Two Theorems about BPP

Adleman's Theorem: BPP $\subseteq$ P/poly

Proof: Let L be a set in BPP. Recall that the Chernoff bounds on the tails of the binomial distribution ensure that there is a probabilistic polynomial-time machine M such that $M(x) = L(x)$ with probability at least $1 - 2^{-(n+1)}$. Let m be the maximum number of random bits that M uses on inputs of length n. So $m = \text{poly}(n)$, and M's output on input x is a function of x and a random string $r \in \{0,1\}^m$; this function of x and r is computable in deterministic polynomial time.

Fix a length n, and consider all inputs $x \in \{0,1\}^n$. We say that *r is bad for x* if M outputs the wrong answer on input x and random string r; otherwise, *r is good for x*. Because M's error probability is at most $2^{-(n+1)}$, the number of r's that are bad for any given x is at most $(2^m)/(2^{n+1})$. The total number of r's that are bad for *at least one* x is thus at most $(2^n) \cdot ((2^m)/(2^{(n+1)})) = 2^{m-1}$. (This maximum would be achieved if the set of r's that are bad for x_1 were disjoint from the set of r's that are bad for x_2, for all $x_1 \neq x_2$.) This means that there are $2^m - 2^{m-1} > 0$ strings r that are good for all $x \in \{0,1\}^n$.

Let r_n be a random string that is good for all $x \in \{0,1\}^n$. The circuit C_n that accepts elements of $L \cap \{0,1\}^n$ is "M on inputs of length n, with r_n hardcoded in," i.e., one that computes precisely the function that M computes on inputs of length n when it uses the random string r_n. The proof of Theorem 6.6 (P $\subseteq$ P/poly) shows that $\{C_n\}_{n \geq 1}$ is a polynomial-sized circuit family. $\square$

The Sipser-Gacs Theorem: BPP $\subseteq \Sigma_2^P \cap \Pi_2^P$

Proof: Because BPP is closed under complement, it suffices to show that BPP $\subseteq \Sigma_2^P$. Let L be a language in BPP and M be a machine that accepts L and has error probability at most 2^{-n}. Let $m = \text{poly}(n)$ be the length of the random strings that M uses on inputs $x \in \{0,1\}^n$. We denote by $M(x,r)$ the output of M on input x when M uses random string r.

For $x \in \{0,1\}^n$, let S_x be the set of strings $r \in \{0,1\}^m$ such that $M(x,r) = 1$. If r is chosen uniformly at random from $\{0,1\}^m$, then r is in S_x with probability at most 2^{-n} if $x \notin L$, and r is in S_x with probability at least $1 - 2^{-n}$ if $x \in L$.

Let $k = m/n + 1$, and consider a set $U = \{u_1, u_2, \ldots, u_k\}$ of strings in $\{0,1\}^m$. Each such set U defines a graph G_U on vertex set $\{0,1\}^m$. The edge $\{r,s\}$ is present in $E(G_U)$ if and only if there is a $u_i \in U$ such that $r = s \oplus u_i$, where $\oplus$ denotes bitwise-xor. Let $\Gamma_U(S)$ be the *neighborhood* of $S \subseteq V(G_U)$, i.e., all $r \in V(G_U) = \{0,1\}^m$ such that $r = s \oplus u_i$, for some $u_i \in U$ and $s \in S$.

Note first that, if $x \notin L$, then there is no U such that $\Gamma_U(S_x)$ is all of $V(G_U) = \{0,1\}^m$. Because the degree of each node in G_U is k, the total number of neighbors of S_x is $k \cdot |S_x|$. Because $x \notin L$, $k \cdot |S_x| \leq k \cdot 2^{m-n} = (k/2^n) \cdot 2^m$. Recall that $k = m/n + 1 = \text{poly}(n)$. Thus, $(k/2^n) < 1$, for all sufficiently large n, and $|\Gamma_U(S_x)| = (k/2^n) \cdot 2^m < 2^m = |V(G_U)|$.

We will use *the probabilistic method* to show that, if $x \in L$, there is a U such that $\Gamma_U(S_x)$ is all of $V(G_U) = \{0,1\}^m$. Consider $U = \{u_1, u_2, \ldots, u_k\}$ chosen uniformly at random from all k-element subsets of $\{0,1\}^m$. We wish to prove that, for such a randomly chosen U, the probability that $\Gamma_U(S_x) \neq \{0,1\}^m$ is less than 1. First, we compute the probability that an arbitrary $r \in \{0,1\}^m$ is not in $\Gamma_U(S_x)$. Because U was chosen uniformly at random from all k-element subsets of $\{0,1\}^m$, each u_i is a uniformly random m-bit string. This implies that, for fixed i, the set $S_i = \{s \oplus u_i \text{ s.t. } s \in S_x\}$ is distributed uniformly over all subsets of $\{0,1\}^m$ that have size $|S_x| \geq 2^m - 2^{m-n}$. The probability that $r \notin S_i$ is thus $(2^m - |S_x|)/2^m \leq (2^m - 2^m + 2^{m-n})/2^m = 2^{-n}$. The probability that r is not in $\Gamma_U(S_x)$ is the probability that it is not in S_i for any i, $1 \leq i \leq k$; this probability is at most 2^{-nk}, because the u_i are independent. By the union bound (see Appendix A.2 in your textbook), the probability that there is at least one r that is not in $\Gamma_U(S_x)$ is at most $2^{m-nk} = 2^{-n} < 1$.

The conclusions of the last two paragraphs give us the following Σ_2^P expression for membership in L:

$$x \in L \text{ if and only if } \exists \{u_1, u_2, \ldots, u_k\} \subset \{0,1\}^m \ \forall r \in \{0,1\}^m \ \vee_{i=1}^{k} M(x, r \oplus u_i) = 1.$$

□