreitag,rafael}@cs.cornell.edu
[2] Tel-Aviv University, Tel Aviv-Yafo, Israel

Abstract. We define a framework for analyzing the security of cryptographic protocols that makes minimal assumptions about what a "realistic model of computation is". In particular, whereas classical models assume that the attacker is a (perhaps non-uniform) probabilistic polynomial-time algorithm, and more recent definitional approaches also consider quantum polynomial-time algorithms, we consider an approach that is more agnostic to what computational model is physically realizable.

Our notion of *universal reductions* models attackers as PPT algorithms having access to some arbitrary unbounded *stateful* Nature that cannot be rewound or restarted when queried multiple times. We also consider a more relaxed notion of *universal reductions w.r.t. time-evolving, k-window, Natures* that makes restrictions on Nature—roughly speaking, Nature's behavior may depend on number of messages it has received and the content of the last $k(\lambda)$-messages (but not on "older" messages). We present both impossibility results and general feasibility results for our notions, indicating to what extent the extended Church-Turing hypotheses are needed for a well-founded theory of Cryptography.

1 Introduction

Modern Cryptography relies on the principle that cryptographic schemes are proven secure based on mathematically precise assumptions; these can be general—such as the existence of one-way functions—or specific—such as the hardness of factoring products of large primes. The security proof is a *reduction* that "transforms" any attacker A of a scheme (e.g., a pseudorandom generator) into an attacker A' that breaks the underlying assumption (e.g., inverts an alleged one-way function). More formally, cryptographic security of a single primitive or assumption is often defined as an *interactive game* (a.k.a. a *security game*) between a *challenger* C and an *adversary* A. C sends a random challenge (e.g. a product of two large primes) to A, who tries to respond in such a way—potentially over many rounds—to make the challenger accept (e.g. by sending the individual factors). The game is determined by the challenger C and the primitive is said to be secure if no "realistic" adversary can cause the challenger to accept with some specified probability. (In the sequel, we will abuse notation and often identify the security game simply by the challenger C.) A reduction R from a game with challenger C (i.e., a security game C) to one with challenger C' provides a way to use a successful adversary A in the game C to construct a successful adversary A' in the game C'. This study has been extremely successful, and during the past four decades many cryptographic tasks have been put under rigorous treatment and numerous constructions realizing these tasks have been proposed under a number of well-studied complexity-theoretic hardness assumptions.

In this paper, we revisit what it means to transform the alleged attacker A for the scheme into an attacker A' for the underlying assumption. In particular, the standard cryptographic treatment explicitly assumes that the attacker A is a (perhaps non-uniform) probabilistic polynomial-time (PPT) Turing machine. Thus, when using the scheme in the "real-world", the security proof is only meaningful if this model of the attacker *correctly captures the computational capabilities of a real-life attacker*—that is, the PPT model correctly captures all "real-life" computation that can be feasibly carried out by an attacker in our physical world. The *extended Church-Turing hypothesis* stipulates that this is the case:

> **The extended Church-Turing Hypothesis:** *A probabilistic Turing machine can efficiently simulate any realistic model of computation.*

But whether this hypothesis holds is strictly speaking a religious, as opposed to scientific, belief.[3] Indeed, the advent of quantum computing directly challenges this hypothesis. Based on exciting developments in quantum computation, it is becoming increasingly clear that viewing an adversary simply as a polynomial-time Turing machine, or polynomial-size circuit, may not be so "realistic". Quantum computers have access to qubits that we believe cannot be described with classical bits or run by a classical, polynomial-sized circuit. Furthermore, by the no-cloning theorem [WZ82, Die82], quantum states cannot be copied or re-used, which is a common technique used by many classical security proofs. We remark that this impacts the security of protocols/primitives even for security games where the challenger C is purely classical (i.e., primitives implemented by a classical algorithm that the attacker interacts with using classical communication). In recent years, there has been a successful line of work that

[3] Without getting too deep into Philosophy, it seems reasonable to argue that the Extended Church-Turing Hypothesis does not pass Popper's falsifiability test [Pop05], as we do not have "shared ways of systematically determining" whether a probabilistic Turing machine cannot perform some task (as testified by the fact that the P v.s. NP problem is still open). As such, the statement of the hypothesis is no different from the classic example of "All men are mortal", which according to Popper's theory is not scientific as we do not have systematic procedures for deducing whether a person is immortal. This is in contrast to assumptions such as "Factoring products of random 1000-bit primes is hard for all physically realizable computation devices", as we do have a systematic way of determining whether some such device manages to complete the task—simply run it.

has focused on proving the security of cryptographic protocols against quantum adversaries (see e.g. [Sho94, Gro96, AC02, Wat09, BDF$^+$11, Unr12, Zha12, ARU14, Unr16, Mah18, BS20] for examples of cryptographic attacks, constructions, and techniques in a quantum world). A vast set of new cryptographic techniques have been developed to address the idiosyncrasies of quantum computation and their impact on the security of systems. But, to deduce "real-life" security from such security proofs, we still need to rely on a quantum version of the extended Church-Turing hypothesis (stipulating that quantum polynomial-time algorithms/circuits can simulate all realistic models of computation).

This begs the question: could there be *even more powerful, or even just incomparable, realistic adversaries* beyond quantum polynomial-time adversaries? After all, a hundred years ago, modern computers did not exist, and quantum physics was in its infancy. Consequently, predicting the computational power of an adversary a hundred years into the future seems unreasonable. If the quantum extended Church-Turing hypothesis is wrong, because of the advent of a new type of computation, it would force yet another re-examination of cryptography.

In this work, instead of tailoring security reductions to specific classes of increasingly powerful adversaries, we ask:

Can we have a well-founded theory of Cryptography without making assumptions on the limits of "physically realizable models of computation"?

Concretely, what if some human manages to (repeatedly) break the security of some cryptographic scheme. There is currently a heuristic leap of faith in our cryptography treatment that this human (and any physical phenomena they may be using) can be implemented in PPT/QPT. Can this leap of faith be avoided?

In particular, ideally, we would want a theory of cryptography without making any types of extended Church-Turing hypotheses, where the security of some scheme is *only* based on *falsifiable* assumptions of the type that some computational task cannot be solved by a "physically realizable computation".[4]

Towards this goal, we will focus our attention on *classical primitives* (i.e., security games with challengers C that are classical and where the attacker can communicate with C only by using classical communication), but consider attackers with unknown/unbounded computational capabilities. At first sight, doing to seems to inherently require information-theoretic security (and all the standard limitations thereof). But our approach will instead be to consider a *purely-reduction based framework*: Our framework will provide a way to reduce the security of a game with challenger C to one with a challenger C' without assuming anything about the adversary *other than the fact that it continually wins in C*. Now, rather than proving the security of some primitive C w.r.t.

[4] For concreteness, and to simplify notation, we will model attackers as Turing machines so technically we are still relying on the (more reasonable) non-extended Church-Turing hypothesis. But we highlight that nothing in our treatment requires doing so and none of our results would change if we instead allowed any, even non-computable, attackers. See Section 3 for more discussion.

PPT attackers based on assumptions of the form "C' cannot be broken by PPT attackers", we will view the reduction from C to C' as the main goal: the existence of such a reduction will then imply the statement *"Security of C with respect to any physically realizable attacker holds as long as security of C' holds with respect to any physically realizable attacker"*, without having to impose any restrictions on what the class of "physically realizable attackers" actually is (as long as they only communicate with the challenger C using classical communication). We note that this reduction-based approach follows intuitions similar to those by Rogaway in his influential "formalizing human ignorance" paper [Rog06], where a purely reduction-based approach is also advocated for (but for a different reason, and where the standard notion of a reduction is employed).

Let us emphasize that whereas our framework is not imposing any upper bounds on the class of feasible computation (hence the name "universal"), we will be assuming a lower bound: in accordance with the standard literature, we will use PPT as a *lower bound* on what can be feasibly done by an attacker.[5] (In other words, polynomial-time computations will be considered realistically feasible, today and forever in the future.) Additionally, we will here focus our attention only on cryptographic primitives where the honest players are standard PPT machines (as opposed to e.g. quantum).

1.1 Universal Reductions in a Nut-shell

Towards defining our reduction-based notion of security, we need to start off by specifying the notion of an attacker we consider. An *augmented adversary* (A, Nat) consists of a uniform PPT interactive Turing machine (ITM) A, known as the *attacker*, and a *stateful*, potentially unbounded ITM Nat, known as the *Nature*. We think of A as the part of the augmented adversary that only uses "standard" computational resources, whereas Nat is a shared resource in the world that may have "magical" computational resources. The stateful nature of Nat is what distinguishes our model from more standard models of "black-box" security used in cryptography. We think of A as some real-life attacker (using today's readily available computing infrastructure) that can interact with a physical Nature Nat. Furthermore, A's interactions with Nature may in turn alter Nature. For instance, if Nat can capture quantum physical phenomena, then by the no-cloning theorem [WZ82, Die82], any type of measurements of Nat may alter it in ways that cannot be reversed (without losing information). Thus, statefulness is key for capturing this.

Roughly speaking, we say that there is a *universal reduction* from a security games C to a game C' if for every PPT A, there exists a "transformed" PPT attacker A' such that for every Nature Nat, if the augmented adversary (A, Nat) wins in the security game C, then the augmented adversary (A', Nat) wins in

[5] This model clearly oversimplifies as, say, n^{100} computation is not actually feasible. But we start off with a standard asymptotic treatment to get a model that is easy to work with. In practice, a more concrete treatment is desirable, but we leave this for future work.

the security game C'. In other words, the new transformed attacker A' needs to make use of the same Nature Nat as A.[6] (As the reader may notice, this notion captures an "existential" as opposed to a "constructive" notion of a reduction—that is, we are only required to show that a transformed attacker A' *exists*, as opposed to constructively providing it using an efficient transformation from A; we will also discuss constructive notions of reductions below.) We emphasize that A' may only communicate with Nat; it may not rewind, restart, or see any of the implementation details of Nat. In essence, we require A' to win in C' by making use of Nature, much like the original attacker A did, and taking into account that its interaction with the cosmos may alter it. The reasons we model A and A' as PPT, is that we consider PPT as a *lower bound* on what is currently feasible, and assume that this lower bound is valid not only today but also in the future (i.e., we will be able to only do more computation in the future). Thus we can write security proofs today that hold regardless of how powerful the universe ends up being (i.e. even if the extended Church-Turing hypothesis turns out to be true). All non-PPT computation can be thought of as being inside Nat.

Comparison with Relativized Reductions and UC security. Before proceeding to further formalizing this notion, let us briefly point out some technical similarities and differences with the notion of a *relativized reduction* (see e.g., [IR95]); roughly speaking, a relativized reduction, and the related notion of a black-box reduction, is a reduction that works even if the attacker has access to some arbitrary (perhaps non-efficienctly computable) function (a.k.a. the "oracle"). The main difference between the notion of a universal reduction and that of relativized reductions is that universal reductions can be viewed as reductions that relativize also with respect to an *interactive, stateful* oracle, whereas relativized reductions are only required to work in the presence of a *non-interactive, stateless*, oracle. As we explained above, considering stateful, interactive, Natures is a crucial aspect of our definition; as we shall see shortly, even formalizing how to deal with stateful oracles/Natures will be non-trivial.

We highlight that the idea to consider cryptographic protocols in the presence of an external stateful entity is also not entirely new: the notion of Universally Composable (UC) security [Can01] does exactly this but in a different context—more specifically, in the context of simulation as opposed to in the context of reductions; see Section 1.6 for more discussion on the relationship between universal reductions and UC.

[6] We refer to such reductions as "universal" because they are agnostic to the computational resources of an attacker (and thus can be "universally" applied, independent of the attacker's computational power). Additionally, on a technical level, and as we discuss in more detail shortly, considering security relative to a stateful entity is related to how security is defined in the framework for Universal Composability of Canetti [Can01].

1.2 Formalizing Universal Reductions

To formalize the notion of a universal reduction we first need to define what it means for (A, Nat) to win in some security game C. The standard notion of winning simply requires the attacker to succeed in convincing C *once* with some probability p. For us, since we consider stateful Natures, this will be too weak. A stateful Nature Nat may decide to be helpful in winning with C just once, and then never again, and such a Nature may not be very helpful in breaking some underlying assumption (at least not repeatedly). In the standard models of reductions, this is not a problem since the attacker can simply be restarted, but this is not allowed in our setting. Consequently, to get a meaningful notion of security, we will restrict our attention to (ruling out) attackers that *repeatedly*, or *"robustly"*, win in the security game, no matter what other communications are taking place with the cosmos. In more detail, we consider any history of interaction ρ that Nat may have seen, where an interaction prefix ρ consists of the messages Nature has received and the random coins it may have tossed. We then require (A, Nat) to succeed in winning for C even if (A, Nat) is fed any such prefix ρ. We denote such an interaction, where entities are provided the security parameter 1^λ, as $\langle C \leftrightarrow A \leftrightarrow \mathsf{Nat}(\rho)\rangle(1^\lambda)$.

In other words, we are considering an attacker A that is interacting with some physical stateful Nature Nat with unknown computational capabilities, but also consider the possibility that there are others in the world (captured by the prefix ρ) that have interacted with Nature in ways that are unknown to the attacker. Still, the attacker needs to succeed in breaking C *no matter what those other prior communications are* (i.e. given any transcript of interactions that previously took place). In fact, this transcript may be of any length, that is, more than just polynomial in λ (noting that Nat may have more than polynomially many interactions in the past).[7]

Definition 1 (Robustly Winning Security Game; Informal). *Let C be a challenger in a security game. We say that an* augmented adversary (A, Nat) has robust advantage $a(\cdot)$ in C *if, for every prefix view ρ, security parameter $\lambda \in \mathbb{N}$, it holds that C outputs 1 with probability at least $a(\lambda)$ in the interaction $\langle C \leftrightarrow A \leftrightarrow \mathsf{Nat}(\rho)\rangle(1^\lambda)$.*

Given this notion of robust winning, we can now capture the above-mentioned notion of a universal reduction.

Definition 2 (Universal Reduction; Informal). *Let C and C' be security games. We say that there is a* ϵ-universal reduction from C to C' *if for every PPT A, there exists some PPT A', such that for every* Nat*, if the augmented adversary (A, Nat) has robust advantage $a(\cdot)$ in C, then (A', Nat) has robust advantage $\epsilon'(\cdot)$ in C' where $\epsilon'(\lambda) = \epsilon(\lambda, a(\lambda))$.*

[7] Nevertheless, we note that all our results also hold if restricting the length of ρ to be polynomial.

The function ϵ here quantifies the security degradation of the reduction. Let us briefly mention that one may also consider an *a priori* weaker looking notion of a "win-once" universal reduction, that only requires the *transformed* attacker (A', Nat) to have *non-robust* advantage $\epsilon'(\cdot)$ in C'; that is, (A', Nat) is only required to win once in C' as opposed to robustly/repeatedly (while the original attacker (A, Nat) still needs to have robust advantage). As it turns out, this weaker notion is equivalent to the one provided in Definition 2; see Lemma **??** for more details. We also note that one may consider alternative, seemingly weaker, variants of robustness (e.g., that the attacker only wins an inverse polynomial fraction of the time) but again such a notion turns out to be equivalent (up to a difference in parameters); see the full version [CFP22] of this paper for more details.

Black-box reductions and Dummy adversaries. As mentioned above, the notion of a universal reduction is "existential" as opposed to a "constructive": We do not actually require an efficient transformation taking attackers A to attackers A'; rather, we just need to show that that for every attacker A, the attacker A' *exists.* One could also consider an *a-priori* stronger notion of a *universal black-box reduction* where the transformed attacker A' is defined as $A' = R^A$, where R is fixed PPT (that works for any attacker A). As it turns out, this notion is (again) equivalent to the (existential) notion of a universal reduction provided in Definition 2. The reason for this is that to prove the existence of a universal reduction, and actually also a universal black-box reduction, it suffices to show that the reduction applies just to a so-called "dummy" adversary A_{dummy} that essentially just forwards messages between C and Nat; this, intuitively, follows from the fact that we can always push all the work of a prospective attacker A into Nat (more formally, considering a new Nature Nat' that combines Nat and A). We note that a similar phenomena happens for the notion of UC security [Can01], and we are borrowing the term of a "dummy" adversary from there.

Lemma 1 (Dummy Lemma; Informal). *Let C and C' be security games. Assume that there exists some ϵ and some PPT R_{dummy} such that for every Nat, if the augmented adversary A_{dummy} has robust advantage $a(\cdot)$ in C, then $(R_{\mathsf{dummy}}, \mathsf{Nat})$ has robust advantage $\epsilon'(\cdot)$ in C' where $\epsilon'(\lambda) = \epsilon(\lambda, a(\lambda))$. Then, there exists an* ϵ-universal black-box reduction from C to C'.

We highlight that whereas the actual proof of Lemma 1 indeed follows the above intuition, the formalization is quite subtle and quite different from the proof of the dummy lemma in the UC framework—the key obstacle is dealing with the fact that the attacker needs to win robustly.

Note that as a consequence of Lemma 1, we have that to prove the existence of a universal reduction, we may without loss of generality assume that $A = A_{\mathsf{dummy}}$ (i.e., in essence that Nat is directly breaking C), and thus proving the existence of a universal reduction amounts to showing the existence of a PPT "filter" $A' = R_{\mathsf{dummy}}$ between C' and Nat.

Composition. We additionally note that the notion of a universal reduction composes. Namely, if hardness of C_1 can be based on the hardness of C_2, and hardness of C_2 can be based on the hardness of C_3, then hardness of C_1 can be based on hardness of C_3.

Theorem 1 (Composition Theorem; Informal). *Let C_1, C_2, C_3 be security games. Suppose there exists an ϵ_1-universal reduction from C_2 to C_1, and an ϵ_2-universal reduction from C_3 to C_2. Then, there exists an $\epsilon^\star$-universal reduction from C_3 to C_1 where $\epsilon^\star(\lambda, a) = \epsilon_1(\lambda, \epsilon_2(\lambda, a))$.*

The proof of the composition theorem essentially follows directly from the definition of a universal reduction.

1.3 On the Feasibility of Universal Reductions

We turn to studying the feasibility of universal reductions.

Universal reductions from single-shot, straightline, black-box reductions. We observe that any *straight-line* black-box reduction between C and C' that only invokes the attacker *once* is also a universal reduction. This should not be a surprise since the stateful nature of the attacker in our model never becomes an issue if the reduction only invokes the attacker once. Nevertheless, our model formally demonstrates why such simple types of reductions are advantageous from a (qualitative) security point of view.

Theorem 2 (Universal Reductions from Single-shot Straightline Black-box Reductions; Informal). *Let C and C' be security games. Suppose there exists an ϵ-straightline black-box reduction from C to C' that interacts with the adversary once. Then there exists an ϵ-universal reduction from C to C'.*

Fortunately, many well-known reductions in cryptography fall into this class of reductions: PRG length extension, the GGM construction of PRFs from PRGs [GGM86], IND-CPA secure encryption from PRFs, Naor's bit commitments from PRGs [Nao91], and Lamport's one-time signatures from OWFs [Lam79]. We note that for Lamport's construction, this is straightforward to see. For the rest of the proofs, we rely on a uniform security analysis for a hybrid argument, which for example is provided in [Gol07] for PRG length extension. For the convenience of the reader, we provide brief sketches for the constructions and proofs for all of these primitives in the full version [CFP22] of this paper.

Combining these classical results with Theorem 2, we thus directly get the following corollaries (formally stated in the full paper [CFP22]):

Corollary 1 (PRG length extension; Informal). *Let m be a polynomial and G be an $\lambda + 1$-bit stretch PRG. There exists a $m(\lambda)$-bit stretch PRG G_m and an ϵ-universal reduction from the PRG security of G_m to the PRG secutiy of G for $\epsilon(\lambda, a) = 1/2 + \delta/m(\lambda)$, where $\delta = a - 1/2$.*

Corollary 2 (PRF from PRGs; Informal). *Let G be any PRG. There exists a PRF F and an ϵ-universal reduction from the PRF secutiy of F to the PRG security of G for $\epsilon(\lambda, a) = 1/2 + \delta/\text{poly}(\lambda)$, where $\delta = a - 1/2$.*

Corollary 3 (IND-CPA secure private-key encryption from PRGs; Informal). *Let G be any PRG. There exists a private-key encryption scheme and an ϵ-universal reduction from the IND-CPA security of the encryption scheme to the PRG security of G for $\epsilon(\lambda, a) = 1/2 + \delta/2 - \mu(\lambda)$ for a negligible function μ, where $\delta = a - 1/2$.*

Corollary 4 (Commitment schemes from PRGs; Informal). *Let G be any PRG. There exists a statistically binding commitment scheme and an ϵ-universal reduction from the hiding of the commitment scheme to the PRG security of G for $\epsilon(\lambda, a) = 1/2 + \delta/2$, where $\delta = a - 1/2$.*

Corollary 5 (One-time Signatures from OWFs; Informal). *Let f be any OWF. There exists a signature scheme and an ϵ-universal reduction from the one-time security of the signature scheme to the OWF security of f for $\epsilon(\lambda, a) = a/(2\lambda)$.*

Universal reductions from new single-shot straightline reductions. Often times, security reductions used in the literature do invoke the attacker multiple times, and it may not be clear how such reductions can be translated to work in the setting of universal reductions. We first show that sometimes famous reductions in the literature that require invoking the attacker multiple times can be made single-shot straightline. In particular, we show that the GMW protocol [GMW91] for graph 3-coloring is witness indistinguishable (WI) [FS90] based on a universal reduction to a commitment scheme (and hence PRGs) with a new proof; the standard proof requires rewinding the attacker and would thus not be applicable in our setting. (This proof may be interesting in its own right; as far as we know, the only proof of WI security of GMW with a straight-line reduction is the work of Hofheinz [Hof11] that shows WI security of GMW when the underlying commitment satisfies a notion of selective-opening security. As far as we know, it was an open problem to present a straight-line reduction based just on standard security; this is what we do.)

Theorem 3 (Witness Indistinguishability from PRGs; Informal). *Let G be any PRG. For every language in NP, there exists an interactive proof system (P, V) and an ϵ-universal reduction from the WI of (P, V) to the PRG security of G for $\epsilon(\lambda, a) = 1/2 + \delta/\text{poly}(\lambda)$, where $\delta = a - 1/2$.*

Beyond single-shot straightline reductions. While Theorem 3 provides some initial hope that more reductions in the literature can be made single-shot straightline, there are other classic reductions that we do not know how to make single-shot. In fact, going one step further, we next show that some classic results in the literature cannot be established with respect to universal reductions.

One of our main results shows that Yao's classic result on hardness amplification of any weak one-way functions via direct product [Yao82] cannot be proven with a universal reduction. In fact, we show that hardness of any arbitrary "black-box" one-way function cannot be amplified essentially *at all* using a n-fold direct product. Given a function f, let $f^{(n)}$ denote the n-fold direct product of f:

Theorem 4 (Impossibility of Hardness Amplification; Informal). *Consider some polynomial n, and some function ϵ. Suppose there is an ϵ-universal black-box reduction from the OWF security of $f^{(n)}$ to the OWF security of f that uses only black-box access of f, and that works for any function f. Then, there exists a negligible function μ such that $\epsilon(\lambda, a) \leq a + \mu(\lambda)$.*

Note that there is a trivial reduction that embeds the challenge $f(x)$ a single time into a random location of the output of $f^{(n)}$ that has advantage $\epsilon(\lambda, a) = a$. The above theorem says that no universal reduction can do noticeably better than this trivial reduction, even if considering attackers that succeed with some fixed constant probability, say $\frac{1}{2}$.[8]

To give some intuition behind the proof of Theorem 4, let us recall on a very high-level how Yao's original proof works: given as input y, the reduction embeds y into a random "position" i—letting $y_i = y$, generates random pre-images x_j for $j \neq i$, and lets $y_j = f(x)$, $\vec{y} = y_1 y_2 \ldots$ and then runs $A(\vec{y})$. If A fails, then we repeat the process (a polynomial number of times), again embedding y into a new random position i. Note that this reduction is thus repeatedly running A on *correlated* inputs—the inputs all contain the same string y (but except for that y, they are independent). An augmented adversary could notice these correlations and may stop working in case it sees correlations of this form (i.e., a substring y that is repeated from a previous query). Note that such an attacker still robustly wins in the security game: the probability that a fresh input from the challenger coincides with any previously seen strings is negligible.[9] Now, an arbitrary reduction may not necessarily work in the same way as Yao's reduction. However, at a high level, we show that if the reduction works for *any* function f (and only uses the function as a black-box), then the reduction has to ask A on inputs that are correlated, and thus we can still use a similar type of attacker.

[8] We emphasize that Theorem 4 is ruling out also so-called "parameter-aware" black-box reductions [BBF13], where the reduction may depend on the success probability a of the attacker; note that Yao's original reduction is parameter dependent—more specifically, the number of repetitions is required to be superlinear in the adversary's success probability, and as shown in [LTW05] a dependency on the attackers success probability is inherent for black-box reductions. Theorem 4 rules out also such parameter-aware universal reductions and indeed rules out universal reductions that increase the success probability of the adversary even if assuming that the original attackers success probability is, say, $\frac{1}{2}$.

[9] There is a small subtlety here. Robustness is defined with respect to all previous transcripts, even exponentially long ones, so naively implementing this approach will not work since eventually we can include all possible strings y in the transcripts. Rather, the way we formalize this argument is to consider a Nat that only has "polynomial memory" and checks for repeated strings y in the most recent part of the transcript it is fed.

We additionally show that the universal aspect of Theorem 4 (i.e., that it works for *any* function f) is inherent. If the function f is *rerandomizable* (see the full paper for a formal definition), then we can show a universal reduction for hardness amplification of f—in essence, we show that Yao's reduction directly works. At first sight, this may seem surprising: As mentioned above, Yao's reduction does invoke the attacker multiple time, and does so on correlated inputs (and as discussed above, this correlation lead to problems). Rerandomizability helps overcome this issue and enables the reduction to always feed Nat messages that are independent and have the same distribution.

For our next result, we show that the Goldreich-Levin theorem [GL89] for constructing a OWF with a hardcore predicate from any OWF cannot be turned into a universal reduction, again as long as the underlying OWF is only accessed in a black-box way. For an underlying function g, the Goldreich-Levin theorem shows that the inner product function is hardcore for the "randomized" function $\hat{g}(x,r) = (g(x),r)$. Namely, $\langle x,r \rangle$ cannot be predicted given $(g(x),r)$ where $|x| = |r|$. We extend our impossibility to any predicate h for any length of randomness r (even no randomness).

Theorem 5 (Impossibility of a Goldreich-Levin Theorem; Informal). *Consider some function ϵ and some efficiently computable predicate h . Suppose there is an ϵ-universal black-box reduction from the security of the hardcore predicate h w.r.t. $\hat{g}(x,r) = (g(x),r)$ to the OWF security of g that uses only black-box access to g and that works for function g. Then, there is a negligible function μ such that $\epsilon(\lambda,a) \leq \mu(\lambda)$ for all $a \leq 0.99$.*

The proof relies on similar intuitions to the hardness amplifications result.[10] The above theorem gives an indication of why it may be hard to come up with a universal reduction from PRGs to OWFs as known constructions of PRGs from OWFs rely on the Goldreich-Levin theorem. We leave open the question of whether there exists some alternative way to universally reduce PRGs to OWFs.

Concluding, while we can write nice universal reductions in some settings, we also have some pretty severe impossibility results. To overcome these impossibility results, we additionally consider more relaxed—yet, in our eyes, natural—variants of universal reductions.

1.4 Restricted Classes of Natures

While it is natural to assume that an attacker can affect Nature/the Cosmos, it also seems reasonable (at least in some contexts) to make additional assumption on the class of Natures. In particular, we will consider Natures that act *independently of the content of interactions they had "far back" in the past.* Roughly

[10] Again, we highlight that Theorem 5 rules out also "parameter-aware" reductions that depend on the success probability of the attacker—in fact, it rules out also reductions that only work if the underlying attacker's success probability is 0.99. (As noted in [BBF13], Goldreich-Levin's standard reduction is parameter-aware, and this is inherent as shown in [LTW05].)

speaking, we allow Nature to change over time, and we will allow Nature to be stateful within a single, or a bounded number of, sessions but assume that the actual content of messages received too farin the past (that is, many messages ago) does not significantly affect the behavior of Nature.

In more detail, choose any polynomial function $k(\cdot)$, and consider those natures whose responses depend only on (a) the *number of queries* it has received in the past, (b) the *last* $k(\lambda)$ *messages* that it received, and (c) the randomness that it used to respond to those $k(\lambda)$ messages. We call a nature that satisfies these conditions a *time-evolving* k*-window nature*. We formalize this by requiring that the output of Nature given any two prefixes ρ and ρ' of the *same length* that also share the last $k(\lambda)$ messages and coins, it must be that $\mathsf{Nat}(\rho)$ behaves identically (or ϵ-close to) $\mathsf{Nat}(\rho')$. (The same-length requirement is what allows Nature to evolve over time).

Definition 3 (Time-Evolving k-Window Natures). *Let $k(\cdot)$ be a polynomial function. A Nature machine Nat is said to be a k-window Nature if there exists a negligible function μ s.t. for all machines C, $\lambda \in \mathbb{N}$, and interaction prefixes ρ, ρ', ρ'', where $\|\rho\| = \|\rho'\|$ and $\|\rho''\| = k(\lambda)$, it holds that*

$$\Delta\left(\langle C \leftrightarrow \mathsf{Nat}(\rho \circ \rho'')\rangle(1^\lambda), \langle C \leftrightarrow \mathsf{Nat}(\rho' \circ \rho'')\rangle(1^\lambda)\right) \leq \mu(\lambda).$$

where $\langle C \leftrightarrow S\rangle(1^\lambda)$ denotes the output of C in an interaction with a machine S, Δ denotes statistical distance, $\|\rho\|$ denotes the number of messages contained within ρ, and $\rho \circ \rho''$ denotes prefix concatenation.

Observe that by sending to Nature a sequence of $k(\lambda)$ "dummy messages" $\perp$, we can (roughly speaking) reset the state of a time-evolving k-window Nature, by making it so that its behavior only depends on those dummy messages (and corresponding coins) and the number of messages it received in the past—regardless of the state that Nature started in before receiving those dummy messages. In other words, we can think of an augmented adversary (A, Nat) where Nat is time-evolving k-window (when called repeatedly, each time utilizing the above resetting procedure) as a sequence of attackers $A_1, A_2, \ldots$ such that (1) each individual attacker A_i succeeds in the security game, but (2) the way it succeeds may be different, and (3) the security reduction cannot restart the attacker but may "move on" to the next attacker in the sequence.

As our main result for time-evolving k-window Natures, we show that any *non-adaptive*, straight-line black-box classical reduction can be transformed into a universal reduction, when restricting to time-evolving k-window Natures. In more detail, we refer to a straight-line black-box reduction R as non-adaptive if R interacts with the challenger C and attacker A according to the following pattern:

- R starts by interacting with C for any number of rounds of its choice; at some point it decides that it wants to start communicating with the attacker A.
- At this point, R selects m different PPT machines $M_1, M_2, \ldots, M_m$.

- For each $i \in [m]$, we let M_i communicate (straight-line) with a fresh instance of A, and let a_i denote the output of M_i at the end of the interaction.
- Finally, R gets back the answers $a_1, \ldots, a_m$ and gets to continue interacting with C.

We show:

Theorem 6 (Universal Reductions from Non-adaptive Reductions; Informal). *Let C, C' be challengers. If there exists a non-adaptive straight-line black-box reduction from C to C', then for any polynomial $k(\cdot)$, there exists a universal reduction from C to C' w.r.t. time-evolving k-window Natures.*

At a very high level, the idea behind the proof of Theorem 6 is the following. Recall that (roughly speaking) an augmented adversary, with a time-evolving k-window Nature, can be treated as a sequence of attackers $A_1, A_2, \ldots$ that is fixed ahead of time and utilized in order. Such a sequence of attackers can essentially be turned into a "standard" fully restartable attacker by, at each invocation, choosing a random attacker A_i out of the sequence of attackers. Of course, in a real execution we are forced to utilize $A_1, A_2, \ldots$ in sequence and in order. Fortunately, for any non-adaptive reduction, we can emulate (with inverse polynomial statistical gap) this standard randomized restartable attacker by *permuting* the order of the queries of the reduction, and inserting these queries into a sufficiently long bogus interaction. Note that we here inherently rely on the fact the a time-evolving k-window Nature can be reset so that the last k messages no longer affects its state, so that its behavior depends on only the length of the prefix of messages it receives.

We remark that many (but not all) of the classical reductions in the cryptographic literature are of the non-adaptive type. In particular, these include reductions such as those in Yao's hardness amplification [Yao82] and the Goldreich-Levin Theorem [GL89] (which we proved could not be shown using a "plain" universal reduction). Perhaps surprisingly, our results therefore imply that we can achieve hardness amplification or hard-core bits for attackers that change their behavior across queries (albeit in this limited way).

k-Window Natures. We finally turn our attention to the more restrictive class of simply *k-window Natures* (i.e., not time-evolving), that are identically defined except that we quantify over any two prefixes ρ and ρ' (with the same last $k(\lambda)$ messages and coins), and not just those of equal length. We observe that straight-line black-box reductions, even those that are *adaptive*, that only sequentially invoke the attacker in multiple sessions, directly imply universal reductions w.r.t. k-window Natures; this essentially follows directly from the definition (by using a standard hybrid argument), and by the observation that sending such a Nature k dummy messages resets it to a default state (from which is acts indistinguishably):

Theorem 7 (Universal Reductions from Adaptive Reductions; Informal). *Let C, C' be challengers. If there exists a (possibly adaptive) sequential straight-line black-box reduction from C to C', then for any polynomial $k(\cdot)$, there exists a universal reduction from C to C' w.r.t. k-window Natures.*

1.5 Universal Reductions Imply Standard Reductions

As a sanity check, we finally observe that the existence of a universal reduction from C to C', even one that is only w.r.t. k-window Natures (where $k(\cdot)$ is large enough to bound the number of rounds of interaction with C), implies the existence of a reduction for classic models of attackers such as PPT, non-uniform PPT, quantum polynomial time (QPT), and QPT with non-uniform quantum advice (which we refer to as non-uniform QPT). This follows by noticing that all these models of computations can be captured by a k-window Nature Nat, when used to win a k-round security game C. For the case of PPT, non-uniform PPT, and (uniform) QPT, this is trivial. For non-uniform QPT, it is a bit more problematic since a non-uniform QPT algorithm may make some measurement that ruins the non-uniform advice in a way that makes the algorithm non-restartable. But this issue can be resolved by, for every bound $b(\cdot)$ on the number of restarts, considering a Nat that contains $b(\lambda)$ copies of the non-uniform quantum advice. The resulting attacker (A', Nat) that breaks C' will then still be non-uniform QPT (albeit with longer non-uniform advice than the original attacker breaking C).

Theorem 8 (Classical Reductions from Universal Reductions; Informal). *Let C and C' be security games, and let $k(\cdot)$ be a polynomial function that upper bounds the number of rounds in any interaction with C. Assume there exists a ϵ-universal reduction from C to C' w.r.t. (k, μ)-window Natures, for an arbitrary choice of μ. Then there exists a ϵ-reduction from C to C' w.r.t. PPT, non-uniform PPT, QPT, and non-uniform QPT attackers.*

A Note on Post-quantum Security. Note that Theorem 8 shows that if you can base the security of some (classical) security game C on the security of C' using a universal reduction (even with respect to just k-window Natures), then it implies resilience of C with respect to quantum attackers if assuming that C' is secure with respect to quantum attackers.

Let us highlight, however, that this result only holds true to security games C that themselves are classical. For instance if C is the security game of a PRF and C' is that of a PRG, then we only get quantum security of the PRF with respect to attackers that can get evaluations of the PRF on classical inputs. (As such, the combination of Corollary 2 and Theorem 8 does not subsume the results of Zhandry [Zha12] showing post-quantum PRF security of the GGM construction [GGM86] since Zhandry notably allows the attacker to make quantum queries to the PRF.). In other words, our framework currently only consider primitives where the *honest* players are classical. (Of course, we could extend our model to also deal with quantum security games but we believe it is a more pressing issue to get a "future-proof" notion of security w.r.t., cryptographic primitive and protocols that are run by honest players on classical computers).

1.6 Conclusions, Related and Future Work

Interpreting our Results. Our results demonstrate both limitations and feasibility of universal reductions—that is, the feasibility of a foundation for cryptographic security without making extended Church-Turing type assumptions about the class of physically-realistic computations. This paper is only a first step—we have not done an extensive survey of all the reductions in the literature, and we have not investigated all primitives out there; notably, we have focused only on the most basic primitives/reductions. We leave an exploration of more advanced primitives, such as zero-knowledge proofs and secure computation for future work.

Taken together, our result provide a new qualitative understanding of how different types of restrictions on black-box reductions result in security w.r.t. stronger classes of attacker. In particular, when restricting our attention to straight-line black-box reductions: (1) reductions that only invoke the attacker once, yield the strongest form of "plain" universal reduction, (2) reductions that are non-adaptive yield universal reductions w.r.t. time-evolving k-window Natures, and (3) adaptive ones yield universal reductions w.r.t. k-window Natures, for any choice of polynomial $k(\cdot)$.

So given our three different notions of security (which we have shown all imply standard notions of security), which one should we aim to achieve? Obviously the strongest form of plain universal reduction is the most desirable as it allows us to argue security while making only minimal "religious" assumptions about the class of physically-feasible computation. Our results demonstrate that indeed this notion is achievable for many constructions of interest (e.g., for primitives proven secure using straight-line black-box reduction that call the attacker once, or for some cases even multiple times when the queries are independent). Our impossibility results, however, also demonstrate important limitations, showing that in some situations, stronger types of "religious" assumptions about the class of feasible computation are required. The class of time-evolving k-window Natures seems like a reasonable midpoint between expressivity of the theory and the assumptions made on the class of physically-feasible computation.

More Justification for Time-Evolving k-Window Natures. Let us briefly comment on the recent and independent of work of Bitansky, Brakerski, and Kalai ([BBK22]), who study the quantum security of non-interactive reductions. Similar to us, they propose a framework to deal with stateful attackers, and show that non-adaptive reductions (with a polynomial solution space, including decisional assumptions) imply post-quantum security with a *uniform* reduction. In more detail, [BBK22] leverages the main result of Chiesa et al. [CMSZ21] that shows how to effectively "rewind" quantum attackers for a restricted class of protocols so that they effectively become time-evolving but otherwise stateless (or rather, bounded memory)—[BBK22] refer to such attackers as persistent solvers. Next, [BBK22] rely on a proof that is very similar to the proof of our Theorem 6 to show that non-adaptive black-box straight-line reductions can be applied to such attackers.

Note that our Theorem 8 shows that universal reduction w.r.t. not only time-evolving k-window, but also simply k-window Natures (which by Theorem 7 are implied by also adaptive straight-line black-box reductions) imply quantum security but it requires using a *non-uniform* reduction. By relying on the results of [CMSZ21], [BBK22] effectively show that universal reductions w.r.t. time-evolving k-window Natures have the advantage that the reduction for quantum security—for *specific* security games—becomes fully uniform. Consequently, we take the works of [CMSZ21, BBK22] as further evidence that restricting attention to universal reductions w.r.t. time-evolving k-window Natures is meaningful.

Comparison to Universal Composition (UC). Let us highlight that some of the intuition behind our definition take inspiration from the framework for Universal Composability (UC) by Canetti [Can01]. In particular, a simulator in the UC framework needs to interact with the attacker in a black-box straight line fashion in the presence of any environment, without the power of rewinding or restarting the environment. Clearly, there are many similarities between the notion of an environment and our notion of Nature. As such, one may be tempted to hope that UC protocols automatically have universal reductions. This intuition is misleading (as demonstrated e.g., by our Theorem 4). The reason for this is that whereas the simulator in the UC framework is required to be straight-line (and the attacker/environment is allowed to be fully stateful), the *security proof/reduction* used to argue that the simulation is "correct" (i.e., indistinguishable from the real execution in the eyes of the environment) may very well use rewinding (and in fact often does). In more detail, standard proofs in the UC framework still assume that the environment is a non-uniform PPT machine to reduce security to some assumption (e.g., one-wayness of a function).

It is also worthwhile to compare universal reductions to UC security with an *unbounded environment* (in analogy with how we consider Natures that are unbounded). While such a notion of UC security indeed also would be "future-proof" in the sense that it does not make any assumptions about computational limits on the class of physically realizable computations, the problem with such a notion is that it only enables information-theoretically secure protocols, whereas our goal here is to develop a computational theory of cryptography that is "future-proof". One could consider defining primitives (e.g., one-way functions, PRGs, signatures) as UC functionalities, and consider whether one functionality can be implemented in a UC way using some other functionality with respect to a computationally-unbounded environment; as far as we are aware, such a method has not previously been advocated for and is in line with what we are doing here. However, we highlight that doing this is non trivial for several reasons: (1) it is non trivial to define standard cryptographic primitives as UC functionalities (e.g., how does one define an idealized one-way function); (2) such a treatment would require presenting a *straight-line reduction* that is required to work even if the environment (i.e., Nature in our langauge) only helps the attacker to succeed *once*; as we have argued above, such a notion is overly strong (and it is trivial to present impossibility results for it). In contrast, by focusing directly on a reduction-based framework, we can (1) define primitives in the standard game-based way, (2) only require the reduction to work for attackers that win *robustly* (i.e., repeatedly) to rule out trivial cases when Nature helps the attacker to win just a single time.

Let us finally mention that a natural way to define protocol security in a both universally-composable and universally-reducible way would be to consider the standard UC definition of security, but requiring that the security reductions used to prove indistinguishability of the simulation are universal. We leave an exploration of such protocols for future work.

Comparison to Abstract Cryptography. We end by noting that the frameworks for *abstract cryptography* [MR11], and *constructive cryptography* [Mau11], among other things also have as a goal of building up a theory of cryptography that is independent of the model of computation used to model an adversary. While these frameworks were used to analyze how to obtain higher-level functionality (e.g., secure channel) from advanced primitives (e.g., secure encryption and MACs) and also used to analyze some building blocks (for instance see [Mau02, MP04, MPR07]), as far as we can tell, they have not been used to understand the underlying most basic building blocks that we study here (e.g., hardness amplification of one-way functions, whether one-way functions have hard-core bits, etc). At a very high-level, the idea is to view security reductions among primitives as simulations of one system in terms of another; these simulations, just as in the UC framework, need to be straight-line, black-box, and only invoke the attacker once. As far as we can tell, consequently, the same two differences as presented w.r.t. UC with an unbounded environment also apply here. Most notably, since we restrict attention to attackers that win repeatedly/robustly, we can obtain feasibility results using reductions that invoke the attacker multiple times (and this is also what makes it significantly more challenging to present impossibility results).

We highlight that also in the constructive cryptography, *computational* simulation has been defined to consider tasks requiring computational assumption, but this is defined by restricting attention to polynomial-time distinguishers, so such computational definitions still rely on a extended Church-Turing assumption. It would be interesting to extend these works by considering a computational notion of indistinguishability based on universal reductions.

2 Overview of Techniques

We now describe our main technical contributions. We direct the reader to the full paper [CFP22] for full proofs and theorem statements.

2.1 The Dummy Lemma

Universal reductions give universal black-box reductions. (See Lemma 1) Consider a "dummy attacker" A_{dummy} that forwards all messages from C to the Nature Nat, and forwards replies from Nat back to C. The "dummy lemma" says (informally) that if there exists a universal reduction R_{dummy} between two security games that works for augmented adversaries of the form $(A_{\mathsf{dummy}}, \mathsf{Nat})$, then there exists a universal reduction that works for any augmented adversary (A, Nat). Moreover, it is constructive, and the resulting reduction uses A in a black-box way. Here, we briefly provide some intuition for why the"dummy lemma" holds.

The key observation is that since R_{dummy} works for any Nature talking with the dummy attacker A_{dummy}, it must in particular also work for the Nature Nat' that internally simulates an attacker A talking to Nat, for any augmented adversary (A, Nat). If (A, Nat) wins some security game C, then $(A_{\mathsf{dummy}}, \mathsf{Nat}')$ should also win an interaction with C, as Nat' is essentially simulating the augmented adversary (A, Nat) inside. Thus, the reduced attacker $(R_{\mathsf{dummy}}, \mathsf{Nat}')$ should also win the game C'. Finally, consider the reduction R^A that internally runs R_{dummy} and forwarding all its attacker messages to its oracle A. Since R_{dummy} is only talking to Nat' in a straightline fashion, intuitively, the augmented adversary (R^A, Nat) should behave exactly like $(R_{\mathsf{dummy}}, \mathsf{Nat}')$ and thus also win C'. Formalizing this intuition, however, is a bit tricky since we need to make sure that $(A_{\mathsf{dummy}}, \mathsf{Nat}')$ is also *robustly* winning in C, which requires a more complicated construction of Nat'; see the full paper [CFP22].

2.2 Straightline Black-Box Reductions and Witness Indistinguishability

We overview why single-shot, straightline, black-box reductions imply universal reductions, and use this to show a witness indistinguishable proof based on a universal reduction to PRG security.

Single-shot Straightline Reductions imply Universal Reductions. (See Theorem 2) We first argue that "single-shot" straightline black-box reductions imply universal reductions. Suppose there is a *classical* straightline, black-box reduction R that succeeds in some security game C' with probability ϵ when making single-shot usage of an adversary A with advantage a in the game C. That is, R interacts with A a single time without any rewinding or restarting. As we shall observe, any such reduction must also "relativize" with respect to any stateful oracle Nat. In more detail, consider some augmented adversary (B, Nat) that has robust advantage a in a game C, and let B' be an adversary that simulates a communication between R and B: Any time R wants to query its adversary A, we direct that communication to B, and any time B wants to query Nature Nat, we direct that communication to Nat. Since for every prefix ρ, we have that $(B, \mathsf{Nat}(\rho))$ wins in C, we also have that for every prefix ρ, $R^{(B, \mathsf{Nat}(\rho))}$ wins in C and thus $(B', \mathsf{Nat}(\rho))$ (which perfectly emulates $R^{(B, \mathsf{Nat}(\rho))}$) does so as well, so (B', Nat) also has robust advantage in C'. Note that this construction crucially relies on the fact that R only invokes its attacker *once* and without rewinding it (so that communication with Nat can be forwarded).

Let us emphasize, however, that universal reductions are not equivalent with single-shot straightline reductions: as we already discussed, we can obtain universal reductions that do reuse the attacker multiple time—we demonstrate this for the case of hardness amplification for rerandomizable functions—and for this task it is easy to see that a straightline single-shot black-box reductions cannot be used (see the full paper for details).

Universal Reductions from Some Classic Reductions. The above observation shows that if we can construct proofs of security using single-shot, straightline, black-box reductions, then we immediately can infer the existence of a universal reduction. We observe that indeed some of the classical proofs of security (for e.g. PRG length extension, PRFs from PRGs, encryption from PRFs, commitments from PRGs, one-time signatures from OWFs) fall into this category; see Corollaries 1 through 5.

Universal Reductions from New Classic Reductions: Witness Indistinguishable Proofs. (See Theorem 3) Many classic cryptographic reductions, however, do require rewinding/restarting the adversary. Most notable are reductions/simulations for notions of privacy in interactive proofs like zero-knowledge [GMR89]. As we shall see, we demonstrate that sometimes these can be "derewinded". In particular, we will focus our attention on a weakening of zero-knowledge, known as *witness indistinguishability* [FS90], and will show how to provide a new single-shot straighline reduction (and as a consequence, a universal reduction) to PRGs. (We hope that this proof will serve as an example of how classic proofs may be "de-rewinded".)

Recall that an interactive proof system [GMR89], (P, V), for an NP language L specifies an interaction between the prover P with access to a witness w and the verifier V, on common input a security parameter 1^λ and a statement x. It should satisfy completeness, meaning on inputs $x \in L$ and w a valid witness for x, $P(w)$ causes V to accept. The other required property is soundness, meaning on input $x \notin L$, no cheating prover $P^\star$ can cause V to accept (with noticeable probability). Sometimes we want additional privacy and security properties for the witness w used. One basic property is witness indistinguishability (WI) [FS90] which requires that no (potentially cheating) verifier $V^\star$ can tell if P is using one witness w_0 or another witness w_1. Note that this might seem like a weak property (e.g., it provides no guarantees for languages with unique witnesses), but it has been shown to be extremely useful for broader cryptographic applications (see e.g. [FS90, DN07, BG08]).

We show that the GMW protocol for graph 3-colorability [GMW91] is WI using a single-shot straightline reduction to PRG security. We note that previous classical proofs showing WI of the GMW protocol first showed that GMW is actually zero-knowledge and then use this to conclude that it also satisfies WI. But this approach requires rewinding the adversary; we shall dispense of this rewinding.

We proceed to recalling the GMW protocol. Let $G = (U, E)$ be the input graph where $U = [n]$. Recall that the prover P in this protocol has access to a valid 3-coloring $w\colon [n] \to [3]$ such that for all $(i, j) \in E$, $w(i) \neq w(j)$. To prove that the graph G is indeed 3-colorable, P samples a random permutation

$\pi\colon [3] \to [3]$ and commits to the colors $c_k = \pi(w(k))$ for all $k \in [n]$. V asks to open a random edge $(i, j) \in E$, and P responds with the openings revealing c_i and c_j. V accepts the interaction if $c_i \neq c_j$ and the openings are valid. Completeness of the protocol can be checked straightforwardly. The protocol has statistical soundness $(1 - 1/|E|)$ (meaning the verifier will catch a cheating prover with probability roughly $1/|E|$) by the statistical binding of the commitment, since at least one edge must be colored incorrectly if G is not 3-colorable. We proceed to argue WI by showing that no cheating verifier $V^\star$ can distinguish interactions with $P(w_0)$ or $P(w_1)$ for any two distinct witnesses w_0 and w_1.

To formalize this claim, we model WI as a security game as follows. We allow the adversary A to select a graph G and two valid witnesses w_0 and w_1. The challenger C samples a bit $b \leftarrow \{0, 1\}$ and proceeds to interact as $P(w_b)$ while A acts as the (potentially cheating) verifier V. After the interaction, A outputs a bit $b^\star$ and C outputs 1 (so A wins) iff $b = b^\star$.

Now suppose that there is an adversary A that distinguishes $P(w_0)$ and $P(w_1)$ with probability $1/2 + \delta$ (namely, it outputs 1 on $P(w_1)$ with probability 2δ more than on $P(w_0)$). We construct a straightline, black-box reduction R that uses A to distinguish two commitments to different values. R first receives a graph G and witnesses w_0 and w_1 from the adversary A. Next, R chooses a random edge $(i', j') \in E$ and random distinct colors for these vertices $c_{i'} \neq c_{j'} \in [3]$. R computes permutations π_0 and π_1 such that $\pi_0(w_0(\cdot))$ and $\pi_1(w_1(\cdot))$ are consistent with the colors $c_{i'}$ and $c_{j'}$. R then sends two sets of messages to a commitment challenger: the first consists of the colors for $\pi_0(w_0(k))$ for all $k \in U \setminus \{i', j'\}$, and the second consists of the colors for $\pi_1(w_1(k))$ for all $k \in U \setminus \{i', j'\}$. R generates commitments for $c_{i'}$ and c_j and then uses the commitments received from the commitment challenger for the other vertices, so R does not know whether it is using w_0 or w_1. A then asks to open a specific edge $(i, j) \in E$, and if (i, j) happens to be (i', j'), R opens the colors $c_{i'}, c_{j'}$. Otherwise, R aborts. If R didn't abort, the interaction is now over and A outputs a guess $b^\star$ for whether the witness was w_0 or w_1. R simply forwards this guess to the commitment challenger.

Note that by definition, R only queries A in a single session and only via black-box access. So, we only need to argue that R succeeds with better than $1/2$ probability assuming that A succeeds with $1/2 + \delta$ probability for some inverse polynomial δ. At a high level, this follows since A's view is identical to a random execution with either $P(w_0)$ or $P(w_1)$, assuming that R does not abort. The key point in arguing this is that any $b \in \{0, 1\}$, for any fixed edge (i', j') and fixed witness w_b, there is a 1-1 mapping between colors $c_{i'}$, $c_{j'}$ and permutations π_b over colors, so picking random colors for $c_{i'}$, $c_{j'}$ and computing the corresponding permutation w.r.t. w_b, is equivalent to picking a random permutation.

Next, since R chose (i', j') randomly and independent of A, the probability that R aborts because $(i', j') \neq (i, j)$ is at most $(1 - 1/|E|)$. So with probability $1/|E|$, A's guess at distinguishing w_0 from w_1 corresponds exactly to whether or not the commitment challenger chose the commitments for $\pi_0(w_0(\cdot))$ or $\pi_1(w_1(\cdot))$. It follows that R succeeds at distinguishing these two cases with probability $1/2 + \delta/|E|$. Further, we can do an additional hybrid over each of the elements in the set to distinguish two individual committed values with probability $1/2 + \delta/(|E| \cdot (|U| - 2))$.

For full details of the above high level argument, we refer the reader to the full paper [CFP22]. The main point is that since this new proof is a single-shot, straightline, black-box reduction, it immediately implies a universal reduction from WI to PRG security.

2.3 Impossibility of Hardness Amplification and Goldreich-Levin

Impossibility of Universal Hardness Amplification. (See Theorem 4) We start by giving an overview for why there is no universal black-box reduction for the proof of hardness amplification with black-box access to the function f. Let f be a one-way function, and define the n-fold direct product function $f^{(n)}$ such that $f^{(n)}(x_1, \ldots, x_n) = (f(x_1), \ldots, f(x_n))$. We show that this construction does not increase the security for generic functions f. Specifically, we consider generic security games C^f and $C^{(n),f}$ for the OWF security of an arbitrary function f and its n-fold product $f^{(n)}$. Suppose there exists a reduction R such that for any f and any augmented adversary (A, Nat) with advantage $a(\lambda)$ at inverting $f^{(n)}$, then the augmented adversary $(R^{(A,f)}, \mathsf{Nat})$ inverts f with advantage $\epsilon(\lambda, a)$. In this overview, we show that if $R^{(A,f)}$ only makes black-box use of the function f via oracle access to f, then it must satisfy $\epsilon(\lambda, a) \leq a + \mu(\lambda)$ for $a = 1/e$ and μ a negligible function.

Our high level approach is as follows. We will construct an augmented adversary (A, Nat) that has robust advantage roughly $1/e$, yet the answers by this attacker can be efficiently simulated in PPT. In more detail, consider some reduction $(R^{(A,f)}, \mathsf{Nat})$ that work for any function f. Such a reduction must also work for a random function $f : \{0,1\}^\lambda \to \{0,1\}^{3\lambda}$, and for random functions, we have the advantage that the reduction won't (except with negligible probability) be able to query the attacker on any point in the range of the function unless it has already queries f on the pre-image. So, it would seem that if we use such a random function, then we can easily emulate a perfect inverter (by simply looking at all the queries made by R to f). There is one main obstacle here: R actually gets some value y in the range of f as input (and its goal is to invert this point), and R could of course embed this y into its queries to (A, Nat). We overcome this issue by considering a particular "random-aborting" attacker (A, Nat) that (1) only inverts a $1 - 1/n$ fraction of all values y', and (2) never agrees to invert the same value y' twice. We can show that such an attacker succeeds in robustly inverting $f^{(n)}$ with probability roughly $1/e$. Intuitively, such an attacker "knows" how to invert f with probability $1 - 1/n$, but as we shall see, since (A, Nat) is stateful and never agrees to invert the same value twice we can show that (A, Nat) can only be used to invert f with probability roughly $1/e$. More precisely, we show how to correctly simulate this attacker with probability $1 - 1/e$ by a PPT simulator S that simply aborting whenever we see a query that contains a component y_i for which we do not know a pre-image (through one of the f queries made by R). Thus, if $(R^{(A,f)}, \mathsf{Nat})$ inverts a random function f

with probability $\epsilon(\lambda, a(\lambda))$, it follows that $(R^{(A,f)}, S^f)$ will invert f with probability $\epsilon(\lambda, 1/e) - 1/e - \mathsf{negl}(\lambda)$. Since R, A, and S are efficient, this probability must be bounded by a negligible function, so $\epsilon(\lambda, 1/e) \leq 1/e + \mathsf{negl}(\lambda)$.

Let us proceed to defining the augmented adversary (A, Nat). The augmented adversary (A, Nat) interacts in the OWF security game of $f^{(n)}$, so A receives queries of the form $(y_1, \ldots, y_n)$. A will simply forward these queries to Nat, who responds with either $\perp$ or the correct inverse $(r_1, \ldots, r_n)$, based on the following procedure:

1. For each y_i in the query, if Nat has previously seen a query for y_i in ρ or if y_i is not in the image of f, it sets r_i to be $\perp$.
2. Next, it flips a coin and with probability roughly $1/n$ just sets r_i to be $\perp$
3. If r_i has not been set to $\perp$, Nat sets r_i to be any preimage in $f^{-1}(y_i)$.
4. Finally, if any r_i was set to $\perp$, Nat responds to the entire query with $\perp$. Otherwise, it responds with the inverse $(r_1, \ldots, r_n)$.

We argue that (A, Nat) will invert a *random* challenge $(f(x_1), \ldots, f(x_n))$ with constant probability, *for all possible prefixes* ρ. In particular, a random challenge $(y_1, \ldots, y_n)$ will always have that each y_i is in the image of f. Additionally, no matter what the history is, a random challenge will not collide with any past query with high probability (formally we need to restrict to only looking at the most recent $\lambda^{\log \lambda}$ queries in case ρ has super-polynomial length). So the only reason Nat outputs $\perp$ is if any of its coin flips tell it to set r_i to be $\perp$, but this happens with probability at most $1 - (1 - 1/n)^n \approx 1 - 1/e$. Thus, the augmented adversary (A, Nat) succeeds with probability roughly $1/e$.

We now argue that Nat can be efficiently simulated. The main reason is that because Nat only needs to reply to queries the first time it sees them, we only need to simulate *a single response* for the challenge $y = f(x)$ that the reduction receives. This is much easier than simulating multiple responses that may include y in various ways. Specifically, the simulator S simulates any queries that R makes to either Nat or f, without the use of Nat. Whenever S simulates a query to f, it records the responses before forwarding the reply back to R. To simulate a query $(y_1, \ldots, y_n)$ to Nat, S proceeds exactly as Nat except that it doesn't actually know how to invert f. Namely, it can still reject y_i values it has seen before, and flip a coin to ignore certain inputs. It tries to invert any y_i value it sees by looking at the queries R has made to f, and uses such a value if one exists.

It remains to argue that S diverges from the behavior of Nat with small probability. S diverges whenever R makes a query $(y_1, \ldots, y_n)$ where R has not queried some y_i before, or if y_i has multiple pre-images. But because f is a random function from λ to 3λ bits, the probability R can guess an element in the image of f without querying it is negligible (other than its input $y = f(x)$, and the probability that f is not injective is negligible). Thus, we only need to deal with when it queries $y = f(x)$ *for the first time*. But Nat outputs $\perp$ in that case with probability $\approx 1 - 1/e$, so S and Nat only diverge with probability roughly $1/e$!

Finally, it follows that if R, given access to Nat, inverts a random $(y_1, \ldots, y_n)$ with probability $1/e + 1/p(\lambda)$ for some polynomial p, then R given access to the simulator S will invert a random f with probability at least $1/p(\lambda)$, which is impossible. So $(R^{(A,f)}, \mathsf{Nat})$ must invert f with probability at most $\epsilon(\lambda, a) \leq a + \mu(\lambda)$ for $a = 1/e$ and some negligible function μ.

For the above proof, we note that we crucially rely on the fact that (A, Nat) is an augmented adversary because it only ever inverts individual y_i values that it has never seen before. Let us also point out that by setting the abort probability more carefully, we can make the proof go through also when are required to construct an attacker that succeeds with much higher probability a (and not just $1/e$). A rigorous proof is in the full paper.

Impossibility of a Universal Goldreich-Levin Theorem. (See Theorem 5) We briefly discuss the impossibility of a universal reduction for the Goldreich-Levin theorem. The high level idea and proof structure is similar to the impossibility of hardness amplification.

Recall that the Goldreich-Levin theorem shows that, for any one-way function g, the function $f(x, r) = (g(x), r)$ is a one-way function with hardcore predicate $h(x, r) = \langle x, r \rangle$ for $|x| = |r|$. Let us first outline why the security of the hardcore predicate h cannot be based on the OWF security of g via a universal reduction, when the reduction only has oracle access to the function g.

Similar to the above impossibility for hardness amplification, we construct an augmented adversary (A, Nat) with advantage a where Nat can be efficiently simulated by a machine S for a random function $g\colon \{0,1\}^\lambda \to \{0,1\}^{3\lambda}$. Nat only responds to queries of the form $(g(x), r)$ with the value of $h(x, r)$ (with probability roughly a) once per $g(x)$ value. Then, we construct S that simulates Nat (almost) perfectly except on the first query to the challenge $y = g(x)$ from the OWF challenger. However, since the output of Nat is a single bit, S can just guess what Nat would have output! It follows that S will simulate Nat with roughly $1/2$ probability, so if $(R^{(A,f)}, \mathsf{Nat})$ inverts g with probability $\epsilon(\lambda, a)$, then $(R^{(A,f)}, S^f)$ will do so with probability roughly $\epsilon(\lambda, a)/2$. Since R, A, and S are efficient, this implies that $\epsilon(\lambda, a)$ must be negligible.

Note that we did not use anything about $|r|$ or the structure of h in the above overview. In fact, we rule out any hardcore predicate h for constructions $f(x, r) = (g(x), r)$ for any $|r|$ (even no randomness). See the full paper.

2.4 Universal Reductions for Time-Evolving k-Window Natures, from Classical Non-adaptive Reductions

Let $k(\cdot)$ be any polynomial function. We here argue that if there exists a *non-adaptive*, straightline black-box reduction R from some game C to C', then there exists a universal reduction from C to C' w.r.t. time-evolving k-window Natures (see Theorem 6). For now, we focus on the simplified case where C and C' are 1-round games, but we consider a more general definition of a non-adaptive reductions in the full paper.

Recall that a straightline black-box reduction is one where the reduction R only makes black-box use of a classical, stateless adversary A. We say that such a reduction is non-adaptive if (for 1-round games) the reduction R after receiving a challenge message in C', generates m queries $q_1, \ldots, q_m$ for A in the game C, sends them all at once, receives the responses, and then responds to the challenger C'. Suppose there exists such a reduction R that has advantage ϵ in C' after making m non-adaptive queries to a classical adversary A with advantage a in C. Then for any augmented adversary (B, Nat) with robust advantage a, where Nat is additionally a *time-evolving k-window Nature*, we want to construct an augmented adversary (B', Nat) also with advantage close to ϵ. In particular, for any δ, we will construct B' such that (B', Nat) has robust advantage $\epsilon - \delta$. (This B', however, will have larger running time than R^B, where the running time depends on δ.)

As Nat is a time-evolving k-window Nature, we can essentially think of (B, Nat) as specifying ahead of time a sequence of independent, arbitrary algorithms $S_1, S_2, \ldots$ s.t. it uses S_i to respond to the ith query q_i. We achieve this as follows: in order for B' to be able to emulate such a sequence of attackers $S_1, S_2, \ldots$ using only interactive access to Nat, for each query q_i B' will first send k dummy messages to Nat (in essence resetting its state to be independent of the past, depending only on i). Subsequently, to generate a response for q_i, B' will invoke a fresh copy of B, communicate with Nat on behalf of B, send q_i to B, and reply with B's reply. However, this isn't enough, because each $S_i \in S_1, S_2, \ldots$ may respond differently as i increases (albeit each S_i still wins by robust winning). In other words, the augmented adversary changes over time. To apply the classical non-adaptive reduction R, we must somehow use (B, Nat) to emulate a classical adversary that responds to queries repeatedly according to the same distribution, because R might call its oracle multiple times.

Thus, we construct the universal reduction B' as follows. B' receives some challenge from C' and emulates R on this challenge to generate queries $q_1, \ldots, q_m$. B' then generates $m^2/\delta - m$ extra random "dummy" queries, call them $q_{m+1}, \ldots, q_{m^2/\delta}$. It then samples a random permutation $\pi\colon [m^2/\delta] \to [m^2/\delta]$ that it uses to permute the order of all the queries. For each $i \in [m^2/\delta]$, denote $q'_i = q_{\pi(i)}$. B' then uses $S_1, \ldots, S_{m^2/\delta}$ to respond to those queries, using each S_i to generate a response r'_i for q'_i, in order. It then recovers the responses to the original queries by computing $r_i = r'_{\pi(i)}$ for each $i \in [m]$. R' can feed these to R in order to generate a response for the challenger C'. Importantly, B' is able to emulate $S_1, \ldots, S_{m^2/\delta}$ using a single interaction with the stateful Nat, as long as Nat is a time-evolving (k, μ)-window Nature.

At a high level, the reason the universal reduction B' works is that each response r_i is generated using a random S_j for $j \in [m^2/\delta]$. Thus, R's output should be statistically close to the output of R^A where A is a "classical" adversary A that samples a random $j \leftarrow [m^2/\delta]$ and responds with S_j. However, this isn't the case if there are any collisions on the set of m queries that R queries to this classical adversary A—in other words, if some $j \leftarrow [m^2/\delta]$ is chosen twice—but this bad event can be shown to happen only with probability at most δ. It follows that the output of (B', Nat) is at most δ-far from the output of R^A, so if R wins with probability ϵ, then (B', Nat) will win with probability at least $\epsilon - \delta$.

3 Defining Universal Reductions

In this section, we formally present the notion of a universal reduction. A more in depth study of these notions can be found in the full version [CFP22].

3.1 Preliminaries

We let $\mathbb{N} = \{1, 2, 3, \ldots\}$ denote the set of natural numbers, and for any $n \in \mathbb{N}$, we use $[n] = \{1, \ldots, n\}$ to denote the set from 1 to n.

Throughout, we use $\lambda \in \mathbb{N}$ to denote the security parameter. When we say that an event holds for *sufficiently large* $\lambda \in \mathbb{N}$ we mean that there exists an integer $N \in \mathbb{N}$ such that the event holds for all $\lambda \geq N$. In particular, for any function $f \colon \mathbb{N} \to \mathbb{N}$, the set $O(f)$ consists of all functions g such that there exists a constants c such that $g(\lambda) \leq c \cdot f(\lambda)$ for sufficiently large $\lambda \in \mathbb{N}$. We say that a function $f(\lambda)$ is polynomially-bounded if it is in the set $\lambda^{O(1)} = \text{poly}(\lambda)$. We say that a function $\mu \colon \mathbb{N} \to \mathbb{R}$ is negligible if it is asymptotically smaller than any inverse-polynomial function, so for every constant $c > 0$, $\mu(\lambda) \leq \lambda^{-c}$ for sufficiently large $\lambda \in \mathbb{N}$. In this case, we say $\mu \in \mathsf{negl}(\lambda)$.

We use PPT to denote the acronym *probabilistic, polynomial time.* A uniform algorithm A is a constant-size Turing machine. We say that a function f is efficiently computable if there exists a uniform, polynomial-time algorithm A such that $A(x) = f(x)$ for all $x \in \{0, 1\}^\lambda$. A non-uniform algorithm $A = \{A_\lambda\}_{\lambda \in \mathbb{N}}$ is a sequence of algorithms for all $\lambda \in \mathbb{N}$, and we assume for simplicity that A_λ always receives 1^λ as its first input. A non-uniform PPT algorithm is one where the description size of A_λ is bounded by a polynomial as a function of λ.

An interactive Turing machine (ITM) is an algorithm M that receives and sends messages to other ITMs. For two ITMs, A and B, we denote $\langle A(x), B(y) \rangle(z)$ to denote B's output in the interaction between A and B on private inputs x and y, respectively, and on common input z.

3.2 The Definition and Some Consequences

Towards this, let us first recall the standard notion of a *security game*, wherein an ITM *Challenger* C interacts with an ITM *Adversary* A: On common input 1^λ, C interacts with A until C outputs a bit $b \in \{0, 1\}$. If $b = 1$, we say that the adversary *wins*, and we say that A has advantage a if C outputs 1 with probability at least $a(\lambda)$ for all $\lambda \in \mathbb{N}$. The security game is fully specified by the challenger C, and in the sequel we will use security game and challenger interchangeably.

Whereas classically, the adversary is typically a PPT, or a non-uniform PPT, in our context, we will consider security games with respect to *augmented adversaries*: roughly speaking, a PPT attacker A that has access to some potentially unbounded Nature Nat.

$$\begin{tikzcd} C \arrow[r, leftrightarrow] & A \arrow[r, leftrightarrow] & \mathsf{Nat}(\rho) \end{tikzcd}$$

Fig. 1. *Execution in a nutshell.* The PPT challenger C plays an interactive security game with a PPT attacker A. To help with generating responses, A may send queries to a potentially unbounded Nature machine Nat. Note that Nat may have had previous interactions, which we specify using ρ, which comprises prior messages that Nat may have received, as well as any private coins that Nat may have flipped previously. When we omit ρ, we mean that Nat starts from the blank slate (i.e. no prior messages or coins).

Augmented adversaries. In more detail, an augmented adversary (A, Nat) consists of a PPT ITM A, known as the *attacker*, and a stateful, possibly unbounded non-uniform ITM Nat, known as *Nature*. We think of A as the part of the augmented adversary that only uses "standard" computational resources, whereas Nat is a shared resource in the world that may have "magical" computational resources. Note that since Nat is a non-uniform ITM, it may take a non-uniform advice of arbitrary length. We assume that Nat halts on every input message.

Remark 1. All of our definitions—and proofs—work for more powerful Natures as well, even those that output an *arbitrary probability distribution* in response to any interaction prefix (as opposed to one being samplable by a TM). We define Nat as an ITM for convenience: It becomes easier to specify communication, randomness, views, etc. Furthermore, considering uncomputable Natures gives incomparable results: the feasibility results are stronger, but the impossibility results become weaker.

Interaction model and winning security games (once). We consider executions of a security game C interacting with an augmented adversary (A, Nat). We use $\langle C \leftrightarrow A \leftrightarrow \mathsf{Nat}\rangle(1^\lambda)$ to denote an execution between C, A, and Nat, given the security parameter 1^λ as common input. In particular, the challenger C sends queries to and receives responses from the attacker A, who in turn sends queries to and receives responses from the Nature machine Nat. The execution ends when C halts outputting a bit $b \in \{0, 1\}$ representing the outcome of the security game. An ITM in this model is PPT if there is a polynomial upper bound—as a function of λ—on the number of steps it takes during the lifetime of any execution before halting. Formally, $\langle C \leftrightarrow A \leftrightarrow \mathsf{Nat}\rangle(1^\lambda)$ is a random variable over the joint views of C, A, Nat, where the randomness is over the coins of each party. Given an execution $\mathsf{exec} \in \mathsf{Supp}(\langle C \leftrightarrow A \leftrightarrow \mathsf{Nat}\rangle(1^\lambda))$, we let $\mathsf{out}_C[\mathsf{exec}]$ and $\mathsf{view}_C[\mathsf{exec}]$ denote C's output and view, respectively, in the execution exec.

Definition 4 (Winning Security Games). *Let $a \in [0, 1]$ and $\lambda \in \mathbb{N}$ be a security parameter. We say that an augmented adversary (A, Nat) has* advantage a on λ *for a security game C if*

$$\Pr\left[\mathsf{out}_C[\langle C \leftrightarrow A \leftrightarrow \mathsf{Nat}\rangle(1^\lambda)] = 1\right] \geq a.$$

Let $a : \mathbb{N} \to [0, 1]$. The augmented adversary (A, Nat) has advantage $a(\cdot)$ *for a security game C if for all security parameters $\lambda \in \mathbb{N}$, (A, Nat) has advantage $a(\lambda)$ for C on λ.*

Robust winning. We will also be interested in executions involving Nat where Nat *has already had some prior interaction*; intuitively, we will want to capture a notion of what it means for (A, Nat) to "robustly" win in a security game—roughly speaking, that must (A, Nat) "wins" regardless of any prior interaction that Nat has had with the rest of the world.

We capture this by specifying an *interaction prefix* $\rho = (r, q_1, q_2, \ldots)$ for Nat at the beginning of an execution. We can think of ρ as specifying a finite sequence of queries $q_1, q_2, \ldots$ that Nat previously received, as well as the randomness r that Nat used to respond to those queries; thus ρ fully determines the past behavior and the current state of Nat. For any $\rho \in \{0, 1\}^*$ and security parameter $\lambda \in \mathbb{N}$, consider the interaction where Nat is initialized on input 1^λ, with (read-once) random tape prepopulated by r (followed by 0s), and where Nat is reactivated whenever it becomes idle, s.t. when Nat is activated for the ith time, its message tape is prepopulated with q_i (followed by 0s). Recall that an ITM enters an idle state whenever it is ready to receive the next message in the interaction. When there are no more queries in ρ to process, the random tape of Nat is then reset to uniform randomness. We then let $\mathsf{Nat}(1^\lambda, \rho)$ denote Nat in the state reached following the interaction specified by ρ and 1^λ. Let $\|\rho\|$ denote the number of queries sent to Nat in ρ. Finally, the notation $\langle C \leftrightarrow A \leftrightarrow \mathsf{Nat}(\rho)\rangle(1^\lambda)$ refers to an execution on input 1^λ where Nat starts in the state determined by ρ. If the prefix ρ is omitted, then Nat starts without any prior interaction.

We also define what it means to *concatenate* two prefixes $\rho \circ \rho'$, where $\rho = (r, q_1, q_2, \ldots)$ and $\rho' = (r', q'_1, q'_2, \ldots)$. Define r^* to be the contents of the random tape read by Nat in the interaction $\mathsf{Nat}(1^\lambda, \rho)$, including any 0s if r is too short, or trimming extraneous bits of r that $\mathsf{Nat}(1^\lambda, \rho)$ doesn't read if r is too long. Define $\rho \circ \rho' = (r^* \circ r', q_1, q_2, \ldots, q'_1, q'_2, \ldots)$, where $r^* \circ r'$ denotes string concatenation.

We are now ready to define what it means for an augmented adversary (A, Nat) to *robustly* win in a security game.

Definition 5 (Robust Winning). *Let $a \in [0, 1]$ and $\lambda \in \mathbb{N}$ be a security parameter. We say that an augmented adversary (A, Nat) has* robust advantage *a on λ for a security game C if for all $\rho \in \{0, 1\}^*$, $(A, \mathsf{Nat}(\rho))$ has advantage $a(\lambda)$ on λ for C. Let $a : \mathbb{N} \to [0, 1]$. The augmented adversary (A, Nat) has* robust advantage $a(\cdot)$ *for a security game C if for all $\lambda \in \mathbb{N}$, (A, Nat) has robust advantage $a(\lambda)$ for C on λ.*[11]

[11] In the definition of robust winning above, we require that the augmented adversary win a security game for *every* prefix ρ that Nat may have previously seen, even those containing exponentially many messages. A natural alternative is to consider a notion of robust winning that considers only those prefixes with poly(λ) many messages; indeed our impossibilities and feasibilities can both be made to work in that setting, but at the expense of definitional complexity.

Universal reductions. We finally turn to defining the notion of a universal reduction. Roughly speaking, a universal reduction from security games C to C' guarantees that for every augmented adversary (A, Nat) that robustly wins C, there must exist an attacker A' (depending on A only) such that (A', Nat) robustly wins in C' using the same Nature.

Definition 6 (Universal Reductions). *Let $\epsilon \colon \mathbb{N} \times [0,1] \to [0,1]$, C and C' be security games. We say that there is an ϵ-*universal reduction *from C to C' if for all PPT A there exists a PPT A' such that for every augmented adversary (A, Nat) with robust advantage $a(\cdot)$ for C, (A', Nat) has robust advantage $\epsilon(\cdot, a(\cdot))$ for C'.*

Composability of Universal Reductions. We observe that the definition of a universal reduction easily composes:

Lemma 2 (Composition of Universal Reductions). *Let C_1, C_2, C_3 be security games. Suppose there exists an ϵ_1-universal reduction from C_2 to C_1, and an ϵ_2-universal reduction from C_3 to C_2. Then, there exists an $\epsilon^\star$-universal reduction from C_3 to C_1 where $\epsilon^\star(\lambda, a) = \epsilon_1(\lambda, \epsilon_2(\lambda, a))$ for all $\lambda \in \mathbb{N}$ and $a \in [0,1]$.*

Proof. Let (A_3, Nat) be any augmented adversary, and denote $a(\cdot)$ its robust advantage in C_3. Since there is a ϵ_2-universal reduction from C_3 to C_2, then there exists PPT A_2 s.t. (A_2, Nat) has robust advantage $\epsilon_2(\lambda, a(\lambda))$ in C_2 given security parameter λ for all $\lambda \in \mathbb{N}$. Since there is a ϵ_1-universal reduction from C_2 to C_1, then there must exist PPT A_1 s.t. (A_1, Nat) has robust advantage $\epsilon_1(\lambda, \epsilon_2(\lambda, a(\lambda)))$ in C_1 given security parameter λ for all $\lambda \in \mathbb{N}$.

We conclude that there thus exists a $\epsilon^\star$-universal reduction from C_3 to C_1 where $\epsilon^\star(\lambda, a) = \epsilon_1(\lambda, \epsilon_2(\lambda, a))$ for all $\lambda \in \mathbb{N}$ and $a \in [0,1]$. □

Acknowledgements. This work is supported in part by NSF CNS-2149305, NSF Award SATC-1704788, NSF Award RI-1703846, CNS-2128519, AFOSR Award FA9550-18-1-0267, and a JP Morgan Faculty Award. This material is based upon work supported by DARPA under Agreement No. HR00110C0086. Cody Freitag's work was done partially during an internship at NTT Research, and he is also supported in part by the National Science Foundation Graduate Research Fellowship under Grant No. DGE–2139899. Any opinions, findings and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the NSF, the United States Government, or DARPA.

4 References

AC02. Mark Adcock and Richard Cleve. A quantum goldreich-levin theorem with cryptographic applications. In *STACS*, volume 2285 of *Lecture Notes in Computer Science*, pages 323–334. Springer, 2002.

ARU14. Andris Ambainis, Ansis Rosmanis, and Dominique Unruh. Quantum attacks on classical proof systems: the hardness of quantum rewinding. In *2014 IEEE 55th Annual Symposium on Foundations of Computer Science*, pages 474–483. IEEE, 2014.

BBF13. Paul Baecher, Christina Brzuska, and Marc Fischlin. Notions of black-box reductions, revisited. In *International Conference on the Theory and Application of Cryptology and Information Security*, pages 296–315. Springer, 2013.

BBK22. Nir Bitansky, Zvika Brakerski, and Yael Tauman Kalai. Constructive post-quantum reductions. *IACR Cryptol. ePrint Arch.*, page 298, 2022.

BDF$^+$11. Dan Boneh, Özgür Dagdelen, Marc Fischlin, Anja Lehmann, Christian Schaffner, and Mark Zhandry. Random oracles in a quantum world. In *ASIACRYPT*, volume 7073 of *Lecture Notes in Computer Science*, pages 41–69. Springer, 2011.

BG08. Boaz Barak and Oded Goldreich. Universal arguments and their applications. *SIAM J. Comput.*, 38(5):1661–1694, 2008.

BS20. Nir Bitansky and Omri Shmueli. Post-quantum zero knowledge in constant rounds. In *Proceedings of the 52nd Annual ACM SIGACT Symposium on Theory of Computing*, pages 269–279, 2020.

Can01. Ran Canetti. Universally composable security: A new paradigm for cryptographic protocols. In *Proceedings 42nd IEEE Symposium on Foundations of Computer Science*, pages 136–145. IEEE, 2001.

CFP22. Benjamin Chan, Cody Freitag, and Rafael Pass. Universal reductions: Reductions relative to stateful oracles. `https://ia.cr/2022/156`, 2022.

CMSZ21. Alessandro Chiesa, Fermi Ma, Nicholas Spooner, and Mark Zhandry. Post-quantum succinct arguments: Breaking the quantum rewinding barrier. In *FOCS*, pages 49–58. IEEE, 2021.

Die82. DGBJ Dieks. Communication by epr devices. *Physics Letters A*, 92(6):271–272, 1982.

DN07. Cynthia Dwork and Moni Naor. Zaps and their applications. *SIAM J. Comput.*, 36(6):1513–1543, 2007.

FS90. Uriel Feige and Adi Shamir. Witness indistinguishable and witness hiding protocols. In *Proceedings of the twenty-second annual ACM symposium on Theory of computing*, pages 416–426, 1990.

GGM86. Oded Goldreich, Shafi Goldwasser, and Silvio Micali. How to construct random functions. *J. ACM*, 33(4):792–807, 1986.

GL89. Oded Goldreich and Leonid A Levin. A hard-core predicate for all one-way functions. In *Proceedings of the twenty-first annual ACM symposium on Theory of computing*, pages 25–32, 1989.

GMR89. Shafi Goldwasser, Silvio Micali, and Charles Rackoff. The knowledge complexity of interactive proof systems. *SIAM Journal on computing*, 18(1):186–208, 1989.

GMW91. Oded Goldreich, Silvio Micali, and Avi Wigderson. Proofs that yield nothing but their validity for all languages in NP have zero-knowledge proof systems. *J. ACM*, 38(3):691–729, 1991.

Gol07. Oded Goldreich. *Foundations of cryptography: volume 1, basic tools*. Cambridge university press, 2007.

Gro96. Lov K Grover. A fast quantum mechanical algorithm for database search. In *Proceedings of the twenty-eighth annual ACM symposium on Theory of computing*, pages 212–219, 1996.

Hof11. Dennis Hofheinz. Possibility and impossibility results for selective decommitments. *Journal of Cryptology*, 24(3):470–516, 2011.

IR95. Russell Impagliazzo and Steven Rudich. Limits on the provable consequences of one-way permutations. In *Proceedings of the twenty-first annual ACM symposium on Theory of computing*, pages 44–61, 1989 [1995].

Lam79. Leslie Lamport. Constructing digital signatures from a one-way function. Technical report, 1979.

LTW05. Henry Lin, Luca Trevisan, and Hoeteck Wee. On hardness amplification of one-way functions. In *Theory of Cryptography Conference*, pages 34–49. Springer, 2005.

Mah18. Urmila Mahadev. Classical verification of quantum computations. In *FOCS*, pages 259–267. IEEE Computer Society, 2018.

Mau02. Ueli Maurer. Indistinguishability of random systems. In *International Conference on the Theory and Applications of Cryptographic Techniques*, pages 110–132. Springer, 2002.

Mau11. Ueli Maurer. Constructive cryptography–a new paradigm for security definitions and proofs. In *Joint Workshop on Theory of Security and Applications*, pages 33–56. Springer, 2011.

MP04. Ueli Maurer and Krzysztof Pietrzak. Composition of random systems: When two weak make one strong. In *Theory of Cryptography Conference*, pages 410–427. Springer, 2004.

MPR07. Ueli Maurer, Krzysztof Pietrzak, and Renato Renner. Indistinguishability amplification. In *Annual International Cryptology Conference*, pages 130–149. Springer, 2007.

MR11. Ueli Maurer and Renato Renner. Abstract cryptography. In *In Innovations in Computer Science*. Citeseer, 2011.

Nao91. Moni Naor. Bit commitment using pseudorandomness. *J. Cryptol.*, 4(2):151–158, 1991.

Pop05. Karl Popper. *The logic of scientific discovery*. Routledge, 2005.

Rog06. Phillip Rogaway. Formalizing human ignorance. In *International Conference on Cryptology in Vietnam*, pages 211–228. Springer, 2006.

Sho94. Peter W Shor. Algorithms for quantum computation: discrete logarithms and factoring. In *Proceedings 35th annual symposium on foundations of computer science*, pages 124–134. Ieee, 1994.

Unr12. Dominique Unruh. Quantum proofs of knowledge. In *Annual international conference on the theory and applications of cryptographic techniques*, pages 135–152. Springer, 2012.

Unr16. Dominique Unruh. Computationally binding quantum commitments. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pages 497–527. Springer, 2016.

Wat09. John Watrous. Zero-knowledge against quantum attacks. *SIAM Journal on Computing*, 39(1):25–58, 2009.

WZ82. William K Wootters and Wojciech H Zurek. A single quantum cannot be cloned. *Nature*, 299(5886):802–803, 1982.

Yao82. Andrew C Yao. Theory and application of trapdoor functions. In *23rd Annual Symposium on Foundations of Computer Science (SFCS 1982)*, pages 80–91. IEEE, 1982.

Zha12. Mark Zhandry. How to construct quantum random functions. In *FOCS*, pages 679–687. IEEE Computer Society, 2012.