Input-Oblivious Proof Systems
and a Uniform Complexity Perspective on P/poly

Oded Goldreich[*] and Or Meir[†]
Department of Computer Science
Weizmann Institute of Science
Rehovot, ISRAEL.

February 16, 2011

Abstract

We initiate a study of input-oblivious proof systems, and present a few preliminary results regarding such systems. Our results offer a perspective on the intersection of the non-uniform complexity class $\mathcal{P}$/poly with uniform complexity classes such as $\mathcal{NP}$ and $\mathcal{IP}$. In particular, we provide a uniform complexity formulation of the conjecture $\mathcal{NP} \not\subset \mathcal{P}$/poly and a uniform complexity characterization of the class $\mathcal{IP} \cap \mathcal{P}$/poly. These (and similar) results offer a perspective on the attempt to prove circuit lower bounds for complexity classes such as $\mathcal{NP}$, $\mathcal{PSPACE}$, $\mathcal{EXP}$, and $\mathcal{NEXP}$.

Keywords: NP, IP, PCP, ZK, P/poly, MA, BPP, RP, E, NE, EXP, NEXP.

Contents

[*] Partially supported by the Israel Science Foundation (grant No. 1041/08).

[†] Research supported by the Adams Fellowship Program of the Israel Academy of Sciences and Humanities.

1 Introduction

Various types of proof systems play a central role in the theory of computation. In addition to NP-proof systems, which provide the definitional pillar of $\mathcal{NP}$, probabilistic proof systems giving rise to classes such as $\mathcal{IP}, \mathcal{ZK}$ and $\mathcal{PCP}$ have also played a major role. (For further background, see [6, Chap. 9].)

In all these cases, the verification procedure is personificated by a player, called the verifier, which interacts (implicitly or explicitly) with a more powerful entity, called the prover. The nature of this interaction may vary according to the type of proof system being considered, but in all cases the interaction may depend on a common input, which represents the claim being proved and verified. In particular, in all cases, the actions of the prescribed prover may depend on this common input.

In this work we ask how is the expressive power of these proof system effected when the prescribed prover is only given the length of the claim to be proved. We stress that we restrict the power of the prover being referred to in the completeness condition, but maintain the original formulation of the soundness condition. That is, we ask what is the power of input-oblivious provers in each of these proof systems.

1.1 The case of NP

Consider for example the case of $\mathcal{NP}$. Recall that $S \in \mathcal{NP}$ if there exists a polynomial-time (verification) procedure V and a polynomial p such that

Completeness: For every $x \in S$ there exists $w \in \{0,1\}^{p(|x|)}$ such that $V(x,w) = 1$.

Soundness: For every $x \notin S$ and every w, it holds that $V(x,w) = 0$.

We ask whether (for this procedure V or for an alternative one) it holds that for every $n \in \mathsf{N}$ there exists $w \in \{0,1\}^{p(n)}$ such that for every $x \in S_n \stackrel{\text{def}}{=} S \cap \{0,1\}^n$ it holds that $V(x,w) = 1$. Such a string w may be considered a universal NP-witness (for all $x \in S_n$), and its existence yields a poly(n)-sized circuit for deciding S_n (i.e., $S \in \mathcal{P}/\text{poly}$). But does every set in $\mathcal{NP} \cap \mathcal{P}/\text{poly}$ have such universal NP-witnesses? Denoting the class of sets having input-oblivious NP-proofs by $\mathcal{ONP}$, we show that

Theorem 1.1 (on the power of input-oblivious NP-proofs):

1. *$\mathcal{ONP} = \mathcal{NP}$ if and only if $\mathcal{NP} \subset \mathcal{P}/\text{poly}$.*
2. *If $\mathcal{NE} \neq \mathcal{E}$, then $\mathcal{ONP} \neq \mathcal{P}$.*
3. *$\mathcal{RP} \subseteq \mathcal{ONP} \subseteq \mathcal{NP} \cap \mathcal{P}/\text{poly}$.*

While the proofs of all items of Theorem 1.1 are quite easy, we find the foregoing assertions quite interesting. In particular, we highlight the fact that the first item provides a uniform complexity formulation of the conjecture $\mathcal{NP} \not\subset \mathcal{P}/\text{poly}$. We mention that it is not clear whether or not $\mathcal{ONP} = \mathcal{NP} \cap \mathcal{P}/\text{poly}$; ditto whether or not $\mathcal{BPP} \cap \mathcal{NP} \subseteq \mathcal{ONP}$ (or whether "$\mathcal{BPP} \subseteq \mathcal{NP}$ implies $\mathcal{BPP} \subseteq \mathcal{ONP}$").

We also define and study input-oblivious versions of interactive proof systems (i.e., $\mathcal{IP}$), zero-knowledge proof systems (i.e., $\mathcal{ZK}$), and probabilistically checkable proof systems (i.e., $\mathcal{PCP}$).

Related work. Chakaravarthy and Roy [4] considered an input-oblivious version of the symmetric alternation class $\mathcal{S}_2$, and showed that this new class, denoted $\mathcal{O}_2$, contains $\mathcal{BPP}$. They also showed that if $\mathcal{NP} \subset \mathcal{P}/\text{poly}$, then the Polynomial-time Hierarchy collapses to $\mathcal{O}_2$. We note that it is not clear how $\mathcal{O}_2$ relates to $\mathcal{NP}$, but it is syntactically obvious that $\mathcal{O}_2$ contains the class $\mathcal{ONP}$, defined by us.

1.2 Connection to circuit lower bounds

An additional motivation for the study of input-oblivious proof systems comes from their connection to circuit complexity. As we explain below, input-oblivious proof systems may be viewed as a restriction of $\mathcal{P}/\text{poly}$ to advice strings that can be verified. As such, it turns out that, while input-oblivious proof systems are strictly weaker than $\mathcal{P}/\text{poly}$, there are cases in which the the computational limitations of input-oblivious proof systems imply corresponding limitations on $\mathcal{P}/\text{poly}$. Thus, proving that certain classes do not have small circuits is equivalent to proving that these classes have no input-oblivious proof systems. Details follow.

Recall that $\mathcal{P}/\text{poly}$ may be viewed as the class of sets that can be decided by a Turing machine that takes advice. The advice is an arbitrary string of polynomial length, which may depend on the length of the input but not on the input itself. Consider the function $f : \mathsf{N} \to \{0,1\}^*$ that maps each input length to its corresponding advice string. The definition of $\mathcal{P}/\text{poly}$ places no restrictions on the complexity of computing f, and in particular f is not even required to be computable. This feature of the advice makes $\mathcal{P}/\text{poly}$ a powerful class, which can even compute functions that are not computable by Turing machines.

It is a natural question to ask what happens when we place computational restrictions on f. The first restriction that may come to mind is to require that $f(n)$ is computable in time $\text{poly}(n)$. However, restricting $\mathcal{P}/\text{poly}$ in this way results in the class $\mathcal{P}$, and is therefore not very interesting.

A second natural restriction is requiring the function f to be verifiable. In other words, we require that although we may not be able to compute the advice efficiently, we can at least verify its correctness. This idea can be realized in few possible ways, and our notions of input-oblivious proof systems can be thought as such realizations

Our notions of input-oblivious proof systems (e.g., $\mathcal{ONP}$) may be useful towards studying the circuit complexity of the their standard counterparts (resp., $\mathcal{NP}$), because on the one hand these input-oblivious proof systems are strictly weaker than $\mathcal{P}/\text{poly}$, and on the other hand they retains much of the power of $\mathcal{P}/\text{poly}$. As an example, consider the class $\mathcal{ONP}$, On the one hand, $\mathcal{ONP}$ is contained in $\mathcal{NP}$, and is therefore strictly weaker than $\mathcal{P}/\text{poly}$ (since it can not decide uncomputable functions). On the other hand, Theorem 1.1 shows that *if $\mathcal{P}/\text{poly}$ contains $\mathcal{NP}$, then so does $\mathcal{ONP}$*, and this is sense $\mathcal{ONP}$ is quite powerful. A particular interesting corollary of this theorem is that proving circuit lower bounds for $\mathcal{NP}$ is equivalent to separating $\mathcal{ONP}$ from $\mathcal{NP}$.

The foregoing discussion is not resticted to $\mathcal{ONP}$. In Section 3 we consider the class $\mathcal{OIP}$, which is the input-oblivious version of $\mathcal{IP}$. The class $\mathcal{OIP}$ may also be thought of as the class that results from restricting the advice of $\mathcal{P}/\text{poly}$ (i.e., the above function f) to be verifiable by an interactive protocol. We show that

$$\mathcal{OIP} = \mathcal{IP} \cap \mathcal{P}/\text{poly}\,, \text{ which equals } \mathcal{PSPACE} \cap \mathcal{P}/\text{poly}.$$

This equality gives a characterization of $\mathcal{OIP}$ as a powerful restriction of $\mathcal{P}/\text{poly}$. It also implies that proving circuit lower bounds for $\mathcal{PSPACE}$ is equivalent to separating $\mathcal{OIP}$ from $\mathcal{IP}$.

An additional example is the class $\mathcal{OMA}$, the input-oblivious version of $\mathcal{MA}$ (see Section 3). The class $\mathcal{OMA}$ may also be thought of as the class that results by restricting the advice of $\mathcal{P}/\text{poly}$

to be verifiable by in probabilistic polynomial-time (rather than in determinstic polynomial-time). Babai *et al.* [3] showed that if $\mathcal{EXP} \subseteq \mathcal{P}$/poly then $\mathcal{EXP} = \mathcal{MA}$, and their proof implicitly yields the stronger conclusion $\mathcal{EXP} = \mathcal{OMA}$. The latter result may be viewed as saying that $\mathcal{OMA}$, while being a restriction of $\mathcal{P}$/poly, is still sufficiently powerful to contain $\mathcal{EXP}$ if $\mathcal{P}$/poly contains $\mathcal{EXP}$. This implies that in order to prove circuit lower bounds for $\mathcal{EXP}$, it suffices to separate $\mathcal{EXP}$ from $\mathcal{OMA}$.

Similarly, Impagliazzo *et al.* [8] showed that $\mathcal{NEXP} \subseteq \mathcal{P}$/poly implies $\mathcal{NEXP} = \mathcal{MA}$, and implicitly that $\mathcal{NEXP} \subseteq \mathcal{P}$/poly implies $\mathcal{NEXP} = \mathcal{OMA}$. This result too may be interpreted as saying that in order to prove circuit lower bounds for $\mathcal{NEXP}$, it suffices to separate $\mathcal{NEXP}$ from $\mathcal{OMA}$.

We conclude that input-oblivious proof systems such as $\mathcal{ONP}$, $\mathcal{OMA}$, and $\mathcal{OIP}$ can be viewed as powerful restrictions of $\mathcal{P}$/poly, and therefore may serve as a useful target for research on lower bounds.

1.3 Organization and a piece of notation

In Section 2 we study input-oblivious NP-proof systems (ONP). The study of general input-oblivious interactive proof systems (i.e., OIP) and the special case of input-oblivious MA are presented in Section 3. Other forms of input-oblivious probabilistic proof systems are investigated in Section 4.

Recurring notation. For an arbitrary set $S \subseteq \{0,1\}^*$ and $n \in \mathbb{N}$, we denote by S_n the set $S \cap \{0,1\}^n$.

2 Input-Oblivious NP-Proof Systems (ONP)

In continuation to the discussion in the introduction, we define the input-oblivious version of NP-proof systems as follows:

Definition 2.1 (input-oblivious NP-proofs – $\mathcal{ONP}$): *A set S has an* input-oblivious NP-proof system *if there exists a polynomial-time algorithm V and a polynomial p such that the following two conditions hold.*

Completeness: *For every $n \in \mathbb{N}$, there exists $w \in \{0,1\}^{p(n)}$ such that for every $x \in S_n \stackrel{\rm def}{=} S \cap \{0,1\}^n$ it holds that $V(x,w) = 1$. We call w a* universal witness*.*

Soundness: *For every $x \notin S$ and every w, it holds that $V(x,w) = 0$.*

The class $\mathcal{ONP}$ consists of all sets having input-oblivious NP-proof systems.

Clearly, $\mathcal{ONP} \subseteq \mathcal{NP} \cap \mathcal{P}$/poly, since the "universal NP-witnesses" (guaranteed by the completeness condition) can be used as non-uniform advice. We next establish all other claims of Theorem 1.1:

Claim 2.2 $\mathcal{RP} \subseteq \mathcal{ONP}$.

Proof: Let $S \in \mathcal{RP}$. Using error reduction, we obtain a polynomial-time algorithm A and a polynomial p such that for every $x \in S$ it holds that $\Pr_{r \in \{0,1\}^{p(|x|)}}[A(x,r)=1] > 1 - 2^{-|x|}$ (whereas $A(x,r) = 0$ for every $x \notin S$ and r). Thus, there exists a string $r \in \{0,1\}^{p(n)}$ such that $A(x,r) = 1$ for every $x \in S_n$, which yields the desired universal NP-witness (w.r.t $V = A$). ■

Claim 2.3 *$\mathcal{ONP} = \mathcal{NP}$ if and only if $\mathcal{NP} \subset \mathcal{P}/\text{poly}$.*

Proof: Clearly, if $\mathcal{ONP} = \mathcal{NP}$, then $\mathcal{NP} = \mathcal{ONP} \subset \mathcal{P}/\text{poly}$. The proof of the opposite direction uses one main idea of the proof of the Karp–Lipton theorem [9] (i.e., $\mathcal{NP} \subset \mathcal{P}/\text{poly}$ implies that the Polynomial-time Hierarchy collpases to its second level). We follow the presentation in [6, Sec. 3.2.3], where the hypothesis is shown to yield polynomial-size circuits for finding NP-witnesses. Specifically, consider any NP-complete set S, and recall that searching NP-witnesses for $x \in S$ is reducible to deciding S; that is, there exists a relation R such that $S = \{x : \exists w\ (x,w) \in R\}$ and solving the search problem associated with R is reducible to deciding S (cf. [6, Thm. 2.16]). Now, assuming that $\mathcal{NP} \subset \mathcal{P}/\text{poly}$, it follows that this search problem can be solved by polynomial-sized circuits (i.e., by applying the said reduction and using the circuits guaranteed for deciding $S \in \mathcal{NP} \subset \mathcal{P}/\text{poly}$).

The input-oblivious NP-proof system for S will use these (witness finding) circuits as universal witnesses; that is, consider V such that $V(x,w) = 1$ if and only if w is a description of a circuit C_w and $(x, C_w(x)) \in R$, and use w as a universal witness for length n if it describes a $\text{poly}(n)$-size witness-finding circuit for instance length n. Finally, since S is NP-complete (and $S \in \mathcal{ONP}$), it follows that $\mathcal{NP} = \mathcal{ONP}$.[1] ■

Claim 2.4 *If $\mathcal{NE} \neq \mathcal{E}$* (resp., $\mathcal{NE} \not\subseteq \mathcal{BPE}$)*, then $\mathcal{ONP} \neq \mathcal{P}$* (resp., $\mathcal{ONP} \not\subseteq \mathcal{BPP}$).

Proof: Let $S \in \mathcal{NE} \setminus \mathcal{E}$ (resp., $S \in \mathcal{NE} \setminus \mathcal{BPE}$), and let V be a polynomial-time algorithm and c be a constant such that $x \in S$ if and only if there exists $w \in \{0,1\}^N$, where $N = 2^{c|x|}$, such that $V(x,w) = 1$. Defining

$$S' \overset{\text{def}}{=} \{xy : x \in S \wedge |y| = 2^{|x|} - |x|\}, \tag{1}$$

we show that $S' \in \mathcal{ONP}$. Consider a procedure V' such that $V'(xy, uwv) = 1$ if and only if $|y| = 2^{|x|} - |x|$ and $V(x,w) = 1$; that is, on input x' and w', the procedure V' accepts x' if and only if $|x'|$ is a power of two and w' contains a substring that is a NE-witness for the membership of the $\log_2 |x'|$-bit long prefix of x' in the set S. Note that if $xy \in S'$ (and $|y| = 2^{|x|} - |x|$), then there exists $w_x \in \{0,1\}^{|xy|^c}$ such that $V(x, w_x) = 1$. Then, letting $\overline{w}_n = w_{0^n} \cdots w_{1^n} \in \{0,1\}^{2^n \cdot 2^{cn}}$ such that $V(x, w_x) = 1$ if (and only if) $x \in S_n$, it holds that $\overline{w}_n$ is a universal NP-witness for length 2^n: Indeed, for every $z \in S'_{2^n}$ it holds that $V'(z, \overline{w}_n) = 1$, whereas for every $z \notin S'$ and w it holds that $V'(z,w) = 0$. The claim follows, since $S' \notin \mathcal{P}$ (resp., $S' \notin \mathcal{BPP}$). ■

Remark 2.5 (on sparse sets): *The proof of Claim 2.4 can be used to show that every sparse NP-set is in $\mathcal{ONP}$, where a set S is* sparse *if $|S_n| \leq \text{poly}(n)$. The key idea is that if proving membership of any n-bit long string* (in S_n) *can be done by using one of* $\text{poly}(n)$*-many NP-witnesses, then concatenating these witnesses yields a universal NP-witness. The same argument can be applied to show that $\mathcal{NE} = \mathcal{ONE}$, where $\mathcal{ONE}$ is the universal witness analogue of $\mathcal{NE}$* (and so the number of YES-instances of a specific length is polynomial in the length of the corresponding NE-witnesses). *Lastly note that, while every co-sparse is in $\mathcal{P}/\text{poly}$, it is unclear whether every co-sparse NP-set is in $\mathcal{ONP}$.*[2]

[1] We use the fact that if S' is Karp-reducible to a set in $\mathcal{ONP}$, then $S' \in \mathcal{ONP}$. This is obvious if the reduction is length-regular (i.e., it maps instances of the same length to instances of the same length). In general, when reducing S' to S, we may use as universal witnesses for S'_n the concatenation of universal witnesses for S_m for $m = 1, ..., \text{poly}(n)$.

[2] A set S is called co-sparse if $|S_n| \geq 2^n - \text{poly}(n)$. We mention that relative to a random oracle, there exists a co-sparse set in $\mathcal{NP} \setminus \mathcal{ONP}$.

3 Input-Oblivious Interactive Proof Systems (OIP)

When defining an input-oblivious version of $\mathcal{IP}$, we should make sure that the verifier does not communicate the input to the prover, who does not get it. The simplest way to guarantee this feature is to decouple the interaction into two stages: In the first stage, both parties are only presented with the length of the input, and in the second stage the verifier is given the actual input but is disconnected from the prover. Thus, the verifier is decomposed into two parts, denoted V_1 and V_2, and its decision regarding the input x is written as $V_2(x,(P,V_1)(1^{|x|}))$, where $(P,V_1)(1^n)$ denotes the output of V_1 after interacting with the prover P on common input 1^n. (Note that the said output of V_1 may contain its entire view of the interaction with P, and that without loss of generality V_2 may be deterministic (since its coins may be tossed and recorded by V_1).)

Definition 3.1 (input-oblivious interactive proofs – $\mathcal{OIP}$): *A set S has an* input-oblivious interactive proof system *if there exists a probabilistic polynomial-time interactive machine V_1 and a polynomial-time algorithm V_2 such that the following two conditions hold.*

completeness: *There exists a strategy P such that, for every $x \in S$, it holds that* $\Pr[V_2(x,(P,V_1)(1^{|x|}))=1] \geq 2/3$.

If the latter probability always equals 1, then we say that the system has perfect completeness*.*

soundness: *For every $x \notin S$ and every strategy P, it holds that* $\Pr[V_2(x,(P,V_1)(1^{|x|})=1] \leq 1/3$.

The class $\mathcal{OIP}$ consists of all sets having input-oblivious interactive proof systems.

As in the case of $\mathcal{ONP}$, the soundness condition of $\mathcal{OIP}$ maintains the analogous condition of $\mathcal{IP}$.

Theorem 3.2 (on the power of input-oblivious interactive proofs): $\mathcal{OIP} = \mathcal{IP} \cap \mathcal{P}/\text{poly}$. *Furthermore, each set in $\mathcal{IP} \cap \mathcal{P}/\text{poly}$ has an input-oblivious interactive proof system* with perfect completeness.

Proof: To see that $\mathcal{OIP}$ is contained in $\mathcal{P}/\text{poly}$, we first apply error reduction to an input-oblivious interactive proof system for any $S \in \mathcal{OIP}$ such that the error probability on instances of length n is smaller than 2^{-n}. Thus, there exists an output of V_1 (after interacting with P on 1^n), denoted y, such that for every $x \in \{0,1\}^n$ it holds that $x \in S$ if and only if $V_2(x,y) = 1$. This output (i.e., y) can be used as non-uniform advice, which implies that $S \in \mathcal{P}/\text{poly}$.

We now assume that $S \in \mathcal{IP} \cap \mathcal{P}/\text{poly}$, and let $\{C_n\}$ be a family of polynomial-size circuit deciding S. On common input 1^n, the (input oblivious) prover sends C_n to the verifier V_1, and proves to it that C_n is correct (i.e., that for every $x \in \{0,1\}^n$ it folds that $C_n(x) = 1$ iff $x \in S$). Note that the latter assertion can be verified in polynomial-space, and hence it can be proved by an interactive proof (with perfect completeness) [13, 14]. The output of V_1 equals C_n if V_1 were convinced by the proof, and is the identically zero circuit otherwise. Finally, on input x and y (representing V_1's output), algorithm V_2 outputs $C_y(x)$, where C_y is the circuit represented by the string y. Thus, $S \in \mathcal{OIP}$ (and furthermore S has an input-oblivious interactive proof with perfect completeness). ■

The class OMA. The class $\mathcal{OMA}$ is the input-oblivious version of $\mathcal{MA}$, which in turn is a randomized version of $\mathcal{NP}$ (in which the final verification of witnesses is probabilistic). In terms of input-oblivious interactive proofs (i.e., $\mathcal{OIP}$), the class $\mathcal{OMA}$ contains sets having a uni-directional interactive proof system of perfect completeness (in which, first the prover sends a message, and then the verifier tosses some coins). We observe that Lautemann's argument [12], which has been used to show $\mathcal{BPP} \subseteq \mathcal{MA}$, allows showing that $\mathcal{BPP} \subseteq \mathcal{OMA}$.

Proposition 3.3 *$\mathcal{BPP} \subseteq \mathcal{OMA}$.*

Proof: Let $S \in \mathcal{BPP}$ and consider an algorithm A such that $\Pr_{r\in\{0,1\}^{p(|x|)}}[A(x,r) = \chi_S(x)] > 1 - 2^{-|x|}$, where $\chi_S(x) = 1$ is $x \in S$ and $\chi_S(x) = 0$ otherwise. Recall that the standard argument asserts that for every $x \in S$ there exists $s_1, ..., s_{p(|x|)} \in \{0,1\}^{p(|x|)}$ such that for every $r \in \{0,1\}^{p(|x|)}$ it holds that $\bigvee_{i\in[p(|x|)]} A(x, r \oplus s_i) = 1$, whereas for any $x \notin S$ and $s_1, ..., s_{p(|x|)} \in \{0,1\}^{p(|x|)}$ it holds that $\Pr_{r\in\{0,1\}^{p(|x|)}}[\bigvee_{i\in[p(|x|)]} A(x, r \oplus s_i) = 1]$ is smaller than $p(|x|)/2^{|x|}$. We just note that, for every $n \in \mathsf{N}$, there exists $s_1, ..., s_{p(|x|)+|x|} \in \{0,1\}^{p(|x|)}$ such that for every $x \in S_n$ and every $r \in \{0,1\}^{p(|x|)}$ it holds that $\bigvee_{i\in[p(|x|)+|x|]} A(x, r \oplus s_i) = 1$. ■

4 Input-Oblivious Versions of PCP and ZK

In this section, we consider input-oblivious versions of the classes PCP (Probabilistically Checkable Proofs) and ZK (Zero-Knowledge interactive proofs). In both cases, we provide evidence that the said classes extend beyond the obvious (e.g., beyond $\mathcal{P}$), but note that they are unlikely to contain all of $\mathcal{ONP}$.

4.1 Input-Oblivious PCP

For sake of simplicity, we focus on PCP system of logarithmic randomness complexity and constant query complexity, and identify such systems with the term PCP.

Definition 4.1 (input-oblivious probabilistically checkable proofs – OPCP): *A set S has an* input-oblivious PCP system *if there exists a probabilistic polynomial-time oracle machine V of logarithmic randomness complexity and constant query complexity such that the following two conditions hold.*

Completeness: *For every $n \in \mathsf{N}$ there exists an oracle π_n such that, on input any $x \in S_n$ and access to the oracle π_n, machine V always accepts x.*

Soundness: *For every $x \notin S$ and every oracle π, on input x and access to oracle π, machine V rejects x with probability at least $\frac{1}{2}$.*

The class $\mathcal{OPCP}$ consists of all sets having input-oblivious PCP systems.

Clearly, $\mathcal{OPCP} \subseteq \mathcal{ONP}$, but the converse may not hold. It is not even clear that every sparse NP-set is in $\mathcal{OPCP}$. Still, Claim 2.4 extends to $\mathcal{OPCP}$.

Proposition 4.2 (on the power of input-oblivious PCPs): *If $\mathcal{NE} \neq \mathcal{E}$* (resp., *$\mathcal{NE} \not\subseteq \mathcal{BPE}$*), *then $\mathcal{OPCP} \neq \mathcal{P}$* (resp., *$\mathcal{OPCP} \not\subseteq \mathcal{BPP}$*).

Proof: Recalling the construction used in the proof of Claim 2.4, we obtain a set S' in $\mathcal{ONP}$ that has the following additional property: There exist a polynomial-time computable length-preserving function, denoted f, such that f maps all n-bit long strings to a polynomial-time constructible set of representatives while maintaining membership in S'; that is, the following conditions hold.

1. The set $\{f(z) : z \in \{0,1\}^n\}$ is poly(n)-time constructible;
2. $z \in S'$ if and only if $f(z) \in S'$.

(Referring to the set S' as defined in Eq. (1), consider $f(xy) = x0^{|y|}$, where $|y| = 2^{|x|} - |x|$.) Thus, proving membership of an arbitrary n-bit long string in S' reduces to proving membership in S' of the corresponding representative, which means that we need only take care of poly(n)-many instance-witness pairs. Applying the PCP Theorem (cf. [2, 1]) to the inputs in the range of f along with corresponding NP-witnesses, we obtain the desired input-oblivious PCP. Specifically, on input z, the verifier computes $r \leftarrow f(z)$, determines the index of r in the set $\{f(s) : s \in \{0,1\}^{|z|}\}$, and accesses the corresponding portion of the proof oracle, where the latter portion contains the proof oracle produced for the input $f(z)$ using a corresponding NP-witness (which may just be a universal NP-witness for length $|z| = |f(z)|$). ■

Digest. The proof of Proposition 4.2 does not use the fact that $S' \in \mathcal{ONP}$, but rather uses the additional structure guaranteed by the polynomial-time computable function f. This seems required since in standard PCP constructions the proof oracle depends on the input (and not only on the corresponding NP-witness).[3] Note that it is even unclear whether $\mathcal{RP}$ is in $\mathcal{OPCP}$, although clearly $\mathcal{P} \subseteq \mathcal{OPCP}$. On the other hand, we note that Condition (1) can be relaxed such that it is only required that the set $R_n \stackrel{\rm def}{=} \{f(z) : z \in \{0,1\}^n\}$ has poly(n)-size (rather than being poly(n)-time constructible). Actually, it suffices to required that $|R_n \cap S_n| \leq p(n)$, for some fixed polynomial p (and all n). This relaxation is shown to suffice by using a suitable hashing scheme to map elements of R_n to indices in, say, $[3p(n)]$ such that no two elements are mapped to the same index, and using these indices as in the proof of Proposition 4.2. Specifically, we use a poly(n)-size family of efficiently computable hashing functions, H_n, that map $\{0,1\}^n$ to $[3p(n)]$ such that for every two distinct $a, b \in \{0,1\}^n$ it holds that $\Pr_{h \in H_n}[h(a) = h(b)] < 1/2p(n)$.[4] On input $z \in \{0,1\}^n$, the modified verifier computes $r \leftarrow f(z)$, selects uniformly $h \in H_n$, and accesses the portion of the proof oracle that corresponds to $(h, h(r))$, which is supposed to contain a proof that there exists $w \in R_n \cap S_n$ such that $h(f(w)) = v$, where $v \leftarrow h(r)$. Note that the latter NP-assertion refers only to h and v (and n), and so we may use any NP-witness for it (and obtain a corresponding PCP oracle proof). Hence, the completeness condition is satisfied by a proof oracle that is a concatenation of proofs for the various possible values of (h, v), whereas on input $z \in S_n$ the verifier always accesses a portion that corresponds to a valid assertion (since it uses $v = h(f(z))$). The soundness condition holds because any $z \notin S_n$ is mapped with constant probability to an $h \circ f$-image (for a random $h \in H_n$) that has no $h \circ f$-preiamge in $R_n \cap S_n$.

4.2 Input-Oblivious ZK

The class $\mathcal{OZK}$ consists of sets having an input-oblivious interactive proof system in which the prescribed prover is zero-knowledge in the standard (complexity oriented) sense.[5] This definition requires efficient simulation of the (prescribed) verifier's view of the interaction, based solely on the verifier's actual input. (Indeed, here we refer to the verifier as the combination of the two stages,

[3] Thus, it is not clear that a universal NP-witness yields a universal PCP proof oracle.

[4] Such constructions are presented in [7, 10, 11, 15].

[5] The standard (complexity theoretic) definition of zero-knowledge requires efficient simulation of the view of the prescribed verifier (of the interaction with the prover); a stronger definition, commonly used in cryptography (cf. [5, Sec. 4.3.1]), requires efficient simulation of the view of arbitrary probabilistic polynomial-time adversaries. We note that our positive results extend also to the general (i.e., adversarial verifier) notion of zero-knowledge.

denoted V_1 and V_2, and note that this combined verifier gets the actual input (rather than merely its length).)[6]

Clearly, $\mathcal{BPP} \subseteq \mathcal{OZK}$ (since any set in $\mathcal{BPP}$ has an input-oblivious interactive proof system in which the prescribed prover does nothing, and hence is easily simulatable). It turns out that $\mathcal{OZK}$ may extend beyond $\mathcal{BPP}$ only in the case of sets for which it is hard to find YES-instances of any desired length.

Proposition 4.3 (on the power of input-oblivious zero-knowledge proofs):

1. *If $S \in \mathcal{OZK}$ and there exists a probabilistic polynomial-time algorithm A such that* $\Pr[A(1^n) \in S_n] \geq 2/3$ *holds for all sufficiently large n, then $S \in \mathcal{BPP}$.*

2. *If $S \in \mathcal{ZK}$ and $|S_n| \leq 1$ for all sufficiently large n, then $S \in \mathcal{OZK}$. Thus, if $\mathcal{NE} \not\subseteq \mathcal{BPE}$ and one-way functions exist, then $\mathcal{OZK} \not\subseteq \mathcal{BPP}$.*

Proof: For the negative result of Part 1 we may weaken the definition of zero-knowledge, and only consider simulating the output of the first stage (rather than the verifier's view of this stage). That is, referring to the notation in Definition 3.1, we consider the requirement that, on input $x \in S$, one can efficiently simulate $(P, V_1)(1^{|x|})$; that is, there exists a probabilistic polynomial-time machine M such that $\{M(x)\}_{x \in S}$ and $\{(P, V_1)(1^{|x|})\}_{x \in S}$ are computationally indistinguishable (by polynomial-size circuits). Let S and A be as in Part 1, and let P, V_1, M be as above (and V_2 as in Definition 3.1). Actually, we assume (w.l.o.g.) that the interactive proof has error probability at most 0.1 (rather than at most 1/3). Then, for all but finitely many $z \in S$ and all $x \in \{0,1\}^{|z|}$, it holds that

$$\Pr[V_2(x, M(z))=1] \;=\; \Pr[V_2(x, (P, V_1)(1^{|z|}))=1] \pm 0.01,$$

because otherwise x can be incorporated in a small circuit that distinguishes $M(z)$ from $(P, V_1)(1^{|z|})$. Thus, for all but finitely many x, it holds that

$$\Pr[V_2(x, M(A(1^{|x|})))=1] \;=\; \Pr[V_2(x, (P, V_1)(1^{|x|}))=1] \pm 0.35,$$

because $\Pr[A(1^{|x|}) \in S_{|x|}] > 0.66$. This suggests an efficient probability decision procedure for S: *On input x, invoke $V_2(x, M(A(1^{|x|})))$, and rule accordingly.* Observing that this decision procedure has error probability at most $0.1 + 0.35 = 0.45$, it follows that $S \in \mathcal{BPP}$.

Turning to Part 2, we first consider a set $S \in \mathcal{ZK}$ such that $|S_n| \leq 1$, and show that it is in $\mathcal{OZK}$. On input 1^n, the prover first determines the unique n-bit string in S_n (or halts if no such string exists), sends it to the verifier, then the two parties proceed using the standard zero-knowledge proof, and at the end the verifier (i.e., V_2) checks whether the input equals the n-bit long string sent by the prover (at the beginning of the interaction). Thus, $S \in \mathcal{OZK}$. Lastly, assuming $\mathcal{NE} \not\subseteq \mathcal{BPE}$ (and the existence of one-way functions), we obtain a set $S \in \mathcal{ZK} \setminus \mathcal{BPP}$ such that $|S_n| \leq 1$ (by combining a twist on the construction presented in the proof of Claim 2.4 with a standard zero-knowledge proof for sets in $\mathcal{NP}$).[7] ■

[6] A stronger requirement (which mandates simulating the first stage based solely on the length of the actual input) is discussed in Remark 4.5.

[7] Given $S' \in \mathcal{NE} \setminus \mathcal{BPE}$, consider the unary set $S = \{1^{2^{|x|}+\text{idx}(x)-1} : x \in S'\}$, where idx$(x)$ is the index of x in the standard lexicographic order of all $|x|$-bit strings. Clearly $S \in \mathcal{NP} \setminus \mathcal{BPP}$ and $|S_n| \leq 1$. Recall that the standard construction of zero-knowledge proofs for sets in $\mathcal{NP}$ uses any one-way function [5, Sec. 4.4].

Remark 4.4 ($\mathcal{OZK}$ may extend beyond $\mathcal{ONP}$): *While $\mathcal{OZK} \subseteq \mathcal{OIP}$ holds trivially, assuming that $\mathcal{NE} \neq \mathcal{ESPACE}$ yields that $\mathcal{OZK}$ extends beyond $\mathcal{NP}$. Analogously to the proof of Part 2 in Proposition 4.3, the foregoing assumption yields a unary set in $\mathcal{PSPACE} \setminus \mathcal{NP}$, and using zero-knowledge proofs for sets in $\mathcal{IP}$* (cf. [5, Thm. 4.4.12]) *we are done.*

Remark 4.5 (strong zero-knowledge): *We say that an input-oblivious interactive proof system is* strongly zero-knowledge *if one can efficiently simulate the verifier's view of the first stage based solely on $1^{|x|}$* (rather than based on x)*. It is easy to see that such proof systems exist only for sets in $\mathcal{BPP}$, even if it is only required to efficiently simulate the verifier's output of the first stage* (i.e., $(P, V_1)(1^{|x|})$) *based on $1^{|x|}$.*

References

[1] S. Arora, C. Lund, R. Motwani, M. Sudan and M. Szegedy. Proof Verification and Intractability of Approximation Problems. *JACM*, Vol. 45, pages 501–555, 1998. Preliminary version in *33rd FOCS*, 1992.

[2] S. Arora and S. Safra. Probabilistic Checkable Proofs: A New Characterization of NP. *JACM*, Vol. 45, pages 70–122, 1998. Preliminary version in *33rd FOCS*, 1992.

[3] L. Babaim, Lance Fortnow and Carsten Lund. Non-Deterministic Exponential Time has Two-Prover Interactive Protocols. *Computational Complexity*, Vol. 1, pages 3–40, 1991.

[4] V.T. Chakaravarthy and S. Roy. Oblivious Symmetric Alternation. In *23rd STACS*, Springer LNCS 3884, pages 230–241, 2006.

[5] O. Goldreich. *Foundation of Cryptography – Basic Tools.* Cambridge University Press, 2001.

[6] O. Goldreich. *Computational Complexity: A Conceptual Perspective.* Cambridge University Press, 2008.

[7] O. Goldreich and A. Wigderson. Tiny Families of Functions with Random Properties: A Quality–Size Trade–off for Hashing. *Journal of Random structures and Algorithms*, Vol. 11 (4), pages 315–343, 1997. Preliminary version in *26th STOC*, 1994.

[8] Russell Impagliazzo, Valentine Kabanets and Avi Wigderson. In Search of an Easy Witness: Exponential Time vs. Probabilistic Polynomial Time. In *IEEE Conference on Computational Complexity*, pages 2–12, 2001.

[9] R.M. Karp and R.J. Lipton. Some Connections Between Nonuniform and Uniform Complexity Classes. In *12th STOC*, pages 302-309, 1980.

[10] H. Krawczyk. LFSR-based Hashing and Authentication. In *CRYPTO'94*, LNCS (Vol. 839), Springer, pages 129–139, 1994.

[11] H. Krawczyk. New Hash Functions For Message Authentication. In *EuroCrypt'95*, LNCS (Vol. 921), Springer, pages 301–310, 1995.

[12] C. Lautemann. BPP and the Polynomial Hierarchy. *IPL*, Vol. 17, pages 215–217, 1983.

[13] C. Lund, L. Fortnow, H. Karloff, and N. Nisan. Algebraic Methods for Interactive Proof Systems. *JACM*, Vol. 39, No. 4, pages 859–868, 1992. Preliminary version in *31st FOCS*, 1990.

[14] A. Shamir. IP = PSPACE. *JACM*, Vol. 39, No. 4, pages 869–877, 1992. Preliminary version in *31st FOCS*, 1990.

[15] A. Srinivasan and D. Zuckerman. Computing with Very Weak Random Sources. *SICOMP*, Vol. 28 (4), pages 1433–1459, 1999. Preliminary version in *35th FOCS*, 1994.