Homework 17

Joe Baker, Brett Schreiber, Brian Knotten

February 20, 2018

27

Assume that our coin-flip Turing machine M doesn't require that ρ be efficiently computable.

Let $\rho = \frac{1}{\pi}$.

The complete binary representation of $\frac{1}{\pi}$ is infinitely long. Any finite representation is not completely accurate, since $\frac{1}{\pi}$ is irrational.

By the law of large numbers, since $\frac{1}{\pi}$ represented in binary is infinitely long with no repeated patterns, the probability that it must contain a substring of bits which is the encoding of a Turing machine that decides membership in the language $HALT = \{\langle M, x| \ M \text{ halts on } x\}$ in polynomial time goes to 1 as the number of bits in $\frac{1}{\pi}$ goes to infinity.

Since M generates a Turing machine to decide the halting problem as an intermediate step, removing the requirement that ρ be efficiently computable implies that M can decide an undecidable language in polynomial time.

28

Show that (NP $\cup$ co-NP) $\subseteq$ PP.

Proof:
We will show that Boolean Satisfiability $\in$ PP. Since Boolean Satisfiability is NP-complete, then any $L \in$ NP $\rightarrow L \in$ PP.
Let M be a probabilistic Turing machine which takes a Boolean formula F as input. Then does the following:

1. Pick a random assignment of variables for F
2. Verify if F is satisfied
3. If F is satisfied, then accept
4. If F is not satisfied, accept with $P(1/2)$ or reject with $P(1/2)$

First note that each step can clearly be done in polynomial time. P(M accepts) $= 1/2 + P$(random assignment is satisfied) and since P(random assignment is satisfied) > 0, then P(M accepts) $> 1/2$ and P(M rejects) $< 1/2$. Thus Boolean Satisfiability $\in$ PP since M is a Turing machine which meets the requirements for PP. So NP is contained in PP.

Since PP is closed under complement and contains NP, it must also contain co-NP. Finally, (NP $\cup$ co-NP) $\subseteq$ PP.