1 An alternative definition of BPP

Similar to the verification characterization of NP, we also have an alternative definition of BPP, using deterministic TMs.

**Definition 1** (BPP, alternative). *A language L is in BPP if there exists a polynomial-time TM M and a polynomial p(·), such that for every x ∈ {0, 1}*, Pr_r[M(x, r) = L(x)] ≥ 3/4, where r is drawn uniformly at random from all strings of length p(|x|).*

In other words, for $L \in \text{BPP}$, there is a TM such that, if $x \in L$, then at least $3/4$ fraction of all certificates is valid, and if $x \notin L$, then at most $1/4$ fraction is. Recall that if $L \in \text{NP}$, then there is a TM such that $x \in L$ if and only if there is at least one certificate.

The equivalence between Definition 1 and the previous definition using PTM can be shown in the same way as the equivalence between two definitions of NP. The standard amplification method (using Chernoff bound) also applies to this definition. We can define RP and coRP similarly in this way as well. An immediate consequence of such definitions is that $\text{RP} \subseteq \text{NP}$ and $\text{coRP} \subseteq \text{coNP}$.

2 BPP in $\text{P/poly}$

It is obvious that $\text{RP} \subseteq \text{NP}$ — RP requires at least $3/4$ fraction of certificates are valid, whereas NP merely requires at least one. However, what about BPP? An easy upper bound is that $\text{BPP} \subseteq \text{PSPACE}$, since we can use the polynomial space to go through all certificates. It is not a priori obvious that BPP is in any class below it.

Our first upper bound is that BPP can be simulated by the circuit model, shown by Adleman [Adl78].

**Theorem 1.** $\text{BPP} \subseteq \text{P/poly}$. 

**Proof.** Let $L \in \text{BPP}$ be a language. Because of the amplification of BPP, there is a TM $M$ such that $\Pr_x[M(x, r) \neq L(x)] \leq 2^{-n-1}$, where $x$ is an input, $n = |x|$, $r \in \{0, 1\}^m$ is the random choices of length $m$, bounded by some polynomial in $n$. For every $x$, there are at most $2^m \cdot 2^{-n-1}$ random choices that lead to a wrong output. Hence, the total amount of such “bad” strings over all possible inputs of length $n$ is at most $2^n \cdot 2^m \cdot 2^{-n-1} = 2^{m-1}$. There are $2^m$ choices of $r$ in total, meaning that there exists at least one string that is correct for all input $x$ of length $n$. Let $M'$ be a TM with such a “universally good” $r$ as the advice. Then $M'$ is correct on all inputs, implying that $L \in \text{P/poly}$. 

In particular, Theorem 1, together with the Karp-Lipton theorem, implies that if SAT has a BPP algorithm, then PH collapses. This is an evidence against solving SAT efficiently by randomized algorithms.

3 BPP in $\Sigma^P_2$

Another upper bound for BPP is that it is inside the polynomial hierarchy, shown by Sipser [Sip83] and Lautemann [Lau83].

Theorem 2. $\text{BPP} \subseteq \Sigma^P_2$.

For a language $L \in \text{BPP}$ and the corresponding $M$ in Definition 1, with proper amplification, we have that

$$x \in L \Rightarrow \Pr_r[M(x, r) = 1] \geq 1 - 2^{-n};$$

$$x \notin L \Rightarrow \Pr_r[M(x, r) = 1] \leq 2^{-n},$$

where $r$ is a uniformly at random string in $\{0, 1\}^m$ for $m > n$ (because we can always append some useless random bits) and $m$ is bounded by a polynomial in $n$. Hence, there is a huge gap between the two cases in terms of the fraction of certificates (almost-all vs. almost-none). One way to distinguish the two cases is to use “bit masks”. For two vectors $x$ and $y$, let $x \oplus y$ be the bitwise XOR, and for a set $S$, $S \oplus x = \{x \oplus y \mid y \in S\}$. Let $k := \lceil \frac{m}{n} \rceil + 1$ so that $\frac{m}{n} < k < \frac{m}{n} + 2$.

Lemma 3. Let $n$ be a sufficiently large integer, and $m$ an integer bounded by a polynomial in $n$. For every set $S \subseteq \{0, 1\}^m$ with $|S| \leq 2^{m-n}$ and every $k$ vectors $u_1, \ldots, u_k$, $\bigcup_{i=1}^k (S \oplus u_i) \neq \{0, 1\}^m$.

Proof. Obviously, $|S| = |S \oplus u_i|$. Hence, $|\bigcup_{i=1}^k (S \oplus u_i)| \leq k |S| \leq (\frac{m}{n} + 2)2^{m-n} < \frac{m+2n}{n^2} \cdot 2^m$. Since $m$ is bounded by a polynomial in $n$, $|\bigcup_{i=1}^k (S \oplus u_i)| < 2^m$ as long as $n$ is sufficiently large. \hfill \Box

What Lemma 3 is saying is that in the “no” case, even if we shift the set of certificates $k$ times, it cannot cover the whole space. In contrast, we have the following.

Lemma 4. Let $n$ be a sufficiently large integer, and $m$ an integer bounded by a polynomial in $n$. If $S \subseteq \{0, 1\}^m$ with $|S| \geq (1 - 2^{-n}) 2^m$, then there exist $k$ vectors $u_1, \ldots, u_k$ such that $\bigcup_{i=1}^k (S \oplus u_i) = \{0, 1\}^m$.

Proof. We draw $k$ vectors uniformly at random from $\{0, 1\}^m$ as the bit masks $u_1, \ldots, u_k$. To show the lemma, we need to show that with strictly positive probability, $\bigcup_{i=1}^k (S \oplus u_i) = \{0, 1\}^m$. For each $r \in \{0, 1\}^m$, let $B_r$ be the “bad event” that $r \notin \bigcup_{i=1}^k (S \oplus u_i)$. Then what we want to show is that $\Pr \left[ \bigwedge_{r \in \{0, 1\}^m} \neg B_r \right] > 0$, or equivalently, $\Pr \left[ \bigvee_{r \in \{0, 1\}^m} B_r \right] < 1$.  

2
By the union bound,
\[
\Pr \left[ \bigvee_{r \in \{0,1\}^m} B_r \right] \leq \sum_{r \in \{0,1\}^m} \Pr[B_r].
\]

For an individual \( r \) and any \( i \in [k] \), \( \Pr[r \not\in (S \oplus u_i)] = \Pr[r \oplus u_i \not\in S] \leq 2^{-n} \). This is because that since \( u_i \) is uniformly at random, \( r \oplus u_i \) is also uniformly at random. Since \( u_i \)'s are mutually independent and \( k > \frac{m}{n} \),
\[
\Pr[B_r] = \Pr \left[ \bigwedge_{1 \leq i \leq k} r \not\in (S \oplus u_i) \right] = \prod_{i=1}^{k} \Pr[r \not\in (S \oplus u_i)] \leq 2^{-nk} < 2^{-m}.
\]

Hence, \( \Pr \left[ \bigvee_{r \in \{0,1\}^m} B_r \right] < 2^m \cdot 2^{-m} = 1 \), which is what we want to show.

The proof of Lemma 4 is called the probabilistic method, introduced by Paul Erdős. The usual setup is that, in order to show the existence of some object, we show that drawing a random object from a larger space, the probability of the desired one is strictly positive. For more of its use, see the fantastic book by Alon and Spencer [AS16].

**Proof of Theorem 2.** Let \( L \) be a language in \( \text{BPP} \) and \( M \) the corresponding TM as in Definition 1, with proper amplification. Lemma 3 and Lemma 4 imply that \( x \in L \) if and only if
\[
\exists u_1, \ldots, u_k \in \{0,1\}^m, \quad \forall r \in \{0,1\}^m, \quad r \in \bigcup_{i=1}^{k} (S \oplus u_i).
\]

This is equivalent to
\[
\exists u_1, \ldots, u_k \in \{0,1\}^m, \quad \forall r \in \{0,1\}^m, \quad \bigvee_{i=1}^{k} M(x, r \oplus u_i) = 1.
\]

Recall that \( k = \left\lceil \frac{m}{n} \right\rceil + 1 \). Thus \( \bigvee_{i=1}^{k} M(x, r \oplus u_i) \) can be simulated by a single TM with only polynomial overhead in its running time, so this is a \( \Sigma_2^p \) statement and \( L \in \Sigma_2^p \).

It is easy to see that \( \text{BPP} \) is closed under complement (\( \text{BPP} = \text{coBPP} \)). Namely, if \( L \in \text{BPP} \), then \( \overline{L} \in \text{BPP} \) as well (why?). Hence, Theorem 2 implies that \( \text{BPP} \subseteq \Sigma_2^p \cap \Pi_2^p \), which is “barely” above \( \text{NP} \).

**Remark (Bibliographic).** Sipser [Sip83] showed that \( \text{BPP} \subseteq \text{PH} \), and Lautemann [Lau83] improved it to \( \text{BPP} \subseteq \Sigma_2^p \cap \Pi_2^p \). Relevant chapters are [AB09, Chapter 7.5] and [Pap94, Chapter 11].
References


