Delegating Computation via No-Signaling Strategies

Yael Tauman Kalai
Microsoft Research, MIT

January 1, 2018

Abstract

Efficient verification of computation, also known as *delegation of computation*, is one of the most fundamental notions in computer science, and in particular it lies at the heart of the P vs. NP question.

This article contains a high level overview of the evolution of proofs in computer science, and shows how this evolution is instrumental to solving the problem of delegating computation. We highlight a curious connection between the problem of delegating computation and the notion of *no-signaling strategies* from quantum physics.

1 Introduction

The problem of delegating computation considers the setting where a computationally weak device (the client) wishes to offload his computations to a powerful device (the server). Such a client may not trust the server, and would therefore want the server to accompany the result of each computation with an *easy-to-verify proof* of correctness. Clearly, the time it takes to verify such a proof should be significantly lower than the time it takes to do the computation from scratch, since otherwise there is no point of delegating this computation to begin with. At the same time, it is desirable that the time it takes to generate a proof is not too high (i.e., not significantly higher than doing the computation) since otherwise it will be too costly to delegate this computation.

Efficient delegation carries significance to applications. In many cases, computation today is asymmetric, where lightweight computations are done locally, and large computational tasks are performed off-site (e.g. by a cloud server). In addition, complex computations are often delegated to powerful (possibly untrusted) hardware. The ability to verify that the computation is carried out correctly without investing significant computational resources is obviously useful in these situations. The applicability of delegation schemes goes even further. For example, efficient verification of computation is used today as a building block in one of the prominent, and widely used, crypto currencies [BCG+14].

Aside from its practical value, efficient verification of computation is one of the most fundamental notions in computer science. Indeed, one of the most basic computational objects in computer science is the complexity class NP, which is defined as the class of problems whose computation can be verified "efficiently", where an "efficient" computation is defined as one that takes polynomial time.

Unfortunately, the power of the complexity class NP is still unknown, and the question of whether $\mathsf{NP} \neq \mathsf{P}$, which is arguably *the* most important open problem in computer science, remains open.[1] Thus, in a sense, we don't even know if verifying a proof is easier than finding a proof from scratch! Moreover, even under the widely believed assumption that $\mathsf{NP} \neq \mathsf{P}$, it is widely believed that general T-time computations do not have a proof that is verifiable in time significantly smaller than T. This seems to pose an insurmountable barrier to the problem of delegating general purpose computations.

[1] The complexity class P consists of the class of problems that can be computed in polynomial time.

We overcome this barrier by abandoning the traditional (thousands-year-old) notion of a proof being a piece of text, and instead utilize a beautiful and instrumental line of work, motivated by cryptography, where various alternative proof models were proposed and studied. These proof models include *interactive proofs*, *multi-prover interactive proofs* and *probabilistically checkable proofs*, which we elaborate on below.

Jumping ahead, in this article we show how this line of work, together with the use of cryptography and an intriguing connection to *no-signaling strategies* from quantum physics, can be used to construct secure delegation schemes.

Interactive Proofs. The notion of interactive proofs was defined by Goldwasser, Micali, and Rackoff [GMR88]. Their goal was to construct *zero-knowledge proofs*, which intuitively are proofs that reveal no information, beyond the validity of the statement being proven. Goldwasser et. al. noticed that such a notion is not achievable using traditional proofs, and hence they introduced a new proof model, which they called *interactive proofs*.

In contrast to a traditional proof, an interactive proof is an interactive process between a prover and an efficient (i.e., polynomial time) verifier. Interestingly (and oddly), the verifier is allowed to toss coins, and let these coin tosses determine the questions he asks the prover. Importantly, whereas traditionally, it is required that false statements do not have a valid proof, here we require that a (cheating) prover cannot convince the verifier of the correctness of any false statement, except with very small probability (over the verifier's coin tosses). We denote the class of all languages that have an interactive proof by IP. Note that allowing the verifier to be randomized is crucial, since otherwise, the prover can predict the verifier's questions and can send a single document emulating the entire interaction. Thus without the use of randomness we would get $\mathsf{IP} = \mathsf{NP}$.

Interestingly, with the use of randomness, it seems that the class IP is significantly more powerful that the class NP. The celebrated results of [LFKN92, Sha92] prove that $\mathsf{IP} = \mathsf{PSPACE}$, where PSPACE is the class of all languages that can be computed by a Turing machine that uses polynomial space and with arbitrarily long runtime. This class of PSPACE is believed to be significantly larger than NP, and hence the IP proof system seems to be very powerful. Looking into the $\mathsf{IP} = \mathsf{PSPACE}$ theorem more closely, it says that any computation that takes time T and space S has an interactive proof, where the verifier runs in time proportional to S (and the length of the statement being proven). However, the runtime of the prover is significantly higher than T. In the original works of [LFKN92, Sha92] the runtime of the prover was proportional to 2^{S^2}, and thus is super-polynomial (in T) even for log-space computations (i.e., computations that take space $O(\log T)$).[2]

1.1 Our Goal: Doubly-Efficient Proofs

The computation delegation challenge requires not only efficiently verifiable proofs but in actuality *doubly efficiently verifiable proofs* [GKR08]. Such proofs require the complexity of the verifier to be efficient *without paying* a noticeable penalty in increasing the provers running time. This is in contrast to the results of the early 90's, that were focused on the question of which computations have an "easy to verify proof" of correctness, without putting any restriction on the runtime of the prover.

In the most basic setting, in a *delegation scheme* a prover P proves to a verifier V the correctness of an arbitrary time T computation. Our most basic goal is to construct a delegation scheme, with the following three properties.

1. Verifying a proof should be easier than running the computation from scratch, and in particular should take time significantly less than T. Otherwise, the weak device will simply run the computation on its own in the first place.

2. Proving the correctness of a computation should not be "much harder" than running the computation, and in particular should take time at most $\mathsf{poly}(T)$ (for some polynomial poly). Indeed, if proving

[2] We emphasize that almost all (natural) computations require space at least $\log T$, since even holding an index (or pointer) to a location in the computation tableau requires space $\log T$.

requires say an exponential blowup in runtime, then even powerful devices will not be able to prove the correctness of computations.

3. It is impossible to prove the correctness of a false statement (except with very small probability). Otherwise, these proofs will have no meaning. This latter requirement is known as *soundness.*

Unfortunately, achieving these three properties (and even achieving only properties (1) and (3)) simultaneously is widely believed to be impossible. This follows from the fact that $\mathsf{IP} \subseteq \mathsf{PSPACE}$, which implies that delegating computations that require large space (say, space proportional to the runtime), is simply impossible! Nevertheless, one can still consider delegating certain limited classes of computations.

Goldwasser et. al. [GKR08] constructed such a delegation scheme for computations that are computable in "low depth". Intuitively, low depth computations correspond to computations that are highly parallelizable. In [GKR08] it was shown how to delegate any time T and depth D computation, where the runtime of the prover is proportional to D (and the instance size), and the runtime of the prover is $\mathsf{poly}(T)$. Thus, for low-depth computations, this is a doubly-efficient interactive proof. Moreover, it is quite simple and (almost) efficient enough to use in practice. Indeed, many systems based on the [GKR08] blueprint were implemented (for example, [CMT12, TRMP12, Tha13, VSBW13, BTVW14, WHG+16, WJB+17, ZGK+17]), with the goal of using these systems in our day-to-day lives.

The following fundamental problem remains open: Does the $\mathsf{IP} = \mathsf{PSPACE}$ theorem hold if we restrict the prover in the interactive proof to be efficient? Namely, does every T-time S-space computation has an interactive proof where the verifier runs in time proportional to the space S (and the statement length), and the prover runs in time $\mathsf{poly}(T)$?

Significant progress was recently made by Reingold, Rothlum and Rothblum [RRR16], who proved that for every constant $\epsilon > 0$, every T-time S-space computation have an interactive proof where the verifier runs in time proportional to $S \cdot T^{\epsilon}$ (and the statement length), and the prover runs in time $\mathsf{poly}(T)$. But the fundamental problem above remains a very interesting open problem.

In the rest of this article, we focus on the general problem of delegating *any* T-time computation. As we saw, in order to achieve this goal we must depart from the interactive proof model, since this proof model is not powerful enough.

Multi-Prover Interactive Proofs. The notion of multi-prover interactive proofs was defined by Ben-Or, Goldwasser, Kilian and Wigderson [BGKW88]. This notion, similarly to the interactive proof notion, was defined with a cryptographic goal in mind.

Shortly after Goldwasser et. al. [GMR88] introduced the notions of interactive proofs and zero-knowledge proofs, Goldreich, Micali, and Wigderson [GMW87] showed that every interactive proof can be made zero-knowledge, assuming the existence of a one-way function (a function that is easy to compute but hard to invert). The goal of Ben-Or et. al. [BGKW88] was to construct an *information theoretic* zero-knowledge proof, without relying on any computational assumptions. This is believed to be impossible in the interactive proof model, which led them to define the *multi-prover interactive proof* (MIP) model.

In this model, the verifier interacts with two (or more) provers. Importantly, it is assumed that these two provers do not communicate during the protocol. Intuitively, this can be enforced by placing the two provers in different rooms (without any connection to the outside world).

Beyond enabling the construction of information theoretic zero-knowledge proofs, this proof model was proven to be extremely powerful. It was proven by Babai, Fortnow and Lund [BFL91] that any T-time computation can be proved to be correct, using a two-prover interactive proof, where the verifier sends a single query to each prover, and each prover replies with an answer. The queries and answers consist of only $\mathsf{polylog}(T)$ bits, and the runtime of the verifier is $n \cdot \mathsf{polylog}(T)$, where n is the length of the input. Moreover, the runtime of the provers is $\mathsf{poly}(T)$, as desired.

In addition, it was shown that the above holds also for *non-deterministic computations*. In other words, it was shown that any proof of length T can be converted to a 2-prover interactive proof as above where the two queries and two answers are of length $\mathsf{polylog}(T)$. In the language of complexity theory, [BFL91]

proved that MIP = NEXP.[3] Intuitively, the reason this model is so powerful is that it is hard to cheat in a "consistent" manner. Indeed, known 2-prover interactive proof systems consist of a bunch of cross examinations (or consistency checks).

Thus, if we were willing to assume the existence of two non-communicating provers, then we could use these results from the early 90's to construct a delegation scheme, where the client interacts with two servers, and soundness is ensured as long as these two servers do not interact during the proof process. However, we do not want to make such an assumption, since in many applications (such as for crypto-currencies) this is not a realistic assumption, and for other applications (such as cloud computing) the non-communicating assumption may be too strong, or at the very least simply expensive.

Nevertheless, we show how cryptography can be used to emulate two (or more) non-communicating provers using a single prover.

Probabilistically Checkable Proofs. Shortly after this MIP model was introduced, it was noticed that this model is equivalent to the fascinating notion of *probabilistically checkable proofs* (PCP's), which are (non-interactive) proofs that can be verified by reading only a few of their bits. It was observed by Fortnow et. al. [FRS94] that any MIP can be trivially converted into a PCP, by writing down for each prover the answers to all the possible queries of the verifier. Since there are known MIP schemes where the length of each query (and each answer) is $O(\log T)$ the number of possible queries is at most $\mathsf{poly}(T)$, and the size of each answer is at most $O(\log T)$. Thus, this entire list of queries and answers is of length at most $\mathsf{poly}(T)$. Hence, one can verify this proof by running the verifier and sampling a few queries (one for each prover), and reading only the answers corresponding to these queries.

Since this observation, there has been a beautiful line of work (eg., [FGL$^+$91, BFLS91, AS92, ALM$^+$98]), culminating with the remarkable PCP theorem that says that any proof of length T can be converted into a probabilistically checkable one, of length $\mathsf{poly}(T)$, where the verifier needs to read only *three* bits of the proof in order to be convinced that the statement is true with constant probability, and this soundness probability can be amplified by repetition. Moreover, to verify the correctness of the proof the verifier only needs to do a single polynomial time (in the statement size) computation, which is independent of the answers, followed by a single $\mathsf{polylog}(T)$-time computation.

Probabilistically checkable proofs seem very relevant to the problem of delegating computation, since verifying a PCP can be done very efficiently (reading only a few bits of the proof). However, the length of the PCP is $\mathsf{poly}(T)$, and thus even communicating (and storing) this proof is too expensive. If communication and storage were free then indeed PCPs would yield a delegation scheme.

To summarize, despite the beautiful evolution of proofs in computer science starting from the late 80's, it seems that this tremendous progress still does not solve our problem of delegating computation: PCPs require storing a long proof (as long as the computation at hand), multi-prover interactive proofs require assuming two non-communicating provers, and interactive proofs are not general enough to delegate all computations (only bounded space computations). Moreover, as we mentioned, constructing a doubly-efficient interactive proofs for all bounded space computations remains an open problem. Finally, we mention that interactive proofs require many rounds of interaction between the prover and the verifier, and one of the major goals of delegating computation is to obtain non-interactive solutions.

Therefore, in the context of delegating computation, this line of work suffers from significant limitations. Somewhat surprisingly, it has been shown that cryptography can be used to remove many of these limitations.

1.2 Cryptography to the Rescue

It turns out that cryptography can be used to convert any PCP or MIP scheme into a delegation scheme. At first, the use of cryptography may seem quite surprising, since the problem at hand does not seem related to cryptography in any way, since we are not concerned with privacy, only in proving correctness. Nevertheless,

[3] We slightly abuse notation, and throughout this article we denote by MIP the class of all languages that have a multi-prover interactive proof (see Definition 5), and we also denote by MIP any specific multi-prover interactive proof system.

we show how using cryptography one can shrink a long PCP into a short one, and how one can simulate a multi-prover interactive proof via a single prover. To this end, we need to relax the soundness condition, to *computational soundness.*

Computational Soundness. Rather than requiring that it is impossible to prove the validity of a false statement, we require that it is *"practically impossible"* to prove the validity of a false statement. More specifically, we require that it is impossible to prove a false statement only for *computationally-bounded* (e.g., polynomial time) cheating provers. Yet, a computationally all powerful cheating prover may be able to cheat. Honest provers are also required to be efficient (i.e., computationally bounded), in keeping with the philosophy that security should hold against adversaries who are at least as powerful as honest parties. Such proof systems are also known in the literature as *argument systems* [BCC88] or *computationally sound proofs* [Mic94] (as opposed to *statistically sound* proofs that ensure that even a computationally unbounded cheating prover cannot convince a verifier to accept a false statement).

Typically, computational soundness relies on a computational hardness assumption, such as the assumption that it is hard to factor large composite numbers (known as the Factoring Assumption). In this case the soundness guarantee is that if a cheating prover can convince the verifier to accept a false statement (with high probability), then this prover can be used to break the Factoring Assumption. Most of the work in the literature on delegating computation, considers the setting of computational soundness, where we require soundness to hold only against cheating provers who cannot break some underlying cryptographic assumption (such as the Factoring Assumption).

Very loosely speaking, the literature on computation delegation can be partitioned into three categories. The first constructs delegation schemes from any PCP scheme by using the notion of *collision resistant hash functions* to "shrink" the long PCP. The second constructs delegation schemes from any MIP scheme by using cryptography to emulate the many (non-communicating) provers using a single prover. The third uses the notion of *obfuscation* to construct a delegation scheme directly (without using the beautiful evolution of proofs in computer science, summarized above). In this article we focus on the second category. In what follows, we slightly elaborate on the line of work in the first category, and due to lack of space, we do not elaborate on the works in the third category.

Delegation from PCP schemes. Kilian [Kil92] showed how to use a *collision resistant hash function* to convert any PCP scheme into a 4-message delegation scheme for any deterministic (or even non-deterministic) computation.

Many of the applications where a delegation scheme is used (such as in crypto-currencies) require the proof to be *non-interactive*. A non-interactive delegation scheme consists of public parameters (generated honestly by the verifier). These public parameters are used to generate proofs that consist of a *single* message, and soundness holds even if a (cheating) prover chooses the statement to be proved as a function of the public parameters.

Micali [Mic94] proved that a similar approach to the one by Kilian, yields a non-interactive delegation scheme in the so called "Random Oracle Model" [BR93]. Specifically, his scheme uses a hash function, and security is proven assuming the adversary only makes black-box use of this hash function. However, the Random Oracle Model is known to be insecure in general, and there are examples of schemes that are secure in the Random Oracle Model, yet are known to be insecure when the random oracle is replaced with any (succinct) hash function [CGH04, Bar01, GK03].

Since this seminal work of Micali, there has been a long line of followup works (eg., [Gro10, Lip12, DFH12, GGPR13, BCI$^+$13, BCCT13, BCC$^+$14]), constructing a delegation scheme without resorting to the Random Oracle Model. However, these delegation schemes were proven secure under very strong and non-standard "knowledge assumptions". Knowledge assumptions are different from standard complexity assumptions, and (similarly to the Random Oracle Model) they restrict the class of adversaries considered to those which compute things in a certain way.[4]

[4]For example, the Knowledge-of-Exponent assumption [Dam92] assumes that any adversary that given (g, h) computes

Our focus. In this article we focus on the second line of work, which constructs a non-interactive delegation scheme based on a *standard* cryptographic assumption. This line of work is based on a curious connection, noted in [KRR13, KRR14], between the problem of delegating computation and the concept of *no-signaling strategies* from quantum physics.

The starting point is an elegant method introduced by Biehl et. al. [BMW99], for converting any MIP into a 1-round delegation scheme. In what follows, for the sake of simplicity, we describe this method using a *fully homomorphic encryption scheme*, though weaker primitives (such a computational private retrieval scheme) are known to suffice. A fully homomorphic encryption scheme is a secure encryption scheme that allows to do computations on encrypted data (we refer the reader to Section 2.3, and to Definition 8 for the precise definition). Starting from the breakthrough work of Gentry [Gen09] and of Brakerski and Vaikuntanathan [BV11], such homomorphic encryption schemes were constructed based on the Learning with Error Assumption, which is a standard and well established cryptographic assumption.

The [BMW99] method. Loosely speaking, the [BMW99] method takes any MIP scheme and converts it into the following 1-round delegation scheme: The verifier of the delegation scheme computes all the queries for the MIP provers, and sends all these queries to a (single) prover, each encrypted using a different (freshly generated) key corresponding to an FHE scheme. The prover who receives all these encrypted queries, computes for each of the MIP provers its response homomorphically, underneath the layer of the FHE encryption.

This method was considered to be a heuristic, since no proof of soundness was given. The intuition for why this heuristic was believed to be sound is that when a cheating prover answers each of the queries, the other queries are encrypted using different (independently generated) keys, and hence these other queries are completely hidden. Surprisingly, despite this intuition, Dwork et. al. [DLN+04] and Dodis el. al. [DHRW16] showed that this heuristic, in general, is insecure. Intuitively the reason is that the soundness of the MIP is ensured only against cheating provers that answer each query *locally*, only as a function of the corresponding query. In this delegation scheme a cheating prover is not restricted to use local strategies. Rather the security of the FHE scheme ensures that each answer (provided by a cheating prover) does not "signal" information about the other queries, since if it did then we could use this prover to break the security of the FHE scheme.

However, there are strategies that are neither signaling nor local. Such strategies are known in the quantum literature as *no-signaling* strategies (and are formally defined in Section 3.2). The intuition above suggests that these no-signaling strategies are useless. However, in the quantum literature it is well known that this is not necessarily the case.

In a series of work, starting from [KRR13, KRR14], it was proven that if the underlying MIP is sound against (statistically) no-signaling strategies, then the delegation scheme resulting from the [BMW99] heuristic is sound. Moreover, these works constructed for any T-time (deterministic) computation an MIP with (statistical) no-signaling soundness, with communication complexity $\mathsf{polylog}(T)$, and where the runtime of the verifier is $n \cdot \mathsf{polylog}(T)$. This led to the first 1-round delegation scheme for arbitrary (deterministic) computations based on standard cryptographic assumptions. Moreover, these works were later generalized to include RAM computations [KP15], 1-round delegation with adaptive soundness [BHK17], and even generalized to non-deterministic space-bounded computations [BKK+17].

As opposed to the previous line of work, where anyone can verify the proof since all that is needed for verification is the public parameters and the proof, in this line of work the proofs are *privately verifiable*, meaning that in order to verify the proof one needs to know a "secret state" generated together with the public parameters.[5]

In a very recent work, Paneth and Rothblum [PR17] provide a blue-print that generalizes the approach taken in this line of work, to obtain publicly verifiable delegation schemes. However, currently we do not know how to realize this blue-print based on standard cryptographic assumptions.

We mention that the third line of work, that constructs delegation schemes based on obfuscation (e.g., [CHJV15, KLW15, BGL+15, CH16, ACC+16, CCC+16]), achieve public verifiable delegation schemes for

(g^z, h^z), must do so by "first" computing z and then computing (g^z, h^z).

[5] Indeed, the secret keys of the FHE scheme are needed in order to decrypt the answers and verify correctness.

deterministic computations. However, known constructions of obfuscation are built on shaky grounds, and are not known to be secure based on standard assumptions.[6]

The question of constructing a *publicly verifiable* 1-round delegation scheme for general computations under standard assumptions remains a fascinating open question. In addition, the question of constructing a 1-round delegation scheme for general *non-deterministic* computations (beyond space-bounded computations) under standard assumptions (and even under obfuscation type assumptions) remains a fascinating open question.

2 Preliminaries

We model efficient algorithms as probabilistic polynomial time (PPT) algorithms, formally modeled as Turing machines. We denote by $\mathsf{DTIME}(T)$ the class of all the languages that can be computed by a *deterministic* Turing machine that on input x runs in time $T(|x|)$ (i.e., terminates within $T(|x|)$ steps). We denote by $\mathsf{NTIME}(T)$ the class of all the languages that can be computed by a *non-deterministic* Turing machine that on input x runs in time $T(|x|)$.

Throughout this article we use λ to denote the security parameter. This value determines the security level of our schemes. Taking larger values of λ results with better security, though the prover(s) and verifier run in time polynomial in λ, and thus the efficiency of the scheme degrades as we increase λ. The prover(s) and the verifier take as input 1^λ, and the reason we give λ in unary is since we allow our algorithms to run in polynomial time, and we want to allow them to run in time polynomial in λ.

Definition 1. *A function $\nu : \mathbb{N} \to \mathbb{N}$ is said to be negligible if for every polynomial $p : \mathbb{N} \to \mathbb{N}$, there exists a constant $c > 0$ such that for every $\lambda > c$ it holds that $\nu(\lambda) \leq \frac{1}{p(\lambda)}$.*

For a distribution $\mathcal{A}$, we denote by $a \leftarrow \mathcal{A}$ a random variable distributed according to $\mathcal{A}$ (independently of all other random variables).

Definition 2. *Two distribution ensembles $\{\mathcal{X}_\lambda\}_{\lambda\in\mathbb{N}}$ and $\{\mathcal{Y}_\lambda\}_{\lambda\in\mathbb{N}}$ are said to be* computationally indistinguishable *if for every PPT distinguisher $\mathcal{D}$,*

$$|\Pr_{x\leftarrow\mathcal{X}_\lambda}[\mathcal{D}(x)=1] - \Pr_{y\leftarrow\mathcal{Y}_\lambda}[\mathcal{D}(y)=1]| = \mathsf{negl}(\lambda).$$

They are said to be statistically indistinguishable *if the above holds for every (even computationally unbounded) distinguisher $\mathcal{D}$.*

2.1 Delegation Schemes

In what follows, we define the notion of a 1-round delegation scheme and a non-interactive delegation scheme. We require that the first message sent by the verifier does not depend on the statement to be proven. In the literature, this is often not explicitly required, and we add this requirement to the definition since our constructions achieve this desirable property. We define delegation schemes for non-deterministic languages, though we emphasize that this includes also deterministic languages, since any deterministic computation can be thought of as a non-deterministic one where the non-deterministic advice is empty.

Definition 3. *Fix any $T : \mathbb{N} \to \mathbb{N}$ and any $L \in \mathsf{NTIME}(T)$. A 1-round delegation scheme (P, V) for the language L, has the following properties.*

1. **Structure:** *The algorithm V can be partitioned into two PPT algorithms $V = (V_1, V_2)$, where V_1 is a PPT algorithm that generates parameters $(\mathsf{pp}, \mathsf{st}) \leftarrow V(1^\lambda)$. To prove that $x \in L$, upon receiving pp and x, the prover P runs in time $\mathsf{poly}(\lambda, T(|x|))$ and computes $\mathsf{pf} \leftarrow P(x, \mathsf{pp})$. The algorithm V_2 takes as input $(x, \mathsf{pf}, \mathsf{st})$ and outputs a bit, indicating whether he accepts or rejects the proof pf with respect to the public parameters corresponding to his secret state st.*

[6] We mention that these schemes are also not non-interactive, in the sense that soundness holds only if the false statement does not depend on the public parameters.

2. **Completeness:** *For every security parameter* 1^λ*, and every* $x \in L$ *such that* $|x| \leq 2^\lambda$*,*

$$\Pr[V_2(x, \mathsf{pf}, \mathsf{st}) = 1] = 1$$

where the probability is over $(\mathsf{pp}, \mathsf{st}) \leftarrow V_1(1^\lambda)$ *and over* $\mathsf{pf} \leftarrow P(x, \mathsf{pp})$*.*

3. **Soundness:** *For every* PPT *(cheating) prover* $P^* = (P_1^*, P_2^*)$*,*

$$\Pr[V_2(x, \mathsf{pf}, \mathsf{st}) = 1 \ \wedge \ (x \notin L)] = \mathsf{negl}(\lambda)$$

where the probability is over $x \leftarrow P_1^*(1^\lambda)$*, over* $(\mathsf{pp}, \mathsf{st}) \leftarrow V_1(1^\lambda)$ *and over* $\mathsf{pf} \leftarrow P_2^*(x, \mathsf{pp})$*.*

4. **Efficiency:** *The communication complexity is* $\mathsf{poly}(\lambda, \log T(|x|))$*. The honest verifier runs in time* $|x| \cdot \mathsf{polylog}(T(|x|)) + \mathsf{poly}(\lambda, \log T(|x|))$*, and the honest prover runs in time* $\mathsf{poly}(\lambda, T(|x|))$ *(given non-deterministic advice for* $x \in L$*).*

Definition 4. *A* non-interactive *delegation scheme for a language* $L \in \mathsf{NTIME}(T)$*, has the same properties as a 1-round delegation scheme except that the soundness condition is replaced with the following* adaptive soundness *condition:*

Adaptive Soundness: *For every* PPT *(cheating) prover* P^**,*

$$\Pr[V_2(x, \mathsf{pf}, \mathsf{st}) = 1 \ \wedge \ (x \notin L)] = \mathsf{negl}(\lambda)$$

where the probability is over $(\mathsf{pp}, \mathsf{st}) \leftarrow V_1(1^\lambda)$ *and over* $(x, \mathsf{pf}) \leftarrow P^*(\mathsf{pp})$*.*

2.2 Multi-Prover Interactive Proofs

In what follows, we define the notion of a multi-prover interactive proof (MIP). Let L be a language. In a 1-round k-prover interactive proof, $k = k(\lambda)$ provers, $P_1, \ldots, P_k$, try to convince a (probabilistic) verifier V, that $x \in L$. The input x is known to all parties.

In the traditional works on MIP, it was required that the verifier's runtime on input $(1^\lambda, x)$ is at most $\mathsf{poly}(|x|, \lambda)$ and the honest provers' runtime could be unbounded.[7] We change these efficiency requirements to align with the requirements of a delegation scheme. In particular, we require that if $L \in \mathsf{NTIME}(T)$ then the runtime of the verifier is at most $|x| \cdot \mathsf{polylog}(T(|x|)) + \mathsf{poly}(\lambda, \log T(|x|))$ and the runtime of the (honest) provers is at most $\mathsf{poly}(\lambda, T(|x|))$ (assuming they are given the non-deterministic advice for $x \in L$).

The proof consists of only one round. Given a security parameter 1^λ (which determines the soundness), and a random string, the verifier generates $k = k(\lambda)$ queries, $q_1, \ldots, q_k$, one for each prover, and sends them to the k provers. Each prover responds with an answer that depends only on its own individual query. That is, the provers on input x (and associated non-deterministic advice) respond with answers $a_1, \ldots, a_k$, where for every i we have $a_i \leftarrow P_i(x, q_i)$. Finally, the verifier decides wether to accept or reject based on the answers that it receives (as well as the input x and the random string).

Definition 5. *Fix any* $T : \mathbb{N} \to \mathbb{N}$ *and any* $L \in \mathsf{NTIME}(T)$*. We say that* $(V, P_1, \ldots, P_k)$ *is a one-round* k*-prover interactive proof system (*MIP*) for* L *if the following properties are satisfied:*

1. **Structure:** *The verifier consists of two* PPT *algorithms,* $V = (V_1, V_2)$*, where* $(q_1, \ldots, q_k, \mathsf{st}) \leftarrow V_1(1^\lambda)$*. Namely, the queries do not depend on the statement proven.*

2. **Completeness:** *For every security parameter* 1^λ*, and every* $x \in L$ *such that* $|x| \leq 2^\lambda$*,*

$$\Pr\left[V_2(x, q_1, \ldots, q_k, a_1, \ldots, a_k, \mathsf{st}) = 1\right] = 1 - \mathsf{negl}(\lambda),$$

where the probability is over $(q_1, \ldots, q_k, \mathsf{st}) \leftarrow V_1(1^\lambda)$ *and over* $a_i \leftarrow P_i(x, q_i)$ *for every* $i \in [k]$*.*

[7] To be precise, the traditional definition does not even include a security parameter. The verifier is required to run in time $\mathsf{poly}(|x|)$ and soundness is required to hold with constant probability (say $1/2$).

3. **Soundness:** *For every $\lambda \in \mathbb{N}$, every $x \notin L$ (whose size may depend on λ), and any (computationally unbounded, possibly cheating) provers $P_1^*, \ldots, P_k^*$,*

$$\Pr\left[V_2(x, q_1, \ldots, q_k, a_1, \ldots, a_k, \mathsf{st}) = 1\right] = \mathsf{negl}(\lambda),$$

where the probability is over $(q_1, \ldots, q_k, \mathsf{st}) \leftarrow V_1(1^\lambda)$ and over $a_i \leftarrow P^*(x, q_i)$ for every $i \in [k]$.

4. **Efficiency:** *The communication complexity is $\mathsf{poly}(\lambda, \log T)$. The verifier runs in time $|x| \cdot \mathsf{polylog}(T(|x|)) + \mathsf{poly}(\lambda, \log T(|x|))$, and the prover runs in time $\mathsf{poly}(\lambda, T(|x|))$ (assuming he has non-deterministic advice for $x \in L$).*

Theorem 1. *[BFL91] For any $T : \mathbb{N} \to \mathbb{N}$ and any language $L \in \mathsf{NTIME}(T)$, there exists a 2-prover interactive proof (V, P_1, P_2) for L satisfying Definition 5.*

The holy grail of the area of computation delegation, is to achieve the guarantees of Theorem 1 with a *single* prover. Unfortunately, as we mentioned, this dream is too good to be true, since the $\mathsf{IP} = \mathsf{PSPACE}$ theorem says that a single prover can only prove the correctness of bounded space computations. Moreover, known interactive proofs for PSPACE require many rounds, and the class of languages that can be proved via a 1-round interactive proof is widely believed to be quite limited.

In Section 3.1, we present a method first proposed by Biehl et. al. [BMW99], that converts any MIP scheme into a single prover delegation scheme, using the aid of cryptography, and in particular using a computational private information retrieval (PIR) scheme. In this article, for the sake of simplicity, we present this method using a fully homomorphic encryption (FHE) scheme, which is a stronger assumption than a PIR scheme. We chose to present this method using an FHE scheme (as opposed to a PIR scheme) only because we find the terminology to be simpler. We emphasize that all the results presented from now on hold with a PIR scheme as well.

2.3 Fully Homomorphic Encryption (FHE)

We start by defining a *public-key encryption* scheme. Such a scheme consists of three probabilistic polynomial-time algorithms $(\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$, and is defined over some message space $\mathcal{M}$. The key generation algorithm Gen, when given as input a security parameter 1^λ, outputs a pair $(\mathsf{pk}, \mathsf{sk})$ of public and secret keys. The encryption algorithm, Enc, on input a public key pk, and a message $m \in \mathcal{M}$, outputs a ciphertext $\hat{m}$, and the decryption algorithm, Dec, when given the ciphertext $\hat{m}$ and the secret key sk, outputs the original message m (with overwhelming probability).

Definition 6. *A public key encryption over a message space $\mathcal{M}$ consists of three PPT algorithms $(\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ such that for every $m \in \mathcal{M}$,*

$$\Pr[\mathsf{Dec}(\hat{m}, \mathsf{sk}) = m] = 1 - \mathsf{negl}(\lambda),$$

where the probability is over $(\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{Gen}(1^\lambda)$, and over $\hat{m} \leftarrow \mathsf{Enc}(m, \mathsf{pk})$.

Definition 7. *[GM84] A public-key encryption scheme $(\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ is (semantically) secure if for every PPT algorithm $\mathcal{A}$, for every $\lambda \in \mathbb{N}$ and for every two messages $m_1, m_2 \in \mathcal{M}$ such that $|m| = |m'|$,*

$$\left|\Pr\left[\mathcal{A}(\mathsf{pk}, \hat{m}_1) = 1\right] - \Pr\left[\mathcal{A}(\mathsf{pk}, \hat{m}_2) = 1\right]\right| = \mathsf{negl}(\lambda)$$

where the probabilities are over $(\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{Gen}(1^\lambda)$, over $\hat{m}_1 \leftarrow \mathsf{Enc}(m_1, pk)$, and over and over $\hat{m}_2 \leftarrow \mathsf{Enc}(m_2, pk)$.

Definition 8. *A tuple of PPT algorithms $(\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec}, \mathsf{Eval})$ is a* fully-homomorphic encryption scheme *over the message space $\{0,1\}^*$ if $(\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ is a public-key encryption scheme over the message space $\{0,1\}^*$, and in addition the following condition holds:*

Homomorphic Evaluation: Eval *takes as input a public key* pk*, a circuit* $C : \{0,1\}^k \to \{0,1\}^\ell$*, where* $k, \ell \leq \mathsf{poly}(\lambda)$*, and a ciphertext* $\hat{m}$ *that is an encryption of a message* $m \in \{0,1\}^k$ *with respect to* pk*, and outputs a string* ψ *such that for every* $C : \{0,1\}^k \to \{0,1\}^\ell$*, where* $k, \ell \leq \mathsf{poly}(\lambda)$*, and every* $m \in \{0,1\}^k$*,*

$$\Pr[\mathsf{Dec}(\psi, \mathsf{sk}) = C(m)] = 1 - \mathsf{negl}(\lambda),$$

where the probability is over $(\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{Gen}(1^\lambda)$*, over* $\hat{m} \leftarrow \mathsf{Enc}(m, \mathsf{pk})$*, and over* $\psi = \mathsf{Eval}(\mathsf{pk}, C, \hat{m})$*.*

Moreover, the length of ψ *is polynomial in* λ *and* ℓ *(and is independent of the size of* C*).*

Starting from the breakthrough work of Gentry [Gen09], and of Brakerski and Vaikuntanathan [BV11], such homomorphic encryption schemes were constructed based on the standard Learning with Error Assumption [Reg03]. The message space in these constructions is $\mathcal{M} = \{0,1\}$, though one can use these schemes to encrypt any message in $\{0,1\}^*$ by encrypting the message in a bit-by-bit manner.

3 From MIP to Non-Interactive Delegation

Notation. Throughout this section we denote by $k = k(\lambda)$ the number of provers in the MIP scheme. For a vector $a = (a_1, \ldots, a_k)$ and a subset $S \subseteq [k]$, we denote by a_S the sequence of elements of a that are indexed by indices in S, that is, $a_S = (a_i)_{i \in S}$.

3.1 The [BMW99] Heuristic

Biehl et. al. [BMW99] suggested a heuristic for converting any MIP into a 1-round delegation scheme, by using a computational private information retrieval (PIR) scheme. As mentioned above, we present this heuristic using a fully homomorphic encryption (FHE) scheme (see Definition 8).

The Biel et. al. heuristic is natural and elegant. Loosely speaking, the idea is the following: The verifier of the delegation scheme computes all the queries for the MIP provers, and sends all these queries to the (single) prover, each encrypted using an FHE scheme, where each query is encrypted with its own (freshly generated) key. The prover then computes for each of the MIP provers its response homomorphically, underneath the layer of the FHE encryption.

In what follows we give a formal description of the Biehl et. al. heuristic.

The [BMW99] Heuristic. Fix any language L, an MIP scheme $(V, P_1, \ldots, P_k)$ for L, and an FHE scheme $(\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec}, \mathsf{Eval})$. Consider the following 1-round delegation scheme $(P^{\mathsf{del}}, V^{\mathsf{del}})$, where $V^{\mathsf{del}} = (V_1^{\mathsf{del}}, V_2^{\mathsf{del}})$:

- The PPT algorithm V_1^{del} takes as input the security parameter 1^λ, and does the following:
  1. Compute $(q_1, \ldots, q_k, \mathsf{st}) \leftarrow V_1(1^\lambda)$.
  2. Run $\mathsf{Gen}(1^\lambda)$ independently k times to generate $\{(\mathsf{pk}_i, \mathsf{sk}_i)\}_{i \in [k]}$.
  3. For every $i \in [k]$ compute $\hat{q}_i \leftarrow \mathsf{Enc}(q_i, \mathsf{pk}_i)$.
  4. Set $\mathsf{pp}^{\mathsf{del}} = (\hat{q}_1, \ldots, \hat{q}_k)$ and set $\mathsf{st}^{\mathsf{del}} = (\mathsf{sk}_1, \ldots, \mathsf{sk}_k, q_1, \ldots, q_k, \mathsf{st})$.
- The prover $P^{\mathsf{del}}(x, \mathsf{pp}^{\mathsf{del}})$ does the following:
  1. Parse $\mathsf{pp}^{\mathsf{del}} = (\hat{q}_1, \ldots, \hat{q}_k)$.
  2. For every $i \in [k]$ compute $\hat{a}_i \leftarrow \mathsf{Eval}(P_i(x, \cdot), \hat{q}_i)$.
  3. Send $(\hat{a}_1, \ldots, \hat{a}_k)$ to the verifier.
- Upon receiving x and $(\hat{a}_1, \ldots, \hat{a}_k)$, the verifier $V_2^{\mathsf{del}}(x, \hat{a}_1, \ldots, \hat{a}_k, \mathsf{st}^{\mathsf{del}})$ does the following:
  1. Parse $\mathsf{st}^{\mathsf{del}} = (\mathsf{sk}_1, \ldots, \mathsf{sk}_k, q_1, \ldots, q_k, \mathsf{st})$.
  2. For each $i \in [k]$ compute $a_i \leftarrow \mathsf{Dec}(\hat{a}_i, \mathsf{sk}_i)$.

3. Accept if and only if $V_2(x, q_1, \ldots, q_k, a_1, \ldots, a_k, \mathsf{st}) = 1$.

This is a beautiful and natural heuristic. it is easy to see that it satisfies the efficiency and completeness properties of a delegation scheme. The main question is:

Is this Heuristic Sound?

The intuition for why this heuristic was believed to be sound is the following: When a cheating prover answers each of the queries, the other queries are encrypted using different (independently generated) keys, and hence are indistinguishable from encryptions of 0. Therefore, each answer should be indistinguishable from the answer the cheating prover would have provided in the case where the other queries were all 0, and clearly having encryptions of 0 cannot help a prover cheat, since he can generate these encryptions on his own.

Surprisingly, despite this intuition, Dwork et. al. [DLN$^+$04] showed that this heuristic, in general, can be insecure. The reason is that the soundness of the MIP is ensured only against cheating provers that answer each query *locally*, only as a function of the corresponding query. In this delegation scheme a cheating prover is not restricted to use local strategies. Rather the security of the FHE scheme ensures that each answer (provided by a cheating prover) does not "signal" information about the other queries, since if it did then we could use this prover to break the security of the FHE scheme.

However, there are strategies that are neither signaling nor local. Dwork et. al. [DLN$^+$04] refer to such strategies as "spooky interactions". Such strategies are known in the quantum literature as *no-signaling* strategies (defined formally in Section 3.2, below). The intuition above suggests that these no-signaling strategies are useless. However, in the quantum literature it is well known that this is not the case.

Very recently, [DHRW16] showed that indeed the [BMW99] heuristic is insecure! Specifically, they construct an MIP scheme and a FHE scheme, for which when applying the [BMW99] heuristic to these MIP and FHE schemes, the resulting delegation scheme is not sound. To this end, they construct an MIP scheme whose soundness can be broken via a no-signaling strategy, and this no-signaling strategy can be implemented under the layer of the FHE.

3.2 MIPs with No-Signaling Provers

The works of [KRR13, KRR14] attempt to prove the soundness of the [BMW99] heuristic, by considering a variant of the MIP model, where the cheating provers are more powerful.

In the standard MIP model, each prover answers his own query locally, without knowing the queries that were sent to the other provers. The no-signaling model allows each answer to depend on all the queries, as long as for any subset $S \subset [k]$, and any queries q_S for the provers in S, the distribution of the answers a_S, conditioned on the queries q_S, is independent of all the other queries.

Intuitively, this means that the answers a_S do not give the provers in S information about the queries of the provers outside S, except for information that they already have by seeing the queries q_S.

Formally, denote by D the alphabet of the queries and denote by Σ the alphabet of the answers. For every $q = (q_1, \ldots, q_k) \in D^k$, let $\mathcal{A}_q$ be a distribution over Σ^k. We think of $\mathcal{A}_q$ as the(joint) distribution of the answers for queries q.

Definition 9. *We say that the family of distributions $\{\mathcal{A}_q\}_{q \in D^k}$ is* no-signaling *if for every subset $S \subset [k]$ and every two sequences of queries $q, q' \in D^k$, such that $q_S = q'_S$, the following two random variables are identically distributed:*

- *a_S, where $a \leftarrow \mathcal{A}_q$*
- *a'_S where $a' \leftarrow \mathcal{A}_{q'}$*

If the two distributions are computationally (resp. statistically) indistinguishable (see Definition 2), rather than identical, we say that the family of distributions $\{\mathcal{A}_q\}_{q \in D^k}$ is computationally (resp. statistically) no-signaling.

Definition 10. *An* MIP $(V, P_1, \ldots, P_k)$ *for a language* L *is said to be sound against no-signaling strategies (or provers) if the following (more general) soundness property is satisfied:*

Ni-Signaling Soundness: *For every* $\lambda \in \mathbb{N}$*, every* $x \notin L$*, and any no-signaling family of distributions* $\{\mathcal{A}_q\}_{q \in D^k}$*,*

$$\Pr[V_2(x, q_1, \ldots, q_k, a_1, \ldots, a_k, \mathsf{st}) = 1] = \mathsf{negl}(\lambda)$$

where the probability is over $(q_1, \ldots, q_k, \mathsf{st}) \leftarrow V_1(1^\lambda)$ *and over* $(a_1, \ldots, a_k) \leftarrow \mathcal{A}_{(q_1,\ldots,q_k)}$*.*

If this property is satisfied for any computationally (resp. statistically) no-signaling family of distributions $\{\mathcal{A}_q\}_{q \in D^k}$*, we say that the* MIP *has soundness against computationally (resp. statistically) no-signaling strategies.*

No-signaling strategies were first studied in physics in the context of Bell inequalities by Khalfin and Tsirelson [KT85] and Rastall [Ras85], and they gained much attention after they were reintroduced by Popescu and Rohrlich [PR94]. MIPs that are sound against no-signaling provers were extensively studied in the literature (see for example [Ton09, BLM+05, AII06, KKM+08, IKM09, Hol09, Ito10]). We denote the class of MIP's that are sound against no-signaling provers by $\mathsf{MIP}^{\mathsf{NS}}$.

The study of MIPs that are sound against no-signaling provers was originally motivated by the study of MIPs with provers that share entangled quantum states. No-signaling provers are allowed to use arbitrary strategies, as long as their strategies cannot be used for communication between any two disjoint sets of provers. By the physical principle that information cannot travel faster than light, a consequence of Einstein's special relativity theory, it follows that if the provers are placed far enough apart, then the only strategies that can be realized by these provers, even if they share entangled quantum states, are no-signaling strategies.

Moreover, the principle that information cannot travel faster than light is a central principle in physics, and is likely to remain valid in any future ultimate theory of nature, since its violation means that information could be sent from future to past. Therefore, soundness against no-signaling strategies is likely to ensure soundness against provers that obey a future ultimate theory of physics, and not only the current physical theories that we have, that are known to be incomplete.

The study of MIPs that are sound against no-signaling provers is very appealing also because no-signaling strategies have a simple mathematical characterization.

Ito el. al. [IKM09] proved that the set of languages in $\mathsf{MIP}^{\mathsf{NS}}$ contains PSPACE and is contained in EXP. We emphasize that they use the traditional MIP definition, which allows the honest provers to be computationally unbounded, and indeed in their $\mathsf{MIP}^{\mathsf{NS}}$ for PSPACE the provers run in super-polynomial time. Moreover, they assume the verifier runs in time at most $\mathsf{poly}(|x|)$ (which is the traditional requirement).[8] We note that if they used our efficiency requirement where the verifier is allowed to run in time $|x| \cdot \mathsf{polylog}(T) + \mathsf{poly}(\lambda, \log T)$, and the communication complexity is at most $\mathsf{poly}(\lambda, \log T)$, they would get that each $\mathsf{MIP}^{\mathsf{NS}}$ is contained in $\mathsf{DTIME}(T)$.

For the case of *two* provers, Ito [Ito10] showed that the corresponding complexity class is contained in (and therefore equal to) PSPACE. This is in contrast to the class MIP (with soundness against local strategies), which is known to be equal to NEXP.

The connection between MIPs with no-signaling soundness and computation delegation was first observed in [KRR13]. Loosely speaking, they prove that the [BMW99] heuristic is sound when applied to any MIP that is secure against *statistically* no-signaling strategies, denoted by $\mathsf{MIP}^{\mathsf{sNS}}$.[9] In [KRR13, KRR14] they also characterize the exact power of MIPs that are secure against statistically no-signaling provers, and prove that $\mathsf{MIP}^{\mathsf{sNS}} = \mathsf{EXP}$. More specifically, they prove the following theorem.

Theorem 2. *[KRR13, KRR14] For any* $T : \mathbb{N} \to \mathbb{N}$*, and any language in* $L \in \mathsf{DTIME}(T)$*, there exists an* MIP *with statistical no-signaling soundness (as in defined in Definitions 5 and 10).*

[8] They show that one can find the best no-signaling strategy for the provers by solving an exponential (in $|x|$) size linear program.

[9] Their result relies on the stronger assumption that the underlying FHE is not only secure against PPT adversaries, but also again quasi-polynomial time adversaries.

In particular, these works prove the following theorem.

Theorem 3. *[KRR13, KRR14] For any $T : \mathbb{N} \to \mathbb{N}$ and any $L \in \mathsf{DTIME}(T)$ there exists a 1-round delegation scheme for L (as defined in Definition 3), assuming the existence of an FHE scheme that is secure against quasi-polynomial time adversaries.*

To achieve non-interactive delegation (as opposed to 1-round delegation) we need to use an MIP scheme that is sound against *adaptive* no-signaling strategies, as defined in [BHK17].

Definition 11. *An MIP $(V, P_1, \ldots, P_k)$ for a language L is said to be* adaptively *sound against no-signaling strategies (or provers) if the following adaptive soundness property is satisfied:*

Adaptive Soundness: *For every $\lambda \in \mathbb{N}$ and any no-signaling family of distributions $\{\mathcal{A}_q\}_{q \in D^k}$,*

$$\Pr\left[V_2(x, q_1, \ldots, q_k, a_1, \ldots, a_k, \mathsf{st}) = 1\right] = \mathsf{negl}(\lambda)$$

where the probability is over $(q_1, \ldots, q_k, \mathsf{st}) \leftarrow V_1(1^\lambda)$ and over $(x, a_1, \ldots, a_k) \leftarrow \mathcal{A}_{(q_1,\ldots,q_k)}$, where x should be thought of as corresponding to an additional (dummy) query q_0, and thus should signal no information about the other queries $q_1, \ldots, q_k$.

If this property is satisfied for any computationally (resp. statistically) no-signaling family of distributions $\{\mathcal{A}_q\}_{q \in D^k}$, we say that the MIP has adaptive soundness against computationally (resp. statistically) no-signaling strategies.

We denote the class of MIP scheme that have adaptive soundness against computational no-signaling strategies by $\mathsf{MIP}^{\mathsf{adaptive-cNS}}$. Brakerski et. al. [BHK17] proved that $\mathsf{MIP}^{\mathsf{adaptive-cNS}} = \mathsf{EXP}$. More specifically, they prove the following theorem, which is a strengthening of Theorem 2.

Theorem 4. *For any $T : \mathbb{N} \to \mathbb{N}$, and any language in $L \in \mathsf{DTIME}(T)$, there exists an MIP with adaptive computational no-signaling soundness (as in defined in Definitions 5 and 11).*

In addition, they proved that applying the [BMW99] heuristic to any MIP that is *adaptively* sound against computational no-signaling strategies, results with a *non-interactive delegation scheme* that is sound assuming the standard (PPT) security of the underlying FHE scheme.

Theorem 5. *[BHK17] For every $T : \mathbb{N} \to \mathbb{N}$ and for every $L \in \mathsf{DTIME}(T)$ there exists a non-interactive delegation scheme for L (as defined in Definition 4), assuming the existence of an FHE scheme.*

Due to lack of space we do not provide any intuition behind the proof of Theorem 4, and instead provide a proof sketch of Theorem 5 (assuming Theorem 4).

Proof Sketch of Theorem 5. Fix an FHE scheme $(\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec}, \mathsf{Eval})$, a time bound $T = T(\lambda)$, and a language $L \in \mathsf{DTIME}(T)$. Let

$$\mathsf{MIP}^{\mathsf{adaptive-cNS}} = (V, P_1, \ldots, P_k)$$

be an MIP for L with adaptive soundness against computationally no-signaling strategies. The existence of such a proof system for L follows from Theorem 4.

The non-interactive delegation scheme, denoted by $(V^{\mathsf{del}}, P^{\mathsf{del}})$ is the one obtained by applying the [BMW99] heuristic to $\mathsf{MIP}^{\mathsf{adaptive-cNS}}$ and FHE.

Suppose for contradiction that there exists a cheating prover P^* such that for infinitely many $\lambda \in \mathbb{N}$,

$$\Pr[V_2^{\mathsf{del}}(x, \mathsf{pf}, \mathsf{st}) = 1 \ \wedge \ (x \notin L)] \geq \frac{1}{\mathsf{poly}(\lambda)}$$

where the probability is over $(\mathsf{pp}, \mathsf{st}) \leftarrow V_1^{\mathsf{del}}(1^\lambda)$ and over $(x, \mathsf{pf}) \leftarrow P^*(\mathsf{pp})$.

We use P^* to construct an adaptive computational no-signaling strategy that contradicts the adaptive soundness condition of $\mathsf{MIP}^{\mathsf{adaptive-cNS}}$.

To this end, for every possible set of queries $q = (q_1, \ldots, q_k)$, consider the distribution of answers $\mathcal{A}_q$ defined as follows:

1. For every $i \in [k]$ sample $(\mathsf{pk}_i, \mathsf{sk}_i) \leftarrow \mathsf{Gen}(1^\lambda)$.
2. For every $i \in [k]$ sample $\hat{q}_i \leftarrow \mathsf{Enc}(q_i, \mathsf{pk}_i)$.
3. Let $\mathsf{pp} = (\hat{q}_1, \ldots, \hat{q}_k)$.
4. Compute $(x, \mathsf{pf}) \leftarrow P^*(\mathsf{pp})$.
5. Parse $\mathsf{pf} = (\hat{a}_1, \ldots, \hat{a}_k)$.
6. For every $i \in [k]$ decrypt $a_i \leftarrow \mathsf{Dec}(\hat{a}_i, \mathsf{sk}_i)$.
7. Output $(x, a_1, \ldots, a_k)$.

To reach a contradiction it remains to argue that the strategy $\{\mathcal{A}_q\}$ is computationally no-signaling. This follows from the security of the underlying FHE scheme. We omit the proof due to lack of space.

References

[ACC+16] P. Ananth, Y. Chen, K. Chung, H. Lin, and W. Lin. Delegating RAM Computations with Adaptive Soundness and Privacy. In *Theory of Cryptography - 14th International Conference, TCC 2016-B, Beijing, China, October 31 - November 3, 2016, Proceedings, Part II*, pages 3–30, 2016.

[ALM+98] S. Arora, C. Lund, R. Motwani, M. Sudan, and M. Szegedy. Proof Verification and the Hardness of Approximation Problems. *J. ACM*, 45(3):501–555, 1998.

[AS92] S. Arora and S. Safra. Probabilistic Checking of Proofs; A New Characterization of NP. In *33rd Annual Symposium on Foundations of Computer Science, Pittsburgh, Pennsylvania, USA, 24-27 October 1992*, pages 2–13, 1992.

[AII06] D. Avis, H. Imai, and T. Ito. On the relationship between convex bodies related to correlation experiments with dichotomic observables. *Journal of Physics A: Mathematical and General, 39(36)*, 39(36):11283, 2006.

[BFLS91] L. Babai, L. Fortnow, L. A. Levin, and M. Szegedy. Checking Computations in Polylogarithmic Time. In *Proceedings of the 23rd Annual ACM Symposium on Theory of Computing*, pages 21–31, 1991.

[BFL91] L. Babai, L. Fortnow, and C. Lund. Non-Deterministic Exponential Time has Two-Prover Interactive Protocols. *Computational Complexity*, 1:3–40, 1991.

[BKK+17] S. Badrinarayanan, Y. T. Kalai, D. Khurana, A. Sahai, and D. Wichs. Non-Interactive Delegation for Low-Space Non-Deterministic Computation. Cryptology ePrint Archive, Report 2017/1250, 2017. `https://eprint.iacr.org/2017/1250`.

[Bar01] B. Barak. How to Go Beyond the Black-Box Simulation Barrier. In *FOCS*, pages 106–115, 2001.

[BLM+05] J. Barrett, N. Linden, S. Massar, S. Pironio, S. Popescu, and D. Roberts. Nonlocal correlations as an information-theoretic resource. *Physical Review A, 71(022101)*, 71(2):022101, 2005.

[BR93] M. Bellare and P. Rogaway. Random Oracles are Practical: A Paradigm for Designing Efficient Protocols. In D. E. Denning, R. Pyle, R. Ganesan, R. S. Sandhu, and V. Ashby, editors, *ACM Conference on Computer and Communications Security*, pages 62–73. ACM, 1993.

[BGKW88] M. Ben-Or, S. Goldwasser, J. Kilian, and A. Wigderson. Multi-Prover Interactive Proofs: How to Remove Intractability Assumptions. In *Proceedings of the 20th Annual ACM Symposium on Theory of Computing*, pages 113–131, 1988.

[BCG+14] E. Ben-Sasson, A. Chiesa, C. Garman, M. Green, I. Miers, E. Tromer, and M. Virza. Zerocash: Decentralized Anonymous Payments from Bitcoin. In *2014 IEEE Symposium on Security and Privacy, SP 2014, Berkeley, CA, USA, May 18-21, 2014*, pages 459–474, 2014.

[BMW99] I. Biehl, B. Meyer, and S. Wetzel. Ensuring the Integrity of Agent-Based Computations by Short Proofs. In *Proceedings of the Second International Workshop on Mobile Agents*, MA '98, pages 183–194, London, UK, UK, 1999. Springer-Verlag.

[BCC+14] N. Bitansky, R. Canetti, A. Chiesa, S. Goldwasser, H. Lin, A. Rubinstein, and E. Tromer. The Hunting of the SNARK. *IACR Cryptology ePrint Archive*, 2014:580, 2014.

[BCCT13] N. Bitansky, R. Canetti, A. Chiesa, and E. Tromer. Recursive composition and bootstrapping for SNARKS and proof-carrying data. In D. Boneh, T. Roughgarden, and J. Feigenbaum, editors, *Symposium on Theory of Computing Conference, STOC'13, Palo Alto, CA, USA, June 1-4, 2013*, pages 111–120. ACM, 2013.

[BCI+13] N. Bitansky, A. Chiesa, Y. Ishai, R. Ostrovsky, and O. Paneth. Succinct Non-interactive Arguments via Linear Interactive Proofs. In *TCC*, pages 315–333, 2013.

[BGL+15] N. Bitansky, S. Garg, H. Lin, R. Pass, and S. Telang. Succinct Randomized Encodings and their Applications. *IACR Cryptology ePrint Archive*, 2015:356, 2015.

[BTVW14] A. J. Blumberg, J. Thaler, V. Vu, and M. Walfish. Verifiable computation using multiple provers. *IACR Cryptology ePrint Archive*, 2014:846, 2014.

[BRF13] D. Boneh, T. Roughgarden, and J. Feigenbaum, editors. *Symposium on Theory of Computing Conference, STOC'13, Palo Alto, CA, USA, June 1-4, 2013*. ACM, 2013.

[BHK17] Z. Brakerski, J. Holmgren, and Y. T. Kalai. Non-interactive delegation and batch NP verification from standard computational assumptions. In *Proceedings of the 49th Annual ACM SIGACT Symposium on Theory of Computing, STOC 2017, Montreal, QC, Canada, June 19-23, 2017*, pages 474–482, 2017.

[BV11] Z. Brakerski and V. Vaikuntanathan. Efficient Fully Homomorphic Encryption from (Standard) LWE. In *IEEE 52nd Annual Symposium on Foundations of Computer Science, FOCS 2011, Palm Springs, CA, USA, October 22-25, 2011*, pages 97–106, 2011.

[BCC88] G. Brassard, D. Chaum, and C. Crépeau. Minimum Disclosure Proofs of Knowledge. *J. Comput. Syst. Sci.*, 37(2):156–189, 1988.

[CGH04] R. Canetti, O. Goldreich, and S. Halevi. The random oracle methodology, revisited. *J. ACM*, 51(4):557–594, 2004.

[CH16] R. Canetti and J. Holmgren. Fully Succinct Garbled RAM. In *ITCS*, pages 169–178. ACM, 2016.

[CHJV15] R. Canetti, J. Holmgren, A. Jain, and V. Vaikuntanathan. Succinct Garbling and Indistinguishability Obfuscation for RAM Programs. In *STOC*, pages 429–437. ACM, 2015.

[CCC+16] Y. Chen, S. S. M. Chow, K. Chung, R. W. F. Lai, W. Lin, and H. Zhou. Cryptography for Parallel RAM from Indistinguishability Obfuscation. In *ITCS*, pages 179–190. ACM, 2016.

[CMT12] G. Cormode, M. Mitzenmacher, and J. Thaler. Practical verified computation with streaming interactive proofs. In S. Goldwasser, editor, *Innovations in Theoretical Computer Science 2012, Cambridge, MA, USA, January 8-10, 2012*, pages 90–112. ACM, 2012.

[Dam92] I. Damgård. Towards Practical Public Key Systems Secure Against Chosen Ciphertext Attacks. In *Proceedings of CRYPTO91*, pages 445–456, 1992.

[DFH12] I. Damgård, S. Faust, and C. Hazay. Secure Two-Party Computation with Low Communication. In *Theory of Cryptography - 9th Theory of Cryptography Conference, TCC 2012, Taormina, Sicily, Italy, March 19-21, 2012. Proceedings*, pages 54–74, 2012.

[DHRW16] Y. Dodis, S. Halevi, R. D. Rothblum, and D. Wichs. Spooky Encryption and Its Applications. In *Advances in Cryptology - CRYPTO 2016 - 36th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 14-18, 2016, Proceedings, Part III*, pages 93–122, 2016.

[DLN+04] C. Dwork, M. Langberg, M. Naor, K. Nissim, and O. Reingold. Succinct Proofs for NP and Spooky Interactions. Unpublished manuscript, available at `http://www.cs.bgu.ac.il/~kobbi/papers/spooky_sub_crypto.pdf`, 2004.

[FGL+91] U. Feige, S. Goldwasser, L. Lovász, S. Safra, and M. Szegedy. Approximating Clique is Almost NP-Complete (Preliminary Version). In *FOCS*, pages 2–12. IEEE Computer Society, 1991.

[FRS94] L. Fortnow, J. Rompel, and M. Sipser. On the Power of Multi-Prover Interactive Protocols. *Theor. Comput. Sci.*, 134(2):545–557, 1994.

[GGPR13] R. Gennaro, C. Gentry, B. Parno, and M. Raykova. Quadratic Span Programs and Succinct NIZKs without PCPs. In *Advances in Cryptology - EUROCRYPT 2013, 32nd Annual International Conference on the Theory and Applications of Cryptographic Techniques, Athens, Greece, May 26-30, 2013. Proceedings*, pages 626–645, 2013.

[Gen09] C. Gentry. Fully homomorphic encryption using ideal lattices. In *Proceedings of the 41st Annual ACM Symposium on Theory of Computing, STOC 2009, Bethesda, MD, USA, May 31 - June 2, 2009*, pages 169–178, 2009.

[GMW87] O. Goldreich, S. Micali, and A. Wigderson. How to play ANY mental game. In *STOC '87: Proceedings of the nineteenth annual ACM symposium on Theory of computing*, pages 218–229, 1987.

[GK03] S. Goldwasser and Y. T. Kalai. On the (In)security of the Fiat-Shamir Paradigm. In *FOCS*, pages 102–, 2003.

[GKR08] S. Goldwasser, Y. T. Kalai, and G. N. Rothblum. Delegating computation: interactive proofs for muggles. In *STOC*, pages 113–122, 2008.

[GM84] S. Goldwasser and S. Micali. Probabilistic Encryption. *Journal of Computer and System Sciences*, 28(2):270–299, 1984.

[GMR88] S. Goldwasser, S. Micali, and R. L. Rivest. A Digital Signature Scheme Secure Against Adaptive Chosen-Message Attacks. *SIAM J. Comput.*, 17(2):281–308, 1988.

[Gro10] J. Groth. Short Pairing-Based Non-interactive Zero-Knowledge Arguments. In *ASIACRYPT*, volume 6477 of *Lecture Notes in Computer Science*, pages 321–340. Springer, 2010.

[Hol09] T. Holenstein. Parallel Repetition: Simplification and the No-Signaling Case. *Theory of Computing*, 5(1):141–172, 2009.

[Ito10] T. Ito. Polynomial-Space Approximation of No-Signaling Provers. In *ICALP (1)*, pages 140–151, 2010.

[IKM09] T. Ito, H. Kobayashi, and K. Matsumoto. Oracularization and Two-Prover One-Round Interactive Proofs against Nonlocal Strategies. In *IEEE Conference on Computational Complexity*, pages 217–228, 2009.

[KP15] Y. T. Kalai and O. Paneth. Delegating RAM Computations. *IACR Cryptology ePrint Archive*, 2015:957, 2015.

[KRR13] Y. T. Kalai, R. Raz, and R. D. Rothblum. Delegation for bounded space. In D. Boneh, T. Roughgarden, and J. Feigenbaum, editors, *Symposium on Theory of Computing Conference, STOC'13, Palo Alto, CA, USA, June 1-4, 2013*, pages 565–574. ACM, 2013.

[KRR14] Y. T. Kalai, R. Raz, and R. D. Rothblum. How to delegate computations: the power of no-signaling proofs. In *STOC*, pages 485–494. ACM, 2014.

[KKM+08] J. Kempe, H. Kobayashi, K. Matsumoto, B. Toner, and T. Vidick. Entangled Games are Hard to Approximate. In *FOCS*, pages 447–456, 2008.

[KT85] L. A. Khalfin and B. S. Tsirelson. Quantum and quasi-classical analogs of Bell inequalities. In *In Symposium on the Foundations of Modern Physics*, pages 441–460, 1985.

[Kil92] J. Kilian. A Note on Efficient Zero-Knowledge Proofs and Arguments (Extended Abstract). In *STOC*, pages 723–732, 1992.

[KLW15] V. Koppula, A. B. Lewko, and B. Waters. Indistinguishability Obfuscation for Turing Machines with Unbounded Memory. In *STOC*, pages 419–428. ACM, 2015.

[Lip12] H. Lipmaa. Progression-Free Sets and Sublinear Pairing-Based Non-Interactive Zero-Knowledge Arguments. In *TCC*, pages 169–189, 2012.

[LFKN92] C. Lund, L. Fortnow, H. J. Karloff, and N. Nisan. Algebraic Methods for Interactive Proof Systems. *J. ACM*, 39(4):859–868, 1992.

[Mic94] S. Micali. CS Proofs (Extended Abstracts). In *35th Annual Symposium on Foundations of Computer Science, Santa Fe, New Mexico, USA, 20-22 November 1994*, pages 436–453. IEEE Computer Society, 1994. Full version in [Mic00].

[Mic00] S. Micali. Computationally Sound Proofs. *SIAM J. Comput.*, 30(4):1253–1298, 2000.

[PR17] O. Paneth and G. N. Rothblum. On Zero-Testable Homomorphic Encryption and Publicly Verifiable Non-interactive Arguments. In *Theory of Cryptography - 15th International Conference, TCC 2017, Baltimore, MD, USA, November 12-15, 2017, Proceedings, Part II*, pages 283–315, 2017.

[PR94] S. Popescu and D. Rohrlich. Quantum nonlocality as an axiom. *Foundations of Physics*, 24(3):379–385, 1994.

[Ras85] P. Rastall. Locality, Bell's theorem, and quantum mechanics. *Foundations of Physics*, 15(9):963–972, 1985.

[Reg03] O. Regev. New lattice based cryptographic constructions. In *Proceedings of the 35th Annual ACM Symposium on Theory of Computing*, pages 407–416, 2003.

[RRR16] O. Reingold, G. N. Rothblum, and R. D. Rothblum. Constant-round interactive proofs for delegating computation. In *Proceedings of the 48th Annual ACM SIGACT Symposium on Theory of Computing, STOC 2016, Cambridge, MA, USA, June 18-21, 2016*, pages 49–62, 2016.

[Sha92] A. Shamir. IP = PSPACE. *Journal of the ACM*, 39(4):869–877, 1992.

[Tha13] J. Thaler. Time-Optimal Interactive Proofs for Circuit Evaluation. In *Advances in Cryptology - CRYPTO 2013 - 33rd Annual Cryptology Conference, Santa Barbara, CA, USA, August 18-22, 2013. Proceedings, Part II*, pages 71–89, 2013.

[TRMP12] J. Thaler, M. Roberts, M. Mitzenmacher, and H. Pfister. Verifiable Computation with Massively Parallel Interactive Proofs. In *4th USENIX Workshop on Hot Topics in Cloud Computing, HotCloud'12, Boston, MA, USA, June 12-13, 2012*, 2012.

[Ton09] B. Toner. Monogamy of non-local quantum correlations. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Science*, 465(2101):59–69, 2009.

[VSBW13] V. Vu, S. T. V. Setty, A. J. Blumberg, and M. Walfish. A Hybrid Architecture for Interactive Verifiable Computation. In *2013 IEEE Symposium on Security and Privacy, SP 2013, Berkeley, CA, USA, May 19-22, 2013*, pages 223–237, 2013.

[WHG+16] R. S. Wahby, M. Howald, S. J. Garg, A. Shelat, and M. Walfish. Verifiable ASICs. In *IEEE Symposium on Security and Privacy, SP 2016, San Jose, CA, USA, May 22-26, 2016*, pages 759–778, 2016.

[WJB+17] R. S. Wahby, Y. Ji, A. J. Blumberg, A. Shelat, J. Thaler, M. Walfish, and T. Wies. Full accounting for verifiable outsourcing. *IACR Cryptology ePrint Archive*, 2017:242, 2017.

[ZGK+17] Y. Zhang, D. Genkin, J. Katz, D. Papadopoulos, and C. Papamanthou. vSQL: Verifying Arbitrary SQL Queries over Dynamic Outsourced Databases. In *2017 IEEE Symposium on Security and Privacy, SP 2017, San Jose, CA, USA, May 22-26, 2017*, pages 863–880, 2017.