Adaptive Distributional Security
A Framework for Input-Adaptive Cryptography*

Cruz Barnum
University of Illinois Urbana-Champaign
cruzjb2@illinois.edu

David Heath
University of Illinois Urbana-Champaign
daheath@illinois.edu

Abstract

It is often desirable to break cryptographic primitives into two components: an input-independent *offline* component, and a cheap *online* component used when inputs arrive. Security of such online/offline primitives must be proved in the *input-adaptive* setting: the adversary chooses its input adaptively, based on what it sees in the offline-phase. Proving security in the input-adaptive setting can be difficult, particularly when one wishes to achieve simulation security and avoid idealized objects like a random oracle (RO).

This work proposes a framework for reasoning about input-adaptive primitives: *adaptive distributional security* (ADS). Roughly, an ADS primitive provides security when it is used with inputs drawn from one of two distributions that are themselves hard to distinguish. ADS is useful as a framework for the following reasons:

- An ADS definition can often circumvent impossibility results imposed on the corresponding simulation-based definition. This allows us to decrease the online-cost of primitives, albeit by using a weaker notion of security.
- With care, one can typically upgrade an ADS-secure object into a simulation-secure object (by increasing cost in the online-phase).
- ADS is robust, in the sense that (1) it enables a form of composition and (2) interesting ADS primitives are highly interconnected in terms of which objects imply which other objects.
- Many useful ADS-secure objects are plausibly secure from straightforward symmetric-key cryptography.

We start by defining the notion of an ADS encryption (ADE) scheme. A notion of input-adaptive encryption can be easily achieved from RO, and the ADE definition can be understood as capturing the concrete property provided by RO that is sufficient to achieve input-adaptivity. From there, we use ADE to achieve ADS variants of garbled circuits and oblivious transfer, to achieve simulation-secure garbled circuits, oblivious transfer, and two-party computation, and prove interconnectedness of these primitives. In sum, this results in a family of objects with extremely cheap online-cost.

Keywords: Input-Adaptivity; Simulators; MPC; Garbled Circuits

*This work is inspired by a question asked by Prof. Adam Groce at Reed College about whether simulation-based security is too strong in some settings, leading to artifacts in protocol design [A. Groce, Personal Communication, Spring 2020].

Contents

1 Introduction

The Input-Adaptive Setting. Many cryptographic primitives/protocols can be made more flexible by splitting execution into two phases: an *offline-phase*, which runs before party inputs are known, and an *online-phase*. Often, it is possible to move most protocol cost to the offline-phase, resulting in a highly efficient online-phase. This can be desirable, as it allows the parties to complete most of the work when their computational resources are free. Once the input becomes available, which might happen when computational resources are in high demand, they complete the protocol in a timely manner.

While it can be desirable to split a protocol up this way, proving the split secure can be problematic. The challenge is that the adversary can base its protocol input on the offline-phase's protocol transcript. We refer to this general setting as the *input-adaptive* setting: the adversary adaptively chooses its input, based on the offline-phase. This setting presents a challenge to achieving provable security as, for example, there is no way to force the adversary to commit to its input before the offline-phase begins—that input is simply not yet defined. Because of this, it can be challenging to show that input-adaptivity does not give the adversary noticeable advantage. This challenge is especially apparent when trying to prove an offline/online primitive satisfies strong security properties, such as simulation-based security.

Proving that a primitive achieves a notion of simulation security is highly desirable, as it is well known that such notions are composable, allowing to quickly and modularly construct sophisticated protocols. But in the input-adaptive setting it can be challenging to build a simulator, as the simulator must emulate adversarial behavior, and, very roughly, it is hard to perform this emulation as the simulator must commit to an offline-phase transcript that will (from the adversary's point of view) correspond to an input that has not yet been chosen.

An Example Primitive: Garbling Schemes. Consider the well-studied problem of adaptively secure garbled circuit (GC) schemes [BHR12b]. GC is a powerful primitive allowing a garbler to construct from a circuit C a privacy-preserving encoding $\tilde{C}$ [Yao86]. Given $\tilde{C}$ and an encoded input $\tilde{x}$, an evaluator can obtain $C(x)$ while learning nothing beyond the output. Specifically, the cleartext input x is kept private. GC naturally enables round-efficient two-party computation (2PC) protocols.

To improve online efficiency, one might consider sending $\tilde{C}$ in advance, before the input x is known. Then, in the online-phase, the parties seem to only need to exchange an input encoding $\tilde{x}$. This proposed construction could reduce online communication such that it is independent of the circuit size, and even of the circuit's output size, a clear instance of the advantage of input-adaptivity.

Unfortunately, building a garbling scheme that is secure in the input-adaptive setting is notoriously challenging. In fact, if we consider the above attempt — where online communication is independent of the output size — proving simulation security is not just challenging, it is *impossible*. Specifically, any simulation-secure adaptive garbling scheme must have communication that scales with the size of the output [AIKW13]. Notably, this kind of impossibility result — where online-phase cost must scale with output size — is somewhat ubiquitous in the input-adaptive setting. For instance, a similar restriction also applies to an input-adaptive variant of simulation-secure 2PC [HW15].

Returning to our example of adaptive garbling, we note that even when one adjusts their garbling scheme to avoid this impossibility result, achieving input-adaptive security remains a challenge.

Input-Adaptivity and the ROM. Interestingly, the challenge of input-adaptivity can often be essentially eliminated by considering security in an idealized model, such as the random oracle (RO) model (ROM). Roughly, when working in the ROM, we can adjust the offline-phase such that particular messages are encrypted with the output of RO. This adjustment allows simulation security, as the simulator can program the RO to adjust the offline-phase transcript after the adversary chooses its input. This strategy is in particular known to work for garbling schemes [BHR12a].

The strategy of encrypting with RO is troubling, in particular because it can circumvent the kinds of impossibility results discussed above. Note that if one limits the flexibility of the ROM by preventing the simulator from programming the RO (known as the non-programmable ROM (NPROM) [FLR+10]), im-

possibility results once again apply [AIKW13, BHKO23]. However, even when restricting to the NPROM, there still seems to be a very significant gap between what we can achieve from RO and what we can achieve from standard model assumptions. As an example, the standard model adaptive garbling schemes rely either on sophisticated and asymptotically expensive (in the online-phase) tricks [HJO+16], or on public-key cryptographic assumptions [GS18, GOS18], which are not needed in the non-adaptive setting; in contrast, many state-of-the-art garbling schemes that were designed for the non-adaptive setting are *automatically* secure in the adaptive setting when the scheme's underlying encryption scheme is treated as an NPRO [BHKO23, GYW+23].

Our goals in this work. Based on the above discussion, there are two high-level questions that we wish to address:

- Simulation-secure input-adaptive primitives generally require an online-phase that scales with output size. *When can we achieve input-adaptive primitives whose online-cost does not scale with the output size?*
- There is a significant efficiency gap between standard model-based input-adaptive primitives and corresponding primitives built from RO, even when we restrict use of RO, e.g. via the NPROM. *What concrete property does RO provide that makes achieving input-adaptive security so much easier?*

1.1 Our Contribution

In seeking to answer the above questions, we propose a security framework that captures much of the nuance of input-adaptivity. We refer to this framework as the *adaptive distributional (AD) security (ADS)* framework.

ADS vs Simulation-Security. ADS-based security definitions tend to avoid impossibility results associated with simulation security. For instance, our notion of adaptive-distributional GC (ADGC) circumvents the need for the online-phase to scale with output size. This is because the notion of ADS is weaker than that of simulation-based security.

Despite this relative weakness, we show examples (Theorems 11 and 12) where an ADS-based object can be used as a black-box to construct an analogous object that attains adaptive simulation-based security and (within constant factors of) optimal online-cost. Thus, ADS as a framework allows one to plug together protocols, each of which do not scale in the online-phase with the size of their output, then to upgrade the result with simulation security by paying for the output size of the outermost protocol only.

In other words, the ADS framework *is not* meant as an *alternative* to simulation security, but rather as a framework for defining primitives *for the purpose of achieving* efficient adaptive simulation-based security.

Encryption Definition for Input-Adaptivity. As discussed above, many input-adaptive problems become far easier in the ROM. One attractive property of such RO-based solutions is that they achieve a balance between security and practicality, as we can give fast candidate constructions to instantiate the RO (such as SHA or, in certain instances, AES). However, it is well known that RO — programmable or not — is uninstantiable [CGH98], which calls into question the security of such schemes.

It would be far more desirable to find a falsifiable complexity assumption sufficient to prove security of particular input-adaptive solutions. Such an assumption might be plausibly instantiated in the same way as RO, but allowing us far greater confidence that the resulting system may indeed be secure. Returning to our example of garbled circuits, an analogous approach was taken with respect to the popular Free XOR optimization [KS08]. While standard garbling can be proven secure given only a one-way function [LP09], the Free XOR optimization originally leveraged RO. Follow-on work defined a concrete property of a hash function called *circular correlation robustness* (CCR) that is sufficient to prove security of the Free XOR technique [CKKZ12]. This CCR definition is trivially achieved by RO, so there is no change in efficiency, and its introduction increases confidence that Free XOR is plausibly secure.

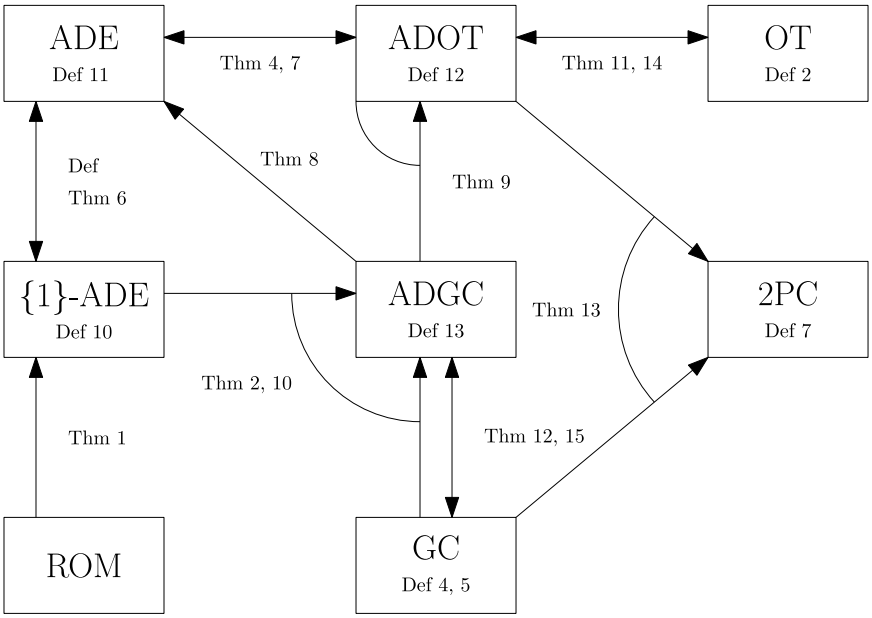

Figure 1: An implication diagram illustrating how the primitives in this work relate to each other and the reference definitions and theorems. Arrows $A \to B$ means that primitive A can be used to construct B, maintaining succinctness when possible. Arrows/primitives connected by arcs are implications that require multiple primitives. For example, ADGC can be built from {1}-ADE *and* GC as seen in Theorems 2 and 10.

As one of our main contributions, we use our ADS framework to specify a security definition for encryption called adaptive distributional encryption (ADE). Analogous to CCR and Free XOR, ADE provides a falsifiable complexity assumption on an encryption scheme that we demonstrate is sufficient to achieve input-adaptivity for several problems. Even the strongest variant of ADE is *trivially* achieved by RO. Thus, when used correctly, ADE can give security to input-adaptive techniques with the same efficiency as RO and while increasing our confidence that the instantiated scheme is plausibly secure.

Additional ADS Constructions. Beyond ADE, we also apply the ADS framework to a variant[1] of Oblivious Transfer (OT) and garbled circuits, called adaptive distributional OT (ADOT) and adaptive distributional GC (ADGC). For each of these primitives, we are able to obtain AD security with a succinct online-phase. In particular, the online-cost of the OT and GC variants is the size of garbled circuit labels (for any selectively secure scheme), and the cost of ADE applied to garbled material.

To top off the contributions of the new framework, we also make use of recent advances in garbled circuit security from DCR to achieve a very succinct and highly practical ADOT scheme that has an online-phase of $O(n + \mathsf{poly}(\lambda))$, along with similar implications for ADGC and ADE.

We show that this new ADOT scheme can be used to achieve practical, near-optimal online-cost, constant online round 2PC from ADE and DCR with online-cost $2m + O(n + \mathsf{poly}(\lambda))$, where n is the number of input bits and m the number of output bits.[2] We note that prior work [AIKW13, DLL24] was able to achieve similar results, but required unreserved use of the ROM, which we avoid.

[1] In our considered variant of OT, the sender's messages are fixed in the offline-phase, and the receiver's choice bits are learned in the online-phase. This primitive is, for example, useful as a building block in 2PC protocols, where the sender's messages might be cryptographic labels needed to evaluate a garbled circuit.

[2] The $2m$ factor comes from the fact that only one party gets the output and needs to send it to the other party, but in the case that only one party needs to get input this can be avoided.

1.2 Overview of Our Approach

To demonstrate the ADS framework, we describe our first concrete contribution: a security definition for an encryption scheme that captures the nuance of input-adaptivity, which we call *adaptive distributional encryption* (ADE).

In its most general form, an ADE scheme allows one to simultaneously encrypt a vector of n messages, resulting in a master key k and single ciphertext c. In the online-phase, one can choose an *opening* x, where x is a string that is the member of some set of legal opening choices $\mathcal{X}_n$ called the *openings set*. Given k and x, one can efficiently compute a succinct key k_x, which allows one to decrypt only those messages indicated by x. Hence, an ADE scheme fundamentally has a notion of input-adaptivity: the ciphertext c is sent in the offline-phase, and the short opening key k_x is sent in the online-phase. Our notion of security for ADE will serve as a template for our framework more generally.

In order to define security of an ADE scheme, we first require an intermediate definition that relates two "similar" distributions, which we refer to as $\mathcal{X}_n$-distributional indistinguishability. Here, we consider a distribution D_0 (resp. D_1) that outputs a list of messages $\mathbf{m}^0$ (resp. $\mathbf{m}^1$), as well as a "context" σ_0, ξ_0 (resp. σ_1, ξ_1). The notion of a context is central to ADS. In short, the side-information σ corresponds to the notion of "what the adversary sees in the offline-phase", and ξ is an efficient function that corresponds to the notion of "what the adversary sees in the online-phase". $\mathcal{X}_n$-distributional indistinguishability is defined with respect to the following game (see Definition 8 for formal presentation):

- First, the challenger uniformly chooses either D_0 or D_1, and samples $\sigma, \xi, \mathbf{m}$ from the chosen distribution.
- Next, the challenger gives the adversary the offline-phase information σ. In response, the adversary is allowed to choose which messages in $\mathbf{m}$ it would like to open by giving an opening $x \in \mathcal{X}_n$. In addition, the adversary is allowed to submit a "query" to the online-phase, called y.
- Finally, the challenger provides the adversary with online-phase information $\xi(y)$ and the chosen messages $\mathbf{m}_x$.

Distributions D_0 and D_1 are $\mathcal{X}_n$-indistinguishable if no adversary can reliably guess whether the challenger chose D_0 or D_1.

Now we are ready to define security of an ADE scheme. In particular, the security of a $\mathcal{X}_n$-ADE scheme is defined with respect to an arbitrary pair of $\mathcal{X}_n$-indistinguishable distributions D_0 and D_1 (see Definition 11 for formal presentation):

- First, the challenger uniformly chooses either D_0 or D_1, and samples $\sigma, \xi, \mathbf{m}$ from the chosen distribution.
- Next, the challenger uses the encryption scheme to encrypt $\mathbf{m}$, resulting in ciphertext c and master key k.
- The challenger gives the adversary both σ and c. In response, the adversary is allowed to chooses which messages in $\mathbf{m}$ it would like to open by giving an opening $x \in \mathcal{X}_n$. In addition, the adversary is allowed to submit a "query" to the online-phase, called y.
- The challenger constructs the opening key k_x, then gives k_x and $\xi(y)$ to the adversary.

The encryption scheme satisfies ADE security if no adversary can reliably guess whether the challenger chose D_0 or D_1. Informally, ADE captures the notion that if two processes are indistinguishable, given that adversary does not have access to some message in the offline-phase, then they should should still be indistinguishable when only an encrypted version of that message is given in the offline-phase.

We emphasize the similar structure of the above two security definitions. Each of our security definitions will follow the same structure, where there is a notion of an offline- and online-phase, as well as contexts σ and ξ. It is in this sense that ADS is a *framework* for security.

The compositional nature of ADS is also readily apparent. Notice that we can reformulate the definition of ADE security as an *instance* of distribution indistinguishability, albeit where the message vector is trivial (i.e., empty). In particular, in the ADE scheme offline-phase, the challenger sends both side information σ and ciphertext c, but we can imagine absorbing the ciphertext c into a larger piece of side information $\sigma' = (\sigma, c)$; similarly, we can imagine absorbing the opening key into the online-phase information $\xi'((y, x)) = (\xi(y), \mathsf{k}_x)$. The resulting σ', ξ' can be packaged into a distribution the above notion of distribution indistinguishability. More generally, when constructing an ADS object that delegates a task to some other ADS object in a compositional manner, we can absorb the callee's side information σ, ξ into that of the caller.

We remark that the ADE security definition is somewhat similar to that of so-called SOA-CPA encryption [DNRS99, BHY09, FHKW10, LDL+14, ORV14], except that we are crucially interested in also having a succinct opening functionality, and our notion of security is slightly stronger.

ADE exemplifies the dynamic of ADS, and in this work we explore applying this framework to other primitives, including oblivious transfer (Definition 12) and garbling schemes (Definition 13).

1.3 Roadmap of our approach

In the main body of this work we introduce definitions for ADE, ADOT, and ADGC in Section 3 and then we demonstrate a construction pathway to achieve ADOT and 2PC with online-costs $O(n + \mathsf{poly}(\lambda))$ and $2m + O(n + \mathsf{poly}(\lambda))$ respectively. In particular, this pathway mostly consists of simple relational results that generally show the interconnectedness of ADS primitives, which we hope demonstrates the utility and elegance of the ADS framework as a whole.

Pathway. We give the pathway in steps.

- **(Theorem 1)** We first start by showing that there exists a so-called universal ADE scheme that can be built from RO with online-cost $n\lambda$. Technically, this ADE scheme is secure for any opening $\mathcal{X}_n$, but we only require that this result implies the existence of succinct $\{1\}$-ADE, which only encrypts one message and always ends up opening the message.

- **(Theorem 2)** From $\{1\}$-ADE and any selectively secure garbling scheme, we construct an ADGC scheme with online-cost $|\tilde{x}| + \mathsf{poly}(\lambda)$ for label size $|\tilde{x}|$ by simply encrypting the garbled circuit $\tilde{C}$ and decoding table d with $\{1\}$-ADE. We send the ciphertext for that in the offline-phase, and we send the $\{1\}$-ADE scheme's succinct key in the online-phase.

- **(Fact 1)** We then combine the above transformation with recent results in the space of arithmetic garbled circuits with good rate [BLLL23] to achieve a garbling scheme with very short keys in the online-phase. Namely, our ADGC scheme assumes decisional composite residuosity (DCR) and has online label size $O(n + \mathsf{poly}(\lambda))$. Note that while the [BLLL23] garbling technique manipulates arithmetic values, our resulting ADGC scheme manipulates Boolean labels.

- **(Theorem 8)** We perform a trivial transformation on the above ADGC scheme to obtain universal ADE for the same online cost: $O(n + \mathsf{poly}(\lambda))$ bits.

- **(Theorem 4)** Then we show that $\mathcal{X}_n^{1/2}$-ADE can be used to construct $\mathcal{X}_n^{1/2}$-ADOT, where $\mathcal{X}_n^{1/2}$ is the openings ensemble associated with standard 1-out-of-2 OT, where the receiver chooses between two messages in a batched fashion. This construction makes use of Beaver's preprocessed OT [Bea95] by running the preprocessing step in the offline-phase, and then encrypting the possible responses for each message using ADE, so that instead of having to send the responses directly in the online-phase, which would be too large, we can instead send a succinct decryption key.

- **(Corollary 1)** We then combine the prior two results to construct an $\mathcal{X}_n^{1/2}$-ADOT scheme with online-cost $O(n + \mathsf{poly}(\lambda))$, which is succinct in the message sizes.

- **(Theorems 2 and 15)** Again, we give a trivial transformation from ADGC to get adaptive GC schemes by including an output table of necessary size to abide by the [AIKW13]. In particular, we can take any selectively secure GC scheme with projective labels of total size $n\lambda$ and turn it into an adaptive GC scheme with label cost $n\lambda$ and online-cost m.
- **(Corollary 2)** Lastly, we can use the above ADOT scheme and the above adaptive GC scheme to get 2PC with online-cost $2m + O(n + \mathsf{poly}(\lambda))$ using the ADOT scheme in place of an OT scheme for the normal 2PC from garbled circuits construction.

Framework. Each step in the above pathway consists of a quite simple proof, but in combination we can achieve state-of-the-art results from {1}-ADE and DCR using only general transformational theorems. Figure 1 depicts the relationships between ADS objects described by our theorems. Note that for space reasons and because many of them are straightforward, many of the depicted theorems can be found in the appendix.

1.4 Prior Work

Preprocessed OT. There is considerable work in finding OT protocols that do not require the use of expensive cryptography in the time-critical online-phase of the protocol, first posited in [Bea95], who pointed out that random OT could be used in the preprocessing step to give a cryptography-free online-phase protocol. This idea was expanded upon by [IKNP03] giving a technique called OT extension, in which a small number of OTs in either preprocessing- or online-phases could be extended to an arbitrary number of real OTs using symmetric-key primitives. There also exist an extensive literature on adaptive k out of n OT schemes [CNs07, GH08, JL09, RKP09, KNP10, GH11, KNP11, Zha11, LLM+17], that achieves near $O(k)$ online-phase cost. Every one of the above OT protocols sends something the size of the received messages in the online-phase, so batched 1-out-of-2 OT costs $O(nm)$ bits at least.

Lower-Bounds. It was shown in [AIKW13] that adaptive simulation-based *randomized encodings* — a more general version of garbling scheme Definition 5) — must send something in the online-phase at least the size of the output of the circuit for general circuits. Further, [HW15] gave a similar lower-bound for adaptive simulation-based 2PC against malicious adversaries. In particular, OT is an example of a 2PC functionality that suffers from the [HW15] lower-bound, even when the messages are known by the sender in the offline-phase. [KKPW21] also demonstrated that the naive adaptive version of Yao's garbled circuit technique, in which the Yao-style garbled circuit is sent in the offline-phase, cannot be shown secure from CPA secure encryption without modification.[3] Likewise, the works of [BHKO23, GYW+23] showed that the lower-bound of [AIKW13] does apply to the NPROM, but not to protocols in the programmable ROM (PROM) [FLR+10], giving another separation result between the two models.

Adaptive Garbled Circuits. Garbled circuits seem like a prime candidate for preprocessing, since the vast majority of the protocol material — the garbled circuit itself — can be generated independently of the inputs of the parties. In fact, some work prior to 2012 used selectively-secure garbling schemes adaptively in an attempt to get one-time programs [GKR08, GGP10]; however, [BHR12a] pointed out that selectively secure garbling schemes are not adaptive by default. Further, [AIKW13] later gave a lower-bound that simulation-based adaptive garbled circuits must send something at least the size of the output of the circuit in the online-phase. A similar lower-bound is given in [HW15] for 2PC, which states that maliciously-secure simulation-based adaptive 2PC schemes must also send something at least the size of the output of the circuit in the online-phase. In particular, batched 1-out-of-2 OT is an example of a worst-case functionality by [HW15].

[3] More precisely, there is a necessary security loss that is exponential in the depth of the circuit being garbled.

Weaker Adaptive Garbling Notions. Weaker definitions for adaptive garbling have been proposed as well to get around the lower-bound of [AIKW13] while maintaining some notion of security. The first of these was adaptive indistinguishable garbling posed by [JW16], which is secure only for adversaries that first choose two circuits C_0 and C_1 in the offline-phase, and then choose inputs x_0 and x_1 in the online-phase such that $C_0(x_1) = C_1(x_1)$. Standard model constructions of indistinguishability-based security that circumvent the lower-bound of [AIKW13] are only known for log-depth circuits. Recently, [ABK+23] posed a framework for a hybrid distributional and simulation based security notion that is plausibly not subject to the [AIKW13] lower-bound; however, no standard model constructions are posited. Unfortunately, neither weakening is known to be able to construct simulation-based adaptive security, even when the necessary one-time pad the size of the output message is given. We point out that these results are important for demonstrating that there is hope to circumvent the lower-bound of [AIKW13].

Adaptive Garbling from RO. The first garbling scheme to be shown adaptively secure is that of [BHR12a] in the programmable ROM and achieves an online-phase of size $O(n\lambda)$ for number of inputs n by using the RO to one-time pad the garbled circuit.[4] Work by [AIKW13] also gives a protocol for achieving online-cost $n + o(n)$ in the PROM frp, various public-key assumptions (DDH, LWE, or RSA) for randomized encodings, which is near enough to garbling that we mention it here. The $n + o(n)$ scheme was improved in [DLL24] to also allow for reuse based on PROs and ring-LWE. Recently, [BHKO23, GYW+23] showed that many popular garbling schemes are adaptively secure from in the NPROM without modification.[5] The online-phase for protocols using [BHKO23, GYW+23] scale only with the size of the input labels and decoding table, which must abide by the [AIKW13] lower-bound; however, the works of [BHR12a, AIKW13, DLL24] circumvent the lower-bound using the programmable RO.

Adaptive Garbling from the Standard Model. There is a large variety of standard model adaptive garbling schemes. Starting with results from OWFs, the work of [HJO+16] achieves an online-phase of $O(w\lambda)$ for maximum circuit width w from a primitive called somewhere equivocal encryption. Further, [JW16] showed that the naive adaptive version of Yao's garbling scheme *is* adaptively secure for log-depth circuits. From public-key assumptions, [AS16] achieves an online-phase of size $O(n\lambda + m)$ for output-size m from indistinguishability obfuscation. The state of art for near-optimal garbling is given from the work [GS18, GOS18], which achieves an online-phase of size $O(n + m + \log C)$ from laconic OT — which is known from CDH. None of the above schemes are considered to be viable in practice, due either to concrete efficiency concerns [AS16, HJO+16, GS18, GOS18] or for not being able to be applied to a sufficiently expressive circuit class [JW16].

Other Proposed Standard Model Security Definitions. The work of [BHK13] gives a framework for specifying standard model security properties of hash functions, which they call the universal computational extractor (UCE) framework. [BHK13] gives four security definitions under the UCE framework, as well as constructions for many primitives under those definitions, including a construction for simulation-based adaptive garbled circuits with online-phase at least the output size. [BHK13] states that their UCE framework falls into the category of *second-degree assumptions*, which are assumptions that are not phrased as an interactive game between a challenger and an adversary. In particular, in UCE, the adversary is split into two phases a "source" and a "distinguisher". The source phase runs first, and it may pass "leakage" to the distinguisher, but, crucially, this leakage is constrained by the UCE security definitions. Thus, while this framework is not expressible as a standard game-based definition, it is still falsifiable, in contrast to models such as NPROM for which we simply take a hash function to behave as an NPRO. Follow-on work [ST17] applied the UCE methodology to the context of pseudorandom permutations.

Notably, the aims of UCE are similar to those of this work, and in particular the authors explicitly give results in the input-adaptive setting. However, the {1}-ADE primitive (Definition 10) that we propose

[4] The protocol given in [BHR12a] is garbling-scheme agnostic, and so using better protocols may yield better online-phases.

[5] [GYW+23] shows a stronger cryptanalytic claim that using a popular fixed-key AES based hash function does not lose a factor square root security when fixed key AES is modeled as a non-programmable invertible permutation.

— which when combined with common standard model assumptions is sufficient for all our results — is a first-degree assumption, and so it cannot imply UCE security [BHK13]. Thus, it is either the case that UCE and ADS are formally incomparable, or that ADS is a weaker assumption.

In terms of their impact, UCE and ADS are also incomparable: UCE obtains security for a large number of interesting problems, including for some that are known to be impossible from a first-degree assumption [Wic13]; ADS definitions gives new notions of security for a narrower range of problems, but does so using standard first-degree assumptions. Because ADS is a weaker notion of security, it can be plausibly achieved from simpler primitives. For example, ADE can be plausibly instantiated from any PRG (See discussion around Conjecture 1), whereas UCE explicitly cannot, and requires a stronger notion of a cryptographic hash function [BHK13] or fixed-key AES [ST17].

Section 6 expands our discussion of possible connections between ADS and UCE.

2 Preliminaries

This section presents our general notation as well as formal definitions of existing objects, including OT, garbling schemes, and 2PC. The main non-standard concept here is that of an *opening family* (Notation 2); other definitions are relatively standard, and we invite the reader to refer to them as needed.

2.1 Notation and Concepts

Notation 1 (General Notation)**.**

- $X \approx Y$ *denotes that distributions* X *and* Y *are computationally indistinguishable.*
- $\mathcal{O}$ *denotes a random oracle.*
- $a, b, \tau \leftarrow \langle A(x), B(y) \rangle$ *denotes that interactive processes* A *and* B *act on respective inputs* x *and* y*. As a result of the interaction,* A *outputs* a *and* B *outputs* b*. The third output* τ *denotes the protocol transcript.*

Because adaptive security depends on opening subsets of encryptions, many definitions will be associated with a family of valid openings $\mathcal{X}_n$ for associated number of messages n. As such, we will use the following notation.

Notation 2 (Opening Family)**.** *An opening family* $\{\mathcal{X}_n\}_{n\in\mathbb{N}}$ *is a set ensemble where each set* $\mathcal{X}_n \subset \{0,1\}^n$ *corresponds to a valid way to open a message vector of length* n*, such that for* $x \in \mathcal{X}_n$*,* $x_i = 1$ *corresponds to the* i*-th message in a message vector being opened and vice versa for* $x_i = 0$*. The opened message vector can also be written* $\mathbf{m}_x = \{\mathbf{m}_i\}_{x_i=1}$ *and we also assume that* $x \in^? \mathcal{X}_n$ *can be checked efficiently in the length of* x*.*

We assume that every opening set includes the element 0^n*, which corresponds to no opening, i.e. an abort. As a slight abuse of notation and when clear from context, we will simply write* $\mathcal{X}_n$ *in place of* $\{\mathcal{X}_n\}_{n\in\mathbb{N}}$*.*

Notation 3 (Common Opening Ensemble)**.** *We give two common opening ensembles that we will use in this work. The first is a vector 1-out-of-2 selection, in which for each pair of messages, one chooses to open exactly 1 element of the pair. In regex, the ensemble is described as* $\{01, 10\}^{n/2}$*. We denote a 1-out-of-2 selection by* $\mathcal{X}_n^{1/2}$*.*

2.2 Primitive Definitions

Definition 1 ($\mathcal{X}_n$-Oblivious Transfer)**.** *An* ***oblivious transfer*** *scheme (OT) w.r.t. openings* $\mathcal{X}_n$ *consists of 4 potentially interactive PPT protocols* $\Pi = (Send_0, Recv_0, Send_1, Recv_1)$ *that satisfy the following condition.*

- ***Correctness:*** *For all parameters $\lambda, m, n \in \mathbb{N}$, messages $\mathbf{m} \in \{0,1\}^{n \times m}$, openings $x \in \mathcal{X}_n$, and the following*

$$s_0, r_0, \_ \leftarrow \left\langle \mathit{Send}_0(1^\lambda, \mathbf{m}), \mathit{Recv}_0(1^\lambda) \right\rangle$$
$$\_, \mathbf{m}', \_ \leftarrow \langle \mathit{Send}_1(s_0), \mathit{Recv}_1(r_0, x) \rangle$$

then it should be that $\mathbf{m}' = \mathbf{m}_x$.

- ***Online Communication Complexity:*** *For the following*

$$s_0, r_0, \_ \leftarrow \left\langle \mathit{Send}_0(1^\lambda, \mathbf{m}), \mathit{Recv}_0(1^\lambda) \right\rangle$$
$$\_, \_, \tau_1 \leftarrow \langle \mathit{Send}_1(s_0), \mathit{Recv}_1(r_0, x) \rangle$$

let T be such that $|\tau_1| \leq T(\lambda, m, n)$ for all $\mathbf{m} \in \{0,1\}^{n \times m}$ and $x \in \mathcal{X}_n$. We say that T is the online communication complexity of Π, and Π is succinct when $T(\lambda, m, n) = o(nm)$, but can scale with λ arbitrarily.

Remark 1. *We give a very general definition for OT that any OT scheme with preprocessing may be able to conform to. The various phases are not meant to prescribe any round complexity.*

Remark 2. *The definition for OT above is not the usual definition since the messages the sender has must be known in the offline-phase. This is because it is information-theoretically impossible to have a scheme in which the sender only knows the messages in the online-phase but maintains succinctness.*

Definition 2 ($\mathcal{X}_n$-Adaptive Simulation Oblivious Transfer Security)**.** *An $\mathcal{X}_n$-OT scheme as in Definition 1 is **(adaptively) simulation secure** if it satisfies the following conditions.*

- ***Simulation Receiver Privacy:*** *There exists a PPT simulator $\mathcal{S}$ such that for all parameters $m, n \in \mathbb{N}$, messages $\mathbf{m} \in \{0,1\}^{n \times m}$, inputs $x \in \mathcal{X}_n$, and for the following*

$$R_0, R_1 \leftarrow\$ \{0,1\}^*$$
$$s_0, r_0, \tau_0 \leftarrow \left\langle \mathit{Send}_0(1^\lambda, \mathbf{m}; R_0), \mathit{Recv}_0(1^\lambda) \right\rangle$$
$$\_, \_, \tau_1 \leftarrow \langle \mathit{Send}_1(s_0; R_1), \mathit{Recv}_1(r_0, x) \rangle$$

then it should be that

$$\{\mathbf{m}, R_0, R_1, \tau_0, \tau_1\} \approx \{\mathcal{S}(1^\lambda, 1^m, 1^n)\}$$

- ***Adaptive Simulation Sender Privacy:*** *There exists a PPT simulator $\mathcal{S} = (\mathcal{S}_1, \mathcal{S}_2)$ such that for all parameters $m, n \in \mathbb{N}$, messages $\mathbf{m} \in \{0,1\}^{n \times m}$, for all PPT adversaries[6] $\mathcal{A}$, and for the following games*

$\mathrm{Real}^{\mathrm{sim-ot}}_{\mathcal{A},\mathbf{m}}(\lambda)$	$\mathrm{Ideal}^{\mathrm{sim-ot}}_{\mathcal{A},\mathcal{S},\mathbf{m}}(\lambda)$
$R_0, R_1 \leftarrow\$ \{0,1\}^*$	$x, S \leftarrow \mathcal{S}_1^{\mathcal{A}}(1^\lambda)$
$s_0, r_0, \tau_0 \leftarrow \left\langle \mathit{Send}_0(1^\lambda, \mathbf{m}), \mathit{Recv}_0(1^\lambda; R_0) \right\rangle$	**if** $x \in \mathcal{X}_n$:
$x, S \leftarrow \mathcal{A}(1^\lambda, R_0, \tau_0)$	**return** $\mathcal{S}_2^{\mathcal{A}}(S, \mathbf{m}_x)$
$\_, \_, \tau_1 \leftarrow \langle \mathit{Send}_1(s_0), \mathit{Recv}_1(r_0, x; R_1) \rangle$	**else** :
return (S, R_1, τ_1)	**return** $\perp$

then it should be the case that

$$\{\mathrm{Real}^{\mathrm{sim-ot}}_{\mathcal{A},\mathbf{m}}(\lambda)\} \approx \{\mathrm{Ideal}^{\mathrm{sim-ot}}_{\mathcal{A},\mathcal{S},\mathbf{m}}(\lambda)\}$$

[6] Adversaries should always output some $x \in \mathcal{X}_n$.

Definition 3 (Garbled Circuits)**.** *A **garbling** scheme is a tuple of PPT algorithms:*

$$\Pi = (\mathit{Garble}, \mathit{Encode}, \mathit{Eval}, \mathit{Decode})$$

Π *must satisfy the following correctness condition:*

- ***Correctness:*** *For all parameters $\lambda, m, n \in \mathbb{N}$, inputs $x \in \{0,1\}^m$, and circuits $C : \{0,1\}^n \to \{0,1\}^m$, and the following*

$$\begin{aligned} \tilde{C}, e, d &\leftarrow \mathit{Garble}(1^\lambda, C) \\ \tilde{x} &\leftarrow \mathit{Encode}(e, x) \\ \tilde{y} &\leftarrow \mathit{Eval}(\tilde{C}, \tilde{x}) \\ y &\leftarrow \mathit{Decode}(d, \tilde{y}) \end{aligned}$$

  then it should be the case that $y = C(x)$.

- ***Online Communication Complexity:*** *For the following*

$$\begin{aligned} \tilde{C}, e, d &\leftarrow \mathit{Garble}(1^\lambda, C) \\ \tilde{x} &\leftarrow \mathit{Encode}(e, x) \end{aligned}$$

  let T be such that $|d| + |\tilde{x}| \leq T(\lambda, C)$ for adaptive garbling scheme (AGC) (Definition 5) or $|d| + |\tilde{x}| + |\tilde{C}| \leq T(\lambda, C)$ for selective garbling scheme (SGC) (Definition 4) for all circuits $C : \{0,1\}^n \to \{0,1\}^m$. We say that T is the online communication complexity of Π, and that Π is circuit-succinct if $T(\lambda, C) = o(|C|)$ and succinct if $T(\lambda, C) = o(m)$.[7] *Π can be succinct and scale with λ arbitrarily.*

Definition 4 (Selective Simulation Garbled Circuit Security)**.** *A garbled circuit scheme Π as in Definition 3 can be selectively simulation secure in the manner shown below.*

- ***Selective Simulation Privacy:*** *There exists a PPT simulator $\mathcal{S}$ such that for all parameters $m, n \in \mathbb{N}$, for all circuits $C : \{0,1\}^n \to \{0,1\}^m$, for all PPT adversaries $\mathcal{A}$, and for the following*

$$\begin{aligned} \tilde{C}, e, d &\leftarrow \mathit{Garble}(1^\lambda, C) \\ \tilde{x} &\leftarrow \mathit{Encode}(e, x) \end{aligned}$$

  the it should be that

$$\{\tilde{C}, \tilde{x}, d\} \approx \{\mathcal{S}(1^\lambda, \Phi(C), C(x))\}$$

for topology of circuit $\Phi(C)$.

Definition 5 (Adaptive Simulation Garbled Circuit Security)**.** *A garbled circuit scheme Π as in Definition 3 can be **adaptively simulation secure** in the manner shown below.*

- ***Adaptive Simulation Privacy:*** *There exists a PPT simulator $\mathcal{S} = (\mathcal{S}_1, \mathcal{S}_2)$ such that for all parameters $m, n \in \mathbb{N}$, for all circuits $C : \{0,1\}^n \to \{0,1\}^m$, for all PPT adversaries $\mathcal{A}$, and for the following games*

$\mathrm{Real}^{\mathrm{sim-prv}}_{\mathcal{A},C}(\lambda)$	$\mathrm{Ideal}^{\mathrm{sim-prv}}_{\mathcal{A},\mathcal{S},C}(\lambda)$
1 : $\tilde{C}, e, d \leftarrow \mathit{Garble}(1^\lambda, C)$	1 : $\tilde{C}, \tilde{x}, S_{\mathcal{S}} \leftarrow \mathcal{S}_1(1^\lambda, \Phi(C))$
2 : $x, S \leftarrow \mathcal{A}(1^\lambda, \tilde{C})$	2 : $x, S_{\mathcal{A}} \leftarrow \mathcal{A}(1^\lambda, \tilde{C})$
3 : $\tilde{x} \leftarrow \mathit{Encode}(e, x)$	3 : $d \leftarrow \mathcal{S}_2(S_{\mathcal{S}}, C(x))$
4 : **return** $(S, \tilde{x}, d)$	4 : **return** $(S_{\mathcal{A}}, \tilde{x}, d)$

[7] The lower-bound for online communication complexity of simulation-based garbling as in Definition 5 is the size of the output of the circuit, shown in [AIKW13], which makes this particular definition of succinctness interesting to beat.

for topology of circuit $\Phi(C)$ the it should be that

$$\{\text{Real}^{\text{sim-prv}}_{\mathcal{A},C}(\lambda)\} \approx \{\text{Ideal}^{\text{sim-prv}}_{\mathcal{A},\mathcal{S},C}(\lambda)\}$$

Definition 6 (2-Party Computation)**.** *A **2 party computation** scheme (2PC) is a tuple of 4 potentially interactive PPT protocols $\Pi = (\mathit{Offline}_0, \mathit{Offline}_1, \mathit{Online}_0, \mathit{Online}_1)$ with the following properties.*

- ***Correctness:*** *For all parameters $\lambda, m, n \in \mathbb{N}$, circuits $C : \{0,1\}^{n_0} \times \{0,1\}^{n_1} \to \{0,1\}^m$ such that $n_0 + n_1 = n$, all inputs $x_0 \in \{0,1\}^n_0$ and $x_1 \in \{0,1\}^{n_1}$, and for the following*

$$s_0, s_1, \_ \leftarrow \left\langle \mathit{Offline}_0(1^\lambda, C), \mathit{Offline}_1(1^\lambda, C) \right\rangle$$
$$y'_0, y'_1, \_ \leftarrow \left\langle \mathit{Online}_0(s_0, x_0), \mathit{Online}_1(1^\lambda, x_1) \right\rangle$$

  then it should be that $y'_0 = y_0$ and $y'_1 = y_1$ for $(y_0, y_1) = C(x_0, x_1)$.

- ***Online Communication Complexity:*** *For the following*

$$s_0, s_1, \_ \leftarrow \left\langle \mathit{Offline}_0(1^\lambda, C), \mathit{Offline}_1(1^\lambda, C) \right\rangle$$
$$\_, \_, \tau_1 \leftarrow \left\langle \mathit{Online}_0(s_0, x_0), \mathit{Online}_1(1^\lambda, x_1) \right\rangle$$

  let T be such that $|\tau_1| \leq T(\lambda, C)$ for all circuits $C : \{0,1\}^{n_0} \times \{0,1\}^{n_1} \to \{0,1\}^m$ such that $n_0 + n_1 = n$ and all inputs $x_0 \in \{0,1\}^{n_0}$ and $x_1 \in \{0,1\}^{n_1}$. We say that T is the online communication complexity of Π, and that Π is circuit-succinct if $T(\lambda, C) = o(|C|)$ and succinct if $T(\lambda, C) = o(m)$.[8] *Π can be succinct and scale with λ arbitrarily.*

Definition 7 (Adaptive Simulation 2PC)**.** *A 2PC scheme as in Definition 6 is **adaptively simulation secure** if it satisfies the following condition for both parties.*

- ***Simulation P_b Privacy:*** *There exists a PPT simulator $\mathcal{S} = (\mathcal{S}_1, \mathcal{S}_2)$ such that for all parameters $m, n \in \mathbb{N}$, circuits $C : \{0,1\}^{n_0} \times \{0,1\}^{n_1} \to \{0,1\}^m$, inputs $x_b \in \{0,1\}^{n_b}$, and for the following games*

$\text{Real}^{b-\text{sim}-2\text{pc}}_{\mathcal{A},C,x_b}(\lambda)$

1 : $R_0, R_1 \leftarrow\!\!\$\ \{0,1\}^*$
2 : $s_b, s_{1-b}, \tau_0 \leftarrow \left\langle \mathit{Offline}_b(1^\lambda, C), \mathit{Offline}_{1-b}(1^\lambda, C; R_0) \right\rangle$
3 : $x_{1-b}, S \leftarrow \mathcal{A}(1^\lambda, R_0, \tau_0)$
4 : $\_, \_, \tau_1 \leftarrow \langle \mathit{Online}_b(s_b, x_b), \mathit{Online}_{1-b}(s_{1-b}, x_{1-b}; R_1) \rangle$
5 : **return** (S, R_1, τ_1)

$\text{Ideal}^{b-\text{sim}-2\text{pc}}_{\mathcal{A},\mathcal{S},C,x_b}(\lambda)$

1 : $x_{1-b}, S_{\mathcal{S}} \leftarrow \mathcal{S}_1^{\mathcal{A}}(1^\lambda)$
2 : **return** $\mathcal{S}_2^{\mathcal{A}}(S_{\mathcal{S}}, C(x_0, x_1))$

  then it should be the case that

$$\{\text{Real}^{b-\text{sim}-2\text{pc}}_{\mathcal{A},C,x_b}(\lambda)\} \approx \{\text{Ideal}^{b-\text{sim}-2\text{pc}}_{\mathcal{A},\mathcal{S},C,x_b}(\lambda)\}$$

[8] The lower-bound of [HW15] shows that malicious simulation-based security cannot achieve better than $\Omega(m)$ for some circuits in the online-phase, which makes this particular definition of succinctness an interesting target to beat.

3 A New Framework: Adaptive Distributional Security

This section formalizes the main ideas of the ADS framework and applies ADS to various primitives.

We first introduce a central definition: $\mathcal{X}_n$-indistinguishability. Normally, two distributions are indistinguishable if the full outputs of the distributions are indistinguishable. In the following definition, distributions output side information σ, efficient side information function ξ, and messages $\mathbf{m}$ that can only be opened according to $x \in \mathcal{X}_n$. The distributions should be indistinguishable when some adversary is first given σ, then produces an opening $x \in \mathcal{X}_n$ and side choice y, and finally sees $\mathbf{m}_x$ and $\xi(y)$.

Ultimately, our goal is to introduce a *composable* framework for security definitions related to input-adaptivity, and the side information σ and ξ act as our framework's glue. In particular, σ and ξ give a formal context by which a security definition can capture the abstract notion of the adversary's view of the past — namely σ — and the adversary's view of the future — namely ξ. As an example of this, consider an adaptive garbling scheme, which sends a garbled circuit in advance (hence, $\sigma = \tilde{C}$), then the evaluator decides their input and gets corresponding labels (hence, $\mathbf{m}_x = \tilde{x}$), and finally the evaluator receives the decoding table ($\xi(y) = d$). Using the notion of $\mathcal{X}_n$-indistinguishability, we can talk abstractly about many different adaptive cryptographic systems in a distributional manner:

Definition 8 ($\mathcal{X}_n$-Indistinguishable Distributions)**.** *We say that distributions D_0 and D_1 are **indistinguishable w.r.t. openings** $\mathcal{X}_n$ if for all $n, m \in \mathbb{N}$ and for all PPT adversaries $\mathcal{A}$*

$\mathrm{Game}^{\mathrm{d-ind}}_{\mathcal{A},D}(\lambda, m, n)$

1 : $\sigma, \xi, \mathbf{m} \leftarrow\!\!\$\ D(\lambda, m, n)$
2 : $x, y, S \leftarrow \mathcal{A}(1^\lambda, \sigma)$
3 : **if** $x \notin \mathcal{X}_n$:
4 : $\quad x \leftarrow 0^n$
5 : **return** $(S, \xi(y), \mathbf{m}_x)$

$$\{\mathrm{Game}^{\mathrm{d-ind}}_{\mathcal{A},D_0}(\lambda, m, n)\} \approx \{\mathrm{Game}^{\mathrm{d-ind}}_{\mathcal{A},D_1}(\lambda, m, n)\}$$

for PPT program ξ.

Remark 3. *Technically, the above definition does not explicitly need $\mathbf{m}$ to be present in the view since this information can be baked into ξ to output $\mathbf{m}_x$ along with other side information. We make explicit $\mathbf{m}$ for simplicity of notation.*

We also give a notion of distributional indistinguishability for distributions that output a circuit that is evaluated, instead of distributions that only open messages.

Definition 9 (Indistinguishable Circuit Distributions)**.** *We say that distributions D_0 and D_1 are **indistinguishable circuit distributions** if for all PPT adversaries $\mathcal{A}$*

$\mathrm{Game}^{\mathrm{c-ind}}_{\mathcal{A},D}(\lambda, m, n)$

1 : $\sigma, \xi, C \leftarrow\!\!\$\ D(\lambda, m, n)$
2 : $x, y, S \leftarrow \mathcal{A}(1^\lambda, \sigma)$
3 : **if** $x \notin \mathcal{X}_n$:
4 : $\quad x \leftarrow 0^n$
5 : **return** $(S, \xi(y), C(x))$

$$\{\mathrm{Game}^{\mathrm{c-ind}}_{\mathcal{A},D_0}(\lambda, m, n)\} \approx \{\mathrm{Game}^{\mathrm{c-ind}}_{\mathcal{A},D_1}(\lambda, m, n)\}$$

for efficient circuits $C : \{0,1\}^n \to \{0,1\}^m$ *with the same topology*[9] *when output by either* D_0 *or* D_1*, and PPT program* ξ*.*

Now, we formalize a central primitive: Adaptive Distributional Encryption (ADE). ADE is an encryption scheme that prevents an adversary from breaking some other ongoing processes that might be related to whatever is being encrypted, even when we give a decryption key to the adversary later. In contrast, most off-the-shelf standard model encryption schemes tend to lose their security guarantees if the secret key is ever revealed. We capture this security notion by giving an interactive game over $\{1\}$-indistinguishable distributions – $\mathcal{X}_n$-indistinguishable distributions in which the single message is always opened – and requiring that any such distributions maintain indistinguishability even when the adversary is given an encryption of the single message followed in the online-phase by a decryption key.

Definition 10 (Adaptive Distributional Encryption)**.** *An* ***adaptive distributional encryption*** *scheme (ADE) is an encryption scheme* $\Pi = (Enc, Dec)$ *with the following properties.*

- ***Correctness:*** *For all parameters* $\lambda, m \in \mathbb{N}$*, messages* $\overline{m} \in \{0,1\}^m$*, and for*

$$c, \mathsf{k} \leftarrow Enc(1^\lambda, \overline{m})$$
$$\overline{m}' \leftarrow Dec(\mathsf{k}, c)$$

  it should be that $\overline{m}' = \overline{m}$*.*

- ***Security:*** *For all parameters* $m \in \mathbb{N}$*,* $\{1\}$*-indistinguishable distributions* D_0 *and* D_1*, for all PPT adversaries* $\mathcal{A}$*, and for game*

$\mathrm{Game}^{\mathrm{ade}}_{\Pi,\mathcal{A},D}(\lambda, m)$

1 : $\sigma, \xi, \overline{m} \leftarrow\!\!\$\ D(\lambda, m, 1)$
2 : $c, \mathsf{k} \leftarrow Enc(1^\lambda, \overline{m})$
3 : $y, S \leftarrow \mathcal{A}(1^\lambda, \sigma, c)$
4 : **return** $(S, \xi(y), \mathsf{k})$

  it should be that

$$\{\mathrm{Game}^{\mathrm{ade}}_{\Pi,\mathcal{A},D_0}(\lambda, m)\} \approx \{\mathrm{Game}^{\mathrm{ade}}_{\Pi,\mathcal{A},D_1}(\lambda, m)\}$$

- ***Online Succinct:*** *For the following*

$$\_, \mathsf{k} \leftarrow Enc(1^\lambda, \overline{m})$$

  let T *be such that* $|\mathsf{k}| \leq T(\lambda, m)$ *for all* $\overline{m} \in \{0,1\}^m$*. We say that* T *is the online complexity of* Π*, and* Π *is succinct when* $T(\lambda, m) = o(m)$*, but can scale with* λ *arbitrarily.*

We also give an extension to ADE that applies to encrypting a set of messages and then opening some subset.

Definition 11 ($\mathcal{X}_n$-Adaptive Distributional Encryption)**.** *An adaptive distributional encryption scheme* ***w.r.t. openings*** $\mathcal{X}_n$ *consists of 3 PPT protocols* $\Pi = (Enc, Open, Dec)$ *with the following properties.*

[9] The specific definition of topology can depend on the protocol using the notion.

- ***Correctness:*** *For all parameters $\lambda, m, n \in \mathbb{N}$, messages $\mathbf{m} \in \{0,1\}^{n \times m}$, openings $x \in \mathcal{X}_n$, and the following*

$$\begin{aligned} c, \mathsf{k} &\leftarrow \mathit{Enc}(1^\lambda, \mathbf{m}) \\ \mathsf{k}_x &\leftarrow \mathit{Open}(\mathsf{k}, x) \\ \mathbf{m}' &\leftarrow \mathit{Dec}(\mathsf{k}_x, x, c) \end{aligned}$$

then it should be the case that $\mathbf{m}' = \mathbf{m}_x$. If $x \notin \mathcal{X}_n$, then Open should treat x as if it is 0^n.

- ***Security:*** *For all parameters $m, n \in \mathbb{N}$, $\mathcal{X}_n$-indistinguishable distributions D_0 and D_1, for all PPT adversaries $\mathcal{A}$, and for game*[10]

$\mathrm{Game}^{\mathrm{ade}}_{\Pi,\mathcal{A},D}(\lambda, m, n)$

1 : $\sigma, \xi, \mathbf{m} \leftarrow\!\!\$\ D(\lambda, m, n)$
2 : $c, \mathsf{k} \leftarrow \mathit{Enc}(1^\lambda, \mathbf{m})$
3 : $x, y, S \leftarrow \mathcal{A}(1^\lambda, \sigma, c)$
4 : $\mathsf{k}_x \leftarrow \mathit{Open}(\mathsf{k}, x)$
5 : **return** $(S, \xi(y), \mathsf{k}_x)$

it should be that

$$\{\mathrm{Game}^{\mathrm{ade}}_{\Pi,\mathcal{A},D_0}(\lambda, m, n)\} \approx \{\mathrm{Game}^{\mathrm{ade}}_{\Pi,\mathcal{A},D_1}(\lambda, m, n)\}$$

- ***Online Succinct:*** *For the following*

$$\begin{aligned} \_, \mathsf{k} &\leftarrow \mathit{Enc}(1^\lambda, \mathbf{m}) \\ \mathsf{k}_x &\leftarrow \mathit{Open}(\mathsf{k}, x) \end{aligned}$$

let T be such that $|\mathsf{k}_x| \leq T(\lambda, m, n)$ for all $\mathbf{m} \in \{0,1\}^{n \times m}$ and $x \in \mathcal{X}_n$. We say that T is the online complexity of Π, and Π is succinct when $T(\lambda, m, n) = o(nm)$, but can scale with λ arbitrarily.

Remark 4. *If a protocol is an $\mathcal{X}_n$-ADE for any $\mathcal{X}_n$, we will say that it is a universal ADE scheme.*

Remark 5. *From now on we will refer to the ADE scheme of Definition 10 as $\{1\}$-ADE, since it can be viewed as an $\mathcal{X}_n$-ADE scheme where there is only one message and it is opened.*

The OT and GC variant security definitions are similar in structure to their simulation-based counterparts, but with intuitive ADS-style definitions that require indistinguishability over $\mathcal{X}_n$- or circuit-indistinguishable distributions instead of over all message or all circuits, as it is in simulation-based security.

Definition 12 ($\mathcal{X}_n$-Adaptive Distributional Oblivious Transfer Security)**.** *An $\mathcal{X}_n$-OT scheme Π as in Definition 1 is **adaptively distributionally secure** if it satisfies the following conditions.*

- ***Adaptive Simulation Receiver Privacy:*** *Same as (adaptive) simulation receiver privacy as in Definition 2.*
- ***Adaptive Distributional Sender Privacy:*** *For all parameters $m, n \in \mathbb{N}$, $\mathcal{X}_n$-indistinguishable distributions D_0 and D_1, for all PPT adversaries*[11] *$\mathcal{A}$, and for game*

[10] The following game has the same name as the one used in Definition 10 since this is an extension, we will view this one as the cannonical version.

[11] Adversaries should always output some $x \in \mathcal{X}_n$.

$\underline{\text{Game}^{\text{adot}}_{\Pi,\mathcal{A},D}(\lambda,m,n)}$

1 : $\sigma,\xi,\mathbf{m} \leftarrow\!\$\ D(\lambda,m,n)$

2 : $R_0,R_1 \leftarrow\!\$\ \{0,1\}^*$

3 : $s_0,r_0,\tau_0 \leftarrow \left\langle \mathit{Send}_0(1^\lambda,\mathbf{m}),\mathit{Recv}_0(1^\lambda;R_0)\right\rangle$

4 : $x,y,S \leftarrow \mathcal{A}(1^\lambda,\sigma,R_0,r_0)$

5 : $\_,\_,\tau_1 \leftarrow \langle \mathit{Send}_1(s_0),\mathit{Recv}_1(r_0,x;R_1)\rangle$

6 : **return** $(S,\xi(y),R_1,\tau_1)$

it should be that

$$\{\text{Game}^{adot}_{\Pi,\mathcal{A},D_0}(\lambda,m,n)\} \approx \{\text{Game}^{adot}_{\Pi,\mathcal{A},D_1}(\lambda,m,n)\}$$

Definition 13 (Adaptive Distributional Garbled Circuit Security)**.** *An* ***adaptive distributional garbled circuits*** *scheme (ADGC) consists of 4 PPT protocols* $\Pi = (\mathit{Garble},\mathit{Encode},\mathit{Eval},\mathit{Decode})$ *that satisfy the following conditions.*

- ***Correctness:*** *Same as correctness in Definition 5.*
- ***Adaptive Distributional Privacy:*** *For all indistinguishable circuit distributions* D_0 *and* D_1*, for all PPT adversaries* $\mathcal{A}$*, and for game*

$\underline{\text{Game}^{\text{adgc}}_{\mathcal{A},D}(\lambda)}$

1 : $\sigma,\xi,C \leftarrow\!\$\ D(\lambda)$

2 : $\tilde{C},e,d \leftarrow \mathit{Garble}(1^\lambda,C)$

3 : $x,y,S \leftarrow \mathcal{A}(1^\lambda,\sigma,\tilde{C})$

4 : $\tilde{x} \leftarrow \mathit{Encode}(e,x)$

5 : **return** $(S,\xi(y),\tilde{x},d)$

it should be that

$$\{\text{Game}^{\text{adgc}}_{\mathcal{A},D_0}(\lambda)\} \approx \{\text{Game}^{\text{adgc}}_{\mathcal{A},D_1}(\lambda)\}$$

4 Constructions

We give the proofs from the pathway that we discussed in Section 1.3.

4.1 Constructing Universal ADE

We first show that the naive construction of a semantically secure encryption scheme from RO is a universal ADE scheme. In the following construction we give a Gen function, but for an ADE scheme this would be built into the Enc procedure.

Construction 1 (Universal ADE from ROM)**.** *The implementation from an RO is as follows.*

$$\begin{aligned}\mathit{Gen}(1^\lambda) &:= \text{ sample uniformly from } \{0,1\}^\lambda\\ \mathit{Enc}(\mathsf{k},\overline{m}) &:= \mathcal{O}(\mathsf{k}) \oplus \overline{m}\\ \mathit{Dec}(\mathsf{k},c) &:= \mathcal{O}(\mathsf{k}) \oplus c\end{aligned}$$

We then show security by considering that the one-time padded messages are generally information-theoretically hidden unless some key corresponding to an unopened message is found, which can only happen negligibly often.

Theorem 1 (Universal ADE from ROM)**.** *The simple ROM encryption scheme (Construction 1) is also an $\mathcal{X}_n$-ADE scheme for any opening ensemble $\mathcal{X}_n$ (Definition 11) with online-cost $T(\lambda, m, n) = n\lambda$*

Proof. We first note that Construction 1 is clearly correct for the opening function of simply sending the keys associated with openings, and has size λ keys per encryption, and so the online-cost is $T(\lambda, m, n) = n\lambda$.

Security. To show security we introduce the two following games. Note that we will assume that the distinguisher is given the oracle $\mathcal{O}$ at the end of the protocol. We also give the distribution D and output function ξ oracle access so that the scheme is oracle-composable and so that the scheme is closer to a standard model game, since D and ξ would always be able to depend on any standard model implementation.

$\text{Game}^{\text{ade}}_{\Pi,\mathcal{A},D}(\lambda, m, n)$	$\text{Hyb}_{\Pi,\mathcal{A},D}(\lambda, m, n)$
$\sigma, \xi, \mathbf{m} \leftarrow\!\!\$\ D^{\mathcal{O}}(\lambda, m, n)$	$\{\mathsf{k}_i \leftarrow\!\!\$\ \{0,1\}^\lambda\}_{i\in[n]}$
$\{\mathsf{k}_i \leftarrow\!\!\$\ \{0,1\}^\lambda\}_{i\in[n]}$	$\{c_i \leftarrow\!\!\$\ \{0,1\}^m\}_{i\in[n]}$
$\{c_i \leftarrow \mathcal{O}(\mathsf{k}_i) \oplus \mathbf{m}_i\}_{i\in[n]}$	$\sigma, \xi, \mathbf{m} \leftarrow\!\!\$\ D^{\mathcal{O}}(\lambda, m, n)$
$x, S \leftarrow \mathcal{A}^{\mathcal{O}}(1^\lambda, \sigma, \{c_i\}_{i\in[n]})$	$x, S \leftarrow \mathcal{A}^{\mathcal{O}}(1^\lambda, \sigma, \{c_i\}_{i\in[n]})$
return $(S, \xi^{\mathcal{O}}(x), \{\mathsf{k}_i\}_{x_i=1})$	**program** $\{\mathcal{O}(\mathsf{k}_i) := c_i \oplus \mathbf{m}_i\}_{x_i=1}$
	return $(S, \xi^{\mathcal{O}}(x), \{\mathsf{k}_i\}_{x_i=1})$

Then we are tasked with showing for any $\mathcal{X}_n$-indistinguiable distributions D_0 and D_1 that

$$\{\text{Game}^{\text{ade}}_{\Pi,\mathcal{A},D_0}(\lambda, m, n)\} \approx \{\text{Game}^{\text{ade}}_{\Pi,\mathcal{A},D_1}(\lambda, m, n)\}$$

which we will do in two steps. First note that for $b \in \{0, 1\}$,

$$\{\text{Game}^{\text{ade}}_{\Pi,\mathcal{A},D_b}(\lambda, m, n)\} \approx \{\text{Hyb}_{\Pi,\mathcal{A},D_b}(\lambda, m, n)\}$$

Note that the only way for $\mathcal{A}$ to tell the difference between the above two games is to guess a key k_i and witness that an opened key was reprogrammed (or not) by hitting a key in advance of outputting a choice. Note that even though the distributions D are given oracle access and are potentially computationally unbounded, this setting was shown to be the same as one in which an adversary gets to program polynomially many locations of the oracle in advance of the protocol [Unr07]. Since the keys are drawn uniformly either case occurs negligibly often.[12]

Then the second step is to show

$$\{\text{Hyb}_{\Pi,\mathcal{A},D_0}(\lambda, m, n)\} \approx \{\text{Hyb}_{\Pi,\mathcal{A},D_1}(\lambda, m, n)\}$$

Since the encryption does not contain any information about the distributional output at the time that $\mathcal{A}$ is outputting a decision, the above games reduce to the $\mathcal{X}_n$-indistinguishable distributions definition. □

The above proof technique was featured in [BHKO23]. Ample discussion about the technique can be found there.

We then give a conjecture that Construction 1 can be instantiated with AES, followed by discussion.

Construction 2 (Generator from AES)**.** *We give a standard stream pseudorandomness generator from AES as a pseudorandom function $F_k(\cdot)$:*

$$G(s) := F_s(0)||F_s(1)||F_s(2)||\cdots$$

for as much pseudorandomness as needed.

[12] This proof also serves to show that Construction 1 would also work from the auxiliary-input ROM of [Unr07].

Conjecture 1 (AES Generator is a Universal ADE scheme)**.** *Replacing the RO calls $\mathcal{O}(\mathsf{k})$ in Construction 1 with the generator $G(\mathsf{k})$ in Construction 2 is a Universal ADE (Definition 11).*

This conjecture stems from two points. First, the use of ROM in Construction 1 is mild: it uses input keys that are uniform and unstructured. Second, to our knowledge, there are no known attacks against using a generator (even contrived generators) to encrypt n independent messages, then allowing the adversary to open encryption seeds corresponding to some number of the messages. The second point is well studied by works related to selective-opening-attack (SOA) encryption [DNRS99, BHY09, FHKW10, LDL+14, ORV14]. If Conjecture 1 is constructively shown to be false for AES, we feel that such an attack would illuminate why security in the adaptive setting is concretely difficult, rather than difficult in the abstract. In light of the above, we are confidant that Conjecture 1 makes a reasonable assumption for use in future cryptographic works and as a standard-model-like assumption more generally.

4.2 ADGC from {1}-ADE

We show how using {1}-ADE to encrypt the garbled material of a selectively secure garbled circuit scheme achieves ADGC security with key-sized online-cost.

Construction 3 (ADGC from {1}-ADE and Selectively Secure Garbled Circuits)**.** *The protocol simply encrypts the garbled circuit $\tilde{C}$ and decoding table d together, sending the ciphertext in the offline-phase. Then the garbler sends the decryption key in the online-phase as the new decryption table.*

Then we demonstrate that Construction 3 is an ADGC scheme. We do this by viewing the selectively secure garbled circuit scheme as a pair of {1}-indistinguishable distributions, in which D_b is the real execution with the garbled material as the message and label encoding as the output of $\xi(y)$ for distributionally drawn circuit C_b. We show that D_0 and D_1 are {1}-indistinguishably using simulation security, since simulation security is stronger than ADS (see Theorem 15). Then we see that the above is in a form to immediately apply {1}-ADE.

Theorem 2 (ADGC from {1}-ADE and Garbled Circuits)**.** *Construction 3 is an ADGC scheme (Definition 13) from {1}-ADE and a selectively secure garbling scheme (Definitions 4 and 10) with online-cost $T(\lambda, C) = T_{ADE}(\lambda, |\tilde{C}| + |d|, 1)$ for garbled circuit $\tilde{C}$ and decoding table d.*

Proof.

Correctness. The construction simply applies encryption to $\tilde{C}$ and d, which is then revealed on schedule, and so the evaluator has the proper material for evaluating the circuit.

Adaptive Distributional Privacy. Let D_0 and D_1 be indistinguishable circuit distributions. Then consider the following statement using Definition 4 for $\sigma_b, \xi_b, C_b \leftarrow\!\!\$\ D_b$ for $b \in \{0,1\}$ and efficient adversary $\mathcal{A}$. For

$$\begin{aligned} x_b, y_b, S_b &\leftarrow \mathcal{A}(1^\lambda, \sigma_b) \\ \tilde{C}_b, e, d_b &\leftarrow \text{Garble}(1^\lambda, C_b) \\ \tilde{x}_b &\leftarrow \text{Encode}(e, x_b) \end{aligned}$$

we have that

$$\begin{aligned} &\{\sigma_0, \xi_0(y_0), S_0, \tilde{C}_0, \tilde{x}_0, d_0\} \approx \{\sigma_0, \xi_0(y_0), S_0, \mathcal{S}(1^\lambda, \Phi(C_0), C(x_0))\} \\ \approx \{\sigma_1, \xi_1(y_1), S_1, \mathcal{S}(1^\lambda, \Phi(C_1), C(x_1))\} &\approx \{\sigma_1, \xi_1(y_1), S_1, \tilde{C}_1, \tilde{x}_1, d_1\} \end{aligned}$$

The jump from Real to Ideal holds due to selective simulation privacy, since the adversary can internally mind σ_b and $\xi_b(y_b)$, and then jump between the Ideal games is due to the circuit indistinguishability of the distributions D_0 and D_1.

We can then consider viewing the above garbling scheme as a distribution on which $\{1\}$-ADE can apply. Consider distributions D'_0 and D'_1 implemented in the following way.

$D'_b(\lambda, m' = |\tilde{C}| + |d|, n' = 1)$

$\sigma, \xi, C \leftarrow\!\!\$\ D_b(\lambda, m, n)$

$\tilde{C}, e, d \leftarrow \text{Garble}(1^\lambda, C)$

$\xi'(y||y') := (\xi(y), \text{Encode}(e, y'))$

$\overline{m} \leftarrow (\tilde{C}, d)$

return $(\sigma, \xi', \overline{m})$

Then we see that by the real game indistinguishability we showed earlier that the distributions D'_0 and D'_1 are $\mathcal{X}'_n$-indistinguishable at least for $\mathcal{X}'_1 = \{1\}$. Then we see that we can immediately apply $\{1\}$-ADE to get an adaptively secure ADGC scheme by encrypting the only element $(\tilde{C}, d)$. □

4.3 Succinct Universal ADE from $\{1\}$-ADE and DCR

We make use of recent advances in low-rate garbled circuits to build an efficient ADE scheme.

Fact 1 (Constant-Rate Label Decomposition Gadget [BLLL23])**.** *There exists an* ***arithmetic garbling scheme*** *(Definition 5) from DCR that takes in an arithmetic label $\tilde{x}$ of size $O(n + \mathsf{poly}(\lambda))$ corresponding to an input x of length m that computes the functionality $\mathbf{m}_x$ for messages of arbitrary length.*

We use a trivial appendix theorem, Theorem 8, to turn any ADGC scheme into a universal ADE scheme of the same cost in conjunction with Fact 1 and the prior Theorem 2 to get a succinct universal ADE scheme by composing the aforementioned statements.

Theorem 3 (Succinct Universal ADE from $\{1\}$-ADE and DCR)**.** *There exists Universal ADE from $\{1\}$-ADE and DCR (Definitions 10 and 11) with online-cost $T(\lambda, m, n) = O(n + \mathsf{poly}(\lambda))$ if $T_{ADE}(\lambda, m, 1) = \mathsf{poly}(\lambda)$.*

Proof. Firstly, Fact 1 gives us a garbling scheme with label cost $|\tilde{x}| = O(n + \mathsf{poly}(\lambda))$. Secondly, we saw from Theorem 2 that given a garbling scheme and $\{1\}$-ADE, we can encrypt the garbled circuit and decoding table as in [BHR12a] to construct an ADGC scheme with online-phase of size $|\tilde{x}| + \mathsf{poly}(\lambda)$, which becomes $O(n + \mathsf{poly}(\lambda))$ when using the construction from Fact 1. Then we make use of a Theorem 8—which shows that ADGC implies universal ADE by simply ignoring hiding for the evaluator's choice—to get a universal ADE scheme of the same online-cost $O(n + \mathsf{poly}(\lambda))$. □

4.4 Succinct $\mathcal{X}_n^{1/2}$-ADOT from $\mathcal{X}_n^{1/2}$-ADE and DCR

We demonstrate how to get $\mathcal{X}_n^{1/2}$-ADOT by applying $\mathcal{X}_n^{1/2}$-ADE as a wrapper over Beaver's preprocessing OT protocol [Bea95] to get a succinctness guarantee that scales only with the online-cost of the ADE scheme.

Construction 4 ($\mathcal{X}_n^{1/2}$-ADOT from $\mathcal{X}_n^{1/2}$-ADE and $\mathcal{X}_n^{1/2}$-OT)**.** *The protocol first invokes Beaver's preprocessing OT [Bea95] and the sender encrypts the output options prescribed from [Bea95] using ADE and sends the ciphertexts. Then in the online-phase the sender finishes the scheme from [Bea95] but decrypts the ciphertexts using ADE instead of sending the openings directly. Figure 2 demonstrates the protocol flow.*

Theorem 4 ($\mathcal{X}_n^{1/2}$-ADOT from $\mathcal{X}_n^{1/2}$-ADE and $\mathcal{X}_n^{1/2}$-OT)**.** *Construction 4 is an $\mathcal{X}_n^{1/2}$-ADOT (Definition 12) from $\mathcal{X}_n^{1/2}$-ADE and selectively secure $\mathcal{X}_n^{1/2}$-OT (Definitions 2 and 11)*

Proof. We first note that correctness of the over protocol comes primarily from beaver's preprocessing OT correctness, since the ADE scheme merely moves the online-phase step to the offline-phase and opens the OT the same way. In particular, $x' \in \mathcal{X}_n^{1/2}$ iff $x \in \mathcal{X}_n^{1/2}$.

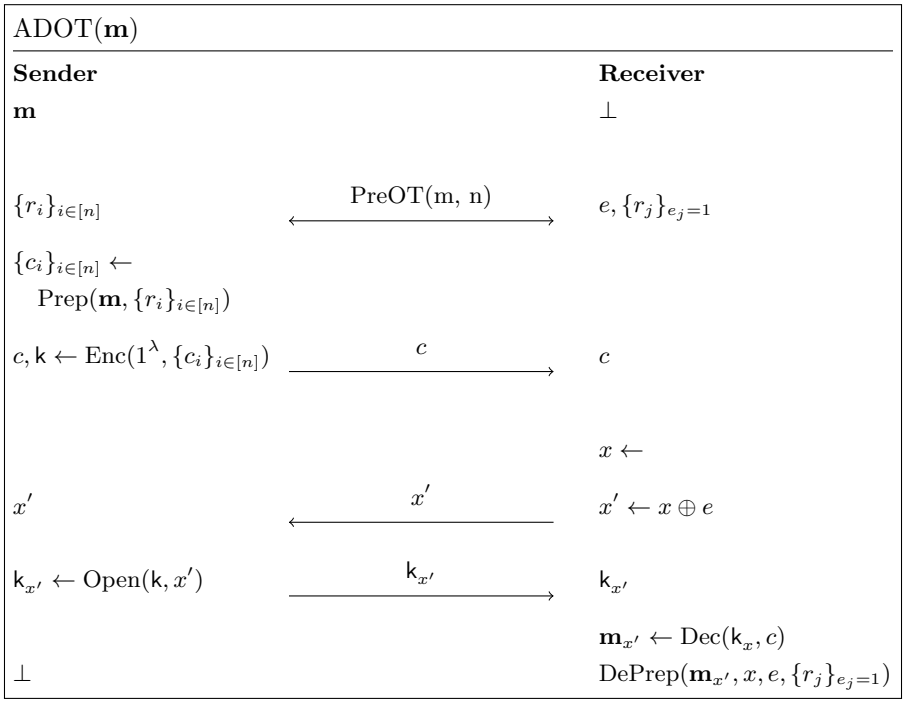

Figure 2: Protocol flow of Construction 4. First recall that n is twice the number of choices, since it is the number of total messages being chosen from. The PreOT step leaves the sender with n one-time pads and the receiver with one from each pair selected by some uniform string e'. e is the expanded version of e' so that $e \in \mathcal{X}_n^{1/2}$ based on choices from e' like a traditional OT scheme. The Prep procedure puts together the messages to send as in the online-phase, preparing for either way that the receiver could choose, by computing the following:

$$\{c_{i||b} \leftarrow (r_{i||b} \oplus [\mathbf{m}]_{i||0}, r_{i||(1-b)} \oplus [\mathbf{m}]_{i||1})\}_{i\in[n/2],b\in\{0,1\}}$$

And of the output message $\mathbf{m}_{x'}$, only the half of each message decided by x is the decrypted part since the other half is garbage. Hence DePrep computes:

$$\{\overline{m}_j \leftarrow ([\mathbf{m}_{x'}]_j)_a \oplus r_{j||b}\}_{x_{j||a}=1=e_{j||b}}$$

Security. If we view Beaver's preprocessing OT as a distribution where the preprocessing step is encoded in σ and the online phase consists of some messages that appear from the other party (the $\mathbf{m}$ distribution), then we find that applying ADE simply moves the online-phase messages to the offline-phase by definition, since the messages are known in-advance. □

Then by applying Theorem 3 we can use Theorem 4 to get a succinct $\mathcal{X}_n^{1/2}$ from our running succinct universal ADE scheme.

Corollary 1 (Succinct $\mathcal{X}_n^{1/2}$-ADOT from $\{1\}$-ADE and DCR)**.** *There exists an $\mathcal{X}_n^{1/2}$-ADOT (Definition 12) from $\{1\}$-ADE and DCR (Definition 10) with a constant number of online rounds and online-cost $T(\lambda, m, n) = O(n + \mathsf{poly}(\lambda))$ if $T_{ADE}(\lambda, m, 1) = \mathsf{poly}(\lambda)$.*

Then note that using Theorems 2 and 15, the latter of which gives a trivial construction of AGC from ADGC with the same online-cost with an added m for a one-time-pad decoding table, to get an AGC scheme from any SGC scheme with non-label online-cost m. Then, by applying Theorem 13 we also get the following.

Corollary 2 (Succinct 2PC from $\{1\}$-ADE and DCR)**.** *There exists 2PC (Definition 7) from $\{1\}$-ADE and DCR (Definition 10) with a constant number of online rounds and online-cost $T(\lambda, C) = 2m + O(n + \mathsf{poly}(\lambda))$ if $T_{ADE}(\lambda, m, 1) = \mathsf{poly}(\lambda)$.*

5 Note on Broader Theorems

The statements of Theorem 3 and Corollaries 1 and 2 can be made more broad than is presented in this work. We gave the theorem and corollary statements for $\{1\}$-ADE schemes with online-cost $T(\lambda, m, 1) = \mathsf{poly}(\lambda)$ to make it more clear where online-cost comes from and to make the chain of theorems easier to follow without needing to get into the specific details of constructions, such as that from [BLLL23] mentioned in Fact 1. In this section we will give the upper-bound for the cost of $\{1\}$-ADE for which the resulting primitives from Theorem 3 and Corollaries 1 and 2 are still succinct.

- **Theorem 3:** The construction given in Fact 1 mostly scales in cost with the so-called decomposition gadget, which takes as input a compressed arithmetic DCR label and decomposes it into Yao-style labels for each bit in the arithmetic label. The decomposition gadget from [BLLL23] scales in $O(w\kappa + \kappa^3)$ a w-bit label and DCR security parameter κ. As such, as long as $T_{ADE}(\lambda, O(mn\kappa + \kappa^3), 1) = o(mn)$ or rather $T_{ADE}(\lambda, m', 1) = o(m')$ the resulting universal ADE scheme is succinct. The resulting scheme would have cost $T(\lambda, m, n) = O(n + T_{ADE}(\lambda, O(mn\kappa + \kappa^3), 1))$
- **Corollaries 1 and 2:** The cost of the constructions used to prove these corollaries scales with Fact 1 in the same way. As such we find that an $\{1\}$-ADE scheme with online-cost $T_{ADE}(\lambda, m, 1) = o(m)$ should suffice. The resulting schemes would have costs

$$T_{ADOT}(\lambda, m, n) = O(n + T_{ADE}(\lambda, O(mn\kappa + \kappa^3), 1))$$

and

$$T_{2PC}(\lambda, m, n) = 2m + O(n + T_{ADE}(\lambda, O(mn\kappa + \kappa^3), 1))$$

respectively.

6 Discussion and Future Work

We presented many constructions in this work that illustrate how interconnected adaptive distributional security is as a framework; however, we feel that there could be even more ways in which the protocols in this work can be more efficient or further reaching. We posit some future directions below.

More ADS Settings. The scope of protocols that we applied the ADS framework to is considerably narrow. We wanted to give meaningful examples of the framework to popular secure computation settings; however, we feel that the general principals explored in this work may be applicable to many areas, in the same way that simulation-based security is ubiquitous.

MPC. We found 2PC to be readily accepting of the ADOT scheme in the place of simulation-based OT (Theorem 13); however, it is not clear how to use the preprocessing step to get a similar scheme for any number of parties. There is already work on the topic of achieving MPC from garbling and OT schemes [BMR90], so it seems like an interesting addition to also consider the case of preprocessing.

Adaptivity Without Modification. In the introduction we mentioned that [BHKO23, GYW+23] are able to achieve adaptive security of various popular garbling schemes in NPROM *without* modifying the original popular schemes. In this work, we also present adaptive garbling, but by making a modification to the underlying scheme. Our approach only require that the garbled material be encrypted, which is still quite practical, but certainly getting security by simply replacing the CPA encryption scheme with {1}-ADE would be more attractive.

We found that taking unmodified Yao's scheme, for example, and attempting to show security from {1}-ADE as the encryption scheme leads to exponential loss in security in the depth of the circuit, similar to the results from [KKPW21] for CPA secure encryption. We wonder if the unmodified Yao's from CPA impossibility result of [KKPW21] can be adapted to show the same for {1}-ADE.

{1}-ADE and the Standard Model. In this work we only relate {1}-ADE to the standard model taxonomy by constructing {1}-ADE from a RO; otherwise, we do not show how {1}-ADE fits into the broader structure of minicrypt. Interestingly, it is trivial to show that $\mathcal{X}_n^{1/2}$-ADE implies OWFs (Theorem 5), and this proof can be adaptive to many succinct $\mathcal{X}_n$-ADE schemes; however, such a proof for {1}-ADE is not as forthcoming. Complexity-based questions of interest include how {1}-ADE relates to OWFs, other hash function primitive — such as collision resistant hashes or corelation robust/resistant hashes — or other encryption schemes — such as SOA-CPA encryption.

Recall that because {1}-ADE is a first-degree assumption, it cannot imply UCE [BHK13] or security in follow-on work by [ST17]. We feel that it would be very interesting if it were to be shown that UCE-security implies {1}-ADE, as, for example this would imply {1}-ADE is incomparable with correlation-resistant hash functions [BHK13].

Succinct ADE from DDH. We chose to give a succinct protocol for $\mathcal{X}_n^{1/2}$-ADE from {1}-ADE and DCR in this work to illustrate how prior work can be utilized in combination with {1}-ADE to achieve state-of-the-art results; however, there may be a bespoke protocol from DDH or other, more common assumptions that achieve similar results. In particular, we felt that the works of [CNs07, GH08, JL09, RKP09, KNP10, GH11, KNP11, Zha11, LLM+17] were exploring methods by which to achieve succinct OT from DDH already; however, these protocols are limited to k out of n simulation-based OT since such setting are the only ones for which succinct simulation-based protocols could exist. Now that the notion of ADOT exists and supports a notion of succinctness for any opening regime, we hope that future efforts can be applied to finding succinct ADOT schemes.

References

[ABK+23] Estuardo Alpirez Bock, Chris Brzuska, Pihla Karanko, Sabine Oechsner, and Kirthivaasan Puniamurthy. Adaptive distributional security for garbling schemes with $\mathcal{O}(|x|)$ online complexity. In Jian Guo and Ron Steinfeld, editors, *ASIACRYPT 2023, Part I*, volume 14438 of *LNCS*, pages 139–171. Springer, Singapore, December 2023.

[AIKW13] Benny Applebaum, Yuval Ishai, Eyal Kushilevitz, and Brent Waters. Encoding functions with constant online rate or how to compress garbled circuits keys. In Ran Canetti and Juan A. Garay, editors, *CRYPTO 2013, Part II*, volume 8043 of *LNCS*, pages 166–184. Springer, Berlin, Heidelberg, August 2013.

[AS16] Prabhanjan Ananth and Amit Sahai. Functional encryption for turing machines. In *Theory of Cryptography: 13th International Conference, TCC 2016-A, Tel Aviv, Israel, January 10-13, 2016, Proceedings, Part I 13*, pages 125–153. Springer, 2016.

[Bea95] Donald Beaver. Precomputing oblivious transfer. In Don Coppersmith, editor, *CRYPTO'95*, volume 963 of *LNCS*, pages 97–109. Springer, Berlin, Heidelberg, August 1995.

[BHK13] Mihir Bellare, Viet Tung Hoang, and Sriram Keelveedhi. Instantiating random oracles via UCEs. In Ran Canetti and Juan A. Garay, editors, *CRYPTO 2013, Part II*, volume 8043 of *LNCS*, pages 398–415. Springer, Berlin, Heidelberg, August 2013.

[BHKO23] Cruz Barnum, David Heath, Vladimir Kolesnikov, and Rafail Ostrovsky. Adaptive garbled circuits and garbled RAM from non-programmable random oracles. Cryptology ePrint Archive, Report 2023/1527, 2023.

[BHR12a] Mihir Bellare, Viet Tung Hoang, and Phillip Rogaway. Adaptively secure garbling with applications to one-time programs and secure outsourcing. In Xiaoyun Wang and Kazue Sako, editors, *ASIACRYPT 2012*, volume 7658 of *LNCS*, pages 134–153. Springer, Berlin, Heidelberg, December 2012.

[BHR12b] Mihir Bellare, Viet Tung Hoang, and Phillip Rogaway. Foundations of garbled circuits. In Ting Yu, George Danezis, and Virgil D. Gligor, editors, *ACM CCS 2012*, pages 784–796. ACM Press, October 2012.

[BHY09] Mihir Bellare, Dennis Hofheinz, and Scott Yilek. Possibility and impossibility results for encryption and commitment secure under selective opening. In Antoine Joux, editor, *EUROCRYPT 2009*, volume 5479 of *LNCS*, pages 1–35. Springer, Berlin, Heidelberg, April 2009.

[BLLL23] Marshall Ball, Hanjun Li, Huijia Lin, and Tianren Liu. New ways to garble arithmetic circuits. In Carmit Hazay and Martijn Stam, editors, *EUROCRYPT 2023, Part II*, volume 14005 of *LNCS*, pages 3–34. Springer, Cham, April 2023.

[BMR90] Donald Beaver, Silvio Micali, and Phillip Rogaway. The round complexity of secure protocols (extended abstract). In *22nd ACM STOC*, pages 503–513. ACM Press, May 1990.

[CGH98] Ran Canetti, Oded Goldreich, and Shai Halevi. The random oracle methodology, revisited (preliminary version). In *30th ACM STOC*, pages 209–218. ACM Press, May 1998.

[CKKZ12] Seung Geol Choi, Jonathan Katz, Ranjit Kumaresan, and Hong-Sheng Zhou. On the security of the "free-XOR" technique. In Ronald Cramer, editor, *TCC 2012*, volume 7194 of *LNCS*, pages 39–53. Springer, Berlin, Heidelberg, March 2012.

[CNs07] Jan Camenisch, Gregory Neven, and abhi shelat. Simulatable adaptive oblivious transfer. In Moni Naor, editor, *EUROCRYPT 2007*, volume 4515 of *LNCS*, pages 573–590. Springer, Berlin, Heidelberg, May 2007.

[DLL24] Marian Dietz, Hanjun Li, and Huijia Lin. How to compress garbled circuit input labels, efficiently. *Cryptology ePrint Archive*, 2024.

[DNRS99] Cynthia Dwork, Moni Naor, Omer Reingold, and Larry J. Stockmeyer. Magic functions. In *40th FOCS*, pages 523–534. IEEE Computer Society Press, October 1999.

[FHKW10] Serge Fehr, Dennis Hofheinz, Eike Kiltz, and Hoeteck Wee. Encryption schemes secure against chosen-ciphertext selective opening attacks. In Henri Gilbert, editor, *EUROCRYPT 2010*, volume 6110 of *LNCS*, pages 381–402. Springer, Berlin, Heidelberg, May / June 2010.

[FLR+10] Marc Fischlin, Anja Lehmann, Thomas Ristenpart, Thomas Shrimpton, Martijn Stam, and Stefano Tessaro. Random oracles with(out) programmability. In Masayuki Abe, editor, *ASIACRYPT 2010*, volume 6477 of *LNCS*, pages 303–320. Springer, Berlin, Heidelberg, December 2010.

[GGP10] Rosario Gennaro, Craig Gentry, and Bryan Parno. Non-interactive verifiable computing: Outsourcing computation to untrusted workers. In Tal Rabin, editor, *CRYPTO 2010*, volume 6223 of *LNCS*, pages 465–482. Springer, Berlin, Heidelberg, August 2010.

[GH08] Matthew Green and Susan Hohenberger. Universally composable adaptive oblivious transfer. In Josef Pieprzyk, editor, *ASIACRYPT 2008*, volume 5350 of *LNCS*, pages 179–197. Springer, Berlin, Heidelberg, December 2008.

[GH11] Matthew Green and Susan Hohenberger. Practical adaptive oblivious transfer from simple assumptions. In Yuval Ishai, editor, *TCC 2011*, volume 6597 of *LNCS*, pages 347–363. Springer, Berlin, Heidelberg, March 2011.

[GKR08] Shafi Goldwasser, Yael Tauman Kalai, and Guy N. Rothblum. One-time programs. In David Wagner, editor, *CRYPTO 2008*, volume 5157 of *LNCS*, pages 39–56. Springer, Berlin, Heidelberg, August 2008.

[GOS18] Sanjam Garg, Rafail Ostrovsky, and Akshayaram Srinivasan. Adaptive garbled RAM from laconic oblivious transfer. In Hovav Shacham and Alexandra Boldyreva, editors, *CRYPTO 2018, Part III*, volume 10993 of *LNCS*, pages 515–544. Springer, Cham, August 2018.

[GS18] Sanjam Garg and Akshayaram Srinivasan. Adaptively secure garbling with near optimal online complexity. In Jesper Buus Nielsen and Vincent Rijmen, editors, *EUROCRYPT 2018, Part II*, volume 10821 of *LNCS*, pages 535–565. Springer, Cham, April / May 2018.

[GYW+23] Xiaojie Guo, Kang Yang, Xiao Wang, Yu Yu, and Zheli Liu. Unmodified half-gates is adaptively secure - so is unmodified three-halves. Cryptology ePrint Archive, Report 2023/1528, 2023.

[HJO+16] Brett Hemenway, Zahra Jafargholi, Rafail Ostrovsky, Alessandra Scafuro, and Daniel Wichs. Adaptively secure garbled circuits from one-way functions. In Matthew Robshaw and Jonathan Katz, editors, *CRYPTO 2016, Part III*, volume 9816 of *LNCS*, pages 149–178. Springer, Berlin, Heidelberg, August 2016.

[HW15] Pavel Hubacek and Daniel Wichs. On the communication complexity of secure function evaluation with long output. In Tim Roughgarden, editor, *ITCS 2015*, pages 163–172. ACM, January 2015.

[IKNP03] Yuval Ishai, Joe Kilian, Kobbi Nissim, and Erez Petrank. Extending oblivious transfers efficiently. In Dan Boneh, editor, *CRYPTO 2003*, volume 2729 of *LNCS*, pages 145–161. Springer, Berlin, Heidelberg, August 2003.

[JL09] Stanislaw Jarecki and Xiaomin Liu. Efficient oblivious pseudorandom function with applications to adaptive OT and secure computation of set intersection. In Omer Reingold, editor, *TCC 2009*, volume 5444 of *LNCS*, pages 577–594. Springer, Berlin, Heidelberg, March 2009.

[JW16] Zahra Jafargholi and Daniel Wichs. Adaptive security of yao's garbled circuits. In *Theory of Cryptography Conference*, pages 433–458. Springer, 2016.

[KKPW21] Chethan Kamath, Karen Klein, Krzysztof Pietrzak, and Daniel Wichs. Limits on the adaptive security of yao's garbling. In Tal Malkin and Chris Peikert, editors, *CRYPTO 2021, Part II*, volume 12826 of *LNCS*, pages 486–515, Virtual Event, August 2021. Springer, Cham.

[KNP10] Kaoru Kurosawa, Ryo Nojima, and Le Trieu Phong. Efficiency-improved fully simulatable adaptive OT under the DDH assumption. In Juan A. Garay and Roberto De Prisco, editors, *SCN 10*, volume 6280 of *LNCS*, pages 172–181. Springer, Berlin, Heidelberg, September 2010.

[KNP11] Kaoru Kurosawa, Ryo Nojima, and Le Trieu Phong. Generic fully simulatable adaptive oblivious transfer. In Javier Lopez and Gene Tsudik, editors, *ACNS 11International Conference on Applied Cryptography and Network Security*, volume 6715 of *LNCS*, pages 274–291. Springer, Berlin, Heidelberg, June 2011.

[KS08] Vladimir Kolesnikov and Thomas Schneider. Improved garbled circuit: Free XOR gates and applications. In Luca Aceto, Ivan Damgård, Leslie Ann Goldberg, Magnús M. Halldórsson, Anna Ingólfsdóttir, and Igor Walukiewicz, editors, *ICALP 2008, Part II*, volume 5126 of *LNCS*, pages 486–498. Springer, Berlin, Heidelberg, July 2008.

[LDL+14] Junzuo Lai, Robert H. Deng, Shengli Liu, Jian Weng, and Yunlei Zhao. Identity-based encryption secure against selective opening chosen-ciphertext attack. In Phong Q. Nguyen and Elisabeth Oswald, editors, *EUROCRYPT 2014*, volume 8441 of *LNCS*, pages 77–92. Springer, Berlin, Heidelberg, May 2014.

[LLM+17] Benoît Libert, San Ling, Fabrice Mouhartem, Khoa Nguyen, and Huaxiong Wang. Adaptive oblivious transfer with access control from lattice assumptions. In Tsuyoshi Takagi and Thomas Peyrin, editors, *ASIACRYPT 2017, Part I*, volume 10624 of *LNCS*, pages 533–563. Springer, Cham, December 2017.

[LP09] Yehuda Lindell and Benny Pinkas. A proof of security of Yao's protocol for two-party computation. *Journal of Cryptology*, 22(2):161–188, April 2009.

[ORV14] Rafail Ostrovsky, Vanishree Rao, and Ivan Visconti. On selective-opening attacks against encryption schemes. In Michel Abdalla and Roberto De Prisco, editors, *SCN 14*, volume 8642 of *LNCS*, pages 578–597. Springer, Cham, September 2014.

[RKP09] Alfredo Rial, Markulf Kohlweiss, and Bart Preneel. Universally composable adaptive priced oblivious transfer. In Hovav Shacham and Brent Waters, editors, *PAIRING 2009*, volume 5671 of *LNCS*, pages 231–247. Springer, Berlin, Heidelberg, August 2009.

[ST17] Pratik Soni and Stefano Tessaro. Public-seed pseudorandom permutations. In Jean-Sébastien Coron and Jesper Buus Nielsen, editors, *EUROCRYPT 2017, Part II*, volume 10211 of *LNCS*, pages 412–441. Springer, Cham, April / May 2017.

[Unr07] Dominique Unruh. Random oracles and auxiliary input. In Alfred Menezes, editor, *CRYPTO 2007*, volume 4622 of *LNCS*, pages 205–223. Springer, Berlin, Heidelberg, August 2007.

[Wic13] Daniel Wichs. Barriers in cryptography with weak, correlated and leaky sources. In Robert D. Kleinberg, editor, *ITCS 2013*, pages 111–126. ACM, January 2013.

[Yao86] Andrew Chi-Chih Yao. How to generate and exchange secrets (extended abstract). In *27th FOCS*, pages 162–167. IEEE Computer Society Press, October 1986.

[Zha11] Bingsheng Zhang. Simulatable adaptive oblivious transfer with statistical receiver's privacy. In Xavier Boyen and Xiaofeng Chen, editors, *ProvSec 2011*, volume 6980 of *LNCS*, pages 52–67. Springer, Berlin, Heidelberg, October 2011.

Appendices

A Relational Theorems

Theorem 5 ($\mathcal{X}_n^{1/2}$-ADE implies One-Time Semantically-Secure Symmetric Key Encryption)**.** *If succinct standard model $\mathcal{X}_n^{1/2}$-ADE (Definition 11) exists, then there exists a symmetric key encryption scheme with succinct keys and one-time semantic security.*

Proof.

Construction. First decide on some method for choosing n from $m = |\overline{m}|$ and λ such that

$$T_{ADE}(\lambda, m/n, 2n) = o(m)$$

Some such choice exists if the ADE scheme is succinct. Then to encrypt $\overline{m}$, for each m/n-length portion, make an additive share of the message portion to end up with $\mathbf{m} = \{[\overline{m}_i]_0, [\overline{m}_i]_1\}_{i\in[m/n]}$. Then encrypt $\mathbf{m}$ using the ADE scheme and let the key be two openings (decided in advance) that jointly open the whole message. To decrypt, simply open the ciphertext twice and then sum each of the shares.

Correctness and Succinctness. Correctness follows immediately from ADE correctness and the correctness of the shares. Note that the openings are asymptotically smaller than the encrypted message, and so sending 2 openings is still asymptotically smaller.

Security. Consider the security of the scheme for two messages $\overline{m}_0$ and $\overline{m}_1$. We can rephrase the encryption portion of the protocol as sampling $\mathbf{m}$ from a distribution D_b that is dependent on $\overline{m}_b$ that creates the block shares and encrypts that. We point out that D_0 and D_1 are $\mathcal{X}_n^{1/2}$-indistinguishable distributions. As such, even if the adversary were to receive any $\mathcal{X}_n^{1/2}$ opening of their choice, which distribution was output would be hidden, which by extension also hides the encrypted message. As such, the adversary cannot distinguish the ciphertexts given no openings whatsoever. □

Theorem 6 (Succinct $\mathcal{X}_n$-ADE implies $\{1\}$-ADE)**.** *Let Π be an $\mathcal{X}_n$-ADE scheme (Definition 11) such that for noticeable n there exist $x_n \in \mathcal{X}_n$ such that $T_{ADE}(\lambda, m, n) = o(mHW(x_n))$ for Hamming weight $HW(x_n)$ of x_n.*[13] *Then Π can be used directly as an $\{1\}$-ADE scheme (Definition 10).*

Proof. We point out that the message of interest $\overline{m}$ can be encoded into $\mathbf{m}_{x_n}$ directly, with all the other portions of $\mathbf{m}$ set to 0. Then the key to open the encryption is just k_{x_n}, which has size $o(|\overline{m}|) = o(mHW(x_n))$ by definition. □

Construction 5 ($\mathcal{X}_n$-ADE from $\mathcal{X}_n$-ADOT)**.** *Let G be a pseudo-random generator with sufficient stretch. Let* $\mathrm{Dummy}_{\mathrm{L}}(\tau)$ *be an interactive procedure that simply responds as the left-hand-party in some interactive protocol according to transcript τ and then outputs $\perp$.*

$\underline{Enc(1^\lambda, \mathbf{m})}$

1: $R_0, R_1 \leftarrow\!\!\$\ \{0,1\}^\lambda$

2: $s_0, r_0, \tau_0 \leftarrow \left\langle Send_0(1^\lambda, \mathbf{m}), Recv_0(1^\lambda; G(R_0)) \right\rangle$

3: **return** $((R_0, \tau_0), (s_0, r_0, R_1))$

[13] In other words, we count the number of 1s in the string x_n, which corresponds to the number of openings.

$\mathit{Open}(\mathsf{k}, x)$

1 : $(s_0, r_0, R_1) \leftarrow \mathsf{k}$

2 : $\_, \_, \tau_1 \leftarrow \langle \mathit{Send}_1(s_0), \mathit{Recv}_1(r_0, x; G(R_1)) \rangle$

3 : **return** (R_1, τ_1)

$\mathit{Dec}(\mathsf{k}_x, x, c)$

1 : $(R_0, \tau_0) \leftarrow c$

2 : $(R_1, \tau_1) \leftarrow \mathsf{k}_x$

3 : $\_, r_0, \_ \leftarrow \left\langle \mathrm{Dummy}_L(\tau_0), \mathit{Recv}_0(1^\lambda; G(R_0)) \right\rangle$

4 : $\_, \mathbf{m}', \_ \leftarrow \langle \mathrm{Dummy}_L(\tau_1), \mathit{Recv}_1(r_0, x; G(R_1)) \rangle$

5 : **return** $\mathbf{m}'$

Theorem 7 ($\mathcal{X}_n$-ADE from $\mathcal{X}_n$-ADOT)**.** *$\mathcal{X}_n$-ADE (Definition 11) can be generally constructed from $\mathcal{X}_n$-ADOT (Definition 12) with online-cost $T(\lambda, m, n) \leq T_{ADOT}(\lambda, m, n) + \lambda$.*

Proof. We start by describing the construction. First note that by Theorem 11 that we have OWFs, since simulation-based OT implies OWFs. We then prove the theorem for Construction 5.

Correctness. Because we only care about security for the sender, it is fine that the sender selects the randomness for the receiver in order to be able to leverage the ADOT scheme to quickly compute a decryption key τ_1. However, to make sure that the ADOT scheme does not function improperly as a result of having non-uniform randomness as input, we use a pseudo-random generator to get around this issue.

Also note that the transcript τ_1 is bounded in size by the online-phase size of the ADOT scheme. If we only send the transcript consisting of sender messages then it could potentially be smaller, which accounts for the inequality in the theorem statement.

Security. The adverary's view in the ADE scheme is the same as the receiver's in the ADOT scheme. Thus, ADE security (Definition 11) of the above scheme holds immediately, since adaptive distributional sender privacy (Definition 12) ensures distributional indistinguishability as well. □

Theorem 8 (Universal ADE from ADGC)**.** *Universal ADE (Definition 11) can be generally constructed from ADGC (Definition 13) with online-cost $T(\lambda, m, n) = T_{ADGC}(\lambda, C')$ for circuit C' that checks $x \in \mathcal{X}_n$ and encodes the output messages projectively from the input.*

Proof. We first give a very short construction of ADE that we will show the theorem for.

Construction. Given messages $\mathbf{m}$, the sender devises a circuit $C'_{\mathbf{m}}(x)$ that simply checks that $x \in \mathcal{X}_n$ and outputs $\mathbf{m}_x$ if in the set, and $\mathbf{m}_{0^n}$ otherwise. Then the sender simply follows the garbling procedure and also acts as the challenger — being send x in plaintext and responding with labels directly.

Correctness. Correctness follows directly from the garbling procedure.

Security. The garbling scheme maintains adaptive distributional privacy (Definition 13), and so the output $\mathbf{m}_x$ is also private just as in the ADE security game (Definition 11). □

Theorem 9 (Universal ADOT from ADGC and $\mathcal{X}_n^{1/2}$-ADOT)**.** *Universal ADOT (Definition 12) for any opening ensemble $\mathcal{X}_n$ can be generally constructed from a projective ADGC and $\mathcal{X}_n^{1/2}$-ADOT (Definitions 12 and 13) with online-cost $T(\lambda, m, n) = T_{ADOT}(\lambda, |\tilde{x}|/n, n) + T_{ADGC}(\lambda, C')$ where C' is the circuit that computes $C'(x) = \mathbf{m}_x$.*

Proof. This construction looks similar to the construction used in the above proof of Theorem 8, except that we will be using $\mathcal{X}_n^{1/2}$-ADOT to make the receiver's input private.

Construction. Given messages $\mathbf{m}$, the sender devises a circuit $C'_{\mathbf{m}}(x)$ that simply outputs $\mathbf{m}_x$. Then the sender simply follows the ADGC garbling procedure and uses ADOT to transfer the input labels from the receiver obliviously.

Correctness. Correctness follows immediately from the ADOT scheme for the labels over the ADGC scheme, and likewise for the ADGC assuming the labels were correctly handed over.

Adaptive Distributional Sender Privacy. Let D_0 and D_1 be $\mathcal{X}_n$-indistinguishable distributions that are the targets of the new $\mathcal{X}_n$-ADOT scheme.

Consider new distributions D'_0 and D'_1 defined below that mimic the above protocol but with an idealized $\mathcal{X}_n^{1/2}$-ADOT functionality.

$D'_b(\lambda, m, n)$

1 : $\sigma, \xi, \mathbf{m} \leftarrow\!\!\$\ D_b(\lambda, m, n)$
2 : $\tilde{C}, e, d \leftarrow \text{Garble}(1^\lambda, C'_{\mathbf{m}})$
3 : $\xi'(y) := (\xi(y), d)$
4 : **return** $((\sigma, \tilde{C}), \xi', e)$

We note that distributions D'_0 and D'_1 are $\mathcal{X}_n^{1/2}$-indistinguishable by ADGC adaptive distributional privacy. As such, the $\mathcal{X}_n^{1/2}$-ADOT maintains adaptive distributional sender privacy.

Also note that because we only reapply distributional security recursively a constant number of times, the multiplicative factor loss is also a constant and so we retain only a polynomial loss in security instead of an exponential loss.

Adaptive Simulation Receiver Privacy. Recall that ADGC is non-interactive except for the use of $\mathcal{X}_n^{1/2}$-ADOT. As such, adaptive simulation receiver privacy reduces to the same privacy guarantee from $\mathcal{X}_n^{1/2}$-ADOT.

□

Construction 6 (ADGC from {1}-ADE and Adaptively Secure Garbled Circuits)**.** *The protocol simply encrypts the decoding table d, sending the ciphertext in the offline-phase. Then the garbler sends the decryption key in the online-phase as the new decryption table.*

Theorem 10 (ADGC from {1}-ADE and Adaptively Secure Garbled Circuits)**.** *Construction 3 is an ADGC scheme (Definition 13) from {1}-ADE and an adaptively secure garbling scheme (Definitions 5 and 10) with online-cost $T(\lambda, C) = T_{ADE}(\lambda, |d|, 1)$ for decoding table d.*

Proof.

Correctness. The construction simply applies encryption d, which is then revealed on schedule, and so the evaluator has the proper material for evaluating the circuit.

Adaptive Distributional Privacy. Let D_0 and D_1 be indistinguishable circuit distributions. Then consider the following statement about the games from Definition 5 for $\sigma_b, \xi_b, C_b \leftarrow\!\$\ D_b$ for $b \in \{0,1\}$ and efficient adversary $\mathcal{A}$ that only queries $\xi_b(x)$ once after the offline-phase is past.

$$\begin{aligned}\text{Real}^{\text{sim-prv}}_{\mathcal{A}^{\sigma_0},\xi_0,C_0}(1^\lambda) &\approx \text{Ideal}^{\text{sim-prv}}_{\mathcal{A}^{\sigma_0},\xi_0,\mathcal{S},C_0}(1^\lambda)\\ \approx \text{Ideal}^{\text{sim-prv}}_{\mathcal{A}^{\sigma_1},\xi_1,\mathcal{S},C_1}(1^\lambda) &\approx \text{Real}^{\text{sim-prv}}_{\mathcal{A}^{\sigma_1},\xi_1,C_1}(1^\lambda)\end{aligned}$$

The jump from Real to Ideal holds due to adaptive simulation privacy, and then jump between the Ideal games is due to the circuit indistinguishability of the distributions D_0 and D_1.

We can then consider viewing the above garbling scheme as a distribution on which $\{1\}$-ADE can apply. Consider distributions D'_0 and D'_1 implemented in the following way.

$D'_b(\lambda, m' = |d|, n' = 1)$

1 : $\sigma, \xi, C \leftarrow\!\$\ D_b(\lambda, m, n)$
2 : $\tilde{C}, e, d \leftarrow \text{Garble}(1^\lambda, C)$
3 : $\xi'(y||y') := (\xi(y), \text{Encode}(e, y'))$
4 : $\overline{m} \leftarrow d$
5 : **return** $(\sigma, \xi', \overline{m})$

Then we see that by the real game indistinguishability we showed earlier that the distributions D'_0 and D'_1 are $\mathcal{X}'_n$-indistinguishable for $\mathcal{X}'_1 = \{1\}$. Then we see that we can immediately apply $\{1\}$-ADE to get an adaptively secure ADGC scheme by encrypting the only element d. □

B Adaptive Simulation from Adaptive Distributional Security

One can recover the simulation-based adaptivity definitions from adaptive distributional security by adding back in the online-phase one-time pad. We formally show this for each related security definition in the following.

Theorem 11 (Adaptive $\mathcal{X}_n$-OT from $\mathcal{X}_n$-ADOT)**.** *Adaptive $\mathcal{X}_n$-OT (Definition 2) can be generally constructed from $\mathcal{X}_n$-ADOT (Definition 12) with online-cost $T(\lambda, m, n) = T_{ADOT}(\lambda, m, n) + mn$.*

Proof. We first give the transformation from ADOT.

Construction. Let $\mathbf{m}$ be the messages that are to be selected from by the receiver. The sender first uniformly samples $\mathbf{m}'$ from $\{0,1\}^{m\times n}$ and then conducts ADOT over $\mathbf{m}'$, then sends $\mathbf{m} \oplus \mathbf{m}'$ in the online-phase afterwards, which the receiver uses to decrypt their messages.

Sender Privacy. We first consider viewing the above protocol as distributions. In the real distribution D_0, there is no side information σ_0, but we have that $\xi_0(x) := \mathbf{m}' \oplus \mathbf{m}$ with messages $\mathbf{m}'$. We compare this to a second distribution D_1 that also has no side information, and sets $\xi_0 := \mathbf{m}''$ with messages $\mathbf{m}_x$ such that $\mathbf{m}''_x = \mathbf{m}'_x \oplus \mathbf{m}_x$ while the other messages are uniform. We find that D_0 and D_1 are perfectly $\mathcal{X}_n$-indistinguishable.

Since the above protocol does not depend at all on the encrypted messages until the one-time pad at the very end, and since ADOT is self-extracting, a simulator can easily query the ideal functionality to obtain $\mathbf{m}_x$ and use the one-time pad in the online-phase to have the adversary decrypt $\mathbf{m}_x$, where the parts of the one-time pad corresponding to unopened messages is set to uniform.

Receiver Privacy. Simulation-based receiver privacy reduces directly to ADOT receiver privacy, because the sender receives the same messages from the receiver in both protocols. □

Theorem 12 (Adaptive Garbled Circuits from ADGC)**.** *Adaptive garbled circuits (Definition 5) can be generally constructed from ADGC (Definition 13) with online-cost* $T(\lambda, C) = T_{ADGC}(\lambda, C') + m$ *for* C' *having potentially* $O(m)$ *more gates*[14]*.*

Proof. We first give the transformation from ADGC.

Construction. Let C be the desired circuit to show security for. Then to garble C first consider a new topology $\Phi(C')$ that can compute both $C(\cdot) \oplus y$ and C'_y for all constants $y \in \{0,1\}^m$, where C' is a circuit that always outputs $y \in \{0,1\}^n$. Then compute $C(\cdot) \oplus y$ using ADGC with topology $\Phi(C')$ and provide y as the last message, which the evaluator can use to decipher the output. Note that $\Phi(C')$ should not add more than $O(m)$ gates, enough to xor the output conditioned on some constant.[15]

Simulation Privacy. Let D_0 and D_1 be distributions that follow the template of indistinguishable circuit distributions (Definition 9) such that D_0 outputs $C(\cdot) \oplus y$ and $\xi(x) = y$ for uniformly sampled y and D_1 outputs C'_y and $\xi(x) = C(x) \oplus y$ for uniformly sampled y. We note that these two distributions are perfectly indistinguishable, and so we find that after garbling using ADGC that they should remain indistinguishable. Since the circuit of D_1 does not depend on x in the online-phase, a simulator can simply run the protocol using circuit C'_y, get the output from the adversary, which is self extracting, and then set the decoding pad to $C(x) \oplus y$.

□

Theorem 13 (Adaptive 2PC from $\mathcal{X}_n^{1/2}$-ADOT and Adaptive GC)**.** *Adaptive 2PC (Definition 7) can be generally constructed from* $\mathcal{X}_n^{1/2}$*-ADOT and AGC (Definitions 5 and 12) with online-cost* $T(\lambda, C) = T_{AGC}(\lambda, C) + T_{ADOT}(\lambda, m', n)$ *where* m' *is the size of labels given by the AGC.*

Proof. We first provide the transformation.

Constructions. The construction is the same as the OT and GC construction for achieving MPC [Yao86, BHR12b], except with ADOT instead of OT. This is to say that OT is conducted over the input wire labels to the garbled circuit, since it should be projective. The labels corresponding to the garbler party's input can also be included in the ADOT, where the garbler party can just send the evaluator party a masked version of their input. For the security we will assume w.l.o.g. that the evaluator party is the only party with input due to the aforementioned masking technique.

Garbler Party Privacy. We start by reinterpreting the garbling privacy security games as indistinguishable circuit distributions (Definition 9). For convenience, we give the games from adaptive simulation privacy in Definition 5 here again.

$\text{Real}^{\text{sim-prv}}_{\mathcal{A},C}(1^\lambda)$	$\text{Ideal}^{\text{sim-prv}}_{\mathcal{A},\mathcal{S},C}(1^\lambda)$
1 : $\tilde{C}_0, e_0, d_0 \leftarrow \text{Garble}(1^\lambda, C)$	1 : $\tilde{C}_1, \tilde{x}_1, S_{\mathcal{S}} \leftarrow \mathcal{S}_1(1^\lambda, C)$
2 : $x, S \leftarrow \mathcal{A}(1^\lambda, \tilde{C})$	2 : $x, S_{\mathcal{A}} \leftarrow \mathcal{A}(1^\lambda, \tilde{C})$
3 : $\tilde{x}_0 \leftarrow \text{Encode}(e_0, x)$	3 : $d_1 \leftarrow \mathcal{S}_2(S_{\mathcal{S}}, C(x))$
4 : **return** $(S, \tilde{x}_0, d_0)$	4 : **return** $(S_{\mathcal{A}}, \tilde{x}, d_1)$

[14] C' is constructed like C but outputs a uniform value that is known in advance, which usually does not require adding more gates for a garbling scheme that hides gates that output constants

[15] In practice, most garbling schemes, and thus ADGC schemes, are able to hide which gate is being garbled, in which case the output gates can either be the true gates or const gates with the output value hardcoded.

Then consider distributions D_0 and D_1 such that D_b outputs $(\sigma_b, \mathbf{m}^b, \xi_b)$ that are sampled in the following way:

$$\begin{aligned}\sigma_b &:= \tilde{C}_b \\ \mathbf{m}^0 &:= e_0 \\ \mathbf{m}^1 &:= \text{Interleave}(\tilde{x}_1) \\ \xi_b(x) &:= d_b\end{aligned}$$

where $\text{Interleave}(\tilde{x})$ interleaves the values of $\tilde{x}$ so that no matter which choices the receiver makes they will always end up with $\tilde{x}$.

Then we see that by garbled circuit simulation privacy that distributions D_0 and D_1 must be $\mathcal{X}_n^{1/2}$-indistinguishable (Definition 8). As such, $\mathcal{X}_n^{1/2}$-ADOT can be used to implement the encoding function.

Evaluator Party Privacy. Note that the garbling step is non-interactive excepting the projective choice step. As such, we find that privacy of evaluator's input x comes down to receiver privacy from ADOT. □

C Adaptive Simulation is Adaptive Distributional Security

We show in this section that adaptive simulation definitions are also adaptively distributional. Admittedly, the following theorems are very straightforward and similar; we include them for completeness.

Theorem 14 (Adaptive $\mathcal{X}_n$-OT is $\mathcal{X}_n$-ADOT)**.** *Adaptive $\mathcal{X}_n$-OT schemes (Definition 2) with sender privacy simulators that do not rewind back to the offline-phase once in the online-phase are also $\mathcal{X}_n$-ADOT schemes (Definition 12).*

Proof.

Adaptive Distributional Sender Privacy. We first note that simulation security should be robust against side information σ and $\xi(y)$, since the adversary could know and run these objects themselves. For distributions D_0 and D_1 that are $\mathcal{X}_n$-indistinguishable, we find that for $\sigma_b, \xi_b, \mathbf{m}^b \leftarrow\!\!\$\ D_b$ the following holds.

$$\begin{aligned}&\{\text{Real}^{\text{sim-ot}}_{\mathcal{A}^{\sigma_0},\xi_0,\mathbf{m}^0}(1^\lambda)\} \approx \{\text{Ideal}^{\text{sim-ot}}_{\mathcal{A}^{\sigma_0},\xi_0,\mathcal{S},\mathbf{m}^0}(1^\lambda)\} \\ \approx &\{\text{Ideal}^{\text{sim-ot}}_{\mathcal{A}^{\sigma_1},\xi_1,\mathcal{S},\mathbf{m}^1}(1^\lambda)\} \approx \{\text{Real}^{\text{sim-ot}}_{\mathcal{A}^{\sigma_1},\xi_1,\mathbf{m}^1}(1^\lambda)\}\end{aligned}$$

for games taken from Definition 2, where the adversary is given σ_b and runs an oracle of ξ_b exactly once after the offline-phase on their input. We use the specific simulation fact here to make sure that the simulator cannot force the adversary to query ξ_b multiple times, since this is not allowed in the distinguishing games for $\mathcal{X}_n$-indistinguishability. Note that the simulated worlds are indistinguishable due to the $\mathcal{X}_n$-indistinguishability of the distributions.

Adaptive Simulation Receiver Privacy. This requirement is the same in both definitions, so it is trivially satisfied. □

Theorem 15 (Adaptive Garbled Circuits is ADGC)**.** *Adaptive garbled circuit schemes (Definition 5) are also ADGC schemes (Definition 13).*

Proof. The proof of this is nearly identical to that of Theorem 14. We write it here for completeness.

Adaptive Distributional Privacy. We note again that simulation security should be robust against side information σ and $\xi(y)$, since the adversary could know and run these objects themselves. For indistinguishable circuit distributions D_0 and D_1, we find that for $\sigma_b, \xi_b, C_b \leftarrow\!\!\$\ D_b$ that the following holds.

$$\begin{aligned}\{\text{Real}^{\text{sim}-\text{prv}}_{\mathcal{A}^{\sigma_0},\xi_0,C_0}(1^\lambda)\} &\approx \{\text{Ideal}^{\text{sim}-\text{prv}}_{\mathcal{A}^{\sigma_0},\xi_0,\mathcal{S},C_0}(1^\lambda)\} \\ \approx \{\text{Ideal}^{\text{sim}-\text{prv}}_{\mathcal{A}^{\sigma_1},\xi_1,\mathcal{S},C_1}(1^\lambda)\} &\approx \{\text{Real}^{\text{sim}-\text{prv}}_{\mathcal{A}^{\sigma_1},\xi_1,C_1}(1^\lambda)\}\end{aligned}$$

for games taken from Definition 5, where the adversary is given σ_b and runs an oracle of ξ_b exactly once after the offline phase on their input. Note that the simulated worlds are indistinguishable due to the circuit indistinguishability of the distributions. □