An Overview of the Hybrid Argument

An Excerpt From: *The Theory of Hash Functions and Random Oracles—An Approach to Modern Cryptography*

Marc Fischlin Arno Mittelbach

Cryptoplexity, Technische Universität Darmstadt, Germany
www.cryptoplexity.de
marc.fischlin@cryptoplexity.de mail@arno-mittelbach.de

Abstract. The hybrid argument is a fundamental and well-established proof technique of modern cryptography for showing the indistinguishability of distributions. As such, its details are often glossed over and phrases along the line of "this can be proven via a standard hybrid argument" are common in the cryptographic literature. Yet, the hybrid argument is not always as straightforward as we make it out to be, but instead comes with its share of intricacies. For example, a commonly stated variant says that if one has a sequence of hybrids $H_0, \ldots, H_t$, and each pair H_i, H_{i+1} is computationally indistinguishable, then so are the extreme hybrids H_0 and H_t. We iterate the fact that, in this form, the statement is only true for constant t, and we translate the common approach for general t into a rigorous statement.

The paper here is not a research paper in the traditional sense. It mainly consists of an excerpt from the book *The Theory of Hash Functions and Random Oracles—An Approach to Modern Cryptography* (Information Security and Cryptography, Springer, 2021), providing a detailed discussion of the intricacies of the hybrid argument that we believe is of interest to the broader cryptographic community. The excerpt is reproduced with permission of Springer.

1 Introduction

The hybrid argument is a fundamental proof technique and at the heart of many important cryptographic results. Not surprisingly, this includes a number of results that we planned on including in our book *The Theory of Hash Functions—An Approach to Modern Cryptograph* [MF21] (henceforth *the hash book*). To our dismay, after carefully reading a first draft of our chapter on pseudorandomness, we found

that the way we had captured the hybrid argument was not quite correct.

1.1 An Erroneous Version of the Hybrid Argument

Our blunder on the presentation of the hybrid argument was a good reminder how easy it is to trip up on formalizing cryptography and how seemingly small details can have devastating effects. Yet, when researching the literature we found that we were in good company. For example, consider the presentation of the hybrid argument as given on Wikipedia (as of 01/01/2021):

> Formally, to show two distributions D_1 and D_2 are computationally indistinguishable, we can define a sequence of hybrid distributions $D_1 := H_0, H_1, \ldots, H_t =: D_2$ where t is polynomial in the security parameter. Define the advantage of any probabilistic efficient (polynomial-bounded time) algorithm A as
>
> $$\mathsf{Adv}^{\mathsf{dist}}_{H_i,H_{i+1}}(\mathbf{A}) := |\Pr[x \leftarrow\!\$\ H_i : \mathbf{A}(x) = 1] - \Pr[x \leftarrow\!\$\ H_{i+1} : \mathbf{A}(x) = 1]|$$
>
> where the dollar symbol ($) denotes that we sample an element from the distribution at random.
>
> By triangle inequality, it is clear that for any probabilistic polynomial-time algorithm A,
>
> $$\mathsf{Adv}^{\mathsf{dist}}_{D_1,D_2}(\mathbf{A}) \leq \sum_{i=0}^{t-1} \mathsf{Adv}^{\mathsf{dist}}_{H_i,H_{i+1}}(\mathbf{A}). \tag{1}$$
>
> Thus there must exist some k s.t. $0 \leq k < t$ and
>
> $$\mathsf{Adv}^{\mathsf{dist}}_{H_k,H_{k+1}}(\mathbf{A}) \geq \mathsf{Adv}^{\mathsf{dist}}_{D_1,D_2}(\mathbf{A})/t. \tag{2}$$
>
> Since t is polynomially bounded, for any such algorithm A, if we can show that its advantage to distinguish the distributions H_i and H_{i+1} is negligible for every i, then it immediately follows that its advantage to distinguish the distributions $D_1 = H_0$ and $D_2 = H_t$ must also be negligible.
>
> (*Wikipedia, Hybrid argument (Cryptography), 01/01/2021*)

While we concur with Equation (2) the final *asymptotic* conclusion that for proving indistinguishability of H_0 and H_t it suffices to show indistinguishability of H_i and H_{i+1} for every i, is utterly wrong[1] if one allows non-constant t. And yet, this

[1] Unless, as discussed later, one interprets the statement as that there is a universal negligible bound for *all* i, in which case we can indeed show the statement.

statement not only appears on Wikipedia, but also in this form in many research papers and cryptographic lecture notes.

Here is a simple counter example to the above asymptotic version of the hybrid argument. Consider the following sequence $(H_i)_{i\in\mathbb{N}}$ of "very deterministic" random variables, defined as

$$H_i(\lambda) = \begin{cases} 1 & \text{if i} \geq \lambda \\ 0 & \text{otherwise.} \end{cases}$$

Individually, each of the random variables H_i eventually becomes 0. In fact, for each i the random variables H_i and H_{i+1} are computationally and even statistically indistinguishable, because for $\lambda \geq i+2$ both variables only return 0. If one now takes $t(\lambda) := \lambda$ in Wikipedia's hybrid argument above, then H_0 and H_t would be indistinguishable as well, although $H_0(\lambda) = 0$ and $H_\lambda(\lambda) = 1$ are trivial to distinguish.[2]

The counter example is easy to generalize from linear functions $t(\lambda) := \lambda$ to arbitrary non-constant functions $t(\lambda) \in \omega(1)$ by letting $H_i(\lambda)$ become 0 if $t(\lambda)$ exceeds i. Then all hybrids H_i at some point are 0 and thus pairwise statistically close, and yet $H_0(\lambda) = 0$ but $H_t(\lambda) = 1$ for sufficiently large λ. This brings up two questions: The first one is about the situation for *constant* t (and here we show that the hybrid argument as above still holds). The other, more important question is then how the hybrid argument should be stated, because this is essential for results to hold.

1.2 Stating the Hybrid Argument Correctly

Let us first investigate what goes wrong with the above version of the hybrid argument. As mentioned before, we are perfectly aligned with the statement above that for the hybrids $D_1 = H_0, H_1, \ldots, H_{t-1}, D_2 = H_t$ there must be some index k between 0 and t such that

$$\mathsf{Adv}^{\mathsf{dist}}_{H_k,H_{k+1}}(\mathbf{A}) \geq \mathsf{Adv}^{\mathsf{dist}}_{D_1,D_2}(\mathbf{A})/t.$$

The intuition tells us now that, for a polynomial t, a non-negligible advantage to distinguish D_1 and D_2 must yield a non-negligible advantage against H_k and H_{k+1}. The problem with this line of reasoning is that this index k may depend on the

[2]Noteworthy, this straightforward counter example already appears on crypto stack exchange (`crypto.stackexchange.com/questions/64110`) for another occasionally encountered false claim in cryptography. This claim is that the polynomial sum $\sum_{i=1}^{t} \epsilon_i(\lambda)$ of negligible functions is again negligible. If we use the 0-1-variables H_i above as probabilities (such that all probabilities eventually become 0 and are thus individually negligible) it still holds that the sum $\sum_{i=1}^{\lambda} H_i(\lambda)$ is constantly 1 and not negligible.

security parameter λ and may thus change with each security parameter. If we now have an infinite number of hybrids, we may spread out the infinite number of "distinguishing indexes" k on the hybrid pairs such that we only have a few indexes k for each pair H_i and H_{i+1}.

To make this tangible, consider once again our running counter example of $H_i(\lambda)$ being 0 in case $\lambda > i$ and 1 otherwise, again with $t(\lambda) := \lambda$. Then we actually know that for each λ the hybrids $H_{t-1}(\lambda) = H_{\lambda-1}(\lambda) = 0$ and $H_t(\lambda) = H_\lambda(\lambda) = 1$ are easy to tell apart for the given parameter λ. But this only holds for the single value λ, and for all other values $\lambda' > \lambda$ the hybrids $H_{\lambda-1}(\lambda') = H_\lambda(\lambda') = 0$ are perfectly indistinguishable.

An immediate consequence from the considerations above, which we also show formally below, is that for constant $t(\lambda)$ the "simple" hybrid argument holds. The reason is that if we have an infinite number of "distinguishing indexes" k we must hit at least one of the constant number of neighbored pairs H_i, H_{i+1} infinitely often, yielding a distinguisher against this pair of hybrids.[3]

Besides the confirmation of the hybrid argument for constant t, and the counter example for non-constant t, we are still left with the question if and how we should put the hybrid argument. We argue below that the hybrid argument also holds as above if we can additionally give a universal negligible bound $\epsilon(\lambda)$ on the distinguishing advantage against any pair of hybrids H_i and H_{i+1}. In this case we would have

$$\epsilon(\lambda) \geq \mathsf{Adv}^{\mathsf{dist}}_{H_k,H_{k+1}}(\mathbf{A}) \geq \mathsf{Adv}^{\mathsf{dist}}_{D_1,D_2}(\mathbf{A})/t(\lambda).$$

and the claim now follows from the fact that $t(\lambda) \cdot \epsilon(\lambda)$ is negligible for polynomial $t(\lambda)$ and negligible $\epsilon(\lambda)$.

Another approach is to state the hybrid argument as it is usually used in proofs, namely, by reducing the indistinguishability of the hybrids $H_0, H_1, \ldots, H_t$ to the indistinguishability of some random variables X and Y. That is, we assume that we have a reduction, or transformation, $T(1^\lambda, i, z)$ taking some index $i \in [0, \ldots, t-1]$ and a sample z, either from $X(1^\lambda)$ or from $Y(1^\lambda)$, but where T is oblivious about z's origin. The transformation T is such that it generates a sample from $H_i(\lambda)$ if z stems from X, and from $H_{i+1}(\lambda)$ if z is from Y. If we can represent the hybrids $H_0, \ldots, H_t$ via such a transformation based on random variables X and Y, and X and Y are

[3] One could now argue that the hybrid argument in general refers to a bound λ' on the number of variables in the polynomial $t(\lambda')$ which is different from the parameter λ in the security game, and in this sense constant. But this is not how one usually conducts the hybrid argument in cryptographic proofs, where the number of hybrids depends on the same parameter λ. For example, consider the proof that an adversary making multiple encryption queries in a CPA-attack cannot gain a significant advantage over the single-query setting, such that the number of queries and therefore the number of hybrids is determined by the adversary and therefore constitutes a polynomial in λ, too.

computationally indistinguishable, then so are H_0 and H_t for any polynomial t. This is the formal statement we prove below (Theorem 3.8). We note that the proof relies on index i being chosen uniformly, such that one can also phrase the hybrid argument with regard to indistinguishability under uniformly chosen index i. Theorem 3.8 captures both variants.

1.3 The Hybrid Argument in the Literature

One finds plenty of examples of careful statements about the hybrid argument in the literature. Alas, at the same time one often also encounters the simplified reasoning, arguing about indistinguishability of neighbored hybrids and concluding that the extreme hybrids must also be indistinguishable. Similarly, one occasionally finds in modern research papers the statement that the polynomial sum of negligible functions is again negligible.[4] We expect these results and hybrid arguments to hold nonetheless, because usually one could state them in the more rigorous form as we have done above.

In research publications the hybrid argument is usually merely sketched. In text books and lecture notes it is often explained by example only, commonly when stretching pseudorandom outputs and arguing the indistinguishability of the resulting generator. This is legit and helps to focus on the important aspects under considerations (such as the pseudorandomness of the stretched output). It nonetheless leaves an uneasy feeling that such a fundamental technique is then often applied in more general contexts without having a precise reference.

For comparison, induction proofs in advanced math courses are, very often, only glanced over as well. Analogously to cryptography, basic math courses often explain the induction proof technique by example, too, without unfolding the full theory of Peano's axiom of induction. Yet, scholars are then usually also exposed to pitfalls when applying the induction technique; one of the most famous alleged induction paradoxes is that *all horses are of the same color*. This cautionary treatment does not seem too common for the hybrid method in cryptography where the idea of the hybrid argument is often given with an explicit or implicit reference to the false version ("show that neighbored hybrids are indistinguishable" instead of "neighbored hybrids are related via a reduction").

Let us list two examples where the hybrid argument has been put carefully. Naturally, our list is not exhaustive and we have probably missed some good examples and references, but the point here is to emphasize that our viewpoint has

[4] Despite the scientific paradigm to support claims by arguments we have decided not to list such works explicitly; the readers may check recent publications and cryptographic lecture notes for such glitches themselves.

been raised in the literature before. The first example is the lecture notes of Boneh and Shoup [BS20, Section 3.4.3] in which they argue—once more by the pseudorandomness example—that the hybrid method needs to be dealt with cautiously. Their treatment of the hybrid argument is bound to the example of pseudorandomness, though. The second example, which comes close to our goal to state a hybrid argument generally and abstractly, is the work by Brzuska et al. [BDF+18] in the eprint version [BDLF+18, Appendix B]. They state the hybrid argument abstractly in a code-based game-playing framework. Our treatment here instead follows the computational setting of random variables. Restricting ourselves to random variables (instead of games) allows for easier access but still suffices for a large number of applications.

1.4 Outline

As remarked earlier, the issues with the hybrid argument are not new and have been pointed out by other researchers before. We do feel, however, that it is important to stress the intricacies of foundational results and, in case of the hybrid argument, stating it as a "helper statement" (rather than explaining it by example) not only helps understanding but also facilitates proofs based on the hybrid argument. In *the hash book* we tried to do just that and Section 3 here is a verbatim excerpt from the book that we hope may serve as a reference for the hybrid argument to benefit both learners and practitioners of cryptography. (And if it spikes interest in the book which has a lot more to offer than just a thorough treatment of the hybrid argument then, admittedly, that would be a welcome side effect.) In Section 2 of the paper we briefly introduce relevant notation, definitions and basic facts which are introduced elsewhere in the book.[5]

2 Preliminaries

We go by the convention that small letters such as i and j denote natural numbers and we denote the security parameter by λ. For $i \in \mathbb{N}$ we denote by $[i]$ the set $\{1, 2, \ldots, i\}$. Indexes are 1-based, unless stated otherwise.

Negligible functions. We begin with a definition of negligible functions $\epsilon : \mathbb{N} \mapsto \mathbb{R}$ for security parameter $\lambda \in \mathbb{N}$ and note that, henceforth, we set polynomials such as p or q in bold face without serifs.

[5] All definitions and statements are taken from the hash book [MF21]. The text is reproduced with the permission of Springer.

Definition 2.1 (Negligible Function)**.** *A function $\epsilon : \mathbb{N} \to \mathbb{R}$ is negligible if for any polynomial $\mathsf{p} : \mathbb{N} \to \mathbb{R}^+$ there exists an integer Λ such that for all $\lambda \geq \Lambda$ we have*

$$\epsilon(\lambda) \leq \frac{1}{\mathsf{p}(\lambda)}.$$

We write $\epsilon(\lambda) = \mathsf{negl}(\lambda)$ for a not further specified negligible function and give the following basic properties without proof.

Theorem 2.2 (Basic Properties of Negligible Functions)**.** *Let $\epsilon, \epsilon' : \mathbb{N} \to \mathbb{R}$ be negligible functions, $\delta : \mathbb{N} \to \mathbb{R}$ be a non-negligible function, and $\mathsf{q} : \mathbb{N} \to \mathbb{R}^+$ be a polynomial. Then the following holds:*

1. *The sum of two negligible functions is negligible:*

$$\epsilon(\lambda) + \epsilon'(\lambda) = \mathsf{negl}(\lambda).$$

2. *A polynomial factor still leaves the function negligible:*

$$\mathsf{q}(\lambda) \cdot \epsilon(\lambda) = \mathsf{negl}(\lambda).$$

3. *Subtracting only a negligible function from a non-negligible one yields a non-negligible function:*

$$\delta(\lambda) - \epsilon(\lambda) \neq \mathsf{negl}(\lambda).$$

4. *A function which is bounded by a negligible function is negligible itself: Let $\gamma : \mathbb{N} \to \mathbb{R}$ be a function such that there exists Λ with $\gamma(\lambda) \leq \epsilon(\lambda)$ for all $\lambda \geq \Lambda$. Then γ is negligible.*

By combining items 1 and 2 we can show that also the sum of *constantly many* negligible functions is negligible.

Proposition 2.3. *Let ϵ_i for $i \in \mathbb{N}$ be a sequence of negligible functions and let $\mathsf{q}_i : \mathbb{N} \to \mathbb{R}^+$ be polynomials. Then for any constant $c \in \mathbb{N}$ the sum is negligible:*

$$\sum_{i=1}^{c} \mathsf{q}_i(\lambda)\epsilon_i(\lambda) = \mathsf{negl}(\lambda).$$

Random variables. Let $X = (X_\lambda)_{\lambda \in \mathbb{N}}$ be a sequence of random variables; one for each security parameter $\lambda \in \mathbb{N}$. Then we write $x \leftarrow\!\!\$\ X(1^\lambda)$ to denote sampling a

value from distribution X_λ. Giving the security parameter λ as input to X, possibly in unary, takes on a more algorithmic standpoint such that we can easily switch back and forth between (efficient) algorithms and distributions. When clear from context we also often simply speak of a random variable X meaning a sequence of random variables $X = (X_\lambda)_{\lambda\in\mathbb{N}}$ indexed by the security parameter. In particular we consider X, Y, and Z to always be sequences of random variables.

Indistinguishability. The hybrid argument is used to show that two distributions (resp. two (sequences of) random variables) are indistinguishable. While the hybrid argument is usually used to argue *computational* indistinguishability we will also use *perfect* and *statistical* indistinguishability, both of which are based on the notion of *statistical distance*.

Definition 2.4 (Statistical Distance)**.** *The* statistical distance *of two sequences of random variables $X = (X_\lambda)_{\lambda\in\mathbb{N}}$ and $Y = (Y_\lambda)_{\lambda\in\mathbb{N}}$ is defined as*

$$\mathsf{SD}_{X,Y}(\lambda) := \frac{1}{2} \cdot \sum_{z\in\{0,1\}^*} \Big|\Pr\Big[X(1^\lambda) = z\Big] - \Pr\Big[Y(1^\lambda) = z\Big]\Big|.$$

With that, we can define the three notions of indistinguishability.[6]

Definition 2.5 (Indistinguishability)**.** *Let $X = (X_\lambda)_{\lambda\in\mathbb{N}}$ and $Y = (Y_\lambda)_{\lambda\in\mathbb{N}}$ be two sequences of random variables.*

Perfectly indistinguishable. *We say that the two random variables are* perfectly indistinguishable*, denoted by $X \stackrel{p}{=} Y$, if for all $\lambda \in \mathbb{N}$ their statistical distance is 0.*

Statistically indistinguishable. *We say that the two random variables are* statistically indistinguishable*, denoted by $X \stackrel{s}{\approx} Y$, if their statistical distance $\mathsf{SD}_{X,Y}(\lambda)$ is negligible.*

Computationally indistinguishable. *We say that the two random variables are* computationally indistinguishable*, denoted by $X \stackrel{c}{\approx} Y$, if for all efficient algorithms $\mathcal{D}$ the advantage*

$$\mathsf{Adv}^{\text{indist}}_{X,Y,\mathcal{D}}(\lambda) := \Big|\Pr\Big[\mathcal{D}(1^\lambda, X(1^\lambda)) = 1\Big] - \Pr\Big[\mathcal{D}(1^\lambda, Y(1^\lambda)) = 1\Big]\Big|$$

is negligible.

[6]Note that our notation for the distinguishing advantage slightly differs from that of the Wikipedia article that we have used in the introduction. We denote the advantage of distinguisher $\mathcal{D}$ distinguishing samples from distributions X and Y by $\mathsf{Adv}^{\text{indist}}_{X,Y,\mathcal{D}}(\lambda)$ which would be $\mathsf{Adv}^{\text{indist}}_{X,Y}(\mathcal{D})$ in Wikipedia's notation.

Finally, we will use the fact that all indistinguishability notions form an equivalence relation.

Proposition 2.6. *Let $X = (X_\lambda)_{\lambda\in\mathbb{N}}, Y = (Y_\lambda)_{\lambda\in\mathbb{N}}$, and $Z = (Z_\lambda)_{\lambda\in\mathbb{N}}$ be sequences of random variables. Then for all above indistinguishability notions $\approx \in \{\stackrel{p}{=}, \stackrel{s}{\approx}, \stackrel{c}{\approx}\}$ it holds that:*

Reflexivity: $X \approx X.$

Symmetry: $X \approx Y \implies Y \approx X.$

Transitivity: $X \approx Y, Y \approx Z \implies X \approx Z.$

3 The Hybrid Argument (Excerpt from The Hash Book)[7]

The hybrid argument is a general technique to show that two distributions of a certain form are (computationally) indistinguishable and is thus often used in game hopping to show that two adjacent games are negligibly close. To motivate the technique we revert to random variables. Since games can be viewed as special cases of random variables our discussion applies to games as well. Assume that we have two random variables X and Y which are computationally indistinguishable. Now consider the t-fold repetition $X_{\times t}$ of X which is the random variable which on input 1^λ outputs the vector $(x_1, \ldots, x_t)$ of t independent samples $x_i \leftarrow\!\!\$\ X(1^\lambda)$ of X. It is convenient to write $X_{\times t} = (X, X, X, \ldots, X)$ with the understanding that each entry corresponds to an independent copy of the random variable X. Define $Y_{\times t}$ analogously. Are these t-fold repetitions also computationally indistinguishable if X and Y are?

The *hybrid method* (or hybrid argument) provides an answer to such questions. The idea is to consider a sequence of hybrid random variables $H^0, \ldots H^t$, such that H^0 corresponds to $X_{\times t}$ and H^t to $Y_{\times t}$, and each transition from H^i to H^{i+1} only corresponds to an indistinguishable change.[8] In other words, the sequence H^0 to H^t causes a gradual shift from $X_{\times t}$ to $Y_{\times t}$ but such that the intermediate steps are not harmful, implying that the two extreme hybrids must be indistinguishable. In the case of the t-fold repetitions the hybrids could, starting from the vector $(X, X, X, \ldots, X)$ of X-samples, step-wise replace one copy of X with Y until all

[7] The excerpt is taken almost verbatim from Section 3.2.2 of the hash book [MF21]. We adjusted cross references to be consistent in this paper. Any other alteration is marked in [brackets] and omissions are marked with an ellipsis [...]. The text is reproduced with the permission of Springer.

[8] While we usually consider indexes to start at 1 it is here convenient to have 0-based indices for hybrids as this simplifies the writing of sums in later analyses.

occurrences have been replaced:

$$\begin{array}{lcl}
H^0 & = & (\ X,\ X,\ X,\ \ldots\ X,\ X\ ) \\
H^1 & = & (\ Y,\ X,\ X,\ \ldots\ X,\ X\ ) \\
H^2 & = & (\ Y,\ Y,\ X,\ \ldots\ X,\ X\ ) \\
 & & \vdots \\
H^{t-1} & = & (\ Y,\ Y,\ Y,\ \ldots\ Y,\ X\ ) \\
H^t & = & (\ Y,\ Y,\ Y,\ \ldots\ Y,\ Y\ )
\end{array}$$

Intuitively, since each pair H^i and H^{i+1} only differs in the $(i+1)$st entry, namely, X or Y, and X and Y are computationally indistinguishable, all hybrids should be close. Our goal is to provide a formal proof for this intuition. To this end, let us first formalize the statement that we want to prove in this section via the hybrid argument.

Lemma 3.1. *Let $X = (X_\lambda)_{\lambda\in\mathbb{N}}$ and $Y = (Y_\lambda)_{\lambda\in\mathbb{N}}$ be two sequences of random variables that are computationally indistinguishable, that is, $X \stackrel{c}{\approx} Y$. Then for any polynomial $\mathsf{t} : \mathbb{N} \to \mathbb{N}$ also the t-fold repetitions*

$$X_{\times\mathsf{t}} := (X_\lambda^1, X_\lambda^2, \ldots, X_\lambda^{\mathsf{t}(\lambda)})_{\lambda\in\mathbb{N}} \quad \text{and} \quad Y_{\times\mathsf{t}} := (Y_\lambda^1, Y_\lambda^2, \ldots, Y_\lambda^{\mathsf{t}(\lambda)})_{\lambda\in\mathbb{N}}$$

are computationally indistinguishable. Here all X_λ^i (resp. Y_λ^i) are independent copies of random variable X_λ (resp. Y_λ).

Note that in Proposition 2.6 we have shown that our indistinguishability notions satisfy the properties of an equivalence relation. While we have shown that transitivity holds, that is, if $X \approx Y$ and $Y \approx Z$, then also $X \approx Z$, we have also mentioned that transitivity only holds for a constant number of steps. This implies that the desired result easily follows for a constant number t of intermediate hybrids. But for cryptographic applications we often make a polynomially number of hops, such that we need to make some additional stipulation on the random variables.

In the following we formalize the *hybrid argument* which extends the constant transitivity of indistinguishability notions to polynomially many steps under certain assumptions. The different requirements for the hybrid argument to work can be subtle, and indeed they are often glossed over in the cryptographic literature. We will here present four versions of the hybrid argument that can be used in different situations. Each version comes with different assumptions and conclusions and there is no strict hierarchy between the various versions.

Remark 3.2. We will present the following sections for computational indistinguishability as this is usually the setup in which the hybrid argument is used. However, all

the results can also be shown for statistical indistinguishability. The proofs will be almost identical except that we consider unbounded algorithms. Furthermore, note that for perfect indistinguishability the transitivity rule can trivially be applied an arbitrary number of times.

3.1 Constant Number of Hybrids

We begin with the simplest form of the hybrid argument for a constant number of hybrids. That is, for a constant $t \in \mathbb{N}$ we consider sequences of random variables (or hybrids) $H^0, H^1, \ldots, H^t$ and show that if any two neighboring hybrids are indistinguishable, then so are hybrids H^0 and H^t. A common example are game-hopping-based proofs where one often has a constant number of game hops in the proof. This result for a constant number of hybrids immediately follows from Proposition 2.6, but it helps to formalize the proof for the upcoming discussions.

Theorem 3.3 (Hybrid Argument for Constant Number of Hybrids)**.** *Let $t \in \mathbb{N}$ be a fixed integer and let $H^0, H^1, \ldots, H^t$ be sequences of random variables (i.e., $H^i = (H^i_\lambda)_{\lambda\in\mathbb{N}}$). Then it holds that*

$$\forall i \in [t-1] : H^i \stackrel{c}{\approx} H^{i+1} \implies H^0 \stackrel{c}{\approx} H^t.$$

Proof. Let us fix an arbitrary algorithm $\mathcal{A}$ that can distinguish distributions H^0 and H^t with advantage $\mathsf{Adv}^{\text{indist}}_{H^0,H^t,\mathcal{A}}(\lambda)$:

$$\mathsf{Adv}^{\text{indist}}_{H^0,H^t,\mathcal{A}}(\lambda) = \Big|\Pr\Big[\mathcal{A}(1^\lambda, H^0(1^\lambda)) = 1\Big] - \Pr\Big[\mathcal{A}(1^\lambda, H^t(1^\lambda)) = 1\Big]\Big|.$$

By "adding the overall term of 0" this can be rewritten in a telescoping sum as follows:

$$\begin{aligned}
= \Big|&\Pr\Big[\mathcal{A}(1^\lambda, H^0(1^\lambda)) = 1\Big] - \Pr\Big[\mathcal{A}(1^\lambda, H^1(1^\lambda)) = 1\Big] \\
&+ \Pr\Big[\mathcal{A}(1^\lambda, H^1(1^\lambda)) = 1\Big] - \Pr\Big[\mathcal{A}(1^\lambda, H^2(1^\lambda)) = 1\Big] \\
&+ \Pr\Big[\mathcal{A}(1^\lambda, X^2(1^\lambda)) = 1\Big] - \Pr\Big[\mathcal{A}(1^\lambda, H^3(1^\lambda)) = 1\Big] \\
&+ \ldots \\
&+ \Pr\Big[\mathcal{A}(1^\lambda, H^{t-1}(1^\lambda)) = 1\Big] - \Pr\Big[\mathcal{A}(1^\lambda, H^t(1^\lambda)) = 1\Big]\Big|.
\end{aligned}$$

More compactly:

$$= \left|\sum_{i=0}^{t-1} \Pr\Big[\mathcal{A}(1^\lambda, H^i(1^\lambda)) = 1\Big] - \Pr\Big[\mathcal{A}(1^\lambda, H^{i+1}(1^\lambda)) = 1\Big]\right|.$$

Finally, applying the triangle inequality $|a+b| \leq |a| + |b|$ we derive

$$\begin{aligned} &\leq \sum_{i=0}^{t-1} \Big| \Pr\Big[\mathcal{A}(1^\lambda, H^i(1^\lambda)) = 1\Big] - \Pr\Big[\mathcal{A}(1^\lambda, H^{i+1}(1^\lambda)) = 1\Big] \Big| \\ &= \sum_{i=0}^{t-1} \mathsf{Adv}^{\text{indist}}_{H^i, H^{i+1}, \mathcal{A}}(\lambda). \end{aligned} \tag{3}$$

By assumption $H^i \stackrel{c}{\approx} H^{i+1}$, and thus the advantages in Equation (3) are negligible for any PPT adversary $\mathcal{A}$. As a constant sum of negligible functions is negligible (see Proposition 2.3 [...]) it follows that also Equation (3) denotes a negligible function, which concludes the proof. □

3.2 Polynomial Number of Hybrids with a Universal Distinguishing Bound

As discussed earlier, a constant number of hybrids is often not sufficient for cryptographic applications. Instead, we usually require a polynomial number of hybrids $H^0, H^1, \ldots, H^{\mathsf{p}(\lambda)}$ for some polynomial p. Note that in this case we consider random variables where both the number of hybrids as well as the hybrids themselves depend on the security parameter. To capture the dependency of the index of the hybrid from the security parameter we usually consider a function $I : \mathbb{N} \to \mathbb{N}$ which maps every parameter λ to the index $i = I(\lambda) \in \{0, 1, \ldots, \mathsf{p}(\lambda)\}$ in question. With this we denote by H^I the hybrid given by $H^I(1^\lambda) := H^{I(\lambda)}(1^\lambda)$ which samples according to the ith hybrid for parameter 1^λ, where $i = I(\lambda)$. We call the function I p-*indexing* if $I(\lambda) \in \{0, 1, \ldots, \mathsf{p}(\lambda) - 1\}$ for all λ. It is sometimes also convenient to write H^{I+1} for the distribution $H^{I(\lambda)+1}(1^\lambda)$ and H^{p} for $H^{\mathsf{p}(\lambda)}(1^\lambda)$.

When attempting to show that $H^i \stackrel{c}{\approx} H^{i+1}$ for all $i \in \{0, 1, \ldots, \mathsf{p}(\lambda) - 1\}$ implies that also $H^0 \stackrel{c}{\approx} H^{\mathsf{p}}$ via the proof for the constant number of hybrids we run into difficulties in the very last step of the proof. Before, in Equation (3), we argued that the sum of a constant number of negligible functions is negligible. But as we have seen in [Section 1] the sum of a polynomial number of negligible functions does not have to be negligible and, consequently, assuming only $H^i \stackrel{c}{\approx} H^{i+1}$ is insufficient for the polynomial case of the hybrid argument.

We can even show that the hybrid argument of Theorem 3.3 in general fails for a polynomial number $\mathsf{p}(\lambda)$ of hybrids. To this end let $\mathsf{p}(\lambda) = \lambda$ be linear and for any $i \in \{0, 1, \ldots, \mathsf{p}(\lambda)\}$ set

$$H^i(1^\lambda) = \begin{cases} 1 & \text{if } i \geq \lambda \\ 0 & \text{else} \end{cases}$$

Then for any given i we have $H^i \overset{c}{\approx} H^{i+1}$ and even $H^i \overset{s}{\approx} H^{i+1}$ since both random variables eventually become 0, i.e., $H^i(1^\lambda) = H^{i+1}(1^\lambda) = 0$ for all $\lambda \geq i+2$. But we now have a linear number of hybrids, too, such that we can still find a non-vanishing hybrid for any given security parameter λ. In fact, if one compares $H^0(1^\lambda) = 0$ and $H^\lambda(1^\lambda) = 1$ then it follows that the hybrids H^0 and H^{p} for the growing $\mathsf{p}(\lambda) = \lambda$ are clearly not indistinguishable, even though any two neighboring hybrids H^i, H^{i+1} are individually close.

In the following we will see three variants of how we can work around the issue. The easiest is to assume that for each adversary $\mathcal{A}$ there exists a single negligible function ϵ that upper bounds the distinguishing advantage for any pair of neighboring hybrids. In this case we can upper bound Equation (3) (translated to polynomially many hybrids) as

$$\mathsf{Adv}^{\text{indist}}_{H^0,H^{\mathsf{p}(\lambda)},\mathcal{A}}(\lambda) \leq \sum_{i=0}^{\mathsf{p}(\lambda)-1} \mathsf{Adv}^{\text{indist}}_{H^i,H^{i+1},\mathcal{A}}(\lambda) \leq \sum_{i=0}^{\mathsf{p}(\lambda)-1} \epsilon(\lambda) = \mathsf{p}(\lambda) \cdot \epsilon(\lambda).$$

Since a negligible function times a polynomial remains negligible (see Theorem 2.2 [...]) we thus obtain a negligible upper bound for advantage $\mathsf{Adv}^{\text{indist}}_{H^0,H^{\mathsf{p}(\lambda)},\mathcal{A}}(\lambda)$. This is formalized in the following theorem:

Theorem 3.4 (Hybrid Argument for Universal Distinguishing Bound)**.** *Let $\mathsf{p} : \mathbb{N} \to \mathbb{N}$ be a polynomial and let $H^0, H^1, H^2, \ldots$ be sequences of random variables (i.e., $H^i = (H^i_\lambda)_{\lambda\in\mathbb{N}}$). If for adversary $\mathcal{A}$ there exists a function ϵ such that there exists an integer $\Lambda \in \mathbb{N}$ such that for all $\lambda \geq \Lambda$ and all p-indexing functions I we have*

$$\mathsf{Adv}^{\text{indist}}_{H^I,H^{I+1},\mathcal{A}}(\lambda) \leq \epsilon(\lambda),$$

then it holds that

$$\mathsf{Adv}^{\text{indist}}_{H^0,H^{\mathsf{p}(\lambda)},\mathcal{A}}(\lambda) \leq \mathsf{p}(\lambda) \cdot \epsilon(\lambda)$$

for all $\lambda \geq \Lambda$. If ϵ is negligible then it follows that H^0 and H^{p} are computationally indistinguishable.

Finding a fixed upper bound for all neighboring hybrids is usually not easy, and Theorem 3.4 gives no indication of how to go about finding such an ϵ. Next, we will discuss sufficient conditions for the hybrid distributions such that we can immediately bound the distinguishing advantage $\mathsf{Adv}^{\text{indist}}_{H^0,H^{\mathsf{p}(\lambda)},\mathcal{A}}(\lambda)$. These will yield the final two variants of the hybrid argument.

3.3 Polynomial Number of Hybrids: Non-uniform Variant

The following variant is based on the non-uniform model of computation [...] in which an algorithm for each input length (and thus for each different security parameter) can have hardwired advice of polynomial length. The idea will be to exploit such advice to find a non-uniform upper bound on the distinguishing advantage of distributions H^0 and H^{p}.

Let us go back to the proof for the case of constantly many hybrids and there to the last step: Equation (3). Translated to polynomially many hybrids here we have that

$$\mathsf{Adv}^{\text{indist}}_{H^0,H^{\mathsf{p}(\lambda)},\mathcal{A}}(\lambda) \leq \sum_{i=0}^{\mathsf{p}(\lambda)-1} \mathsf{Adv}^{\text{indist}}_{H^i,H^{i+1},\mathcal{A}}(\lambda).$$

Now, for each $\lambda \in \mathbb{N}$ there must exist an $i_{\mathsf{max}} \in \{0,1,\ldots,\mathsf{p}(\lambda)-1\}$ that maximizes the distinguishing advantage. This we can capture as a p-indexing function of the security parameter as

$$I_{\mathsf{max}}(\lambda) := \underset{i\in\{0,1,\ldots,\mathsf{p}(\lambda)-1\}}{\arg\max} \mathsf{Adv}^{\text{indist}}_{H^i,H^{i+1},\mathcal{A}}(\lambda)$$

and with this, we can rewrite the above as

$$\begin{aligned}
\mathsf{Adv}^{\text{indist}}_{H^0,H^{\mathsf{p}(\lambda)},\mathcal{A}}(\lambda) &\leq \sum_{i=0}^{\mathsf{p}(\lambda)-1} \mathsf{Adv}^{\text{indist}}_{H^i,H^{i+1},\mathcal{A}}(\lambda) \\
&\leq \sum_{i=0}^{\mathsf{p}(\lambda)-1} \mathsf{Adv}^{\text{indist}}_{H^{I_{\mathsf{max}}},H^{I_{\mathsf{max}}+1},\mathcal{A}}(\lambda) \\
&= \mathsf{p}(\lambda) \cdot \mathsf{Adv}^{\text{indist}}_{H^{I_{\mathsf{max}}},H^{I_{\mathsf{max}}+1},\mathcal{A}}(\lambda). \qquad (4)
\end{aligned}$$

Note that $H^{I_{\mathsf{max}}}$ is a single distribution which, for each security parameter, "picks" a hybrid from the set of hybrids $H^0, H^1, \ldots, H^{\mathsf{p}(\lambda)-1}$. If we could now show that $H^{I_{\mathsf{max}}} \overset{\text{c}}{\approx} H^{I_{\mathsf{max}}+1}$, then we have once more found an upper bound on the distinguishing advantage between distributions H^0 and H^{p}. Showing that $H^{I_{\mathsf{max}}} \overset{\text{c}}{\approx} H^{I_{\mathsf{max}}+1}$ is, however, not straightforward since the helper function I_{max} may not be efficiently computable—it may even be uncomputable. [T]he non-uniform model of computation [and] non-uniform algorithms with polynomial advice may allow for "computing" uncomputable functions. It is that property that we will exploit shortly by hardwiring the function values $I_{\mathsf{max}}(\lambda)$ into a non-uniform distinguisher.

The above discussion tells us where to look for the designated breaking point (via I_{max}), but does not yet give any hint on how to exploit this point. The idea for

using the breaking point will be to take a "reductionist" approach. Usually when one aims to apply the hybrid method the indistinguishability of the extreme hybrids is related to the indistinguishability of two simpler random variables X and Y. A concrete example is our introductory question about the indistinguishability of the t-fold repetitions $X_{\times t}$ and $Y_{\times t}$ of X and Y. The indistinguishability of the hybrids $H^0 = (X, X, \ldots, X)$ and $H^t = (Y, Y, \ldots, Y)$ should somehow follow from $X \overset{c}{\approx} Y$ in the sense that any successful distinguisher against H^0 and H^t should allow us to build a successful distinguisher against X and Y. In our common terminology this is a reduction, or viewed vice versa in light of constructions of random variables, we transform X and Y into the hybrids.

More formally, we start with two random variables X and Y for which we can show that $X \overset{c}{\approx} Y$. In addition we require the existence of an efficient (possibly non-uniform) transformation T such that $T(i, X) \overset{p}{\equiv} H^i$ and $T(i, Y) \overset{p}{\equiv} H^{i+1}$. That is, if T receives the index $i = I(\lambda)$ and is given a sample from $X(1^\lambda)$ it implements the ith hybrid, and if given a sample from $Y(1^\lambda)$ it implements the $(i+1)$st hybrid. Below we use the shorthand $T(I, X)(1^\lambda) := T(1^\lambda, I(\lambda), X(1^\lambda))$.

Note that we require the transformation T to be identically distributed to the hybrids. If we required only statistically indistinguishability, then the theorem would not hold anymore. Namely, consider once more the efficient hybrids $H^i(1^\lambda)$, which are 1 if $i \geq \lambda$ and 0 else. Each hybrid itself is statistically close to the constant 0 function, and any neighboring hybrids are statistically close to each other. For these hybrids the constant transformation $T(i, z) = 0$ would be statistically close to any hybrid (for arbitrary random variables X, Y), and yet the extreme hybrids $H^0(1^\lambda) = 0$ and $H^\lambda(1^\lambda) = 1$ would be easy to distinguish.

Example 3.5. *As an example transformation consider once again our opening example of a t-fold repetition of variables X and Y (page 9) for efficiently sampleable X and Y. Here, the ith hybrid was defined as $(Y_{\times i},\ X_{\times t-i})$ and we can thus define the transformation T as*

$$T(i, z) := (Y_1, Y_2, \ldots, Y_i, z, X_1, X_2, \ldots, X_{t-i-1}),$$

where values Y_j (for $j \in [i]$) and X_k (for $k \in [t-i-1]$) are independent samples of random variables Y and X, respectively. Note that for this we need to assume that distributions X and Y are efficiently sampleable. Now if value z is sampled from $X(1^\lambda)$ then $T(i, z) \overset{p}{\equiv} H^i$ and, similarly, if z is drawn according to $Y(1^\lambda)$ then $T(i, z) \overset{p}{\equiv} H^{i+1}$.

For each $\lambda \in \mathbb{N}$ the value of the p-indexing function $I_{\mathsf{max}}(\lambda)$ is an integer from the set $\{0, 1, \ldots \mathsf{p}(\lambda) - 1\}$ and thus clearly polynomially bounded in its size. If we consider a non-uniform algorithm we can thus embed the function value $I_{\mathsf{max}}(1^\lambda)$ in

its advice for security parameter λ. With this, we can now construct a *non-uniform* adversary $\mathcal{B}$ as follows. Non-uniform adversary $\mathcal{B}$ will have the index function value $I_{\mathsf{max}}(\lambda)$ hardwired in its advice for parameter λ, and we write $\mathcal{B}[I_{\mathsf{max}}]$ to make this explicit. In addition the algorithm $\mathcal{B}$ will use the efficient transformation T and the (efficient) adversary $\mathcal{A}$ (from Equation (4)) in a black-box way such that

$$\mathsf{Adv}^{\text{indist}}_{H^{I_{\mathsf{max}}},H^{I_{\mathsf{max}}+1},\mathcal{A}}(\lambda) = \mathsf{Adv}^{\text{indist}}_{X,Y,\mathcal{B}^{\mathcal{A},T}[I_{\mathsf{max}}]}(\lambda). \tag{5}$$

Since, by assumption, distributions X and Y are computationally indistinguishable we have that the right-hand side is negligible, which yields the desired bound on distributions H^0 and $H^{\mathsf{p}(\lambda)}$:

$$\begin{aligned}\mathsf{Adv}^{\text{indist}}_{H^0,H^{\mathsf{p}(\lambda)},\mathcal{A}}(\lambda) &\leq \mathsf{p}(\lambda) \cdot \mathsf{Adv}^{\text{indist}}_{H^{I_{\mathsf{max}}(\lambda)},H^{I_{\mathsf{max}}(\lambda)+1},\mathcal{A}}(\lambda) \\ &= \mathsf{p}(\lambda) \cdot \mathsf{Adv}^{\text{indist}}_{X,Y,\mathcal{B}^{\mathcal{A},T}[I_{\mathsf{max}}]}(\lambda) \\ &= \mathsf{p}(\lambda) \cdot \mathsf{negl}(\lambda).\end{aligned}$$

It remains to construct efficient non-uniform adversary $\mathcal{B}^{\mathcal{A},T}[I_{\mathsf{max}}]$ which plays to distinguish distributions X and Y internally using transformation T and adversary $\mathcal{A}$ and which has function value $I_{\mathsf{max}}(1^\lambda)$ hardcoded as part of its non-uniform advice. Adversary $\mathcal{B}^{\mathcal{A},T}[I_{\mathsf{max}}]$ gets as input a value z which is either drawn from X or from Y for parameter 1^λ. It proceeds as follows:

$\mathcal{B}^{\mathcal{A},T}[I_{\mathsf{max}}](1^\lambda, z)$

1 : $i \leftarrow I_{\mathsf{max}}(\lambda)$ // Read out value from advice
2 : $z^* \leftarrow\!\$\; T(1^\lambda, i, z)$
3 : $b' \leftarrow\!\$\; \mathcal{A}(1^\lambda, z^*)$
4 : **return** b'

If value z was sampled from distribution $X(1^\lambda)$, then by the definition of transformation T we have that z^* is sampled according to hybrid $H^{I_{\mathsf{max}}(\lambda)}(1^\lambda)$. If instead z came from $Y(1^\lambda)$, then z^* is sampled according to hybrid $H^{I_{\mathsf{max}}(\lambda)+1}(1^\lambda)$. Adversary $\mathcal{B}$ thus perfectly simulates the distinguishing experiment for adversary $\mathcal{A}$, thereby establishing the equality in Equation (5).

Remark 3.6. Using the above variant of the hybrid argument yields a non-uniform reduction, and [...] it is not always clear how to interpret such reductions [see, for example, [KM12, KM13, BL13] or Section 1.3.3.2 of the hash book]. We thus here do not formulate a theorem statement for the non-uniform case, but instead discuss how we can obtain a uniform version of the above.

Remark 3.7. Above we asked from transformation T that it is such that $T(I, X) \stackrel{\text{p}}{\equiv} H^I$ and $T(I, Y) \stackrel{\text{p}}{\equiv} H^{I+1}$ for two distributions X and Y. In fact, it is sufficient that

$$\mathsf{Adv}^{\text{indist}}_{H^{I_{\mathsf{max}}(\lambda)}, H^{I_{\mathsf{max}}(\lambda)+1}, \mathcal{A}}(\lambda) \leq \mathsf{Adv}^{\text{indist}}_{T(I_{\mathsf{max}}(\lambda), X), T(I_{\mathsf{max}}(\lambda), Y), \mathcal{A}}(\lambda)$$

which is necessarily the case if $T(I, X) \stackrel{\text{p}}{\equiv} H^I$ and $T(I, Y) \stackrel{\text{p}}{\equiv} H^{I+1}$. However, exploiting the non-uniformity of $\mathcal{B}$ and thus T we can also derandomize the transformation and fix the best possible coins as part of the advice such as to maximize the distinguishing probability. This technique is also known as *coin fixing*: Instead of using random choices we use a (precomputed) sequence of choices that maximizes the advantage and which is embedded in the non-uniform advice. With this, the non-uniform version of the hybrid argument can be used to, for example, show that the t-fold repetition of random variables X and Y is computationally indistinguishable given that $X \stackrel{\text{c}}{\approx} Y$ even if X and Y are not efficiently sampleable. Here note that in Example 3.5 above we required X and Y to be efficiently sampleable. With coin fixing we could instead define transformation T as

$$T(i, z) := (y_1, y_2, \ldots, y_i, z, x_1, x_2, \ldots, x_{t-i-1}),$$

where y_i and x_i are part of the advice and chosen such that the distinguishing advantage is maximized. However, as mentioned above it is not clear how to interpret such a result in practice, which is why we do not pursue the matter of coin fixing and non-uniform hybrid arguments further.

3.4 Polynomial Number of Hybrids: Uniform Variant

In order to adapt the above non-uniform argument to the uniform case we need slightly stronger requirements, namely that transformation T is *uniform* and efficient (i.e., a PPT algorithm). As before, we will use transformation T together with adversary $\mathcal{A}$ to construct an adversary $\mathcal{B}$ that distinguishes distributions X and Y. Of course, now $\mathcal{B}$ needs to be uniform and we thus can no longer hardcode the index function value $I_{\mathsf{max}}(\lambda)$. Instead of trying to find the two hybrids that maximize the distinguishing advantage of adversary $\mathcal{A}$ for each security parameter, the idea now will be to simply guess the right hybrids. Why and how this works is best seen in the proof of the statement which we will describe comprehensively next.

While specifying a transformation T can in certain cases make proofs simpler, in other cases it is not quite clear how to choose the underlying distributions X and Y. For these cases we show that it also suffices to prove that

$$H^U \stackrel{\text{c}}{\approx} H^{U+1}, \tag{6}$$

where $U(\lambda)$ is a random variable distributed uniformly in $\{0, 1, \ldots, \mathsf{p}(\lambda) - 1\}$. Choosing the hybrids uniformly at random might look strange at first. Note however, that Equation (6) simply translates to advantage

$$\mathsf{Adv}^{\text{indist}}_{H^{U(\lambda)}, H^{U(\lambda)+1}, \mathcal{A}}(\lambda) := \Big| \Pr\Big[\mathcal{A}(1^\lambda, H^{U(\lambda)}(1^\lambda)) = 1\Big] - \Pr\Big[\mathcal{A}(1^\lambda, H^{U(\lambda)+1}(1^\lambda)) = 1\Big]\Big|,$$

being negligible for all efficient adversaries $\mathcal{A}$, where $U+1$ describes the (dependent) random variable which outputs the same as U, incremented by 1. Here, adversary $\mathcal{A}$ gets as input a sample chosen as follows: First we choose $i \leftarrow\!\!\$\ \{0, 1, \ldots, \mathsf{p}(\lambda) - 1\}$ to then sample a value from hybrid $H^i(1^\lambda)$ on the left or a value $H^{i+1(1^\lambda)}$ on the right. If we can show that the advantage is negligible for any efficient adversary $\mathcal{A}$ then we can also argue that distributions H^0 and H^{p} are computationally indistinguishable. This is formalized as item 2 of the following theorem. For item 1 recall that a function $I : \mathbb{N} \mapsto \mathbb{N}$ is p-indexing if $I(\lambda) \in \{0, 1, \ldots, \mathsf{p}(\lambda) - 1\}$ for all λ.

Theorem 3.8 (The Hybrid Argument (uniform case))**.**
Let $\mathsf{p} : \mathbb{N} \to \mathbb{N}$ be a polynomial and let $H^0, H^1, H^2, \ldots$ be sequences of random variables (i.e., $H^i = (H^i_\lambda)_{\lambda \in \mathbb{N}}$).

1. *Assume that there exist random variables X and Y with $X \overset{c}{\approx} Y$. Assume further that for any PPT algorithm $\mathcal{A}$ there exists a PPT algorithm T such that $T(I, X) \overset{p}{=} H^I$ and $T(I, Y) \overset{p}{=} H^{I+1}$ for all p-indexing functions I. Then H^0 and H^{p} are computationally indistinguishable. In particular, for any PPT distinguisher $\mathcal{A}$ there exists a PPT distinguisher $\mathcal{B}$ with*
$$\mathsf{Adv}^{\text{indist}}_{H^0, H^{\mathsf{p}(\lambda)}, \mathcal{A}}(\lambda) \leq \mathsf{p}(\lambda) \cdot \mathsf{Adv}^{\text{indist}}_{X, Y, \mathcal{B}}(\lambda).$$

2. *Let $U(\lambda)$ denote the random variable distributed uniformly in $\{0, 1, \ldots, \mathsf{p}(\lambda) - 1\}$. Then, for any PPT distinguisher $\mathcal{A}$ it holds that*
$$\mathsf{Adv}^{\text{indist}}_{H^0, H^{\mathsf{p}(\lambda)}, \mathcal{A}}(\lambda) \leq \mathsf{p}(\lambda) \cdot \mathsf{Adv}^{\text{indist}}_{H^{U(\lambda)}, H^{U(\lambda)+1}, \mathcal{A}}(\lambda).$$

Remark 3.9. Note that the transformation T in the first item may depend on algorithm $\mathcal{A}$. [...]

Remark 3.10. The statement in the second point provides the hybrid samples as input to the adversary. [One can consider a more general version of this claim] where the hybrids are given as oracles and the adversary may interact with the corresponding hybrids. Inspecting the proof below one sees that this version follows the same line of argument.

Proof of Theorem 3.8. We will first prove item 1, the hybrid argument with a transformation T. The first part of the proof is identical to the proof for constantly many hybrids (Theorem 3.3), and we here thus only provide the abbreviated version. Let us fix an arbitrary algorithm $\mathcal{A}$ that can distinguish distributions H^0 and H^{p} with advantage $\mathsf{Adv}^{\text{indist}}_{H^0,H^{\mathsf{p}(\lambda)},\mathcal{A}}(\lambda)$:

$$\mathsf{Adv}^{\text{indist}}_{H^0,H^{\mathsf{p}(\lambda)},\mathcal{A}}(\lambda) = \Big|\Pr\Big[\mathcal{A}(1^\lambda, H^0(1^\lambda)) = 1\Big] - \Pr\Big[\mathcal{A}(1^\lambda, H^{\mathsf{p}(\lambda)}(1^\lambda)) = 1\Big]\Big|.$$

By again "adding the overall term of 0" this can be rewritten as

$$= \left|\sum_{i=0}^{\mathsf{p}(\lambda)-1} \Pr\Big[\mathcal{A}(1^\lambda, H^i(1^\lambda)) = 1\Big] - \Pr\Big[\mathcal{A}(1^\lambda, H^{i+1}(1^\lambda)) = 1\Big]\right|. \tag{7}$$

We next build our algorithm $\mathcal{B}^{\mathcal{A},T}$ which tries to distinguish inputs z produced either from random variable X or from random variable Y. Algorithm $\mathcal{B}^{\mathcal{A},T}$ internally uses adversary $\mathcal{A}$ and transformation T (which in turn may depend on $\mathcal{A}$) as follows:

$\mathcal{B}(1^\lambda, z)$

1 : $i \leftarrow\!\!\$\ \{0, 1, \ldots, \mathsf{p}(\lambda) - 1\}$
2 : $z' \leftarrow\!\!\$\ T(1^\lambda, i, z)$
3 : $b' \leftarrow\!\!\$\ \mathcal{A}(1^\lambda, z')$
4 : **return** b'

Assume for the moment that i has been chosen already in line 1 and is fixed. If value z is a sample from $X(1^\lambda)$ then $z' \leftarrow\!\!\$\ T(1^\lambda, i, z)$ corresponds to a random sample from $T(i, X)$ for parameter 1^λ and is thus distributed as $H^i(1^\lambda)$ by assumption about T. It follows that $\mathcal{B}$'s output in this case has the same distribution as $\mathcal{A}(1^\lambda, H^i(1^\lambda))$. If, on the other hand, for the same i the input z stems from $Y(1^\lambda)$, then $T(i, z)$ is a random sample according to $T(i, Y)$ and thus distributed as a sample from $H^{i+1}(1^\lambda)$. In this case $\mathcal{B}$'s output has the same distribution as $\mathcal{A}(1^\lambda, H^{i+1}(1^\lambda))$. Hence we have

$$\begin{aligned}
&\Big|\Pr\Big[\mathcal{B}(1^\lambda, X(1^\lambda)) = 1\Big] - \Pr\Big[\mathcal{B}(1^\lambda, Y(1^\lambda)) = 1\Big]\Big| \\
&\quad = \left|\sum_{j=0}^{\mathsf{p}(\lambda)-1} \Big(\Pr\Big[\mathcal{B}(1^\lambda, X(1^\lambda)) = 1 \wedge i = j\Big] - \Pr\Big[\mathcal{B}(1^\lambda, Y(1^\lambda)) = 1 \wedge i = j\Big]\right| \\
&\quad = \left|\sum_{j=0}^{\mathsf{p}(\lambda)-1} \Pr[i = j] \cdot \Big(\Pr\Big[\mathcal{B}(1^\lambda, X(1^\lambda)) = 1 \;\Big|\; i = j\Big] \right. \\
&\qquad\qquad \left. - \Pr\Big[\mathcal{B}(1^\lambda, Y(1^\lambda)) = 1 \;\Big|\; i = j\Big]\Big)\right|.
\end{aligned}$$

Here the first equality is due to marginalizing over all possible choices of index $i \in \{0, 1, \ldots, \mathsf{p}(\lambda) - 1\}$ and the second is due to rewriting the probabilities as conditional probabilities. By noting that $\Pr[i = j] = \frac{1}{\mathsf{p}(\lambda)}$ for all values of $j \in \{0, 1, \ldots, \mathsf{p}(\lambda) - 1\}$ and that, furthermore, by definition of algorithm $\mathcal{B}$ we have that $\Pr\Big[\mathcal{B}(1^\lambda, X(1^\lambda)) = 1 \;\Big|\; i = j\Big] = \Pr\Big[\mathcal{A}(1^\lambda, H^j(1^\lambda)) = 1\Big]$, we can rewrite the above as

$$= \frac{1}{\mathsf{p}(\lambda)} \cdot \left| \sum_{j=0}^{\mathsf{p}(\lambda)-1} \Pr\Big[\mathcal{A}(1^\lambda, H^j(1^\lambda)) = 1\Big] - \Pr\Big[\mathcal{A}(1^\lambda, H^{j+1}(1^\lambda)) = 1\Big] \right|.$$

And, finally, using the presentation as a telescoping sum, we get

$$= \frac{1}{\mathsf{p}(\lambda)} \cdot \Big|\Pr\Big[\mathcal{A}(1^\lambda, H^0(1^\lambda)) = 1\Big] - \Pr\Big[\mathcal{A}(1^\lambda, H^{\mathsf{p}(\lambda)}(1^\lambda)) = 1\Big]\Big|.$$

Note that in case $\mathcal{A}$ is efficient then $\mathcal{B}$ only performs efficient steps, because transformation T is PPT by assumption and can be incorporated into the code of $\mathcal{B}$. Hence, we derive that the advantage of $\mathcal{A}$ against H^0 and $H^{\mathsf{p}(\lambda)}$ is at most $\mathsf{p}(\lambda)$ times the advantage of $\mathcal{B}$ against X and Y. Since this advantage is negligible by assumption so must be the advantage of $\mathcal{A}$. This concludes the proof for item 1.

Proof of item 2. For item 2 (the hybrid argument without a transformation T) we start once more with an arbitrary algorithm $\mathcal{A}$ and rewrite its distinguishing advantage for hybrids H^0 and $H^{\mathsf{p}(\lambda)}$ as in Equation (7) above:

$$\mathsf{Adv}^{\text{indist}}_{H^0, H^{\mathsf{p}(\lambda)}, \mathcal{A}}(\lambda) = \Big|\Pr\Big[\mathcal{A}(1^\lambda, H^0(1^\lambda)) = 1\Big] - \Pr\Big[\mathcal{A}(1^\lambda, H^{\mathsf{p}(\lambda)}(1^\lambda)) = 1\Big]\Big|,$$

which can be rewritten in a telescoping sum as

$$= \left| \sum_{i=0}^{\mathsf{p}(\lambda)-1} \Pr\Big[\mathcal{A}(1^\lambda, H^i(1^\lambda)) = 1\Big] - \Pr\Big[\mathcal{A}(1^\lambda, H^{i+1}(1^\lambda)) = 1\Big] \right|.$$

Noting that we can replace index i with random variable $U(\lambda)$ and conditioning on $U(\lambda) = i$, we get

$$= \left| \sum_{i=0}^{\mathsf{p}(\lambda)-1} \Pr\Big[\mathcal{A}(1^\lambda, H^{U(\lambda)}(1^\lambda)) = 1 \;\Big|\; U(\lambda) = i\Big] \right.$$
$$\left. - \Pr\Big[\mathcal{A}(1^\lambda, H^{U(\lambda)+1}(1^\lambda)) = 1 \;\Big|\; U(\lambda) = i\Big] \right|.$$

Finally, with the definition of conditional probabilities ($\Pr[A \mid B] = \frac{\Pr[A \wedge B]}{\Pr[B]}$) and noting that, furthermore, for any $i \in \{0, 1, \ldots, \mathsf{p}(\lambda) - 1\}$ we have that $\Pr[U(\lambda) = i] = \frac{1}{\mathsf{p}(\lambda)}$, the above can be rewritten to yield the desired result:

$$\begin{aligned}
&= \mathsf{p}(\lambda) \cdot \Bigg| \sum_{i=0}^{\mathsf{p}(\lambda)-1} \Pr\Big[\mathcal{A}(1^\lambda, H^{U(\lambda)}(1^\lambda)) = 1 \wedge U(\lambda) = i\Big] \\
&\qquad\qquad - \Pr\Big[\mathcal{A}(1^\lambda, H^{U(\lambda)+1}(1^\lambda)) = 1 \wedge U(\lambda) = i\Big]\Bigg| \\
&= \mathsf{p}(\lambda) \cdot \Big|\Pr\Big[\mathcal{A}(1^\lambda, H^{U(\lambda)}(1^\lambda)) = 1\Big] - \Pr\Big[\mathcal{A}(1^\lambda, H^{U(\lambda)+1}(1^\lambda)) = 1\Big]\Big| \\
&= \mathsf{p}(\lambda) \cdot \mathsf{Adv}^{\mathrm{indist}}_{H^{U(\lambda)}, H^{U(\lambda)+1}, \mathcal{A}}(\lambda).
\end{aligned}$$

Here the second equality is a simple "demarginalization"[...]. This concludes the proof of item 2 of the theorem. □

Remark 3.11. It is instructive to think about where the above proof fails when we consider super-polynomially many hybrids. Let $\mathsf{e} : \mathbb{N} \to \mathbb{N}$ be super-polynomial, that is, a function such that for all $c \in \mathbb{N}$

$$\mathsf{e}(\lambda) \in \omega(\lambda^c).$$

Function e could for example be an exponential function such as 2^λ. Now, if e is a super-polynomial then

$$\frac{1}{\mathsf{e}(\lambda)} \cdot \mathsf{Adv}^{\mathrm{indist}}_{H^0, H^{\mathsf{e}(\lambda)}, \mathcal{A}}(\lambda)$$

may be negligible even if $\mathsf{Adv}^{\mathrm{indist}}_{H^0, H^{\mathsf{e}(\lambda)}, \mathcal{A}}(\lambda)$ is non-negligible. Thus the argument no longer works.

3.5 Applying the Hybrid Argument

This concludes our discussion of hybrid arguments. To summarize we recall the steps required in order to show that two distributions A and B are computationally indistinguishable using Theorem 3.8. In a first step we need to come up with a path $H^0, H^1, \ldots, H^{\mathsf{p}(\lambda)}$ of at most polynomially many hybrids (intermediate distributions) that form a path from A to B. These should be such that the first hybrid matches distribution A and the last hybrid matches distribution B, i.e., $H^0 \stackrel{\mathrm{p}}{=} A$ and $H^{\mathsf{p}} \stackrel{\mathrm{p}}{=} B$. For the second step we now have two possibilities. Either we can show that hybrids H^i and H^{i+1} are computationally indistinguishable for uniformly random i (this corresponds to item 2 of the theorem), or alternatively, we can relate the hybrids to two computationally indistinguishable distributions X and Y via some transformation T such that $T(i, X) \stackrel{\mathrm{p}}{=} H^i$ and $T(i, Y) \stackrel{\mathrm{p}}{=} H^{i+1}$.

An example of how Theorem 3.8 can be applied was already given as Example 3.5 above where we considered the t-fold repetition $X_{\times t}$ and $Y_{\times t}$. Here $X_{\times t}$ and $Y_{\times t}$ are the two distributions A and B and the computationally indistinguishable distributions that form the basis of transformation T are the singular versions X and Y. [...]

References

[BDF+18] Chris Brzuska, Antoine Delignat-Lavaud, Cédric Fournet, Konrad Kohbrok, and Markulf Kohlweiss. State separation for code-based game-playing proofs. In Thomas Peyrin and Steven Galbraith, editors, *Advances in Cryptology – ASIACRYPT 2018, Part III*, volume 11274 of *Lecture Notes in Computer Science*, pages 222–249, Brisbane, Queensland, Australia, December 2–6, 2018. Springer, Heidelberg, Germany. (Cited on page 6.)

[BDLF+18] Chris Brzuska, Antoine Delignat-Lavaud, Cedric Fournet, Konrad Kohbrok, and Markulf Kohlweiss. State separation for code-based game-playing proofs. Cryptology ePrint Archive, Report 2018/306, 2018. `https://eprint.iacr.org/2018/306`. (Cited on page 6.)

[BL13] Daniel J. Bernstein and Tanja Lange. Non-uniform cracks in the concrete: The power of free precomputation. In Kazue Sako and Palash Sarkar, editors, *Advances in Cryptology – ASIACRYPT 2013, Part II*, volume 8270 of *Lecture Notes in Computer Science*, pages 321–340, Bengalore, India, December 1–5, 2013. Springer, Heidelberg, Germany. (Cited on page 16.)

[BS20] Dan Boneh and Victor Shoup. A graduate course in applied cryptography, version 0.5, January 2020. `https://toc.cryptobook.us/`. (Cited on page 6.)

[KM12] Neal Koblitz and Alfred Menezes. Another look at non-uniformity. Cryptology ePrint Archive, Report 2012/359, 2012. `http://eprint.iacr.org/2012/359`. (Cited on page 16.)

[KM13] Neal Koblitz and Alfred Menezes. Another look at non-uniformity. *Groups Complexity Cryptology*, 5(2):117–139, 2013. (Cited on page 16.)

[MF21] Arno Mittelbach and Marc Fischlin. *The Theory of Hash Functions and Random Oracles—An Approach to Modern Cryptography*. Information Security and Cryptography. Springer International Publishing, 2021. See also `https://hash-book.info`. (Cited on pages 1, 6, and 9.)