# From Logarithmic advice to Single-bit Advice

Oded Goldreich, Madhu Sudan, and Luca Trevisan

**Abstract.** Building on Barak's work (*Random'02*), Fortnow and Santhanam (*FOCS'04*) proved a time hierarchy for probabilistic machines with one bit of advice. Their argument is based on an implicit translation technique, which allow to translate separation results for short (say logarithmic) advice (as shown by Barak) into separations for a single-bit advice. In this note, we make this technique explicit, by introducing an adequate translation lemma.

**Keywords:** Machines that take advice, separations among complexity classes.

An early version of this work appeared as TR04-093 of *ECCC*.

## 1 Introduction and High Level Description

Trying to address the open problem of providing a probabilistic time hierarchy, Barak [1] presented a time hierarchy for slightly non-uniform probabilistic machines. Specifically, he showed that, in presence of double-logarithmic advice, there exists a hierarchy of probabilistic polynomial-time. Subsequently, Fortnow and Santhanam [2] showed that a similar hierarchy holds in the presence of a single-bit advice. Their argument is based on an implicit translation technique, which allow to translate separation results for short (say logarithmic) advice into separations for a single-bit advice. In this note, we make this technique explicit, by introducing an adequate translation lemma and showing that applying it to Barak's result [1] yields the aforementioned result of [2].

Interestingly (as in [2]), we rely on the fact that Barak [1] actually shows a time separation that holds even when the more time-restricted machine is given a somewhat longer advice. In contrast, arguably, the more natural statement of such results refers to machines that use the same advice length.[1]

The basic idea underlying the proof in [2] is that short advice can be incorporated in the (length of the) instance of a padded set, while using a single bit of advice to indicate whether or not the resulting instance length encodes a valid advice. For this to work, the length of the resulting instance should indicate a unique length of the original instance as well as a value of a corresponding advice (for this instance length).

[1] That is, in order to show, say, that BPtime($n^3$)/1 is not contained in BPtime($n^2$)/1, we use the fact that Barak showed that BPtime($n^6$)/ $\log n$ is not contained in BPtime($n^4$)/$2\log n$ (rather than that BPtime($n^6$)/ $\log n$ is not contained in BPtime($n^4$)/ $\log n$).

Suppose we wish to treat a set $S$ that is decidable (within some time bound) using *eight* bits of advice. Viewing the possible values of the advice as integers in $\{0,1,...,255\}$, we define a (padded) set $S'$ as follows: the string $x0^{255|x|+i}$ is in $S'$ if and only if $x \in S$ and $i$ is an adequate advice for instances of length $|x|$. Note that $S'$ can be decided using a single bit of advice that indicates whether the instance length encodes a valid advice for $S$. Specifically, the advice bit for length $m$ instances (of $S'$) is 1 if and only if $m$ mod 256 is a valid advice for instances of length $\lfloor m/256 \rfloor$ (of $S$). Thus, on input $y = x0^{255|x|+i}$, where $i \in \{0,...,255\}$, we accept if and only if the advice bit is 1 and the original machine accepts $x$ when given advice $i$.

Note that we should also show that if $S$ is undecidable using less time (and, say, *nine* bits of advice), then $S'$ is correspondingly hard (even using a single bit of advice). This is shown by using a machine for deciding $S'$ as a subroutine for deciding $S$, while using part of the advice (given for deciding $S$) for determining an adequate instance for $S'$. In other words, we present a non-uniform reduction of $S$ to $S'$, where the non-uniformity is accounted for by the longer advice allowed in deciding $S$.

## 2 Preliminaries

We consider advice-taking probabilistic machines, denoting by $M(a,x)$ the output distribution of machine $M$ on input $x$ and advice $a$. We denote by $\mathrm{BPtime}(T)/A$ the class of sets decidable by advice-taking probabilistic machines of time complexity $T$ and advice complexity $A$. That is, $S \in \mathrm{BPtime}(T)/A$ if there exists a probabilistic machine $M$ and a sequence of strings $(a_n)_{n\in\mathbb{N}}$ such that the following conditions hold:

1. For every $n \in \mathbb{N}$, it holds that $|a_n| = A(n)$.
2. For every $x \in \{0,1\}^*$, on input $x$ and advice $a_{|x|}$, machine $M$ makes at most $T(|x|)$ steps.
3. For every $x \in \{0,1\}^*$, it holds that $\Pr[M(a_{|x|},x) = \chi_S(x)] \geq 2/3$, where $\chi_S(x) = 1$ if $x \in S$ and $\chi_S(x) = 0$ otherwise.

We assume that the machine model supports some trivial computations with little overhead. Specifically, we refer to computing the square root of the length of the input in linear time. Our results hold with minor modifications in case the machine model is less flexible (e.g., if computing the square root of the length of the input requires quadratic time).

To simplify the presentation, we will associate binary strings with the integers that they represents. That is, the $\ell$-bit long binary string $\sigma_{\ell-1}\cdots\sigma_0$ will be associated with the integer $\sum_{j=0}^{\ell-1}\sigma_j \cdot 2^j$. Thus, when writing $0^{\sigma_{\ell-1}\cdots\sigma_0}$, we mean a binary string consisting of $\sum_{j=0}^{\ell-1}\sigma_j \cdot 2^j$ zeros.

## 3 Detailed Technical Presentation

We state our translation lemma for probabilistic machines, and note that an analogous lemma holds for deterministic (and non-deterministic) machines.

**Lemma 1** (Translation Lemma): *Suppose that $S$ is a set that is decided by some advice-taking probabilistic machine $M$ in time $T_M(n)$ using $A_M(n) \leq \lfloor \log_2 n \rfloor$ bits of advice, where $n$ denotes the length of the instance of $S$. Suppose further that $S$ is not decided by any $a(n)$-advice probabilistic machine in time $t(n)$, where $a(n) \geq A_M(n)$. Then, there exists a set $S' = S'_M$ that is decided in probabilistic time $T'$ using a single bit of advice, where $T'(m) = T_M(\lfloor\sqrt{m}\rfloor) + m$, but is not decidable by any $(a(\lfloor\sqrt{m}\rfloor) - A_M(\lfloor\sqrt{m}\rfloor))$-advice probabilistic machine in time $t(\lfloor\sqrt{m}\rfloor) - m$, where $m$ denotes the length of the instance of $S'$.*

Needless to say, the lemma can be generalized to handle $A_M(n) = O(\log n)$, in which case $\lfloor\sqrt{m}\rfloor$ should be replaced by $m^{1/O(1)}$.

### 3.1 Using the Translation Lemma

Before proving the Translation Lemma, let us spell-out its main implication.

**Corollary 2** (reducing non-uniformity in BPtime separations): *Let $T, A, t, a : \mathbb{N} \to \mathbb{N}$ such that $a(n) \geq A(n)$. If $\mathrm{BPtime}(T)/A$ contains sets not in $\mathrm{BPtime}(t)/a$, then $\mathrm{BPtime}(T')/1$ contains sets not in $\mathrm{BPtime}(t')/a'$, where $T'(m) \stackrel{\text{def}}{=} T(\lfloor\sqrt{m}\rfloor) + m$, $t'(m) \stackrel{\text{def}}{=} t(\lfloor\sqrt{m}\rfloor) - m$ and $a'(m) \stackrel{\text{def}}{=} a(\lfloor\sqrt{m}\rfloor) - A(\lfloor\sqrt{m}\rfloor)$.*

For example, we can apply Corollary 2 to Barak's result [1] that asserts the existence of a set $S$ in, say, $(\mathrm{BPtime}(n^6)/\log\log n) \setminus (\mathrm{BPtime}(n^4)/\log n)$. Doing so, we conclude that there exists a set in $(\mathrm{BPtime}(m^3)/1) \setminus (\mathrm{BPtime}(m^2)/(0.5\log m - \log\log m))$, which in particular implies $\mathrm{BPtime}(m^2)/1 \subset \mathrm{BPtime}(m^3)/1$. Thus, we can translate Barak's separations, which refer to probabilistic machines with logarithmic advice, into separations that refer to probabilistic machines with a single bit of advice, as established by Fortnow and Santhanam [2]. (This consequence is not surprising, because the Translation Lemma makes explicit the ideas in [2].)

Note that in order to obtain an interesting consequence out of Corollary 2, we need $a(n) \geq A(n)+1$. In contrast, using $a(n) = A(n)$ implies that $\mathrm{BPtime}(T')/1$ contains sets not in $\mathrm{BPtime}(t')$, which holds regardless of the hypothesis and for any choice of $T' > 0$ and $t'$ (even for $t' \gg T'$).

### 3.2 Proving the Translation Lemma

Recall that $M$ decides $S$ in time $T_M$, using advice of length $A_M$, where $A_M(n) \leq \lfloor \log_2 n \rfloor$. Fixing a sequence of advice strings $(a_n)_{a\in\mathbb{N}}$ for machine $M$, we define $S'$ depending on this sequence. Specifically,

$$S' \stackrel{\text{def}}{=} \{x0^{(|x|-1)|x|+a_{|x|}} : x \in S\}. \tag{1}$$

That is, $y = x0^{(|x|-1)|x|+i} \in S'$ if and only if it holds that $x \in S$ and $a_{|x|} = i$. Observe that $|x0^{(|x|-1)|x|+i}| = |x|^2 + i$ and that, for every $m \in \{n^2 + 0, ..., n^2 + 2^{A_M(n)} - 1\}$ (which in turn is contained in $\{n^2, ..., (n+1)^2 - 1\}$), it holds that

$\lfloor\sqrt{m}\rfloor = n$. In what follows, $n$ (resp., $m$) will always denote the length of instances to $S$ (resp., $S'$).

We first show that $S'$ is decidable by a probabilistic machine $M'$ taking one bit of advice and running in time $T_M(\lfloor\sqrt{m}\rfloor) + m$. Machine $M'$ checks whether its input $y \in \{0,1\}^m$ has the form $x0^{(n-1)n+i}$, where $|x| = n = \lfloor\sqrt{m}\rfloor$ and $i < n$, and otherwise rejects $y$ up-front. Given the advice bit $\sigma_m$, machine $M'$ always rejects if $\sigma_m = 0$ and invokes $M$ on input $x$ and advice $i$ (viewed as an $A_M(n)$-bit long string) otherwise. Thus, $M'$ accepts $y = x0^{(|x|+1)|x|+i}$ using advice $\sigma_m$ if and only if $\sigma_m = 1$ and $M$ accepts $x$ using advice $i$. The advice (bit) $\sigma_m$ regarding $m$-bit inputs is determined in correspondence to the aforementioned parsing: the advice bit is 1 if and only if $m = \lfloor\sqrt{m}\rfloor^2 + a_{\lfloor\sqrt{m}\rfloor}$. Indeed, this setting of the advice $\sigma_m$ guarantees that $M'$ accepts $y = x0^{(|x|-1)|x|+i}$ if and only if $x \in S$ and $i = a_{|x|}$. Thus, using adequate advice, $M'$ decides $S'$. Indeed, as required, the running time of $M'$ is $m + T_M(\lfloor\sqrt{m}\rfloor)$, where $m$ steps are used to parse $y$ (into $x$ and $i$) and $T_M(|x|)$ steps are used to emulate $M(i,x)$.

We next show that $S'$ is not decidable by any probabilistic machine that runs in time $t(\lfloor\sqrt{m}\rfloor) - m$ and takes a $(a(\lfloor\sqrt{m}\rfloor) - A_M(\lfloor\sqrt{m}\rfloor))$-bit long advice. Actually, for any monotonically non-decreasing functions $t'$ and $a'$, we will show that if $S'$ is decidable by some probabilistic machine that runs in time $t'(m)$ and takes $a'(m)$ bits of advice, then $S$ is decidable by a probabilistic machine that runs in time $t''(n) = t'(n^2+n) + n^2$ and takes $a''(n) = A_M(n) + a'(n^2+n)$ bits of advice.[2] Suppose that $M'$ is a machine deciding $S'$ as in the hypothesis, and let $\mathtt{adv}_{M'}(m)$ be the advice it uses for $m$-bit inputs. Then consider the following machine $M''$ (designed to decide $S$) whose advice on inputs of length $n$ is the pair $a''_n = (a_n, \mathtt{adv}_{M'}(n^2 + a_n))$. On input $x$ and advice $(i,j)$, machine $M''$ invokes $M'$ on input $x0^{(|x|-1)|x|+i}$ with advice $j$. Thus, $M''$ accepts $x$ when given the (adequate) advice $a''_{|x|}$ if and only if $M'$ accepts $x0^{(|x|-1)|x|+a_{|x|}}$ when given the advice $\mathtt{adv}_{M'}(|x|^2 + a_{|x|})$. It follows that $M''$ decides $S$, and does so within the stated complexities. ■

*Digest:* We defined $S'$ based not only on $S$ but rather based on an adequate advice sequence $(a_n)_{n\in\mathbb{N}}$ that vouches that $S \in \mathrm{BPtime}(T)/A$ (via a machine $M$). Once $S'$ is defined, the proof proceeds in two steps:

1. Relying on the hypothesis that $M$ decides $S$ in time $T$ using advice of length $A$, we establish that $S' \in \mathrm{BPtime}(T')/1$, where $T'(m) = T(\lfloor\sqrt{m}\rfloor) + m$.

   The advice-bit for $S'$ is used in order to facilitate the partition of the instances of $S'$ into two sets: a set of instances $x0^{(|x|-1)|x|+i}$ that satisfy $i = a_{|x|}$, and a set of instances that do not satisfy this condition. Machine $M$ is invoked only for instances of the first type, and instances of the second type are rejected up-front.

[2] Indeed, suppose that $t'(m) = t(\lfloor\sqrt{m}\rfloor) - m$ and $a'(m) = a(\lfloor\sqrt{m}\rfloor) - A_M(\lfloor\sqrt{m}\rfloor)$, then $t''(n) = t'(n^2+n) + n^2 = (t(\lfloor\sqrt{n^2+n}\rfloor) - (n^2+n)) + n^2 < t(n)$ and $a''(n) = A_M(n) + a'(n^2+n) = A_M(n) + (a(n) - A_M(n)) = a(n)$, in contradiction to the lemma's hypothesis.

2. Assuming that $S' \in \mathrm{BPtime}(t')/a'$, we establish that $S \in \mathrm{BPtime}(t)/a$, where $t(n) = t'(n^2+n)+n^2$ and $a(n) = A(n) + a'(n^2+n)$.
   This is done by "reducing" the problem of "deciding $S$ with $a(n)$ bits of advice" to the problem of "deciding $S'$ with $a'(m)$ bits of advice", while the reduction itself uses $A(n) = a(n) - a'(m)$ bits of advice.

## Subsequent work

We mention a subsequent related work by van Melkebeek and Pervyshev [3], which provides a direct proof of a more general result. We still feel that there is interest in the approach taken in the current work (i.e., the translation lemma and its proof).

## References

1. B. Barak. A Probabilistic-Time Hierarchy Theorem for "Slightly Non-uniform" Algorithms. In *Random'02*, LNCS 2483, pages 194–208, 2002.
2. L. Fortnow and R. Santhanam. Hierarchy theorems for probabilistic polynomial time. In *45th FOCS*, pages 316–324, 2004.
3. D. van Melkebeek and K. Pervyshev. A Generic Time Hierarchy for Semantic Models with One Bit of Advice. *Computational Complexity*, Vol. 16, pages 139–179, 2007.