\[ \sum_{\beta_t} |\Pr[P^*(x, \alpha_1 \beta_1 \ldots \alpha_t) = \beta_t] - \Pr[P(x, \alpha_1 \beta_1 \ldots \alpha_t) = \beta_t]| \]

\[ \leq \sum_{\beta_1, \ldots, \beta_t} |\Pr[View_{(P,V^*)}(x, \alpha_1 \ldots \alpha_t) = \alpha_1 \beta_1 \ldots \alpha_t] \cdot \Pr[P^*(x, \alpha_1 \beta_1 \ldots \alpha_t) = \beta_t] - \Pr[View_{(P,V^*)}(x, \alpha_1 \ldots \alpha_t) = \alpha_1 \beta_1 \ldots \alpha_t] \cdot \Pr[P(x, \alpha_1 \beta_1 \ldots \alpha_t) = \beta_t]| \]

\[ \leq \sum_{\beta_1, \ldots, \beta_t} |\Pr[S(x, \alpha_1 \ldots \alpha_t) = \alpha_1 \beta_1 \ldots \alpha_t] \cdot \Pr[P^*(x, \alpha_1 \beta_1 \ldots \alpha_t) = \beta_t] - \Pr[View_{(P,V^*)}(x, \alpha_1 \ldots \alpha_t) = \alpha_1 \beta_1 \ldots \alpha_t] \cdot \Pr[P(x, \alpha_1 \beta_1 \ldots \alpha_t) = \beta_t]| \]

Since \( P^*(x, \alpha_1 \beta_1 \ldots \alpha_t) \) is defined to behave exactly like the simulator on the prover step at round \( t \) given the history so far and the verifier messages \( \alpha_1 \ldots \alpha_t \) as auxiliary input, we may rewrite the first term as \( \Pr[S(x, \alpha_1 \ldots \alpha_t) = \alpha_1 \beta_1 \ldots \alpha_t \beta_t] \). Also, since \( P \) is the prover in the interaction \((P,V^*)\), we may rewrite the last term as: \( -\Pr[View_{(P,V^*)}(x, \alpha_1 \ldots \alpha_t) = \alpha_1 \beta_1 \ldots \alpha_t] \cdot \Pr[P(x, \alpha_1 \beta_1 \ldots \alpha_t) = \beta_t] \). Thus, we get:

\[ \leq \sum_{\beta_1, \ldots, \beta_t} |\Pr[S(x, \alpha_1 \ldots \alpha_t) = \alpha_1 \beta_1 \ldots \alpha_t] - \Pr[View_{(P,V^*)}(x, \alpha_1 \ldots \alpha_t) = \alpha_1 \beta_1 \ldots \alpha_t]| \]

\[ + |\Pr[S(x, \alpha_1 \ldots \alpha_t) = \alpha_1 \beta_1 \ldots \alpha_t] - \Pr[View_{(P,V^*)}(x, \alpha_1 \ldots \alpha_t) = \alpha_1 \beta_1 \ldots \alpha_t]| \cdot \Pr[P^*(x, \alpha_1 \beta_1 \ldots \alpha_t) = \beta_t] \]

\[ \leq 2\delta \]

and we are done. ■

**Corollary 5.4** Let \((P,V)\) be statistical \(ZK\), let \( x \in L \cap \{0,1\}^n \), and let \( \hat{V} \) be any (possibly cheating) verifier. Then

\[ \sum_{\alpha_1 \beta_1, \ldots, \alpha_t} \Pr[View_{(P,\hat{V})}(x) = \alpha_1 \beta_1 \ldots \alpha_t]. \]

\[ \sum_{\beta_t \in \{0,1\}^n} |\Pr[P^*(x, \alpha_1 \beta_1 \ldots \alpha_t) = \beta_t] - \Pr[P(x, \alpha_1 \beta_1 \ldots \alpha_t) = \beta_t]| < 2\delta \]

**Proof:** The second part of Lemma 5.3 guarantees the above inequality for all possible strings \( \alpha_1 \alpha_2 \ldots \alpha_t \). Namely, the view of \((P,V^*)\) with auxiliary input \( (\alpha_1 \ldots \alpha_t) \) contains only verifier messages which are the predetermined \( \alpha_1 \ldots \alpha_t \). Thus, the inequality also holds when the strings \( \alpha_1 \alpha_2 \ldots \alpha_t \) are produced by any distribution space generated by any verifier \( \hat{V} \). Note that the distribution on \( \beta_1 \ldots \beta_{t-1} \) does not change since we use the same prover \( P \) as in the lemma. ■

Note that \( \delta \) is negligible. Combining this Corollary with Theorem 4.3 yields the desired conclusion.
Theorem 5.5 Suppose $L$ has a statistical ZK interactive proof system $(P, V)$. Then there is a PPT oracle machine $P_{\text{eff}}$ such that $(P_{\text{eff}}, V)$ is a statistical ZK interactive proof system for $L$.

Proof: Let $d^*(n)$ equal
\[
\max_{x \in L, |x|=n} \sup_{a \in \{0,1\}^*} d(S^*(x, a), \text{View}_{(P, V^*)}(x, a)) .
\]
We know that this function is negligible.
Let $P^*$ be a universal prover for $(P, V)$. Let $\hat{V}$ be (any) verifier. Let $x \in L \cap \{0,1\}^n$ and $a \in \{0,1\}^*$. We begin by showing that Corollary 5.4 implies that the distance between $\text{View}_{(P, \hat{V})}(x, a)$ and $\text{View}_{(P^*, \hat{V})}(x, a)$ is at most $2g(n)d^*(n)$. Note that since $d^*$ is negligible, so is $2gd^*$.

We bound the difference between these two distributions by an induction on the round number. Namely, the induction claim is that in round $t$, $1 \leq t \leq g(n)$ it holds that
\[
\sum_{\alpha_1 \beta_1 \ldots \alpha_t \beta_t} \left| \Pr[\text{View}_{(P, \hat{V})}(x, a) = \alpha_1 \beta_1 \ldots \alpha_t \beta_t] - \Pr[\text{View}_{(P^*, \hat{V})}(x, a) = \alpha_1 \beta_1 \ldots \alpha_t \beta_t] \right| \leq (t-1) \cdot 2d^*(n).
\]

The verifier behaves in exactly the same manner in both interactions and thus for any prefix $\alpha_1$ both distributions are equal. Combining this with Corollary 5.4 we get that the claim holds for $t=1$.

Let us show the induction step. The induction hypothesis implies that
\[
\sum_{\alpha_1 \beta_1 \ldots \alpha_{t-1} \beta_{t-1}} \left| \Pr[\text{View}_{(P, \hat{V})}(x, a) = \alpha_1 \beta_1 \ldots \alpha_{t-1} \beta_{t-1}] - \Pr[\text{View}_{(P^*, \hat{V})}(x, a) = \alpha_1 \beta_1 \ldots \alpha_{t-1} \beta_{t-1}] \right| \leq (t-1) \cdot 2d^*(n).
\]
Since the verifier steps behave the same in both distributions we also have
\[
\sum_{\alpha_1 \beta_1 \ldots \alpha_t} \left| \Pr[\text{View}_{(P, \hat{V})}(x, a) = \alpha_1 \beta_1 \ldots \beta_{t-1} \alpha_t] - \Pr[\text{View}_{(P^*, \hat{V})}(x, a) = \alpha_1 \beta_1 \ldots \beta_{t-1} \alpha_t] \right| \leq (t-1) \cdot 2d^*(n). \tag{2}
\]

Let us now show that the induction claims hold for round $t$.
\[
\sum_{\alpha_1 \beta_1 \ldots \alpha_t \beta_t} \left| \Pr[\text{View}_{(P, \hat{V})}(x, a) = \alpha_1 \beta_1 \ldots \alpha_t \beta_t] - \Pr[\text{View}_{(P^*, \hat{V})}(x, a) = \alpha_1 \beta_1 \ldots \alpha_t \beta_t] \right|
= \sum_{\alpha_1 \beta_1 \ldots \alpha_t \beta_t} \left| \Pr[\text{View}_{(P, \hat{V})}(x, a) = \alpha_1 \beta_1 \ldots \beta_{t-1} \alpha_t] \cdot \Pr[P(\alpha_1 \beta_1 \ldots \alpha_t) = \beta_t] 
- \Pr[\text{View}_{(P^*, \hat{V})}(x, a) = \alpha_1 \beta_1 \ldots \beta_{t-1} \alpha_t] \cdot \Pr[P^*(\alpha_1 \beta_1 \ldots \alpha_t) = \beta_t] \right|
\]
Similarly to the proof of Corollary 5.4 this can be bounded by:
\[
\leq \sum_{\alpha_1 \beta_1 \ldots \alpha_t \beta_t} \left| \Pr[P(\alpha_1 \beta_1 \ldots \alpha_t) = \beta_t] - \Pr[P^*(\alpha_1 \beta_1 \ldots \alpha_t) = \beta_t] \right|
+ \left| \Pr[\text{View}_{(P, \hat{V})}(x, a) = \alpha_1 \beta_1 \ldots \beta_{t-1} \alpha_t] - \Pr[\text{View}_{(P^*, \hat{V})}(x, a) = \alpha_1 \beta_1 \ldots \beta_{t-1} \alpha_t] \right|
\]

Using Equation (2) and Corollary 5.4 we are done.

The negligible distance of $\text{View}_{(P, \hat{V})}(x, a)$ and $\text{View}_{(P^*, \hat{V}^*)(x, a)}$ implies that $(P^*, V)$ is a statistical ZK interactive proof system for $L$. To see this first note that setting $\hat{V}$ in the above statement to be the honest verifier $V$ implies that the completeness condition holds for $(P^*, V)$. Since $V$ is unchanged the soundness of course still holds. And if $S_{\hat{V}}$ is a statistical ZK $P$-simulator for $\hat{V}$ then it is also a statistical ZK $P^*$-simulator for $\hat{V}$.

Finally we note that by Theorem 4.3 a distribution within negligible distance of $P^*(x, \alpha_1; \ldots, \alpha_i) = P_s(x, \alpha_1; \ldots, \alpha_i)$ is computable in PPT with an NP oracle (denote this machine by $P_{NP}^{e}$). Setting $P_{NP}^{e}$ to be the prover instead of $P^*$ does not foil the statistical ZK property of the interaction, since for any $\hat{V}$ and any auxiliary input $a \in \{0,1\}^*$ it holds that $\text{View}_{(P^*, \hat{V})}(x, a)$ and $\text{View}_{(P_{NP}^{e}, \hat{V})}(x, a)$ have negligible distance. $
$

We remark that perfect ZK can be preserved at the computational price of allowing $P_{eff}$ to have a $\Sigma^p_2$ oracle instead of an NP one:

**Theorem 5.6** Suppose $L$ has a perfect ZK interactive proof system $(P, V)$. Then there is PPT oracle machine $P_{eff}$ such that $(P_{eff}, V)$ is a perfect ZK interactive proof system for $L$.

**Proof:** Use the same argument as in the proof of the previous theorem (except of course use Lemma 5.3 (2) rather than Corollary 5.4), but implement $P_s$ using the uniform generation algorithm of [22] (rather than the almost uniform generation of Theorem 3.2). $
$

It is possible to extend these techniques to derive bounds on the complexity of the prover in a proof of knowledge complexity $k(n) > 0$ (cf. Definition 2.8) while preserving the KC. However, the bounds we get are not nearly as good; a straightforward application of our techniques shows only that if $L$ has an interactive proof of knowledge complexity $k(n) \leq n^{O(1)}$ then $L$ has an interactive proof $(P, V)$ of knowledge complexity $k(n)$ in which $P$ is a probabilistic, exponential time machine with a NEXP oracle. Not only is the complexity much greater than for ZK, but we don’t know how to do better even if we assume $k(n)$ is just 1 (rather than zero). Deriving better bounds on the complexity of non-zero KC provers while preserving the KC is another open question.

In the next section we will see however that if we don’t want to preserve the KC then better bounds are achievable, and this has applications to bounding the complexity of low KC languages.

### 6 Bounds on the Complexity of Low KC languages

In this section we present a result which ties the knowledge complexity of an interactive proof for a language $L$ to the time complexity of $L$. This result relies on an extension to positive KC of our techniques from the previous sections.

Throughout this paper, we use the definition of knowledge complexity in the *fraction* sense [17]. Loosely speaking, we say that a protocol $(P, V)$ has knowledge complexity $k(n)$ if for any verifier $\hat{V}$ there exists a simulator $S_{\hat{V}}$ with the following “good subspace” property. For any $x \in L$ and auxiliary input $a \in \{0,1\}^*$ there exists a subspace (denoted $\text{SUCC}_{x,a}$) of
the set $\{0,1\}^{p(n)}$ of the simulator’s possible random tapes which has density $\geq 2^{-k(n)}$ (i.e., if we choose a tape at random from $\{0,1\}^{p(n)}$ then this tape is in $Succ_{x,a}$ with probability $\geq 2^{-k(n)}$), and the output of the simulator on input $x,a$ and a uniformly chosen random tape in $Succ_{x,a}$ is statistically close to a random element of $View(\cdot,\cdot)(x,a)$. The formal definition appears in §2.

There are two other definitions of non-zero KC: the oracle definition and the hint definition [17]. It is shown in [17] that the oracle definition is equal up to an additive constant to the fraction definition. The proof of the equivalence result there does not change the interactive proof, and specifically, the equivalence holds also when a restriction is imposed on the number of rounds in the interaction. Therefore, our results apply to the oracle definition as well.

Our results apply also to the hint definition. However, with this definition, it is easy to prove even stronger results, and such stronger results already appear in [17]. Nevertheless, several reasons are presented in [17] why not to regard the hint definition as an appropriate measure.

Recall (Definition 2.3) that we have adopted the convention that an interactive proof has a negligible error probability. Let us now proceed to our result.

6.1 Overview

We are given a $g(n)$ round interactive proof system $(P,V)$ for $L$ with statistical KC $k(n)$, and we suppose $g(n)k(n) = O(\log n)$. Our goal is to show that $L \in BPP^{NP}$. We begin with an overview of the proof.

We prove more than required. We show that under the given conditions, there exists an interactive proof for $L$ in which the prover is a PPT machine with an access to an NP oracle. As the prover has the power to run the verifier as well, this implies that $L \in BPP^{NP}$.

Let $S$ be a statistical KC $k(n)$ P-simulator for the honest verifier $V$ (cf. Definition 2.7), and $P_S$ the simulation based prover for $S$ (cf. Definition 4.2). We let $P^*$ be the prover derived from $P_S$ by setting the auxiliary input to the empty string: $P^*(x,\alpha_1\beta_1\ldots\alpha_t) = P_S(x,\lambda,\alpha_1\beta_1\ldots\alpha_t)$ (intuitively, think of $P^*$ as being $P_S$; the difference is only a technicality). We show that $(P^*,V)$ has the following “separability” property: (1) if $x \in L$ then $V$, interacting with $P^*$, accepts with a probability that is greater than $n^{-O(1)}$, whereas, (2) if $x \not\in L$ then $V$, interacting with any prover, accepts with a negligible probability. This is Lemma 6.3 (below).

Next, we use Theorem 4.3 to show that there is PPT machine with an NP oracle that can compute a distribution close enough to $P^*$ such that the separability property is maintained. This gives us a PPT with NP oracle prover $\overline{P}$ such that $(\overline{P},V)$ has the separability property. Finally, we observe that standard amplification techniques can be used to reduce the error probability and transform $(\overline{P},V)$ into an interactive proof system (cf. Definition 2.3) for $L$ without increasing the power of the prover.

Remark 6.1 We note that although the definition of knowledge complexity (Definition 2.8) guarantees a KC $k(n)$ P-simulator for each possible verifier, we use only the KC $k(n)$ P-simulator for the honest verifier $V$. Therefore the validity of the theorem (Theorem 6.4)
requires only that the interactive proof has knowledge complexity \( k(n) \) with respect to the honest verifier.

**Remark 6.2** For proving the result of this section we do not need to use the universal verifier nor to exploit the auxiliary string of the verifier and simulator. Therefore, throughout this section we shall use only the honest verifier with an auxiliary string set to \( \lambda \).

### 6.2 Our Results

The main lemma is the following.

**Lemma 6.3** Suppose \((P, V)\) is a \(g(n)\) round interactive proof for \(L\), and \(S\) is a statistical KC \(k(n)\) P-simulator for \(V\). Suppose also that \(g(n)k(n) = O(\log n)\). Define the prover \(P^*\) by

\[
P^*(x, \alpha_1 \beta_1 \ldots \alpha_t) = P_S(x, \lambda, \alpha_1 \beta_1 \ldots \alpha_t)
\]

where \(P_S\) is the simulation based prover for \(S\). Then \((P^*, V)\) satisfies the following “separability” property:

1. There is a constant \(c \geq 0\) such that for any \(x \in L\) the probability that \(V\) accepts at the end of the interaction with \(P^*\) on common input \(x\) is \(\geq |x|^{-c}\).
2. For any (cheating) prover \(\tilde{P}\) and common input \(x \notin L\) the probability that \(V\) accepts at the end of the interaction with \(\tilde{P}\) is negligible.

Note that we are making no claims about the knowledge complexity of the system \((P^*, V)\) constructed in the above lemma. This is in contrast to our results in §5 where we were trying to preserve the knowledge complexity (in the particular case where this knowledge complexity was zero).

The (technical) proof of Lemma 6.3 appears in §6.3. Let us now state and prove our theorem using Lemma 6.3.

**Theorem 6.4** Suppose \(L\) has a \(g(n)\) round interactive proof with statistical knowledge complexity \(k(n)\), and suppose also that \(g(n)k(n) = O(\log n)\). Then \(L\) is in \(\text{BPP}^{\text{NP}}\).

**Proof:** By assumption there is a \(g(n)\) round interactive proof \((P, V)\) with statistical knowledge complexity \(k(n)\). It suffices to show that there is a PPT machine \(P_{\text{eff}}\) such that \((P_{\text{eff}}^{\text{NP}}, V)\) is an interactive proof system for \(L\).

Let \(S\) be a statistical KC \(k(n)\) P-simulator for \(V\). Let \(P_S\) be the corresponding simulation based prover, and let \(P^*\) be as in Lemma 6.3. Then \((P^*, V)\) has the separability property. We let \(\mathcal{P}\) denote the PPT with NP oracle prover which is given by using \(T\) rather than \(P_S\) in the definition of \(P^*\), where \(T\) is the machine that is guaranteed in Theorem 4.3, setting \(\delta = 2^{-n}\) in that theorem so that the output of \(T\) is \(2^{-n}\)-close to the output of \(P_S\).

Let \(x \in L\). Observe that View\((P^*, V)\)(\(x, \lambda\)) and View\((P_{\text{eff}}^{\text{NP}}, V)\)(\(x, \lambda\)) are \(g2^{-n}\)-close, and thus, on common input \(x\), the difference between the probability that \(P^*\) convinces \(V\) to accept and the probability that \(\mathcal{P}\) convinces \(V\) to accept is at most \(g2^{-n}\). Since \(g2^{-n}\) is negligible, this means that the separability property still holds:
There is a constant \( c \geq 0 \) such that for any \( x \in L \) the probability that \( V \) accepts at the end of the interaction with \( \overline{P} \) on common input \( x \) is \( \geq |x|^{-c} \).

(2) For any (cheating) prover \( \hat{P} \) and common input \( x \notin L \) the probability that \( V \) accepts at the end of the interaction with \( \hat{P} \) is negligible.

Next, standard amplification techniques can be used to turn \((\overline{P}, V)\) into an interactive proof for \( L \) without increasing the complexity of the prover. Finally, we note that a PPT machine with an NP oracle can play the role of both parties \( \overline{P} \) and \( V \) and decide whether \( x \in L \).

### 6.3 Proof of Lemma 6.3

Let us restate Lemma 6.3 and prove it.

**Lemma 6.3** Suppose \((P, V)\) is a \( g(n) \) round interactive proof for \( L \), and \( S \) is a statistical KC \( k(n) \) P-simulator for \( V \). Suppose also that \( g(n)k(n) = O(\log n) \). Define the prover \( P^* \) by

\[
P^*(x, \alpha_1, \beta_1, \ldots, \alpha_t) = P_S(x, \lambda, \alpha_1, \beta_1, \ldots, \alpha_t)
\]

where \( P_S \) is the simulation based prover for \( S \). Then \((P^*, V)\) satisfies the following “separability” property:

(1) There is a constant \( c \geq 0 \) such that for any \( x \in L \) the probability that \( V \) accepts at the end of the interaction with \( P^* \) on common input \( x \) is \( \geq |x|^{-c} \).

(2) For any (cheating) prover \( \hat{P} \) and common input \( x \notin L \) the probability that \( V \) accepts at the end of the interaction with \( \hat{P} \) is negligible.

**Proof:** We begin with some intuition.

Condition (2) follows from the soundness condition of the interactive proof \((P, V)\); the concern is condition (1). Fix an \( x \in L \). Recall that when the random tape of the simulator \( S \) is uniformly picked in \( SUCC_{x, \lambda} \), the output distribution of \( S \) is statistically close to the distribution of the conversations between \( P \) and \( V \). We show that in each round of the interaction our new prover \( P^* \) (using \( P_S \)) picks its answer using a random tape in \( SUCC_{x, \lambda} \) with probability \( \geq 2^{-k} \). Therefore with probability at least \( 2^{-k} = n^{-O(1)} \), \( P_S \) picks all its answers using random tapes from \( SUCC_{x, \lambda} \). Intuitively, he thus gains an advantage which is \( 2^{-k} = n^{-O(1)} \) times the advantage of the original prover \( P \). However this is a simplification, because we must deal with the fact that in all but the first round, \( P_S \) is not sampling uniformly from the space of all coin tosses. Rather, he is sampling conditional to the prefix of the conversation so far being some particular value and in this conditional space it is not always the case that \( SUCC_x \) has density at least \( 2^{-k} \). The formal argument follows.

For simplicity, we first present the proof under the assumption that \( S \) is a perfect (rather than just statistical) KC \( k(n) \) P-simulator for \( V \). The extension to the more general case follows thereafter.

Let \( p \) be the number of coin tosses of \( S \). We call a string \( c \) a \( t \)-th round prover prefix if \( c = \alpha_1, \beta_1, \ldots, \alpha_t, \beta_t \) for some \( l \) bit strings \( \alpha_1, \beta_1, \ldots, \alpha_t, \beta_t \). Throughout this proof, we do not make a real use of the auxiliary input and it is always set to the empty string. In the sequel, we omit the auxiliary from the notations and it should be clear that the empty string is always used. We define \( R_c \) to be the set of random tapes of the simulator that are consistent...
with a \( t \)-th round prover prefix \( c \). Namely,
\[
R_c \overset{\text{def}}{=} \{ r \in \{0, 1\}^{2^n} : S_t(x, r) = c \}
\]
(Recall that \( S_t(x, r) \) denotes the \( t \)-th round prefix of the conversation \( S(x, \lambda, r) \), cf. Definition 4.1.) So, for example: \( R_\lambda = \{0, 1\}^{2^n} \). Similarly, we define \( G_c \) to be the subset of \( R_c \) that contains only random strings from the good subspace \( SUCC_x \):
\[
G_c \overset{\text{def}}{=} R_c \cap SUCC_x
\]
Thus, \( G_\lambda = SUCC_x \) and if \( SUCC_x \) has density \( 2^{-k(n)} \) then \( G_\lambda / R_\lambda = 2^{-k(n)} \). Also, the summation over all possible \( t \)-prefixes yields \( \sum_c G_c = G_\lambda \) and \( \sum_c R_c = R_\lambda \).

We now analyze the interaction between \( P^* \) and \( V \) on input \( x \in I \). Let \( c \) be the \( t \)-th round prover prefix of this interaction. Recall that in the \( t+1 \)-st round \( P_s(x, c) \) picks \( r \) at random from \( R_c \), computes \( S(x, r) \), and outputs \( \beta_{t+1} \), where \( S(x, r) = (R, \alpha_1 \beta_1 \ldots \alpha_{t+1} \beta_{t+1} \ldots \alpha_g \beta_g) \) (cf. Definition 4.2). Let us denote by \( r_t \) the string \( r \) that was chosen by \( P_s \) in the \( t \)-th round. For \( t = 1, \ldots, g \) let us denote by \( GOOD_t \) the event that until round \( t \) only random tapes from \( SUCC_x \) were used. Namely, \( r_j \in SUCC_x \) for all \( j = 1, \ldots, t \). Fix \( GOOD_0 \) to be some event with probability 1.

We will show that for each round \( t = 0, \ldots, g-1 \) it is the case that
\[
\Pr[GOOD_{t+1} | GOOD_t] \geq 2^{-k}.
\]
(3)
It follows that \( \Pr[GOOD_g] \geq 2^{-kg} \geq n^{-d} \), where \( d \) is a constant such that \( g(n)k(n) \leq d \log n \).

We still have to argue that given \( GOOD_g \) the interaction \((P^*, V)\) ends with \( V \) accepting with high probability. Since \( S \) is, by assumption, a perfect KC \( k(n) \) \( P \)-simulator for the interaction \((P, V)\) we know that it outputs the distribution \( \text{View}(P, V)(x) \) when its random tape \( r \) is chosen uniformly from \( SUCC_x \) (cf. Definition 2.7). On the other hand, we know that \( P_s \) chooses each (possible) random tape with equal probability, the event \( GOOD_g \) being true implies that \( r_t \) is uniformly distributed in \( G_c = SUCC_x \cap R_c \) for all rounds \( t \). Therefore, the subspace of \( \text{View}(P, V)(x) \) obtained by conditioning on the event \( GOOD_g \) equals \( \text{View}(P, V)(x) \).

So the probability that \( V \) accepts in the interaction with \( P^* \) is at least \( \Pr[GOOD_g] \) times the probability that \( V \) accepts in the interaction with \( P \). Thus, the completeness of \((P, V)\) implies that the probability that \( V \) accepts in the interaction with \( P^* \) is \( \geq g(n)k(n) \), as desired.

It remains to justify Equation (3). By definition, if we sample the space of random tapes of the simulator once, we hit \( SUCC_x \) with probability at least \( 2^{-k} \). However, when we are sampling (uniformly) a subset of the possible random tapes of the simulator (the subset that is consistent with the conversation so far), it is not necessarily true that we hit \( SUCC_x \) with probability \( \geq 2^{-k} \). Still, Equation (3) can be derived by analyzing the prefix of the conversation (and thus the subset of the simulators possible coin tosses) as a stochastic variable.

Fix a round \( t \) and assume \( GOOD_t \). We will now give a lower bound on the probability of \( GOOD_{t+1} \). All probabilities in this proof are taken over the probability space defined by the interaction between the original verifer \( V \) with the simulation based prover \( P^* \). The summation in the following equation is over all possible \( t \)-th round verifier prefixes, \( c \in \{0, 1\}^{2^t} \), and over all possible verifier message in round \( t+1 \), \( a \in \{0, 1\} \).
\[
q \overset{\text{def}}{=} \Pr[GOOD_{t+1} | GOOD_t]
\]
\[ \begin{align*}
&= \sum_{c, \alpha} \Pr[c \circ \alpha \land \text{GOOD}_{t+1} | \text{GOOD}_t] \\
&= \sum_{c, \alpha} \Pr[\text{GOOD}_{t+1} | c \circ \alpha \land \text{GOOD}_t] \cdot \Pr[c \circ \alpha | \text{GOOD}_t] \\
&= \sum_{c, \alpha} \Pr[\text{GOOD}_{t+1} | c \circ \alpha] \cdot \Pr[c \circ \alpha | \text{GOOD}_t]
\end{align*} \]

The last equality is true since conditioned on the history \( c \circ \alpha \), the event \( \text{GOOD}_{t+1} \) does not depend on \( \text{GOOD}_t \). Next, we note that the event that \( V \) chooses his next message to be \( \alpha \) given the history \( c \) does not depend on the event \( \text{GOOD}_t \). Therefore, partitioning the above summation into a summation on the history so far \( c \) and a summation on the next verifier message \( \alpha \), we get:

\[ q = \sum_c \Pr[c | \text{GOOD}_t] \cdot \sum_{\alpha} \Pr[\alpha | c] \cdot \Pr[\text{GOOD}_{t+1} | c \circ \alpha] \]

Given that the simulator’s coins are picked uniformly in \( \text{SUCC}_x \), we know that the distribution output by the simulator is exactly equal to the distribution of the interaction \((P, V)\).\(^1\)

Thus, given \( \text{GOOD}_t \), we know that the \( t \)-prefixes of the interaction \((P^*, V)\) are distributed the same as the \( t \)-prefixes of the interaction \((P, V)\) and these are distributed equally to the \( t \)-prefixes output by the simulator given that we pick the simulator coins uniformly in \( \text{SUCC}_x \). Therefore, we may write the probability that the prefix \( c \) appears in the interaction \((P^*, V)\) in this case as \( |G_c|/|G_\lambda| \). Namely, the probability of the prefix \( c \) given \( \text{GOOD}_t \) is equal to the number of simulator random tapes in \( \text{SUCC}_x \) that agree with this prefix, divided by the number of random tapes in \( \text{SUCC}_x \). Also, we may replace the probability that \( V \) responds with the string \( \alpha \) to the history \( c \) by \( |G_{c\circ \alpha}|/|G_c| \). Last, we note that the probability of \( \text{GOOD}_{t+1} \) given the history \( c \circ \alpha \) is exactly \( |G_{c\circ \alpha}|/|R_{c\circ \alpha}| \) since \( P^* \) chooses \( r_{t+1} \) randomly in \( R_{c\circ \alpha} \). To summarize, we may write:

\[ q = \sum_{c} \frac{|G_c|}{|G_\lambda|} \cdot \sum_{\alpha} \frac{|G_{c\circ \alpha}|}{|G_c|} \cdot \frac{|G_{c\circ \alpha}|}{|R_{c\circ \alpha}|} \]

\[ = \frac{1}{|G_\lambda|} \cdot \sum_{c} \frac{|G_{c\circ \alpha}|^2}{|R_{c\circ \alpha}|} \]

(5)

By the Cauchi-Schwartz inequality we know that for positive \( x_i, y_i, 1 \leq i \leq n \), it holds that

\[ \left( \sum_{i=1}^{n} \frac{x_i^2}{y_i} \right) \cdot \left( \sum_{i=1}^{n} y_i \right) \geq \left( \sum_{i=1}^{n} x_i \right)^2 \]

Using this with \( x_i = |G_{c\circ \alpha}| \) and \( y_i = |R_{c\circ \alpha}| \), in Equation (5), we get:

\[ q \geq \frac{1}{|G_\lambda|} \cdot \frac{(\sum_{c \circ \alpha} |G_{c\circ \alpha}|)^2}{\sum_{c \circ \alpha} |R_{c\circ \alpha}|} \]

\[ = \frac{1}{|G_\lambda|} \cdot \frac{|G_\lambda|^2}{|R_\lambda|} \]

(Recall that we are now dealing with the more restricted case of perfect knowledge complexity.)

\[ = \frac{|G_\lambda|}{|R_\lambda|} \geq 2^{-k} \]

---

\(^1\)Recall that we are now dealing with the more restricted case of perfect knowledge complexity.
Thus, we get that $\Pr[\text{GOOD}_{t+1} | \text{GOOD}_t] \geq 2^{-k}$ as claimed and we are done with the proof of Lemma 6.3.

6.3.1 The case of statistical knowledge complexity

The case of statistical knowledge complexity is a little more involved. Again we are going to show two things. First, that $\text{GOOD}_g$ has a large enough probability (Claim 6.6 below), and second that given $\text{GOOD}_g$, $V$ accepts with high probability (Claim 6.5 below). Our proof extends the proof for the case of perfect knowledge complexity and in fact, we only show that the difference between the perfect case and the statistical case is small enough.

Recall that in the statistical case it is only guaranteed that the view of $V$ in the interaction $(P, V)$ is statistically close to the output distribution of the simulator when its random tape is picked uniformly in the good subspace $\text{SUCC}_x$.

We are going to define a new prover $P_G^*$ and a new verifier $V_G^*$ who will help us analyse the protocol $(P^*, V)$ which interests us. The prover $P_G^*$ and the verifier $V_G^*$ are fictional entities which do not really appear in any of the interactions. Also, the verifier $V_G^*$ will not necessarily be implementable as a probabilistic polynomial time machine. We define $P_G^*$ to be the simulation based prover that uses only random tapes from $\text{SUCC}_x$. Recall that the simulation based prover on a given history so far $c$, picks uniformly a random tape $r \in R_c$ and outputs the next message of the prover in the interaction $S(x, \lambda, r)$. The prover $P_G^*$ will pick uniformly $r \in G_c$ instead. Thus, if we were considering perfect simulation, we would get that the prover $P_G^*$ acts exactly like the original prover $P$. However, since the simulation is not perfect, we only get some closeness between $P_G^*$ and $P$ which we have to analyse. The verifier $V_G^*$ is defined analogously. Namely, in response to a history so far $c$, it picks uniformly in $G_c$ and answers according to the next verifier step in the conversation $S(x, \lambda, r)$. Note again that $V_G^*$ does not act exactly like $V$ only because the simulation is not perfect.

Consider now the interaction between $(P_G^*, V_G^*)$. This is exactly equal to the output distribution of the simulator given $\text{SUCC}_x$. Two facts follow from this observation. First, we get that the simulator $S$ is a perfect KC $k(n)$ P-simulator for the interaction $(P_G^*, V_G^*)$. Second, by the property of the good subspace $\text{SUCC}_x$ we get that, the interaction $(P_G^*, V_G^*)$ is statistically close to the original interaction $(P, V)$. It can be proven by induction that the interaction between $P_G^*$ to $V$ is statistically close both to the interaction $(P, V)$ and to the interaction $(P_G^*, V_G^*)$. In general, if an interaction $(A, B)$ that has $g$ rounds is $\delta$-close to an interaction $(A', B')$ then it also hold that the interaction $(A', B)$ is $g \cdot \delta$-close to both. (This can be formally proven by a simple induction).

From the fact that $(P_G^*, V_G^*)$ is statistically close to $(P, V)$ we get that $(P_G^*, V_G^*)$ accepts $x$ with probability almost 1. Since the simulator $S$ is a perfect simulator for the interaction $(P_G^*, V_G^*)$, we can use our proof for the perfect case and get that $P^*$ convinces $V_G^*$ with probability at least $n^{-\epsilon}$ for some constant $\epsilon > 0$. However, we are interested in the probability that $P^*$ convinces $V$ on $x$.

Let us first show that given $\text{GOOD}_g$, $V$ accepts $x$ with high probability.

Claim 6.5 Given $\text{GOOD}_g$, the probability that $V$ accepts on input $x \in L$ while talking to $P^*$, is $\geq 1 - \epsilon$ for some negligible fraction $\epsilon$.

Proof: Conditioned on the event $\text{GOOD}_g$, we get that $P^*$ is behaving exactly like $P_G^*$. Now,
since \((P^*_G, V)\) is statistically close to \((P, V)\) and our claim follows.

Next, we show that the probability of the event \(\text{GOOD}_G\) in the interaction \((P^*, V)\) is bigger than some polynomial fraction. Here, we are going to show that the difference between the perfect case and the statistical case is small in this respect.

**Claim 6.6** Let \(x \in L\) and consider the probability space of the interaction \((P^*, V)\) on the input \(x\). Then,

\[
\Pr[\text{GOOD}_G] \geq n^{-O(1)}
\]

**Proof:** We prove that for every \(t, 0 \leq t \leq g - 1,\)

\[
\Pr[\text{GOOD}_{t+1} | \text{GOOD}_t] \geq 2^{-k} - \epsilon_1
\]

for some negligible fraction \(\epsilon_1\). Thus we get that

\[
\Pr[\text{GOOD}_G] = \prod_{t=0}^{g-1} \Pr[\text{GOOD}_{t+1} | \text{GOOD}_t] \geq 2^{-kg} - \epsilon
\]

for some negligible fraction \(\epsilon\), and we are done.

So let us show that Equation (6) holds. From now and on we are going to consider now two probability spaces: the probability space induced by the interaction \((P^*, V)\) and the probability space induced by the interaction \((P^*, V^*_G)\). Therefore, we shall explicitly write for each probability which space is considered.

Since \(S\) is a perfect simulator for the interaction \((P^*, V^*_G)\), our proof for the perfect case implies

\[
\Pr_{(P^*, V^*_G)}[\text{GOOD}_{t+1} | \text{GOOD}_t] \geq 2^{-k}. \tag{7}
\]

We are going to show that when \(V\) participates in the interaction instead of \(V^*_G\), this property does not change significantly. Specifically, we will show that

\[
\epsilon_1 \overset{\text{def}}{=} \Pr_{(P^*, V^*_G)}[\text{GOOD}_{t+1} | \text{GOOD}_t] - \Pr_{(P^*, V)}[\text{GOOD}_{t+1} | \text{GOOD}_t] \tag{8}
\]

is a negligible fraction. This is enough since Equation (6) follows from Equation (7) and Equation (8).

Again, we sum over all \(t\)-round prefixes \(c \in \{0, 1\}^{t-2l}\) and over all possible verifier responses \(\alpha \in \{0, 1\}^l:\)

\[
\epsilon_1 = \sum_{c, \alpha} \Pr_{(P^*, V^*_G)}[\text{GOOD}_{t+1} | c \circ \alpha] \cdot \Pr_{(P^*, V)}[c \circ \alpha | \text{GOOD}_t] - \Pr_{(P^*, V)}[c \circ \alpha | \text{GOOD}_t]
\]

Since the probability of the event \(\text{GOOD}_{t+1}\) conditioned on the prefix \(c \circ \alpha\) depends only on the behavior of the prover \(P^*\) which is identical in both distributions, we may write:

\[
\epsilon_1 = \sum_{c, \alpha} \Pr_{(P^*, V^*_G)}[c \circ \alpha] \cdot \left( \Pr_{(P^*, V)}[c \circ \alpha | \text{GOOD}_t] - \Pr_{(P^*, V)}[c \circ \alpha | \text{GOOD}_t] \right)
\]

\[
\leq \sum_{c, \alpha} \Pr_{(P^*, V^*_G)}[c \circ \alpha] - \Pr_{(P^*, V)}[c \circ \alpha]
\]

\[
= \sum_{c, \alpha} \Pr_{(P^*_G, V)}[c \circ \alpha] - \Pr_{(P^*_G, V)}[c \circ \alpha] \tag{9}
\]
The last equality holds since conditioning on \( \text{GOOD}_t \) in all relevant prover steps is equivalent to the prover being \( P^*_G \). Finally, since the interaction \((P^*_G, V)\) is statistically close to \((P^*_G, V^*_G)\), we get that the expression in Equation (9) is negligible and we are done. ■

7 Conclusions and Open Problems

We showed that if \( L \) has a statistical ZK proof then it has a statistical ZK proof with a prover who runs in PPT with an NP oracle. This was previously only known given complexity assumptions. Our first question is whether one can remove assumptions from other similar problems. In particular, can one unconditionally establish any of the following?

1. If \( L \) has a statistical ZK proof then it has a statistical ZK proof with perfect completeness (i.e. the verifier accepts with probability 1 when \( x \in L \); cf. [14])

2. If \( L \) has a statistical ZK proof then it has a statistical ZK proof with a black-box simulator.

3. If \( L \) has an interactive proof which is statistical ZK with respect to the honest verifier, then it has a statistical ZK interactive proof.

We recall that these results are known given complexity assumptions [5].

Second, on the subject of the power of the prover. Our bound of PPT with NP oracle does not depend on the complexity of the language. Can one find “tighter” relationships between the complexity of \( L \) and the complexity of a statistical ZK prover for \( L \)? For example, what can one say about statistical ZK in the model of [4] where the prover is PPT with an oracle for \( L \)?

Acknowledgments

We thank Oded Goldreich and Avi Wigderson for helpful remarks and proof simplifications.

References


