New Perspectives on Zero-Knowledge Multi-Prover Interactive Proofs

Claude Crépeau[1] ⋆ and Nan Yang[2] ⋆⋆

[1] McGill University, Montréal, Québec, Canada. crepeau@cs.mcgill.ca
[2] Concordia University, Montreal, Quebec, Canada. na_yan@encs.concordia.ca

Abstract. In multi-prover interactive proofs (MIPs), the verifier can provide non-local resources for the provers intrinsically. In most cases, this is undesirable. Existing proofs of soundness do not account for the verifier's non-local potential. We show that this may be a problem for many MIPs. We provide a solution by constructing a generalization of the MIP model, of which standard MIPs are a special case. This new model accounts for both the prover and the verifier's non-local correlations. A new property of multi-prover zero-knowledge naturally emerges as a result.

1 Introduction

An *interactive proof* is a dialog between two parties: a polynomial-time *verifier* and an all-powerful *prover* [1, 2]. They agree ahead of time on some language L and a string x. The prover wishes to convince the verifier that $x \in L$. If this is true, the prover should succeed almost all the time; if not, the prover should fail almost all the time. This is a generalization of the complexity class **NP**, except instead of simply being handed a polynomial-sized witness, the verifier is allowed to quiz the prover. The set of languages that admit an interactive proof is called **IP**.

An interactive proof is *zero-knowledge* if the verifier learns nothing except the truth of "$x \in L$". This is usually defined by saying that a *distinguisher* is unable to tell apart a real conversation between the prover and the verifier, and one which is generated by a lone polynomial-time *simulator*. The set of zero-knowledge interactive proofs [1] is called **ZKIP**.

One of the most important results regarding interactive proofs is that **IP** = **ZKIP** = **PSPACE**, which follows from seminal works of [3] and [4, 5]. However, the only known way to achieve the **ZKIP** = **PSPACE** is through the use of commitments which, in the single-prover model, is dependent on complexity assumptions.

The *multi-prover* model was introduced in [6]. This model consists of multiple, non-communicating provers talking to a single verifier. The inspiration for this model was that of a detective interrogating a number of suspects, each of whom is isolated in a separate room. The suspects may share a strategy before being separated, but once the interrogation begins they are no longer able to talk to one another. The main motivation for studying this model was to remove the complexity assumptions used in the commitment schemes. We will abbreviate "multi-prover interactive proof" as MIP (resp. "zero-knowledge multi-prover interactive proof" as ZKMIP) and the set of languages which can be accepted by MIPs (resp. ZKMIPs) as the boldface **MIP** (resp. **ZKMIP**).

An important consequence of having multiple provers is that the verifier can use one prover to check the consistency of other provers' answers. This gives the (weak) verifier more power over the (all-powerful) provers. Consequently, through the works of [6–9], it was shown that **MIP** = **ZKMIP** = **NEXP**. That is, any language in **NEXP** can be accepted by a MIP (optionally by a zero-knowledge MIP) without any computational assumptions.

⋆ Supported in part by FRQNT (INTRIQ) and NSERC (CryptoWorks21 and Discovery grant program).
⋆⋆ Supported in part by Professors Václav Chvátal, Jeremy Clark, Claude Crépeau, and David Ford.

1.1 (ZK)MIP Blind Spot

We have identified a blind spot in what we call the "standard" MIP model (one verifier talking to a number of provers) that is not addressed in existing literature. As a lead-up to describing this blind spot, we invite the readers to consider the following ridiculous two-prover protocol:

Protocol 1. *(Ridiculous Protocol)*

1. Verifier sends Prover 1 a random string S.
2. Prover 1 replies with a string T.
3. Verifier sends Prover 2 the string T.
4. Prover 2 replies with a string S'.
5. Verifier accepts if $S = S'$.

Suppose that we claim the following ridiculous theorem:

Theorem 2. *(Ridiculous Theorem) The probability that the verifier accepts in the Ridiculous Protocol is exponentially small.*

Proof. (Ridiculous Proof) By the definition of MIPs, the provers cannot communicate. If Prover 2 can output an S' that is the same as the uniformly random S that only Prover 1 knows, then they must have communicated. Contradiction. □

The reader is astute in pointing out that steps 2 and 3 of the Ridiculous Protocol clearly show that the verifier is helping the provers by relaying the very answer it is supposed to keep secret. The Ridiculous Proof of the Ridiculous Theorem overlooked the blind spot that is the verifier's interactions. This is our point, exaggerated.

The blind spot in the standard MIP model is what we shall call "non-local contamination" by the verifier. For example, a verifier talking to one prover *and then* talking to another prover risks unwittingly helping the provers (up to) signal. However, the most important (and the most subtle) of those contaminations are ones where the verifier helps the provers perform a *no*-signaling correlation; examples of this can be found in the following section, and also in [10].

In existing MIP literature, the proofs of soundness do not account for this blind spot. It is easy to see the Ridiculous Verifier as clearly contaminating (in fact, steps 2 and 3 *signals* for the provers). It is not so easy when the verifier is more complex. It is an even subtler point when we consider that the verifier could be helping the provers in a no-signaling manner. We believe that proofs within the standard model must be reconsidered in light of this observation. We will further discuss this last point in section 3.

To clarify, we are not claiming that any particular existing MIP protocol is unsound, only that their proofs of soundness either missed the above point, or implicitly assumes it. We would like to make this explicit. We wish to draw the community's attention to this situation and offer our solution: a multi-prover, multi-verifier model which we shall call *locality-explicit* multi-prover interactive proofs (LE-MIP). MIPs in this form have prover-verifier pairs who are talking, but no communication *between* any of the pairs. At the end of a locality-explicit protocol, a special, read-only verifier accepts or rejects. Locality-explicit protocols do not have to worry about non-local contamination by the verifier. This new model offers the following advantages:

1. The provers and verifiers are guaranteed to be local (i.e., a very strong notion of no-communicating), if desired.
2. Any non-local resources of provers and verifiers are made explicit.
3. It is possible to enforce "honest non-locality" on the provers by having the *verifier* provide them with non-local resources. Our model makes this explicit.
4. A new property of zero-knowledge emerges naturally as a result.

1.2 Our Contributions

- We explain the aforementioned blind spot with the standard (single-verifier) MIP model (section 3).
- We describe the locality-explicit model and justify its definition by expanding on its advantages over the standard model (section 4).
- We show that, in the LE-MIP model, a new, stronger property of zero-knowledge naturally emerges.(section 4.1).
- We describe a protocol which is local-verifier, local-prover and zero-knowledge which accepts oracle-3-SAT, achieving zero-knowledge without needing the provers to authenticate any messages, and prove its security (section 5).
- We describe how to simulate the above protocol with simulators which have only a specific no-signaling advantage (section 5.2).

2 Previous Work

The early claims by Ben-Or, Goldwasser, Kilian and Wigderson that **ZKMIP** = **MIP** from [6] and [9] use multi-round protocols and their (honest) verifiers are inherently signaling. This is precisely the situation we address in this work. Proving soundness is quite subtle in this case because the provers could use the (signaling) verifier to break binding of the commitments. In particular, soundness will not be valid if the protocol is composed concurrently with other executions of itself or even used as a sub-routine. In recent conversations with Kilian [11], we have realized that controlling the impact of *signaling* via the verifier has been a concern since the early days of MIPs. In particular, extra care had to be taken in the zero-knowledge protocols described in [6] and [9] because the verifier couriered messages from one prover to the other. The protocols as they are might be sound but it is not fully proven. However, it is also clear that no considerations had been given to general non-local correlations possible via the verifier. If soundness rests on the binding property of a commitment scheme (such as those zero-knowledge proofs) and this binding property rests on the inability to achieve a certain non-local correlation then impossibility to achieve this correlation via the verifier must be demonstrated.

The reader may think that the entire issue we address may seem trivial because it is a known fact that multi-round MIPs may be reduced to a single round using techniques of Lapidot-Shamir [12] and Feige-Lovasz [13]. Nevertheless, if we are interested in *zero-knowledge* MIPs, commitment schemes are generally used to obtain the zero-knowledge property and thus the single-round structure is lost in the process. Although single-round protocols bypass verifier's non-local contamination problems we describe in this work, converting multi-round protocols into single-round ones is highly inefficient and complex. Preserving zero-knowledge while achieving single-round has turned out to be a major challenge. Practically, keeping a multi-round protocol's structure, using only commitments to achieve zero-knowledge is very appealing.

In [12], Lapidot-Shamir proposed a parallel ZKMIP for **NEXP**, but they removed the zero-knowledge claim in the journal version [14] of their work without any explanation as of why. Feige and Kilian [15] were the last ones to follow this approach combining techniques drawn from Lapidot-Shamir [12], Feige-Lovasz [13] and Dwork, Feige, Kilian, Naor, and Safra, [16] to achieve a "2-prover 1-round 0-knowledge" proof for **NEXP**. As far as we can tell, this is the only paper in the ZKMIP literature that appears to address the problems that we will discuss. However, note that the analysis of [15] is partly based of that of [12], and the journal version of Feige-Kilian [17] does not contain their prior claim of zero-knowledge either. All other ZKMIPs for **NEXP** in the literature are multi-round, and thus our work applies to them.

Similar issues are possible using more recent results such as Ito-Vidick's proof [18] that **NEXP** $\subseteq$ **MIP*** and Kalai, Raz and Rothblum's proof [19] that **MIP**^{ns} = **EXP**; the multi-round structure of their protocols requires that any straightforward extensions to **ZKMIP*** and

$\mathbf{ZKMIP}^{ns}$ via commitment schemes be analyzed carefully and the locality of the verifiers be established.

At the time of writing this paper, Chiesa, Forbes, Gur, and Spooner [20] discovered a proof that $\mathbf{NEXP} \subseteq \mathbf{ZKMIP}^*$. Their construction is based on refinements of Ito-Vidick's proof and along the lines of Feige-Kilian, building on algebraic structures to bypass the need of commitment schemes. Unfortunately, this work is too recent to be assessed as to how it is related to ours.

Bellare, Feige, and Kilian [21] considered a multi-verifier model similar to ours in order to analyze the role of randomness in multi-prover proofs. This is completely unrelated to our goal of analyzing verifier non-local contamination.

Finally, the notion of relativistic commitment schemes put forward by Kent [22] leads to several results [23–25] where a similar multi-verifier model is necessary in order to assess spatial separation of the provers.

3 The Standard MIP Model

Multi-prover interactive proofs were introduced in [6]. The intuition for their model was that of a detective interrogating two suspects held in different rooms. This was formalized as follows:

Definition 1. *Let $P_1, \ldots, P_k$ be computationally unbounded Turing machines and let V be a probabilistic polynomial-time Turing machine. All machines have a read-only input tape, a read-only auxiliary-input tape, a private work tape and a random tape. The P_i's share a joint, infinitely long, read-only random tape. Each P_i has a write-only communication tape to V, and vice-versa. We call $(P_1, \ldots, P_k, V)$ a* k-prover interactive protocol (k-prover IP).

This model is essentially equivalent to that of Bell [26] who introduced his famous Bell's inequality to distinguish *local* parties from *entangled* parties.

Zero-knowledge MIPs were also defined in [6]:

Definition 2. *Let $(P_1, \ldots, P_k, V)$ be a k-prover IP for a language L.Let $\mathbf{view}(P_1, \ldots, P_k, V, x)$ denote the verifier's incoming and outgoing messages with the provers, including his coin tosses. We say that $(P_1, \ldots, P_k, V)$ is* perfect zero-knowledge *for L if there exists an expected polynomial-time machine M such that for all V', $\mathbf{view}(P_1, \ldots, P_k, V', x)$ and $M(x)$ are identically distributed.*

Let us call the above two definitions the *standard MIP model*. There have also been augmentations of the model by giving the provers various non-local resources, such as entanglement [18], or arbitrary no-signalling power [19].

The first work to point out the aforementioned blind spot in the standard MIP model, although it was not worded explicitly, was [10]. In order to understand their point, we need to understand the following two-prover protocol.

Protocol 3. *(BGKW-type commitment for bit b)*

P_1 and P_2 pre-share a random n-bit string w.

1. V sends a random n-bit strings r to P_2.
2. P_2 replies with $x \leftarrow b \times r \oplus w$.
3. P_1 announces to V a string w'.
4. V accepts iff $(w' \oplus x) \in \{0, r\}$.

This is a two-prover commitment protocol. Steps 1 and 2 commit, while steps 3 and 4 unveil. An intuitive proof of its binding condition is that, since the provers cannot signal, and they both need to know r in order to unveil the commitment in the way they want, therefore they cannot cheat. This intuition is incomplete, as was pointed out in [10], because breaking the binding condition *does not require signaling*. The following protocol, known as a PR-box, can be used to break binding without signaling.

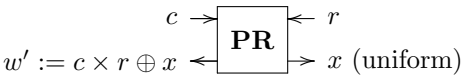

Fig. 1. a **PR**-box

By having P_1, P_2 obtain w', x via the PR-box, P_1 can unveil the commitment the way it wishes, c. This fact will become extremely important in Sections 5 and 4.1.

The punchline of [10] is that *the verifier itself can act as a PR-box for the provers without violating their no-signaling assumption*. Consider the following:

1. Any security proof of protocol 3 must show that it does not contain a PR-box as a subroutine.
2. More generally, any security proof of a protocol must show that no subroutine within itself can be commandeered by the provers to achieve a non-local functionally (like the PR-box).
3. Composition of protocols, for instance between the committing and the opening of commitments, must be done in such a way that provably does not create a non-local box.

The solution proposed in [10] was that of *verifier isolation*. Informally, this means that any message an "isolating" verifier sends to a set S of provers must be computed solely from messages that are received from S. The end result is that an isolating verifier can never accidentally implement a PR-box and, in general, it will always enforce the locality of the provers. In a sense, we can think of an isolating verifier as "local". Our new model will make this more precise and more general.

Furthermore, existing zero-knowledge MIPs such as [9] *require* that the verifier courier an authenticated message between the provers in order to obtain soundness while ensuring zero-knowledge. The gist of it goes like this:

1. V asks P_1 some questions.
2. V wants to check one of P_1's answers with P_2 for consistency.
3. In order for zero-knowledge to hold, V *must* ask P_2 a question it has already asked P_1.
4. P_1 authenticates a question with a key that was committed at the beginning of the protocol and sends it to V.
5. V sends the question and the authentication to P_2, who proceeds only if authentication succeeds.

Steps 4 and 5 consists of V sending a message from P_1 to P_2. Proofs that this act does not contaminate non-locally (such as simulating a PR-box) is not found in any existing MIP. This needs to be proven, and the proof contained in [9] does not address this issue. Moreover, the zero-knowledge protocol of [9] allows P_1 to send an arbitrary message to P_2 (via the authentication key). Therefore, one cannot compose such a protocol in a nested fashion (as a subroutine call) since the inner instance would violate the no-communication assumption of the outer instance. For more details on the problems of the standard **MIP** model, see [27].

Existing simulators for zero-knowledge protocols such as those found in [9] needs to know how to break commitments in order to simulate. The simulator accomplishes this by acting as both provers, thereby receiving the secret string r which was meant for one prover only. This standard model of zero-knowledge gives the simulator *unnecessary power*, in a sense. We will discuss this further in section 4.1.

4 Locality-Explicit MIP

The standard MIP model allows the verifier to non-locally contaminate the provers. We neutralize this problem by defining a model with multiple verifiers, each of which talks to a single prover; in turn, each prover talks to a single verifier. There are no communication tapes between the verifiers, nor are there between provers. There is a special verifier V_0 which *only reads* the outputs of the other verifiers; this is the verifier that will decide to accept or reject membership to L. We call this model "locality-explicit" since the provers and verifiers are explicitly local, and if any non-local resources (such as entanglement) are available to them, then it is explicitly specified via a supplementary entity named $\widehat{P}$ for the provers and $\widehat{V}$ for the verifiers.

This model is a *generalization* of the standard model because the special setting where $\widehat{P}$ is empty and $\widehat{V}$ signals for the verifiers corresponds to the standard MIP model.

Definition 3. *An* interactive Turning machine *(ITM) is a Turing machine augmented with the following tapes:*

- *k_1 read-only incoming communication tapes.*
- *k_2 write-only outgoing communication tapes.*
- *Private work, auxiliary-input, and random tapes.*

An ITM A can signal to an ITM B if A's write-only outgoing tape is B's read-only incoming tape.

Definition 4. *Let $(\widehat{P}, P_1, \ldots, P_k, \widehat{V}, V_0, V_1, \ldots, V_k)$ be a tuple of ITMs, where the P's are computationally all-powerful and the V's are polynomial-time. For each i, there are two-way communication tapes between V_i and P_i, and that for all j, there is a two-way communication tape between $\widehat{V}$ and V_j and also between $\widehat{P}$ and P_j. In addition, for each ℓ, there is a read-only tape going from V_ℓ to V_0 (where V_0 reads). Then, this is said to be a* locality-explicit multi-prover interactive proof.

We call $\widehat{P}$ and $\widehat{V}$ correlators *and say that the provers and verifiers are $\widehat{P}$-*local *and $\widehat{V}$-*local *respectively.*

It is perhaps easier to understand our definition with the help of figure 2.

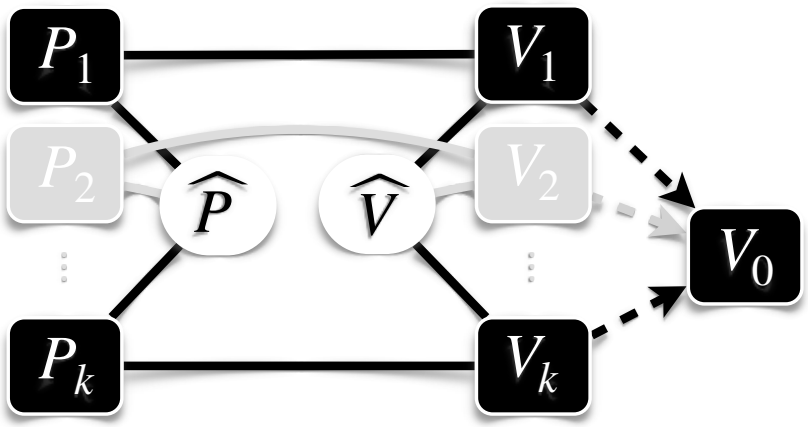

Fig. 2. Locality-Explicit MIP

The solid lines represents two-way communication and the dashed arrows represents one-way communication, with the arrow indicating the direction of information flow.

We can define that an LE-MIP accepts a language L if the usual soundness and completeness conditions hold:

Definition 5. *An LE-MIP* $(\widehat{V}, V_0, V_1, \ldots, V_k, \widehat{P}, P_1, \ldots, P_k)$ *accepts a language* L *if and only if*

- *(completeness)* $\forall x \in L, \mathbf{Pr}[V_0(x, t_1, \ldots, t_k) = \mathsf{accept}] > 2/3$,
- *(soundness)* $\forall x \notin L, \forall P'_1, \ldots, P'_k, \mathbf{Pr}[V_0(x, t_1, \ldots, t_k) = \mathsf{accept}] < 1/3$,

where t_i *is the read-only tape from* V_i *to* V_0 *at the end of the interaction of* V_i *with* P_i *(or* P'_i*) on input* x*.*

Note that we do not quantify over $\widehat{P}$ (nor $\widehat{V}$), as we want to use them not as (possibly malicious) participants to the protocol, but as a description of non-local resources available to the provers and verifiers.

Definition 6. *An LE-MIP is* local *if* $\widehat{V} = \widehat{P} = \varnothing$ *and all of the provers' (resp. verifiers') random tapes are initialized with the same uniformly random string* R *(resp. verifiers with another, independent uniformly random string* S*)*[***]*.*

Note that (single-verifier) standard MIPs in which provers do not have non-local resources are equivalent to LE-MIPs where $\widehat{P} = \varnothing$ and $\widehat{V}$ acts as a bulletin board. That is, a single verifier communicating with multiple provers is equivalent to multiple verifiers communicating with provers and each other.

In standard MIPs, it is possible that the honest (single) verifier bridges the provers non-locally. If a protocol does not desire this – and most existing MIPs do not – it must be proven. With local LE-MIPs, the special verifier V_0 decides to accept or reject. This verifier cannot communicate with anyone else, avoiding the aforementioned non-local contamination.

4.1 Zero-Knowledge LE-MIPs

Zero-knowledge is defined by simulations, the fundamental idea that if a transcript can be produced by an entity (simulator) with no more power than one (verifier) interrogating all-powerful provers, then no knowledge is gained.

The simulator of single-prover IP and standard MIP are equal to the verifier in computational power, but they do have "advantages" which allow them to fake transcripts. For single-prover IPs, the simulator is allowed to rewind computation; for standard MIPs, the simulator is given a (commitment-breaking) secret. Those advantages are, of course, independent of knowledge.

LE-MIPs naturally induces a new advantage for the simulator: non-local correlations. This is a very powerful advantage. Using the correct non-local correlations, simulators do not need to rewind, do not need to pretend to be multiple (isolated) provers, and do not need to know any commitment-breaking secrets. Multiple, no-signaling simulators can even produce transcripts in "real-time" (example will follow) if the proper correlations are used.

Definition 7. *Let* $\mathcal{M} = (\widehat{M}, M_1, \ldots, M_k)$ *be a tuple of polynomial-time ITMs. Each machine has a random tape, and every random tape is initialized with the same random bits. For* $1 \leq i \leq k$*, there is a two-way communication tape between* $\widehat{M}$ *and* M_i*. There are no communication tapes between any of the* M_i*'s. Then this is called a tuple of* locality-explicit simulators *and* $\widehat{M}$ *is the* locality class *of* $\mathcal{M}$*, which will be abbreviated* $\widehat{M}$-local*.*

[***] By $\varnothing$ we mean the empty correlator that provides everyone with nothing at all as output.

Definition 8. *Let $\mathcal{S} = (\widehat{P}, P_1, \ldots, P_k, \widehat{V}, V_0, V_1, \ldots, V_k)$ be an LE-MIP for language L. If there exists a correlator $\widehat{M}$ such that for all verifiers $(\widehat{V}', V_0', V_1', \ldots, V_k')$, there exists $(M_1, \ldots, M_k)$, such that the transcripts of conversations between*

$$(\widehat{P}, P_1, \ldots, P_k, \widehat{V}', V_0', V_1', \ldots, V_k')$$

and

$$(\widehat{M}, M_1, \ldots, M_k, \widehat{V}', V_0', V_1', \ldots, V_k')$$

are identically distributed, where $(\widehat{M}, M_1, \ldots, M_k)$ is a tuple of locality-explicit simulators, then we say that $\mathcal{S}$ is a perfectly indistinguishable, $\widehat{M}$-local zero-knowledge LE-MIP *for L.*

Our motivations for the above definitions are twofold.

First, a simulator (or simulators) should not have more power than necessary. If two *local* simulators can output for two *local* verifiers, then it is not necessary to have a single simulator (equivalent to two *signaling* simulators) do the job. Allowing simulators to signal (equivalently, having a single simulator) in the multi-prover setting is analogous to allowing unbounded running-time simulation in single-prover zero-knowledge. In general, finding the minimal $\widehat{M}$ that will allow simulation may be of some theoretic interest.

Second, the non-locality of simulators is a characterization of the resilience of zero-knowledge. A protocol which local simulators can withstand arbitrary (malicious) verifiers is more resilient than one which signaling simulators are needed.

This may be of practical interest, if transcripts are timestamped. For example, under the relativistic assumption that one may not signal faster-than-light, one may be able to distinguish two spatially separated simulators from two spatially separated verifiers, if the simulators need to signal (transmit a commitment-breaking secret) in order to generate a transcript. On the other hand, if two entangled simulators are sufficient to produce the transcript, then they are indistinguishable from real verifiers and provers. Our protocol 7 can be modified as to let entangled simulators do their work, without needing PR-boxes or signaling. Details in section 5.

4.2 The Power of LE-MIPs

Local LE-MIPs form a subclass of standard MIPs. They are, by design, more restricted in what you can make the verifier do. An immediate question is whether this is *too* restrictive. Perhaps, in all interesting cases, it is necessary for a single verifier to go back-and-fourth between provers, using previous discussions to generate new questions.

The answer is that, of all the literature we have surveyed, almost all protocols can be re-written in a local-verifier manner without any loss of functionality. We explicitly demonstrate this for the multi-prover protocol for oracle-3-SAT in [8]. The protocol details can be found in the appendix. For the purpose of our discussion, we only need to look at the general form of the protocol:

Protocol 4. *(BFL Classic, Single-Verifier)*

1. V asks P_1 some questions non-adaptively.
2. V chooses a question Q from the pool of questions which were asked to P_1.
3. V asks Q to P_2.
4. V accepts if the interaction with P_1 was successful, and the answer from P_2 is consistent with those of P_1.

The crucial observation is that V does not *adaptively* ask questions to P_1. Therefore, the questions asked on that entire side of the conversation can selected in advance, and thus they can be shared in advance with a second verifier. We can therefore naturally rewrite the BFL classic protocol as a local LE-MIP in the following way. The reader can check the details in the appendix, and in section 3 of [8].

Protocol 5. *(BFL as an LE-MIP)*

1. V_1 prepares the questions which it will ask P_1.
2. V_1 chooses a question Q from the above list and shares it with V_2.
3. LE-MIP begins. All parties are local as per definitions.
4. V_1 asks the questions to P_1.
5. V_2 asks Q to P_2.
6. V_0, reading the responses, decides to accept or reject, based on the same criteria as in protocol 4.

The BFL protocol is for oracle-3-SAT, which is **NEXP**-complete. Rewritten as a local LE-MIP, it circumvents all non-locality issues we have mentioned. Thus, we can conclusively say that "local **LE-MIP**" = **MIP** = **NEXP**; no transformation to single-round MIP necessary, and no need to invoke the general theory of PCPs.

5 A Local, Zero-Knowledge LE-MIP for NEXP

The question which follows naturally is whether there exists a *zero-knowledge*, local LE-MIP for **NEXP**. The existing technique for achieving zero-knowledge in MIP [6, 9] requires the (single) verifier to courier an authenticated message between provers. This is not possible with local-verifier LE-MIPs. We show that there is a way around that constraint.

By adapting the protocol from [8], we will exhibit a protocol with the following properties:

1. The provers and verifiers are local: $\widehat{V} = \widehat{P} = \varnothing$.
2. The simulators need only access to instances of PR-boxes to work. That is, $\widehat{M}$ simply computes indexed instances of PR-boxes. We will abbreviate this as "PR-local."

Let us call the set of multi-prover protocols with these properties "PR-local **ZK**, local **LE-MIP**". This implies that "PR-local **ZK**, local **LE-MIP**" = **ZKMIP** = **NEXP**.

The generic way of turning an interactive proof into a zero-knowledge one is by running it in committed form [6, 9]. With this technique, provers commit their answers instead of directly responding, and use cryptographic techniques to convince the verifier that the answers are correct.

As shown in section 4.2, the BFL protocol can be turned into a local LE-MIP. If we try to turn it into a zero-knowledge LE-MIP by having the provers commit their answers (for example using protocol 3 as commitment), we run into a problem. In order to achieve zero-knowledge, the provers *must* ensure that the question P_2 receives from V_2 is one of the questions which V_1 has asked P_1. On the other hand, since the provers and verifiers are local, the provers cannot communicate, nor can they ask the verifiers to courier authenticated messages between them.

Our solution essentially asks the provers to (strongly-universal-2) hash the selected committed answer with a key that is based on the verifier's question. We force V_2 to behave honestly (to ask a question that V_1 has asked) by making bad questions meaningless. If the verifiers ask the provers the same question, they will receive the same hash of the same answer. Otherwise, they will receive two unrelated random hash values.

We need the PR commitment (protocol 6), which is secure in the local setting as previously proved in [22, 10, 23].

5.1 The Protocols

The following is a PR-type commitment that is perfectly concealing and statistically binding. In general, we use the commitment-box notation "$\boxed{b}$" as the name of a commitment to bit b in the next two protocols.

Protocol 6. *A statistically binding, perfectly concealing commitment protocol to bit b.*

All parties agree on a security parameter 1^k.
P_1 and P_2 partition their private random tape into two k-bit strings w_1, w_2.

Pre-computation phase:
- V_1 samples two k-bit strings z_1, z_2 independently and uniformly, and provides them to V_2.
- V_1 sends z_1 to P_1 and V_2 sends z_2 to P_2.

Commit phase:
- P_1 commits b to V_1 as $\boxed{b} = (b \times z_1) \oplus w_1$, where $b \times z_1$ is a multiplication in $\mathbb{F}_{2^n}$.
- P_2 sends V_2: $d = (w_1 \times z_2) \oplus w_2$.

Unveiling phase:
- P_1 sends w_1, w_2 to V_1.
- V_1 computes $b = 1$ if $\boxed{b} \oplus w_1 = z_1$, or $b = 0$ if $\boxed{b} = w_1$.
- V_0 **rejects** if $\boxed{b} \oplus w_1$ is anything but z_1 or 0, or if $d \oplus w_2 \neq w_1 \times z_2$ and **accepts** b otherwise.

Below is the zero-knowledge, local LE-MIP for oracle-3-SAT (Protocol 7). The basis of protocol 7 is the localized BFL protocol we presented in section 4.2 (details in the appendix). A note on notation: for a circuit f, we will denote $f\left(\boxed{x}\right)$ as the gate-by-gate committed circuit evaluated with x as the input. We also use statements such as "P_1 proves to V_1 that $\boxed{\Omega_1}$ was computed correctly". The reader is expected familiarity with zero-knowledge computations on committed circuits as put forward by [28, 29, 4, 9].

Protocol 7. *A local zero-knowledge LE-MIP for oracle-3-SAT*

Let $x = (B, r, s)$, an instance of oracle-3-SAT, be the common input, let $k = |x| = r + 3s + 3$, and let Λ be the verifier's program in protocol 11 (see appendix).

1. **Pre-computation:**
   (a) V_1 samples two k-bit strings z_1, z_2 independently and uniformly, and provides them to V_2.
   (b) V_1 selects $k + 3$ random bit strings $R_1, ..., R_{k+3}$ (size specified implicitly by Λ) and evaluates the circuit of Λ using the R_i as randomness, resulting in questions $Q_1, ..., Q_{k+3}$, and provides them to V_2
   (c) V_1 randomly chooses i, $1 \leq i \leq k + 3$, the index of an oracle query that will be made to both P_1 and P_2. V_1 provides i to V_2.
   (d) V_1 sends z_1 to P_1 and V_2 sends z_2 to P_2 for future commitments.
   (e) All parties agree on a family of strongly-universal-2 hash functions $\{H_i\}$ indexed by k-bit keys.
   (f) P_1 and P_2 agree on a k-bit key γ, an index to the above family.

(g) P_1 commits $\boxed{\gamma}$ to V_1.

2. **Sumcheck with oracle:**
   - Let f be the arithmetization obtained in protocol 10, let z be a string from I^r and $Q_{k+1}, Q_{k+2}, Q_{k+3}$ be strings of I^s as generated in protocol 11. V_1 and P_1 execute protocol 10 in committed form. At the end of this phase, P_1 shows that the committed final value is equal to

$$f\Big(z, Q_{k+1}, Q_{k+2}, Q_{k+3}, \boxed{A(Q_{k+1})}, \boxed{A(Q_{k+2})}, \boxed{A(Q_{k+3})}\Big),$$

     an evaluation in committed form of f using the committed values that were used during the protocol's loop. If this fails, V_1 instructs V_0 to reject.

3. **Multilinearity test:**
   (a) For $1 \leq i \leq k$:
      i. V_1 sends Q_i to P_1,
      ii. P_1 commits his answer as $\boxed{A(Q_i)}$.
   (b) P_1 and V_1 evaluate a circuit description of Λ in committed form with inputs $\boxed{A(Q_1)}, \ldots, \boxed{A(Q_k)}$ to verify proper linearity among them. P_1 unveils the circuit's committed output. If it rejects, V_1 instructs V_0 to reject.

4. **Consistency test:**
   (a) V_1 sends i to P_1.
   (b) P_1 computes $\boxed{\Omega_1} = \boxed{A(Q_i)} \oplus H_{\boxed{\gamma}}(Q_i)$ and sends $\boxed{\Omega_1}$ to V_1.
   (c) P_1 proves to V_1 that $\boxed{\Omega_1}$ was computed correctly, from the existing commitments.
   (d) P_1 unveils $\boxed{\Omega_1}$ for V_1, who gets Ω_1.
   (e) V_2 sends Q_i to P_2 (recall that this was pre-agreed in step 1.(c))
   (f) P_2 responds to V_2 with $\Omega_2 = A(Q_i) \oplus H_\gamma(Q_i)$.
   (g) V_0 accepts if and only if all of the following conditions are met:
      - $\Omega_1 = \Omega_2$
      - All commitments which have been unveiled are valid.
      - V_1 did not reject in the two previous cases

5.2 Proofs of Security

Locality

Since the protocol is written as an LE-MIP in which $\widehat{P} = \widehat{V} = \varnothing$, the protocol is local by definition 6.

Completeness

Completeness follows from the completeness of the underlying protocol [8], and the fact that the commitment protocol (protocol 6) is well-defined for honest provers (who will never send a commitment that they cannot unveil).

Soundness

Without loss of generality, we may assume that the soundness error in the BFL protocol to be 1/3, through sequential amplification. The probability that our commitment scheme (protocol 6) fails binding is exponentially small in k. Local probabilistic provers are equivalent to local deterministic provers. This is because the success probability α of randomized provers of breaking soundness is an average over the randomized provers' random tapes. Each instance of a random tape represents a deterministic strategy. Therefore there is a deterministic strategy which succeeds with probability at least α, and hence we only need to consider local deterministic provers.

Since P_1 is deterministic, we may unambiguously consider what happens if we were to "rewind" the prover machine. Suppose that at some point P_1 unveils a particular commitment c to 0. We rewind P_1 and let V_1 make different choices before that point. Suppose that, with these alternate choices, P_1 then unveils c to 1 (an attempt to break binding). Because of locality, P_1's behavior is independent of what P_2 receives (namely z_2). Therefore, there is only *one* such z_2 which V_0 will ultimately accept as a valid unveiling of c in both ways (recall that our commitment is statistically binding).

Therefore, in the worst case, for every commitment there exists a sequence of interactions between V_1 and P_1 such that P_1 will attempt to break the binding of that commitment. Each such commitment-breaking corresponds to at most one string z_2 that will actually work.

Let us denote the set of such binding-breaking strings by B. If $z_2 \notin B$, then the provers *will not break binding*, and the soundness error is reduced to that of the underlying protocol (at most 1/3). On the other hand, since $|B| < \mathbf{poly}(k)$, the probability that $z_2 \in B$ is at most $\mathbf{poly}(k)/2^k$.

Therefore, the soundness error of our protocol is at most

$$Pr[z_2 \notin B \text{ and underlying protocol accepts}] + Pr[z_2 \in B] \leq \frac{1}{3} + \frac{\mathbf{poly}(k)}{2^k}.$$

Zero-Knowledge The simulation will be divided in two parts. In the first part, the simulator produces a transcript of the *pre-computation*, *multilinearity test* and *sumcheck with oracle* parts, which involves only interactions with V_1. In the second part, the simulator will fake a valid *consistency test*.

Protocol 8. *(Perfectly Indistinguishable, PR-Local Simulator for Protocol 7, Part 1)*

The setup:
- Let $(\widehat{M}, M_1, M_2)$ be a set of locality-explicit simulators.
- M_1 and M_2 can send $\widehat{M}$ an index along with a bit.
- $\widehat{M}$ completes the indexed PR box (protocol 3) for both simulators.

The simulation strategy:
1. The simulators agree on unique indices for every commitment used in the protocol.
2. M_1 interacts with V_1 the way P_1 would. Whenever P_1 should commit, M_1 commits to random bits, just like the single-simulator from section 5.
3. For each commitment, V_2 sends M_2 a string s. M_2 sends to $\widehat{M}$ the index of the commitment and s.
4. $\widehat{M}$ runs the PR box (protocol 3) and replies with V_2's half of the output.
5. Whenever M_1 needs to unveil a commitment, it can be unveiled in the way M_1 desires by sending the corresponding index and bit to $\widehat{M}$.
6. $\widehat{M}$ completes the corresponding PR box which outputs t. $\widehat{M}$ sends t to M_1.
7. M_1 sends t to V_1.

The second part (the consistency test) can be done by having the simulators ignore the question.

Protocol 9. *(Perfectly Indistinguishable, PR-Local Simulator for Protocol 7, Part 2)*

1. V_1 sends i to M_1.
2. M_1 computes $\boxed{\Omega_1} = H_{\boxed{\gamma}}(Q_i)$.
3. Using $\widehat{M}$ to break binding, M_1 convinces V_1 that $\boxed{\Omega_1}$ is actually $\boxed{A(Q_i)} \oplus H_{\boxed{\gamma}}(Q_i)$.
4. M_1 unveils $\boxed{\Omega_1}$ for V_1, who gets $\Omega_1 = H_\gamma(Q_i)$.
5. V_2 sends Q'_i to M_2.
6. M_2 responds with $\Omega_2 = H_\gamma(Q'_i)$.

By the properties of the strongly-universal-2 hash H, if $Q_i = Q'_i$ then $\Omega_1 = \Omega_2$. Otherwise $\Omega_1 \neq \Omega_2$ with probability exponentially close to one. This produces the result as desired. The simulators then feed the transcripts to V_0, and terminates simulation.

5.3 Entangled Simulators

The binding condition of commitment used above (protocol 6) can be broken given PR-boxes. However, if the verifier were willing to tolerate approximately 15% of errors in the provers' unveiling string (z_1 or 0), then it is possible to break binding with shared entanglement [30] while maintaining soundness against local provers. Using this weakened version of commitment in place of protocol 6 still yields a local LE-MIP for oracle-3-SAT, but easier to simulate (using weaker non-local resources). We leave the details of this modified protocol to the reader.

6 Conclusions and Future Work

We close with three open questions.

First, although protocol 7 is a *local* LE-MIP, the only known ways of simulating the transcript are to give the simulators some kind of non-local resource such as a PR box (or a fully signaling box, but that is not necessary). We do not know whether it is possible to simulate protocol 7 with *local* simulators, but we are unable to show this to be impossible.

Second, as of the time of this writing, it is an open question whether **NEXP** $\subsetneq$ **MIP*** [18]. Under the locality-explicit setup, we ask a slightly more general question: does there exist a correlator $\widehat{P}$ and a corresponding LE-MIP which accepts a language $\notin$ **NEXP**? We remind the reader that characterizing the complexity classes of MIPs where the provers have non-local resources are generally open questions.

Third, although the verifier's non-local contamination is undesirable (in the standard MIP model) and is the motivation for this work, is it possible to turn it into a resource? For example, given local provers, let the verifier provide them with some non-local resources, such PR boxes or entanglement that can be simulated in polynomial-time. This can be seen as "enforceable honest non-local resources." Malicious provers would not be able to use these resources at will. Perhaps this concept would be useful in the design of multi-prover protocols.

Acknowledgements

We would like to thank G. Brassard, A. Chailloux, S. Fehr, J. Kilian, S. Laplante, J. Li, A. Leverrier, A. Massenet, S. Ranellucci, L. Salvail, C. Schaffner, and T. Vidick for various discussions about earlier versions of this work. We would also like to thank Jeremy Clark for his insightful comments. Finally, we are grateful to Raphael Phan and Moti Yung for inviting us to publish a lead-up paper to this work as an *Insight Paper* at MyCrypt 2016.

References

1. S. Goldwasser, S. Micali, and C. Rackoff, "The knowledge complexity of interactive proof-systems," *SIAM. J. Computing*, vol. 18, pp. 186–208, Feb. 1989.
2. L. Babai, "Trading group theory for randomness," in *Proceedings of the Seventeenth Annual ACM Symposium on Theory of Computing*, pp. 421–429, May 1985.
3. A. Shamir, "IP = PSPACE," *J. ACM*, vol. 39, pp. 869–877, Oct. 1992.
4. R. Impagliazzo and M. Yung, "Direct minimum-knowledge computations," in *Advances in Cryptology: Proceedings of Crypto '87* (C. Pomerance, ed.), vol. 293, pp. 40–51, Springer-Verlag, 1988.
5. M. Ben-Or, O. Goldreich, S. Goldwasser, J. Håstad, J. Kilian, S. Micali, and P. Rogaway, "Everything provable is provable in zero-knowledge," in *Proceedings of the 8th Annual International Cryptology Conference on Advances in Cryptology*, CRYPTO '88, (London, UK, UK), pp. 37–56, Springer-Verlag, 1990.
6. M. Ben-Or, S. Goldwasser, J. Kilian, and A. Wigderson, "Multi-prover interactive proofs: How to remove intractability assumptions," in *Proceedings of the Twentieth Annual ACM Symposium on Theory of Computing*, STOC '88, (New York, NY, USA), pp. 113–131, ACM, 1988.
7. L. Fortnow, J. Rompel, and M. Sipser, "On the power of multi-prover interactive protocols," *Theor. Comput. Sci.*, vol. 134, pp. 545–557, Nov. 1994.
8. L. Babai, L. Fortnow, and C. Lund, "Non-deterministic exponential time has two-prover interactive protocols," *Comput. Complex.*, vol. 2, pp. 374–374, Dec. 1992.
9. J. Kilian, *Uses of randomness in algorithms and protocols*. MIT Press, 1990.
10. C. Crépeau, L. Salvail, J.-R. Simard, and A. Tapp, "Two provers in isolation," in *Advances in Cryptology – ASIACRYPT 2011: 17th International Conference on the Theory and Application of Cryptology and Information Security, Seoul, South Korea, December 4-8, 2011. Proceedings*, (Berlin, Heidelberg), pp. 407–430, Springer Berlin Heidelberg, 2011.
11. J. Kilian, "Personal e-mail communication," July 2018.
12. D. Lapidot and A. Shamir, "Fully parallelized multi prover protocols for nexp-time (extended abstract)," in *32nd Annual Symposium on Foundations of Computer Science, San Juan, Puerto Rico, 1-4 October 1991*, pp. 13–18, IEEE Computer Society, 1991.
13. U. Feige and L. Lovász, "Two-prover one-round proof systems: Their power and their problems (extended abstract)," in *Proceedings of the Twenty-fourth Annual ACM Symposium on Theory of Computing*, STOC '92, (New York, NY, USA), pp. 733–744, ACM, 1992.
14. D. Lapidot and A. Shamir, "Fully parallelized multi-prover protocols for nexp-time," *J. Comput. Syst. Sci.*, vol. 54, no. 2, pp. 215–220, 1997.
15. U. Feige and J. Kilian, "Two prover protocols: low error at affordable rates," in *Proceedings of the Twenty-Sixth Annual ACM Symposium on Theory of Computing, 23-25 May 1994, Montréal, Québec, Canada* (F. T. Leighton and M. T. Goodrich, eds.), pp. 172–183, ACM, 1994.
16. C. Dwork, U. Feige, J. Kilian, M. Naor, and S. Safra, "Low communication 2-prover zero-knowledge proofs for NP," in *Advances in Cryptology - CRYPTO '92, 12th Annual International Cryptology Conference, Santa Barbara, California, USA, August 16-20, 1992, Proceedings* (E. F. Brickell, ed.), vol. 740 of *Lecture Notes in Computer Science*, pp. 215–227, Springer, 1992.
17. U. Feige and J. Kilian, "Two-prover protocols - low error at affordable rates," *SIAM J. Comput.*, vol. 30, no. 1, pp. 324–346, 2000.
18. T. Ito and T. Vidick, "A multi-prover interactive proof for nexp sound against entangled provers," in *Proceedings of the 2012 IEEE 53rd Annual Symposium on Foundations of Computer Science*, FOCS '12, (Washington, DC, USA), pp. 243–252, IEEE Computer Society, 2012.

19. Y. T. Kalai, R. Raz, and R. D. Rothblum, "How to delegate computations: The power of no-signaling proofs," in *Proceedings of the Forty-sixth Annual ACM Symposium on Theory of Computing*, STOC '14, (New York, NY, USA), pp. 485–494, ACM, 2014.
20. A. Chiesa, M. A. Forbes, T. Gur, and N. Spooner, "Spatial isolation implies zero knowledge even in a quantum world," *Electronic Colloquium on Computational Complexity (ECCC)*, vol. 25, p. 44, 2018.
21. M. Bellare, U. Feige, and J. Kilian, "On the role of shared randomness in two prover proof systems," in *Third Israel Symposium on Theory of Computing and Systems, ISTCS 1995, Tel Aviv, Israel, January 4-6, 1995, Proceedings*, pp. 199–208, IEEE Computer Society, 1995.
22. A. Kent, "Unconditionally secure bit commitment," *Phys. Rev. Lett.*, vol. 83, pp. 1447–1450, Aug 1999.
23. T. Lunghi, J. Kaniewski, F. Bussières, R. Houlmann, M. Tomamichel, S. Wehner, and H. Zbinden, "Practical relativistic bit commitment," *Phys. Rev. Lett.*, vol. 115, p. 030502, Jul 2015.
24. E. Adlam and A. Kent, "Deterministic relativistic quantum bit commitment," *CoRR*, vol. abs/1504.00943, 2015.
25. A. Chailloux and A. Leverrier, "Relativistic (or 2-prover 1-round) zero-knowledge protocol for NP secure against quantum adversaries," in *Advances in Cryptology – EUROCRYPT 2017: 36th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Paris, France, April 30 – May 4, 2017, Proceedings, Part III*, pp. 369–396, Springer International Publishing, 2017.
26. J. S. Bell, "On the Einstein-Podolsky-Rosen paradox," *Physics*, vol. 1, pp. 195–200, 1964.
27. C. Crépeau and N. Yang, "Multi-prover interactive proofs: Unsound foundations," in *Paradigms in Cryptology – Mycrypt 2016. Malicious and Exploratory Cryptology: Second International Conference, Mycrypt 2016, Kuala Lumpur, Malaysia, December 1-2, 2016, Revised Selected Papers*, pp. 485–493, Springer International Publishing, 2017.
28. G. Brassard and C. Crépeau, "Zero-knowledge simulation of boolean circuits (extended abstract)," in *Advances in Cryptology: Proceedings of Crypto '86* (A. M. Odlyzko, ed.), vol. 263, pp. 223–233, Springer-Verlag, 1987.
29. G. Brassard and C. Crépeau, "Non-transitive transfer of confidence: A perfect zero-knowledge interactive protocol for SAT and beyond," in 27^{th} *Symp. of Found. of Computer Sci.*, pp. 188–195, IEEE, 1986.
30. G. Brassard, A. Broadbent, and A. Tapp, "Multi-party pseudo-telepathy," in *Algorithms and Data Structures* (F. Dehne, J.-R. Sack, and M. Smid, eds.), (Berlin, Heidelberg), pp. 1–11, Springer Berlin Heidelberg, 2003.

Appendix: Babai, Fortnow and Lund's MIP for Languages in NEXP

This section describes a variant of the multi-prover protocol for oracle-3-SAT found in [8]. We refer to this as the BFL protocol, or BFL classic.

Definition 9. *Let $r, s > 0$ be integers. Let z, b_1, b_2, b_3 be strings of variables, where $|z| = r$ and $|b_i| = s$. Let $B(z, b_1, b_2, b_3, t_1, t_2, t_3)$ be a Boolean formula in $r + 3s + 3$ variables. A Boolean function $A : \{0,1\}^s \to \{0,1\}$ is a 3-satisfying oracle for B if*

$$B(z, b_1, b_2, b_3, A(b_1), A(b_2), A(b_3)) = 1$$

for every string z, b_1, b_2, b_3.

B is oracle-3-satisfiable if such a function A exists.

The Oracle-3-SAT problem (B, r, s) asks whether a Boolean formula B is oracle-3-satisfiable, where r and s denote the lengths of z and b_i, as above.

Lemma 1. *Oracle-3-SAT is* **NEXP***-complete.*

Definition 10. *Let $\mathbb{F}$ be an arbitrary field. Let $\phi : \{0,1\}^m \to \{0,1\}$ be a Boolean function. An* arithmetization *of ϕ is a polynomial $f(x_1, \ldots, x_m) \in \mathbb{F}[X_1, \ldots, X_m]$ such that for all $z \in \{0,1\}^m$, $\phi(z) = 0 \Leftrightarrow f(z) = 0$. A specific one is given in [8], proposition 3.1 .*

Equivalently, the $\phi(z) = 0 \Leftrightarrow f(z) = 0$ condition can be replaced with $\phi(z) = 1 \Leftrightarrow f(z) = 0$.

Protocol 10. *(Sumcheck Protocol)*

Let $\phi(x_1, \ldots, x_m)$ be the 3-CNF formula which the prover P is trying to show to be a tautology to a verifier V. Let $\mathbb{F}$ be a field of sufficient size (of order at least $(3c + 1)m$ will suffice where c is the number of clauses of ϕ).

1. V takes ϕ and computes its arithmetization f according to [8] Proposition 3.1 and sends it to P.
2. V and P agree on a set $I \subset \mathbb{F}$ of size at least $2dm$ where d is the degree of f.
3. V assigns $b_0 = 0$, which is supposed to be equal to the sum

$$\sum_{x_1=0}^{1} \cdots \sum_{x_m=0}^{1} f(x_1, \ldots, x_m)^2 = 0$$

4. $i \leftarrow 1$.
5. P sends the coefficients of the univariate polynomial in x,

$$g_i(x) = h(r_1, \ldots, r_{i-1}, x) = \sum_{x_{i+1}=0}^{1} \cdots \sum_{x_m=0}^{1} f(r_1, \ldots, r_{i-1}, x, x_{i+1}, \ldots, x_m)^2$$

6. V checks whether $b_{i-1} = g_i(0) + g_i(1)$. If not, abort.
7. V chooses a random $r_i \in I$, computes $b_i = g_i(r_i)$ and sends r_i to P.
8. If $i \leq m$ then $i \leftarrow i + 1$ and go to step 4.
9. V checks whether $b_m = f(r_1, \ldots, r_m)^2$.

Protocol 11. *(Babai, Fortnow and Lund's MIP for Oracle-3-SAT)*

Given (B, r, s) as common input.

1. (sumcheck with oracle) V and P_1 execute protocol 10. Let $(Q_{k+1}, Q_{k+2}, Q_{k+3}) = (r_{r+1}...r_{r+s}, r_{r+s+1}...r_{r+2s}, r_{r+2s+1}...r_{r+3s}) \in (I^s)^3$ be V's questions during this phase.
2. (multilinearity test) V asks P_1 to simulate an oracle storing the function A. V queries P_1 with random, linearly related values in I^s. If any response does not satisfy linearity, abort protocol. Let $Q_1, \ldots, Q_k \in I^s$ be V's questions during this phase.
3. (non-adaptiveness test) V chooses uniformly at random an i such that $1 \leq i \leq k+3$ and asks Q_i to P_2. If P_2's answer differs from that of P_1, reject. Otherwise accept.
